

UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE ESTATÍSTICA E INVESTIGAÇÃO OPERACIONAL



**MODELOS DE PROGRAMAÇÃO LINEAR INTEIRA
PARA PROBLEMAS DE DIMENSIONAMENTO
E ENGENHARIA DE TRÁFEGO
DE REDES DE TELECOMUNICAÇÕES**

Pedro Ferrão Patrício

DOUTORAMENTO EM ESTATÍSTICA E INVESTIGAÇÃO OPERACIONAL
Especialidade: Optimização

2009

UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE ESTATÍSTICA E INVESTIGAÇÃO OPERACIONAL



**MODELOS DE PROGRAMAÇÃO LINEAR INTEIRA
PARA PROBLEMAS DE DIMENSIONAMENTO
E ENGENHARIA DE TRÁFEGO
DE REDES DE TELECOMUNICAÇÕES**

Pedro Ferrão Patrício

Tese orientada por: Professor Doutor Luís Eduardo Neves Gouveia

DOUTORAMENTO EM ESTATÍSTICA E INVESTIGAÇÃO OPERACIONAL
Especialidade: Optimização

2009

Resumo e *Abstract*

Resumo

Na presente tese propomos modelos de Programação Linear Inteira para problemas de dimensionamento e engenharia de tráfego de redes de telecomunicações. Estes problemas têm como denominador comum a necessidade de estabelecer mais do que uma ligação entre pares de nodos origem/destino, de modo que a comunicação entre eles tenha qualidade de serviço (QoS) – medida através do atraso associado – e seja mantida na eventualidade de uma falha de equipamento. Nos dias de hoje, sobrevivência e QoS em redes de telecomunicações são cada vez mais fundamentais, dada a crescente dependência das nossas actividades em relação às mesmas redes.

Estudamos as propriedades dos modelos propostos em termos teóricos e testamos a sua eficiência através de instâncias geradas para o efeito. Em particular:

1. Analisamos diferentes classes de (sub)modelos que garantem a existência de D caminhos entre dois nodos;
2. Comparamos as formulações que se obtêm com base nesses submodelos no contexto de um problema de desenho de redes MPLS;
3. Analisamos o modo como diferentes mecanismos de sobrevivência influenciam o custo das soluções de um problema de desenho e dimensionamento de redes MPLS sobre redes WDM. Propomos ainda uma abordagem heurística para este problema, dada a dificuldade envolvida na determinação da solução óptima de instâncias de maior dimensão;

4. Estudamos duas classes de modelos – desagregados e agregados – para dois problemas de encaminhamento de tráfego definidos sobre as soluções referidas em 3.

Os resultados teóricos e computacionais discutidos na presente tese demonstram, segundo cremos, a importância de desenvolver bons modelos de Programação Linear Inteira.

Palavras chave: Programação Linear Inteira; Restrições de Salto; Reformulações; Telecomunicações.

Esta tese foi realizada com o apoio financeiro da Fundação para a Ciência e a Tecnologia (SFRH/ BD/ 28837/ 2006).

Abstract

In this thesis, we develop Integer Linear Programming models to address network dimensioning and traffic engineering problems in telecommunications networks. Given the ever increasing need for reliable and delay sensitive services, our main focus is on survivability and delay constraints.

We study the theoretical properties of the proposed models and we test their efficiency. In particular:

1. We analyse different classes of (sub)models which guarantee the existence of D paths between two nodes;
2. Based on those submodels, we compare the different formulations thus obtained for a MPLS network design problem;
3. In the context of a MPLS over WDM network dimensioning problem, we analyse the extent to which network costs are determined by different survivability mechanisms. We also develop a heuristic approach to this problem;
4. We study two classes of models – disaggregated and aggregated – to address two traffic engineering problems defined over the network solutions mentioned in 3.

The theoretical and computational results discussed in this thesis clearly demonstrate, we believe, the value of good problem formulations using Integer Linear Programming.

Keywords: Integer Linear Programming; Hop-Constraints; Reformulations; Telecommunications.

Dedicatória

Aos meus pais.

Agradecimentos

Poder ter desenvolvido e agora submeter este trabalho é um privilégio que devo ao Professor Doutor Luís Gouveia. Sem a sua paciente orientação, este momento não teria sido possível, assim como a felicidade de, nestes últimos anos, ter feito aquilo de que efectivamente gosto. Um especial e sentido muito obrigado.

Ao Professor Doutor Amaro de Sousa, a minha gratidão pelas úteis sugestões e pela disponibilidade relativamente aos aspectos tecnológicos deste trabalho.

Ao Departamento de Matemática da Universidade da Beira Interior e também aos meus colegas de trabalho agradeço o incentivo recebido e a dispensa de serviço docente nos últimos meses da preparação desta tese.

Aos meus Pais, pela formação humana que me deram e pelas oportunidades que me proporcionaram a todos os níveis. Pelo seu carinho e amor... e por nos últimos tempos terem compreendido a minha ausência.

À minha Avó, pela força que sempre me transmitiu. À Tia Fernanda, pelas constantes mensagens de apoio.

À Nagui, por consistentemente adivinhar o meu estado de espírito a 300km de distância. E por aquele telefonema.

À Mónica, pela infinita paciência e compreensão. Pelas palavras certas.

A todos os meus familiares e amigos, pelo encorajamento e confiança transmitidos. Finalmente, ao Mac, para quem o caminho até à taça de Friskies com menor número de saltos nunca teve dificuldade.

Conteúdo

Resumo e <i>Abstract</i>	iii
Resumo	iii
<i>Abstract</i>	v
Dedicatória	vii
Agradecimentos	ix
I Introdução	1
I.1 Determinação de um Caminho com Restrição de Salto	2
I.1.1 Modelo Genérico	4
I.1.2 Modelo Tradicional	4
I.1.3 Modelo Hop-Indexed	7
I.1.4 Comparação de Modelos	15
I.2 Determinação de D Caminhos com Restrições de Salto	17
I.2.1 Modelo Genérico	17
I.2.2 Modelo Tradicional	18
I.2.3 Modelo Hop-Indexed	21
I.2.4 Modelo Hop-Indexed Agregado	24
I.2.5 Comparação de Modelos	28
I.3 Conclusões	31
II Desenho e Dimensionamento de Redes de Telecomunicações	33
II.1 Desenho de uma rede MPLS	37

II.1.1	Definição do Problema	38
II.1.2	Formulação Genérica	39
II.1.3	Formulações TRAD-MCF, HOP-MCF e HOP-A-MCF	42
II.1.4	Eliminação de Variáveis	45
II.1.5	Resultados Computacionais	48
II.2	Desenho e Dimensionamento de uma rede MPLS sobre uma rede WDM	57
II.2.1	Descrição do Problema	58
II.2.2	Formulação HOP-A-MCF-II	65
II.2.3	Resultados Computacionais	71
II.2.4	Heurística de Duas Fases	80
II.2.5	Resultados Computacionais	83
II.3	Conclusões	88
III	Encaminhamento Ótimo de Tráfego	89
III.1	Definição dos Problemas	92
III.2	Modelos Desagregados	95
III.2.1	Funções Objectivo Baseadas em Variáveis Associadas a Arcos	96
III.2.1.1	Minimização do Número Médio de Saltos	97
III.2.1.2	Minimização do Número Máximo de Saltos	97
III.2.2	Funções Objectivo Baseadas em Variáveis Associadas a Lacetes	98
III.2.2.1	Minimização do Número Médio de Saltos	99
III.2.2.2	Minimização do Número Máximo de Saltos	100
III.3	Modelos Agregados	101
III.3.1	Funções Objectivo Baseadas em Variáveis Associadas a Arcos	102
III.3.1.1	Minimização do Número Médio de Saltos	103
III.3.1.2	Minimização do Número Máximo de Saltos	104
III.3.2	Funções Objectivo Baseadas em Variáveis Associadas a Lacetes	106
III.3.2.1	Minimização do Número Médio de Saltos	106
III.3.2.2	Minimização do Número Máximo de Saltos	107
III.4	Comparação das Duas Classes de Modelos	109
III.4.1	Minimização do Número Médio de Saltos	109

III.4.2 Minimização do Número Máximo de Saltos	113
III.5 Resultados Computacionais	116
III.5.1 Minimização do Número Médio de Saltos	118
III.5.2 Minimização do Número Máximo de Saltos	124
III.5.3 Impacto da Variação do Tráfego Original	129
III.6 Conclusões	137
IV Trabalho Futuro	139
Bibliografia	141

Lista de Tabelas

II.1	Resultados computacionais das instâncias com 25 nodos, considerando as formulações <i>TRAD-MCF</i> , <i>HOP-MCF</i> e <i>HOP-A-MCF</i> sem eliminação de variáveis.	51
II.2	Resultados computacionais das instâncias com 50 nodos, considerando as formulações <i>TRAD-MCF</i> , <i>HOP-MCF</i> e <i>HOP-A-MCF</i> sem eliminação de variáveis.	52
II.3	Resultados computacionais das instâncias com 25 nodos, considerando as formulações <i>TRAD-MCF</i> , <i>HOP-MCF</i> e <i>HOP-A-MCF</i> com eliminação de variáveis.	53
II.4	Resultados computacionais das instâncias com 50 nodos, considerando as formulações <i>TRAD-MCF</i> , <i>HOP-MCF</i> e <i>HOP-A-MCF</i> com eliminação de variáveis.	54
II.5	Resultados computacionais relativos à rede NSFNet, obtidos através da formulação <i>HOP-A-MCF-II</i>	74
II.6	Resultados computacionais relativos à rede EON, obtidos através da formulação <i>HOP-A-MCF-II</i>	78
II.7	Resultados computacionais da Fase I.	84
II.8	Resultados computacionais relativos à Fase II.	85
II.9	Comparação dos valores das soluções obtidas através da heurística de duas fases com os das obtidas com o modelo <i>HOP-A-MCF-II</i>	86
III.1	Conversão dos valores óptimos das funções objectivo dos modelos para a minimização do número médio de saltos nos reais valores óptimos. .	118

III.2	Resultados computacionais dos modelos D - <i>Média-Arcos_L</i> , D - <i>Média-Lacetes_L</i> , <i>A-Média-Arcos_L</i> e <i>A-Média-Lacetes_L</i> sob o mecanismo <i>Path Diversity</i> ($\Delta = D$).	119
III.3	Resultados computacionais dos modelos D - <i>Média-Arcos_L</i> , D - <i>Média-Lacetes_L</i> , <i>A-Média-Arcos_L</i> e <i>A-Média-Lacetes_L</i> sob o mecanismo <i>Path Protec-</i> <i>tion</i> ($\Delta = D - 1$).	120
III.4	Resultados computacionais do modelo <i>A-Média-Lacetes</i> , para $ S = 6$ e $ S = 8$, sob <i>Path Diversity</i> ($\Delta = D$),	122
III.5	Resultados computacionais do modelo <i>A-Média-Lacetes</i> , para $ S = 6$ e $ S = 8$, sob <i>Path Protection</i> ($\Delta = D - 1$),	123
III.6	Resultados computacionais dos modelos D - <i>Máx-Arcos_L</i> , D - <i>Máx-Lacetes_L</i> , <i>SD-Máx-Arcos_L</i> , <i>A-Máx-Arcos_L</i> e <i>A-Máx-Lacetes_L</i> sob o mecanismo <i>Path Diversity</i> ($\Delta = D$).	125
III.7	Resultados computacionais dos modelos D - <i>Máx-Arcos_L</i> , D - <i>Máx-Lacetes_L</i> , <i>SD-Máx-Arcos_L</i> , <i>A-Máx-Arcos_L</i> e <i>A-Máx-Lacetes_L</i> sob o mecanismo <i>Path Protection</i> ($\Delta = D - 1$).	125
III.8	Resultados computacionais do modelo <i>A-Máx-Arcos</i> , para $ S = 6$ e $ S = 8$, sob <i>Path Diversity</i> ($\Delta = D$),	127
III.9	Resultados computacionais do modelo <i>A-Máx-Arcos</i> , para $ S = 6$ e $ S = 8$, sob <i>Path Protection</i> ($\Delta = D - 1$),	128
III.10	Impacto do volume de tráfego total no “custo” das soluções obtidas sob <i>Path Diversity</i>	133
III.11	Impacto do volume de tráfego total no “custo” das soluções obtidas sob <i>Path Protection</i>	134
III.12	Número de instâncias não admissíveis sob cada mecanismo de sobre- vivência, bem como para cada par de valores de ϵ e de D	135

Lista de Figuras

I.1	Um caminho que não cumpre com a restrição de salto com $H = 4$ e um <i>caminho-4</i> -(p, q)	3
I.2	Uma solução fraccionária de $TRAD_L$, considerando $H = 4$	6
I.3	Grafo original $G = (V, A)$, em cima, e o correspondente grafo expandido $G_{pq} = (V_{pq}, A_{pq})$, considerando $H = 3$	10
I.4	Dois <i>trajectos-3</i> -(p, q) sobre $G = (V, A)$ e os correspondentes caminhos em $G_{pq} = (V_{pq}, A_{pq})$	11
I.5	Caminho alternativo ao representado a tracejado na figura I.4, em termos do grafo original G , bem como do grafo expandido G_{pq}	15
I.6	Três ($D = 3$) <i>caminhos-4</i> -(p, q)	18
I.7	Valores das variáveis do modelo $TRAD^D$ relativamente aos três <i>caminhos-4</i> -(p, q) ilustrados na figura I.6	20
I.8	Três ($D = 3$) <i>trajectos-3</i> -(p, q) sobre o grafo original G e os correspondentes caminhos sobre os grafos expandidos G_{pq}^1, G_{pq}^2 e G_{pq}^3 , sobrepostos	25
I.9	Agregação de fluxos em G e na sobreposição de G_{pq}^1, G_{pq}^2 e G_{pq}^3 (ou $G_{pq}^{1,2,3}$)	26
II.1	Uma rede MPLS	37
II.2	Uma solução admissível para o problema de desenho da rede MPLS, considerando $D = 2$ e $H = 3$ e tendo por base o exemplo ilustrado na figura II.1	40
II.3	Ilustração do processo de eliminação das variáveis das formulações $TRAD-MCF, HOP-MCF$ e $HOP-A-MCF$	46
II.4	Instância N25-3	49
II.5	Solução fraccionária de $TRAD-MCF_L$, considerando $H = 2$	55

II.6	Um <i>label switched path</i> (LSP) da rede MPLS utilizando <i>lightpaths</i> da rede WDM	58
II.7	Quantidade de tráfego que atravessa cada arco sob <i>Path Diversity</i> e sob <i>Path Protection</i>	62
II.8	Ilustração do facto de se poderem obter redes de menor custo à medida que D aumenta, sob o mecanismo <i>Path Protection</i>	63
II.9	Modelação das restrições de atenuação óptica: o grafo original G e o grafo expandido G' obtido considerando $L = 2000$	66
II.10	O grafo expandido (G') da rede NSFNet com $ S = 6$ e $ S = 8$	72
II.11	Relação entre os custos das soluções obtidas sob cada mecanismo de sobrevivência e sob nenhuma protecção (caso $D = 1$), para $ S = 6$	75
II.12	Relação entre os custos das soluções obtidas sob cada mecanismo de sobrevivência e sob nenhuma protecção (caso $D = 1$), para $ S = 8$	75
II.13	O grafo original (G) representando a rede EON, com $ S = 8$	78
II.14	Heurística de Duas Fases: obtenção do grafo $G'' = (V', E'')$	81
III.1	Exemplo ilustrativo dos problemas de <i>traffic engineering</i> : Minimização do número médio e máximo de saltos	93
III.2	Exemplo de uma rede $N = (X, U)$, solução do problema de dimensionamento	117
III.3	Ilustração do cálculo dos limites “topológicos”	130

Capítulo I

Introdução

Neste capítulo modelamos em Programação Linear Inteira as restrições comuns aos problemas que serão apresentados nos capítulos II e III.

Na secção I.1, começamos por discutir modelos para a determinação de um único caminho entre dois nodos com restrição de salto num determinado grafo. Recorrentemente, em áreas como telecomunicações, transporte, entre outras, não basta encontrar um caminho entre um par de nodos origem/destino (servidores, cidades, *etc*). É também necessário que esse caminho não contenha mais do que um determinado número de arestas/arcs (saltos). No caso particular de problemas na área das telecomunicações, esta condição está associada ao atraso que a informação veiculada ao longo desse caminho sofre ao atravessar os servidores e/ou ligações intermédios. Os modelos propostos na secção I.1 diferenciam-se em termos das respectivas relaxações lineares, pelo que fornecem limites inferiores (ou superiores) para os valores óptimos dos problemas nos quais serão inseridos, em regra, diferentes.

Na secção I.2, a nossa atenção recairá sobre modelos que especifiquem D caminhos entre dois nodos, cada um dos quais com restrições de salto. A motivação reside no facto de que em redes de telecomunicações se pretende usualmente que estas sejam resistentes a falhas, falhas estas que podem ocorrer num dos seus elementos (nodos e/ou arestas), pelo que deverá haver dois ou mais caminhos disjuntos (nos nodos ou arestas) entre uma origem e um destino. Como veremos, os modelos apresentados nesta secção não garantem que esses D caminhos são disjuntos, o que no entanto

é simples de garantir através da inclusão de restrições adicionais (ver capítulos II e III). Como na secção anterior, os modelos propostos distinguem-se em termos das respectivas relaxações lineares. A qualidade dos respectivos limites inferiores (ou superiores), relativamente aos valores óptimos dos problemas nos quais serão inseridos, será atestada no capítulo II.

I.1 Determinação de um Caminho com Restrição de Salto

Seja $G = (V, A)$ um grafo orientado, em que V denota o conjunto de nodos e A identifica o conjunto de arcos. Considerando dois nodos $p, q \in V$, pretende-se determinar um caminho entre p e q que não contenha mais do que um determinado número, H , de arcos (saltos).

Note-se que este número máximo de saltos poderia ser dado em termos do número de arestas se considerássemos à partida um grafo não orientado. No entanto, com base no trabalho de Maganti e Wolsey [27], entre outros, sabe-se que é possível obter melhores modelos (*i.e.*, modelos mais compactos e/ou cuja relaxação linear forneça um melhor limite superior ou inferior) ao formular o problema num grafo orientado. De facto, note-se que o grafo orientado $G = (V, A)$ que começámos por considerar pode resultar da orientação de um grafo não orientado $G = (V, E)$, bastando para tal substituir cada aresta $\{i, j\} \in E$ por dois arcos (i, j) e (j, i) .

Com o intuito de apresentar nesta secção modelos de Programação Linear Inteira para a determinação de um caminho entre p e q com restrição de salto, consideremos as definições I.1.1, I.1.2 e I.1.3, que estabelecem os seguintes conceitos:

Definição I.1.1 *Um trajecto- (p, q) é uma sequência de arcos $\{(i_1, j_1), \dots, (i_k, j_k)\}$ tal que $i_1 = p$, $j_k = q$ e $j_m = i_{m+1}$ para todo o $m = 1, \dots, k - 1$.*

Definição I.1.2 *Um caminho- (p, q) é um trajecto- (p, q) em que não há repetição de nodos.*

Definição I.1.3 Um caminho- H -(p, q) é um caminho-(p, q) com no máximo H arcos.

A figura I.1 ilustra um caminho de p a q que não cumpre com a restrição de salto considerando $H = 4$, bem como um exemplo de um caminho-4-(p, q).

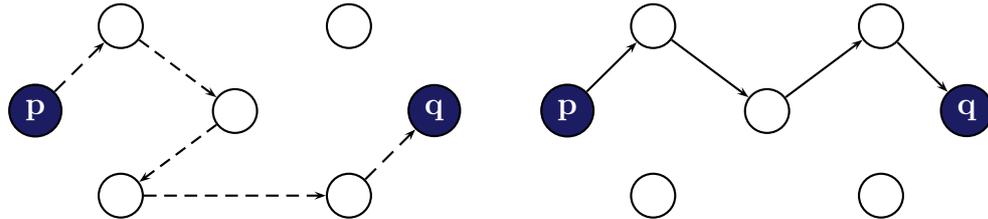


Figura I.1: Um caminho que não cumpre com a restrição de salto com $H = 4$ (à esquerda) e um caminho-4-(p, q) (à direita).

Assim, por outras palavras, para um dado $H \in \mathbb{N}$ e dois nodos $p, q \in V$, nesta secção apresentamos modelos para a determinação de um caminho- H -(p, q) sobre o grafo $G = (V, A)$. Importa no entanto fazer desde já a seguinte observação. Nenhum dos modelos que serão apresentados modela exactamente um caminho- H -(p, q). Na realidade, a solução de cada modelo consiste num conjunto de arcos que contém um caminho- H -(p, q). No entanto, havendo custos associados aos arcos e/ou na presença de outras restrições nos problemas em que esses (sub)modelos forem inseridos (como acontece nos capítulos II e III), as soluções obtidas são de facto caminhos- H -(p, q). Conforme será explicado com maior detalhe mais à frente, é bastante difícil descrever o envolvente convexo do conjunto de vectores de incidência de um caminho- H -(p, q) pois o problema de optimização correspondente é *NP-difícil*. Assim, ao invés de tentarmos encontrar uma boa formulação para a determinação de um caminho- H -(p, q), a nossa atenção recairá sobre boas formulações para um conjunto de restrições mais simples, *i.e.*, preocupar-nos-emos em determinar um conjunto de arcos que “apenas” contenha um caminho- H -(p, q).

Neste sentido, nas próximas subsecções, começamos por introduzir um modelo genérico em que a modelação de um conjunto de arcos que contém um caminho- H -(p, q) é escrita através de uma restrição genérica (*i.e.*, não é descrita matematicamente). Com base neste modelo, consideramos uma descrição tradicional dessa

restrição genérica proposta por Balakrishnan e Altinkemer [2]. Seguidamente, com base no trabalho de Gouveia no contexto do Problema da Árvore de Suporte/Steiner de Custo Mínimo com Restrições de Salto [11], apresentamos uma descrição alternativa cuja relaxação linear possui pontos extremos inteiros.

I.1.1 Modelo Genérico

Começamos então por apresentar um modelo genérico (*GEN*) que nos permitirá determinar um *caminho-H*-(p, q). Consideremos um conjunto de variáveis binárias definidas da seguinte forma:

$$y_{ij} = \begin{cases} 1, & \text{se o arco } (i, j) \text{ pertence à solução} \\ 0, & \text{c. c.} \end{cases}, (i, j) \in A.$$

Deste modo, podemos então escrever:

Modelo GEN

$$\{(i, j) \in A : y_{ij} = 1\} \text{ contém um } \textit{caminho-H}-(p, q) \tag{1}$$

$$y_{ij} \in \{0, 1\}, (i, j) \in A. \tag{2}$$

A restrição genérica **(1)** garante que os arcos pertencentes à solução contêm um *caminho-H*-(p, q) e será explicitada matematicamente nas subsecções seguintes. As restrições **(2)** definem o domínio das variáveis do modelo. Note-se que não se consideram as variáveis $y_{qj}, \forall (q, j) \in A$, nem as variáveis $y_{ip}, \forall (i, p) \in A$, visto que estas variáveis indicam se um arco chega ao nodo inicial p ou se um arco parte do nodo final q . Desta forma, essas variáveis são simplesmente omitidas no modelo.

I.1.2 Modelo Tradicional

De acordo com Ahuja Magnanti e Orlin [1], bem como Balakrishnan e Altinkemer [2], podemos descrever a restrição **(1)** do modelo *GEN* utilizando apenas as variáveis

$y_{ij}((i, j) \in A)$, da seguinte forma:

$$\sum_{j:(p,j) \in A} y_{pj} = 1 \quad (3a)$$

$$\sum_{j:(i,j) \in A} y_{ij} - \sum_{j:(j,i) \in A} y_{ji} = 0, i \neq p, q \quad (3b)$$

$$\sum_{j:(j,q) \in A} y_{jq} = 1 \quad (3c)$$

$$\sum_{(i,j) \in A} y_{ij} \leq H. \quad (3d)$$

As restrições **(3a)**-**(3c)** são as usuais restrições de caminho, especificando um caminho de p para q . Realce-se contudo que estas restrições definem um conjunto de arcos que contém um caminho, permitindo a repetição de nodos e/ou arcos assim como a existência de circuitos desconexos do caminho de p a q . Para garantirmos que, de facto, esse caminho tem no máximo H arcos, será necessário considerar a restrição de cardinalidade adicional **(3d)**. Assim, substituindo a restrição **(1)** do modelo *GEN* pelas do sistema **(3a)**-**(3d)**, obtemos o modelo doravante designado por *TRAD* (modelo tradicional):

Modelo TRAD

$$\begin{aligned} & \mathbf{(3a)}, \mathbf{(3b)}, \mathbf{(3c)}, \mathbf{(3d)} \\ & y_{ij} \in \{0, 1\}, (i, j) \in A. \end{aligned} \quad (2)$$

Este modelo é válido para a determinação de um *caminho- H -(p, q)* pois, como referimos antes, na presença de custos associados aos arcos e/ou outras restrições nas formulações em que este modelo for inserido, a sua solução consiste de facto num *caminho- H -(p, q)*.

É bem conhecido o facto de que os pontos extremos da relaxação linear das restrições **(3a)**-**(3c)** são inteiros uma vez que a matriz associada é totalmente unimodular. Esta propriedade perde-se, no entanto, na presença da restrição de cardinalidade adicional **(3d)**, ou seja, a relaxação linear de *TRAD*, *TRAD_L*, poderá ter pontos extremos fraccionários. De forma a ilustrar este facto, consideremos a figura I.2, que

ilustra um exemplo de um grafo $G = (V, A)$ em que se considera $H = 4$ bem como custos associados a cada arco $(i, j) \in A$ (representados junto a cada arco do grafo do lado esquerdo). Note-se que o custo da única solução inteira admissível é 7. No lado direito da figura, assumindo que se pretende determinar o $\text{caminho-4-}(p, q)$ de menor custo, representamos a solução óptima de $TRAD_L$: junto a cada arco $(i, j) \in A$ encontra-se o valor óptimo da respectiva variável y_{ij} . Note-se que $\sum_{(i,j) \in A} y_{ij} \leq 4 = H$ e que o custo da solução é 6.

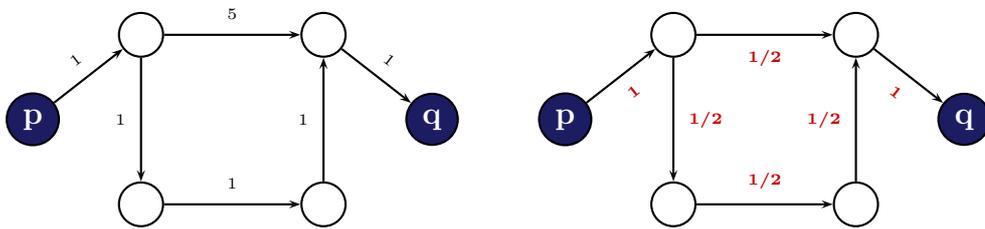


Figura I.2: Uma solução fraccionária de $TRAD_L$, considerando $H = 4$. À esquerda, os valores junto a cada aresta $\{i, j\}$ representam o seu custo ($c_{\{i,j\}}$); à direita, os valores junto a cada arco (i, j) representam o valor da respectiva variável y_{ij} .

Importa agora referir que Dahl e Gouveia, em [9], apresentaram uma descrição linear do envolvente convexo dos vectores de incidência de $\text{caminhos-}H\text{-}(p, q)$, com $H = 2$ e $H = 3$, usando apenas as variáveis y_{ij} acima definidas. Para $H > 3$, os mesmos autores determinaram um conjunto de desigualdades que definem facetas, ficando uma descrição completa do referido envolvente por determinar. De facto, tal descrição no espaço das variáveis y_{ij} para todos os valores de H é uma tarefa difícil: o problema de optimização correspondente é *NP-difícil* para custos gerais associados aos arcos. Note-se aliás que, considerando custos negativos, o problema é equivalente ao problema do caminho Hamiltoniano de custo mínimo de p a q . Na realidade, encontrar uma descrição completa do envolvente convexo dos vectores de incidência de um $\text{caminho-}H\text{-}(p, q)$ utilizando apenas as variáveis y_{ij} é já uma tarefa bastante complicada para (tão só) $H = 4$.

Por esta razão, na próxima subsecção apresentamos um modelo para um $\text{trajecto-}H\text{-}(p, q)$, que apesar de ser uma relaxação de um $\text{caminho-}H\text{-}(p, q)$ é simultaneamente um fortalecimento da restrição genérica (1) do modelo genérico.

I.1.3 Modelo Hop-Indexed

Nesta subsecção apresentamos uma descrição alternativa da restrição genérica (1) do modelo *GEN*. Designaremos o modelo resultante por *HOP*, uma vez que utilizaremos um novo conjunto de variáveis associadas aos arcos $(i, j) \in A$ com um índice adicional indicando a posição (salto/*hop*) em que cada arco é percorrido.

Começamos então por considerar o fortalecimento da restrição (1) do modelo genérico, dado por:

$$\{(i, j) \in A : y_{ij} = 1\} \text{ é um } \textit{trajecto-H-(p, q)}. \quad (4)$$

Note-se, em primeiro lugar, que esta restrição é de facto um fortalecimento de (1) visto que num conjunto de arcos que contenha um *caminho-H-(p, q)* poderá haver arcos que formem circuitos desconexos. Por outro lado, note-se que os elementos definidos por (4) são relaxações de um *caminho-H-(p, q)* no sentido em que a repetição de arcos (e de nodos) é permitida. A razão pela qual consideramos esta relaxação prende-se com os factos de que: como mencionado anteriormente, ser bastante difícil descrever o envolvente convexo de *caminhos-H-(p, q)*; bem como, veremos mais à frente nesta subsecção, haver uma relação entre um *trajecto-H-(p, q)* e um caminho (não restringido no número de arcos) num grafo expandido. Este último facto sugere que encontrar uma descrição do envolvente convexo dos vectores de incidência de *trajectos-H-(p, q)* deverá ser significativamente mais fácil que encontrar o de *caminhos-H-(p, q)*. Note-se, no entanto, que os dois envolventes são iguais para $H < 4$ uma vez que nestes casos *trajectos* são *caminhos*. Para $H = 4$, Dahl, Foldnes e Gouveia [8] concluíram que a descrição do envolvente convexo dos vectores de incidência de *trajectos-H-(p, q)* no espaço das variáveis y_{ij} requer um número exponencial de desigualdades (apesar de, como mencionado anteriormente, ser mais simples do que a descrição do envolvente convexo relativo a *caminhos-H-(p, q)*), pelo que se torna evidente a necessidade de recorrer a modelos estendidos.

Apesar de numa primeira análise não parecer vantajoso pensar na descrição do envolvente convexo dos vectores de incidência de *trajectos- H -(p, q)*, considere-se um custo c_{ij} associado a cada arco $(i, j) \in A$ e assumase que se pretende determinar o *trajecto- H -(p, q)* de custo mínimo em G . Este problema pode ser modelado como um problema de caminho mais curto não restringido no número de arcos num grafo expandido acíclico, G_{pq} . Este grafo é constituído por níveis, sendo cada nodo $i \in V \setminus \{p\}$ do grafo original replicado $H - 1$ vezes, uma por cada posição que esse nodo poderá ocupar num *trajecto- H -(p, q)* em G . Usando uma formulação “tradicional” de fluxos para o problema do caminho mais curto em G_{pq} , cuja relaxação linear tem pontos extremos inteiros, conseguiremos escrever uma formulação exacta para o problema sobre o grafo original e, assim, descrever o envolvente convexo dos vectores de incidência de *trajectos- H -(p, q)*. A formulação sobre G_{pq} utiliza variáveis associadas aos arcos deste grafo, variáveis estas que são diferentes das variáveis y_{ij} utilizadas antes. Assim, essa formulação é uma descrição estendida do envolvente convexo dos vectores de incidência de *trajectos- H -(p, q)* e, uma vez que a dimensão do grafo expandido depende polinomialmente da dimensão do grafo original, é também uma formulação compacta. Esta descrição estendida pode então ser combinada com o modelo *GEN*, pelo que obtemos desse modo uma alternativa ao modelo *TRAD*. No que se segue, adoptamos as ideias de Gouveia aplicadas no contexto do Problema da Árvore de Suporte/Steiner de Custo Mínimo com Restrições de Salto [11]. O enquadramento da reformulação apresentada nesse trabalho foi proposto por Martin [29] (para uma abordagem mais geral, ver também [30]).

Como mencionado atrás, G_{pq} tem $H - 1$ níveis, em cada um dos quais existe uma cópia de cada nodo $i \in V \setminus \{p\}$ do grafo original G (uma por cada posição que esse nodo poderá ocupar num *trajecto- H -(p, q)* em G). No total, G_{pq} é constituído por $H + 1$ níveis: no nível zero existe apenas uma cópia do nodo origem p ; no nível H existe também apenas uma cópia do nodo destino q ; e, nos restantes $H - 1$ níveis, existem as referidas cópias de cada nodo $i \in V \setminus \{p\}$. Desta forma, sendo V_{pq} o conjunto de nodos de G_{pq} , temos:

$$V_{pq} = \{(p, 0)\} \cup \{(i, h) : i \in V \setminus \{p\} \wedge 1 \leq h \leq H - 1\} \cup \{(q, H)\}.$$

E o conjunto dos arcos de G_{pq} , A_{pq} , é definido da seguinte forma:

$$A_{pq} = \{((i, h), (j, h + 1)) : (i = p \wedge h = 0) \vee \\ \vee (i, j \in V \setminus \{p\} \wedge 1 \leq h \leq H - 2) \vee (j = q \wedge h = H - 1)\}.$$

Para cada $i \neq j \in V$, o custo de cada arco $((i, h), (j, h + 1)) \in A_{pq}$ é dado pelo custo c_{ij} associado ao arco $(i, j) \in A$ do grafo original G . Se $i = j$ o respectivo custo é nulo para qualquer h , pois corresponde a situações em que se efectua um lacete sobre o nodo $i \in V$ (como será ilustrado mais à frente).

Na figura I.3 apresentamos um exemplo de um grafo original $G = (V, A)$ e, considerando $H = 3$, o correspondente grafo expandido $G_{pq} = (V_{pq}, A_{pq})$. Para ambos os grafos, os custos associados a cada arco encontram-se representados junto dos mesmos. A figura I.4 ilustra dois *trajectos-3*-(p, q) em $G = (V, A)$ e os correspondentes caminhos (não restringidos no número de saltos) em $G_{pq} = (V_{pq}, A_{pq})$. Na mesma figura, note-se que ambos os caminhos efectuem 3 saltos de $(p, 0)$ a $(q, 3)$ em G_{pq} , no entanto, o trajecto correspondente ao caminho a tracejado efectua apenas 2 saltos de p a q , tendo em conta arcos originais de A , e um lacete no nodo q (correspondente ao arco de $(q, 2)$ a $(q, 3)$ em G_{pq}).

Uma formulação para o problema do caminho mais curto não restringido no número de saltos sobre G_{pq} pode ser obtida facilmente associando variáveis binárias a cada arco $((i, h - 1), (j, h)) \in A_{pq}$. Estas variáveis definem-se da seguinte forma:

$$z_{ij}^h = \begin{cases} 1, & \text{se o arco } ((i, h - 1), (j, h)) \text{ pertence à solução} \\ 0, & \text{c. c.} \end{cases}.$$

Podemos então escrever as usuais restrições de conservação de fluxo (ver [11]) para o problema do caminho mais curto não restringido no número de saltos de $(p, 0)$ a (q, H) sobre o grafo G_{pq} , obtendo assim uma formulação compacta (note-se que a dimensão de G_{pq} depende da dimensão de G e do parâmetro H polinomialmente).

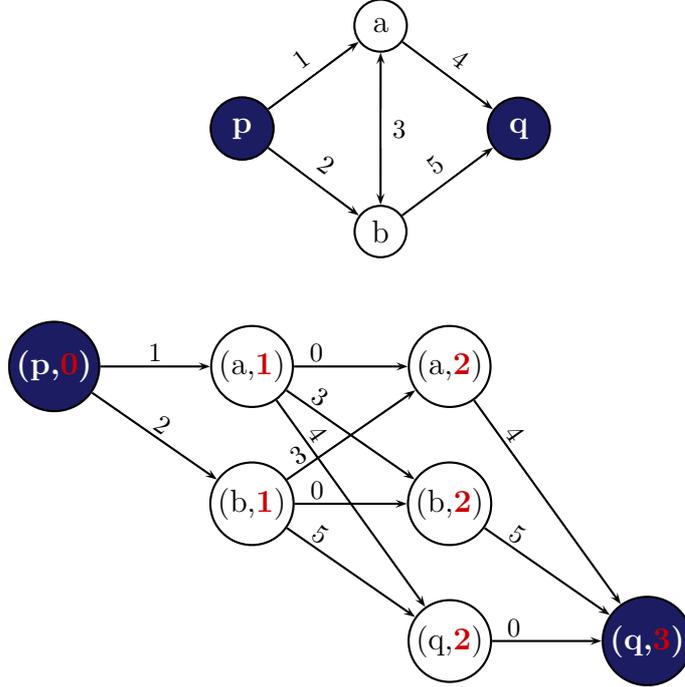


Figura I.3: Grafo original $G = (V, A)$, em cima, e o correspondente grafo expandido $G_{pq} = (V_{pq}, A_{pq})$, considerando $H = 3$. Junto a cada arco encontra-se o custo associado.

Repare-se, agora, que dada a correspondência existente entre um caminho de $(p, 0)$ a (q, H) em G_{pq} e um *trajecto- H* - (p, q) no grafo original G , podemos interpretar as variáveis z_{ij}^h como indicando se o arco (i, j) é percorrido, no máximo (devido à possível utilização de arcos entre (i, h) e $(i, h+1)$ em G_{pq} , para um $i \in V \setminus \{p\}$ e $h = 1, \dots, H-1$, arcos esses que correspondem a lacetes sobre i em G), na posição h no *trajecto- H* - (p, q) . Note-se que a construção do grafo expandido garante que quando $i = p$ então $h = 1$, *i.e.*, um arco que saia do nodo p está sempre na primeira posição. Além disso, as variáveis z_{qq}^h , com $h = 2, \dots, H$, associadas a lacetes, também são consideradas. Desta forma, podemos escrever uma formulação estendida para a determinação de um *trajecto- H* - (p, q) sobre G da seguinte forma:

$$\sum_{j:(p,j) \in A} z_{pj}^1 = 1 \quad (5a)$$

$$\sum_{j:(i,j) \in A} z_{ij}^{h+1} - \sum_{j:(j,i) \in A} z_{ji}^h = 0, \quad i \neq p; h = 1, \dots, H-1 \quad (5b)$$

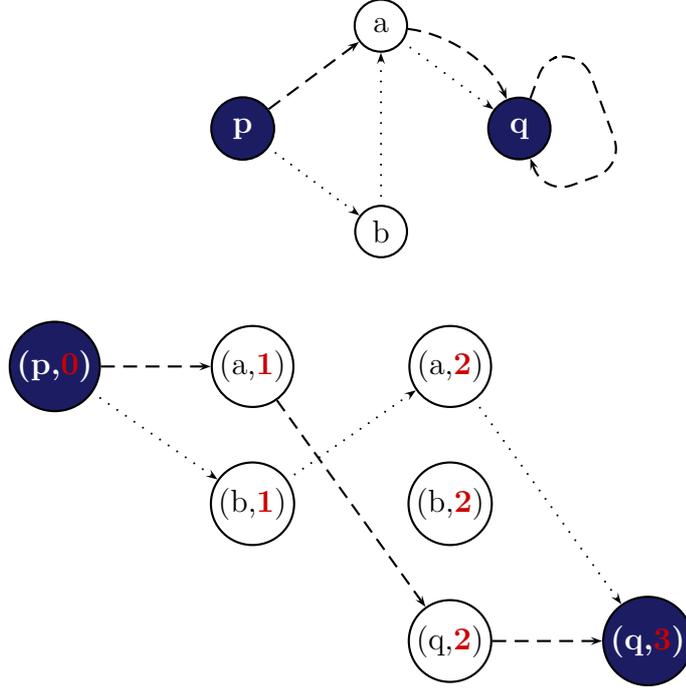


Figura I.4: Dois *trajectos-3*-(p, q) sobre $G = (V, A)$, em cima, e os correspondentes caminhos (não restringidos no número de saltos) em $G_{pq} = (V_{pq}, A_{pq})$.

$$\sum_{j:(j,q) \in A} z_{jq}^H = 1 \quad (5c)$$

$$z_{ij}^h \in \{0, 1\}, \quad (i, j) \in A; h = 1, \dots, H \quad (6a)$$

$$z_{ii}^h \in \{0, 1\}, \quad i \in V \setminus \{p, q\}; h = 2, \dots, H - 1 \quad (6b)$$

$$z_{qq}^h \in \{0, 1\}, \quad h = 2, \dots, H. \quad (6c)$$

A restrição **(5a)** estabelece que existe um arco com origem em p na posição $h = 1$. As restrições **(5b)** garantem que um arco chega ao nodo i na posição h se e só se um arco sai do mesmo nodo na posição $h + 1$, para qualquer $i \neq p$ e $h = 1, \dots, H - 1$. A restrição **(5c)** garante que existe um arco a chegar ao nodo q na posição máxima admissível, H . Finalmente, as restantes restrições **(6a)**-**(6c)** definem o domínio das variáveis.

O resultado seguinte advém da forma como construímos o grafo G_{pq} , bem como da correspondência entre um caminho não restringido no número de saltos entre $(p, 0)$

a (q, H) em G_{pq} e um *trajecto- H* -(p, q) no grafo original G , e, ainda, do facto de que o modelo acima apresentado ser um modelo de fluxos sobre o grafo expandido G_{pq} (a matriz das restrições é totalmente unimodular).

Resultado I.1

Os pontos extremos do polítopo definido pela relaxação linear das restrições (5a)-(6c) são inteiros.

Deste modo, podemos afirmar que quaisquer que sejam os custos c_{ij} a solução óptima de um problema sujeito a (5a)-(6c) não se altera se relaxarmos a integralidade das variáveis. Inclusivamente, podemos substituir as restrições (6a)-(6c) por restrições de não negatividade, que denotaremos por (7a)-(7c), na medida em que as restrições (5a)-(5c) tornam os limites superiores das variáveis redundantes.

Observe-se que as novas variáveis z_{ij}^h relacionam-se com as variáveis y_{ij} , definidas na apresentação do modelo *GEN*, através das igualdades (8), pois, de facto, um arco (i, j) pertence à solução se e só se for percorrido numa qualquer posição h :

$$y_{ij} = \sum_{h=1, \dots, H} z_{ij}^h, (i, j) \in A. \quad (8)$$

Pelo resultado I.1, denotando o envolvente convexo dos vectores de incidência de *trajecto- H* -(p, q) por $\text{conv}(W_{pq})$, podemos afirmar que a projecção do polítopo definido por (5a)-(5c) e (7a)-(7c) no espaço das variáveis y_{ij} através das igualdades (8) é dado por $\text{conv}(W_{pq})$, ou seja:

Resultado I.2 (Gouveia 1998)

$$\{y_{ij} : z_{ij}^h \text{ satisfaz (5a)-(5c), (7a)-(7c) e (8)}\} = \text{conv}(W_{pq}).$$

Demonstração: A inclusão “ \supseteq ” resulta da definição de envolvente convexo de um conjunto, bem como do facto das restrições (5a)-(5c), (7a)-(7c) e (8) constituírem um modelo válido para a determinação de um *trajecto- H* -(p, q).

Para mostrar que “ \subseteq ” também se verifica, considere-se uma solução $\{y_{ij}^*\}$, ponto

extremo do primeiro conjunto mas não pertencente a $\text{conv}(W_{pq})$. Note-se que, relativamente a essa solução, não existe nenhuma solução $\{z_{ij}^h\}$ que, satisfazendo as condições (8), seja ponto extremo do primeiro conjunto: caso isso acontecesse a solução $\{y_{ij}\}$ seria inteira e pertenceria necessariamente a $\text{conv}(W_{pq})$.

Seja c_{ij} um vector de custos tal que a solução $\{y_{ij}\}$ seja a única solução óptima (projectada) do problema de minimização correspondente. O resultado I.1 estabelece que, para qualquer vector de custos c_{ij} , a formulação estendida apresentada para o problema CMCRS tem solução óptima inteira, incluindo o caso em que $c_{ij} = c_{ij}$. Designe-se essa solução por $\{z_{ij}^h\}$. Seja $\{y_{ij}\}$ a solução projectada que se obtém a partir de $\{z_{ij}^h\}$ através de (8): necessariamente $\{y_{ij}\}$ é uma solução inteira, pelo que pertence a $\text{conv}(W_{pq})$ e tem o mesmo custo que $\{y_{ij}\}$.

Deste modo, $\{y_{ij}\}$ não é a única solução óptima do problema, o que contradiz a hipótese, ou $\{y_{ij}\} = \{y_{ij}\}$, o que prova o resultado. ■

Assim, uma vez que as restrições (5a)-(6c) modelam um *trajecto-H*-(p, q), temos então uma segunda forma de explicitar matematicamente a restrição (1) do modelo *GEN* (como mencionámos antes, note-se que em verdade as referidas restrições modelam um fortalecimento da restrição genérica (1)). Como observado anteriormente, as variáveis z_{ij}^h relacionam-se com as variáveis y_{ij} , definidas na apresentação do modelo genérico, através das igualdades $y_{ij} = \sum_{h=1, \dots, H} z_{ij}^h$, para todo o $(i, j) \in A$. Substitua-se então a restrição (1) do modelo *GEN* pelas restrições do modelo acima e incluam-se as restrições de ligação das variáveis y_{ij} e z_{ij}^h . Desta forma, obtemos o seguinte modelo:

Modelo HOP

(5a), (5b), (5c), (6a), (6b), (6c)

$$y_{ij} = \sum_{h=1, \dots, H} z_{ij}^h \tag{8}$$

$$y_{ij} \in \{0, 1\}, (i, j) \in A. \tag{2}$$

Realçamos, de novo, o facto deste modelo ser compacto, visto a dimensão do grafo

expandido G_{pq} que deu origem às restrições **(5a)**-**(6c)** depender polinomialmente da dimensão do grafo original G . Note-se ainda que as restrições de ligação **(8)** permitem rescrever o modelo HOP sem as variáveis y_{ij} .

Observe-se agora com maior detalhe o significado das variáveis lacete z_{ii}^h ($i \in V \setminus \{p\}, h = 2, \dots, H$) do modelo HOP . Vimos já que estas variáveis estão associadas a arcos da forma $((i, h), (i, h + 1))$ do grafo expandido G_{pq} , correspondendo a lacetes no nodo i , no máximo na posição h , no *trajecto- H - (p, q)* em G . É precisamente devido a estas variáveis lacete que as restrições **(5a)**-**(6c)** do modelo HOP modelam correctamente a determinação de um *trajecto- H - (p, q)* . Note-se que no caso em que esse caminho tem menos que H saltos, as variáveis lacete são necessárias para que haja precisamente H variáveis com valor 1. Claramente, poderá haver várias soluções nestas condições. Considere-se de novo o exemplo do caminho a tracejado da figura I.4 (em que $z_{pa}^1 = z_{aq}^2 = z_{qa}^3 = 1$). Uma outra solução possível seria $z_{pa}^1 = z_{aa}^2 = z_{aq}^3 = 1$, ilustrada na figura I.5 sobre os grafos original e expandido.

Observe-se agora que o número de soluções alternativas poderá ser reduzido se simplesmente eliminarmos as variáveis lacete z_{ii}^h , com $i \neq q$. Ignoramos portanto arcos da forma $((i, h), (i, h + 1))$, com $i \neq q$ e $h = 1, \dots, H - 1$, em G_{pq} , o que é equivalente em termos do grafo original G a aceitar lacetes nas posições h , com $h = 2, \dots, H$, apenas em q .

Assim, se para um determinado h (com $2 \leq h \leq H$) tivermos $z_{qq}^h = 1$, significa que o *trajecto- H - (p, q)* tem exactamente $h - 1$ saltos. Deste modo, para todo o $h = 1, \dots, H$ e $(i, j) \in A$, podemos interpretar as variáveis anteriormente definidas, da seguinte forma:

$$z_{ij}^h = \begin{cases} 1, & \text{se o arco } (i, j) \text{ é percorrido exactamente na posição } h \\ 0, & \text{c. c.} \end{cases} .$$

Pelo exposto, temos ainda as variáveis lacete (sobre q , apenas), para todo o $h = 2, \dots, H$, definidas como:

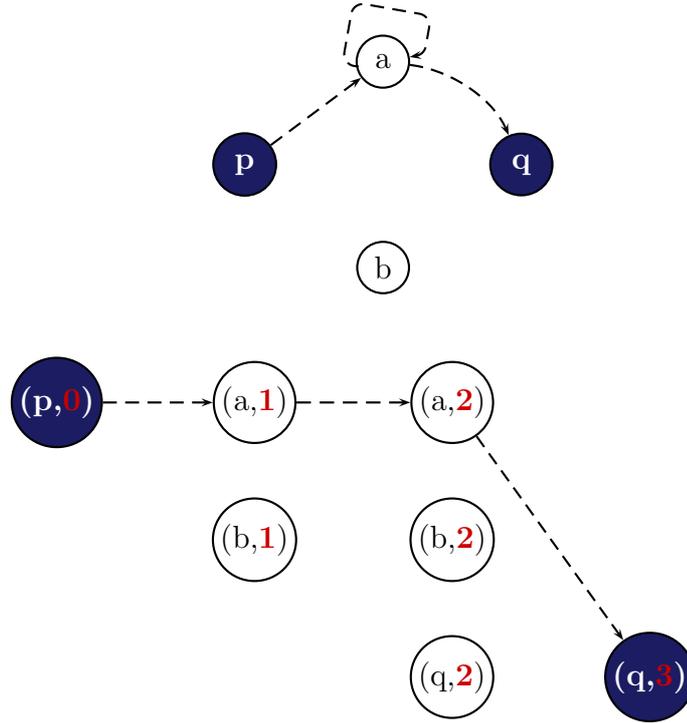


Figura I.5: Caminho alternativo ao representado a tracejado na figura I.4, em termos do grafo original G , em cima, bem como do grafo expandido G_{pq} .

$$z_{qq}^h = \begin{cases} 1, & \text{se o lacete } (q, q) \text{ é percorrido exactamente na posição } h \\ 0, & \text{c. c.} \end{cases} .$$

Deste modo, as restrições **(6b)** podem ser eliminadas do modelo HOP . Ao longo do presente trabalho, de ora em diante e pelas razões referidas, consideramos que o modelo HOP contém apenas estas variáveis lacete sobre q e, por conseguinte, interpretaremos as variáveis do modelo conforme ultimamente descritas.

I.1.4 Comparação de Modelos

Nesta subsecção procederemos à comparação das duas descrições da restrição genérica **(1)** do modelo GEN apresentadas nas subsecções anteriores. Isto é, estabeleceremos a relação entre os conjuntos de soluções admissíveis dos modelos $TRAD$ e HOP , tendo naturalmente em conta as respectivas relaxações lineares.

Relembre-se que os pontos extremos de $F(TRAD_L)$ – sendo $F(TRAD_L)$ o conjunto de soluções admissíveis da relaxação linear do modelo $TRAD$ – nem sempre são inteiros (conforme ilustrado na figura I.2), pelo que, naturalmente, $F(TRAD_L) \supseteq conv((\mathbf{1}), (\mathbf{2}))$ e note-se que, por sua vez, $conv((\mathbf{1}), (\mathbf{2})) \supseteq conv(W_{pq})$. Denotando por $Proj_Y(F(HOP_L))$ a projecção de $F(HOP_L)$ no espaço das variáveis y_{ij} através das restrições de ligação **(8)**, o resultado I.2 permite-nos estabelecer que:

Resultado I.3 (*Gouveia 1998*)

$$Proj_Y(F(HOP_L)) \subseteq F(TRAD_L).$$

Deste modo, para um qualquer problema que contenha restrições de caminho entre dois nodos com restrição de salto (entre outras), ao considerar o modelo HOP temos a garantia de que a formulação resultante fornecerá um limite inferior (ou superior) nunca pior do que considerando o modelo $TRAD$.

Concluimos assim esta secção, frisando que apesar do maior número de variáveis e restrições da formulação HOP , esta é exacta e compacta (como vimos, a sua dimensão depende polinomialmente da dimensão do grafo original G).

I.2 Determinação de D Caminhos com Restrições de Salto

Nesta secção discutiremos modelos para a determinação de D caminhos entre dois nodos num grafo, cada um dos quais com não mais que H saltos. Adaptar-se-ão os modelos propostos na secção I.1 e, com base no modelo *HOP*, será proposto um novo modelo que utiliza um único conjunto de variáveis (enquanto que as adaptações dos modelos já apresentados – *TRAD* e *HOP* – contêm D conjuntos de variáveis, um para cada caminho entre p e q).

Nesta secção continuaremos a ter em conta um grafo orientado $G = (V, A)$, assim como as definições I.1.1, I.1.2 e I.1.3.

À semelhança da exposição feita na secção anterior, preocupar-nos-emos apenas em modelar um conjunto de arcos que contenha D *caminhos- H -(p, q)*, dada a dificuldade associada à descrição exacta de um *caminho- H -(p, q)* (que, como referimos antes, está relacionada com o facto desse caminho ser elementar). Note-se, de novo, que na presença de outras restrições nos problemas em que os modelos apresentados nesta secção forem inseridos, bem como se houver custos associados aos arcos, as soluções que se obtêm são de facto *caminhos- H -(p, q)*.

I.2.1 Modelo Genérico

Pretendendo determinar D caminhos entre dois nodos $p, q \in V$ cada um dos quais com não mais do que H arcos (saltos) sobre um grafo $G = (V, A)$, o modelo genérico que se segue – genérico no mesmo sentido que anteriormente (ver subsecção I.1.1), *i.e.*, por não explicitar matematicamente as restrições que garantem a existência de D *caminhos- H -(p, q)* – utiliza, para qualquer arco $(i, j) \in A$, as seguintes variáveis inteiras:

$$f_{ij} = \text{número de caminhos que utilizam o arco } (i, j).$$

Deste modo, podemos então escrever o seguinte modelo genérico:

Modelo GEN^D

$$\{(i, j) \in A : f_{ij} > 0\} \text{ contém } D \text{ caminhos-}H\text{-}(p, q) \quad (9)$$

$$f_{ij} \in \mathbb{Z}_0^+, (i, j) \in A. \quad (10)$$

A restrição genérica (9) garante que os arcos utilizados contêm D caminhos- H -(p, q). Note-se que esses D caminhos não são necessariamente distintos, conforme ilustrado na figura I.6, podendo partilhar arcos e/ou nodos.

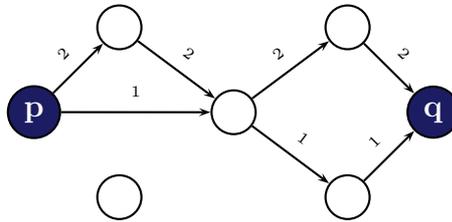


Figura I.6: Três ($D = 3$) caminhos-4-(p, q). Junto a cada arco (i, j), encontra-se o valor da respectiva variável f_{ij} .

À semelhança da exposição efectuada na secção I.1, a restrição genérica (9) será explicitada matematicamente nas subsecções seguintes.

I.2.2 Modelo Tradicional

Começamos por realçar que devido às restrições de salto é necessário distinguir cada um dos D caminhos entre p e q . De facto, considerando as variáveis f_{ij} definidas na subsecção anterior, facilmente se adaptariam as restrições (3a)-(3c) apresentadas na subsecção I.1.2 de forma a modelar D caminhos entre p e q . No entanto, a restrição de cardinalidade (3d) não é directamente adaptável uma vez que pretendemos que cada um dos D caminhos entre p e q tenha no máximo H saltos.

Deste modo, utilizaremos D conjuntos de variáveis binárias, semelhantes às definidas na subsecção I.1.2, mas distinguindo por que caminho um arco é utilizado ou

não. Considerem-se então, para quaisquer $(i, j) \in A$ e $d = 1, \dots, D$, as variáveis

$$y_{ij}^d = \begin{cases} 1, & \text{se o arco } (i, j) \text{ é utilizado pelo } d\text{-ésimo caminho-}H\text{-}(p, q) \\ 0, & \text{c. c.} \end{cases}.$$

Note-se que, à semelhança do modelo *TRAD*, não se consideram as variáveis $y_{qj}^d, \forall (q, j) \in A, \forall d = 1, \dots, D$, nem as variáveis $y_{ip}^d, \forall (i, p) \in A, \forall d = 1, \dots, D$, visto que estas variáveis indicam se, para um qualquer caminho d , um arco chega ao nodo inicial p ou se um arco parte do nodo final q . Desta forma, essas variáveis são simplesmente omitidas no modelo.

Utilizando as variáveis definidas acima, podemos então explicitar a restrição genérica **(9)** através de D sistemas de equações, sendo cada um deles similar ao apresentado na subsecção I.1.2 (ver também [2]), da seguinte forma:

$$\sum_{j:(p,j) \in A} y_{pj}^d = 1, d = 1, \dots, D \quad (11a)$$

$$\sum_{j:(i,j) \in A} y_{ij}^d - \sum_{j:(j,i) \in A} y_{ji}^d = 0, i \neq p, q; d = 1, \dots, D \quad (11b)$$

$$\sum_{j:(j,q) \in A} y_{jq}^d = 1, d = 1, \dots, D \quad (11c)$$

$$\sum_{(i,j) \in A} y_{ij}^d \leq H, d = 1, \dots, D. \quad (11d)$$

Relembre-se que, para cada $d = 1, \dots, D$, as restrições **(11a)**-**(11c)** são as restrições de caminho usuais, especificando o d -ésimo caminho de p a q , e a correspondente restrição **(11d)** restringe o número de saltos desse caminho a H .

Note-se que, para cada arco $(i, j) \in A$, o número de caminhos que atravessa esse arco é dado por $\sum_{d=1, \dots, D} y_{ij}^d$, ou seja, as variáveis y_{ij}^d relacionam-se com as variáveis f_{ij} do modelo genérico através das igualdades

$$f_{ij} = \sum_{d=1, \dots, D} y_{ij}^d, (i, j) \in A. \quad (12)$$

Substituindo a restrição **(9)** do modelo GEN^D pelas do sistema **(11a)**-**(12)**, obtemos um modelo válido para a determinação de um conjunto de arcos que contém

D caminhos- H -(p, q), doravante designado por $TRAD^D$:

Modelo $TRAD^D$

(11a), (11b), (11c), (11d)

$$f_{ij} = \sum_{d=1, \dots, D} y_{ij}^d, (i, j) \in A \quad (12)$$

$$f_{ij} \in \mathbb{Z}_0^+, (i, j) \in A. \quad (10)$$

Observe-se que as restrições de ligação (12) permitem rescrever este modelo sem as variáveis f_{ij} . Note-se ainda que, à semelhança do modelo GEN^D , os D caminhos modelados pelo sistema de restrições (11a)-(11d) não são necessariamente distintos (tanto em termos de arcos como de nodos). De facto, essas restrições são formadas por D subsistemas independentes, cada um deles modelando um subconjunto de arcos que contém um caminho- H -(p, q), em que as variáveis y_{ij}^d e $y_{ij}^{d'}$ não se relacionam entre si, para qualquer par de valores $d, d' = 1, \dots, D$ com $d \neq d'$. A figura I.7 ilustra o mesmo exemplo da figura I.6, em que junto a cada arco (i, j) se encontra o valor da respectiva variável y_{ij}^d (junto aos arcos a cheio, tracejado e picotado encontram-se os valores das variáveis y_{ij}^1 , y_{ij}^2 e y_{ij}^3 , respectivamente).

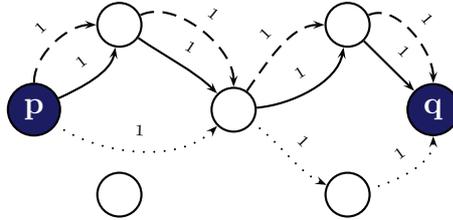


Figura I.7: Valores das variáveis y_{ij}^1 , y_{ij}^2 e y_{ij}^3 (junto aos arcos a cheio, tracejado e picotado, respectivamente), relativamente aos três ($D = 3$) caminhos-4-(p, q) ilustrados na figura I.6.

Note-se que os pontos extremos de $TRAD_L^D$ poderão ser fraccionários, pois para cada $d = 1, \dots, D$ as restrições (11a)-(11d) são idênticas às restrições do modelo $TRAD$ apresentado na secção I.1.2 (cuja relaxação linear tem pontos extremos não necessariamente inteiros, como ilustrado anteriormente na figura I.2).

Como observado no final da subsecção I.1.2, note-se que, para cada $d = 1, \dots, D$, se desconhece uma descrição completa do envolvente convexo dos vectores de incidência de *caminhos- H* -(p, q) usando apenas as variáveis y_{ij}^d [9]. À semelhança da secção I.1, apresentamos de seguida o modelo HOP^D , cuja relaxação linear, para cada $d = 1, \dots, D$, é uma descrição estendida do envolvente convexo dos vectores de incidência do d -ésimo *trajecto- H* -(p, q) para qualquer H .

I.2.3 Modelo Hop-Indexed

À semelhança da exposição efectuada na subsecção I.1.3, apresentamos agora uma descrição de um fortalecimento da restrição genérica (9), que, como veremos, tem vantagens relativamente à descrição da mesma restrição genérica dada pelo modelo $TRAD^D$.

Consideremos então o seguinte fortalecimento de (9):

$$\{(i, j) \in A : f_{ij} > 0\} \text{ forma } D \text{ trajectos-}H\text{-(}p, q\text{)}.$$

Começamos então por recordar que um *trajecto- H* -(p, q) sobre o grafo original $G = (V, A)$ pode ser visto como um caminho não restringido no número de saltos num grafo expandido. Considerando um qualquer caminho d , com $d = 1, \dots, D$, seja G_{pq}^d esse grafo expandido. Dada então a correspondência existente entre um caminho de $(p, 0)$ a (q, H) em G_{pq}^d e um *trajecto- H* -(p, q) no grafo original G , para cada $d = 1, \dots, D$, bem como tendo em conta a exposição feita na subsecção I.1.3 (relativamente à eliminação das variáveis lacete sobre nodos diferentes de q), considerem-se as seguintes variáveis binárias, para quaisquer $(i, j) \in A$, $h = 1, \dots, H$ e $d = 1, \dots, D$:

$$z_{ij}^{hd} = \begin{cases} 1, & \text{se o arco } (i, j) \text{ é percorrido exactamente} \\ & \text{na posição } h \text{ pelo } d\text{-ésimo } \textit{trajecto-}H\text{-(}p, q\text{)} \quad , \\ 0, & \text{c. c.} \end{cases}$$

e ainda as variáveis lacete (sobre q , apenas), para todo o $h = 2, \dots, H$ e todo o

$d = 1, \dots, D$, definidas como

$$z_{qq}^{hd} = \begin{cases} 1, & \text{se o lacete } (q, q) \text{ é percorrido exactamente} \\ & \text{na posição } h \text{ pelo } d\text{-ésimo } \textit{trajecto-H-(p, q)} \\ 0, & \text{c. c.} \end{cases} .$$

Utilizando estas variáveis, podemos então explicitar o fortalecimento da restrição genérica **(9)** do modelo GEN^D da seguinte forma:

$$\sum_{j:(p,j) \in A} z_{pj}^{1d} = 1, \quad d = 1, \dots, D \quad (13a)$$

$$\sum_{j:(i,j) \in A} z_{ij}^{h+1,d} - \sum_{j:(j,i) \in A} z_{ji}^{hd} = 0, \quad i \neq p; h = 1, \dots, H-1; d = 1, \dots, D \quad (13b)$$

$$\sum_{j:(j,q) \in A} z_{jq}^{Hd} = 1, \quad d = 1, \dots, D \quad (13c)$$

$$z_{ij}^{hd} \in \{0, 1\}, \quad (i, j) \in A; h = 1, \dots, H; d = 1, \dots, D \quad (14a)$$

$$z_{qq}^{hd} \in \{0, 1\}, \quad h = 2, \dots, H; d = 1, \dots, D. \quad (14b)$$

Tendo em conta a exposição efectuada na subsecção I.1.2, podemos afirmar que, para cada $d = 1, \dots, D$, o conjunto de soluções admissíveis da relaxação linear do d -ésimo sub-sistema das restrições **(13a)**-**(14b)** tem pontos extremos inteiros. Ou seja, podemos replicar o resultado I.1 D vezes, como se segue:

Resultado I.4

*Para cada $d = 1, \dots, D$, os pontos extremos do polítopo definido pela relaxação linear das restrições **(13a)**-**(14b)** são inteiros.*

Deste modo, para cada subsistema d das restrições acima apresentadas, com $d = 1, \dots, D$, podemos relaxar as restrições de integralidade das variáveis e, até, substituir as respectivas restrições **(14a)**-**(14b)** por restrições de não negatividade (na medida em que **(13a)**-**(13c)** tornam os limites superiores das variáveis redundantes). Assim, num qualquer problema com uma dada estrutura de custos em que o sub-sistema resultante for integrado, a sua solução óptima não se altera ao fazer a

referida substituição.

Note-se agora que, para cada arco $(i, j) \in A$, as variáveis z_{ij}^{hd} relacionam-se com as variáveis f_{ij} do modelo GEN^D através das igualdades seguintes:

$$f_{ij} = \sum_{d=1, \dots, D} \sum_{h=1, \dots, H} z_{ij}^{hd}, (i, j) \in A. \quad (15)$$

Substituindo a restrição (9) do modelo GEN^D pelas do sistema (13a)-(14b), obtemos um modelo válido para a determinação de D caminhos- H -(p, q), que designaremos por HOP^D (recorde-se que na presença de outras restrições e/ou custos associados aos arcos em problemas mais gerais, as soluções obtidas são de facto caminhos- H -(p, q):

Modelo HOP^D

(13a), (13b), (13c), (14a), (14b)

$$f_{ij} = \sum_{d=1, \dots, D} \sum_{h=1, \dots, H} z_{ij}^{hd}, (i, j) \in A \quad (15)$$

$$f_{ij} \in \mathbb{Z}_0^+, (i, j) \in A. \quad (10)$$

Novamente, observe-se que as restrições de ligação (15) permitem rescrever este modelo sem as variáveis f_{ij} . Note-se ainda que as variáveis z_{ij}^{hd} do d -ésimo subsistema de (13a)-(14b) não estão relacionadas com as variáveis $z_{ij}^{hd'}$ do d' -ésimo subsistema, com $d, d' = 1, \dots, D$ e $d \neq d'$. Assim, à semelhança da observação efectuada anteriormente para os modelos GEN^D e $TRAD^D$, note-se que os D caminhos- H -(p, q) modelados por HOP^D não são necessariamente distintos (quer em termos de arcos como de nodos).

Para concluir esta subsecção, realçamos que o modelo HOP^D tem D conjuntos de variáveis, um para um dos D caminhos- H -(p, q) que se pretendem determinar. Na subsecção seguinte, veremos que é possível diminuir o número de variáveis sem perder as boas características (referidas no resultado I.4) do modelo que acabámos de analisar.

I.2.4 Modelo Hop-Indexed Agregado

Nesta subsecção apresentamos uma terceira descrição da restrição genérica **(9)** do modelo GEN^D . Como veremos mais à frente, apesar do novo modelo ser uma versão agregada do modelo HOP^D , pelo que o designaremos por $HOP-A^D$, ao considerarmos esse modelo integrado em problemas mais gerais temos a garantia de que a qualidade dos limites das respectivas relaxações lineares não é alterada.

Como motivação, começemos por considerar o modelo HOP^D da subsecção anterior e apresentar um grafo original G sobre o qual existem $D = 3$ *trajectos- H* -(p, q), com $H = 3$, e os correspondentes caminhos sobre os grafos expandidos G_{pq}^1 , G_{pq}^2 e G_{pq}^3 . A figura I.8 ilustra esse exemplo, em que os referidos grafos expandidos se encontram sobrepostos e onde junto aos arcos a cheio, tracejado e picotado se encontram os valores das variáveis z_{ij}^{h1} , z_{ij}^{h2} e z_{ij}^{h3} , respectivamente.

Uma vez que para cada $d = 1, \dots, D$, o grafo expandido G_{pq}^d cumpre implicitamente com as restrições de salto, o exemplo apresentado sugere que não há necessidade em distinguir cada um dos D caminhos. De facto, como vimos na subsecção I.2.3, dada a correspondência entre o d -ésimo *trajecto- H* -(p, q) em G e um caminho no grafo expandido G_{pq}^d , para cada $d = 1, \dots, D$, podemos agregar os fluxos que atravessam arcos com nodos extremos iguais, como ilustrado na figura I.9.

Observemos agora que o processo ilustrado na figura I.9 corresponde a agregar em $d = 1, \dots, D$ as restrições **(13a)**, **(13b)**, **(13c)**, **(14a)** e **(14b)** do modelo HOP^D , bem como fazer a substituição de variáveis da forma $w_{ij}^h = \sum_{d=1, \dots, D} z_{ij}^{hd}$, para qualquer $(i, j) \in A \cup \{(q, q)\}$ e $h = 1, \dots, H$. Desta forma, podemos interpretar as novas variáveis do seguinte modo:

$$w_{ij}^h = \text{número de } \textit{trajectos-}H\textit{-(}p, q\textit{) que percorrem o arco } (i, j) \\ \text{exactamente na posição } h.$$

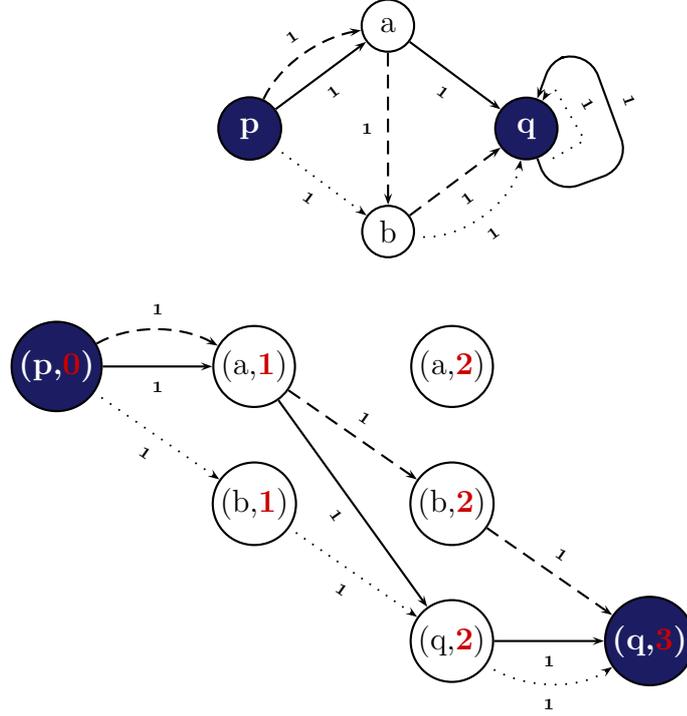


Figura I.8: Três ($D = 3$) *trajectos-3- (p, q)* sobre o grafo original G , em cima, e os correspondentes caminhos sobre os grafos expandidos G_{pq}^1 , G_{pq}^2 e G_{pq}^3 , sobrepostos em baixo. Junto aos arcos a cheio, tracejado e picotado encontram-se os valores das variáveis z_{ij}^{h1} , z_{ij}^{h2} e z_{ij}^{h3} , respectivamente.

Obtemos assim uma forma alternativa de explicitar o fortalecimento da restrição genérica (9) apresentado na subsecção anterior, dado por:

$$\sum_{j:(p,j) \in A} w_{pj}^1 = D \quad (16a)$$

$$\sum_{j:(i,j) \in A} w_{ij}^{h+1} - \sum_{j:(j,i) \in A} w_{ji}^h = 0, \quad i \neq p; h = 1, \dots, H-1 \quad (16b)$$

$$\sum_{j:(j,q) \in A} w_{jq}^H = D \quad (16c)$$

$$w_{ij}^h \in \{0, 1, \dots, D\}, \quad (i, j) \in A; h = 1, \dots, H \quad (17a)$$

$$w_{qq}^h \in \{0, 1, \dots, D\}, \quad h = 2, \dots, H. \quad (17b)$$

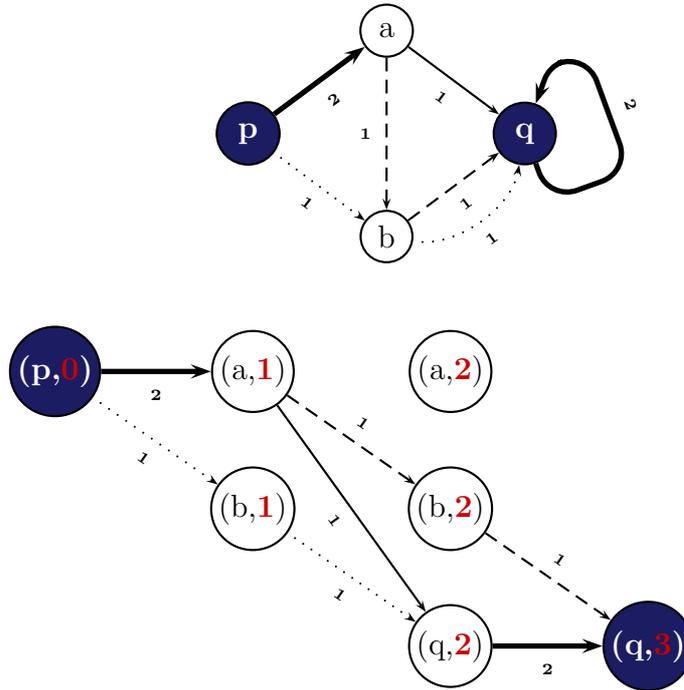


Figura I.9: Agregação de fluxos em G , em cima, e na sobreposição de G_{pq}^1 , G_{pq}^2 e G_{pq}^3 (ou $G_{pq}^{1,2,3}$), em baixo, com base no exemplo da figura I.8.

Note-se que este é um modelo de fluxos num grafo expandido, grafo este que resulta da sobreposição dos D grafos expandidos (G_{pq}^d , com $d = 1, \dots, D$) associados ao modelo HOP^D . Seja $G_{pq}^{1, \dots, D}$ esse grafo expandido e observe-se que os seus conjuntos de nodos e arcos coincidem com os conjuntos V_{pq} e A_{pq} , respectivamente, definidos anteriormente na subsecção I.1.2. Deste modo e como ilustrado nas figuras I.8 e I.9, existe uma correspondência entre D *trajectos*- H -(p, q) sobre o grafo original G e D caminhos não restringidos no número de arcos sobre $G_{pq}^{1, \dots, D}$. Note-se que (como observado nos modelos anteriores) os referidos D caminhos sobre G poderão (obviamente) não ser distintos. Novamente, note-se ainda que nas soluções obtidas considerando as restrições **(16a)**-**(17b)** em conjunto com outras restrições e/ou na presença de custos associados aos arcos obtemos, de facto, D *caminhos*- H -(p, q) conforme a definição I.1.3.

Uma vez que a matriz das restrições acima apresentadas é totalmente unimodular, podemos estabelecer o seguinte resultado (similar aos resultados I.1 e I.4):

Resultado I.5

Os pontos extremos do polítopo definido pela relaxação linear das restrições (16a)-(17b) são inteiros.

Deste modo, podemos substituir as restrições (17a) e (17c) por restrições de não negatividade, visto que as restrições (16a)-(16b) tornam os limites superiores da relaxação da integralidade das variáveis redundantes.

Tendo em conta as restrições de ligação (15) do modelo HOP^D , naturalmente que, para qualquer arco $(i, j) \in A$, a relação entre as novas variáveis w_{ij}^h acima definidas e as variáveis f_{ij} do modelo GEN^D é dada pelas igualdades $f_{ij} = \sum_{h=1, \dots, H} w_{ij}^h$. Substituindo a restrição (9) do modelo GEN^D pelas do sistema (16a)-(17b), obtemos, pelas mesmas razões que foram apresentadas anteriormente, um modelo válido para a determinação de D caminhos- H -(p, q), que designaremos por $HOP-A^D$:

Modelo $HOP-A^D$

$$(16a), (16b), (16c), (17a), (17b)$$

$$f_{ij} = \sum_{h=1, \dots, H} w_{ij}^h, (i, j) \in A \tag{18}$$

$$f_{ij} \in \mathbb{Z}_0^+, (i, j) \in A. \tag{10}$$

Note-se que, dadas as restrições de ligação (18) (e similarmente às observações efectuadas relativamente aos modelos apresentados nas subsecções anteriores), o modelo $HOP-A^D$ pode ser rescrito sem as variáveis f_{ij} .

À semelhança dos modelos HOP e HOP^D , note-se que este modelo contém variáveis lacete w_{qq}^h , com $h = 2, \dots, H$, que modelam situações em que pelo menos um dos D *trajectos- H -(p, q)* tem menos que H arcos (*i.e.*, $w_{jq}^h \geq 1$ para um $(j, q) \in A, j \in V \setminus \{q\}$ e um $h = 1, \dots, H - 1$).

Observemos agora que, pelo resultado I.5, podemos afirmar que a projecção do conjunto de soluções admissíveis da relaxação linear de $HOP-A^D$ no espaço das variáveis f_{ij} , através das igualdades (18), tem pontos extremos inteiros.

Resultado I.6

Os pontos extremos de $Proj_F(F(HOP-A_L^D))$ são inteiros.

Concluimos esta subsecção realçando a importância deste resultado e o facto do modelo $HOP-A^D$, sendo uma versão agregada de HOP^D , utilizar apenas $1/D$ das variáveis ($w_{ij}^h = \sum_{d=1, \dots, D} z_{ij}^{hd}$, $\forall (i, j) \in A \cup \{(q, q)\}$, $\forall h = 1, \dots, H$). Na próxima subsecção analisaremos em maior detalhe a relação entre estas duas formas de representar a restrição genérica (9) do modelo GEN^D .

I.2.5 Comparação de Modelos

Nesta subsecção efectuaremos a comparação dos modelos $TRAD^D$, HOP^D e $HOP-A^D$ apresentados nas subsecções anteriores, tendo em conta os conjuntos de soluções admissíveis das respectivas relaxações lineares.

O resultado apresentado de seguida estabelece a relação entre as relaxações lineares dos modelos $TRAD^D$ e HOP^D . Denote-se a projecção do conjunto de soluções admissíveis da relaxação linear do modelo $TRAD^D$ no espaço das variáveis f_{ij} por $Proj_F(F(TRAD_L^D))$, obtida através das igualdades (12). Similarmente, denote-se por $Proj_F(F(HOP_L^D))$ a projecção no espaço das variáveis f_{ij} do conjunto de soluções admissíveis da relaxação linear do modelo HOP^D através das igualdades (15). Podemos então estabelecer que:

Resultado I.7

$$Proj_F(F(HOP_L^D)) \subseteq Proj_F(F(TRAD_L^D)).$$

Demonstração: Começemos por substituir, para cada arco $(i, j) \in A$, as restrições de ligação (16) do modelo HOP^D pelas igualdades $y_{ij}^d = \sum_{h=1, \dots, H} z_{ij}^{hd}$, para qualquer $d = 1, \dots, D$ (observe-se que, para cada $d = 1, \dots, D$, a relação entre as variáveis y_{ij}^d e z_{ij}^{hd} é descrita por (8) – ver subsecção I.1.3), e ainda $f_{ij} = \sum_{d=1, \dots, D} y_{ij}^d$. Note-se que esta modificação não prejudica o modelo HOP^D em termos da sua relaxação linear e, simultaneamente, permite-nos comparar os dois modelos em questão no espaço das

variáveis y_{ij} .

Para cada $d = 1, \dots, D$, denotemos o conjunto de soluções admissíveis da relaxação linear do modelo $TRAD^D$ sem as restrições de ligação (12) (como observado anteriormente, podemos rescrever o modelo $TRAD^D$ sem as variáveis f_{ij}) por $F(TRAD_L^{D,d})$. Similarmente e para o mesmo d , seja $F(HOP_L^{D,d})$ o conjunto de soluções admissíveis da relaxação linear do modelo HOP^D sem as restrições de ligação (15). Usemos agora o resultado I.3 da subsecção I.1.4. De facto, para um qualquer $d = 1, \dots, D$, a projecção de $F(HOP_L^{D,d})$ no espaço das variáveis y_{ij} através de $y_{ij}^d = \sum_{h=1, \dots, H} z_{ij}^{hd}$ está, segundo esse resultado, contida em $F(TRAD_L^{D,d})$.

Deste modo e tendo em conta as restrições de ligação (12) e (15), tem-se que $Proj_F(F(HOP_L^D)) \subseteq Proj_F(F(TRAD_L^D))$.

■

Vejamos agora a relação existente entre os modelo HOP^D e $HOP-A^D$. O facto deste modelo ser uma versão agregada do primeiro sugere que a sua relaxação linear poderá não ser tão apertada quanto a de HOP^D . No entanto, denotando por $Proj_F(F(HOP_L^D))$ e $Proj_F(F(HOP-A_L^D))$ as projecções dos conjuntos de soluções admissíveis das relaxações lineares de HOP^D e $HOP-A^D$ através de (15) e (18), respectivamente, no espaço das variáveis f_{ij} , podemos estabelecer que:

Resultado I.8

$$Proj_F(F(HOP_L^D)) = Proj_F(F(HOP-A_L^D)).$$

Demonstração: Recorde-se que as variáveis z_{ij}^{hd} do primeiro modelo se relacionam com as variáveis w_{ij}^h através das igualdades $w_{ij}^h = \sum_{d=1, \dots, D} z_{ij}^{hd}$, para quaisquer $(i, j) \in A$ e $h = 1, \dots, H$. Adicionem-se estas igualdades ao modelo HOP^D (o que não prejudica a relaxação linear do mesmo modelo) e denote-se por $Proj_W(F(HOP_L^D))$ a projecção, através das mesmas igualdades, do conjunto de soluções admissíveis de HOP_L^D (sem as restrições de ligação $f_{ij} = \sum_{d=1, \dots, D} \sum_{h=1, \dots, H} z_{ij}^{hd}$) no espaço das variáveis w_{ij}^h . Provemos então que $Proj_W(F(HOP_L^D)) = F(HOP-A_L^D)$. Note-se que ao adicionar

as igualdades $w_{ij}^h = \sum_{d=1, \dots, D} z_{ij}^{hd}$ ao modelo HOP^D a inclusão “ \subseteq ” fica imediatamente provada através das mesmas igualdades, pois qualquer solução admissível da relaxação linear de HOP^D satisfaz necessariamente a soma em $d = 1, \dots, D$ das restrições (13a)-(14b). Para provar a inclusão no sentido inverso, observe-se que, dada uma qualquer solução admissível da relaxação linear de $HOP-A^D$, facilmente se obtém uma solução admissível da relaxação linear de HOP^D fazendo $z_{ij}^{h1} = \dots = z_{ij}^{hD} = \frac{1}{D}w_{ij}^h$, para quaisquer $(i, j) \in A$ e $h = 1, \dots, H$. Deste modo e tendo em conta as restrições de ligação (15) e (18), tem-se que $Proj_F(F(HOP_L^D)) = Proj_F(F(HOP-A_L^D))$.

■

O modelo $HOP-A^D$ possui então claras vantagens relativamente à versão desagregada que lhe deu origem: à medida que D aumenta, o modelo agregado torna-se mais interessante visto que o número de variáveis de que faz uso decresce à razão $\frac{1}{D}$ relativamente ao número de variáveis do modelo HOP^D .

I.3 Conclusões

Neste capítulo discutimos diferentes formulações de Programação Linear Inteira para a determinação de um ou mais caminhos com restrições de salto entre dois nodos. Começámos por apresentar formulações (genéricas) em que as restrições de salto não são explicitadas matematicamente, apresentando de seguida descrições possíveis dessas restrições genéricas segundo diferentes abordagens. Compararam-se as relaxações lineares dos modelos resultantes, tendo ficado provadas as vantagens dos modelos baseados no trabalhos de Gouveia [11] em relação a modelos tradicionais. Como veremos no próximo capítulo, as vantagens dos primeiros modelos (baseados em variáveis com um índice adicional associado à posição em que um determinado arco é percorrido) revelar-se-ão, em termos da qualidade dos limites dados pelas relaxações lineares, bem como em termos do esforço computacional envolvido na determinação da solução óptima inteira, das formulações nas quais esses (sub)modelos serão inseridos.

Nos próximos capítulos deste trabalho, estudaremos vários problemas de telecomunicações em que há a necessidade (entre outras) de estabelecer D caminhos com restrições de salto entre vários pares de nodos $p, q \in S$, onde S é um subconjunto de nodos de V . Para tal, os modelos discutidos na secção I.2 deste capítulo poderão ser facilmente adaptados redefinindo as variáveis utilizadas e replicando os modelos para cada par de nodos $p, q \in S$. Como vimos, nenhum desses modelos especifica exactamente D caminhos- H -(p, q), pois na realidade garantem apenas que a sua solução consiste num conjunto de arcos que contém D caminhos- H -(p, q). No entanto, havendo custos associados aos arcos e/ou na presença de outras restrições nos problemas abordados nos capítulos II e III), as soluções obtidas são de facto D caminhos- H -(p, q), para qualquer $p, q \in S$.

Capítulo II

Desenho e Dimensionamento de Redes de Telecomunicações

Problemas de desenho e dimensionamento de redes têm sido extensivamente estudados nos últimos anos. Devido à importância que têm vindo a ocupar na vida quotidiana a diversos níveis, redes de telecomunicações têm ganho considerável destaque, tornando-se naturalmente objecto de intensa investigação. A crescente necessidade de transmissões de alta velocidade com elevados índices de fiabilidade e qualidade de serviço (QoS) têm motivado o estudo de diversas aplicações da vida real, habitualmente designadas por problemas de *network design*.

A atenção dada a problemas de desenho e dimensionamento de redes de telecomunicações deve-se essencialmente a dois factores. O primeiro consiste na liberalização do mercado das telecomunicações que exige dos operadores a oferta de serviços a preços competitivos. A apetência do mercado para novos e mais variados serviços que o sucesso da Internet criou nos utilizadores de telecomunicações constitui o segundo factor. Desta forma os operadores sentem necessidade de: i) adoptar novas tecnologias que lhes permitam suportar múltiplos serviços (telefonía, comunicação de dados, televisão interactiva e múltiplos outros serviços) numa única infra-estrutura de rede; e ii) configurar de uma forma mais criteriosa as suas redes de telecomunicações. O objectivo destas preocupações é reduzir custos de operação e manutenção de modo a praticar preços mais competitivos para os serviços suportados.

Neste capítulo abordamos dois problemas de otimização que surgem no contexto de redes de telecomunicações. A exposição feita na secção I.2 será fundamental uma vez que, em ambos os problemas, existe um conjunto de serviços, cada um dos quais definido por um nodo inicial e um nodo final que é necessário suportar através de D caminhos com restrições de salto entre os seus nodos extremos.

O primeiro problema, abordado na secção II.1, consiste no desenho de uma rede formada por uma única camada, mais concretamente uma rede *Multi-Protocol Label Switching* (MPLS). Representando esta rede através de um grafo em que existem custos associados à utilização dos seus nodos e arestas, pretende-se determinar um subgrafo de custo mínimo que suporte todos os serviços que existirem nessa rede com QoS. Para tal, são consideradas as já amplamente referidas restrições de salto, modelando o atraso máximo associado à transmissão de informação desde o nodo origem até ao nodo destino de cada serviço. Adicionalmente, existem restrições de sobrevivência, *i.e.*, é necessário mais do que um caminho suportando cada serviço e que os mesmos sejam disjuntos nos nodos de modo a garantir que as redes resistam a falhas. É possível modelar estas últimas restrições, como veremos, através da inclusão de simples condições adicionais.

Relativamente a este problema, consideramos apenas o caso $D = 2$, *i.e.*, o caso em que se exige a existência de dois caminhos suportando cada serviço. Discutimos três formulações de Programação Linear Inteira para diferentes valores de H , formulações estas obtidas a partir dos submodelos *TRAD*², *HOP*² e *HOP-A*² abordados na secção I.2. Considerando redes de teste com até 50 nodos e mais de 400 arestas, relatamos a nossa experiência computacional que atesta a veracidade dos resultados I.7 e I.8.

Na secção II.2, abordamos um problema de desenho e dimensionamento de duas camadas de rede em simultâneo: uma camada inferior do tipo *Wavelength Division Multiplexing* (WDM), que confere a transmissão de dados da rede MPLS da camada superior. Neste problema, veremos que é possível representar as duas camadas de rede através de um único grafo (com custos associados a nodos, como no problema

anterior, bem como às arestas – sendo neste problema dependentes da quantidade de equipamento que for necessário instalar em cada aresta). Na camada inferior, existem restrições de atenuação óptica, enquanto que na camada superior, tal como no primeiro problema, existem restrições de QoS e de sobrevivência. Neste nível superior, visto que cada serviço existente possui um tráfego associado, existem ainda restrições de largura de banda, *i.e.*, a capacidade instalada em cada aresta deve acomodar o tráfego total que a atravesse. Assim, resumidamente, neste problema pretende-se determinar um subgrafo de custo mínimo, custo este que depende não só dos nodos e arestas seleccionados como também da capacidade instalada em cada aresta seleccionada, que robustamente suporte todos os serviços (*i.e.*, tendo em conta as restrições de sobrevivência) com qualidade de serviço.

Dados os resultados obtidos no capítulo anterior, respeitantes às relações existentes entre os submodelos $TRAD^D$, HOP^D e $HOP-A^D$, bem como tendo em conta os resultados computacionais obtidos no estudo do problema da secção II.1 do presente capítulo, este segundo problema será abordado com base no submodelo $HOP-A^D$. Efectuamos um estudo comparativo considerando diferentes valores para D (2, 3 e 4), avaliando o impacto de diferentes mecanismos de sobrevivência (que dependem do modo como é encaminhado o tráfego de cada serviço através dos D caminhos que o suportam) no custo total das soluções óptimas obtidas para este problema.

Relativamente a problemas de desenho de redes envolvendo restrições de sobrevivência, *i.e.*, caminhos disjuntos nos nodos (ou arestas) entre determinados pares de nodos, existem múltiplas referências na literatura entre as quais se destacam os trabalhos de Raghavan [34], Stoer [40], Raghavan e Magnanti [35], Magnanti e Raghavan [26], Balakrishnan, Magnanti e Mirchandani [3], bem como Borne *et al.* [6].

No que toca a problemas de desenho de redes com restrições de salto e as suas aplicações, destacamos os trabalhos de Balakrishnan e Altinkemer [2], Gouveia [11], LeBlanc, Chifflet e Mahey [24] (apresentando diversas aplicações modeladas através de restrições de salto) e, ainda, o recente trabalho de Gouveia *et al.* [12] em que as restrições de salto são, à semelhança do presente trabalho, usadas para modelar restrições de QoS numa rede MPLS sobre WDM.

Mais recentemente, têm surgido na literatura diversas abordagens para problemas de desenho de redes com restrições de sobrevivência e de salto de custo mínimo, *i.e.*, problemas em que se pretende determinar um subgrafo de custo mínimo de um grafo em que existem um ou mais serviços (ou pares de nodos origem/destino), para os quais terão que haver D caminhos, com no máximo H saltos cada, disjuntos nas arestas e/ou nodos. Orłowski e Wessaely [33] discutem formulações envolvendo um número exponencial de variáveis e estudam o impacto de restrições de salto, sob diversos mecanismos de sobrevivência, no custo total das redes consideradas. Huygens, Mahjoub, e Pesneau [17], abordam o problema com um único serviço, $D = 2$ e disjunção nas arestas. Estes autores apresentam formulações de Programação Linear Inteira e concluem que o polítopo da respectiva relaxação linear é inteiro para $H = 2$ e $H = 3$. Para $H \leq 3$, a formulação apresentada pode ser facilmente estendida para o caso em que existe mais do que um par de nodos origem/destino (para $H \geq 4$, a formulação deixa de ser válida), no entanto, as desigualdades do modelo resultante não caracterizam completamente o polítopo associado. No trabalho de Huygens *et al.* [18], os autores discutem formulações com um número exponencial de restrições para o problema com vários serviços, $D \geq 2$, disjunção nas arestas, $H = 2$ e $H = 3$. Os autores demonstram que o problema é *NP-difícil* quando se consideram pares origem/destino com raiz num determinado nodo e custos unitários e tem resolução em tempo polinomial se o grafo for completo. Os mesmos autores apresentam diversas classes de desigualdades válidas para o problema, fornecendo condições necessárias e/ou suficientes para que as mesmas definam facetas. Destacamos ainda o trabalho de Bley [5], que estuda a complexidade computacional do problema com um único serviço, qualquer D , disjunção nos nodos e qualquer H , mostrando que sob determinada estrutura de custos o problema tem resolução em tempo polinomial para $H \leq 4$.

Neste trabalho, em particular no presente capítulo, apresentamos formulações de Programação Linear Inteira, baseadas nos submodelos apresentados no capítulo anterior, para problemas de desenho de redes de telecomunicações, formulações essas que poderão ser facilmente adaptadas para problemas mais gerais que envolvam restrições de sobrevivência e de salto.

II.1 Desenho de uma rede MPLS

Nesta secção discutimos diferentes formulações de Programação Linear Inteira, compactas (*i.e.*, com um número polinomial de variáveis e restrições), para um problema de desenho de redes MPLS.

A tecnologia subjacente a redes MPLS tem a capacidade de dar resposta aos problemas de performance e escalabilidade que os fornecedores de acesso a redes IP (Internet Protocol) desejam debelar. Com a tecnologia MPLS, as redes IP irão suportar no futuro diferentes serviços com diferentes requisitos de QoS, aumentando a eficiência e fiabilidade das mesmas. Em particular, esta tecnologia melhora a leitura/envio de pacotes de informação em determinados nodos da rede e beneficia do facto da forma como se processa a transmissão de informação correspondente a um serviço não estar restringida pelas de outros serviços [12].

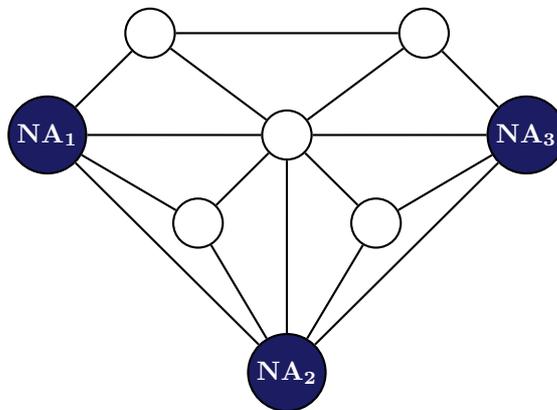


Figura II.1: Uma rede MPLS com 3 nodos de acesso (a cheio), 5 nodos candidatos à instalação de nodos de trânsito e as ligações que é possível utilizar.

Uma rede MPLS (exemplificada na figura II.1) contém nodos especiais, denominados por *Label Switching Routers* (LSRs), que podem ser de dois tipos: nodos de

acesso ou nodos de trânsito. Nodos do primeiro tipo são os nodos extremo (origens e/ou destinos) de um ou mais serviços. Os nodos de trânsito poderão ter que existir na rede – o que corresponde à instalação de equipamento em alguns dos nodos da rede, de entre um conjunto de nodos candidatos (e que não inclui os nodos de acesso) – de forma a possibilitar a comunicação entre nodos de acesso, sendo a sua localização objecto de decisão. O encaminhamento de um pacote IP (informação) desde um nodo de acesso inicial até ao correspondente nodo de acesso final é efectuado através de um caminho, designado por *Label Switched Path* (LSP), e baseia-se na etiqueta que esse pacote recebe na origem (etiqueta na qual consta o destino e a QoS requerida por esse pacote). O encaminhamento assim baseado implica que, como referido acima, a transmissão de informação de um determinado serviço não é condicionada pelas de outros serviços, o que representa uma vantagem relativamente a redes IP mais tradicionais.

II.1.1 Definição do Problema

A tecnologia MPLS tem portanto a capacidade de suportar serviços com requisitos que a Internet actual não tem. No presente problema, consideramos dois desses requisitos.

O primeiro está relacionado com as referidas restrições de salto: consiste num limite máximo de atraso associado à transmissão de pacotes, sendo fundamental em serviços de voz e/ou vídeo, por exemplo. Um pacote IP sofre um atraso em cada LSR que atravessar desde a sua origem até ao respectivo destino. Uma vez que por cada aresta $\{i, j\}$ que for atravessada de i para j teremos que instalar um nodo de trânsito em j , esse atraso é dado pelo número de saltos (H) que forem efectuados pelo caminho (LSP) ao longo do qual o referido pacote for enviado.

O segundo requisito compreende as restrições de sobrevivência, ou seja: para qualquer serviço, *i.e.*, entre qualquer par de nodos de acesso origem/destino, terão que existir $D = 2$ caminhos disjuntos nos nodos. Isto porque um corte na transmissão da informação é naturalmente intolerável em serviços de voz, por exemplo.

Representando a rede MPLS por um grafo não orientado $G = (V, E)$, seja $S \subset V$ o conjunto de nodos de acesso mencionado anteriormente. Qualquer par de nodos $p, q \in S$, com $p < q$, define um serviço, pelo que o número total de serviços a suportar será igual a $|S| \times (|S| - 1)/2$.

Assumamos que a localização de cada nodo de acesso em S é conhecida. Todos os restantes nodos $i \in V \setminus S$ são candidatos à instalação de um nodo de trânsito, a que está associado um custo a_i . Consideremos ainda que à utilização de cada aresta $\{i, j\} \in E$ está associado um custo $c_{\{i,j\}}$, custo este proporcional à distância euclidiana entre os respectivos nodos extremo.

O presente problema consiste portanto em determinar um subgrafo de custo mínimo cumprindo com os requisitos mencionado. Isto é, pretendemos saber em que nodos deverão ser instalados nodos de trânsito, bem como que arestas utilizar, minimizando o custo total do equipamento instalado/utilizado e de forma que na topologia resultante haja $D = 2$ caminhos disjuntos nos nodos entre qualquer par origem/destino e cada um desses caminhos não tenha mais do que H saltos. A figura II.2 ilustra uma solução admissível para o exemplo representado pela figura II.1, considerando $D = 2$ e $H = 3$ (em que, note-se, o nodo de acesso 2 desempenha o papel de um nodo de trânsito num dos caminhos entre os nodos de acesso 1 e 3.).

II.1.2 Formulação Genérica

Com base no modelo GEN^D apresentado na subsecção I.2.1, começamos por apresentar uma formulação genérica para o presente problema.

Note-se agora que havendo a necessidade de distinguir a direcção em que qualquer aresta é atravessada, consideramos o conjunto de arcos A que contém os “arcos” (i, j) e (j, i) por cada aresta $\{i, j\} \in E$. Desta forma, nas próximas secções, se um arco (i, j) for percorrido, significa que a aresta $\{i, j\}$ é percorrida de i para j .

Assim, a referida formulação genérica envolve três conjuntos de variáveis, que definimos da seguinte forma:

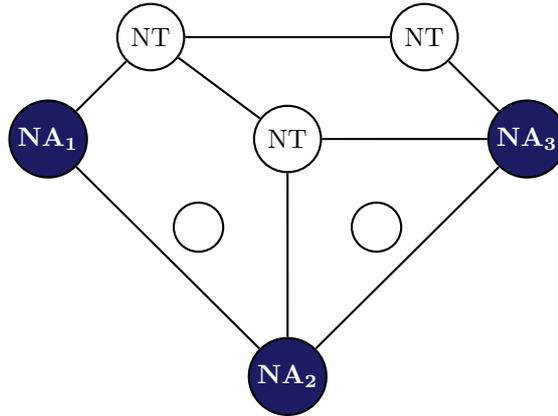


Figura II.2: Uma solução admissível para o problema de desenho da rede MPLS, considerando $D = 2$ e $H = 3$ (tendo por base o exemplo ilustrado na figura II.1). Note-se que o nodo de acesso 2 desempenha o papel de um nodo de trânsito num dos caminhos entre os nodos de acesso 1 e 3.

$$f_{ij}^{pq} = \text{número de caminhos de } p \text{ para } q \text{ que percorrem o arco } (i, j), \\ (i, j) \in A; p, q \in S;$$

$$x_{\{i,j\}} = \begin{cases} 1, & \text{se a aresta } \{i, j\} \text{ pertence à solução} \\ 0, & \text{c. c.} \end{cases}, \{i, j\} \in E;$$

$$N_i = \begin{cases} 1, & \text{se um nodo de trânsito é instalado no nodo } i \\ 0, & \text{c. c.} \end{cases}, i \in V \setminus S.$$

Note-se que as variáveis f_{ij}^{pq} assim definidas são apenas uma generalização das variáveis f_{ij} definidas para o modelo GEN^D , visto que no problema aqui abordado existem vários serviços, *i.e.*, vários pares de nodos origem/destino ($p, q \in S$) e não apenas um só.

As variáveis $x_{\{i,j\}}$ são necessárias para que o custo de uma aresta $\{i, j\} \in E$ seja contabilizado se houver um número (não nulo) de serviços que a utilizem. Já relativamente às variáveis N_i , estas representam simplesmente a decisão de instalar ou não um nodo de trânsito em cada nodo $i \in V \setminus S$.

Desta forma, podemos escrever a seguinte formulação genérica para o problema

em questão:

Formulação Genérica

$$\text{Min} \sum_{\{i,j\} \in E} c_{\{i,j\}} x_{\{i,j\}} + \sum_{i \in V \setminus S} a_i N_i$$

sujeito a:

$$\{(i, j) \in A : f_{ij}^{pq} > 0\} \text{ contém 2 caminhos-}H\text{-}(p, q), p, q \in S \quad (19)$$

$$f_{ij}^{pq} + f_{ji}^{pq} \leq x_{\{i,j\}}, \{i, j\} \in E; p, q \in S \quad (20)$$

$$\sum_{i:(i,j) \in A} f_{ij}^{pq} \leq N_j, j \in V \setminus S; p, q \in S \quad (21)$$

$$\sum_{i:(i,j) \in A} f_{ij}^{pq} \leq 1, j \in S \setminus \{p, q\}; p, q \in S \quad (22)$$

$$f_{ij}^{pq} \in \mathbb{Z}_0^+, (i, j) \in A; p, q \in S \quad (23)$$

$$x_{\{i,j\}} \in \{0, 1\}, \{i, j\} \in E \quad (24)$$

$$N_i \in \{0, 1\}, i \in V \setminus S. \quad (25)$$

A função objectivo desta formulação expressa o custo total que se pretende minimizar, dado pela soma dos custos associados à utilização de arestas e à instalação de nodos de trânsito. Observe-se que as restrições genéricas **(19)** não são mais do que uma extensão da restrição genérica **(9)** (ver subsecção I.2.1), considerando $D = 2$, para o caso em que temos mais do que um par de nodos origem/destino. Essas restrições garantem que para cada serviço, com nodos origem/destino $p, q \in S$, existem dois caminhos- H -(p, q). As restrições **(20)** garantem que se uma aresta for percorrida (em qualquer sentido) então essa aresta terá que pertencer à solução. Através das restrições **(21)** temos a garantia que um nodo visitado ao longo de um caminho (que suporte um qualquer serviço) terá um nodo de trânsito instalado. Estas restrições em conjunto com as restrições **(22)** garantem que, para qualquer serviço, os $D = 2$ caminhos que o suportam são disjuntos nos nodos. Note-se que, num qualquer caminho de p para q ($p, q \in S$) poderão existir nodos de acesso intermédios. Estes nodos não requerem a instalação de nodos de trânsito, o que se deve ao facto de um nodo de acesso possuir a tecnologia necessária para desempenhar o papel de encaminhamento de um nodo de trânsito. Note-se ainda que disjunção nos nodos implica disjunção

nas arestas, pelo que, caso não fosse necessário decidir pela inclusão ou não de cada aresta na solução, as restrições (20) seriam redundantes. Por último, as restrições (23), (24) e (25) representam o domínio das variáveis definidas anteriormente.

II.1.3 Formulações TRAD-MCF, HOP-MCF e HOP-A-MCF

Como vimos anteriormente nas subsecções I.2.2, I.2.3 e I.2.4, explicitámos a restrição genérica (9) do modelo GEN^D de três formas diferentes (tendo assim obtido os modelos $TRAD^D$, HOP^D e $HOP-A^D$). Como observado acima, note-se que, considerando $D = 2$, as restrições (19) da formulação genérica apresentada na subsecção anterior são uma extensão dessa restrição (9) dado que existem no presente problema diversos pares de nodos origem/destino $p, q \in S$, *i.e.*, vários serviços. Redefinimos então as variáveis y_{ij}^d , z_{ij}^{hd} e w_{ij}^h usadas anteriormente, incluindo índices adicionais associados a esses nodos origem/destino, da seguinte forma:

$$y_{ij}^{dpq} = \begin{cases} 1, & \text{se o arco } (i, j) \text{ é percorrido} & (i, j) \in A; \\ & \text{pelo } d\text{-ésimo } \textit{caminho-H-(p, q)} & , d = 1, 2; \\ 0, & \text{c. c.} & p, q \in S; \end{cases}$$

$$z_{ij}^{hdpq} = \begin{cases} 1, & \text{se o arco } (i, j) \text{ é percorrido} & (i, j) \in A; \\ & \text{exactamente na posição } h & h = 1, \dots, H \\ & \text{pelo } d\text{-ésimo } \textit{caminho-H-(p, q)} & , d = 1, 2; \\ 0, & \text{c. c.} & p, q \in S; \end{cases}$$

$$w_{ij}^{hpq} = \text{número de } \textit{caminhos-H-(p, q)} \text{ que percorrem o arco } (i, j) \\ \text{exactamente na posição } h, (i, j) \in A; h = 1, \dots, H; p, q \in S.$$

Podemos agora, para cada par de nodos $p, q \in S$, explicitar as restrições genéricas (19) de forma análoga à efectuada nas subsecções I.2.2, I.2.3 e I.2.4 relativamente à restrição genérica (9). Denotemos cada um dos submodelos assim obtidos por

$TRAD^{2pq}$, HOP^{2pq} e $HOP-A^{2pq}$, respectivamente. Ou seja, mais concretamente, temos para cada $p, q \in S$:

Modelo $TRAD^{2pq}$

$$\begin{aligned} \sum_{j:(p,j) \in A} y_{pj}^{dpq} &= 1, d = 1, 2 \\ \sum_{j:(i,j) \in A} y_{ij}^{dpq} - \sum_{j:(j,i) \in A} y_{ji}^{dpq} &= 0, i \neq p, q; d = 1, 2 \\ \sum_{j:(j,q) \in A} y_{jq}^{dpq} &= 1, d = 1, 2 \\ \sum_{(i,j) \in A} y_{ij}^{dpq} &\leq H, d = 1, 2 \\ y_{ij}^{dpq} &\in \{0, 1\}, (i, j) \in A; h = 1, \dots, H; d = 1, 2 \\ f_{ij}^{pq} &= \sum_{d=1,2} y_{ij}^{dpq}, (i, j) \in A; \end{aligned}$$

Modelo HOP^{2pq}

$$\begin{aligned} \sum_{j:(p,j) \in A} z_{pj}^{1dpq} &= 1, d = 1, 2 \\ \sum_{j:(i,j) \in A} z_{ij}^{h+1,dpq} - \sum_{j:(j,i) \in A} z_{ji}^{hdpq} &= 0, i \neq p; h = 1, \dots, H - 1; d = 1, 2 \\ \sum_{j:(j,q) \in A} z_{jq}^{Hdpq} &= 1, d = 1, 2 \\ z_{ij}^{hdpq} &\in \{0, 1\}, (i, j) \in A; h = 1, \dots, H; d = 1, 2 \\ z_{qq}^{hdpq} &\in \{0, 1\}, h = 2, \dots, H; d = 1, 2 \\ f_{ij}^{pq} &= \sum_{d=1,2} \sum_{h=1, \dots, H} z_{ij}^{hdpq}, (i, j) \in A; \end{aligned}$$

Modelo $HOP-A^{2pq}$

$$\begin{aligned} \sum_{j:(p,j) \in A} w_{pj}^{1pq} &= 2 \\ \sum_{j:(i,j) \in A} w_{ij}^{h+1,pq} - \sum_{j:(j,i) \in A} w_{ji}^{hpq} &= 0, i \neq p; h = 1, \dots, H - 1 \\ \sum_{j:(j,q) \in A} w_{jq}^{Hpq} &= 2 \\ w_{ij}^{hpq} &\in \{0, 1, 2\}, (i, j) \in A; h = 1, \dots, H \end{aligned}$$

$$w_{qq}^{hpq} \in \{0, 1, 2\}, h = 2, \dots, H$$

$$f_{ij}^{pq} = \sum_{h=1, \dots, H} w_{ij}^{hpq}, (i, j) \in A.$$

Sejam F ($TRAD^{2pq}$), F (HOP^{2pq}) e F ($HOP-A^{2pq}$) os conjuntos de soluções admissíveis destes submodelos para cada $p, q \in S$. Podemos escrever três formulações diferentes para o problema de desenho da rede MPLS aqui abordado, $TRAD-MCF$, $HOP-MCF$ e $HOP-A-MCF$ (onde MCF é a abreviatura de “*multi commodity flow*”), substituindo as restrições genéricas (20) por

$$f_{ij}^{pq}, y_{ij}^{dpq} \in F (TRAD^{2pq}), p, q \in S,$$

$$f_{ij}^{pq}, z_{ij}^{hdpq} \in F (HOP^{2pq}), p, q \in S,$$

e

$$f_{ij}^{pq}, w_{ij}^{hpq} \in F (HOP-A^{2pq}), p, q \in S,$$

respectivamente. Note-se que para quaisquer $(i, j) \in A$ e $p, q \in S$ as variáveis y_{ij}^{dpq} , z_{ij}^{hdpq} e w_{ij}^{hpq} acima definidas se relacionam com as variáveis f_{ij}^{pq} da Formulação Genérica através das igualdades $f_{ij}^{pq} = \sum_{d=1,2} y_{ij}^{dpq}$, $f_{ij}^{pq} = \sum_{d=1,2} \sum_{h=1, \dots, H} z_{ij}^{hdpq}$ e $f_{ij}^{pq} = \sum_{h=1, \dots, H} w_{ij}^{hpq}$, respectivamente, o que nos permite escrever as diferentes formulações sem as variáveis f_{ij}^{pq} . Observe-se que na formulação $HOP-A-MCF$, na presença das restrições (21) e (22) resultantes, podemos restringir as variáveis w_{ij}^{hpq} a valores binários.

Note-se ainda que as formulações $HOP-MCF$ e $HOP-A-MCF$ contêm variáveis lacete, z_{qq}^{hdpq} e w_{qq}^{hpq} (com $h = 2, \dots, H$), respectivamente, para modelar as situações, recorde-se, em que um *caminho-H*-(p, q) tem menos de H arcos (*i.e.*, $z_{jq}^{hdpq} = 1$ para um determinado $d = 1, 2$ ou $w_{jq}^{hpq} \geq 1$, para um $(j, q) \in A, j \in V \setminus \{q\}$ e um $h = 1, \dots, H - 1$).

Por último, note-se que, como já referimos antes no capítulo I, os submodelos $TRAD^{2pq}$, HOP^{2pq} e $HOP-A^{2pq}$ não modelam exactamente dois *caminhos-H*-(p, q). No entanto, na presença de outras restrições ((20)-(25)) e custos, neste caso associados a arestas, nas soluções das formulações $TRAD-MCF$, $HOP-MCF$

e *HOP-A-MCF* resultantes os caminhos obtidos são de facto *caminhos-H*-(p, q).

O próximo resultado II.1 estabelece as relações entre os valores óptimos das relaxações lineares das formulações *TRAD-MCF*, *HOP-MCF* e *HOP-A-MCF*. De facto, note-se que a estrutura destas formulações é semelhante, pelo que basta ter em conta os submodelos relativos à determinação de $D = 2$ *caminhos-H*-(p, q) abordados no capítulo anterior. Recorde-se então que para cada par de nodos origem/destino $p, q \in S$ se provou que $Proj_F \left(F(HOP_L^{Dpq}) \right) \subset Proj_F \left(F(TRAD_L^{Dpq}) \right)$ e $Proj_F \left(F(HOP_L^{Dpq}) \right) = Proj_F \left(F(HOP-A_L^{Dpq}) \right)$ (ver os resultados I.7 e I.8). Deste modo, denotando por $v(TRAD-MCF_L)$, $v(HOP-MCF_L)$ e $v(HOP-A-MCF_L)$ os valores óptimos das relaxações lineares das formulações *TRAD-MCF*, *HOP-MCF* e *HOP-A-MCF*, respectivamente, podemos estabelecer que:

Resultado II.1

$$v(TRAD-MCF_L) \geq v(HOP-MCF_L) = v(HOP-A-MCF_L).$$

Ou seja, as relaxações lineares das formulações *HOP-MCF* e *HOP-A-MCF* fornecem-nos limites inferiores para o valor óptimo do problema de desenho da rede MPLS de qualidade nunca inferior ao limite inferior fornecido pela relaxação linear da formulação *TRAD-MCF*. Como veremos na próxima subsecção, a relação de “ \geq ” deste resultado verifica-se, regra geral, em sentido estrito.

II.1.4 Eliminação de Variáveis

Observe-se agora que nas três diferentes formulações referidas na subsecção anterior existem variáveis supérfluas. Considere-se um serviço com nodo origem p e destino q e um arco $(i, j) \in A$: se a soma dos comprimentos dos caminhos mais curtos em termos do número de saltos de p a i e de j a q for igual ou superior a H , então o arco (i, j) não poderá naturalmente ser percorrido num *caminho-H*-(p, q). Deste modo, as variáveis correspondentes aos referidos serviço e aresta poderão ser eliminadas das diferentes formulações consideradas. No caso em que $i \neq p$ e $j \neq q$, poderemos eliminar

D , $D \times (H - 2)$ e $H - 2$ variáveis, respectivamente, das formulações *TRAD-MCF*, *HOP-MCF* e *HOP-A-MCF* (note-se que $H - 2$ nas expressões anteriores advém do facto de que, uma vez que $i \neq p$ e $j \neq q$, então o arco em questão nunca poderá ser percorrido nem na primeira nem na última posição – as respectivas variáveis não são sequer definidas nas formulações *HOP-MCF* e *HOP-A-MCF*). A figura II.5 ilustra um exemplo em que se considera $H = 4$, um determinado serviço com nodos origem/destino p e q e em que se ignoram as restrições que obrigam os D caminhos a serem disjuntos. Note-se então que todas as variáveis relativas aos arcos (i, j) e (j, k) poderão ser eliminadas. De facto, sendo s_{ij} o comprimento do caminho mais curto entre i e j em termos do número de saltos, note-se que $s_{pi} + s_{jq} = 1 + 3 \geq H$, bem como $s_{pj} + s_{kq} = 2 + 2 \geq H$.

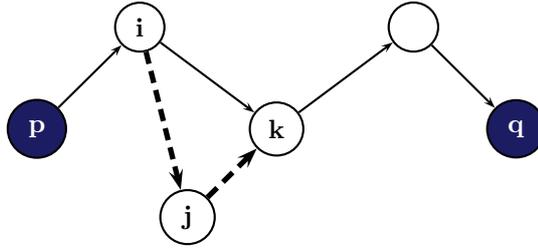


Figura II.3: Ilustração do processo de eliminação das variáveis das formulações *TRAD-MCF*, *HOP-MCF* e *HOP-A-MCF*: considerando $H = 4$, todas as variáveis relativas aos arcos (i, j) e (j, k) podem ser eliminadas.

Mais ainda, dada a informação adicional que as variáveis z_{ij}^{hdpq} e w_{ij}^{hpq} contêm, poderemos eliminar mais variáveis nas formulações *HOP-MCF* e *HOP-A-MCF*. Note-se que no exemplo representado na figura II.5, o arco (i, k) , por exemplo, nunca poderá ser percorrido na posição 3 (para além das posições 1 e $H = 4$, cujas variáveis, como referido anteriormente, não são definidas à partida) pois $s_{kq} = 2$. Assim, as variáveis z_{ik}^{d3pq} , com $d = 1, \dots, D$, e w_{ik}^{3pq} poderão ser também eliminadas.

Assim, torna-se evidente a necessidade de conhecer os comprimentos dos caminhos mais curtos em termos do número de saltos entre qualquer par de nodos em V , para o que se pode utilizar o Algoritmo de Floyd, de forma a excluir das formulações em estudo todas as variáveis supérfluas. Como referido acima, sendo s_{ij} o comprimento do caminho mais curto, em termos de número de saltos, entre os nodos i e j , $\forall i, j \in V$,

podemos então especificar o processo de eliminação de variáveis através do seguinte pseudo-algoritmo:

Para todo o $p, q \in S$:

Para toda a $(i, j) \in A$:

Se $s_{pi} + s_{jq} \geq H$, então:

eliminar as variáveis $y_{ij}^{dpq}, \forall d$

eliminar as variáveis $z_{ij}^{hdpq}, \forall h, \forall d$

eliminar as variáveis $w_{ij}^{hpq}, \forall h$

Para todo o $l = 1, \dots, H$:

Se $s_{jq} \geq l$, então:

eliminar as variáveis $z_{ij}^{hdpq}, H - l < h \leq H, \forall d$

eliminar as variáveis $w_{ij}^{hpq}, H - l < h \leq H$

Para todo o $l = 2, \dots, H$:

Se $s_{pi} \geq l$, então:

eliminar as variáveis $z_{ij}^{hdpq}, 2 \leq h \leq l, \forall d$

eliminar as variáveis $w_{ij}^{hpq}, 2 \leq h \leq l$

Ou seja, por outras palavras: se a condição $s_{pi} + s_{jq} \geq H$ se verificar, significa que ao percorrer o arco (i, j) se efectua um salto adicional além do número máximo de saltos permitido (H), pelo que as variáveis mencionadas poderão ser eliminadas. Adicionalmente, relativamente às formulações *HOP-MCF* e *HOP-A-MCF*, poderemos eliminar mais variáveis: se, para um $l = 1, \dots, H$, $s_{jq} \geq l$, então o arco (i, j) só poderá ser percorrido nas posições inferiores ou iguais a $H - l$; por outro lado, se, para um $l = 2, \dots, H$, $s_{pi} \geq l$, então as posições em que o arco (i, j) poderá ser percorrido são as posições superiores a l .

Note-se que poderá haver alguma redundância seguindo este algoritmo de eliminação de variáveis e, ainda, que no caso em que existe ligação directa entre um par de nodos origem/destino $p, q \in S$, *i.e.*, existe um arco $(i, j) = (p, q)$, o algoritmo continua a ser válido.

Conforme veremos na secção dos resultados computacionais, a eliminação de variáveis nas diferentes formulações de acordo com o algoritmo descrito acima facilitará a resolução de instâncias do problema em questão.

II.1.5 Resultados Computacionais

Neste secção apresentamos os resultados computacionais relativos às formulações *TRAD-MCF*, *HOP-MCF* e *HOP-A-MCF*. Para podermos comparar estas formulações criámos cinco instâncias euclidianas, de modo a representar diferentes redes MPLS.

As redes N25-1, N25-2 e N25-3 têm 25 nodos, enquanto que as redes N50-1 e N50-2 têm 50 nodos. Considerando grelhas de dimensão 4000×4000 ($|V| = 25$) e 5000×5000 ($|V| = 50$), dispuseram-se todos os nodos aleatoriamente nas mesmas. Dadas as distâncias euclidianas entre qualquer par de nodos de cada instância, consideraram-se as arestas cujos nodos extremos não distam mais do que 2000 (por motivos tecnológicos, não convém que as ligações entre nodos tenham um comprimento superior a este valor). Assim, para as instâncias N25-1, N25-2, N25-3, N50-1 e N50-2 criadas, resultaram conjuntos de arestas tais que $|E| = 127$, $|E| = 129$, $|E| = 131$, $|E| = 407$ e $|E| = 427$, respectivamente. O custo de cada aresta $\{i, j\} \in E$ é $c_{\{i,j\}} = d_{\{i,j\}}/100$, em que $d_{\{i,j\}}$ é a distância euclidiana entre os nodos i e j .

Em cada rede gerada desta forma, alguns dos nodos foram seleccionados como nodos de acesso (nodos em S) enquanto que os restantes, como referido antes, são nodos candidatos à instalação de nodos de trânsito. O custo associado à instalação de um nodo de trânsito no nodo i é $a_i = 100$, com $i \in V \setminus S$. Considerámos a selecção de cinco nodos de acesso ($|S| = 5$, o que perfaz 10 serviços a suportar, pois qualquer par de nodos $p, q \in S$, com $p < q$, define um serviço) nas redes N25-1 e N50-1 e dez nodos de acesso ($|S| = 10$, o que perfaz 45 serviços a suportar) nas redes N25-2, N25-3 e N50-2. Em todas as redes, os nodos de acesso seleccionados foram aqueles mais distantes do centro euclidiano da rede, à excepção da rede N25-3 em que dois dos nodos de acesso são os nodos mais próximos do mesmo centro.

Adicionalmente, apenas para garantir que o problema de desenho da rede MPLS terá soluções admissíveis, garantiu-se que as redes criadas contêm, pelo menos, $D = 2$ caminhos- H -(p, q) (com $H \geq 4$) disjuntos nos nodos entre qualquer par de nodos em S . Por último, note-se que o custo associado à utilização de uma aresta $\{i, j\} \in E$ é, no máximo, igual a 20 e que o custo de instalar um nodo de trânsito num nodo $i \in V \setminus S$ é constante, pelo que estes últimos custos serão sempre pelo menos cinco vezes superiores em relação ao custos de utilização de uma aresta (esta relação entre custos prende-se com o tipo de equipamento que é necessário em cada um dos casos). Na figura II.4 encontra-se representada, como exemplo, a rede N25-3.

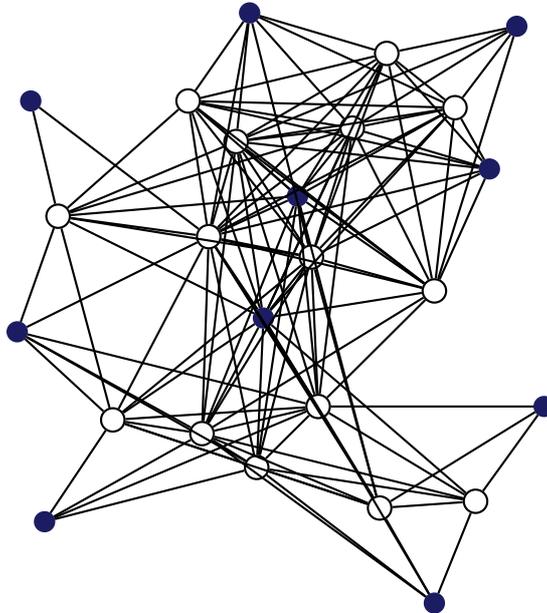


Figura II.4: Instância N25-3: a cheio, os nodos de acesso; os restantes nodos são nodos candidatos à instalação de nodos de trânsito.

Todos os resultados computacionais apresentados nesta subsecção foram obtidos através do CPLEX 7.0, num Pentium III a 450Mhz com 128Mb of RAM, em que se estabeleceu um tempo limite de dois dias para a determinação da solução óptima de cada instância considerando qualquer uma das formulações em estudo.

Para melhor podermos comparar as formulações *TRAD-MCF*, *HOP-MCF* e

HOP-A-MCF, considerámos diferentes valores para H : 4, 5 e 6, uma vez que a maior parte das instâncias são não admissíveis para $H = 3$ e também porque para $H = 7$ o problema resultante é praticamente equivalente ao caso em que não existem restrições de salto. Considerámos ainda as referidas formulações com e sem eliminação de variáveis supérfluas, conforme descrito na subsecção anterior.

Assim, considerando em primeiro lugar as formulações sem eliminação de variáveis, as tabelas II.1 e II.2 contêm a informação que caracteriza as topologias das instâncias de 25 e de 50 nodos, respectivamente. Sendo cada instância resolvida através das formulações *TRAD-MCF*, *HOP-MCF* e *HOP-A-MCF*, apresentamos para cada combinação possível: o *gap* da respectiva relaxação linear (*gap*), o tempo de CPU, em segundos, consumido na determinação da solução óptima da mesma relaxação linear (CPU RL) e ainda o tempo de CPU, também em segundos, gasto na obtenção da solução óptima (CPU ópt.). Na última coluna de cada tabela pode ainda consultar-se o valor óptimo de cada instância.

Nas tabelas II.1 e II.2, relativamente às instâncias de que se desconhece o valor óptimo após dois dias de CPU, são indicados os melhores limites inferior e superior conhecidos ao fim desse tempo.

Repare-se que, de facto, os *gaps* obtidos mostram as relações entre os valores óptimos das relaxações lineares das formulações *TRAD-MCF*, *HOP-MCF* e *HOP-A-MCF* mencionadas no resultado II.1. Observe-se ainda que em todas as instâncias a relação de “ \geq ” do mesmo resultado se verifica no sentido estrito, *i.e.*, $v(\text{TRAD-MCF}_L) > v(\text{HOP-MCF}_L)$.

Em termos do esforço computacional necessário à obtenção das soluções óptimas das formulações em questão, note-se que nas instâncias testadas não se pode dizer que a relaxação linear da formulação *TRAD-MCF* exigiu sempre mais tempo de CPU relativamente às formulações restantes. No entanto, na obtenção da solução óptima inteira, repare-se que a formulação *HOP-MCF* (à excepção da instância N25-1, para $H = 5$ e $H = 6$) consumiu menos tempo de CPU e de forma bastante vincada. Por seu turno, a formulação *HOP-A-MCF* exigiu consistentemente menor esforço computacional relativamente à formulação *HOP-MCF* – recorde-se que, uma vez que

Tabela II.1: Resultados computacionais das instâncias com 25 nodos, considerando as formulações *TRAD-MCF*, *HOP-MCF* e *HOP-A-MCF* sem eliminação de variáveis.

rede	V	S	E	H		TRAD-MCF	HOP-MCF	HOP-A-MCF	ópt.	
N25-1	25	5	127	4	<i>gap</i>	1,4%	0,3%	0,3%	904	
					CPU RL	7,18	0,9	0,33		
					CPU ópt.	662,13	18,21	4,66		
				5	<i>gap</i>	2,1%	0,7%	0,7%		
					CPU RL	2,21	8	2,54		
					CPU ópt.	712,4	278,05	86,21		
	6	<i>gap</i>	1,1%	0,6%	0,6%					
		CPU RL	0,71	8,05	2,51					
		CPU ópt.	501,07	1214,58	159,65					
	N25-2	25	10	129	4	<i>gap</i>	4,2%	0,2%	0,2%	910
						CPU RL	188,64	29,85	10,51	
						CPU ópt.	2 dias	702,57	65,84	
5					<i>gap</i>	3,8%	1,4%	1,4%		
					CPU RL	71,63	211,25	53,9		
					CPU ópt.	2 dias	48598,1	38576,5		
6		<i>gap</i>	3,4%	1,2%	1,2%					
		CPU RL	14,38	561,32	136,16					
		CPU ópt.	2 dias	76340,9	35313,5					
N25-3		25	8+2	131	4	<i>gap</i>	1,8%	0,4%	0,4%	952
						CPU RL	212,75	43,66	15,56	
						CPU ópt.	2 dias	3791,7	644,31	
	5				<i>gap</i>	4,1%	2,0%	2,0%		
					CPU RL	122,04	330,94	110,09		
					CPU ópt.	2 dias	2 dias	85640,2		
	6	<i>gap</i>	—	—	—					
		CPU RL	81,94	559,47	131,83					
		CPU ópt.	2 dias	2 dias	2 dias					

Tabela II.2: Resultados computacionais das instâncias com 50 nodos, considerando as formulações *TRAD-MCF*, *HOP-MCF* e *HOP-A-MCF* sem eliminação de variáveis.

rede	V	S	E	H		TRAD-MCF	HOP-MCF	HOP-A-MCF	ópt.	
N50-1	50	5	407	4	<i>gap</i>	24,7%	0,1%	0,1%	1410	
					CPU RL	185,58	2,43	1,33		
					CPU ópt.	131378,5	3,84	1,49		
					<i>gap</i>	12,2%	0,6%	0,6%		
					5 CPU RL	218,35	88,13	15,1		
					5 CPU ópt.	2 dias	1554,6	377,2		
						1027,55 a 1617				
				6	<i>gap</i>	5,6%	0,8%	0,8%	1040	
					6 CPU RL	157,01	321,41	49,84		
					6 CPU ópt.	2 dias	5212,94	213,72		
							1001,25 a 1072			
					<i>gap</i>	8,6%	0,1%	0,1%		1284
4 CPU RL	3248,5	74,53	34,06							
4 CPU ópt.	2 dias	214,37	71,98							
		1189,69 a -								
N50-2	50	10	427	4	<i>gap</i>	8,6%	0,1%	0,1%	1284	
					4 CPU RL	3248,5	74,53	34,06		
					4 CPU ópt.	2 dias	214,37	71,98		
					<i>gap</i>	4,9%	0%	0%		
					5 CPU RL	1340,56	2648,89	721,67		
					5 CPU ópt.	2 dias	2684,5	468,4		
						1127,12 a 1292				
				6	<i>gap</i>	4,1%	0%	0%	1142	
					6 CPU RL	1266,45	14121,4	2237,7		
					6 CPU ópt.	2 dias	8354,35	1520,93		
							1097,79 a 2998			

$D = 2$, a primeira formulação tem metade das variáveis da segunda.

Por último, observe-se que nas instâncias em que o tempo de CPU máximo de dois dias foi excedido, os melhores limites inferior e superior fornecidos pelo CPLEX das formulações *HOP-MCF* e *HOP-A-MCF* são sempre de melhor qualidade do que aqueles obtidos na formulação *TRAD-MCF*.

Nas tabelas II.3 e II.4 apresentam-se os resultados computacionais das formulações *TRAD-MCF*, *HOP-MCF* e *HOP-A-MCF* cujas variáveis supérfluas foram eliminadas conforme descrito na subsecção II.1.3. Considerando as mesmas instâncias

N25-1 a N50-2 que anteriormente, as tabelas II.3 e II.4 contêm a mesma informação que as tabelas anteriores (II.1 e II.2).

Tabela II.3: Resultados computacionais das instâncias com 25 nodos, considerando as formulações *TRAD-MCF*, *HOP-MCF* e *HOP-A-MCF* com eliminação de variáveis.

rede	V	S	E	H		TRAD-MCF	HOP-MCF	HOP-A-MCF	ópt.					
N25-1	25	5	127	4	<i>gap</i>	0,3%	0,3%	0,3%						
					CPU RL	0,46	0,55	0,27	904					
					CPU ópt.	3,33	18,75	4,54						
					<i>gap</i>	0,7%	0,7%	0,7%						
					5	CPU RL	1,19	5,5	1,8	895				
					CPU ópt.	99,7	300,73	41,72						
					<i>gap</i>	1,1%	0,6%	0,6%						
					6	CPU RL	1,22	9,56	2,2	884				
					CPU ópt.	416,14	431,1	142,97						
					<i>gap</i>	0,2%	0,2%	0,2%						
					N25-2	25	10	129	4	CPU RL	24,43	29,41	10,18	910
										CPU ópt.	145,15	171,15	221,44	
<i>gap</i>	2,4%	1,4%	1,4%											
5	CPU RL	62,13	305,88	68,61						884				
CPU ópt.	2 dias	62767,1	15776,3											
		877,46 a 898												
<i>gap</i>	2,9%	1,2%	1,2%											
6	CPU RL	25,74	670	163,59						872				
CPU ópt.	2 dias	59138,3	20271,8											
		860,54 a 880												
<i>gap</i>	0,4%	0,4%	0,4%											
N25-3	25	8+2	131	4						CPU RL	40,66	42,7	15,22	952
					CPU ópt.	4212,1	3926,9	828,9						
					<i>gap</i>	3,6%	2,0%	2,0%						
					5	CPU RL	115,69	326,68	94,84	952				
					CPU ópt.	2 dias	2 dias	2 dias						
							928,14 a 984	936,66 a 965	947,53 a 956					
					<i>gap</i>	—	—	—						
					6	CPU RL	53,33	901,27	115,24	—				
					CPU ópt.	2 dias	2 dias	2 dias						
							914,5 a 960	917,18 a 962	923,26 a 940					

Tabela II.4: Resultados computacionais das instâncias com 50 nodos, considerando as formulações *TRAD-MCF*, *HOP-MCF* e *HOP-A-MCF* com eliminação de variáveis.

rede	V	S	E	H		TRAD-MCF	HOP-MCF	HOP-A-MCF	ópt.	
N50-1	50	5	407	4	<i>gap</i>	0,4%	0,1%	0,1%	1410	
					CPU RL	1,61	2,16	0,9		
					CPU ópt.	80,1	3,05	0,95		
				5	<i>gap</i>	0,8%	0,6%	0,6%		
					CPU RL	29,83	102,25	16,05		
					CPU ópt.	1616,2	1369,56	303,87		
	6	<i>gap</i>	2,2%	0,8%	0,8%					
		CPU RL	47,39	495,49	79,47					
		CPU ópt.	68046,4	2450,15	1255,64					
	N50-2	50	10	427	4	<i>gap</i>	0,1%	0,1%	0,1%	1284
						CPU RL	77,01	71,62	32,32	
						CPU ópt.	326,57	211,37	88,93	
5					<i>gap</i>	0,7%	0%	0%		
					CPU RL	1285,79	3040,75	653,38		
					CPU ópt.	116704,8	1930,3	429,88		
6		<i>gap</i>	1,8%	0%	0%					
		CPU RL	1100,8	10704,6	1486,5					
		CPU ópt.	2 dias	10724,4	1240,1					

1126,25 a 2662

Em termos da qualidade do limite inferior dado pelo valor óptimo da respectiva relaxação linear, a formulação *TRAD-MCF* sem variáveis supérfluas tornou-se bastante competitiva, especialmente para $H = 3$, uma vez que o limite inferior melhora significativamente (em relação à mesma formulação com as referidas variáveis). O mesmo já não acontece relativamente às formulações *HOP-MCF* e *HOP-A-MCF*: a eliminação de variáveis não melhorou os respectivos limites inferiores. Note-se que os *gaps* obtidos evidenciam uma vez mais o resultado II.1 da subsecção II.1.3 mas a relação de “ \geq ” em sentido estrito ocorre menos frequentemente que no caso das formulações sem eliminação de variáveis.

Relativamente ao esforço computacional necessário à obtenção das soluções óptimas das relaxações lineares das três formulações, verifica-se que a formulação *TRAD-MCF*

foi quase sempre mais rápida que a formulação *HOP-MCF*, tendo sido no entanto comparável à formulação *HOP-A-MCF*. Por outro lado, na obtenção da solução óptima inteira, repare-se que a formulação *HOP-MCF* consumiu geralmente menos tempo de CPU que a formulação *TRAD-MCF*, sendo que por vezes a diferença voltou a ser bastante significativa (à semelhança do que aconteceu com as formulações sem eliminação de variáveis). Por sua vez, a formulação *HOP-A-MCF* exigiu consistentemente menor esforço computacional relativamente à formulação *HOP-MCF*, como anteriormente.

Novamente, à semelhança das tabelas II.1 e II.2, relativamente às instâncias sobre as quais se desconhece o valor óptimo após dois dias de CPU, são apresentados os melhores limites inferior e superior fornecidos pelo CPLEX ao fim desse tempo. Observe-se que nesses casos os limites das formulações *HOP-MCF* e *HOP-A-MCF* são sempre de melhor qualidade do que aqueles obtidos com a formulação *TRAD-MCF*.

De forma a explicar por que razão o processo de eliminação de variáveis supérfluas melhora significativamente os limites inferiores fornecidos pela relaxação linear da formulação *TRAD-MCF* mas não melhora aqueles obtidos com as relaxações lineares das formulações *HOP-MCF* e *HOP-A-MCF*, considere-se a situação em que $H = 2$ e o d -ésimo caminho suportando um serviço com nodos origem/destino p e q . Um ponto extremo fraccionário da relaxação linear da formulação *TRAD-MCF* encontra-se ilustrado na figura II.5.

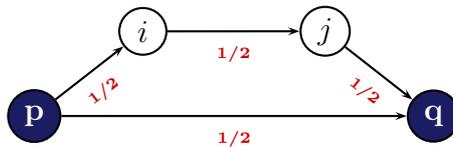


Figura II.5: Solução fraccionária de $TRAD-MCF_L$, considerando $H = 2$. Para um determinado caminho d de p a q , os valores junto a cada arco (i, j) representam o valor da respectiva variável y_{ij}^{dpq} .

Uma vez que $H = 2$, a variável y_{ij}^{dpq} é eliminada de acordo com o processo de eliminação descrito anteriormente (pois $s_{pi} + s_{jq} \geq 2$). No entanto, recorde-se que

para as formulações *HOP-MCF* e *HOP-A-MCF* tal solução (ou seja, um caminho que atravesse a aresta $\{i, j\}$) não é admissível, pois variáveis com índice $h > 2$ não são definidas (no caso, $z_{jq}^{3dpq}, \forall d$, e w_{jq}^{3pq}).

Para concluir esta secção, note-se que os resultados computacionais obtidos com as formulações *TRAD-MCF* e *HOP-A-MCF* sem variáveis supérfluas mostram que, para as instâncias mais fáceis (*i.e.*, aquelas com menores tempos de CPU), os dois modelos são equivalentes pois os tempos de resolução são da mesma magnitude. No entanto, a formulação *HOP-A-MCF* destaca-se por exigir muito menor esforço de CPU nas instâncias mais difíceis: todas as instâncias cujos tempos de resolução excedem 200 segundos através da formulação *TRAD-MCF* foram resolvidas muito mais rapidamente.

II.2 Desenho e Dimensionamento de uma rede MPLS sobre uma rede WDM

Nesta secção abordamos um problema de desenho e dimensionamento de duas camadas de rede em simultâneo: uma camada inferior do tipo *Wavelength Division Multiplexing* (WDM), que confere a transmissão de dados de uma rede MPLS que se encontra numa camada superior. Neste problema pretende-se determinar um subgrafo de custo mínimo que suporte todos os serviços existentes nessa rede conjunta, tendo em conta diversas restrições.

A rede WDM da camada inferior consiste na infra-estrutura sobre a qual assenta a rede MPLS, sendo formada por nodos e arestas de natureza totalmente óptica: os primeiros são designados por OXC (*Optical Cross-Connects*) enquanto que as ligações entre os mesmos são constituídas por fibras ópticas. Entende-se por um *lightpath* uma qualquer concatenação de nodos OXC e fibras ópticas que estabelece a ligação entre um par de OXCs sobre os quais se encontram colocados LSRs (nodos de acesso ou de trânsito da rede MPLS da camada superior). Nodos de trânsito poderão ter que ser instalados sobre alguns OXCs, uma vez que um *lightpath* não poderá ter um comprimento superior a um determinado valor de modo a evitar dispersão de sinal (na subsecção II.2.2 explicar-se-á com maior detalhe esta condição). Cada LSP (*i.e.*, o caminho que suporta cada serviço, estabelecendo a comunicação entre o respectivo nodo de acesso de origem e o de destino) é encaminhado sobre a topologia de *lightpaths* que existir na rede WDM. A figura II.6 representa um LSP entre os nodos de acesso NA_1 e NA_2 na rede MPLS, o qual assenta sobre *lightpaths* da rede WDM (assumindo que o comprimento dos *lightpaths* representados não excede o referido valor máximo), representados a cheio.

O sucesso de redes MPLS sobre WDM prende-se com a partilha de valências tecnológicas de ambas as redes, o que torna possível o suporte de múltiplos serviços com exigentes requisitos de QoS. No entanto, como referido acima, na rede WDM existem restrições de atenuação óptica, relativas ao comprimento máximo que um *lightpath* poderá ter; enquanto que na rede MPLS da camada superior existem restrições de

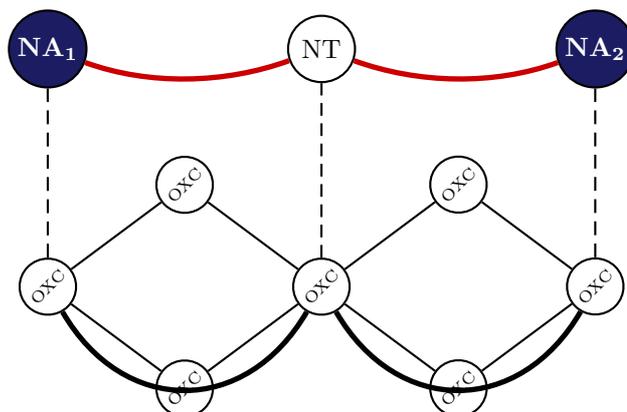


Figura II.6: Um LSP de NA_1 a NA_2 (rede MPLS), utilizando os *lightpaths* da rede WDM (representados a cheio).

QoS e de sobrevivência (como no problema abordado na secção anterior), bem como restrições de largura de banda.

Na subsecção II.2.1 descrevemos em maior detalhe o problema de desenho e dimensionamento em simultâneo de redes MPLS sobre WDM quando dois mecanismos de sobrevivência são adoptados - *Path Diversity* e *Path Protection*. Apresentaremos de seguida, na subsecção II.2.2, a formulação de Programação Linear Inteira para o mesmo problema, bem como resultados computacionais, analisando o impacto dos diferentes mecanismos de sobrevivência no custo total das soluções obtidas. Por último, dada a dificuldade envolvida na determinação da solução óptima de instâncias mais complexas, apresentamos na subsecção II.2.4 uma abordagem heurística para o problema que, como veremos, nos permitirá obter soluções admissíveis de boa qualidade com menor esforço computacional.

II.2.1 Descrição do Problema

Considere-se que a topologia da rede WDM é conhecida e represente-se a mesma por um grafo não orientado $G = (V, E)$, sendo V o conjunto de nodos OXC e E o conjunto de arestas (ligações de fibra óptica entre OXCs). À semelhança da subsecção

II.1.1, seja $S \subset V$ o conjunto de nodos OXC sobre os quais há nodos de acesso (da rede MPLS) sendo que qualquer par de nodos $p, q \in S$, com $p < q$, define um serviço (pelo que, como anteriormente, o número total de serviços a suportar será igual a $|S| \times (|S| - 1)/2$). Os restantes nodos de V são nodos sobre os quais é possível instalar nodos de trânsito. Considere-se ainda que existe um tráfego associado a cada serviço, *i.e.*, existe uma matriz T contendo o valor do tráfego (t_{pq}) entre qualquer par de nodos de acesso $p, q \in S$, com $p < q$. Cada serviço terá que ser suportado através de D caminhos (LSPs) disjuntos nos nodos OXC onde existam nodos de acesso ou de trânsito (da rede MPLS) colocados, sendo que cada um desses caminhos não poderá ter mais do que H saltos. Para além disso, o volume de tráfego total que atravessa cada *lightpath* da rede WDM não pode exceder a capacidade instalada no mesmo, cada *lightpath* não poderá ter um comprimento superior a um determinado valor (L) e um dos mecanismos de sobrevivência terá que ser implementado.

Uma vez que existem custos associados à instalação de nodos de trânsito e à capacidade instalada em cada *lightpath*, o presente problema consiste na determinação dos nodos OXC em que devem ser instalados nodos de trânsito e da capacidade a instalar em cada *lightpaths* que minimiza o custo total, de modo que todos os serviços sejam suportados de acordo com as condicionantes referidas acima.

Uma vez que, nas secções anteriores, foram já explicadas as restrições de salto comuns ao problema abordado nesta secção, descrevemos de seguida os mecanismos de sobrevivência e o modo como estão associados às restrições de sobrevivência e de largura de banda enunciadas em cima.

Em primeiro lugar, relativamente às restrições de sobrevivência, note-se que poderão ocorrer falhas tanto na rede WDM da camada inferior (num nodo OXC ou numa fibra óptica, através de um corte por exemplo) como na rede MPLS (num nodo LSR, de acesso ou de trânsito). Existem diversos trabalhos que consideram mecanismos de protecção de falhas na camada inferior, sendo o denominado “1 : n ” (com $n \geq 1$) o mecanismo mais commumente utilizado. Sob este mecanismo, para cada serviço existente, garante-se a existência de um *lightpath* (de protecção) operacional

caso pelo menos um dos n *lightpaths* que suportam esse serviço falhe. Mais recentemente, o mecanismo *Demand-wise Shared Protection*, proposto em [22], garante que na ocorrência de uma falha num *lightpath*, cada serviço é sempre suportado por uma determinada percentagem de *lightpaths*

Um mecanismo de protecção da rede WDM não previne obviamente a recuperação da rede conjunta (MPLS sobre WDM) na eventualidade de um nodo LSR da camada superior falhar, pelo que é imperativo considerar um mecanismo de protecção nesta camada. Os mecanismos que têm sido propostos neste âmbito em [38] e [28] não são mais do que a generalização do mecanismo “1 : n ” para redes ópticas do tipo WDM.

No presente trabalho, consideraremos apenas mecanismos de protecção da rede MPLS, ou seja, contemplaremos apenas a eventualidade de uma falha de um nodo de trânsito ou de acesso. Como referido antes, uma solução do problema abordado nesta secção consiste numa configuração relativamente à localização de nodos de trânsito e ao dimensionamento de *lightpaths*, de custo mínimo. Esta configuração em conjunto com a topologia da rede WDM poderão ser utilizadas para desenvolver um mecanismo de protecção global, *i.e.*, garantindo a protecção de falhas em ambas as camadas.

Assim, relativamente à rede MPLS, consideramos dois mecanismos de protecção – *Path Diversity* e *Path Protection* – garantindo que esta camada é resistente à falha de um nodo LSR. Sob ambos os mecanismos, terão naturalmente que existir $D > 1$ caminhos disjuntos nos nodos entre qualquer par de nodos de acesso $p, q \in S$. De seguida, descrevemos a forma como esses mecanismos diferem entre si.

Path Diversity é um mecanismo adequado para situações em que, apesar de se pretender proteger o tráfego existente na rede, não existe necessidade de proteger a totalidade do mesmo. Sob este mecanismo, consideramos que o tráfego de cada serviço é igualmente dividido pelos D caminhos disjuntos nos nodos que o suportam (pelo que cada caminho transporta $(1/D) \times 100\%$ do tráfego desse serviço). O modo como esta divisão do tráfego é efectuada apresenta duas vantagens: em primeiro lugar, pode ser facilmente realizada através de um simples algoritmo implementado nos nodos de acesso de origem de cada serviço, nodos estes que têm que processar um elevado número de pacotes; e, em segundo, maximiza a quantidade de tráfego protegido,

mais precisamente $[(D - 1)/D] \times 100\%$ do tráfego total de cada serviço é protegido em caso de falha de um LSR (note-se que, se o tráfego não for igualmente dividido pelos D caminhos, a percentagem de tráfego de facto protegida não é garantidamente máxima).

Path Protection é um mecanismo de sobrevivência que visa essencialmente garantir protecção total do tráfego existente na rede. Sob este mecanismo, se um dos D caminhos que suportam cada serviço deixar de estar operacional devido à falha de um dos LSR ao longo desse caminho, então os restantes $D - 1$ caminhos terão que transportar 100% do tráfego desse serviço (o que é similar ao mecanismo “1 : n ” referido anteriormente). Para tal, torna-se necessário dividir $[1/(D - 1)] \times 100\%$ do tráfego de cada serviço por cada um dos D caminhos que o suportam. Desta forma, estamos a minimizar a capacidade adicional de tráfego (além do prescrito pela matriz T) que cada conjunto de D caminhos transporta. Note-se que essa capacidade adicional de $[1/(D - 1)] \times 100\%$ é de facto a menor possível, tendo em conta o objectivo da protecção total: qualquer outra divisão de tráfego pelos D caminhos que requeira menor capacidade adicional não garante protecção total em caso de falha de um LSR. Sob *Path Protection*, um dos D caminhos é visto como um caminho de protecção (ou de *backup*) e os restantes $D - 1$ caminhos como caminhos de serviço. O tráfego de cada serviço é portanto dividido igualmente pelos caminhos de serviço, o que, à semelhança do mecanismo anterior, pode ser implementado através de um simples algoritmo no respectivo nodo de acesso de origem, quer em condições de operação normais, quer nos casos em que ocorre uma falha de um LSR. Por último, note-se que *Path Protection* pode ser visto como uma generalização do já referido *Demand-wise Shared Protection* aplicado a caminhos disjuntos nos nodos.

Para exemplificar o modo como estes mecanismos de sobrevivência funcionam, considere-se um serviço com nodos de acesso origem e destino $p, q \in S$ respectivamente, com um tráfego associado $t_{pq} = 0.3$, bem como $D = 2$ e $D = 3$. A figura II.7 mostra a quantidade de tráfego que atravessa cada arco sob *Path Diversity* e sob *Path Protection* (primeiros e segundos valores associados a cada arco, respectivamente), para $D = 2$ e $D = 3$ (à esquerda e direita, respectivamente). Note-se que se

um LSR falhar (indicado com “X”), as percentagens de tráfego protegidas sob *Path Diversity* são de 50 e 66% para $D = 2$ e $D = 3$, respectivamente, ao passo que sob *Path Protection* a percentagem de tráfego protegida é naturalmente de 100% para qualquer D .

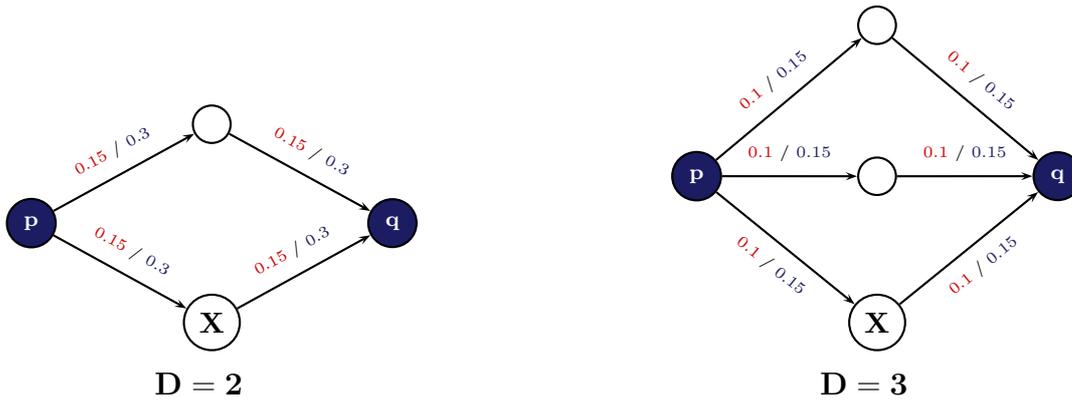


Figura II.7: Quantidade de tráfego que atravessa cada arco sob *Path Diversity* e sob *Path Protection* (primeiros e segundos valores junto a cada arco, respectivamente), com $t_{pq} = 0.3$, para $D = 2$ e $D = 3$.

Adicionalmente, note-se que sob o mecanismo *Path Diversity* a capacidade que existe em excesso na rede é nula para qualquer D . Por outro lado, este exemplo ilustra uma propriedade interessante do mecanismo *Path Protection*: à medida que D aumenta, a capacidade que existe em excesso na rede diminui (para $D = 2$ é necessária uma capacidade adicional de 100%, enquanto que para $D = 3$ o valor dessa capacidade é de apenas 50%, para $D = 4$ seria de 33%, etc). Este facto sugere então que sob este mecanismo, ao considerar $D = 3$, por exemplo, poderemos obter redes de menor custo (visto que existe um custo associado à capacidade instalada em cada *lightpath*) do que com $D = 2$. A figura II.8 ilustra uma situação em que tal pode ocorrer. Considerem-se três nodos de acesso $p, q, r \in S$, que definem três serviços, com tráfegos associados unitários, *i.e.*, $t_{pq}, t_{pr}, t_{qr} = 1$. Considere-se também que a capacidade de um *lightpath* é igualmente unitária e assumam-se que o comprimento de uma aresta entre qualquer par de nodos é igual ao comprimento máximo que um *lightpath* poderá ter. Na figura, linhas a cheio, a picotado e a tracejado representam os caminhos de p para q , de p para r e de q para r , respectivamente.

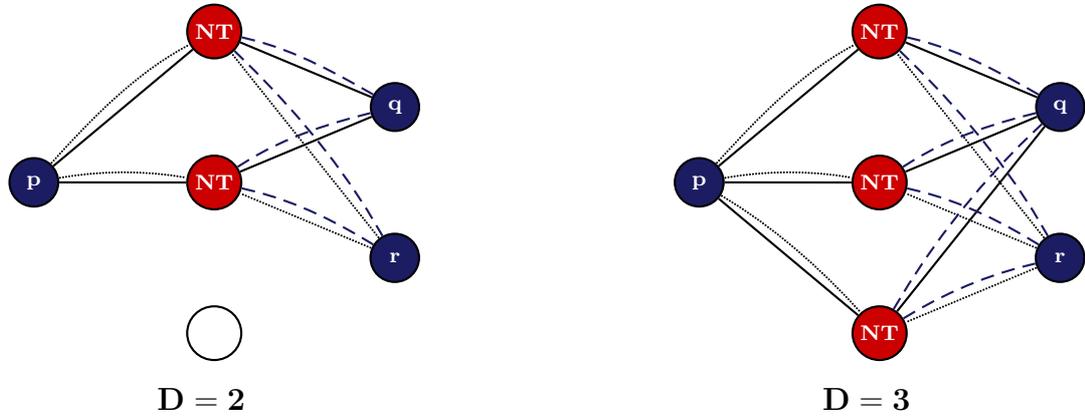


Figura II.8: Ilustração do facto de se poderem obter redes de menor custo à medida que D aumenta, sob o mecanismo *Path Protection*.

Note-se que neste exemplo, para $D = 2$, todo o caminho que suporta cada serviço transporta um tráfego unitário, pelo que o tráfego total que atravessa cada *lightpath* é igual a 2. Sendo a capacidade de cada *lightpath* unitária, serão necessários 12 *lightpaths* no total de forma a suportar todos os serviços existentes na rede. Serão ainda necessários 2 nodos de trânsito (como referido antes, assumimos que o comprimento de uma aresta entre qualquer par de nodos é, neste exemplo, igual ao comprimento máximo que um *lightpath* poderá ter), aos quais está associado um determinado custo. Já para $D = 3$, como qualquer caminho suportando um determinado serviço transporta 50% do tráfego associado a esse serviço, o tráfego total que atravessa cada *lightpath* é unitário, pelo que serão necessários 9 *lightpaths* no total nesta situação. O número de nodos de trânsito necessários é superior (3), em comparação com o caso $D = 2$, mas note-se que a relação entre os custos de instalação de nodos de trânsito e de *lightpaths* pode ser tal que a solução apresentada para $D = 3$ seja de menor custo total (pois utiliza menos 3 *lightpaths*), mesmo sendo o custo de instalação de um nodo de trânsito tipicamente superior. Além disso, repare-se que numa rede mais geral em que existam mais serviços, $D = 3$ pode não exigir necessariamente a instalação de um maior número de nodos de trânsito do que $D = 2$.

O exemplo representado na figura II.8 levanta a seguinte questão: será vantajoso, sob o mecanismo de sobrevivência *Path Protection*, considerar valores maiores que $D = 3$? Note-se que para $D = 2$ os caminhos de cada serviço transportam 200% do tráfego associado, enquanto que, para $D = 3$, os caminhos transportam 150%, o que representa uma redução de 25%. Para $D = 4$, obtemos uma redução adicional de cerca de 11% (uma vez que os caminhos que suportam cada serviço transportam 133% do respectivo tráfego). Torna-se então evidente que ao considerarmos valores para D ainda maiores, a referida redução obtida será cada vez menor, pelo que justificamos assim a razão pela qual no presente trabalho não consideramos valores maiores do que $D = 4$. Acresce ainda à justificação o facto que o esforço de gestão de mais do que quatro LSPs para cada serviço existente numa rede pode tornar-se proibitivo para o operador da rede. Na subsecção dos resultados computacionais veremos que, sob *Path Protection*, se obtêm soluções de menor custo com $D = 3$ do que com $D = 2$ consistentemente, mas o mesmo já não acontece quando consideramos $D = 4$ ao invés de com $D = 3$.

Para finalizar esta subsecção, note-se que, sob ambos os mecanismos de sobrevivência, a divisão igualitária do tráfego de cada serviço pelos D caminhos que o suportam pode ser efectuada nos respectivos nodos de origem através de um simples algoritmo *Round-robin* (segundo este algoritmo, os pacotes de cada serviço ocupam uma fila de espera própria e a transmissão de dados é efectuada dando prioridade ao serviço cujos pacotes estão em fila de espera há mais tempo). De facto, a recepção desordenada dos pacotes nos nodos destino não constitui um problema pois o atraso de recepção dos mesmos é minimizado através das restrições de salto.

Na próxima subsecção, dados os resultados obtidos no capítulo anterior, respeitantes às relações existentes entre os submodelos $TRAD^D$, HOP^D e $HOP-A^D$, bem como tendo em conta os resultados computacionais obtidos no estudo do problema da secção II.1, este segundo problema será modelado com base apenas no submodelo $HOP-A^D$. Efectuamos um estudo comparativo considerando diferentes valores para D , avaliando o impacto dos diferentes mecanismos de sobrevivência no custo total

das soluções óptimas obtidas para este problema.

II.2.2 Formulação HOP-A-MCF-II

Sendo a rede WDM representada através do grafo $G = (V, E)$, onde V representa o conjunto de nodos OXC e E o conjunto de arestas, $S \subset V$ o conjunto de nodos OXC sobre os quais estão colocados os nodos de acesso da rede MPLS e $V \setminus S$ o conjunto de nodos OXC em que é possível instalar nodos de trânsito, note-se que existe uma distância física entre OXC's e esses nodos da camada superior MPLS que, por ser usualmente bastante pequena, neste trabalho consideramos negligenciável.

Começamos então por abordar as restrições de atenuação óptica referidas na secção anterior: o comprimento máximo de um *lightpath* não poderá exceder um determinado valor, que denotaremos por L . Quer isto dizer que, considerando um qualquer serviço com origem p e destino q , a concatenação de fibras ópticas e OXC's (formando um *lightpath*) em cada um dos D caminhos que suportam esse serviço só pode ser efectuada se a soma dos comprimentos das fibras ópticas não exceder L . Caso exceda, é necessário instalar um nodo de trânsito ou haver um nodo de acesso $r \neq p, q$ (um nodo de acesso desempenhar as funções de um nodo de trânsito entre um par de diferentes nodos de acesso) num dos OXC's dessa concatenação de modo que qualquer *lightpath* não ultrapasse o comprimento máximo admissível.

Uma vez que o custo associado a um *lightpath* é proporcional ao seu comprimento, um *lightpath* será sempre instalado ao longo do caminho mais curto entre os respectivos nodos extremo (nodos OXC sobre os quais existem nodos de acesso ou são instalados nodos de trânsito). Desta forma e seguindo [12], ao determinar o caminho mais curto em G entre qualquer par de vértices de V , através do algoritmo de Floyd, e seleccionando aqueles cujo comprimento não excede L , obtemos um grafo expandido $G' = (V, E')$ em que cada aresta corresponde a um caminho em G que satisfaz a restrição de atenuação óptica. Assim, sobre G' , para cada um dos D caminhos que suportam cada serviço, temos apenas que garantir que um nodo de trânsito é instalado ou que existe um nodo de acesso em todo o nodo $j \neq p, q$ que seja visitado

ao longo desse caminho. A vantagem de considerar o grafo expandido G' é que sobre este grafo as restrições de atenuação óptica estão implicitamente garantidas. Apesar de tipicamente se obter $|E'| \gg |E|$, desconhecemos uma descrição dessas restrições em Programação Linear Inteira, pelo que nesta secção trabalharemos doravante sobre G' .

A figura II.9 pretende ilustrar a forma como se obtém o grafo expandido G' , considerando um exemplo com três nodos de acesso ($S = \{p, q, r\}$), em que a comprimento de cada fibra óptica se encontra junto à aresta que a representa e em que se considera $L = 2000$.

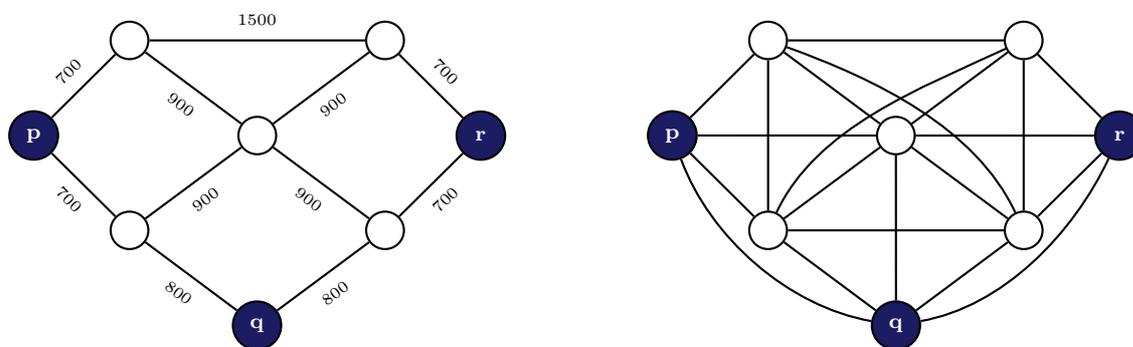


Figura II.9: Modelação das restrições de atenuação óptica: à esquerda, o grafo original G , em que os valores indicados junto a cada aresta representam o comprimento da respectiva fibra óptica; à direita, o grafo expandido G' obtido considerando $L = 2000$.

Frise-se que os mecanismos de sobrevivência considerados nesta subsecção se ocupam apenas com a protecção da rede MPLS: de facto, ao trabalhar sobre G' , duas arestas de E' poderão representar dois caminhos não disjuntos nos nodos OXC em G .

Observe-se então que, sobre G' , apenas precisamos de ter em conta as restrições da rede MPLS, *i.e.*, para qualquer $p, q \in S (p < q)$ terão que existir D caminhos disjuntos nos nodos, cada um dos quais com no máximo H saltos, bem como os mecanismos de sobrevivência mencionados na subsecção anterior.

À semelhança da secção anterior, note-se que havendo a necessidade de distinguir a direcção em que qualquer aresta é atravessada, consideramos o conjunto de arcos A' que contém os “arcos” (i, j) e (j, i) por cada aresta $\{i, j\} \in E'$ (pelo que, no que

se segue, se um arco (i, j) for percorrido, significa que a aresta $\{i, j\}$ é percorrida de i para j).

Deste modo, para modelar adequadamente o presente problema, recordemos a definição das variáveis w_{ij}^{hpq} e N_i já utilizadas na secção anterior (necessárias à determinação dos D caminhos suportando cada serviço e à localização dos nodos de trânsito) e consideremos ainda um conjunto adicional de variáveis necessário ao dimensionamento de cada *lightpath*. Uma vez que neste trabalho consideramos que a capacidade de cada *lightpath* é unimodular, interessa saber apenas a quantidade de *lightpaths* a instalar em cada aresta de G' . Temos então as seguintes variáveis:

$$w_{ij}^{hpq} = \begin{array}{l} \text{número de } \textit{caminhos-H-(p, q)} \text{ que percorrem o arco } (i, j) \\ \text{exactamente na posição } h, (i, j) \in A'; h = 1, \dots, H; \\ p, q \in S; \end{array}$$

$$N_i = \begin{cases} 1, & \text{se um nodo de trânsito é instalado no nodo } i \\ 0, & \text{c. c.} \end{cases}, i \in V \setminus S;$$

$$u_{\{i,j\}} = \text{número de } \textit{lightpaths} \text{ instalados na aresta } \{i, j\}, \{i, j\} \in E'.$$

À semelhança da exposição feita na secção anterior, defina-se $F(HOP-A^{Dpq})$ como o conjunto de soluções admissíveis do submodelo $HOP-A^{Dpq}$, para qualquer $p, q \in S$:

Modelo HOP-A^{Dpq}

$$\begin{aligned} \sum_{j:(p,j) \in A'} w_{pj}^{1pq} &= D \\ \sum_{j:(i,j) \in A'} w_{ij}^{h+1,pq} - \sum_{j:(j,i) \in A'} w_{ji}^{hpq} &= 0, i \neq p; h = 1, \dots, H - 1 \\ \sum_{j:(j,q) \in A'} w_{jq}^{Hpq} &= D \\ w_{ij}^{hpq} &\in \{0, 1, \dots, D\}, (i, j) \in A'; h = 1, \dots, H \\ w_{qq}^{hpq} &\in \{0, 1, \dots, D\}, h = 2, \dots, H. \end{aligned}$$

A formulação que apresentamos de seguida inclui dois parâmetros, α e β : o primeiro é a capacidade de cada *lightpath*, enquanto que o segundo modela o mecanismo de sobrevivência que se pretende implementar. Sendo a_i o custo de instalação de um nodo de trânsito no nodo $i \in V \setminus S$ e $c_{\{i,j\}}$ o custo de instalação de um *lightpath* na aresta $\{i, j\}$, podemos então formular o problema da seguinte forma:

Formulação HOP-A-MCF-II

$$\text{Min} \sum_{\{i,j\} \in E'} c_{\{i,j\}} u_{\{i,j\}} + \sum_{i \in V \setminus S} a_i N_i$$

sujeito a:

$$w_{ij}^{hpq} \in F(HOP-A^{Dpq}), \quad p, q \in S \quad (26)$$

$$\sum_{i:(i,j) \in A'} \sum_{h=1, \dots, H} w_{ij}^{hpq} \leq N_j, \quad p, q \in S; j \in V \setminus S \quad (27)$$

$$\sum_{i:(i,j) \in A'} \sum_{h=1, \dots, H} w_{ij}^{hpq} \leq 1, \quad p, q \in S; j \in S \setminus \{p, q\} \quad (28)$$

$$\sum_{p,q \in S} \beta t_{pq} \left(\sum_{h=1, \dots, H} w_{ij}^{hpq} + \sum_{h=1, \dots, H} w_{ji}^{hpq} \right) \leq \alpha \times u_{\{i,j\}}, \quad \{i, j\} \in E' \quad (29)$$

$$u_{\{i,j\}} \geq 0 \text{ e inteiras}, \quad \{i, j\} \in E' \quad (30)$$

$$N_i \in \{0, 1\}, \quad i \in V \setminus S. \quad (25)$$

A função objectivo desta formulação expressa a minimização do custo associado à instalação de *lightpaths* e de nodos de trânsito. As restrições (26) garantem a existência de D caminhos suportando cada serviço, cada um dos quais com no máximo H saltos (ver subsecção I.2.4). As restrições (27) e (28) obrigam que, para cada serviço, esses D caminhos sejam disjuntos nos nodos (LSR's da rede MPLS): enquanto o primeiro conjunto de restrições garante que se um arco (i, j) for percorrido em alguma posição então terá que se atribuir um valor unitário à variável N_j , ou seja, em cada nodo $j \in V \setminus S$ visitado por um dos D caminhos de p para q terá se instalar um nodo de trânsito; e, uma vez que um nodo de acesso $j \neq p, q$ pode desempenhar as funções de um nodo de trânsito entre p e q , o segundo conjunto de restrições garante que esse nodo j poderá ser visitado no máximo por um dos D caminhos de p para q . As desigualdades (29) são as conhecidas restrições de

network loading, garantindo que um número suficiente de *lightpaths* é instalado na aresta $\{i, j\}$ tendo em conta o tráfego total que a atravessa (em ambos os sentidos). O coeficiente β modela o mecanismo de sobrevivência que se pretende implementar: sob *Path Diversity*, dever-se-á considerar $\beta = 50\%$, $\beta = 33\%$ e $\beta = 25\%$ para $D = 2$, $D = 3$ e $D = 4$, respectivamente; já sob *Path Protection*, dever-se-á fazer $\beta = 100\%$, $\beta = 50\%$ e $\beta = 33\%$ para $D = 2$, $D = 3$ e $D = 4$, respectivamente. As restrições (30) e (25) expressam o domínio das variáveis. Por último, note-se que no sub-modelo $HOP-A^{Dpq}$ as restrições $w_{ij}^{hpq} \in \{0, 1, \dots, D\}$ podem ser rescritas como $w_{ij}^h \in \{0, 1\}$, para quaisquer $(i, j) \in A'$, $h = 1, \dots, H$ e $p, q \in S$, na presença das restrições (27) e (28) da formulação acima.

Adicionalmente, de modo a obter melhores limites inferiores através da resolução da relaxação linear desta formulação, são-lhe acrescentados dois conjuntos de desigualdades válidas, obtidas através da observação das características de *network loading* e dos requisitos de sobrevivência. As primeiras desigualdades consistem em:

$$\sum_{e:\{p,j\} \in E'} u_e \geq \max \left\{ D, \left\lceil \frac{D \cdot \beta \cdot \sum_{q \in S} (t_{pq} + t_{qp})}{\alpha} \right\rceil \right\}, \quad p \in S. \quad (31)$$

Estas restrições garantem que o número de *lightpaths* com um dos nodos extremo no nodo de acesso p terá que ser pelo menos igual ao máximo entre: i) D , pois terão sempre que existir D caminhos suportando qualquer serviço e , por isso, um *lightpath* por caminho; e ii) o menor inteiro maior que a quantidade total de tráfego que flui de e para p dividida pela capacidade de um *lightpath* – este segundo termo da maximização resulta de um conhecido arredondamento de restrições de capacidade válido para restrições de *network loading* (ver, por exemplo, [25]). Nos testes computacionais incluímos todas estas restrições, por se revelarem bastante eficazes em termos da qualidade da relaxação linear e também por serem em número reduzido ($|S|$).

Já o segundo conjunto de desigualdades válidas contém um maior número de restrições. Observe-se que, para qualquer $p, q \in S$, se uma aresta $\{i, j\} \in E'$ é percorrida por um caminho numa qualquer posição, então terá que ser instalado pelo menos um *lightpath* em $\{i, j\}$. Podemos então escrever:

$$u_{\{i,j\}} \geq \sum_{h=1,\dots,H} \left(w_{ij}^{hpq} + w_{ji}^{hpq} \right), \quad p, q \in S; e = \{i, j\} \in E'. \quad (32)$$

Note-se no entanto que estas restrições são eficientes apenas quando o tráfego que flui na aresta $\{i, j\}$ é menor que 1, pois caso contrário estas desigualdades são dominadas pelas restrições (29). O número de restrições (32) que não melhora o limite inferior dado pela relaxação linear da formulação apresentada é, usualmente, bastante elevado. Por esta razão, apenas incluímos no modelo as desigualdades que de facto não são redundantes seguindo um processo iterativo, como se descreve de seguida.

Começamos por resolver a relaxação linear do modelo sem quaisquer restrições (32). Averigua-se, então, quais destas restrições são violadas pela solução óptima determinada, adicionando-as ao modelo anterior. Resolve-se a relaxação linear do novo modelo, obtendo-se uma nova solução óptima. Examinam-se as restantes restrições (32) que poderão ser violadas pela nova solução óptima e, caso o sejam, resolve-se o modelo acrescido dessas restrições. O processo continua assim sucessivamente até que todas as restantes desigualdades (32) sejam verificadas por uma nova solução óptima.

Mais formalmente, sejam *HOP-A-MCF-II* a formulação a resolver sem qualquer restrição (32), *HOP-A-MCF-II_L* a sua relaxação linear e $x^*(HOP-A-MCF-II_L)$ a respectiva solução óptima e, ainda, R o conjunto de todas as restrições (32). Considere-se também o conjunto $R_{x^*(HOP-A-MCF-II_L)} = \{\text{conjunto de restrições de } R \text{ violadas por } x^*(HOP-A-MCF-II_L)\}$. Considere-se ainda U , o sub-conjunto de R formado pelas restrições (32) que pretendemos incluir na formulação. Podemos assim descrever o processo de selecção da seguinte forma:

Inicialização

Calcular $x^*(HOP-A-MCF-II_L)$

$U = \{\}$

Passo 1

Determinar $R_{x^*(HOP-A-MCF-II_L)}$

$$U \leftarrow U \cup R_{x^*(HOP-A-MCF-II_L)}$$

$$R \leftarrow R \setminus R_{x^*(HOP-A-MCF-II_L)}$$

Passo 2

Se $R_{x^*(HOP-A-MCF-II_L)} = \{\}$, então STOP

Caso contrário, acrescentar as restrições de U à formulação $HOP-A-MCF-II$, calcular $x^*(HOP-A-MCF-II_L)$ e voltar ao passo 1.

Após a conclusão deste processo iterativo, a formulação $HOP-A-MCF-II$ encontra-se acrescida das restrições de U , sendo esse modelo resolvido até à optimalidade. Na próxima subsecção relatamos a nossa experiência computacional considerando essa formulação perante diferentes cenários relativamente aos valores de D e T , bem como aos mecanismos de sobrevivência referidos anteriormente.

II.2.3 Resultados Computacionais

Nesta subsecção apresentamos os resultados computacionais relativos à formulação $HOP-A-MCF-II$ (que, recorde-se inclui as restrições **(31)** e um subconjunto de restrições **(32)** seleccionadas conforme o processo descrito atrás). Considerámos como estudos de caso duas redes do tipo WDM: a rede NSFNet (National Scientific Foundation Network), com $|V| = 14$ e $|E| = 23$; e a rede EON (European Optical Network), com $|V| = 19$ e $|E| = 39$. Para ambos os casos, considerámos o comprimento máximo de um *lightpath* é $L = 2500km$. Com base neste valor, determinando os respectivos grafos expandidos (ver subsecção anterior), obtivemos $|E'| = 55$ e $|E'| = 150$, respectivamente para as redes NSFNet e EON.

Relativamente à rede NSFNet, tivemos em conta dois cenários principais nos testes computacionais: $|S| = 6$ e $|S| = 8$, perfazendo 15 e 28 serviços, respectivamente. A figura II.10 representa o grafo expandido (G') da rede NSFNet nos dois cenários mencionados.

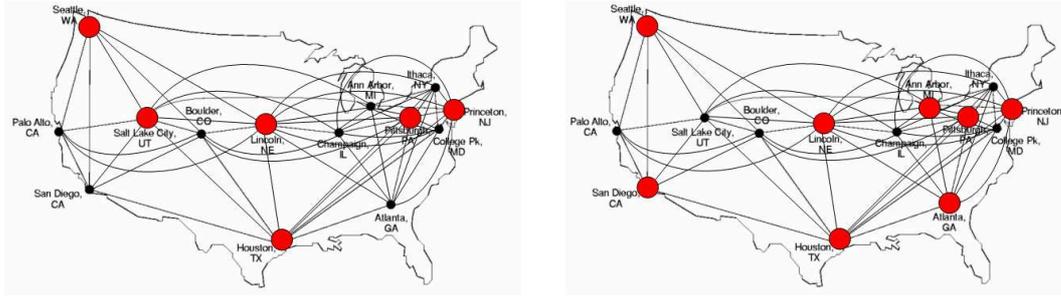


Figura II.10: O grafo expandido (G') da rede NSFNet com $|S| = 6$, à esquerda, e $|S| = 8$, à direita.

À semelhança dos testes computacionais efectuados na secção anterior, considerámos que o custo associado à instalação de um nodo de trânsito no nodo i é $a_i = 100$ ($i \in V \setminus S$). Considerámos ainda que o custo associado à instalação de um *lightpath* sobre a aresta $\{i, j\}$ é dado por $c_{\{i, j\}} = (\text{distância entre } i \text{ e } j) \times 0.03$, $\{i, j\} \in E'$. A distância entre i e j consiste em uma das seguintes hipóteses: i) a distância Euclidiana entre i e j , caso $\{i, j\} \in E$; ou ii) a distância do caminho mais curto entre i e j , caso $\{i, j\} \in E' \setminus E$ (i.e., para arestas que tenham sido acrescentadas ao grafo original G). Assumimos que a capacidade de cada *lightpath* é de $\alpha = 1$, a que corresponde um valor de 10 Gbps (*gigabits* por segundo). De resto, relativamente aos valores de tráfego associados a cada serviço, t_{pq} , $p, q \in S$ ($p < q$), considerámos quatro sub-cenários distintos de forma a poder avaliar a variação do custo das soluções sob diferentes condições de tráfego total. Assim, os valores t_{pq} foram gerados aleatoriamente, seguindo distribuições uniformes nos intervalos $[0, 0.4]$, $[0, 0.6]$, $[0, 0.8]$ e $[0, 1.0]$ (a que correspondem médias de tráfego de 2, 3, 4 e 5 Gbps).

Relativamente às restrições de salto, considerámos um valor fixo para H , visto que o objectivo nesta subsecção consiste em estudar o impacto dos diferentes mecanismos de sobrevivência sob variadas condições no custo das soluções obtidas. Assim, em todos os testes realizados, assumiu-se $H = 4$ (para $H = 3$, a maior parte das diferentes instâncias geradas são não admissíveis). No que toca ao número de caminhos disjuntos nos nodos e os mecanismos de sobrevivência, testámos os casos $D = 2$, $D = 3$ e $D = 4$ sob cada mecanismo.

As diferentes instâncias do problema abordado nesta subsecção foram resolvidas através do CPLEX 9.0, com um tempo limite de três dias, num Pentium IV a 3.4GHz e 1Gb de RAM.

A tabela II.5 contém os resultados computacionais obtidos para a rede NSFNet considerando os dois cenários principais ($|S| = 6$ e $|S| = 8$) e os quatro sub-cenários relativos às diferentes matrizes de tráfego. A tabela contém os valores ótimos das instâncias geradas bem como o respectivo tempo de CPU (em segundos), para cada valor de D considerado e quer sob *Path Diversity* ou *Path Protection*. Para avaliar o impacto (no custo das soluções) de garantir sobrevivência, a tabela contém ainda os valores das mesmas instâncias quando não se exige nenhuma protecção ($D = 1$). Existem instâncias que o CPLEX não conseguiu resolver devido a problemas de memória, para as quais apresentamos os melhores limites inferiores e superiores encontrados ao fim de um determinado tempo (assinalado com “*”).

Para melhor percebermos os valores constantes desta tabela e com a intenção de mais facilmente avaliar o impacto no custo das soluções obtidas ao considerar algum tipo de protecção (baseada num dos mecanismos de sobrevivência e um determinado número de caminhos disjuntos) em detrimento de nenhuma protecção, os gráficos das figuras II.11 e II.12 (relativas aos cenários $|S| = 6$ e $|S| = 8$, respectivamente) apresentam a relação entre esses custos. Ou seja, para cada matriz de tráfego T , cada coluna corresponde ao valor $custo(P_{D,\beta})/custo(P_{1,1})$, onde $P_{D,\beta}$ é a instância em que a matriz de tráfego T é encaminhada sob um dos mecanismos de sobrevivência (assumindo um valor apropriado para β) através de D caminhos disjuntos nos nodos. Note-se que, uma vez que há instâncias cujos valores ótimos são desconhecidos, a respectiva relação entre custos é determinada usando o melhor limite superior conhecido para essa instância (estes casos estão assinalados com “◇”).

Analise-se, em primeiro lugar, os resultados obtidos sob *Path Diversity* (colunas 2, 3 e 4 da tabela II.5 e primeiros três conjuntos de colunas das figuras II.11 e II.12).

Tabela II.5: Resultados computacionais relativos à rede NSFNet, obtidos através da formulação *HOP-A-MCF-II*.

S	tráfego		sem protecção	Path Diversity			Path Protection		
			D = 1	D = 2 $\beta = 50\%$	D = 3 $\beta = 33\%$	D = 4 $\beta = 25\%$	D = 2 $\beta = 100\%$	D = 3 $\beta = 50\%$	D = 4 $\beta = 33\%$
6	[0, 0.4]	valor óptimo	375	381	621	1074	716	681	1074
		CPU	30s	2s	1s	1s	82s	8s	1s
		valor óptimo	485	509	687	1074	952	835	1118
		CPU	51s	14s	9s	1s	65s	35s	3s
6	[0, 0.8]	valor óptimo	516	516	715	1074	1024	867	1132
		CPU	70s	6s	16s	1s	388s	54s	46s
6	[0, 1.0]	valor óptimo	604	668	783	1131	1208	1052	1285
		CPU	10s	10s	18s	43s	19s	115s	1257s
8	[0, 0.4]	valor óptimo	649	722	883	1151	1272	1173	1271
		CPU	4018s	316s	246s	11s	133810s	22249s	123s
		valor óptimo	849	949	1107	1348	–	–	1520
		melhor limite inferior					1634.91	1413.75	
8	[0, 0.6]	melhor limite superior					1698	1453	
		CPU	971s	4980s	2189s	442s	40561s*	49051s*	503s
		valor óptimo	1010	–	1228	1455	–	–	1677
		melhor limite inferior		1057.94			1938.24	1619.39	
8	[0, 0.8]	melhor limite superior		1094			2003	1714	
		CPU	153174s	67290s*	18943s	382s	51790s*	48689s*	7448s
		valor óptimo	1127	1256	–	1629	–	–	–
		melhor limite inferior			1367.99		2283.07	1877.78	1885.25
8	[0, 1.0]	melhor limite superior			1383		2333	1971	1915
		CPU	260s	65515s	66496s*	9823s	38883s*	36805s*	54423s*

Note-se que apenas duas das instâncias não foram resolvidas até à optimalidade, sendo o *gap* entre os melhores limites inferior e superior encontrados inferior a 3.4%. Relativamente ao caso em que nenhum mecanismo de sobrevivência é exigido, o caso $D = 2$ impõe um custo adicional que se situa entre 0 e 10% para $|S| = 6$, e cerca de 10% para $|S| = 8$. Já para o caso $D = 3$, o custo adicional imposto situa-se entre 30 e 65% para $|S| = 6$, e entre 21 e 36% para $|S| = 8$. Finalmente, para $D = 4$, as mesmas percentagens encontram-se entre 87 e 186% ($|S| = 6$) e 44 e 77% ($|S| = 8$). Estes valores mostram que a protecção de 50% do tráfego total (mecanismo *Path Diversity* com $D = 2$) pode ser garantida com um custo adicional de, em média, apenas 7.5% relativamente ao caso em que não se exige qualquer protecção ($D = 1$). Já se pretendermos proteger 66 ou 75% do tráfego total (casos $D = 3$ ou $D = 4$ sob *Path*

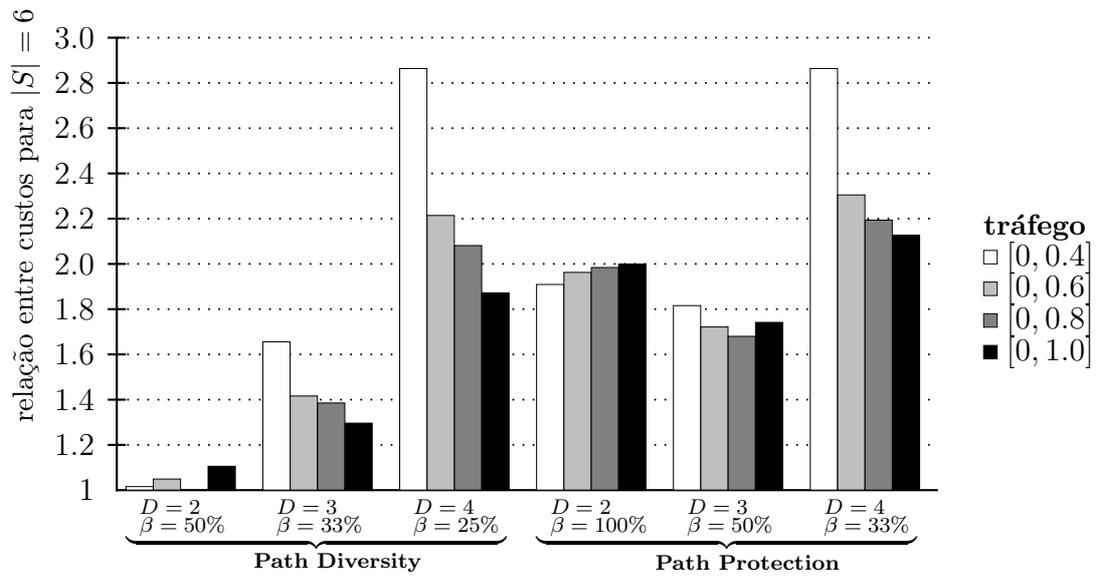


Figura II.11: Relação entre os custos das soluções obtidas sob cada mecanismo de sobrevivência e sob nenhuma protecção (caso $D = 1$), para $|S| = 6$.

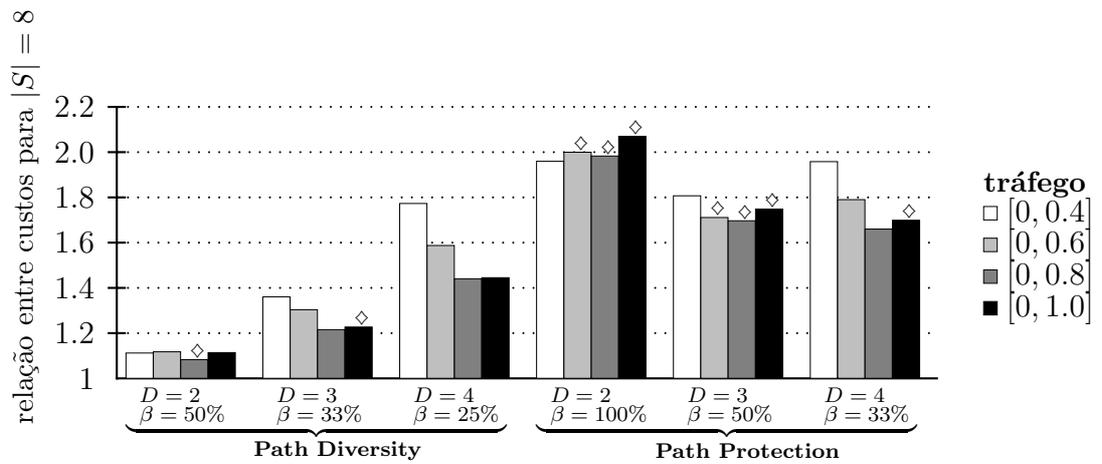


Figura II.12: Relação entre os custos das soluções obtidas sob cada mecanismo de sobrevivência e sob nenhuma protecção (caso $D = 1$), para $|S| = 8$.

Diversity, respectivamente), incorreremos num custo adicional de, em média, 35.8 ou 91%, respectivamente. Naturalmente, caberá ao operador da rede avaliar estes custos adicionais, face às percentagens de tráfego protegidas com $D = 2$, $D = 3$ e $D = 4$.

Analisando agora os resultados obtidos quando se considera o mecanismo de sobrevivência *Path Protection* (colunas 5, 6 e 7 da tabela II.5 e três últimos conjuntos de colunas das figuras II.11 e II.12), comecemos por comparar os casos $D = 2$ e $D = 3$. No cenário $|S| = 6$ (note-se que todas as instâncias foram resolvidas até à optimalidade), os resultados obtidos revelam que, para garantir protecção total através de $D = 2$ ou $D = 3$ caminhos disjuntos nos nodos, se incorre num custo adicional médio (novamente em relação ao caso $D = 1$) de 96.4 ou 74.4%, respectivamente. O que é surpreendente: significa que para o mesmo nível de protecção (100%), as soluções obtidas com $D = 3$, naturalmente mais robustas quando comparadas com as que se obtêm com $D = 2$, têm, em média, menor custo. Mais ainda, note-se que o custo obtido com $D = 3$ é consistentemente menor do que com $D = 2$ em todas as instâncias consideradas.

Para $|S| = 8$, apesar de desconhecermos o valor óptimo da maior parte das instâncias, observe-se que os melhores limites inferiores encontrados com $D = 2$ são sempre superiores aos encontrados com $D = 3$. Usando os melhores limites inferiores do caso $D = 2$, podemos estabelecer um limite inferior para o custo adicional de cerca de 95.8%; já usando os melhores limites superiores do caso $D = 3$, estamos em condições de colocar um limite superior no custo adicional de cerca de 74.1%. Assim, ao exigir três (ao invés de dois) caminhos disjuntos nos nodos significa que obtemos em média soluções de menor custo, o que não deixa de ser novamente surpreendente.

Comparemos agora os resultados obtidos com $D = 3$ e $D = 4$. Para $|S| = 6$, o custo adicional imposto ao considerarmos $D = 4$ é de, em média, 137.3% (e, recorde-se, 77.4% para $D = 3$). Ou seja, um operador que deseje implementar o mecanismo de *Path Protection* através de quatro caminhos disjuntos nos nodos obtém, em média, um agravamento de custos superior ao obtido com $D = 3$. Note-se que esta diferença é consistente não só em média, como também em todas as instâncias testadas. Para $|S| = 8$, apesar de desconhecermos as soluções óptimas de metade das instâncias, recorde-se que, para $D = 3$, obtemos um limite superior do custo adicional de cerca de 74.1% (usando os melhores limites superiores conhecidos), enquanto que para $D = 4$ obtemos um limite inferior do custo adicional de 77.1% (usando os melhores limites inferiores conhecidos). Assim, podemos concluir novamente que, em média, o custo

das soluções com $D = 4$ é superior ao daquelas obtidas com $D = 3$. Note-se no entanto que, uma vez que não se conhecem os valores óptimos de algumas instâncias, esta relação poderá não se verificar para algumas das mesmas. Ainda assim e como foi referido anteriormente, a possível vantagem obtida com $D = 4$ poderá não se justificar face ao esforço de gestão de quatro caminhos para cada serviço existente numa rede por parte do operador da rede.

De uma forma global, tendo em conta o esforço de CPU requerido, podemos afirmar que as instâncias sob o mecanismo de sobrevivência *Path Diversity* são, para o mesmo D , significativamente mais fáceis de resolver do que sob *Path Protection*. Uma possível explicação para que tal aconteça, acreditamos, é o facto do tráfego total ser sempre superior sob o último mecanismo, o que, tendo em conta as restrições de *network loading* (29), torna o problema mais difícil.

Consideremos agora a rede EON. Uma vez que esta rede é bastante mais complexa e, por isso, as instâncias nela baseadas de mais difícil resolução, efectuámos um número reduzido de testes computacionais por forma a podermos ficar com uma noção do comportamento da formulação *HOP-A-MCF-II* em redes maiores. Assim, considerámos apenas o cenário $|S| = 8$ e a matriz de tráfego com valores gerados uniformemente em $[0, 0.8]$. De resto, os custos associados à instalação de *lightpaths* e nodos de trânsito, bem como os parâmetros α e β , são iguais aos considerados anteriormente para a rede NSFNet. A rede original EON encontra-se representada na figura II.13, sendo 150 o número total de arestas em E' (do grafo expandido correspondente).

A tabela II.6 apresenta os resultados computacionais relativos a esta rede. Assim como na tabela anterior, para as instâncias cuja solução óptima não foi possível encontrar devido a problemas de memória ou por o tempo de resolução ter ultrapassado três dias, apresentamos os melhores limites inferior e superior fornecidos pelo CPLEX e assinalamos essas instâncias com “*”.

Uma vez que para a rede EON testámos apenas o cenário $|S| = 8$ com tráfegos em



Figura II.13: O grafo original (G) representando a rede EON, com $|S| = 8$.

Tabela II.6: Resultados computacionais relativos à rede EON, obtidos através da formulação *HOP-A-MCF-II*.

S	tráfego	sem protecção	Path Diversity			Path Protection			
			D = 1	D = 2 $\beta = 50\%$	D = 3 $\beta = 33\%$	D = 4 $\beta = 25\%$	D = 2 $\beta = 100\%$	D = 3 $\beta = 50\%$	D = 4 $\beta = 33\%$
8	[0, 0.8]	valor óptimo	829	–	–	–	–	–	–
		melhor limite inferior		853.07	999.69	1227	1594.71	1317.57	1362.71
		melhor limite superior		880	1041	1232	1660	1384	1459
		CPU	27133s	3 days**	3 days**	3 days**	139471s*	237790s*	258135s*

[0, 0.8], não estamos em condições de efectuar uma análise de custos à semelhança da que foi efectuada para a rede NSFNet. Para além disso, note-se que só conhecemos a solução óptima do caso $D = 1$. Ainda assim, podemos afirmar que os resultados obtidos estão em linha com os da rede NSFNet.

Sob o mecanismo de sobrevivência *Path Diversity*, as melhores soluções inteiras encontradas com $D = 2$, $D = 3$ e $D = 4$ representam limites superiores para os custos adicionais (relativamente ao caso $D = 1$) de cerca de 6.2, 25.6 e 48.6%, respectivamente. Face a estes valores e às percentagens de tráfego protegidas com $D = 2$, $D = 3$ e $D = 4$, cabe ao operador da rede decidir que número de caminhos deve utilizar e que mecanismo de sobrevivência deve implementar.

Sob o mecanismo *Path Protection*, a melhor solução inteira encontrada com $D = 3$ fornece-nos um limite superior para o custo adicional de 66.9%, enquanto que o melhor limite inferior obtido com $D = 2$ representa um limite inferior de 92.4%. Isto é, como vimos para a rede NSFNet, na pior das hipóteses, a redução de custo é significativa quando um operador opta por três caminhos disjuntos nos nodos ao invés de

dois apenas. Acrescente-se ainda que o custo da melhor solução inteira com $D = 4$ é maior que o custo da melhor solução inteira com $D = 3$ (apesar do seu limite inferior ser ligeiramente inferior). Assim, não podemos novamente afirmar que as soluções com $D = 3$ são melhores que as obtidas com $D = 4$ (e recorde-se que mesmo que isso não aconteça, o possível ganho obtido com $D = 4$ pode não compensar a maior complexidade de gestão das redes).

Para finalizar esta secção, observe-se que a formulação *HOP-A-MCF-II* permitiu-nos encontrar a solução óptima da maior parte das instâncias baseadas na rede NSFNet. Os resultados obtidos (melhores limites superiores e inferiores) para as restantes instâncias das redes NSFNet e EON têm *gaps* suficientemente pequenos (o que não impede a avaliação do impacto dos diferentes mecanismos de sobrevivência no custo total das soluções). Ainda que assim seja, o esforço computacional envolvido na obtenção desses *gaps* constituirá certamente um problema em instâncias de maior dimensão e/ou complexidade. Assim, para estes casos, sugerimos uma abordagem heurística como a heurística de duas fases proposta em [12], visto ser facilmente adaptável ao problema com restrições de sobrevivência aqui estudado.

Resumidamente, nesta secção averiguámos a forma como a imposição de mecanismos de sobrevivência (garantida ao nível da rede MPLS, sendo o tráfego existente encaminhado através de D caminhos disjuntos nos nodos) influencia o custo das soluções para o problema de desenho e dimensionamento de redes MPLS sobre WDM aqui apresentado. Para encontrar estas soluções, usámos uma formulação baseada no submodelo *HOP- A^D* apresentado no capítulo I à qual acrescentámos ainda algumas desigualdades válidas. Relativamente aos resultados computacionais, podemos concluir que, sob *Path Protection*, obtemos consistentemente soluções de menor custo com $D = 3$ do que com $D = 2$, mas o custo de soluções com $D = 4$ é, em média, superior ao custo obtido com $D = 3$.

II.2.4 Heurística de Duas Fases

Nesta subsecção apresentamos uma abordagem heurística para o problema de desenho e dimensionamento de redes MPLS sobre WDM, dado que, como vimos na subsecção anterior, as instâncias de maior dimensão e/ou complexidade são bastante difíceis de resolver. Poderá, portanto, haver a necessidade obter soluções admissíveis de boa qualidade num tempo computacional significativamente menor.

A heurística que propomos baseia-se na formulação *HOP-A-MCF-II*, decompondo-a em dois modelos mais simples que são resolvidos sequencialmente. Em primeiro lugar, na fase I, determinamos a localização óptima dos nodos de trânsito considerando apenas as restrições de salto e de disjunção dos D caminhos para cada serviço; seguidamente, tendo em conta a solução óptima da fase I (que, como veremos, leva a uma simplificação do grafo subjacente), na fase II consideramos as restrições de *network loading* determinando o número de *lightpaths* necessário em cada aresta.

Assim, o modelo adequado para a primeira fase é o seguinte:

Fase I

$$\text{Min} \sum_{i \in V \setminus S} a_i N_i$$

sujeito a:

$$w_{ij}^{hpq} \in F(HOP-A^{Dpq}), \quad p, q \in S \quad (26)$$

$$\sum_{i:(i,j) \in A'} \sum_{h=1, \dots, H} w_{ij}^{hpq} \leq N_j, \quad p, q \in S; j \in V \setminus S \quad (27)$$

$$\sum_{i:(i,j) \in A'} \sum_{h=1, \dots, H} w_{ij}^{hpq} \leq 1, \quad p, q \in S; j \in S \setminus \{p, q\} \quad (28)$$

$$N_i \in \{0, 1\}, \quad i \in V \setminus S. \quad (25)$$

Com base na localização dos nodos de trânsito dada pela solução óptima deste modelo (definida através das variáveis N_i), a segunda fase da heurística determina a instalação óptima de *lightpaths* num grafo restrito. Isto é, sendo NT o conjunto de nodos de $V \setminus S$ em que são instalados nodos de trânsito, consideramos um grafo $G'' = (V', E'')$, em que $V' = S \cup NT$ e E'' é o subconjunto de E' cujos elementos têm ambos os extremos em X' . A figura II.14 ilustra com um exemplo, em que $D = 2$,

$H = 2$ e em que supõe que na Fase I são instalados nodos de trânsito nos nodos a e b , como se obtém G'' com $V' = \{p, q, r, a, b\}$ e o correspondente conjunto de arestas $E'' \subset E'$.

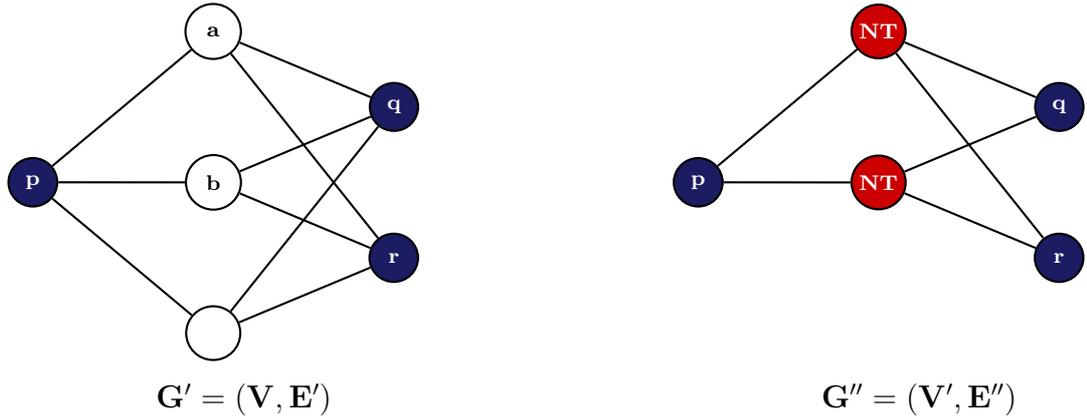


Figura II.14: Transformação do grafo $G' = (V, E')$ no grafo $G'' = (V', E'')$, supondo que dois nodos de trânsito são colocados nos nodos a e b na Fase I.

Realce-se que a solução da Fase I garante que G'' contém D caminhos disjuntos nos nodos, cada um dos quais cumprindo com as restrições de salto, para todos os serviços. Note-se agora que o grafo G'' assim obtido poderá ser substancialmente mais simples que G' . Para além de V' ser formado pelos nodos de acesso e de trânsito apenas, verifica-se usualmente que $E'' \ll E'$. Assim, na Fase II, consideramos as restrições de *network loading* num grafo de menores dimensões, esperando com isso obter soluções admissíveis para o problema de desenho e dimensionamento com menor esforço computacional. Como veremos na secção II.2.5, essas soluções são não só de boa qualidade como são de facto obtidas muito mais facilmente que considerando o modelo (completo) *HOP-A-MCF-II*.

A Fase II pode então ser modelada sobre o grafo $G'' = (V', E'')$ através da seguinte formulação (em que, como anteriormente, por cada aresta $\{i, j\} \in E''$, consideramos que A'' contém os “arcos” (i, j) e (j, i)):

Fase II

$$\text{Min} \sum_{\{i,j\} \in E''} c_{\{i,j\}} u_{\{i,j\}}$$

sujeito a:

$$w_{ij}^{hpq} \in F(HOP-A^{Dpq}), \quad p, q \in S \quad (26')$$

$$\sum_{i:(i,j) \in A''} \sum_{h=1, \dots, H} w_{ij}^{hpq} \leq 1, \quad p, q \in S; j \in S \cup NT \setminus \{p, q\} \quad (28')$$

$$\sum_{p,q \in S} \beta t_{pq} \left(\sum_{h=1, \dots, H} w_{ij}^{hpq} + \sum_{h=1, \dots, H} w_{ji}^{hpq} \right) \leq \alpha \times u_{\{i,j\}}, \quad \{i, j\} \in E'' \quad (29')$$

$$u_{\{i,j\}} \geq 0 \text{ e inteiras, } \{i, j\} \in E''. \quad (30')$$

Observe-se que a função objectivo consiste apenas na soma dos custos de instalação de *lightpaths*. As restrições (26') diferem de (26) no sentido em que este modelo se encontra definido sobre G'' ao invés de G' . Pela mesma razão, as restrições (29) e (30) adaptam-se facilmente, sendo rescritas para cada aresta $e \in E''$, e garantem a instalação de um número suficiente de *lightpaths* tendo em conta o tráfego existente. As restrições (28') – que consistem na adaptação das restrições (28) para os nodos $j \in S \cup NT \setminus \{p, q\}$ – em conjunto com (26') garantem que, sobre G'' , existem D caminhos disjuntos, cada um dos mesmos com não mais que H saltos, para todos os serviços $p, q \in S$.

À semelhança da exposição feita na secção II.2.2, saliente-se que podemos considerar as desigualdades válidas (31) e (32) adaptadas, ou seja, rescrevendo-as com base no grafo G'' subjacente à Fase II. Assim, adicionando todas as primeiras desigualdades e seleccionando um subconjunto das segundas de acordo com o processo descrito na referida secção, conseguimos reduzir mais ainda o esforço computacional na obtenção de soluções óptimas para a Fase II. Os resultados computacionais apresentados na próxima subsecção dizem respeito ao modelo apresentado acima com os conjuntos de desigualdades referidos.

A solução heurística global consiste portanto na localização dos nodos de trânsito dada pela resolução do modelo da Fase I e no dimensionamento das arestas utilizadas no suporte de todos os serviços existentes e na especificação dos respectivos caminhos. O custo total é igual à soma dos custos de ambas as fases.

Claramente que as soluções obtidas segundo este processo são apenas admissíveis. Mais ainda, a nossa experiência computacional indica que na Fase I existem usualmente múltiplas soluções óptimas múltiplas. Nestes casos, apenas foi utilizada uma dessas soluções como ponto de partida para a Fase II, *i.e.*, apesar de G'' poder ter, para a mesma instância, diferentes topologias apenas se considerou aquela que resulta de considerar a solução fornecida pelo CPLEX.

II.2.5 Resultados Computacionais

Para podermos avaliar a qualidade das soluções admissíveis obtidas através da heurística de duas fases proposta na subsecção anterior, consideramos agora as mesmas instâncias que foram testadas com o modelo *HOP-A-MCF-II*, ou seja, as instâncias baseadas nas redes NSFNet e EON. Considerando por isso os mesmos cenários e subcenários para estas redes, como explicado na subsecção II.2.3, bem como os mesmos custo associados à instalação de nodos de trânsito e *lightpaths*, apresentamos de seguida os resultados computacionais obtidos através da heurística.

Considerando então a Fase I, a tabela II.7 resume os resultados obtidos através do modelo apresentado na subsecção anterior. Para cada cenário (note-se que os subcenários de tráfego são ignorados) e para cada D (com $D = 2, 3, 4$), apresentamos o valor óptimo inteiro do modelo, o *gap* (em percentagem) da respectiva relaxação linear e o tempo de CPU (em segundos) que o CPLEX demorou a determinar a solução óptima.

Desta tabela podemos salientar o facto de que em todas as instâncias consideradas a relaxação linear do modelo da Fase I apresenta um *gap* nulo relativamente à solução óptima inteira. Estas soluções são obtidas muito facilmente, com tempos de CPU nunca excedendo um segundo.

Para cada instância, com base na solução óptima da Fase I, construímos o respectivo grafo G'' , conforme descrevemos antes, e passamos à resolução do modelo da

Tabela II.7: Resultados computacionais da Fase I.

rede	S		D = 2	D = 3	D = 4
NSFNet	6	valor óptimo	0	100	300
		gap	0%	0%	0%
		CPU	0.08s	0.08s	0.1s
	8	valor óptimo	0	100	200
		gap	0%	0%	0%
		CPU	0.16s	0.14s	0.1s
EON	8	valor óptimo	0	100	200
		gap	0%	0%	0%
		CPU	0.26s	0.29s	0.27s

Fase II. Como referimos na subsecção anterior, as desigualdades (31) e (32) adaptadas (sobre G'') são válidas para o problema a resolver nesta fase. Adicionando-as ao respectivo modelo de acordo com o processo descrito na secção II.2.2, obtemos os resultados que constam na tabela II.8. Para cada instância, ou seja, considerando todos os cenários e sub-cenários possíveis, assim como os dois mecanismos de sobrevivência com $D = 2, 3, 4$, nessa tabela podemos consultar o seu valor óptimo inteiro e o correspondente tempo (em segundos) de CPU.

De uma forma geral, à semelhança dos resultados obtidos com o modelo (completo) *HOP-A-MCF-II*, verifica-se que as instâncias sob *Path Protection* são mais difíceis de resolver. Verificou-se ainda que para as instâncias cujos tempos de CPU se encontram assinalados com “*”, o CPLEX se deparou com problemas de memória, pelo que são apresentados os melhores limites inferior e superior encontrados ao fim do tempo indicado.

Com base nestes resultados e uma vez que o tempo de CPU necessário à determinação da solução óptima da Fase I é bastante reduzido, estamos em condições de poder afirmar que, de facto, a heurística proposta apresenta soluções admissíveis para o problema em estudo nesta secção com muito menor esforço computacional do que o modelo *HOP-A-MCF-II*. Por exemplo, observe-se que para a instância da rede

Tabela II.8: Resultados computacionais relativos à Fase II.

rede	S	tráfego		Path Diversity			Path Protection			
				D = 2 $\beta = 50\%$	D = 3 $\beta = 33\%$	D = 4 $\beta = 25\%$	D = 2 $\beta = 100\%$	D = 3 $\beta = 50\%$	D = 4 $\beta = 33\%$	
NSF	6	[0,0.4]	valor óptimo	381	521	775	716	581	775	
			CPU	0.21	0.03	0.42	0.5	0.9	0.81	
	6	[0,0.6]	valor óptimo	509	587	775	952	735	819	
			CPU	0.2	0.65	0.68	0.34	1.55	2.09	
	6	[0,0.8]	valor óptimo	516	615	775	1024	767	833	
			CPU	0.1	1.02	0.65	0.95	2.04	2.73	
	6	[0,1.0]	valor óptimo	668	683	832	1208	952	986	
			CPU	0.26	1.34	3.02	0.16	3.97	33.27	
	NSF	8	[0,0.4]	valor óptimo	753	783	951	1317	1081	1071
				CPU	6.5	11.33	2.01	96.54	21.72	7.21
		8	[0,0.6]	valor óptimo	965	1007	1148	1742	1353	1320
				CPU	65.78	78.28	15.62	244	1426.6	10.03
8		[0,0.8]	valor óptimo	1137	1128	1255	2081	–	1477	
			melhor limite superior					1625		
			melhor limite inferior					1594.6		
			CPU	62.52	114.18	8.49	2578.2	16821.9*	907.51	
8		[0,1.0]	valor óptimo	1299	1283	1429	2408	1871	1702	
			CPU	301.42	617.64	460.5	18575.2	1155.5	2919.4	
EON		8	[0,0.8]	valor óptimo	870	–	–	1644	1276	–
				melhor limite superior		937	1065			1226
	melhor limite inferior				927.72	1057.1			1200.48	
	CPU			173.8	16031.2*	122838.3*	1627.5	21781.6	28734.2*	

NSFNet com $|S| = 8$, tráfegos em $[0, 0.4]$, $D = 2$ e sob *Path Protection* o modelo completo demorou 133810 segundos a encontrar a solução óptima enquanto que através da heurística foram gastos apenas cerca de 97 segundos a encontrar uma solução admissível. A qualidade desta (*gap* de 3.5%) e das demais soluções heurísticas pode ser consultada na tabela II.9, onde comparamos o valor dessas soluções (dado pela soma dos custos das duas fases) com os valores óptimos determinados através do modelo completo: para cada instância testada apresentamos ambos os valores e o respectivo *gap* (em percentagem).

Salientamos o facto de que em diversas instâncias o valor da solução heurística é igual ao valor óptimo do modelo *HOP-A-MCF-II*. Nas restantes instâncias, os valores heurísticos apresentam um *gap* de no máximo 7,5%, o que pode ser considerado

Tabela II.9: Comparação dos valores das soluções obtidas através da heurística de duas fases com os das obtidas com o modelo *HOP-A-MCF-II*.

rede	S	tráfego		Path Diversity			Path Protection		
				D = 2	D = 3	D = 4	D = 2	D = 3	D = 4
				$\beta = 50\%$	$\beta = 33\%$	$\beta = 25\%$	$\beta = 100\%$	$\beta = 50\%$	$\beta = 33\%$
NSF	6	[0,0.4]	heurística	381	621	1075	716	681	1075
			gap	0%	0%	0.1%	0%	0%	0.1%
			completo	381	621	1074	716	681	1074
	6	[0,0.6]	heurística	509	687	1075	952	835	1119
			gap	0%	0%	0.1%	0%	0%	0.1%
			completo	509	687	1074	952	835	1118
	6	[0,0.8]	heurística	516	715	1075	1024	867	1133
			gap	0%	0%	0.1%	0%	0%	0.1%
			completo	516	715	1074	1024	867	1132
	6	[0,1.0]	heurística	668	783	1132	1208	1052	1286
			gap	0%	0%	0.1%	0%	0%	0.1%
			completo	668	783	1131	1208	1052	1285
NSF	8	[0,0.4]	heurística	753	883	1151	1317	1181	1271
			gap	4.3%	0%	0%	3.5%	0.7%	0%
			completo	722	883	1151	1272	1173	1271
	8	[0,0.6]	heurística	965	1107	1348	1742	1453	1520
			gap	1.7%	0%	0%	6.5%	2.8%	0%
			completo	949	1107	1348	1635*	1414*	1520
	8	[0,0.8]	heurística	1137	1228	1455	2081	1725**	1677
			gap	7.5%	0%	0%	7.3%	6.5%	0%
			completo	1058*	1228	1455	1939*	1620*	1677
	8	[0,1.0]	heurística	1299	1383	1629	2408	1971	1902
			gap	3.4%	1.1%	0%	5.4%	5.0%	0.8%
			completo	1256	1368*	1629	2284*	1878*	1886*
EON	8	[0,0.8]	heurística	870	1037**	1265**	1644	1376	1426**
			gap	1.9%	3.7%	3.1%	3.1%	4.4%	4.6%
			completo	854*	1000*	1227*	1595*	1318*	1363*

razoável dado o menor esforço computacional envolvido na utilização da heurística. Note-se que para as instâncias assinaladas com “*” e/ou “**” os *gaps* apresentados são na verdade limites superiores dos verdadeiros *gaps*: os valores assinalados com “*” correspondem ao menor inteiro superior ao melhor limite inferior conhecido através

do modelo completo (ver tabelas II.5 e II.6); por outro lado, assinalamos com “**” os melhores limites superiores conhecidos através da heurística de duas fases.

Para finalizar esta subsecção, note-se que os valores obtidos através da heurística proposta nesta secção poderão ser introduzidos como *uppercutoffs* na resolução do modelo *HOP-A-MCF-II*, podendo o esforço computacional envolvido ser assim reduzido.

II.3 Conclusões

Neste capítulo abordamos dois problemas tipicamente denominados na literatura por problemas de *network design*.

Na secção II.1, a nossa atenção recaiu sobretudo na comparação das formulações que se obtêm ao considerar as diferentes formas de modelar as restrições de salto apresentadas no capítulo I. Por outras palavras, nessa secção averiguámos o impacto de considerar diferentes formulações (baseadas nos submodelos $TRAD^D$, HOP^D e $HOP-A^D$) sob o ponto de vista da qualidade dos limites inferiores fornecidos, bem como da rapidez na obtenção das respectivas soluções óptimas inteiras. Com base nos resultados computacionais obtidos, pudemos confirmar o expectável (conforme os resultados teóricos apresentados no capítulo I: a formulação $HOP-A-MCF$ apresenta claras vantagens face às formulações $HOP-MCF$ e $TRAD-MCF$.

Relativamente ao problema de desenho e dimensionamento de duas redes sobrepostas (MPLS sobre WDM) abordado na secção II.2, aproveitámos a eficiente descrição das restrições de salto dada pelo submodelo $HOP-A^D$, apresentando o modelo $HOP-A-MCF-II$. Centrámos então a nossa atenção no modo como diferentes mecanismos de sobrevivência – *Path Diversity* e *Path Protection* – influenciam o custo das soluções desse problema. Dada a dificuldade envolvida na determinação da solução óptima de instâncias de maior dimensão, apresentámos ainda uma abordagem heurística, baseada na decomposição do modelo $HOP-A-MCF-II$ em dois modelos mais simples que são resolvidos sequencialmente, que nos forneceu soluções admissíveis de boa qualidade para o problema e com menor esforço computacional.

Capítulo III

Encaminhamento Ótimo de Tráfego

Neste capítulo abordamos dois problemas de encaminhamento de tráfego numa rede já dimensionada pertencentes à classe de problemas tipicamente denominados na literatura por *traffic engineering*. As redes sobre as quais os problemas se encontram definidos são as resultantes das soluções obtidas na secção II.2. Como vimos, essas soluções resultam da configuração de custo mínimo de duas camadas de rede tendo em conta diversas condições, entre as quais se encontra a necessidade de dimensionar a capacidade de cada *lightpath* dada uma matriz de tráfego T entre qualquer par de nodos origem/destino. Ora esta matriz é usualmente ou uma estimativa (regra geral por excesso) do tráfego que se prevê vir a existir numa rede, ou sujeita a variações ao longo do tempo. Para além destes factores, note-se que os caminhos determinados pelas soluções do problema de dimensionamento podem não ser os caminhos ótimos no que diz respeito a outros critérios, em particular no que se refere à minimização do seu número de saltos.

Assim, dada uma rede já dimensionada, importa saber como encaminhar uma nova matriz de tráfego R satisfazendo as capacidades existentes, bem como restrições de salto e de sobrevivência (*i.e.*, continuando a garantir qualidade de serviço e capacidade de resistência a falhas). Tendo em conta todos os serviços que existirem

numa rede, nesta tese vamos considerar que esse encaminhamento deverá ser óptimo relativamente a um de dois critérios: a minimização do número médio de saltos, ou a minimização do número máximo de saltos. Pretendemos deste modo traduzir o desejo de um operador em garantir que o atraso associado ao suporte de todos os serviços é otimizado, ora numa perspectiva global (considerando o atraso médio de todos os serviços existentes), ora numa perspectiva de pior caso (considerando o serviço com maior atraso associado). Neste capítulo, para além de obviamente tentarmos encontrar boas formulações para os problemas em questão, abordamos duas questões habitualmente consideradas em problemas de *traffic engineering*: o impacto do volume de tráfego total no “custo” (*i.e.*, atraso médio ou máximo associado) das redes e, ainda, o impacto dos erros associados às estimativas de tráfego na admissibilidade das soluções previamente determinadas (estas questões encontram-se explicadas com maior detalhe na secção III.5.3).

Existem muito poucas referências na literatura relativamente à minimização do atraso em problemas de *traffic engineering*, havendo apenas algumas abordagens heurísticas para problemas semelhantes [20, 32, 36, 43]. Existem contudo diversos trabalhos que lidam principalmente com a minimização da capacidade utilizada em cada *lightpath* (ou equivalente), de modo a melhorar a resposta da rede na eventualidade de um falha. No nosso estudo, as restrições de sobrevivência estão implicitamente garantidas pelas soluções obtidas no capítulo anterior (e que servem de base ao problema neste capítulo abordado), pelo que a optimização do número de saltos se torna um critério mais relevante.

As formulações que serão apresentadas neste capítulo baseiam-se nos submodelos HOP^{Dpq} e $HOP-A^{Dpq}$ (ver secção I.2), introduzidos no contexto da determinação de D caminhos com restrições de salto entre um par de nodos. Como observado então, as propriedades desses submodelos conferem claras vantagens às formulações neles baseadas relativamente às formulações baseadas no submodelo tradicional (ver também secção II.1). Neste capítulo a nossa atenção recairá então sobre como desenvolver modelos desagregados e agregados para os problemas de *traffic engineering*,

as relações teóricas entre as duas classes de modelos, bem como uma comparação empírica (através de experiências computacionais) da eficiência dos mesmos modelos.

Como veremos nas próximas secções, as funções objectivo dos modelos desagregados são bastante fáceis de escrever, tanto para a minimização do número médio de saltos como para a minimização do número máximo de saltos. Como mencionado anteriormente, esses modelos têm contudo a desvantagem de utilizarem um conjunto de variáveis para cada um dos D caminhos que se desejarem para cada serviço, pelo que poderemos ter dificuldades em resolver os problemas em estudo à medida que D aumenta. Por outro lado, apesar dos modelos agregados utilizarem menos variáveis (apenas um conjunto de variáveis para cada serviço), o desenvolvimento das funções objectivo correspondentes não será tão imediata: necessitaremos de introduzir variáveis e restrições adicionais para conseguir modelar ambos os problemas. Sendo o processo de desagregação uma ferramenta útil para fortalecer modelos de Programação Linear Inteira, contrariamente ao que se esperaria não há vantagens em utilizar os modelos desagregados. Como veremos, através da nossa experiência computacional, os limites dados pelas relaxações lineares dos modelos desagregados não são, empiricamente, melhores que os limites fornecidos pelas relaxações lineares dos modelos agregados.

O presente capítulo encontra-se dividido do seguinte modo: na próxima secção, começamos por apresentar em maior detalhe os problemas de *traffic engineering* acima resumidos para, depois, apresentar os modelos desagregados e agregados para os mesmos nas secções III.2 e III.3. Para cada classe de modelos, dividimos o nosso estudo conforme expressemos as funções objectivo à custa de variáveis associadas a arcos do grafo original ou de variáveis associadas a lacetes e, para cada caso, apresentamos os modelos adequados para a minimização do número médio de saltos, bem como para a minimização do número máximo de saltos. De seguida, na secção III.4, efectuamos a comparação das relaxações lineares das duas classes de modelos, desagregados e agregados. Apresentamos depois os resultados computacionais obtidos, sob duas perspectivas: nas subsecções III.5.1 e III.5.2 apresentamos dados relativos à eficiência dos diferentes modelos propostos para a minimização do número médio e máximo de

saltos, respectivamente; já na III.5.3 avaliamos o impacto das diferentes variações da matriz de tráfego original sob cada mecanismo de sobrevivência. Terminamos este capítulo com as respectivas conclusões na secção III.6.

III.1 Definição dos Problemas

Considere-se que o grafo não orientado $N = (X, U)$ representa o subgrafo de custo mínimo relativamente ao problema de desenho e dimensionamento de redes MPLS sobre redes WDM, definido sobre o grafo $G' = (V, E')$ (ver secção II.2). O conjunto de nodos X é formado por nodos de acesso e nodos de trânsito, ou seja, $X = S \cup NT$, em que NT denota o conjunto de nodos de $V \setminus S$ em que foram instalados nodos de trânsito. Recordando que as variáveis binárias N_i indicam se um nodo de trânsito é instalado no nodo i , então tem-se $NT = \{i \in V \setminus S : N_i = 1\}$. O conjunto U contém as arestas em que existe pelo menos um *lightpath*, cuja instalação, recorde-se, é necessária ao encaminhamento da matriz de tráfego T , seja sob o mecanismo de sobrevivência *Path Diversity* ou sob *Path Protection*. Sabendo que as variáveis u_e representam o número de *lightpaths* instalados na aresta e , com $e \in E'$, então $U = \{e \in E' : u_e > 0\}$. Sendo então b_e a capacidade instalada em cada aresta $e \in U$, tem-se que $b_e = \alpha u_e$, em que α é a capacidade de cada *lightpath*.

Para todo o serviço com origem em p e destino em q (com $p, q \in S$), N contém naturalmente D caminhos disjuntos nos nodos, cada um dos quais com no máximo H saltos. De entre os D caminhos que suportam cada serviço, considerem-se os Δ caminhos com menor número de saltos, em que $1 \leq \Delta \leq D$. Considerando uma nova matriz de tráfego $R = [r_{pq}]$ ($p, q \in S$, $p < q$), pretendemos então encaminhar cada r_{pq} sobre N através de D caminhos disjuntos nos nodos, cada um dos quais novamente com um número máximo H de saltos, obedecendo a um determinado mecanismo de sobrevivência (*Path Diversity* ou *Path Protection*) e cumprindo com a capacidade b_e instalada em cada aresta. Um encaminhamento óptimo é aquele que minimiza: ou o número médio ou o número máximo de saltos dos Δ caminhos que suportam cada

serviço.

Pelo exposto na subsecção II.2.1, sabemos que sob *Path Diversity* todos os D caminhos transportam tráfego útil, ao passo que sob *Path Protection* apenas os caminhos de serviço o fazem. Sob o ponto de vista de um operador, importa portanto otimizar os caminhos que transportam tráfego útil, pelo que facilmente se constata que o caso $\Delta = D$ corresponderá ao mecanismo de sobrevivência de *Path Diversity*, enquanto que o caso $\Delta = D - 1$ corresponderá a *Path Protection*.

Relativamente à complexidade dos problemas aqui em estudo, Itai *et al*, em [19], provaram que o problema de decisão que consiste em saber se existem D caminhos disjuntos nos nodos, cada um dos quais com no máximo H saltos, é *NP-completo* para $H \geq 4$.

A figura III.1 procura ilustrar os problemas aqui em estudo através de um exemplo com dois serviços (cujos nodos origem/destino são p e q para o primeiro serviço e p' e q' para o segundo), $D = 2$ e $H = 3$.

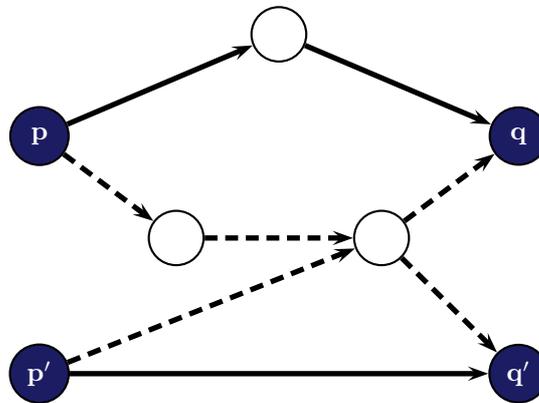


Figura III.1: Uma instância com dois serviços, $D = 2$ e $H = 3$.

Considerando $\Delta = 2$, os dois caminhos que suportam cada serviço estão sujeitos tanto à minimização do número médio de saltos como à minimização do número máximo de saltos. Assim, no primeiro caso obtemos uma média de saltos igual a $((2 + 3) + (2 + 1)) / 4$, enquanto que para o segundo caso o número máximo de saltos

é 3. Já se considerarmos $\Delta = 1$, os caminhos sujeitos a otimização são aqueles representados a cheio e temos: o número médio de saltos igual a $(2 + 1)/2$ e o número máximo de saltos é 2.

III.2 Modelos Desagregados

Tendo por base o modelo HOP^{Dpq} (que, como já foi referido anteriormente, não é mais do que uma extensão do modelo HOP^D – ver subsecção I.2.3 – para diversos pares de nodos origem/destino), recordamos de seguida a definição das variáveis que utilizámos. Note-se, mais uma vez, que havendo a necessidade de distinguir a direcção em que qualquer aresta é atravessada, consideramos o conjunto de arcos A que contém os “arcos” (i, j) e (j, i) por cada aresta $\{i, j\} \in U$. Assim, no que se segue, se um arco (i, j) for percorrido, significa que a aresta $\{i, j\}$ é percorrida de i para j .

$$z_{ij}^{hdpq} = \begin{cases} 1, & \text{se o arco } (i, j) \text{ é percorrido} & (i, j) \in A \cup (q, q); \\ & \text{exactamente na posição } h & h = 1, \dots, H \\ & \text{pelo } d\text{-ésimo caminho-}H\text{-}(p, q) & d = 1, \dots, D; \\ 0, & \text{c. c.} & p, q \in S. \end{cases}$$

Deste modo, adaptamos de seguida as restrições que garantem, para cada serviço $(p, q \in S)$, a existência de D caminhos disjuntos nos nodos, cada um dos quais com no máximo H saltos, sobre $N = (X, U)$. Recorde-se ainda que é necessário garantir que a capacidade instalada em cada aresta $(b_e, e = \{i, j\} \in U)$ não é excedida ao encaminhar uma nova matriz de tráfego R , bem como que um dos mecanismos de sobrevivência, *Path Diversity* ou *Path Protection* (especificados pelo parâmetro β), tem que ser implementado. Deste modo, temos o seguinte sistema de restrições:

$$\sum_{j:(p,j) \in A} z_{pj}^{1dpq} = 1, d = 1, \dots, D; p, q \in S \quad (33a)$$

$$\sum_{j:(i,j) \in A} z_{ij}^{h+1,dpq} - \sum_{j:(j,i) \in A} z_{ji}^{hdpq} = 0, i \neq p; h = 1, \dots, H - 1; d = 1, \dots, D; p, q \in S \quad (33b)$$

$$\sum_{j:(j,q) \in A} z_{jq}^{Hdpq} = 1, d = 1, \dots, D; p, q \in S \quad (33c)$$

$$\sum_{i:(i,j) \in A} \sum_{h=1, \dots, H} \sum_{d=1, \dots, D} z_{ij}^{hdpq} \leq 1, j \in X \setminus \{p, q\}; p, q \in S \quad (34)$$

$$\sum_{p,q \in S} \beta r_{pq} \left(\sum_{h=1, \dots, H} \sum_{d=1, \dots, D} z_{ij}^{hdpq} + \sum_{h=1, \dots, H} \sum_{d=1, \dots, D} z_{ji}^{hdpq} \right) \leq b_e, e = \{i, j\} \in U \quad (35)$$

$$z_{ij}^{hdpq} \in \{0, 1\}, \quad (i, j) \in A; h = 1, \dots, H; d = 1, \dots, D; p, q \in S \quad (36a)$$

$$z_{qq}^{hdpq} \in \{0, 1\}, \quad h = 2, \dots, H; d = 1, \dots, D; p, q \in S. \quad (36b)$$

Note-se agora que, para serviço, podemos incluir as seguintes restrições:

$$\sum_{(i,j) \in A} \sum_{h=1, \dots, H} z_{ij}^{h, d+1, pq} \geq \sum_{(i,j) \in A} \sum_{h=1, \dots, H} z_{ij}^{hdpq}, \quad d = 1, \dots, D - 1; p, q \in S. \quad (37)$$

Estas restrições de *ranking* ordenam os caminhos da forma usual: o d -ésimo caminho não tem mais saltos do que o $(d + 1)$ -ésimo caminho. Como veremos, poderemos prescindir destas restrições nos modelos que serão apresentados, apesar da sua inclusão ser benéfica na resolução de algumas das instâncias testadas.

Como observado anteriormente, estamos interessados em otimizar os Δ melhores caminhos de cada serviço em termos do seu número de saltos (e $\Delta = D$ sob *Path Diversity*, enquanto que sob *Path Protection* $\Delta = D - 1$). No que se segue, abordamos tanto a minimização do número médio de saltos desses Δ caminhos como a minimização do número máximo de saltos dos mesmos Δ caminhos de duas formas diferentes. Ou seja, consideraremos dois tipos de funções objectivo para cada caso: i) escritas com as variáveis associadas aos arcos de N e ii) escritas com as variáveis lacete. Apesar das funções objectivo do primeiro tipo serem mais intuitivas, veremos que as do segundo, assim como algumas restrições adicionais que serão necessárias, usarão um menor número de variáveis.

III.2.1 Funções Objectivo Baseadas em Variáveis Associadas a Arcos

Começamos então por escrever as funções objectivo para a minimização do número médio e máximo de saltos utilizando as variáveis associadas aos arcos do grafo N subjacente.

III.2.1.1 Minimização do Número Médio de Saltos

Neste caso a escrita da função objectivo é imediata uma vez que minimizar o número médio de saltos dos Δ melhores caminhos de cada serviço é equivalente a minimizar a soma do número de saltos. Assim, facilmente escrevemos a seguinte formulação para o problema:

Formulação D-Média-Arcos

$$\text{Min} \sum_{d=1, \dots, \Delta} \sum_{p, q \in S} \sum_{(i, j) \in A} \sum_{h=1, \dots, H} z_{ij}^{hdpq}$$

sujeito a:

$$(33a)-(33c), (34), (35), (36a), (36b).$$

Observe-se que nesta formulação as restrições de *ranking* (37) não são necessárias. De facto a função objectivo garante que na solução óptima desta formulação os Δ caminhos que suportam cada serviço são os caminhos com menor número de saltos. No entanto, note-se que a inclusão das restrições (37) poderá reduzir os tempos de resolução se o conjunto de soluções admissíveis for demasiadamente simétrico. Na prática, através dos resultados computacionais obtidos, observou-se que existem algumas instâncias em que isso acontece, bem como outras instâncias em que se verifica o contrário.

III.2.1.2 Minimização do Número Máximo de Saltos

Neste caso, pretendendo minimizar o número máximo de saltos dos Δ melhores caminhos que suportam cada serviço, aparentemente é necessário incluir as restrições de *ranking* (37), mas veremos que tal não acontece. Considere-se uma variável inteira V que representa o número de saltos do pior caminho, *i.e.*, o caminho com maior número de saltos, dos Δ melhores caminhos que suportam cada serviço. Desta forma, podemos escrever a seguinte formulação:

Formulação D-Máximo-Arcos $Min V$

sujeito a:

$$(33a)-(33c), (34), (35), (36a), (36b)$$

$$\sum_{(i,j) \in A} \sum_{h=1, \dots, H} z_{ij}^{hdpq} \leq V, \quad d = 1, \dots, \Delta; p, q \in S \quad (38)$$

$$V \geq 0 \text{ e inteira.} \quad (39)$$

Note-se que através da função objectivo desta formulação e na presença das restrições (38) e (39) a variável V tomará o valor pretendido pois para cada serviço essa variável é minorada pelo número total de saltos do d -ésimo caminho, com $d = 1, \dots, \Delta$. Ou seja, os piores caminhos de entre os Δ melhores caminhos de cada serviço limitam assim a variável V , fazendo com que esta assuma o valor apropriado. Observe-se que de facto as restrições *ranking* (37) não são necessárias.

Na secção III.4 apresentamos uma versão mais forte desta formulação, ou seja, uma formulação cuja relaxação linear fornece um melhor limite inferior para o valor óptimo do problema.

III.2.2 Funções Objectivo Baseadas em Variáveis Associadas a Lacetes

Nesta subsecção apresentamos as formulações para os problemas em estudo, tendo por base as variáveis lacete z_{qq}^{hdpq} definidas anteriormente. Começemos por observar que as formulações apresentadas na secção anterior satisfazem as igualdades

$$\sum_{(i,j) \in A} z_{ij}^{hdpq} + z_{qq}^{hdpq} = 1, \quad h = 1, \dots, H; d = 1, \dots, D; p, q \in S, \quad (40)$$

visto que numa qualquer posição h , com $h = 1, \dots, H$, de um qualquer caminho que suporte um determinado serviço terá que haver um arco ou lacete, o que resulta da construção dos grafos expandidos associados ao submodelo HOP^D (ver secção I.2.3).

Estas igualdades podem agora ser utilizadas para obter formulações equivalentes

às da secção anterior em termos das respectivas relaxações lineares, de forma que a função objectivo da formulação *D-Média-Arcos*, bem como a função objectivo e as restrições (38) da formulação *D-Máximo-Arcos* possam ser escritas à custa das variáveis lacete. Esperamos desta forma conseguir obter melhores tempos de CPU na resolução de instâncias dos problemas em causa, uma vez que as funções objectivo e restrições serão rescritas com um menor número de variáveis.

Por último, antes de passar à apresentação dessas formulações, observe-se que as restrições de *ranking* (37) podem ser rescritas à custa das variáveis lacete, da seguinte forma:

$$\sum_{h=2,\dots,H} z_{qq}^{h,d+1,pq} \leq \sum_{h=2,\dots,H} z_{ij}^{hdpq}, \quad d = 1, \dots, D - 1; p, q \in S. \quad (41)$$

De facto, se pretendemos que o d -ésimo caminho não tenha mais saltos do que o $(d+1)$ -ésimo caminho, podemos modelar consegui-lo exigindo que o d -ésimo caminho efectue mais lacetes sobre o nodo destino respectivo do que o $(d+1)$ -ésimo caminho.

III.2.2.1 Minimização do Número Médio de Saltos

Neste caso, comecemos por observar que somando as restrições (40) para todo o $h = 1, \dots, H$, $d = 1, \dots, D$ e $p, q \in S$ se obtém

$$\sum_{d=1,\dots,D} \sum_{p,q \in S} \sum_{h=1,\dots,H} \sum_{(i,j) \in A} z_{ij}^{hdpq} = \sum_{d=1,\dots,D} \sum_{p,q \in S} \sum_{h=1,\dots,H} (1 - z_{qq}^{hdpq}). \quad (42)$$

Assim, a minimização de $\sum_{d=1,\dots,\Delta} \sum_{p,q \in S} \sum_{(i,j) \in A} \sum_{h=1,\dots,H} z_{ij}^{hdpq}$ na formulação *D-Média-Arcos* é equivalente à maximização de $\sum_{d=1,\dots,\Delta} \sum_{p,q \in S} \sum_{h=1,\dots,H} z_{qq}^{hdpq}$. Obtemos deste modo a seguinte formulação equivalente:

Formulação D-Média-Lacetes

$$Max \sum_{d=1, \dots, \Delta} \sum_{p, q \in S} \sum_{h=1, \dots, H} z_{qq}^{hdpq}$$

sujeito a:

$$(33a)-(33c), (34), (35), (36a), (36b).$$

Similarmente ao modelo *D-Média-Arcos* e respectivas restrições de *ranking* (37), verifica-se que, apesar da inclusão das restrições (41) (ou as próprias restrições (37)) poder tornar o modelo *D-Média-Lacetes* mais eficiente se o conjunto de soluções admissíveis for demasiadamente simétrico, a nossa experiência computacional revelou que existem instâncias em que isso acontece e outras instâncias em que se verifica o contrário.

III.2.2.2 Minimização do Número Máximo de Saltos

Usando de novo as restrições (40), desta feita considerando $d = 1, \dots, \Delta$, as restrições (38) do modelo *D-Máximo-Arcos* podem ser rescritas como $\sum_{h=1, \dots, H} (1 - z_{qq}^{hdpq}) \leq V$, $d = 1, \dots, \Delta; p, q \in S$. Fazendo $V' = H - V$, obtemos assim um modelo equivalente para a minimização do número máximo de saltos usando as variáveis lacete:

Formulação D-Máximo-Lacetes

$$Max V'$$

sujeito a:

$$(33a)-(33c), (34), (35), (36a), (36b)$$

$$\sum_{h=2, \dots, H} z_{qq}^{hdpq} \geq V', \quad d = 1, \dots, \Delta; p, q \in S \quad (43)$$

$$V' \geq 0 \text{ e inteira.} \quad (44)$$

À semelhança da observação efectuada sobre a formulação *D-Máximo-Arcos*, note-se que ao maximizarmos V' e na presença das restrições (43), (44), essa variável representa o número de lacetes efectuados pelo pior caminho dos Δ melhores caminhos que suportam cada serviço.

III.3 Modelos Agregados

As formulações discutidas na secção anterior são, como referimos, relativamente fáceis de escrever devido à existência de D conjuntos de variáveis por cada serviço existente (um conjunto para cada caminho), o que é uma desvantagem à medida que o valor de D aumenta. Nesta secção, apresentamos modelos agregados, ou seja, modelos que contêm um único conjunto de variáveis por serviço independentemente do número de caminhos pretendido. Dado o menor número de variáveis, esperamos que se torne mais fácil encontrar as soluções óptimas (tanto inteiras quanto lineares) das instâncias dos problemas de *traffic engineering* aqui em estudo. No entanto, uma vez que não conseguimos distinguir os diferentes caminhos de cada serviço (pois temos apenas um conjunto de variáveis por serviço), as funções objectivo já não serão tão imediatas quanto as do capítulo anterior. De facto, para as conseguirmos escrever precisaremos de definir dois novos conjuntos de variáveis, que nos permitirão, para cada serviço, i) contar o número de arcos que são usados numa determinada posição h e ii) verificar se existem arcos numa certa posição h .

À semelhança da secção anterior, começamos por recordar as variáveis utilizadas no submodelo $HOP-A^{Dpq}$ (ver secção I.2.4) que está base das formulações que serão apresentadas. Como mencionámos antes, consideramos o conjunto de arcos A que contém os “arcos” (i, j) e (j, i) por cada aresta $\{i, j\} \in U$.

$$w_{ij}^{hpq} = \text{número de caminhos-}H\text{-}(p, q) \text{ que percorrem o arco } (i, j) \\ \text{exactamente na posição } h, (i, j) \in A \cup (q, q); h = 1, \dots, H; p, q \in S.$$

Desta forma, as seguintes restrições garantem: i) a existência de D caminhos disjuntos nos nodos, cada um dos quais com no máximo H saltos para cada serviço; ii) o cumprimento da capacidade instalada em cada aresta e iii) o encaminhamento da matriz de tráfego R segundo *Path Diversity* ou *Path Protection*:

$$\sum_{j:(p,j) \in A} w_{pj}^{1pq} = 1, p, q \in S \quad (45a)$$

$$\sum_{j:(i,j) \in A} w_{ij}^{h+1,pq} - \sum_{j:(j,i) \in A} w_{ji}^{hpq} = 0, i \neq p; h = 1, \dots, H-1; p, q \in S \quad (45b)$$

$$\sum_{j:(j,q) \in A} w_{jq}^{Hpq} = 1, p, q \in S \quad (45c)$$

$$\sum_{i:(i,j) \in A} \sum_{h=1, \dots, H} w_{ij}^{hpq} \leq 1, j \in X \setminus \{p, q\}; p, q \in S \quad (46)$$

$$\sum_{p,q \in S} \beta r_{pq} \left(\sum_{h=1, \dots, H} w_{ij}^{hdpq} + \sum_{h=1, \dots, H} w_{ji}^{hpq} \right) \leq b_e, e = \{i, j\} \in U \quad (47)$$

$$w_{ij}^{hpq} \in \{0, 1, \dots, D\}, (i, j) \in A; h = 1, \dots, H; p, q \in S \quad (48a)$$

$$w_{qq}^{hpq} \in \{0, 1, \dots, D\}, h = 2, \dots, H; p, q \in S. \quad (48b)$$

Relembre-se que as restrições (45a)-(45c) resultam da agregação em $d = 1, \dots, D$ das restrições (33a)-(33c) apresentadas na secção anterior e que a relação entre as variáveis z_{ij}^{hdpq} e w_{ij}^{hpq} é dada por $\sum_{d=1, \dots, D} z_{ij}^{hdpq} = w_{ij}^{hpq}$.

Note-se que, contrariamente à secção anterior, não conseguimos escrever restrições de *ranking* similares a (37) (ou (41)) precisamente porque, como observado antes, as variáveis w_{ij}^{hpq} não fazem a distinção entre os D caminhos que suportam cada serviço. Note-se ainda que na maior parte dos problemas existe a necessidade dessas supostas restrições de *ranking* mas, como veremos nas próximas subsecções, conseguiremos modelar os mesmos problemas à custa das variáveis adicionais já mencionadas (mas ainda não definidas).

III.3.1 Funções Objectivo Baseadas em Variáveis Associadas a Arcos

Nesta subsecção completamos as formulações para ambos os problemas em estudo, expressando as respectivas funções objectivo em termos das variáveis (w_{ij}^{hpq}) associadas aos arcos do grafo subjacente.

III.3.1.1 Minimização do Número Médio de Saltos

Como referimos antes, para modelar a maior parte dos problemas em estudo usando modelos agregados necessitamos de definir variáveis adicionais. Neste caso, consideremos para cada serviço $p, q \in S$ e cada posição $h = 1, \dots, H$ as seguintes variáveis:

U^{hpq} = número de arcos “interessantes” na posição h do *caminho- H* -(p, q).

Entendemos por arcos “interessantes” aqueles que estão incluídos nos melhores Δ caminhos, pois são estes os caminhos que se pretendem otimizar. A soma $\sum_{h=1, \dots, H} U^{hpq}$ representa então o número de saltos efectuados pelos Δ melhores caminhos de cada serviço, pelo que facilmente se conclui que a função objectivo para a minimização do número médio de saltos será $Min \sum_{p, q \in S} \sum_{h=1, \dots, H} U^{hpq}$.

Recorde-se agora que, para uma dada posição h , todos os arcos são interessantes sob *Path Diversity* ($\Delta = D$) e, sob *Path Protection* ($\Delta = D - 1$), esses arcos são todos menos um (pois podemos obviamente assumir que se existem k arcos nessa posição h , então um dos arcos está incluído no pior caminho, *i.e.*, aquele com maior número de saltos). Generalizando, podemos ainda afirmar que, para uma dada posição h , todos os arcos menos $D - \Delta$ são interessantes, visto que $D - \Delta$ arcos estão incluídos nos piores caminhos que não são contabilizados na minimização da média de saltos dos Δ melhores caminhos.

Desta forma, podemos obter um modelo agregado válido para o problema em questão incluindo as novas variáveis U^{hpq} e as restrições que as definem:

Formulação A-Média-Arcos

$$Min \sum_{p, q \in S} \sum_{h=1, \dots, H} U^{hpq}$$

sujeito a:

$$\begin{aligned} & (45a)-(45c), (46), (47), (48a), (48b) \\ & U^{hpq} \geq \sum_{(i, j) \in A} w_{ij}^{hpq} - (D - \Delta), \quad h = 1, \dots, H; p, q \in S \end{aligned} \quad (49)$$

$$U^{hpq} \in \{0, 1, \dots, \Delta\}. \quad (50)$$

Note-se então que o significado das novas variáveis U^{hpq} é garantido através da função objectivo e das restrições (49) e (50): elas garantem que, para cada serviço $p, q \in S$ e cada posição h , se existirem k arcos nessa posição (*i.e.*, $\sum_{(i,j) \in A} w_{ij}^{hpq} = k$) então $U^{hpq} = \max \{k - (D - \Delta), 0\}$. O segundo termo na expressão de maximização deve-se ao caso em que, sob *Path Protection* ou outros mecanismos para os quais $\Delta < D - 1$, se tem $\sum_{(i,j) \in A} w_{ij}^{hpq} = 0$ (note-se que pelas restrições (50) as variáveis U^{hpq} são inteiras e não negativas).

Observe-se agora que, pelas restrições de ligação (49), ter-se-á sempre $U^{1pq} = \Delta$, pelo que não é necessário escrever essas restrições para $h = 1$. O mesmo se pode dizer relativamente à função objectivo, pois o termo correspondente pode ser tratado como uma constante.

Por último, note-se ainda que, uma vez que sob *Path Diversity* $\Delta = D$, então as novas variáveis U^{hpq} não são necessárias. De facto, é possível escrever um modelo mais simples pelo simples facto de todos os D caminhos serem relevantes, bastando para tal considerar a (intuitiva) função objectivo $\text{Min} \sum_{p,q \in S} \sum_{h=1, \dots, H} \sum_{(i,j) \in A} w_{ij}^{hpq}$.

III.3.1.2 Minimização do Número Máximo de Saltos

Neste caso, começamos por observar que não há necessidade em identificar o serviço que é suportado por um dos Δ melhores caminhos com maior número de saltos. Deste modo, consideramos apenas as seguintes variáveis binárias:

$$V^h = \begin{cases} 1, & \text{se o número de arcos na posição } h, \\ & \text{de pelo menos um caminho, é } \geq (D - \Delta) + 1 \text{ , } h = 1, \dots, H. \\ 0, & \text{c. c.} \end{cases}$$

Sob *Path Diversity* ($\Delta = D$), a existência de pelo menos um arco na posição h significa que há um caminho com pelo menos h saltos. Já sob *Path Protection* ($\Delta = D - 1$), se, para um determinado h , $w_{ij}^{hpq} \geq 2$, significa que pelo menos um dos Δ caminhos tem pelo menos h saltos (uma vez que, como referido anteriormente, um

dos arcos pode estar a ser utilizado pelo caminho de *backup*, ou seja, pelo caminho não relevante). Assim, quer sob *Path Diversity* ou *Path Protection*, estamos interessados em determinar o número de tais posições, de forma a poder calcular o número de saltos do pior caminho.

Note-se que os argumentos apresentados são válidos pois sabemos que, se um qualquer caminho tem um arco na posição h , então, pelas restrições de conservação de fluxo (45a)-(45c), tem também um arco nas posições l , com $l = 1, \dots, h - 1$. Estamos então em condições de escrever o modelo agregado para o presente problema, da seguinte forma:

Formulação A-Máximo-Arcos

$$\text{Min} \quad \sum_{h=1, \dots, H} V^h$$

sujeito a:

$$(45a)-(45c), (46), (47), (48a), (48b)$$

$$\sum_{(i,j) \in A} w_{ij}^{hpq} - (D - \Delta) \leq \Delta \times V^h, \quad h = 1, \dots, H; p, q \in S \quad (51)$$

$$V^h \in \{0, 1\}. \quad (52)$$

À semelhança do modelo apresentado na subsecção anterior, note-se que não é necessário escrever as restrições (51) para $h = 1$, pois nesta posição terá sempre que haver D arcos para cada serviço e, por isso, ter-se-á sempre $V^1 = 1$. Consequentemente, este termo pode ser tratado como uma constante na função objectivo.

O significado das variáveis V^h é garantido através da função objectivo e das restrições (51) e (52). De facto, para cada posição h , temos a garantia que, se houver pelo menos $(D - \Delta) + 1$ arcos nessa posição (*i.e.*, $\sum_{(i,j) \in A} w_{ij}^{hpq} \geq (D - \Delta) + 1$, para algum serviço), então $V^h = 1$.

Por último, note-se que, sob *Path Protection*, quando $\sum_{(i,j) \in A} w_{ij}^{hpq} \leq 1$ para todos os serviços $p, q \in S$, as restrições (51) são satisfeitas tanto com $V^h = 0$ como com $V^h = 1$. No entanto, a função objectivo garante que $V^h = 0$, que é o valor correcto (pois nessa situação todos os $\Delta = D - 1$ melhores caminhos de todos os serviços têm menos que h arcos).

III.3.2 Funções Objectivo Baseadas em Variáveis Associadas a Lacetes

Nesta secção, apresentamos as versões dos modelos agregados para cada um dos problemas aqui abordados expressando as funções objectivo à custa das variáveis lacete w_{qq}^{hpq} .

Começamos por observar que as restrições (45a)-(45c), (48a) e (48b) nos permitem afirmar que o número de lacetes efectuados numa determinada posição h num qualquer nodo destino $q \in S$ é sempre igual à diferença entre D e o número total de arcos $(i, j) \in A$ percorridos nessa mesma posição. Ou seja, as seguintes igualdades são sempre satisfeitas:

$$\sum_{(i,j) \in A} w_{ij}^{hpq} + w_{qq}^{hpq} = D, \quad h = 1, \dots, H; p, q \in S. \quad (53)$$

Assim, à semelhança da exposição feita na secção III.2.2, utilizaremos estas igualdades para obter modelos equivalentes (em termos da sua relaxação linear) para os problemas em estudo e tal que as restrições (49) e (51) sejam escritas à custa das variáveis lacete. Como referimos anteriormente, a potencial vantagem da rescrita dos modelos dessa forma é que tanto a função objectivo como as restrições utilizarão menos variáveis e, por isso, o esforço computacional envolvido na sua resolução poderá ser menor.

III.3.2.1 Minimização do Número Médio de Saltos

Com base nas variáveis U^{hpq} definidas para a formulação *A-Média-Arcos*, considere-se a mudança de variável $V^{hpq} = \Delta - U^{hpq}$, para todo o $p, q \in S$ e $h = 1, \dots, H$. Desta forma, as variáveis V^{hpq} ($h = 1, \dots, H; p, q \in S$) representam o número de lacetes ‘interessantes’ (ou seja, efectuados pelos Δ melhores caminhos) na posição h . Naturalmente, então, que minimizar $\sum_{p,q \in S} \sum_{h=1, \dots, H} U^{hpq}$ é equivalente a maximizar

$\sum_{p,q \in S} \sum_{h=1, \dots, H} V^{hpq}$. Usando agora as igualdades (53), podemos rescrever o modelo *A-Média-Arcos* da seguinte forma:

Formulação A-Média-Lacetes

$$\text{Max} \sum_{p,q \in S} \sum_{h=1, \dots, H} V^{hpq}$$

sujeito a:

$$(45a)-(45c), (46), (47), (48a), (48b)$$

$$V^{hpq} \leq w_{qq}^{hpq}, \quad h = 1, \dots, H; p, q \in S \quad (54)$$

$$V^{hpq} \in \{0, 1, \dots, \Delta\}. \quad (55)$$

À semelhança da observação efectuada sobre a formulação *A-Média-Arcos*, note-se que não é necessário escrever tanto as restrições (54) como a função objectivo para $h = 1$, desta feita porque as variáveis lacete encontram-se definidas “apenas” para $h \geq 2$ e, por isso, ter-se-á sempre $V^{1pq} = 0$.

III.3.2.2 Minimização do Número Máximo de Saltos

Neste caso, considerando as variáveis binárias V^h que foram definidas para a formulação *A-Máximo-Arcos*, consideremos a mudança de variável $Y^h = 1 - V^h$ para qualquer $h = 1, \dots, H$. Naturalmente então que minimizar $\sum_{h=1, \dots, H} V^h$ se torna equivalente a maximizar $\sum_{h=1, \dots, H} Y^h$ e, usando de novo as igualdades (53), podemos escrever a seguinte formulação:

Formulação A-Máximo-Lacetes

$$\text{Max} \sum_{h=1, \dots, H} Y^h$$

sujeito a:

$$(45a)-(45c), (46), (47), (48a), (48b)$$

$$\Delta \times Y^h \leq w_{qq}^{hpq}, \quad h = 1, \dots, H; p, q \in S \quad (56)$$

$$Y^h \in \{0, 1\}. \quad (57)$$

As variáveis Y^h , também elas binárias, indicam se em todos os serviços o número de lacetes efectuados sobre os respectivos nodos destinos na posição h é pelo menos Δ (o que significa que o pior dos Δ melhores caminhos de cada serviço tem no máximo $h - 1$ arcos).

Usando o mesmo argumento que na formulação da subsecção anterior, note-se que não é necessário escrever as restrições **(56)** nem a função objectivo para $h = 1$, pois ter-se-á sempre $Y^1 = 0$.

III.4 Comparação das Duas Classes de Modelos

Nesta secção procedemos à comparação dos modelos desagregados e agregados, em termos das suas relaxações lineares, naturalmente separando o estudo para cada um dos problemas abordados neste capítulo. A nossa atenção foca-se apenas nos modelos baseados em arcos, uma vez que os modelos baseados em lacetes são equivalentes e também porque os primeiros são de mais fácil compreensão.

III.4.1 Minimização do Número Médio de Saltos

Querendo então comparar a relaxação linear do modelo *D-Média-Arcos* com a relaxação linear do modelo *A-Média-Arcos*, recordamos agora o modelo desagregado para esta função objectivo:

Formulação D-Média-Arcos

$$\text{Min} \sum_{d=1, \dots, \Delta} \sum_{p, q \in S} \sum_{(i, j) \in A} \sum_{h=1, \dots, H} z_{ij}^{hdpq}$$

sujeito a:

$$(33a)-(33c), (34), (35), (36a), (36b).$$

De forma a estabelecer a relação entre os dois modelos mais facilmente, começamos por criar um modelo desagregado diferente mas com a garantia que o limite inferior da relaxação linear do novo modelo é igual ao de *D-Média-Arcos_L*. Para tal, utilizaremos as variáveis U^{hpq} definidas para o modelo agregado, recordando que essas variáveis representam, para cada serviço, o número de arcos “interessantes” na posição h . Naturalmente, tem-se:

$$U^{hpq} = \sum_{d=1, \dots, \Delta} \sum_{(i, j) \in A} z_{ij}^{hdpq}, \quad h = 1, \dots, H; p, q \in S. \quad (58)$$

Ao adicionar as igualdades (58) ao modelo *D-Média-Arcos*, podemos obter o novo modelo desagregado rescrevendo a função objectivo da seguinte forma:

$$\text{Min} \sum_{p,q \in S} \sum_{h=1,\dots,H} U^{hpq}. \quad (59)$$

De facto, como mencionado acima, verifica-se facilmente que os limites inferiores dados pelas relaxações lineares do novo modelo e de *D-Média-Arcos* são iguais. Na realidade, podemos até substituir as igualdades das restrições (58) por desigualdades de “ \geq ”, pois na presença da função objectivo (59) a solução óptima da relaxação linear do novo modelo satisfaz sempre essas restrições na igualdade. De seguida, no entanto, construímos uma versão mais fraca deste novo modelo. Observe-se que a relaxação linear da formulação *D-Média-Arcos_L* implica as seguintes desigualdades:

$$\sum_{d=\Delta+1,\dots,D} \sum_{(i,j) \in A} z_{ij}^{hdpq} \leq D - \Delta, \quad h = 1, \dots, H; p, q \in S.$$

Estas desigualdades são, em verdade, bastante intuitivas pois, para cada serviço, o número de caminhos não sujeitos ao critério de optimização é igual a $D - \Delta$. Assim, em cada posição h , o número total de arcos (“não interessantes”, digamos) nessa posição nunca poderá exceder $D - \Delta$.

Se, para o mesmo par $p, q \in S$ bem como para o mesmo h , somarmos estas desigualdades com as respectivas restrições (58), obtemos as seguintes restrições mais fracas:

$$\sum_{d=1,\dots,D} \sum_{(i,j) \in A} z_{ij}^{hdpq} \leq U^{hpq} + (D - \Delta), \quad h = 1, \dots, H; p, q \in S. \quad (60)$$

Através destas restrições e na presença da função objectivo (59), as variáveis U^{hpq} podem ser interpretadas da mesma forma que anteriormente, pelo que podemos substituir as restrições (58) por (60) no novo modelo. Este modelo mais fraco, que denotamos por *WD-Média-Arcos*, é por isso válido para o problema. Note-se que as restrições (58) implicam (60) apenas quando $D - \Delta > 0$; se $D = \Delta$, as restrições (58) e (60) representam as mesmas restrições, o que significa que os modelos *D-Média-Arcos* e *WD-Média-Arcos* são, nesse caso, equivalentes.

Note-se agora que, no caso em que $D - \Delta > 0$, então o modelo *WD-Média-Arcos_L*

poderá fornecer um limite inferior de pior qualidade (do que aquele que se obtém considerando as restrições (58) no modelo D -Média-Arcos $_L$). Em verdade, podemos verificar que existem soluções admissíveis para WD -Média-Arcos $_L$ que não o são relativamente a D -Média-Arcos $_L$ (ou seja, $F(WD\text{-Média-Arcos}_L) \subset F(D\text{-Média-Arcos}_L)$), resolvendo a relaxação linear do modelo mais fraco com a função objectivo

$$\text{Min} \sum_{h=1, \dots, H} \sum_{p, q \in S} \left(U^{hpq} - \sum_{d=1, \dots, \Delta} \sum_{(i, j) \in A} z_{ij}^{hdpq} \right).$$

Se o valor desta função objectivo for negativo para uma qualquer instância, então podemos concluir que pelo menos uma das parcelas é negativa, ou seja, para algum par $p, q \in S$ e alguma posição h , uma das desigualdades (58) não é satisfeita pela relaxação linear de WD -Média-Arcos. De facto, foi isto que aconteceu em algumas instâncias. No entanto, como veremos na secção III.5, para todas as instâncias consideradas nos testes computacionais ambos os modelos forneceram iguais limites inferiores, o que nos leva a desconfiar que os modelos em questão fornecem o mesmo limite inferior para o problema em estudo. Naturalmente, este “facto” não invalida o que mencionámos antes, nomeadamente que $F(WD\text{-Média-Arcos}_L) \subset F(D\text{-Média-Arcos}_L)$, pois esta relação indica-nos apenas que existem funções objectivo para as quais o limite inferior dado pelo modelo mais fraco poderá ser de qualidade inferior. A questão que levantamos aqui é que para as funções objectivo que definem o problema em estudo, os dois modelos podem produzir o mesmo valor.

Podemos resumir estas observações no seguinte resultado:

Resultado III.1

- i) Se $D = \Delta$, $F(WD\text{-Média-Arcos}_L) = F(D\text{-Média-Arcos}_L)$.*
- ii) Se $D > \Delta$, $F(WD\text{-Média-Arcos}_L) \subset F(D\text{-Média-Arcos}_L)$.*
- iii) Se $D > \Delta$, permanece em aberto se $v(WD\text{-Média-Arcos}_L) = v(D\text{-Média-Arcos}_L)$ ou se $v(WD\text{-Média-Arcos}_L) \leq v(D\text{-Média-Arcos}_L)$ e existem instâncias para as quais esta desigualdade é estrita.*

Com o intuito de comparar as relaxações lineares dos modelos D -Média-Arcos

e D -Média-Arcos, comecemos por observar que a função objectivo (59) dos modelos desagregados discutidos em cima é precisamente a função objectivo do modelo agregado. Usando agora o mesmo argumento utilizado na demonstração do resultado I.8 (ver capítulo I), repare-se então que é possível transformar qualquer solução admissível de WD -Média-Arcos $_L$ numa solução admissível de A -Média-Arcos $_L$ através de $w_{ij}^{hpq} = \sum_{d=1, \dots, D} z_{ij}^{hdpq}$ ($h = 1, \dots, H; p, q \in S$) e ambas as soluções têm o mesmo custo. Inversamente, considere-se uma solução admissível de A -Média-Arcos $_L$ e seja w_{ij}^{hpq*} o valor das variáveis w_{ij}^{hpq} nessa solução. Considere-se agora o valor para as variáveis z_{ij}^{hdpq} dado por $z_{ij}^{h1pq} = \dots = z_{ij}^{hDpq} = \frac{w_{ij}^{hpq*}}{D}$. Verifica-se facilmente que a solução assim obtida é admissível para WD -Média-Arcos $_L$ e que tem o mesmo custo que a solução do modelo agregado. Assim, podemos estabelecer que:

Resultado III.2

$$v(WD\text{-Média-Arcos}_L) = v(A\text{-Média-Arcos}_L).$$

Através dos resultados III.1 e III.2, podemos agora escrever o seguinte resultado:

Resultado III.3

- i) Se $D = \Delta$, $v(D\text{-Média-Arcos}_L) = v(A\text{-Média-Arcos}_L)$.
- ii) Se $D > \Delta$, permanece em aberto se $v(D\text{-Média-Arcos}_L) = v(A\text{-Média-Arcos}_L)$ ou se $v(D\text{-Média-Arcos}_L) \geq v(A\text{-Média-Arcos}_L)$ e existem instâncias para as quais esta desigualdade é estrita.

Este resultado é interessante na medida em que para $D = \Delta$ (ou seja, no contexto dos problemas de *traffic engineering*, sob o mecanismo de sobrevivência *Path Diversity*) não temos nenhuma vantagem em utilizar o modelo desagregado. Já para o caso $D > \Delta$ (i.e., sob *Path Protection*, se $\Delta = D - 1$, ou outros mecanismos de sobrevivência não testados neste trabalho) a equivalência permanece em aberto. Como veremos na secção III.5, na nossa experiência computacional os modelos desagregado e agregado forneceram o mesmo limite inferior para o problema em estudo em todas as instâncias testadas. Mais ainda, apesar de apresentarmos os resultados obtidos

considerando apenas $H = 4$, tanto para $H = 3$ como $H = 5$ os resultados são semelhantes.

III.4.2 Minimização do Número Máximo de Saltos

Nesta subsecção pretendemos estabelecer a relação entre as relaxações lineares dos modelos *D-Máx-Arcos* e *A-Máx-Arcos*. No entanto, como veremos através da nossa experiência computacional na próxima secção, verifica-se que *D-Máx-Arcos_L* não domina *A-Máx-Arcos_L*. Estes resultados motivaram-nos a construir um novo modelo desagregado mais forte, como se descreve de seguida.

À semelhança da exposição feita na subsecção anterior, começamos por recordar o modelo desagregado:

Formulação D-Máximo-Arcos

Min V

sujeito a:

$$(33a)-(33c), (34), (35), (36a), (36b)$$

$$\sum_{(i,j) \in A} \sum_{h=1, \dots, H} z_{ij}^{hdpq} \leq V, \quad d = 1, \dots, \Delta; p, q \in S \quad (38)$$

$$V \geq 0 \text{ e inteira.} \quad (39)$$

Como se mostra na secção dos resultados computacionais, a relaxação linear deste modelo não domina a relaxação linear do modelo agregado, pelo que apresentamos de seguida um fortalecimento desse modelo. Note-se então que introduzindo as variáveis binárias V^h (definidas na apresentação do modelo *A-Máx-Arcos*) no modelo desagregado, podemos escrever as seguintes desigualdades:

$$\sum_{(i,j) \in A} z_{ij}^{hdpq} \leq V^h, \quad h = 1, \dots, H; d = 1, \dots, \Delta; p, q \in S. \quad (61)$$

De facto, note-se que as variáveis V^h podem ser interpretadas como indicando se para todos os serviços $p, q \in S$ existe um Δ -ésimo caminho com um arco na posição

h . Note-se ainda que, sob uma outra perspectiva, é possível obter estas restrições fazendo $V = \sum_{h=1, \dots, H} V^h$ e desagregando as restrições (38) em h .

Assim, substituímos as restrições (38) pelas (61) no modelo D - $Máx$ - $Arcos$ e recriamos a sua função objectivo da seguinte forma:

$$\text{Min} \quad \sum_{h=1, \dots, H} V^h. \quad (62)$$

Sendo SD - $Máx$ - $Arcos$ o modelo resultante, podemos então afirmar que:

Resultado III.4

$v(SD\text{-}Máx\text{-}Arcos_L) \geq v(D\text{-}Máx\text{-}Arcos_L)$ e para algumas instâncias a desigualdade é estrita.

Os resultados computacionais obtidos (ver secção III.5) provam a segunda parte deste resultado. De seguida, provamos que a relaxação linear do modelo desagregado fortalecido é pelo menos tão boa quanto a relaxação linear do modelo agregado. Para tal, observe-se que ao somarmos as restrições (61) para todo o $d = 1, \dots, \Delta$ e, por sua vez, somarmos o resultado à soma das restrições

$$\sum_{(i,j) \in A} z_{ij}^{hdpq} \leq 1, \quad d = \Delta + 1, \dots, D; p, q \in S; h = 1, \dots, H \quad (63)$$

para todo o $d = \Delta + 1, \dots, D$, obtemos:

$$\sum_{d=1, \dots, D} \sum_{(i,j) \in A} z_{ij}^{hdpq} \leq \Delta \times V^h + (D - \Delta), \quad p, q \in S; h = 1, \dots, H. \quad (64)$$

Seja WSD - $Máx$ - $Arcos$ o modelo desagregado resultante de termos enfraquecido SD - $Máx$ - $Arcos$ como descrito. Note-se agora que para o caso em que $\Delta = D$, se uma solução óptima da relaxação linear do modelo WSD - $Máx$ - $Arcos$ não satisfizer pelo menos uma restrição de SD - $Máx$ - $Arcos$, conseguimos modificar essa solução de modo que seja admissível para o modelo mais forte e cujo valor da função objectivo seja igual ao da solução original fazendo $z_{ij}^{h1pq} = \dots = z_{ij}^{hDpq} = \frac{w_{ij}^{hpq*}}{D}$ (em que w_{ij}^{hpq*} é o

valor das variáveis $w_{ij}^{hpq} = \sum_{d=1, \dots, D} z_{ij}^{hdpq}$). Desta forma, podemos escrever o seguinte resultado:

Resultado III.5

- i) Se $D = \Delta$, $v(SD\text{-Máx-Arcos}_L) = v(WSD\text{-Máx-Arcos}_L)$.*
- ii) Se $D > \Delta$, permanece em aberto se $v(SD\text{-Máx-Arcos}_L) = v(WSD\text{-Máx-Arcos}_L)$ ou se $v(SD\text{-Máx-Arcos}_L) \geq v(WSD\text{-Máx-Arcos}_L)$ e existem instâncias para as quais esta desigualdade é estrita.*

Por outro lado, note-se que, através das igualdades $w_{ij}^{hpq} = \sum_{d=1, \dots, D} z_{ij}^{hdpq}$, as restrições (64) são exactamente as mesmas que as restrições (51) do modelo agregado. Através dos mesmos argumentos usados na demonstração do resultado III.2, temos agora o seguinte resultado:

Resultado III.6

$$v(WSD\text{-Máx-Arcos}_L) = v(A\text{-Máx-Arcos}_L).$$

Através dos resultados III.5 e III.6, podemos agora estabelecer que:

Resultado III.7

- i) Se $D = \Delta$, $v(SD\text{-Máx-Arcos}_L) = v(A\text{-Máx-Arcos}_L)$.*
- ii) Se $D > \Delta$, permanece em aberto se $v(SD\text{-Máx-Arcos}_L) = v(A\text{-Máx-Arcos}_L)$ ou se $v(SD\text{-Máx-Arcos}_L) \geq v(A\text{-Máx-Arcos}_L)$ e existem instâncias para as quais esta desigualdade é estrita.*

Finalizamos esta secção realçando mais uma vez que, apesar de na próxima secção apresentarmos os resultados obtidos considerando apenas $H = 4$, tanto para $H = 3$ como $H = 5$ os resultados são semelhantes: as relaxações lineares dos modelos $SD\text{-Máx-Arcos}$ e $A\text{-Máx-Arcos}$ forneceram o mesmo limite inferior em todas as instâncias testadas.

III.5 Resultados Computacionais

Como referimos no início do presente capítulo, as redes sobre as quais os problemas em estudo se encontram definidos consistem nas soluções obtidas na secção II.2 para o problema de dimensionamento de redes MPLS sobre redes WDM. Nesta secção consideraremos apenas as soluções obtidas com base na rede NSFNET. Como vimos, essas soluções resultam da configuração de custo mínimo tendo em conta diversas condições, entre as quais se encontra a necessidade de dimensionar a capacidade de cada *lightpath* ao mesmo tempo que se encaminhava uma matriz de tráfego T entre qualquer par de nodos $p, q \in S$ sob um dos mecanismos de protecção *Path Diversity* ou *Path Protection*. Relativamente às restantes condições, importa referir que podemos naturalmente assumir que a rede dimensionada $N = (X, U)$ contém pelo menos D caminhos, com $D = 2, 3$ (nesta secção não se consideram as soluções obtidas com $D = 4$), disjuntos nos nodos entre qualquer par de nodos $p, q \in S$, cada um dos quais com não mais que $H = 4$ saltos.

Recorde-se agora que nas experiências computacionais levadas a cabo para o problema de dimensionamento considerámos dois cenários principais para a rede NSFNET – $|S| = 6$ e $|S| = 8$ – a que correspondem 15 e 28 serviços no total, respectivamente, bem como quatro sub-cenários relativamente às matrizes de tráfego consideradas para cada caso. Os valores t_{pq} destas matrizes, lembre-se, foram gerados aleatoriamente segundo distribuições uniformes nos intervalos $[0, 0.4]$, $[0, 0.6]$, $[0, 0.8]$ e $[0, 1.0]$.

A figura III.2 representa uma dessas soluções obtida com $|S| = 6$, $D = 3$ e tráfegos no intervalo $[0, 0.8]$, em que os nodos a preto representam os nodos $p, q \in S$, o nodo a cinzento é um nodo em $X \setminus S$ (ou seja, um nodo onde foi instalado um nodo de trânsito no problema de dimensionamento) e as arestas mais espessas representam uma maior quantidade de *lightpaths* instalada.

Relativamente às novas matrizes de tráfego R que pretendemos encaminhar sobre N , consideramos dois tipos de “perturbações” das matrizes de tráfego originais. Como referimos antes, uma vez que estas últimas são usualmente resultado de estimativas por excesso do tráfego que efectivamente se verifica em situações normais,

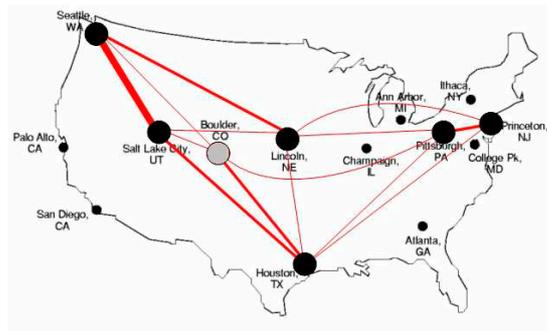


Figura III.2: Exemplo de uma rede $N = (X, U)$, solução do problema de dimensionamento obtida com $|S| = 6$, $D = 3$ e tráfegos em $[0, 0.8]$.

consideramos novos valores de tráfego r_{pq} entre 90 a 97,5% dos valores de tráfego originais, ou seja: $r_{pq} = (1 - \delta)t_{pq}$ ($p, q \in S$), com $\delta = 0.025, 0.05, 0.075$ e 0.1 . Por outro lado, como os valores t_{pq} poderão estar sujeitos a variações ao longo do tempo, consideramos também valores r_{pq} uniformemente distribuídos em intervalos da forma $[(1 - \epsilon)t_{pq}, (1 + \epsilon)t_{pq}]$, com $\epsilon = 0, 0.05, 0.1, 0.15$ e 0.2 . Resumidamente, com estes dois tipos de novo tráfego, pretendemos avaliar a eficiência dos modelos discutidos nas secções anteriores quando as redes N são sujeitas a diferentes “cargas” de tráfego.

Todos os resultados computacionais apresentados nesta subsecção foram obtidos através do CPLEX 10.2, num Intel Core 2 Duo a 2.0Ghz com 512Mb de RAM. Um tempo limite de dois dias para a determinação da solução óptima (inteira) foi imposto em todas as instâncias testadas.

Nas subsecções III.5.1 e III.5.2, apresentamos os resultados computacionais relativos aos diferentes modelos propostos para a minimização do número médio e máximo de saltos, respectivamente, comparando os *gaps* das respectivas relaxações lineares e os correspondentes tempos de CPU. Apresentamos ainda dados detalhados relativos à eficiência dos melhores modelos na determinação da solução óptima inteira de ambos os problemas em estudo, em diferentes cenários de tráfego e sob ambos os mecanismos de sobrevivência.

Na subsecção III.5.3, comparamos as soluções obtidas sob cada mecanismo de sobrevivência, avaliando o impacto do volume de tráfego total no “custo” (ou seja,

atraso médio ou máximo associado) das redes e, ainda, o impacto dos erros associados às estimativas de tráfego na admissibilidade das soluções previamente determinadas.

III.5.1 Minimização do Número Médio de Saltos

Analisando então os resultados obtidos com os modelos propostos para o problema da minimização do número médio de saltos, começamos por salientar que os valores óptimos das funções objectivo de cada um desses modelos não correspondem exactamente ao real número médio de saltos óptimo efectuado pelos Δ melhores caminhos de cada serviço. Para podermos comparar os diferentes modelos é necessário então que esses valores óptimos sejam convertidos nos reais valores óptimos, de acordo com a tabela III.1, onde, para simplificar, s denota o número total de serviços existentes ($s = |S| \times (|S| - 1)/2$) e v o valor óptimo de cada modelo.

Tabela III.1: Conversão dos valores óptimos das funções objectivo dos modelos para a minimização do número médio de saltos nos reais valores óptimos.

Modelo	Função Objectivo	Valor Óptimo Real
D-Média-Arcos	$Min \sum_{d=1, \dots, \Delta} \sum_{p, q \in S} \sum_{(i, j) \in A} \sum_{h=1, \dots, H} z_{ij}^{hdpq}$	$\frac{v}{\Delta \times s}$
A-Média-Arcos	$Min \sum_{p, q \in S} \sum_{h=1, \dots, H} U^{hpq}$	
D-Média-Lacetes	$Max \sum_{d=1, \dots, \Delta} \sum_{p, q \in S} \sum_{h=1, \dots, H} z_{qq}^{hdpq}$	$\frac{\Delta \times s \times H - v}{\Delta \times s}$
A-Média-Lacetes	$Max \sum_{p, q \in S} \sum_{h=1, \dots, H} V^{hpq}$	

Como referimos anteriormente (ver subsecções III.2.1.1 e III.2.2.1), relativamente aos modelos *D-Média-Arcos* e *D-Média-Lacetes*, poderemos incluir ou não as restrições de *ranking* (37) e (41), respectivamente. Em ambos os casos, se o conjunto de soluções admissíveis for demasiadamente simétrico, a inclusão dessas restrições poderá tornar os modelos mais eficientes. No entanto, verificou-se que enquanto isso acontece para algumas instâncias, para outras acontece o contrário. Por esta razão, no que toca aos modelos referidos, os resultados computacionais que apresentamos de seguida dizem respeito às versões sem restrições de *ranking*.

Assim, as tabelas III.2 e III.3 contêm os resultados computacionais relativos às relaxações lineares dos modelos propostos para a minimização do número médio de saltos (tendo em conta os reais valores óptimos das funções objectivo) sob os mecanismos de sobrevivência *Path Diversity* e *Path Protection*, respectivamente. Para cada caso, apresentamos os valores médios dos *gaps* (em percentagem) e dos tempos de CPU (em segundos, entre parêntesis) das relaxações lineares de cada modelo para todas as instâncias testadas com $|S| = 6$, $D = 2, 3$, e todos os sub-cenários de tráfego. Separamos estes valores de acordo com a carga de tráfego que existe em cada instância: “*sub*”, “*igual*” e “*viz*” para os casos em que os novos tráfegos r_{pq} são inferiores ($\delta = 0.025, 0.05, 0.075$ e 0.1), iguais ($\epsilon = 0$) ou sensivelmente iguais ($\epsilon = 0.05, 0.1, 0.15$ e 0.2) aos tráfegos originais (t_{pq}), respectivamente. Note-se que neste último caso (“*viz*”) poderá haver instâncias sem soluções admissíveis uma vez que a carga de tráfego poderá ser superior à capacidade instalada, b_e , em pelo menos uma aresta $e \in U$, pelo que estes casos não são naturalmente contabilizados nos valores médios apresentados.

Tabela III.2: Resultados computacionais dos modelos *D-Média-Arcos_L*, *D-Média-Lacetes_L*, *A-Média-Arcos_L* e *A-Média-Lacetes_L* sob o mecanismo *Path Diversity* ($\Delta = D$).

	D = 2			
	<i>D-Média-Arcos_L</i>	<i>D-Média-Lacetes_L</i>	<i>A-Média-Arcos_L</i>	<i>A-Média-Lacetes_L</i>
<i>sub</i>	1,9% (0,02s)	1,9% (0,02s)	1,9% (0,01s)	1,9% (0,01s)
<i>igual</i>	2,9% (0,02s)	2,9% (0,02s)	2,9% (0,01s)	2,9% (0,01s)
<i>viz</i>	3,7% (0,01s)	3,7% (0,01s)	3,7% (0,01s)	3,7% (0s)
	D = 3			
	<i>D-Média-Arcos_L</i>	<i>D-Média-Lacetes_L</i>	<i>A-Média-Arcos_L</i>	<i>A-Média-Lacetes_L</i>
<i>sub</i>	2,1% (0,03s)	2,1% (0,02s)	2,1% (0,01s)	2,1% (0,01s)
<i>igual</i>	4,2% (0,03s)	4,2% (0,02s)	4,2% (0,01s)	4,2% (0,01s)
<i>viz</i>	1,7% (0,02s)	1,7% (0,02s)	1,7% (0,01s)	1,7% (0,04s)

Relativamente aos modelos *D-Média-Arcos* e *A-Média-Arcos*, salientamos o facto de, tanto sob *Path Diversity* como sob *Path Protection*, os *gaps* das suas relaxações lineares serem iguais, quer em termos médios como para cada instância. Se sob o primeiro mecanismo de sobrevivência ($\Delta = D$) este facto não pode ser surpreendente

Tabela III.3: Resultados computacionais dos modelos D -Média-Arcos $_L$, D -Média-Lacetes $_L$, A -Média-Arcos $_L$ e A -Média-Lacetes $_L$ sob o mecanismo *Path Protection* ($\Delta = D - 1$).

	D = 2			
	D -Média-Arcos $_L$	D -Média-Lacetes $_L$	A -Média-Arcos $_L$	A -Média-Lacetes $_L$
<i>sub</i>	0,2% (0,01s)	0,2% (0,01s)	0,2% (0,01s)	0,2% (0,02s)
<i>igual</i>	1,2% (0,02s)	1,2% (0,02s)	1,2% (0,01s)	1,2% (0,01s)
<i>viz</i>	0% (0,01s)	0% (0,01s)	0% (0,01s)	0% (0,01s)
	D = 3			
	D -Média-Arcos $_L$	D -Média-Lacetes $_L$	A -Média-Arcos $_L$	A -Média-Lacetes $_L$
<i>sub</i>	1,4% (0,03s)	1,4% (0,03s)	1,4% (0,01s)	1,4% (0,02s)
<i>igual</i>	3,5% (0,04s)	3,5% (0,03s)	3,5% (0,51s)	3,5% (0,02s)
<i>viz</i>	3% (0,02s)	3% (0,03s)	3% (0,02s)	3% (0,02s)

(ver alínea *i*) do resultado III.3), já sob o segundo mecanismo ($\Delta = D - 1$) esse facto apenas nos permite concluir a alínea *ii*) do resultado III.3. Note-se que, apesar de não apresentarmos o mesmo tipo de resultados computacionais para $|S| = 8$, as observações anteriores mantêm-se. Em relação aos tempos médios de CPU necessários à determinação da solução óptima da relaxação linear dos referidos modelos, verifica-se, geralmente, uma ligeira vantagem do modelo agregado, especialmente sob *Path Diversity*. Assim, se sob *Path Diversity* é evidente que não ganhamos nada em considerar o modelo desagregado, sob *Path Protection* a possível vantagem desse modelo é anulada devido o seu maior número de variáveis.

Verifica-se ainda que, como esperado, os modelos baseados em lacetes são, ainda que ligeiramente, mais eficientes em termos de tempo de CPU exigido do que os modelos baseados em arcos, o que advém do facto das suas funções objectivo e restrições serem escritas à custa de menos variáveis.

Por último, observe-se que as instâncias com $D = 3$ são geralmente de mais difícil resolução do que as instâncias com $D = 2$.

Com base nestes resultados, acreditamos que o modelo A -Média-Lacetes reúne as melhores condições para apresentarmos os dados relativos à determinação da solução óptima inteira das instâncias consideradas. Assim, considerando esse modelo, apresentamos de seguida os resultados computacionais detalhados para cada instância,

incluindo o cenário $|S| = 8$. Note-se que, apesar de não apresentarmos os resultados obtidos com os outros modelos, os modelos agregados são geralmente mais eficientes que os desagregados. Por exemplo, para uma instância com $D = 3$ (sob *Path Protection*) o modelo *D-Média-Lacetes* levou 11508 segundos a determinar a solução óptima inteira enquanto que o modelo *A-Média-Lacetes* demorou apenas 766 segundos.

Separámos os resultados computacionais obtidos consoante o mecanismo de sobrevivência adoptado. Assim, as tabelas III.4 e III.5 contêm os dados relativos a *Path Diversity* e *Path Protection*, respectivamente. Em ambos os casos, apresentamos para cada instância testada (*i.e.*, considerando $|S| = 6, 8$, $D = 2, 3$ e todos os sub-cenários de tráfego e correspondentes variações expressas pelos valores δ e ϵ mencionados antes): o *gap* (em percentagem) da relaxação linear do modelo *A-Média-Lacetes* e o tempo de CPU (em segundos, entre parêntesis) exigido pelo CPLEX na determinação da solução óptima inteira.

No que toca à eficiência do modelo *A-Média-Lacetes*, observe-se que os tempos de CPU requeridos são na maior parte das instâncias bastante reduzidos, especialmente para aquelas sob o mecanismo *Path Diversity*. Sob *Path Protection*, apenas para as instâncias maiores, com $|S| = 8$, e para cargas de tráfego mais elevadas é que se obtiveram tempos de CPU mais elevados, havendo inclusivamente uma instância cuja solução óptima inteira o CPLEX não conseguiu determinar ao fim de dois dias. Uma possível explicação para este facto é que sob *Path Diversity* todos os D caminhos são relevantes ($\Delta = D$) com respeito relativamente à minimização do número médio de saltos, ao passo que sob *Path Protection* apenas os $\Delta = D - 1$ melhores caminhos são relevantes e, recorde-se, as variáveis w_{ij}^{hpq} do modelo *A-Média-Lacetes* não fazem distinção entre caminhos relevantes ou não. Note-se ainda que para os modelos desagregados as instâncias sob *Path Protection* também são mais difíceis de resolver que as instâncias sob *Path Diversity*.

Note-se que os tráfegos r_{pq} gerados segundo distribuições uniformes nos intervalos $[(1 - \epsilon)t_{pq}, (1 + \epsilon)t_{pq}]$ poderão dar origem a instâncias admissíveis para um ϵ elevado e não admissíveis para um ϵ pequeno. Não indicamos o *gap* relativamente à instância assinalada com * pois o CPLEX foi incapaz de fornecer um limite superior para o

Tabela III.4: Resultados computacionais do modelo *A-Média-Lacetes*, para $|S| = 6$ e $|S| = 8$, sob *Path Diversity* ($\Delta = D$),

<i>tráfego</i>	<i>variação</i>	$ S = 6$		$ S = 8$	
		$D = 2$	$D = 3$	$D = 2$	$D = 3$
[0, 0.4]	$\delta = 0, 1$	0% (< 1s)	0% (< 1s)	0% (< 1s)	0,5% (< 1s)
	$\delta = 0, 075$	0% (< 1s)	0% (< 1s)	0% (< 1s)	0,3% (< 1s)
	$\delta = 0, 05$	0% (< 1s)	0% (< 1s)	0% (< 1s)	0,2% (< 1s)
	$\delta = 0, 025$	0% (< 1s)	0% (< 1s)	0% (< 1s)	0% (< 1s)
	$\epsilon = 0, 00$	0% (< 1s)	0% (< 1s)	0% (< 1s)	2,7% (< 1s)
	$\epsilon = 0, 05$	0% (< 1s)	0% (< 1s)	não admissível	2,7% (< 1s)
	$\epsilon = 0, 10$	0% (< 1s)	0% (< 1s)	não admissível	não admissível
	$\epsilon = 0, 15$	não admissível	0% (< 1s)	não admissível	3,3% (< 1s)
	$\epsilon = 0, 20$	não admissível	0% (< 1s)	não admissível	não admissível
[0, 0.6]	$\delta = 0, 1$	0% (< 1s)	0% (< 1s)	0,4% (< 1s)	0,1% (< 1s)
	$\delta = 0, 075$	0% (< 1s)	1,2% (< 1s)	0,3% (< 1s)	0,6% (< 1s)
	$\delta = 0, 05$	6,1% (< 1s)	1,0% (< 1s)	1,2% (< 1s)	1,1% (< 1s)
	$\delta = 0, 025$	7,0% (< 1s)	0,8% (< 1s)	2,0% (< 1s)	2,7% (< 1s)
	$\epsilon = 0, 00$	7,6% (< 1s)	0,6% (< 1s)	5,6% (< 1s)	4,4% (< 1s)
	$\epsilon = 0, 05$	não admissível	0,7% (< 1s)	não admissível	não admissível
	$\epsilon = 0, 10$	11,3% (< 1s)	0,2% (< 1s)	não admissível	não admissível
	$\epsilon = 0, 15$	6,4% (< 1s)	0,8% (< 1s)	não admissível	não admissível
	$\epsilon = 0, 20$	não admissível	0,5% (< 1s)	não admissível	não admissível
[0, 0.8]	$\delta = 0, 1$	4,8% (< 1s)	0,3% (< 1s)	1,6% (< 1s)	3,9% (< 1s)
	$\delta = 0, 075$	4,1% (< 1s)	2,7% (< 1s)	1,3% (< 1s)	3,7% (< 1s)
	$\delta = 0, 05$	3,3% (< 1s)	3,3% (< 1s)	2,0% (< 1s)	2,8% (< 1s)
	$\delta = 0, 025$	4,5% (< 1s)	6,4% (< 1s)	3,4% (2s)	1,9% (< 1s)
	$\epsilon = 0, 00$	3,8% (< 1s)	12,4% (< 1s)	5,5% (4s)	2,9% (2s)
	$\epsilon = 0, 05$	não admissível	11,8% (< 1s)	7,1% (5s)	não admissível
	$\epsilon = 0, 10$	não admissível	não admissível	não admissível	não admissível
	$\epsilon = 0, 15$	não admissível	não admissível	não admissível	não admissível
	$\epsilon = 0, 20$	não admissível	não admissível	não admissível	não admissível
[0, 1.0]	$\delta = 0, 1$	0% (< 1s)	4,8% (< 1s)	0,7% (< 1s)	1,6% (< 1s)
	$\delta = 0, 075$	0% (< 1s)	5,2% (< 1s)	1,6% (< 1s)	1,9% (< 1s)
	$\delta = 0, 05$	0% (< 1s)	4,5% (< 1s)	1,3% (< 1s)	2,1% (< 1s)
	$\delta = 0, 025$	0% (< 1s)	3,7% (< 1s)	1,0% (< 1s)	1,7% (< 1s)
	$\epsilon = 0, 00$	0% (< 1s)	3,9% (< 1s)	4,1% (< 1s)	3,8% (2s)
	$\epsilon = 0, 05$	0% (< 1s)	3,3% (< 1s)	não admissível	4,8% (2s)
	$\epsilon = 0, 10$	0% (< 1s)	não admissível	não admissível	não admissível
	$\epsilon = 0, 15$	1,7% (< 1s)	não admissível	não admissível	não admissível
	$\epsilon = 0, 20$	9,7% (< 1s)	não admissível	não admissível	não admissível

Tabela III.5: Resultados computacionais do modelo *A-Média-Lacetes*, para $|S| = 6$ e $|S| = 8$, sob *Path Protection* ($\Delta = D - 1$),

<i>tráfego</i>	<i>variação</i>	$ S = 6$		$ S = 8$	
		$D = 2$	$D = 3$	$D = 2$	$D = 3$
[0, 0.4]	$\delta = 0, 1$	0% (< 1s)	0% (< 1s)	0% (< 1s)	0% (< 1s)
	$\delta = 0, 075$	0% (< 1s)	3,6% (< 1s)	0% (< 1s)	0% (< 1s)
	$\delta = 0, 05$	0% (< 1s)	2,9% (< 1s)	0% (< 1s)	0% (3s)
	$\delta = 0, 025$	0% (< 1s)	4,2% (< 1s)	0% (14s)	0% (< 1s)
	$\epsilon = 0, 00$	0% (< 1s)	3,6% (< 1s)	4,4% (1258s)	0% (5s)
	$\epsilon = 0, 05$	não admissível	3,7% (< 1s)	não admissível	não admissível
	$\epsilon = 0, 10$	não admissível	3,6% (< 1s)	não admissível	não admissível
	$\epsilon = 0, 15$	não admissível	não admissível	não admissível	não admissível
	$\epsilon = 0, 20$	não admissível	não admissível	não admissível	não admissível
[0, 0.6]	$\delta = 0, 1$	0% (< 1s)	0% (< 1s)	0% (< 1s)	0,7% (< 1s)
	$\delta = 0, 075$	0% (< 1s)	0% (< 1s)	0% (< 1s)	0,6% (< 1s)
	$\delta = 0, 05$	0% (< 1s)	0% (< 1s)	0% (< 1s)	0,5% (< 1s)
	$\delta = 0, 025$	0% (< 1s)	2,4% (< 1s)	0% (28s)	0,4% (1s)
	$\epsilon = 0, 00$	0% (< 1s)	2,4% (< 1s)	2,4% (6017s)	0,3% (28s)
	$\epsilon = 0, 05$	0% (< 1s)	2,4% (< 1s)	não admissível	1,3% (2714s)
	$\epsilon = 0, 10$	0% (< 1s)	não admissível	não admissível	não admissível
	$\epsilon = 0, 15$	0% (< 1s)	2,4% (< 1s)	não admissível	não admissível
	$\epsilon = 0, 20$	não admissível	não admissível	não admissível	não admissível
[0, 0.8]	$\delta = 0, 1$	0% (< 1s)	0% (< 1s)	0% (< 1s)	0% (< 1s)
	$\delta = 0, 075$	0% (< 1s)	0% (< 1s)	0% (< 1s)	0% (68s)
	$\delta = 0, 05$	0% (< 1s)	0% (< 1s)	0% (91s)	0% (3s)
	$\delta = 0, 025$	4,8% (< 1s)	0% (< 1s)	0% (3s)	0% (8s)
	$\epsilon = 0, 00$	4,8% (< 1s)	0% (< 1s)	2,3% (77s)	0% (15583s)
	$\epsilon = 0, 05$	não admissível	não admissível	não admissível	0% (1587s)
	$\epsilon = 0, 10$	não admissível	não admissível	não admissível	0% (1373s)
	$\epsilon = 0, 15$	não admissível	não admissível	não admissível	não admissível
	$\epsilon = 0, 20$	não admissível	não admissível	não admissível	não admissível
[0, 1.0]	$\delta = 0, 1$	0% (< 1s)	2,0% (< 1s)	1,6% (< 1s)	0% (< 1s)
	$\delta = 0, 075$	0% (< 1s)	1,8% (< 1s)	1,5% (2s)	0% (< 1s)
	$\delta = 0, 05$	0% (< 1s)	1,7% (< 1s)	1,4% (< 1s)	0% (2s)
	$\delta = 0, 025$	0% (< 1s)	3,8% (< 1s)	1,3% (38s)	0% (39s)
	$\epsilon = 0, 00$	0% (< 1s)	8,0% (< 1s)	3,5% (30s)	1,1% (766s)
	$\epsilon = 0, 05$	não admissível	não admissível	não admissível	* (2 dias)
	$\epsilon = 0, 10$	não admissível	não admissível	não admissível	não admissível
	$\epsilon = 0, 15$	não admissível	não admissível	não admissível	não admissível
	$\epsilon = 0, 20$	não admissível	não admissível	não admissível	não admissível

valor óptimo da mesma ao fim de dois dias de CPU.

Para concluir esta subsecção, realçamos a eficiência do modelo proposto, em particular na resolução das instâncias com tráfego abaixo (ver dados para $\delta = 0.025, 0.05, 0.075$ e 0.1) da capacidade para a qual cada uma dessas instâncias foi dimensionada. Em contrapartida, quando a carga de tráfego se situa perto (ver dados para $\epsilon = 0, 0.05, 0.1, 0.15$ e 0.2) dessa capacidade, as instâncias são progressivamente mais difíceis.

III.5.2 Minimização do Número Máximo de Saltos

Relativamente à minimização do número máximo de saltos, considerando os modelos discutidos nas secções anteriores, note-se que os valores óptimos das funções objectivo dos modelos *D-Máx-Arcos*, *SD-Máx-Arcos* e *A-Máx-Arcos* correspondem (directamente) ao real número máximo de saltos efectuado pelos Δ melhores caminhos de cada serviço. Já o mesmo não se passa relativamente aos modelos *D-Máx-Lacetes* e *A-Máx-Lacetes*: para ambos os casos, representando o valor óptimo das suas funções objectivo por v , então o real valor óptimo é dado por $H - v$.

À semelhança dos resultados apresentados na subsecção anterior, as tabelas III.6 e III.7 contêm os resultados computacionais relativos às relaxações lineares dos modelos propostos para o problema em questão (considerando os reais valores óptimos) sob os mecanismos de sobrevivência *Path Diversity* e *Path Diversity*, respectivamente. Como antes, apresentamos para cada caso os valores médios dos *gaps* (em percentagem) e dos tempos de CPU (em segundos, entre parêntesis) das relaxações lineares de cada modelo para todas as instâncias testadas (com $|S| = 6$, $D = 2, 3$, e todos os sub-cenários de tráfego mencionados antes).

Começamos por observar que, contrariamente ao que verificámos nos modelos propostos para a minimização do número médio de saltos, aparentemente, os modelos baseados em lacetes não são mais eficientes em termos de esforço computacional do que os modelos baseados em arcos. Este aspecto confirma-se na determinação da

Tabela III.6: Resultados computacionais dos modelos D -Máx-Arcos $_L$, D -Máx-Lacetes $_L$, SD -Máx-Arcos $_L$, A -Máx-Arcos $_L$ e A -Máx-Lacetes $_L$ sob o mecanismo *Path Diversity* ($\Delta = D$).

		D = 2				
		D -Máx-Arcos $_L$	D -Máx-Lacetes $_L$	SD -Máx-Arcos $_L$	A -Máx-Arcos $_L$	A -Máx-Lacetes $_L$
<i>sub</i>		16,7% (0,02s)	16,7% (0,02s)	16,7% (0,01s)	16,7% (0,01s)	16,7% (0,02s)
<i>igual</i>		16,7% (0,02s)	16,7% (0,02s)	16,7% (0,02s)	16,7% (0,01s)	16,7% (0,02s)
<i>viz</i>		10,4% (0,02s)	10,4% (0,02s)	10,4% (0,01s)	10,4% (0,01s)	10,4% (0,02s)
		D = 3				
		D -Máx-Arcos $_L$	D -Máx-Lacetes $_L$	SD -Máx-Arcos $_L$	A -Máx-Arcos $_L$	A -Máx-Lacetes $_L$
<i>sub</i>		27,7% (0,03s)	27,7% (0,03s)	21,5% (0,03s)	21,5% (0,01s)	21,5% (0,02s)
<i>igual</i>		27,7% (0,03s)	27,7% (0,03s)	21,5% (0,03s)	21,5% (0,01s)	21,5% (0,02s)
<i>viz</i>		31,1% (0,02s)	31,1% (0,03s)	23,6% (0,03s)	23,6% (0,01s)	23,6% (0,02s)

Tabela III.7: Resultados computacionais dos modelos D -Máx-Arcos $_L$, D -Máx-Lacetes $_L$, SD -Máx-Arcos $_L$, A -Máx-Arcos $_L$ e A -Máx-Lacetes $_L$ sob o mecanismo *Path Protection* ($\Delta = D - 1$).

		D = 2				
		D -Máx-Arcos $_L$	D -Máx-Lacetes $_L$	SD -Máx-Arcos $_L$	A -Máx-Arcos $_L$	A -Máx-Lacetes $_L$
<i>sub</i>		0,0% (0,02s)	0,0% (0,02s)	0,0% (0,02s)	0,0% (0,01s)	0,0% (0,02s)
<i>igual</i>		0,0% (0,02s)	0,0% (0,02s)	0,0% (0,02s)	0,0% (0,01s)	0,0% (0,03s)
<i>viz</i>		0,0% (0,02s)	0,0% (0,02s)	0,0% (0,02s)	0,0% (0,01s)	0,0% (0,02s)
		D = 3				
		D -Máx-Arcos $_L$	D -Máx-Lacetes $_L$	SD -Máx-Arcos $_L$	A -Máx-Arcos $_L$	A -Máx-Lacetes $_L$
<i>sub</i>		16,7% (0,03s)	16,7% (0,03s)	16,7% (0,04s)	16,7% (0,01s)	16,7% (0,03s)
<i>igual</i>		16,7% (0,03s)	16,7% (0,04s)	16,7% (0,05s)	16,7% (0,01s)	16,7% (0,03s)
<i>viz</i>		16,7% (0,03s)	16,7% (0,03s)	16,7% (0,03s)	16,7% (0,01s)	16,7% (0,03s)

solução óptima inteira, o que vai contra a nossa expectativa por esses modelos serem escritos à custa de menos variáveis.

Salientamos agora a confirmação empírica do resultado III.4 (ver subsecção III.4.2): sob *Path Diversity*, com $D = 3$, a relaxação linear do modelo SD -Máx-Arcos domina de facto (para várias instâncias individualmente) a relaxação linear do modelo D -Máx-Arcos. Contudo, para todas as instâncias dos restantes casos, a desigualdade do mesmo resultado verifica-se na igualdade.

Note-se que relativamente aos valores óptimos das relaxações lineares dos modelos SD -Máx-Arcos e A -Máx-Arcos, a desigualdade do resultado III.7 não se verifica estritamente para nenhuma instância testada. Note-se ainda que o maior número de variáveis do modelo desagregado prejudica-o em termos de esforço computacional, tanto sob *Path Diversity* como sob *Path Protection*.

À semelhança dos testes computacionais apresentados na subsecção anterior, geralmente as instâncias com $D = 3$ são mais difíceis de resolver que as instâncias com $D = 2$. Mais uma vez, esta observação bem como as anteriores mantêm-se para $|S| = 8$ apesar de não apresentarmos resultados para esse cenário.

Com base nestes resultados, pensamos que o modelo *A-Máx-Arcos* reúne as melhores condições para apresentarmos os dados relativos à determinação da solução ótima inteira das instâncias consideradas. Deste modo, apresentamos de seguida os resultados computacionais individualmente para cada instância considerando esse modelo. Apesar de não apresentarmos detalhadamente os resultados obtidos com os outros modelos, note-se que, devido ao maior número de variáveis dos modelos desagregados, para uma instância com $D = 3$ (sob *Path Protection*) o modelo *SD-Máx-Arcos* levou mais do que um dia a determinar a solução ótima inteira enquanto que o modelo *A-Máx-Arcos* demorou apenas 1487 segundos.

Mais uma vez de modo similar ao adoptado na apresentação de resultados na subsecção anterior, separando os resultados computacionais obtidos consoante o mecanismo de sobrevivência considerado, as tabelas III.8 e III.9 contêm os dados relativos a *Path Diversity* e *Path Protection*, respectivamente. Em ambos os casos, apresentamos o *gap* (em percentagem) da relaxação linear do modelo *A-Máx-Arcos* e o tempo de CPU (em segundos, entre parêntesis) exigido pelo CPLEX na determinação da solução ótima inteira de cada instância testada.

De uma forma geral, note-se que o modelo *A-Máx-Arcos* é bastante eficiente, de um modo particular na resolução das instâncias sob *Path Diversity*. Similarmente ao sucedido no problema da minimização do número médio de saltos, verifica-se que apenas para as instâncias maiores com $|S| = 8$, sob *Path Protection* e maiores cargas de tráfego é que se exigiu um maior esforço computacional. Mais uma vez recorde-se que no modelo agregado considerado as variáveis w_{ij}^{hpq} não fazem distinção entre caminhos relevantes ou não e sob *Path Protection* é necessário identificar o pior caminho dos $D - 1$ melhores caminhos de cada serviço. Realce-se a eficiência do modelo na resolução das instâncias com cargas de tráfego abaixo da capacidade que foram dimensionadas (ver dados para $\delta = 0.025, 0.05, 0.075$ e 0.1). De novo, assinalamos

Tabela III.8: Resultados computacionais do modelo A -Máx-Arcos, para $|S| = 6$ e $|S| = 8$, sob $Path Diversity$ ($\Delta = D$),

tráfego	variação	$ S = 6$		$ S = 8$	
		$D = 2$	$D = 3$	$D = 2$	$D = 3$
[0, 0.4]	$\delta = 0, 1$	25% (< 1s)	33,3% (< 1s)	0% (< 1s)	16,7% (< 1s)
	$\delta = 0, 075$	25% (< 1s)	33,3% (< 1s)	0% (< 1s)	16,7% (< 1s)
	$\delta = 0, 05$	25% (< 1s)	33,3% (< 1s)	0% (< 1s)	16,7% (< 1s)
	$\delta = 0, 025$	25% (< 1s)	33,3% (< 1s)	0% (< 1s)	16,7% (< 1s)
	$\epsilon = 0, 00$	25% (< 1s)	33,3% (< 1s)	0% (< 1s)	16,7% (< 1s)
	$\epsilon = 0, 05$	25% (< 1s)	33,3% (< 1s)	não admissível	16,7% (< 1s)
	$\epsilon = 0, 10$	25% (< 1s)	33,3% (< 1s)	não admissível	não admissível
	$\epsilon = 0, 15$	não admissível	33,3% (< 1s)	não admissível	16,7% (< 1s)
	$\epsilon = 0, 20$	não admissível	33,3% (< 1s)	não admissível	não admissível
[0, 0.6]	$\delta = 0, 1$	16,7% (< 1s)	33,3% (< 1s)	0% (< 1s)	25% (< 1s)
	$\delta = 0, 075$	16,7% (< 1s)	33,3% (< 1s)	25% (< 1s)	25% (< 1s)
	$\delta = 0, 05$	16,7% (< 1s)	33,3% (< 1s)	25% (< 1s)	25% (< 1s)
	$\delta = 0, 025$	16,7% (< 1s)	33,3% (< 1s)	25% (< 1s)	25% (< 1s)
	$\epsilon = 0, 00$	16,7% (< 1s)	33,3% (< 1s)	25% (< 1s)	25% (< 1s)
	$\epsilon = 0, 05$	não admissível	33,3% (< 1s)	não admissível	não admissível
	$\epsilon = 0, 10$	16,7% (< 1s)	33,3% (< 1s)	não admissível	não admissível
	$\epsilon = 0, 15$	16,7% (< 1s)	33,3% (< 1s)	não admissível	não admissível
	$\epsilon = 0, 20$	não admissível	33,3% (< 1s)	não admissível	não admissível
[0, 0.8]	$\delta = 0, 1$	25% (< 1s)	33,3% (< 1s)	0% (< 1s)	16,7% (< 1s)
	$\delta = 0, 075$	25% (< 1s)	33,3% (< 1s)	0% (< 1s)	16,7% (< 1s)
	$\delta = 0, 05$	25% (< 1s)	33,3% (< 1s)	25% (21s)	16,7% (< 1s)
	$\delta = 0, 025$	25% (< 1s)	33,3% (< 1s)	25% (3s)	16,7% (< 1s)
	$\epsilon = 0, 00$	25% (< 1s)	33,3% (< 1s)	25% (2s)	16,7% (3s)
	$\epsilon = 0, 05$	não admissível	33,3% (< 1s)	25% (3s)	não admissível
	$\epsilon = 0, 10$	não admissível	não admissível	não admissível	não admissível
	$\epsilon = 0, 15$	não admissível	não admissível	não admissível	não admissível
	$\epsilon = 0, 20$	não admissível	não admissível	não admissível	não admissível
[0, 1.0]	$\delta = 0, 1$	0% (< 1s)	11% (< 1s)	25% (< 1s)	25% (< 1s)
	$\delta = 0, 075$	0% (< 1s)	11% (< 1s)	25% (< 1s)	25% (< 1s)
	$\delta = 0, 05$	0% (< 1s)	11% (< 1s)	25% (< 1s)	25% (< 1s)
	$\delta = 0, 025$	0% (< 1s)	11% (< 1s)	25% (2s)	25% (< 1s)
	$\epsilon = 0, 00$	0% (< 1s)	11% (< 1s)	25% (2s)	25% (< 1s)
	$\epsilon = 0, 05$	0% (< 1s)	11% (< 1s)	não admissível	25% (< 1s)
	$\epsilon = 0, 10$	0% (< 1s)	não admissível	não admissível	não admissível
	$\epsilon = 0, 15$	0% (< 1s)	não admissível	não admissível	não admissível
	$\epsilon = 0, 20$	0% (< 1s)	não admissível	não admissível	não admissível

Tabela III.9: Resultados computacionais do modelo *A-Máx-Arcos*, para $|S| = 6$ e $|S| = 8$, sob *Path Protection* ($\Delta = D - 1$),

<i>tráfego</i>	<i>variação</i>	$ S = 6$		$ S = 8$	
		$D = 2$	$D = 3$	$D = 2$	$D = 3$
[0, 0.4]	$\delta = 0, 1$	0% (< 1s)	16,7% (< 1s)	0% (< 1s)	0% (< 1s)
	$\delta = 0, 075$	0% (< 1s)	16,7% (< 1s)	0% (156s)	0% (< 1s)
	$\delta = 0, 05$	0% (< 1s)	16,7% (< 1s)	0% (400s)	0% (< 1s)
	$\delta = 0, 025$	0% (< 1s)	16,7% (< 1s)	0% (9s)	0% (2s)
	$\epsilon = 0, 00$	0% (< 1s)	16,7% (< 1s)	0% (38s)	0% (3s)
	$\epsilon = 0, 05$	não admissível	16,7% (< 1s)	não admissível	não admissível
	$\epsilon = 0, 10$	não admissível	16,7% (< 1s)	não admissível	não admissível
	$\epsilon = 0, 15$	não admissível	não admissível	não admissível	não admissível
	$\epsilon = 0, 20$	não admissível	não admissível	não admissível	não admissível
[0, 0.6]	$\delta = 0, 1$	0% (< 1s)	16,7% (< 1s)	0% (< 1s)	0% (< 1s)
	$\delta = 0, 075$	0% (< 1s)	16,7% (< 1s)	0% (97s)	0% (< 1s)
	$\delta = 0, 05$	0% (< 1s)	16,7% (< 1s)	0% (66s)	0% (4s)
	$\delta = 0, 025$	0% (< 1s)	16,7% (< 1s)	0% (59s)	0% (175s)
	$\epsilon = 0, 00$	0% (< 1s)	16,7% (< 1s)	33,3% (64363s)	0% (50s)
	$\epsilon = 0, 05$	0% (< 1s)	16,7% (< 1s)	não admissível	0% (22331s)
	$\epsilon = 0, 10$	0% (< 1s)	não admissível	não admissível	não admissível
	$\epsilon = 0, 15$	0% (< 1s)	16,7% (< 1s)	não admissível	não admissível
	$\epsilon = 0, 20$	não admissível	não admissível	não admissível	não admissível
[0, 0.8]	$\delta = 0, 1$	0% (< 1s)	16,7% (< 1s)	0% (< 1s)	0% (11s)
	$\delta = 0, 075$	0% (< 1s)	16,7% (< 1s)	0% (5s)	0% (246s)
	$\delta = 0, 05$	0% (< 1s)	16,7% (< 1s)	0% (6s)	0% (1s)
	$\delta = 0, 025$	0% (< 1s)	16,7% (< 1s)	0% (< 1s)	0% (2414s)
	$\epsilon = 0, 00$	0% (< 1s)	16,7% (< 1s)	0% (8s)	0% (284s)
	$\epsilon = 0, 05$	não admissível	não admissível	não admissível	0% (16487s)
	$\epsilon = 0, 10$	não admissível	não admissível	não admissível	* (2 dias)
	$\epsilon = 0, 15$	não admissível	não admissível	não admissível	não admissível
	$\epsilon = 0, 20$	não admissível	não admissível	não admissível	não admissível
[0, 1.0]	$\delta = 0, 1$	0% (< 1s)	16,7% (< 1s)	0% (< 1s)	16,7% (< 1s)
	$\delta = 0, 075$	0% (< 1s)	16,7% (< 1s)	0% (< 1s)	16,7% (< 1s)
	$\delta = 0, 05$	0% (< 1s)	16,7% (< 1s)	0% (< 1s)	16,7% (3s)
	$\delta = 0, 025$	0% (< 1s)	16,7% (< 1s)	0% (4s)	16,7% (55s)
	$\epsilon = 0, 00$	0% (< 1s)	16,7% (< 1s)	0% (13s)	16,7% (1487s)
	$\epsilon = 0, 05$	não admissível	não admissível	não admissível	* (2 dias)
	$\epsilon = 0, 10$	não admissível	não admissível	não admissível	não admissível
	$\epsilon = 0, 15$	não admissível	não admissível	não admissível	não admissível
	$\epsilon = 0, 20$	não admissível	não admissível	não admissível	não admissível

com * as instâncias em relação as quais o CPLEX foi incapaz de fornecer um limite superior para o seu valor óptimo ao fim de dois dias de CPU.

Globalmente, tendo em conta os resultados desta subsecção e da anterior, os modelos *A-Média-Lacetes* e *A-Máx-Arcos* têm uma eficiência similar. Relativamente à formulação *A-Média-Lacetes*, os *gaps* da relaxação linear obtidos são significativamente mais pequenos que os da formulação *A-Máx-Arcos*. Transversalmente, sob o ponto de vista do mecanismo de sobrevivência implementado, as instâncias resolvidas sob *Path Protection* apresentam *gaps* médios mais apertados que as instâncias sob *Path Diversity*. Apesar disso, as instâncias sob o primeiro mecanismo de sobrevivência são, como vimos, mais difíceis de resolver.

III.5.3 Impacto da Variação do Tráfego Original

Com base nos resultados apresentados nas duas subsecções anteriores, mais especificamente nos resultados obtidos através dos modelos *A-Média-Lacetes* e *A-Máx-Arcos* (tabelas III.4, III.5, III.8 e III.9), pretendemos agora comparar as soluções obtidas relativamente à influência que o volume de tráfego total tem no “custo” das soluções, *i.e.*, no número médio e máximo de saltos dessas soluções. Para além deste aspecto, sob o ponto de vista de um operador de rede, interessa também saber qual é o impacto dos erros associados às estimativas de tráfego na admissibilidade das soluções previamente determinadas. Por isto, nesta subsecção focaremos a nossa atenção nos valores óptimos de cada instância (ao invés dos *gaps* das relaxações lineares como nas subsecções anteriores).

Começamos então por notar que ao ignorar as restrições de capacidade – (35) ou (47) – das formulações referidas (o que corresponde a uma situação em que a nova matriz de tráfego R estará bastante abaixo da matriz de tráfego T para a qual a rede original foi dimensionada), obtemos naturalmente limites inferiores para os valores óptimos dos problemas aqui abordados. Estes limites inferiores são condicionados

apenas pela topologia do grafo $N = (X, U)$, pelo que os designaremos doravante por limites “topológicos”. Considere-se a figura III.3 ilustrando um exemplo com dois serviços, $t_{pq} = 1$, $t_{p'q'} = 2$, $D = \Delta = 2$, $H = 3$, em que se pretende implementar o mecanismo de *Path Protection* e minimizar o número médio de saltos. Junto a cada aresta do grafo em cima encontra-se o valor da sua capacidade (b_e), enquanto que nos grafos em baixo, junto a cada arco, se encontram os valores das variáveis w_{ij}^{hpq} (a cheio) e $w_{ij}^{hp'q'}$ (a tracejado). Lacetes mais “largos” correspondem a lacetes que são percorridos em posições (h) superiores. Em baixo, do lado esquerdo, representamos a solução óptima quando ignoramos as restrições de capacidade, em que o número médio de saltos é $6/4$; já do lado direito, podemos ver a solução óptima do problema tendo em conta essas restrições de capacidade, em que o número médio de saltos é $7/4$ (note-se que os valores médios indicados correspondem aos reais valores óptimos).

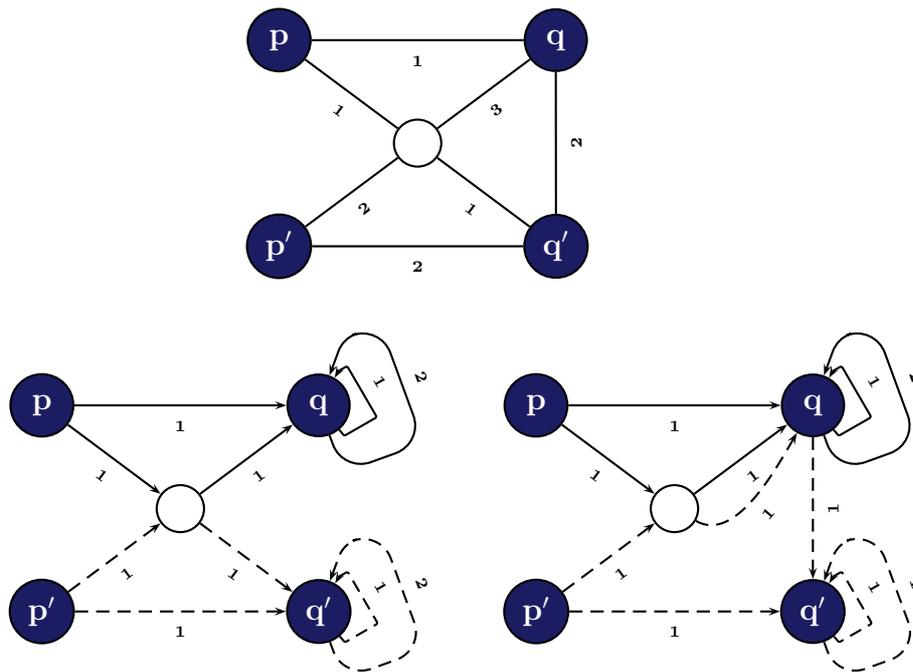


Figura III.3: Ilustração do cálculo dos limites “topológicos”. Em cima, junto a cada aresta encontra-se o valor da sua capacidade (b_e), enquanto que nos grafos em baixo, junto a cada arco, encontram-se os valores das variáveis w_{ij}^{hpq} (a cheio) e $w_{ij}^{hp'q'}$ (a tracejado). Lacetes mais “largos” correspondem a lacetes que são percorridos em posições (h) superiores.

Observe-se que na determinação destes limites topológicos, as formulações utilizadas (ignorando as respectivas restrições de capacidade) foram todas resolvidas em menos de um segundo de CPU.

Pretendendo então avaliar o impacto do volume de tráfego total no “custo” das soluções, bem como o impacto dos erros associados às estimativas de tráfego na admissibilidade das soluções previamente determinadas, separamos os resultados computacionais obtidos em dois conjuntos. Para o primeiro conjunto, consideramos os novos tráfegos r_{pq} abaixo dos tráfegos originais t_{pq} , ou seja, $r_{pq} = (1 - \delta)t_{pq}$, com $\delta = 0.025, 0.05, 0.075$ e 0.1 . Já para o segundo conjunto de resultados, consideramos os novos tráfegos gerados segundo distribuições uniformes nos intervalos $[(1 - \epsilon)t_{pq}, (1 + \epsilon)t_{pq}]$, com $\epsilon = 0, 0.05, 0.1, 0.15$ e 0.2 . Note-se que o parâmetro ϵ corresponde ao erro máximo relativo da estimativa de cada tráfego original t_{pq} , o que nos permitirá averiguar de que forma os erros associados à estimação dos valores de T influenciam a admissibilidade das soluções determinadas no problema de desenho e dimensionamento.

Assim, começando por analisar o primeiro conjunto de resultados, note-se que comparando os valores óptimos das formulações *A-Média-Lacetes* e *A-Máx-Arcos* com os respectivos limites topológicos, podemos avaliar como o volume de tráfego total influencia o “custo” das redes.

As tabelas III.10 e III.11 que se apresentam de seguida contêm os resultados obtidos sob os mecanismos de sobrevivência *Path Diversity* e *Path Protection*, respectivamente. Estas tabelas contêm duas colunas para cada formulação (*A-Média-Lacetes* e *A-Máx-Arcos*), uma para $D = 2$ e outra para $D = 3$. Relativamente a cada instância, podemos consultar a penalidade associada (em percentagem). Esta penalidade é dada pela diferença entre o valor óptimo de cada instância e o respectivo limite topológico, dividida pelo mesmo limite topológico; representa portanto o impacto do volume de tráfego total no “custo” das redes (relativamente ao caso em que as restrições de capacidade são ignoradas). Os valores das penalidades são calculados tendo em conta os reais valores óptimos das soluções (conforme descrito no início das subsecções III.5.1 e III.5.2). Para exemplificar, para a situação da figura III.3 a penalidade associada é

$$(7/4 - 6/4)/(6/4) \times 100\%.$$

Observem-se em primeiro lugar as penalidades que foram obtidas quando consideramos a minimização do número máximo de saltos (colunas relativas à formulação *A-Máx-Arcos*). Neste caso, obtemos uma penalidade de 0% em praticamente todas as instâncias sob ambos os mecanismos de sobrevivência (há apenas um pequeno conjunto de instâncias sob *Path Diversity* – ver tabela III.10, com $D = 2$, tráfegos em $[0, 0.6]$ e $\delta \geq 0.075$ – em que o limite topológico é igual a 3 e os seus valores óptimos são 4). Estes resultados permitem-nos concluir que, regra geral, o valor óptimo para o problema da minimização do número máximo de saltos dos Δ melhores caminhos que suportam cada serviço não é influenciado pelo volume de tráfego total.

Considere-se agora a minimização do número médio de saltos (colunas relativas à formulação *A-Média-Lacetes*). Os resultados obtidos mostram que, à medida que δ aumenta, *i.e.*, quando a matriz de tráfego R se aproxima da matriz de tráfego original T , a penalidade associada a cada instância aumenta também. Uma possível explicação para que isso aconteça reside no facto de que, uma vez que existem capacidades (fixas) nas arestas de U , então um maior volume total de tráfego terá que ser encaminhado através de caminhos com um maior número de saltos. Ainda assim, note-se que para todas as instâncias a penalidade associada é de, no máximo 13.2%. Para $\delta = 0.1$, independentemente do valor de D e do mecanismo de sobrevivência considerado, a penalidade média é de 1.21%, o que consideramos ser um valor bastante pequeno: significa que o encaminhamento de 90% do tráfego original não influencia significativamente, em média, o número médio de saltos dos Δ melhores caminhos de cada serviço, em comparação com as soluções correspondentes aos limites topológicos. Adicionalmente, note-se ainda que para $\delta = 0.025$, o que corresponde a matrizes de tráfego R mais próximas das matrizes T , a penalidade média obtida é de 3.6% para $D = 2$ sob *Path Diversity*, 4.9% para $D = 3$ sob *Path Diversity*, 0.93% para $D = 2$ sob *Path Protection* e 1.89% para $D = 3$ sob *Path Protection*. Estes resultados indicam que: i) o número médio de saltos é menos penalizado sob *Path Protection* do que sob *Path Diversity* (devido ao facto de os caminhos de protecção do primeiro mecanismo não influenciarem o valor da função objectivo, ao passo que sob o segundo mecanismo

Tabela III.10: Impacto do volume de tráfego total no “custo” das soluções obtidas sob *Path Diversity*

S	tráfego	δ	A-Média-Lacetes		A-Máx-Arcos	
			D = 2	D = 3	D = 2	D = 3
6	[0, 0.4]	0,1	0%	0%	0%	0%
		0,075	0%	0%	0%	0%
		0,05	0%	0%	0%	0%
		0,025	0%	0%	0%	0%
6	[0, 0.6]	0,1	0%	0%	0%	0%
		0,075	0%	1,4%	0%	0%
		0,05	6,7%	1,4%	0%	0%
		0,025	8,9%	1,4%	0%	0%
6	[0, 0.8]	0,1	6,5%	1,4%	0%	0%
		0,075	6,5%	4,2%	0%	0%
		0,05	6,5%	5,6%	0%	0%
		0,025	8,7%	9,9%	0%	0%
6	[0, 1.0]	0,1	0%	11,8%	0%	0%
		0,075	0%	13,2%	0%	0%
		0,05	0%	13,2%	0%	0%
		0,025	0%	13,2%	0%	0%
8	[0, 0.4]	0,1	0%	0,6%	0%	0%
		0,075	0%	0,6%	0%	0%
		0,05	0%	0,6%	0%	0%
		0,025	0%	0,6%	0%	0%
8	[0, 0.6]	0,1	1,0%	0,6%	0%	0%
		0,075	1,0%	1,3%	33,3%	0%
		0,05	2,1%	1,9%	33,3%	0%
		0,025	3,1%	3,8%	33,3%	0%
8	[0, 0.8]	0,1	3,1%	4,7%	0%	0%
		0,075	3,1%	5,4%	0%	0%
		0,05	4,1%	6,0%	33,3%	0%
		0,025	6,1%	6,7%	33,3%	0%
8	[0, 1.0]	0,1	1,0%	2,1%	0%	0%
		0,075	2,0%	2,9%	0%	0%
		0,05	2,0%	3,6%	0%	0%
		0,025	2,0%	3,6%	0%	0%

Tabela III.11: Impacto do volume de tráfego total no “custo” das soluções obtidas sob *Path Protection*

S	tráfego	δ	A-Média-Lacetes		A-Máx-Arcos	
			D = 2	D = 3	D = 2	D = 3
6	[0, 0.4]	0,1	0%	0%	0%	0%
		0,075	0%	4,4%	0%	0%
		0,05	0%	4,4%	0%	0%
		0,025	0%	6,7%	0%	0%
6	[0, 0.6]	0,1	0%	0%	0%	0%
		0,075	0%	0%	0%	0%
		0,05	0%	0%	0%	0%
		0,025	0%	2,4%	0%	0%
6	[0, 0.8]	0,1	0%	0%	0%	0%
		0,075	0%	0%	0%	0%
		0,05	0%	0%	0%	0%
		0,025	5,0%	0%	0%	0%
6	[0, 1.0]	0,1	0%	2,4%	0%	0%
		0,075	0%	2,4%	0%	0%
		0,05	0%	2,4%	0%	0%
		0,025	0%	4,9%	0%	0%
8	[0, 0.4]	0,1	0%	0%	0%	0%
		0,075	0%	0%	0%	0%
		0,05	0%	0%	0%	0%
		0,025	0%	0%	0%	0%
8	[0, 0.6]	0,1	0%	1,1%	0%	0%
		0,075	0%	1,1%	0%	0%
		0,05	0%	1,1%	0%	0%
		0,025	0%	1,1%	0%	0%
8	[0, 0.8]	0,1	0%	0%	0%	0%
		0,075	0%	0%	0%	0%
		0,05	0%	0%	0%	0%
		0,025	0%	0%	0%	0%
8	[0, 1.0]	0,1	2,4%	0%	0%	0%
		0,075	2,4%	0%	0%	0%
		0,05	2,4%	0%	0%	0%
		0,025	2,4%	0%	0%	0%

todos os D caminhos estão sujeitos a otimização); e ii) o número médio de saltos é naturalmente penalizado para valores de D maiores.

Analisemos agora o segundo conjunto de resultados. Como referimos antes, averiguar de que forma os erros associados à estimação dos valores de T influenciam a admissibilidade das soluções determinadas no problema de desenho e dimensionamento tem interesse sob o ponto de vista de um operador de rede. Portanto, agora consideramos os resultados obtidos nas duas subsecções anteriores das instâncias em que os novos valores de tráfego r_{pq} foram gerados segundo distribuições uniformes no intervalo $[(1 - \epsilon)t_{pq}, (1 + \epsilon)t_{pq}]$, em que $\epsilon = 0, 0.05, 0.1, 0.15$ e 0.2 representa o erro associado à estimação da matriz T .

Como vimos antes, de uma forma geral os resultado obtidos variam significativamente, na medida em que tanto há instâncias sem soluções admissíveis para (tão só) $\epsilon = 0.05$, como há outras que têm soluções admissíveis com (até) $\epsilon = 0.2$. Na tabela III.12, resumimos o número de instâncias não admissíveis sob cada mecanismo de sobrevivência, cada valor de ϵ (para $\epsilon = 0$ obviamente que todas as instâncias são admissíveis, pois este caso corresponde ao encaminhamento das matrizes de tráfego originais para as quais as redes foram dimensionadas) e cada valor de D .

Tabela III.12: Número de instâncias não admissíveis sob cada mecanismo de sobrevivência, bem como para cada par de valores de ϵ e de D .

ϵ	<i>Path Diversity</i>		<i>Path Protection</i>	
	$D = 2$	$D = 3$	$D = 2$	$D = 3$
0,05	5	2	7	3*
0,10	5	6	7	6*
0,15	6	5	7	7*
0,20	7	7	8	8

Para valores de ϵ pequenos, existe um menor número de instâncias não admissíveis sob *Path Diversity* do que sob *Path Protection*, independentemente do valor de D ; e o número de instâncias não admissíveis é também menor para $D = 3$, independentemente do mecanismo de sobrevivência considerado. Deste modo, podemos concluir que esses casos são menos sensíveis ao erro associado à estimação dos valores de T .

Para valores de ϵ maiores, o número de instâncias admissíveis é bastante reduzido, o que não nos permite efectuar qualquer conclusão. Os valores da tabela assinalados com “*” devem-se ao facto de desconhecermos a solução óptima de algumas instâncias ao fim de dois dias de CPU, pelo que o número de instâncias não admissíveis poderá ser superior ao indicado.

III.6 Conclusões

Neste capítulo abordámos dois problemas de *traffic engineering*, apresentando duas classes de modelos – desagregados e agregados – baseados nos submodelos HOP^{Dpq} e $HOP-A^{Dpq}$ apresentados na secção I.2 do capítulo I.

Infelizmente, apesar das relações estabelecidas nos resultados III.3 e III.7, a nossa experiência computacional indicou que a não há vantagem em usar os modelos desagregados. De facto, como vimos, não só os modelos desagregados têm mais variáveis que os agregados, como, empiricamente, se verificou que os limites inferiores dados pelas relaxações lineares dos modelos desagregados não são melhores que os mesmos limites das relaxações lineares dos modelos agregados. Em verdade, vimos até que a relaxação linear do modelo desagregado originalmente proposto para a minimização do número máximo de saltos (D -Máx-Arcos) não domina a relaxação linear do modelo agregado.

Apesar dos modelos agregados parecerem ser mais eficientes, como trabalho futuro, pretendemos clarificar as questões que ficaram em aberto nos referidos resultados. Seria também interessante tentar diminuir os *gaps* das relaxações lineares dos modelos para a minimização do número máximo de saltos.

Neste capítulo, com base nos testes computacionais efectuados, avaliámos também o impacto do volume de tráfego total no desempenho de uma rede, *i.e.*, averiguámos de que forma é que variações no tráfego influenciam o atraso (médio e máximo) que se verifica nessa rede.

Capítulo IV

Trabalho Futuro

Na sequência do presente trabalho, pensamos que futuramente, tanto para os problemas de *network design* como para os problemas de *traffic engineering* abordados neste trabalho, terá interesse estudar o caso em que as redes devam ser resistentes a duas falhas, e não apenas a uma, apesar de a probabilidade de ocorrência de duas falhas ser pequena. Com esta exigência, teremos obrigatoriamente que considerar $D \geq 3$; note-se ainda que os modelos propostos neste trabalho são facilmente adaptáveis: por exemplo, para $D = 4$ e sob o mecanismo de sobrevivência *Path Protection*, basta considerar $\beta = 50\%$ de forma a garantir protecção total e, relativamente aos problemas de *traffic engineering*, basta considerar $\Delta = D - 2$. Em relação ao problema de desenho e dimensionamento de redes MPLS sobre WDM e à semelhança da comparação efectuada na secção II.2.3, será curioso averiguar se as soluções obtidas sob *Path Protection* com $D = 4$ terão menor custo do que com $D = 3$.

Adicionalmente, uma vez que nos problemas abordados neste trabalho o valor de D é fixo para todos os serviços, pensamos que será interessante estudar o caso em que D pode ser diferente de serviço para serviço. Relativamente ao problema de desenho e dimensionamento, os modelos propostos poderão também ser adaptados com facilidade; no entanto, pensamos que a adaptação dos modelos propostos não será tão imediata para os problemas de *traffic engineering*.

Por último e como tivemos já oportunidade de referir, haverá ainda interesse em clarificar as questões que ficaram em aberto nos resultados III.3 e III.7 no contexto

dos problemas de *traffic engineering*, bem como tentar diminuir os *gaps* das relaxações lineares dos modelos para a minimização do número máximo de saltos.

Nota: partes deste trabalho foram apresentadas em [13], [14] e [15].

Bibliografia

- [1] Ahuja, A.R., Magnanti, T.L., Orlin, J.B., *Network Flows: theory, algorithms and applications*, Prentice Hall, 1993.
- [2] Balakrishnan, A., Altinkemer, K., *Using a Hop-constrained Model to Generate Alternative Communication Network Design*, ORSA Journal on Computing, Vol 4, pp. 192-205, 1992.
- [3] Balakrishnan, A., Magnanti, T.L., Mirchandani, P., *Connectivity-Splitting Models for Survivable Network Design*, Networks, Vol. 43, Issue 1, pp. 10-27, 2004.
- [4] Ben-Ameur, W., *Constrained Length Connectivity and Survivable Networks*, Networks, Vol. 36 (1), pp. 17-33, 2000.
- [5] Bley, A., *On the Complexity of Vertex-Disjoint Length-Restricted Path Problems*, Computational Complexity, Vol. 12, No. 3-4, pp. 131-149, 2003.
- [6] Borne, S., Gourdin, E., Liao, B., Mahjoub, A.R., *Design of Survivable IP-over-Optical Networks*, Annals of Operations Research, Vol. 146, N. 1, pp. 41-73, 2006.
- [7] CPLEX 10.2, *ILOG CPLEX 10.2 Reference Manual*.
- [8] Dahl, G., Foldnes, N., Gouveia, L., *A Note on Hop-Constrained Walk Polytopes*, Operations Research Letters, Vol. 32, pp. 15-22, 2004.

- [9] Dahl, G., Gouveia, L., *On the Directed Hop-Constrained Shortest Path Problem*, Operations Research Letters, Vol. 32, pp. 345-349, 2004.
- [10] Dahl, G., Gouveia, L., Requejo, C., *On Formulations and Methods for the Hop-Constrained Minimum Spanning Tree Problem*, Handbook of Optimization in Telecommunications, pp. 493-515, Springer, 2006.
- [11] Gouveia, L., *Using Variable Redefinition for Computing Lower Bounds for Minimum Spanning and Steiner Trees with Hop Constraints*, INFORMS Journal on Computing, Vol. 10, pp. 180-188, 1998.
- [12] Gouveia, L., Patrício, P., Sousa, A. de, Valadas, R., *MPLS over WDM network design with packet level QoS constraints based on ILP models*, INFOCOM 2003 Proceedings.
- [13] Gouveia, L., Patrício, P.F., Sousa, A.F., *Compact Models for Hop-Constrained Node Survivable Network Design*, Telecommunications Network Planning: Innovations in Pricing, Network Design and Management, Operations Research/Computer Science Interfaces Book Series, Vol. 33, pp. 167-180, Springer, 2006.
- [14] Gouveia, L., Patrício, P.F., Sousa, A.F., *Optimal Survivable Routing with a Small Number of Hops*, Telecommunications Modeling, Policy and Technology, Operations Research/Computer Science Interfaces Book Series, Vol. 44, pp. 253-274, Springer, 2008.
- [15] Gouveia, L., Patrício, P.F., Sousa, A.F., *Hop-Constrained Node Survivable Network Design: An Application to MPLS over WDM*, Networks and Spatial Economics, Vol. 8 (1), pp. 3-21, Springer, 2008.
- [16] Harary, F., *The Maximum Connectivity of a Graph*, Proceedings of the National Academy of Sciences of the United States of America, Vol. 48 (7), pp. 1142-1146, 1962.

- [17] Huygens, D., Mahjoub, A.R., Pesneau, P., *Two Edge-Disjoint Hop-Constrained Paths and Polyhedra*, SIAM Journal on Discrete Mathematics, Vol. 18, Issue 2, pp. 287-312, 2005.
- [18] Huygens, D., Labbé, M., Mahjoub, A.R., Pesneau, P., *Two Edge-Disjoint Hop-Constrained Paths: Valid Inequalities and Branch-and-Cut*, Networks, Vol. 49, Issue 1, pp. 116-133, 2007.
- [19] Itai, A., Perl, Y., Shiloach, Y., *The Complexity of Finding Maximum Disjoint Paths with Length Constraints*, Networks, Vol. 12, Issue 3, pp. 277-286, 1982.
- [20] Juttner, A., Szviatevski, B., Mecs, I., Rajko, Z., *Lagrange relaxation based method for the QoS routing problem*, Proceedings of INFOCOM 2001, Vol. 2, pp. 859-868, 2001.
- [21] Kerivin, H., Mahjoub, A.R., *Design of Survivable Networks: A Survey*, Networks, Vol. 46, Issue 1, pp. 1-21, 2005.
- [22] Koster, A., Zymolka, A., Jager, M., Hulsermann, R., *Demand-wise shared protection for meshed optical networks*, Journal of Network and Systems Management, Vol. 13, Number 1, pp. 35-55, 2005.
- [23] Lawler, E., *Combinatorial Optimization: Networks and Matroids*, Holt, Rinehart and Winston, New York, 1976.
- [24] LeBlanc, L., Chifflet, J., Mahey, P., *Packet Routing in Telecommunication Networks with Path and Flow Restrictions*, INFORMS Journal on Computing, Vol. 11, pp. 188-197, 1999.
- [25] Magnanti, T., Mirchandani, P., Vachani, R., *Modeling and Solving the Two-Facility Capacitated Network Loading Problem*, Operations Research, Vol. 43, pp. 142-157, 1995.

- [26] Magnanti, T., Raghavan, S., *Strong Formulations for Network Design Problems with Connectivity Requirements*, Networks, Vol. 45, Issue 2, pp. 61-79, 2005.
- [27] Magnanti, T., Wolsey, L., *Optimal Trees*, Network Models, Handbooks in Operations Research and Management Science, Vol. 7, pp. 503-615, 1994.
- [28] Mannie, E., *Generalized multi-protocol label switching (GMPLS) architecture*, RFC 3945, Internet Engineering Task Force, Sterling, 2004.
- [29] Martin, R.K., *Generating alternative Mixed-Integer Programming models using variable redefinition*, Operations Research, Vol. 35, pp. 820-831, 1987.
- [30] Martin, R.K., *Large Scale Linear and Integer Optimization: a unified approach*, Kluwer Academic Publishers, 1999.
- [31] Moh, M., Nguyen, B., *QoS-guaranteed one-to-many and many-to-many multicast routing*, Computer Communications, Vol. 26, pp. 652-669, 2003.
- [32] Orda, A., *Routing with end to end QoS guarantees in broadband networks*, IEEE/ACM Transactions on Networking, Vol. 7, No. 3, pp. 365-374, 1999.
- [33] Orlowski, S., Wessaely, R., *The Effect of Hop Limits on Optimal Cost in Survivable Network*, Telecommunications Network Planning: Innovations in Pricing, Network Design and Management, Operations Research/Computer Science Interfaces Book Series, Vol. 33, pp. 151-166, Springer, 2006.
- [34] Raghavan, S., *Formulations and Algorithms for Network Design Problems with Connectivity Requirements*, PhD Thesis, MIT, 1995.
- [35] Raghavan, S., Magnanti, T. L., *Network Connectivity*, In M. Dell Amico, F. Maffioli and S. Martello, Eds., Annotated Bibliographies in Combinatorial Optimization, pp 335-354, John Wiley & Sons, 1997.

- [36] Ramakrishnan, K. G., Rodrigues, M. A., *Optimal routing in shortest-path data networks*, Bell Labs Technical Journal, Vol. 6, No. 1, pp. 117-138, 2001.
- [37] Schrijver, A., *Theory of Linear and Integer Programming*, Wiley, New York, 1986.
- [38] Sharma, V., Hellstrand, F., *Framework for multi-protocol label switching (MPLS)-based recovery*, RFC 3469, Internet Engineering Task Force, Sterling, 2003.
- [39] Sherali, H.D., Ozbay, K., Subramanian, S., *The Time-Dependent Shortest Pair of Disjoint Paths Problem: Complexity, Models and Algorithms*, Networks, Vol. 31, No. 4, pp. 259-272, 1998.
- [40] Stoer, M., *Design of Survivable Networks*, Lecture Notes in Mathematics, N. 1531, Springer Verlag, 1992.
- [41] Swallow, G., *MPLS Advantages for Traffic Engineering*, IEEE Communications Magazine, Vol. 37, N. 12, pp. 54-57, 1999.
- [42] Wolsey, L., *Integer Programming*, John Wiley & Sons, New York, 1998.
- [43] Yuan, X., *Heuristic Algorithms for multiconstrained quality-of-service routing*, IEEE/ ACM Transactions on Networking, Vol. 10, No. 2, pp. 244-256, 2002.