

UNIVERSIDADE DE LISBOA

FACULDADE DE CIÊNCIAS

DEPARTAMENTO DE ESTATÍSTICA E INVESTIGAÇÃO
OPERACIONAL



**PROBLEMA DA ÁRVORE DE SUPORTE
DE CUSTO MÍNIMO COM RESTRIÇÃO
DE GRAU E CUSTOS ASSOCIADOS AOS
NODOS**

Pedro Martins Moura

DOUTORAMENTO EM ESTATÍSTICA E INVESTIGAÇÃO
OPERACIONAL

(Especialidade em Optimização)

2009

UNIVERSIDADE DE LISBOA

FACULDADE DE CIÊNCIAS

DEPARTAMENTO DE ESTATÍSTICA E INVESTIGAÇÃO
OPERACIONAL



**PROBLEMA DA ÁRVORE DE SUPORTE
DE CUSTO MÍNIMO COM RESTRIÇÃO
DE GRAU E CUSTOS ASSOCIADOS AOS
NODOS**

Pedro Martins Moura

DOUTORAMENTO EM ESTATÍSTICA E INVESTIGAÇÃO
OPERACIONAL

(Especialidade em Optimização)

Tese orientada pelo Prof. Doutor Luís Gouveia

2009

RESUMO

Nesta dissertação aborda-se uma variante de Problemas de Árvores em Grafos onde, para além de custos associados às ligações entre os nodos, existem também custos associados ao grau dos nodos. Esta variante é motivada no contexto de redes de telecomunicações, onde estes custos se encontram associados a equipamento de *routing* que é necessário instalar em todo os nodos que estejam ligados a mais do que um nodo na rede. Neste tipo de redes é ainda usual restringir o número máximo de ligações de cada vértice de forma a reduzir interferências de sinal. Esta variante é aplicada a dois problemas clássicos de Árvores em Grafos: o problema da Árvore de Suporte de Custo Mínimo e o problema da Árvore de Steiner com Recolha de Prémios. Incorporando esta variante nas formulações tradicionalmente utilizadas para estes problemas chega-se a modelos não lineares, devido à presença dos custos associados ao grau dos vértices. Duas técnicas de reformulação de modelos são então utilizadas: a técnica de Reformulação por Discretização e a técnica de Reformulação por Caminhos. Ao utilizar qualquer uma destas duas técnicas no modelos tradicionais, obtêm-se modelos lineares. Além disso, as duas técnicas permitem construir conjuntos de desigualdades válidas que ao ser adicionadas a um modelo fortalecem-no no que diz respeito à respectiva relaxação linear. A segunda técnica só pode ser aplicada ao primeiro problema devido à existência de uma estrutura de Saco Mochila, presente neste problema.

Os modelos apresentados são comparados utilizando um conjunto de instâncias com 25 e 50 nodos.

Palavras-chave: Árvore de Suporte, Árvore de Steiner, Reformulação por Discretização, Reformulação por Caminhos, Saco Mochila.

ABSTRACT

In this dissertation a new variant of Tree Problems in Graphs is considered. In this variant, besides the costs associated to the links between the nodes, there also exist costs associated with the degree of the nodes. This variant is motivated in the context of telecommunications networks where this type of costs is associated with routing equipment that has to be installed in every node that is connected with more than one node in the network. In this kind of networks it is usual to limit the number of links in any node to prevent signal interferences. This variant is applied to two classical problems: the Spanning Tree Problem and the Prize-collecting Steiner Tree Problem. By integrating this variant in traditional formulations for these problem one obtains non-linear models, due to the presence of the costs associated with the degree of the nodes. Two reformulations techniques are then used: the Reformulation by Discretization technique and the Reformulation by Paths technique. Linear models are obtained by using any of these two techniques in the traditional models. In addition, different sets of valid inequalities can be constructed with the use of these techniques which, when added to a model, strengthen it in terms of the respective linear relaxation. The second technique can only be applied to the first problem due to the existence of a Knapsack structure present in this problem.

The presented models are compared using a set of instances with 25 and 50 nodes.

Keywords: Spanning Tree, Steiner Tree, Reformulation by Discretization, Shortes-Path Reformulation, Knapsack.

aos meus pais

A hundred times a day I remind myself that my inner and outer life depends on the labor of other, living and dead, and that I must exert myself in order to give in the same measure as I have received and am still receiving.

Albert Einstein

AGRADECIMENTOS

Esta dissertação de doutoramento dificilmente poderia ser levada a cabo sem o apoio de um conjunto de pessoas e instituições que de alguma forma contribuíram para que eu a pudesse realizar. Correndo o risco de me esquecer de alguém, quero deixar aqui os meus agradecimentos pessoais.

Em primeiro lugar quero agradecer ao meu orientador, o Prof. Doutor Luís Gouveia, que sempre confiou em mim, e que aceitou em continuar o trabalho de orientação iniciado com a dissertação de Mestrado. Foram sempre muito estimulantes e educativas as suas ideias e sugestões.

À Faculdade de Ciências e em particular ao Departamento de Estatística e Investigação Operacional do qual faço parte como Assistente desde Julho de 2004, quero agradecer todo o apoio, quer a nível material, quer pela não atribuição de qualquer tipo de serviço docente durante o período de dispensa de serviço que me foi atribuído.

Ao Centro de Investigação Operacional da Fundação da Faculdade de Ciências da Universidade de Lisboa, quero agradecer todos os meios informáticos que me foram colocados ao dispor e também o patrocínio que me permitiu participar em diversas conferências que contribuíram para o meu desenvolvimento científico, necessários a esta dissertação.

Pelo sempre célere apoio logístico e administrativo, quero agradecer à Secretaria e à Biblioteca do Departamento de Estatística e Investigação Operacional.

Ao Prof. Doutor Amaro de Sousa, do Instituto de Telecomunicações da Universidade de Aveiro, quero deixar expresso o meu agradecimento por me ter guiado pelos meandros dos conceitos de redes de telecomunicações, que eu espero ter descrito de uma forma o mais correcta possível.

À São quero expressar o meu terno e eterno obrigado pela amizade e pelo seu olhar paciente e perspicaz.

À Marília quero agradecer pela boa disposição que sempre imperou no nosso gabinete e que facilitou a longas horas de trabalho, assim como, os conselhos úteis com o \LaTeX .

Ao Marco, Mafalda, Catarina, Alda, Paulo, Susana e Eduardo, meus companheiros inseparáveis, um enorme obrigado por estarem sempre presentes.

Pedro Martins Moura
Lisboa, Setembro de 2009

NOTAÇÃO UTILIZADA

Notação associada ao desenho da rede

- V Conjunto de nodos, $V \equiv \{1, \dots, n\}$;
- E Conjunto de arestas (num grafo não orientado), $E \subseteq V^2$;
- $E(i)$ Conjunto de arestas, $e \equiv \{i, j\} \in E$, incidentes no nodo i ;
- A Conjunto de arcos (num grafo orientado), $A \subseteq V^2$;
- $A(S)$ Conjunto de arcos, $(i, j) \in A$, tais que $i, j \in S \subseteq V$ ($A(V) \equiv A$);
- $A(Q, S)$ Conjunto de arcos, $(i, j) \in A$, tais que $i \in Q, j \in S$ e $Q, S \subset V$;
- $A(\bar{S}, S)$ Conjunto de arcos, $(i, j) \in A$, tais que $i \in \bar{S} \equiv V \setminus \{S\}, j \in S$ e $S \subset V$;
- $A^+(i)$ Conjunto de arcos, $(i, j) \in A$, divergentes do nodo i
($A^+(i) \equiv A(\{i\}, V \setminus \{i\})$);
- $A^-(i)$ Conjunto de arcos, $(j, i) \in A$, convergentes no nodo i
($A^-(i) \equiv A(V \setminus \{i\}, \{i\})$);
- $d(i)$ Grau do nodo i no grafo ($d(i) = |E(i)|$);
- $g^+(i)$ Grau externo do nodo i na solução;
- $g^-(i)$ Grau interno do nodo i na solução;
- $g(i)$ Grau do nodo i na solução ($g(i) = g^+(i) + g^-(i)$);
- c_e Custo da aresta $e \in E$;
- c_{ij} Custo do arco $(i, j) \in A$.

Notação tecnológica

- Q Capacidade (número de portas) de cada módulo de *interface*;
- M Quantidade máxima de módulos instalados em cada nodo;
- K_1 Custo (fixo) associado à matriz de *routing*;
- K_2 Custo (variável) associado a cada módulo de *Arco Desagregadas*.

Notação associada aos modelos

- $Adm(P)$ Conjunto de soluções admissíveis do modelo P ;
- $Opt(P)$ Conjunto de soluções óptimas do modelo P ;
- $V(P)$ Valor óptimo do modelo P ;
- LP Relaxação Linear do modelo P ;
- $Conv(\mathcal{X})$ Envoltivo convexo do conjunto \mathcal{X} ;
- $proj_x(\mathcal{R})$ Projecção do poliedro \mathcal{R} no espaço das variáveis x , *i.e.*,
 $\{x : (x, y) \in \mathcal{R} \text{ para algum } y\}$;
- $\Psi^m(\Theta)$ $\sum_{\theta \in \Theta} \psi_\theta^m$;
- $\Psi(\Theta)$ $\sum_{\theta \in \Theta} \psi_\theta$.

Notação diversa

- \mathbb{B}^n $\{0, 1\}^n$, conjunto dos vectores binários de dimensão n

Conteúdo

1	Introdução	1
1.1	Problemas de Árvores	2
1.1.1	Árvore de Suporte de Custo Mínimo	3
1.1.1.1	Restrições de Corte	4
1.1.1.2	Restrições de Eliminação de Subcircuitos	5
1.1.1.3	Sistema de fluxos com comodidade única	7
1.1.1.4	Sistema de fluxos com múltiplas comodidades	10
1.1.2	Árvore de Steiner	14
1.1.3	Árvore de Steiner com Recolha de Prêmios	15
1.1.3.1	Restrições de Corte Generalizadas	20
1.1.3.2	Restrições de Eliminação de Subcircuitos Generalizadas	21
1.1.3.3	Sistema de fluxos com comodidade única	22
1.1.3.4	Sistema de fluxos com múltiplas comodidades	23
1.2	Restrições em Problemas de Árvores	24
1.3	Técnicas de Reformulação	27
1.3.1	Reformulação por discretização	27
1.3.2	Reformulação por caminhos	31
2	Custos dependentes do grau	35

3	O problema (ASupCG)	41
3.1	Modelo não linear	42
3.2	Modelos com variáveis-módulo	44
3.2.1	Modelo de Escolha Múltipla	45
3.2.2	Modelo Incremental	49
3.2.3	Comparação de relaxações lineares: Parte I	52
3.3	Modelo com variáveis-nodo discretizadas	56
3.3.1	Comparação de relaxações lineares: Parte II	59
3.3.2	Resultados Computacionais: Modelo Discretizado	64
3.4	Modelo com variáveis-arco discretizadas	68
3.4.1	Desigualdades <i>Arco Desagregadas</i> no espaço discretizado	75
3.4.2	Resultados Computacionais: Modelos Discretizados fortalecidos	86
3.5	Modelos Árvore de Suporte/Saco Mochila	90
3.5.1	Parte I: Reformulação por caminhos	90
3.5.2	Parte II: Reformulação por caminhos e por discretização	97
3.5.3	Comparação de relaxações lineares: Parte III	103
3.5.4	Resultados Computacionais: Modelo Árvore de Su- porte/Saco Mochila <i>vs</i> Modelo Discretizado	109
3.6	Desigualdades de <i>Arredondamento</i>	113
3.6.1	Resultados Computacionais: Modelos Discretizados for- talecidos com desigualdades de <i>Arredondamento</i>	119
3.7	Conclusões	122
4	O problema (AStRPCG)	123
4.1	Modelo não linear	124
4.2	Modelo com variáveis-nodo discretizadas	127
4.2.1	Resultados Computacionais: Modelo Discretizado	130

<i>CONTEÚDO</i>	xix
4.3 Modelo com variáveis-arco discretizadas	134
4.3.1 Desigualdades <i>Arco Desagregadas</i> no espaço discretizado	137
4.3.2 Resultados Computacionais:	
Modelos Discretizados fortalecidos.	143
5 Conclusões	147
A Demonstrações Suplementares	151
B Geração de instâncias	161
B.1 A distribuição espacial dos nodos	161
B.2 A densidade do grafo	162
B.3 Custos das arestas	163
B.4 Prémios dos nodos clientes	164
B.5 Os parâmetros Q , D e (K_1, K_2)	164
C Escolha do modelo Inteiro	165
C.1 O problema (ASupCG)	165
C.2 O problema (AStRPCG)	167
D Resultados Completos	169

Capítulo 1

Introdução: Revisão de Modelos e Técnicas de Reformulação

Este capítulo encontra-se dividido em três secções principais. Na Secção 1.1 faz-se um resumo histórico de alguns problemas de determinação de árvores em grafos, nomeadamente daqueles que serão abordados nos Capítulos 3 e 4: o Problema da Árvore de Suporte de Custo Mínimo e o Problema da Árvore de Steiner com Recolha de Prémios. Para estes dois problemas são apresentadas algumas formulações que surgem tradicionalmente na literatura, com destaque para aquelas que serão mais tarde utilizadas para os problemas em estudo. A Secção 1.2 introduz alguns problemas de determinação de árvores em grafos com restrições à sua estrutura topológica, dando especial atenção à *restrição de grau máximo* em cada nodo, restrição que será considerada nos dois problemas estudados na dissertação. Na Secção 1.3 são revistas duas técnicas de reformulação de modelos: a reformulação por discretização e a reformulação por caminhos.

1.1 Problemas de Árvores

Qualquer problema de Árvores em Grafos pode ser formulado num grafo conexo não orientado, $G = (V, E, c_e)$, onde $V = \{1, \dots, n\}$ é o conjunto de nodos e $E \subset V^2$ é o conjunto das arestas. Cada aresta $e \in E$ tem um custo, usualmente não negativo, $c_e > 0$, associado à sua utilização. No entanto, num estudo efectuado por Magnanti e Wolsey (ver [43]), os autores concluem que se obtêm formulações mais compactas e/ou com relaxações lineares mais fortes se o problema for definido num grafo orientado. Seguindo este resultado, os problemas em estudo ao longo da dissertação serão modelados numa versão orientada do grafo original, $G = (V, E, c_e)$. O problema da determinação de uma árvore no grafo não orientado torna-se assim no problema da determinação de uma *arborescência* (árvore orientada) num grafo orientado, $G = (V, A, c_{ij})$, obtido do grafo original, onde o conjunto de nodos se mantém e onde A representa o conjunto de arcos. Como qualquer arborescência em G tem de se ser orientada a partir de um nodo específico, é necessário considerar um *nodo-raiz* genericamente designado por r . Pela natureza simétrica dos custos c_{ij} , este nodo r pode ser um nodo escolhido arbitrariamente em V ou num seu subconjunto (quando se conhece, à partida, um conjunto de nodos que obrigatoriamente tem de estar presente na solução) ou um nodo extra, $i = 0$, fictício (quando não se conhecem, à partida, quais os nodos que estarão presentes na solução). Cada aresta do grafo original $e\{i, j\} \in E$ é substituída por dois arcos no grafo orientado, (i, j) , $(j, i) \in A$, com um custo igual ao da aresta. No caso de $r \in V$, cada aresta adjacente ao nodo raiz é substituída por apenas um arco (r, j) , divergente da raiz. Quando a raiz é um nodo fictício, basta adicionar arcos $(0, j)$ com custo nulo, para todos os nodos $j \in V$.

1.1.1 Árvore de Suporte de Custo Mínimo

Um dos mais antigos problemas de determinação de árvores num grafo a ser estudado foi o problema da *Árvore de Suporte de custo mínimo* (ASup) para o qual existem algoritmos que permitem obter a solução óptima em tempo polinomial (ver [40, 50]). O problema é definido da seguinte forma: dado um grafo conexo definido por um conjunto de nodos e um conjunto de arestas com custos associados, o problema consiste em determinar a árvore que inclui todos os nodos do grafo e cujo custo total das arestas escolhidas é mínimo. Num grafo orientado o problema consiste em determinar a arborescência orientada a partir do nodo r , que inclui todos os nodos de V e cuja soma dos custos dos arcos na solução é mínima. Para construir um modelo genérico para o problema (ASup) num grafo orientado, considere-se o seguinte conjunto de variáveis topológicas binárias:

$$\forall (i, j) \in A, \quad x_{ij} = \begin{cases} 1 & \text{se o arco } (i, j) \text{ está na solução} \\ 0 & \text{caso contrário} \end{cases}$$

O modelo genérico para o (ASup) é apresentado na Figura 1.1

$$(SUP) \min \sum_{(i,j) \in A} c_{ij} x_{ij} \quad (1.1)$$

$$s.a : X(A^-(i)) = 1 \quad i \in V \setminus \{r\} \quad (1.2)$$

$$\{ (i, j) \in A : x_{ij} = 1 \} \text{ é conexo} \quad (1.3)$$

$$x_{ij} \in \{0, 1\} \quad (i, j) \in A \quad (1.4)$$

Figura 1.1: Modelo genérico para o (ASup).

As restrições (1.2), designadas por *restrições de grau interno*, garantem que qualquer nodo $i \in V \setminus \{r\}$ está presente na solução, havendo exactamente um arco a chegar a cada nodo excepto para o nodo r , a partir do qual a arborescência está orientada. As restrições (1.3) são aqui apresentadas de uma forma genérica e garantem que a solução é um subgrafo conexo de G . Podem ser representadas por diversos conjuntos equivalentes de restrições lineares. Em conjunto com as restrições (1.2) garantem que a solução tem uma estrutura de arborescência de suporte.

O modelo genérico da Figura 1.1 pode ser usado para obter diversas formulações, utilizando conjuntos de restrições já conhecidos para garantir a conexidade da solução, dos quais se apresentam de seguida quatro conjuntos. Os dois primeiros conjuntos de restrições originam formulações ditas *naturais*, pois apenas envolvem as variáveis topológicas, x_{ij} , enquanto os dois últimos originam formulações ditas *estendidas*, pois para além das variáveis topológicas, x_{ij} , envolvem também um outro conjunto de variáveis adicionais.

1.1.1.1 Restrições de Corte

O primeiro conjunto é uma versão orientada das chamadas *Restrições de Corte RC* (ver [43]):

$$X(A(\overline{S}, S)) \geq 1 \quad S \subseteq V \setminus \{r\}, |S| \geq 2 \quad (1.5)$$

Para qualquer conjunto de nodos que não contenha a raiz, estas restrições, em número exponencial, garantem que haja pelo menos um arco convergente num desses nodos com origem num dos nodos do conjunto complementar (onde se encontra a raiz). Quando $S = \{i\}$, a restrição de corte associada

reduz-se a $X(A^-(i)) \geq 1$ que é dominada pela *restrição de grau interno* (1.2) para o nodo i e portanto só é necessário considerar restrições RC com $|S| \geq 2$. As restrições RC , por si só, não garantem a não existência de subcircuitos. Apenas quando conjugadas com as *restrições de grau interno* (1.2), permitem garantir a estrutura de árvore da solução. De facto, ao somar as *restrições de grau interno* para todos os nodos $i \in V \setminus \{r\}$, obtém-se:

$$\sum_{i \in V \setminus \{r\}} X(A^-(i)) = n - 1$$

Como $X(A^-(r)) = 0$ e $X(A(V)) = \sum_{i \in V} X(A^+(i)) = \sum_{i \in V} X(A^-(i))$ fica garantido que qualquer solução terá apenas $n - 1$ arcos e sendo conexa não poderá conter subcircuitos.

1.1.1.2 Restrições de Eliminação de Subcircuitos

Outro conjunto de restrições lineares que é usual utilizar neste contexto (ver [43]) é uma versão orientada das chamadas *Restrições de Eliminação de Subcircuitos RES*:

$$X(A(S)) \leq |S| - 1 \quad S \subseteq V \setminus \{r\}, |S| \geq 2 \quad (1.6)$$

assim designadas pelo facto de que um circuito que envolva apenas $|S|$ nodos, tem exactamente $|S|$ arcos. A título de exemplo, quando $S \equiv \{i, j\}$ a restrição de eliminação de subcircuitos associada a este conjunto é, $x_{ij} + x_{ji} \leq 1$, que indica que um arco e o seu simétrico não podem estar simultaneamente na solução (ou seja, no grafo original, qualquer aresta na solução só pode ser percorrida num sentido a partir do nodo r). Ao garantir, para todo o conjunto S , que o número máximo de arcos com ambos os nodos contidos em

S é no máximo $|S| - 1$, garante-se a inexistência de subcircuitos em $V \setminus \{r\}$. Visto no grafo $G = (V, A, c_{ij})$, versão orientada do grafo original, não existirem arcos convergentes no nodo r , a presença da raiz em qualquer conjunto S nunca irá criar um circuito, daí só ser necessário garantir as restrições de eliminação de subcircuitos para conjuntos $S \subseteq V \setminus \{r\}$.

Estas restrições por si só não garantem a conexidade da solução¹ mas, quando conjugadas com as *restrições de grau interno* (1.2), garantem a estrutura de árvore da solução. Além disso, utilizando as restrições *RES* para modelar a restrição genérica (1.3), o conjunto de soluções admissíveis da relaxação linear do modelo assim obtido tem todos os pontos extremos inteiros (ver [43]), *i.e.*:

$$Adm(LSUP \sim RES) = conv(\mathcal{AS})$$

onde $(SUP \sim RES) = \{(1.1), (1.2), (1.4) \text{ e } (1.6)\}$ e \mathcal{AS} designa o conjunto de vectores de incidência das árvores de suporte do grafo original $G = (V, E, c_e)$. Na versão não orientada do modelo $(SUP \sim RES)$ o resultado anterior também se verifica. Magnati & Wolsey [43] mostram que, na versão orientada, as restrições *RC* e as restrições *RES* são equivalentes, para qualquer conjunto de nodos que não inclua a raiz. Assim sendo, é possível substituir as restrições *RES* pelas restrições *RC* no modelo, sem que o conjunto de soluções admissíveis da relaxação linear do problema sofra alterações, ou seja:

$$Adm(LSUP \sim RC) = Adm(LSUP \sim RES) = conv(\mathcal{AS})$$

onde $(SUP \sim RC) = \{(1.1), (1.2), (1.4) \text{ e } (1.5)\}$. Na versão não orientada do modelo $(SUP \sim RC)$ o resultado anterior não se verifica. De facto, na versão não orientada, o conjunto $Adm(SUP \sim RC)$ tem geralmente pontos extremos fraccionários.

¹Basta considerar, por exemplo, a solução $x_{ij} = 0 \forall (i, j) \in A$.

1.1.1.3 Sistema de fluxos com comodidade única

A utilização de qualquer um dos conjuntos de restrições *RES* ou *RC* origina modelos com um número exponencial de restrições. Uma forma de reduzir o número de restrições que garantam a conexidade da solução, fazendo uso do facto do problema ser formulado num grafo orientado, consiste em usar formulações *estendidas*. Para tal, é possível combinar a estrutura topológica nas variáveis x_{ij} com uma estrutura de fluxos envolvendo um novo conjunto de variáveis. Obtém-se assim formulações *compactas*, ou seja, com um número polinomial de variáveis e restrições.

Esta estrutura de fluxos serve para modelar caminhos com origem na raiz e destino em cada um dos nodos $i \in V \setminus \{r\}$ e assim garantir a conexidade da solução. Estes caminhos podem ser definidos através de uma única comodidade ou através de múltiplas comodidades. Nos resultados computacionais apresentados nos Capítulos 3 e 4, a sigla *SC* (do inglês *single commodity*) associada a um dado modelo P indicará que no modelo P é utilizado um sistema de fluxos com comodidade única para garantir a conexidade da solução. De igual forma, a sigla *MC* (do inglês *multicommodity*) indicará que no modelo P é utilizado um sistema de fluxos com múltiplas comodidades.

Utilizando apenas uma comodidade é necessário criar, para cada arco $(i, j) \in A$, uma variável $f_{ij} \geq 0$ que represente a quantidade de fluxo que tem origem em r e que percorre o arco (i, j) (neste sentido). Com estas variáveis é possível escrever o sistema de fluxos com comodidade única como é apresentado na Figura 1.2 (considerando $f_{jr} \equiv 0, \forall j \in V \setminus \{r\}$). Segundo a notação usada, $F(A^+(i))$ e $F(A^-(i))$ representam respectivamente, o fluxo total que sai do nodo i e o fluxo total que entra no nodo i .

$$F(A^+(i)) - F(A^-(i)) = \begin{cases} n-1 & i = r \\ -1 & i \in V \setminus \{r\} \end{cases} \quad (1.7a)$$

$$x_{ij} \leq f_{ij} \leq (n-1) \cdot x_{ij} \quad (i, j) \in A \quad (1.7b)$$

$$f_{ij} \geq 0 \quad (i, j) \in A \quad (1.7c)$$

Figura 1.2: Sistema de fluxos SC para o (ASup).

Assim, do nodo raiz ($F(A_r^-) = 0$), saem exactamente $n-1$ unidades de fluxo e em cada um dos restantes nodos, pelas restrições de conservação de fluxo (1.7b), fica retida exactamente uma unidade de fluxo, o que permite interpretar cada variável f_{ij} como representando na solução, o número de nodos na subárvore com raiz no nodo j . As restrições (1.7a) e (1.7b) permitem assim modelar $n-1$ caminhos de fluxo que ligam a raiz a cada um dos restantes nodos. Note-se ainda que a restrição (1.7a) é implicada pelas restrições (1.7b). De facto, somando estas últimas para todos os nodos $i \in V \setminus \{r\}$ é possível deduzir a primeira.

As restrições (1.7c) fazem a ligação entre a estrutura de fluxos e a estrutura topológica da solução: o arco $(i, j) \in A$ tem de estar presente na solução, caso haja fluxo a passar no arco, qualquer seja o seu valor ². Por outro lado, se o arco (i, j) estiver presente na solução então será utilizado para fazer circular pelo menos uma unidade de fluxo.

Quando o valor das variáveis x_{ij} é inteiro, o sistema da Figura 1.2 tem sempre uma solução óptima inteira. No entanto, quando se relaxa a condição de integralidade das variáveis x_{ij} , as restrições de ligação (1.7c) revelam-se fracas (no

²Para os arcos (i, j) , $i \neq r$, é possível ainda apertar o segundo membro da segunda desigualdade para $(n-2) \cdot x_{ij}$, visto apenas nos arcos divergentes da raiz o fluxo poderá ter o valor máximo, $n-1$.

máximo apenas um arco poderá ter fluxo igual a $n - 1$; nos restantes será estritamente inferior). Considere-se $(SUP \sim SC)$ como sendo o modelo de programação inteira, utilizando o sistema da Figura 1.2 para garantir a conexidade da solução, ou seja, $(SUP \sim SC) = \{(1.1), (1.2), (1.4) \text{ e } (1.7a) - (1.7d)\}$. Este modelo é de facto mais fraco que o anterior modelo inteiro $(SUP \sim RC)$, em termos das respectivas relaxações lineares. Considere-se um conjunto $S \subseteq V \setminus \{r\}$, $|S| \geq 2$ e somem-se as respectivas restrições (1.7b):

$$\begin{aligned} \sum_{i \in S} \left(F(A^+(i)) - F(A^-(i)) \right) &= -|S| \Leftrightarrow \\ \Leftrightarrow F(A(\bar{S}, S)) - F(A(S, \bar{S} \setminus \{r\})) &= |S| \Leftrightarrow F(A(\bar{S}, S)) \geq |S| \end{aligned}$$

Somando as restrições de ligação (1.7c) para todo o arco $(i, j) \in A(\bar{S}, S)$ obtém-se

$$(n - 1) \cdot X(A(\bar{S}, S)) \geq F(A(\bar{S}, S))$$

o que conjugando com o anterior resultado permite obter para o conjunto $S \subseteq V \setminus \{r\}$, $|S| \geq 2$:

$$X(A(\bar{S}, S)) \geq \frac{|S|}{n - 1}$$

Ou seja, projectando o sistema de fluxos da Figura 1.2 no espaço das variáveis x , obtêm-se *restrições de corte* do tipo das restrições RC mas mais fracas visto o segundo membro da desigualdade ser $|S|/(n - 1) \leq 1$. Isto permite concluir que:

$$\text{proj}_x(\text{Adm}(LSUP \sim SC)) \subseteq \text{Adm}(LSUP \sim RC) = \text{conv}(\mathcal{AS})$$

1.1.1.4 Sistema de fluxos com múltiplas comodidades

Uma maneira de tornar mais forte o sistema de fluxos consiste em *desagregar* o fluxo, considerando comodidades múltiplas, uma por cada nodo $i \in V \setminus \{r\}$. Para tal considerem-se as variáveis $f_{ij}^k \in \{0, 1\}$, que representam a unidade de fluxo que atravessa o arco (i, j) , enviada da raiz e tendo como destino o nodo $k \in V \setminus \{r\}$. Obviamente ter-se-á, $f_{jr}^k = 0, \forall j, k \in V \setminus \{r\}$ e $f_{kj}^k = 0, \forall (k, j) \in A$. Utilizando a notação $F^k(A^+(i)) = \sum_{(i,j) \in A^+(i)} f_{ij}^k$ e $F^k(A^-(i)) = \sum_{(j,i) \in A^-(i)} f_{ji}^k$, o sistema de fluxos com múltiplas comodidades encontra-se descrito na Figura 1.3.

$$F^k(A^+(i)) - F^k(A^-(i)) = \begin{cases} +1 & k \in V \setminus \{r\}, i = r & (1.8a) \\ 0 & i, k \in V \setminus \{r\}, i \neq k & (1.8b) \\ -1 & k \in V \setminus \{r\}, i = k & (1.8c) \end{cases}$$

$$f_{ij}^k \leq x_{ij} \quad (i, j) \in A, k \in V \setminus \{r, j\} \quad (1.8d)$$

$$f_{ij}^j = x_{ij} \quad (i, j) \in A \quad (1.8e)$$

$$f_{ij}^k \geq 0 \quad (i, j) \in A, k \in V \setminus \{r\} \quad (1.8f)$$

Figura 1.3: Sistema de fluxos *MC* para o (ASup).

Tendo como destino cada um dos nodos $k \in V \setminus \{r\}$, o primeiro conjunto de igualdades indica que do nodo raiz sai exactamente uma unidade de fluxo, o segundo conjunto indica que essa unidade de fluxo não deve ficar retida em nenhum dos outros nodos ao passo que o terceiro conjunto indica que essa unidade deve ficar retida no nodo k . As restrições (1.8d) e (1.8e) relacionam as duas estruturas do modelo e garantem que, caso haja fluxo a atravessar o arco (i, j) , qualquer que seja o seu destino, então esse arco tem de estar presente na solução.

As restrições (1.8e) obrigam a que, caso o arco (i, j) esteja na solução, então o fluxo originado na raiz, que chega ao nodo j , tem de chegar na totalidade pelo arco (i, j) .

Também no caso deste sistema de fluxos, para cada comodidade $k \in V \setminus \{r\}$, a restrição (1.8a) é implicada pelas restrições (1.8b), $\forall i \in V \setminus \{r, k\}$ e pela restrição (1.8c). De facto, somando as restrições (1.8b) $\forall i \in V \setminus \{r, k\}$ com a restrição (1.8c) para um dada comodidade $k \in V \setminus \{r\}$ obtém-se a restrição (1.8a) para essa mesma comodidade:

$$\begin{aligned} & \sum_{i \in V \setminus \{r, k\}} \left(F^k(A^+(i)) - F^k(A^-(i)) \right) - F^k(A^-(k)) = -1 \Leftrightarrow \\ \Leftrightarrow & \left(F^k(A(V \setminus \{r, k\})) + F^k(A^-(k)) \right) - \left(F^k(A(V \setminus \{r, k\})) + F^k(A^+(r)) \right) - \\ & - F^k(A^-(k)) = -1 \Leftrightarrow F^k(A^+(r)) = 1 \end{aligned}$$

Para cada comodidade/nodo k , o sistema de fluxos da Figura 1.3 pode ser visto como um típico problema de fluxos no grafo $G = (V, A)$ entre dois nodos específicos. Os nodos r e k , são respectivamente, o nodo origem e o nodo destino e para cada arco (i, j) , o valor da variável x_{ij} representa a sua capacidade. O problema consiste em enviar uma unidade de fluxo da origem r ao destino k . Como consequência do *Teorema do Fluxo Máximo/Corte de Capacidade Mínima* (ver [1]) existe um vector de fluxo admissível no grafo G (*i.e.*, respeitando as restrições (1.8a)-(1.8f)) se e só se, a capacidade de qualquer corte (\bar{S}, S) , tal que $r \in \bar{S}, k \in S$, for não inferior a 1 (verificando-se assim as restrições *RC* para todos os conjuntos $S \subseteq V \setminus \{r\}$ que contenham o nodo k). Este resultado permite afirmar o seguinte:

$$\text{proj}_x(\text{Adm}(\text{LSUP} \sim \text{MC})) = \text{Adm}(\text{LSUP} \sim \text{RC}) = \text{conv}(\mathcal{AS})$$

onde $(\text{SUP} \sim \text{MC}) = \{(1.1), (1.2), (1.4) \text{ e } (1.8a) - (1.8f)\}$.

O sistema de fluxos da Figura 1.3 é uma versão desagregada do sistema de fluxos da Figura 1.2 e as variáveis de ambos os sistemas relacionam-se da seguinte forma:

$$f_{ij} = \sum_{k \in V \setminus \{r\}} f_{ij}^k \quad (i, j) \in A \quad (1.9)$$

Somando as igualdades (1.8a), $\forall k \in V \setminus \{r\}$, e usando as igualdades de ligação (1.9) obtém-se a igualdade (1.7a). De igual forma, para cada $i \neq r$, somando as igualdades (1.8b), $\forall k \in V \setminus \{r, i\}$, com a igualdade (1.8c) ($k = i$) e usando novamente (1.9), obtém-se a igualdade (1.7b) para o respectivo nodo i . Ao somar as restrições de ligação (1.8d), $\forall k \in V \setminus \{r, j\}$ com a restrição de ligação (1.8e), e usando as igualdades (1.9) obtém-se a restrição de ligação $f_{ij} \leq (n - 1) \cdot x_{ij}$ do sistema SC . Analogamente, ao somar as restrições de (1.8f), $\forall k \in V \setminus \{r, j\}$ com a restrição de ligação (1.8e), e usando as igualdades (1.9) obtém-se a restrição de ligação $x_{ij} \leq f_{ij}$ do sistema SC .

A versão com múltiplas comodidades é de facto mais forte do que a versão com comodidade única; ao passo que, no caso com comodidade única se tem (pelas restrições (1.7c)):

$$x_{ij} \geq \frac{f_{ij}}{n - 1} = \frac{\sum_{k \in V \setminus \{r\}} f_{ij}^k}{n - 1} \quad (i, j) \in A$$

no caso com múltiplas comodidades tem-se (pelas restrições (1.8d)):

$$x_{ij} \geq \max_{k \in V \setminus \{r\}} f_{ij}^k \quad (i, j) \in A$$

ou seja, enquanto no primeiro caso o valor da variável x_{ij} tem de ser não inferior ao valor do fluxo *médio* entre todas as comodidades que usam o arco (i, j) , no segundo caso o valor da variável tem de ser não inferior ao *maior* valor de fluxo de entre todas as comodidades que usam o arco (i, j) . Este raciocínio permite concluir que, embora o número de restrições e de variáveis aumente quando se passa do sistema SC para o sistema MC , a relaxação linear do modelo ($SUP \sim SC$) é mais fraca do que a relaxação linear do modelo ($SUP \sim MC$).

Dos quatro conjuntos de restrições apresentados para descrever as restrições genéricas (1.3), apenas os sistemas de fluxos SC e MC são considerados, aquando da implementação dos modelos descritos no Capítulo 3. A ideia de não escolher apenas o sistema de fluxos mais forte em termos da relaxação linear é a de fazer um estudo comparativo em termos de qualidade do limite inferior *versus* rapidez na obtenção da solução óptima (tanto do modelo inteiro como da sua relaxação linear). A relaxação linear do modelo com um sistema de fluxos SC pode ainda ser fortalecida adicionando desigualdades implicadas pelo modelo com comodidades múltiplas. Como tal, nos resultados computacionais apresentados foram utilizadas as desigualdades RES para conjuntos com $|S| = 2$, $i, j \neq r$:

$$x_{ij} + x_{ji} \leq 1 \quad (i, j) \in A, \quad i < j, \quad i, j \neq r \quad (1.10)$$

que se revelaram bastante eficazes para o efeito ³. A sigla SC^* designará o sistema de fluxos SC reforçado com estas desigualdades.

³Estas desigualdades são implicadas pelo modelo ($SUP \sim MC$) dada a equivalência em termos da respectiva relaxação linear entre este modelo e o modelo ($SUP \sim RES$).

1.1.2 Árvore de Steiner

Um outro problema clássico de árvores em grafos, relacionado com o anterior, surge quando se deixa de exigir que todos os nodos estejam presentes na solução, mas apenas um subconjunto pré-definido de nodos, $V_t \subset V$. Obtém-se assim o problema da *Árvore de Steiner de custo mínimo* (ASt)⁴. Aqui o objectivo consiste em determinar uma *árvore de Steiner*, (*i.e.*, uma árvore que inclui todos os nodos do subconjunto V_t e possivelmente alguns dos outros nodos) que minimize o custo total das arestas escolhidas. Os nodos do subconjunto V_t são geralmente designados por *nodos terminais* e os restantes, $i \in V_s \equiv V \setminus V_t$, são geralmente designados por *nodos de Steiner*. Estes podem ou não ser incluídos na solução por forma a garantir a conexidade da solução e/ou reduzir o custo total da árvore. Os nodos terminais podem ser vistos, no contexto de redes de telecomunicações, como sendo centrais de telecomunicações que obrigatoriamente têm de estar presentes na rede para que haja fornecimento de serviço ao passo que os nodos de Steiner podem ser vistos como pontos de ligação da rede. Este problema, ao contrário do (ASup) é *NP-hard* (ver [25]). Alguns dos primeiros estudos sobre o (ASt) devem-se a Gilbert & Pollack [27] em 1968 e a Dreyfus & Wagner [17] em 1972. Goemans & Myung [30] em 1993 apresentam várias formulações para o problema enquanto Goemans & Williamson [31] em 1995 apresentam uma técnica generalizada de aproximação para uma extensa gama de problemas e aplicam-na ao (ASt). Em 1998, Koch & Martin [38] apresentam um algo-

⁴Este problema deve o seu nome a Jakob Steiner (1796 - 1863), matemático e geómetra suíço. Foi um dos grandes estudiosos do problema de geometria referente à determinação de um ponto adicional cuja soma das distâncias aos três nodos de um triângulo acutângulo seja mínima. Este problema foi pela primeira vez denominado *Problema de Steiner* no livro *What is Mathematics?* (1941) de R. Courant e H. Robbins (ver [12] - 2ª edição, 1996).

ritmo de *branch-and-cut* para resolver o (ASt), baseado numa formulação orientada e utilizando pre-processamento, algoritmos de separação e heurísticas primais. O (ASup) pode ser visto como um caso particular do (ASt) em que todos os nodos são terminais.

Um outro problema, aparentado com o (ASt), compreende para além dos custos das arestas, pesos associados aos nodos da rede (mais concretamente, aos nodos de Steiner já que para os nodos terminais os pesos serão irrelevantes dada a sua presença obrigatória na solução). O objectivo consiste em determinar a *árvore de Steiner* que minimiza a soma do custo total das arestas escolhidas para a solução com o peso total dos nodos (de Steiner) presentes na solução. Define-se assim o problema da *Árvore de Steiner com Pesos nos Vértices*⁵ (AStPV) (ver Goemans [29]). Já antes em 1987, Segev [53] tinha considerado uma variante do (AStPV) em que o conjunto de nodos terminais consiste em apenas um nodo e mostrado que este caso especial é *NP-complete*. O (ASt) pode ser considerado um caso particular do (AStPV) em que o peso de qualquer nodo é nulo.

1.1.3 Árvore de Steiner com Recolha de Prémios

No problema da *Árvore de Steiner com Recolha de Prémios*⁶ (AStRP), além dos custos associados às arestas, existem também prémios, $p_i \geq 0$, associados aos nodos. Mas ao contrário do (AStPV), no (AStRP) não existem nodos terminais, *i.e.*, qualquer nodo pode ou não estar presente na solução. Além disso, o objectivo consiste em determinar a árvore que minimiza a soma do custo total das arestas envolvidas com o prémio total dos nodos que não

⁵Node-Weighted Steiner Tree Problem na terminologia anglo-saxónica.

⁶Prize Collecting Steiner Tree Problem na terminologia anglo-saxónica.

estão presentes na solução (ver [31]). Com a função objectivo construída desta maneira o *prémio* de um nodo pode ser interpretado como uma estimativa da perda potencial de lucro se esse nodo não for incluído na solução. Em certas situações reais os nodos do grafo com "prémios" não negativos, podem ser vistos como potenciais clientes que, em troca de um certo tipo de serviço, oferecem um prémio/contributo; nodos com "prémios" nulos actuam como pontos de ligação por forma a permitir ligações entre os nodos clientes. O objectivo ideal seria ter todos os clientes potenciais na solução por forma a evitar a perda dos respectivos prémios/contributos, mas isto pode encarecer demasiado a solução em termos do custo das ligações da estrutura de árvore que interliga os clientes servidos. Existe portanto, uma contrapartida entre o custo das ligações escolhidas para fazerem parte da solução e os contributos dos clientes servidos. A terminologia "recolha de prémios" (*prize collecting*) foi introduzida pela primeira vez em 1989 por Balas [3] para o problema do *Caixeiro Viajante com Recolha de Prémios*⁷ (CVRP). Neste problema o caixeiro viajante recebe um "prémio" por cada cidade visitada e paga uma penalidade por cada cidade não visitada e o objectivo consiste em minimizar os custos de viagem e as penalidades totais, visitando cidades suficientes para recolher uma quantidade mínima pré-definida de prémio. Desta forma, o conceito de "prémio" utilizado na presente dissertação corresponde ao conceito de "penalidade" utilizado por Balas no (CVRP).

Nos últimos anos muitos trabalhos têm sido publicados sobre o (AStRP). Canuto *et al.* [5] em 2001 descrevem um algoritmo de pesquisa local. Ljubić *et al.* [41] apresentam, com bons resultados, um algoritmo de *branch-and-cut* baseado numa formulação orientada para o problema e utilizam pré-processamento e algoritmos de separação de restrições generalizadas de corte.

⁷Prize Collecting Traveling Salesman Problem na terminologia anglo-saxónica.

Lucena & Resende [42] em 2004, apresentam um algoritmo de *branch-and-cut* baseado numa formulação não orientada para o problema e utilizam pré-processamento e algoritmos de separação de restrições generalizadas de quebra de subcircuitos. Em 2006 Chapovska & Punnen [6] apresentam um estudo sobre variantes do (AStRP) em termos da função objectivo. Em 2006, Uchoa [55] apresenta testes de redução para o (AStRP), embora use a terminologia *nodo terminal* para designar os nodos com prémios estritamente positivo (nodos clientes potenciais). Em 2006 Costa *et al.*, [10] apresentam um estudo sobre a classificação, métodos de resolução e respectivos testes de pré-processamento para algumas generalizações do (ASt), considerando prémios nos nodos. Além do (AStPV) consideram também o *Quota Steiner Tree Problem with Profits* onde o objectivo consiste em minimizar o custo total das arestas da solução garantindo uma quantidade mínima de prémio recolhido, o *Budget Steiner Tree Problem with Profits* onde o objectivo consiste em maximizar o prémio total recolhido garantindo que o custo total das arestas da solução não excede uma limite máximo. Consideram ainda o *Fractional Steiner Tree Problem with Profits*, uma versão não linear onde o valor de qualquer solução é dado pelo quociente entre o prémio total recolhido e a soma de um custo fixo com o custo total das arestas da solução; o objectivo consiste em determinar a solução com maior valor. Mais recentemente, os mesmos autores [11] voltaram ao *Budget Steiner Tree Problem with Profits* e introduziram restrições de salto para limitar superiormente o número de arestas entre qualquer nodo na solução e um dado nodo raiz.

O (AStPV) pode ser visto como um caso particular do (AStRP) em que o prémio associado aos nodos terminais é suficientemente grande. Para cada um dos restantes nodos o seu peso é igual ao simétrico do respectivo prémio.

Para poder criar um modelo genérico para o (AStRP) é preciso ter em conta que ao contrário do problema anterior já não é possível escolher arbitrariamente o nodo-raiz. Pode-se sim, sem perda de generalidade, orientar a solução a partir de um dos nodos com $p_i > 0$, presentes na solução⁸. De facto, a presença na solução de nodos com $p_i = 0$, implica a presença na solução de pelo menos um nodo com $p_i > 0$ e como tal, é sempre possível reduzir o conjunto de "candidatos a raiz" a apenas nodos com $p_i > 0$. Ainda assim, não é possível escolher arbitrariamente um destes nodos já que não se sabe à partida quais deles estarão presentes na solução. É necessário portanto, criar um nodo-raiz fictício, $i = 0$, ligá-lo a todos os nodos com prémio não negativo, garantindo na solução que apenas um destes nodos estará ligado à raiz fictícia, sendo portanto o nodo raiz no grafo original $G = (V, A, c_{ij})$.

Considere-se então a partição do conjunto de nodos, V , em nodos com prémio positivo e nodos com prémio nulo. Os nodos do primeiro conjunto, V_c , são clientes potenciais e serão designados, sem perda de generalidade, por *nodos clientes*. Em oposição, o segundo conjunto, V_{nc} , contém os restantes nodos designados por *nodos não-clientes*. O problema será então modelado num grafo aumentado, $G_0 = (V_0, A_0, c_{ij})$, em que o novo conjunto de nodos é $V_0 = V_c \cup V_{nc} \cup \{0\}$ e o novo conjunto de arcos é $A_0 = A \cup \{(0, j) : j \in V_c\}$.

Quanto às variáveis do modelo, mantêm-se as mesmas variáveis topológicas definidas para o (ASup) estendendo a sua definição aos arcos $(0, j)$, $j \in V_c$. Estas últimas podem também ser interpretados como indicando se o nodo $j \in V_c$ será ou não a raiz da solução no grafo orientado $G = (V, A, c_{ij})$.

⁸Os nodos com prémio $p_i > 0$ são conhecidos à partida.

Recorrendo ainda a um conjunto extra de variáveis binárias z_i definidas como:

$$\forall i \in V, \quad z_i = \begin{cases} 1 & \text{se o nodo } i \text{ está na solução} \\ 0 & \text{caso contrário} \end{cases} \quad (1.11)$$

é possível criar o modelo genérico para o (AStRP) (ver Figura 1.4).

$$(StP) \min \sum_{(i,j) \in A} c_{ij} x_{ij} + \sum_{i \in V_c} p_i \cdot (1 - z_i) \quad (1.12)$$

$$s.a : \quad X(A_0^-(i)) = z_i \quad i \in V \quad (1.13)$$

$$\{ (i, j) \in A_0 : x_{ij} = 1 \} \text{ é conexo} \quad (1.14)$$

$$X(A_0^+(0)) = 1 \quad (1.15)$$

$$x_{ij} \in \{0, 1\} \quad (i, j) \in A_0 \quad (1.16)$$

$$z_i \in \{0, 1\} \quad i \in V \quad (1.17)$$

Figura 1.4: Modelo genérico para o (AStRP).

A função objectivo comporta duas componentes, a primeira referente ao custo total das ligações entre os nodos presentes na solução e a segunda parte diz respeito ao prémio total não recolhido. À parte de uma constante ($\sum_{i \in V_c} p_i$), é possível redefinir a função objectivo como:

$$\max \sum_{i \in V_c} p_i \cdot z_i - \sum_{(i,j) \in A} c_{ij} x_{ij}$$

colocando em evidência o objectivo de recolher o prémio total máximo, descontando o custo total da ligação dos nodos.

As restrições (1.13) generalizam as *restrições de grau interno* escritas para o (ASup) e garantem que apenas os nodos de V , que estiverem presentes na solução, terão um grau interno igual a 1. Destes, e tendo em conta a restrição (1.15), apenas um terá grau interno nulo no grafo $G = (V, A, c_{ij})$, sendo portanto a raiz da solução no grafo original.

Note-se que, devido à simetria dos custos das ligações, uma solução do (AStRP) com p nodos de V_c pode ser representada por p arborescências diferentes no grafo $G_0 = (V_0, A_0, c_{ij})$, dependendo do nodo-cliente que estiver ligado à raiz fictícia. Sendo assim e para eliminar estas situações, reduzindo o espaço de soluções admissíveis do modelo (*StP*), é possível utilizar as seguintes desigualdades de *Assimetria* (ver [7, 41]):

$$x_{0j} \leq 1 - z_i \quad \forall i, j \in V_c, i < j \quad (1.18)$$

Estas restrições fixam a raiz das arborescências no grafo $G = (V, A, c_{ij})$ como sendo o nodo de menor índice, de entre os nodos de V_c presentes na solução. De facto, seja i_1 o menor índice dos nodos clientes pertencentes a uma dada solução. Então, para qualquer nodo $j \in V_c$ presente na solução ter-se-á, $x_{0j} \leq 1 - z_{i_1} = 0$. A restrição (1.15) garante que $x_{0i_1} = 1$ e portanto o nodo i_1 é a raiz da solução no grafo original.

1.1.3.1 Restrições de Corte Generalizadas

Tal como para o (ASup) anteriormente modelado, é possível substituir a restrição genérica (1.14) pelas *Restrições de Corte Generalizadas RCG*, pelas *Restrições de Eliminação de Subcircuitos Generalizadas RESG* ou por sistemas de fluxos, semelhantes aos apresentados anteriormente.

As restrições *RCG* são adaptadas das restrições *RC* apresentadas para o (ASup)⁹:

$$X(A_0(\overline{S}, S)) \geq z_k \quad k \in S, S \subset V, |S| \geq 2 \quad (1.19)$$

Para qualquer conjunto de nodos que não inclua a raiz fictícia, estas restrições garantem a presença de pelo menos um arco convergente em algum dos seus nodos se e só se o conjunto em questão contiver algum nodo da solução. Para o caso, $S \equiv V$, a restrição *RCG* seria escrita como $X(A_0(\{0\}, V)) \geq z_k, \forall k \in V$, dominada pela restrição (1.15) do modelo genérico (*StP*).

1.1.3.2 Restrições de Eliminação de Subcircuitos Generalizadas

As restrições *RESG* são também adaptadas das restrições *RES* apresentadas para o (ASup):

$$X(A(S)) \leq Z(S \setminus \{k\}) \quad k \in S, S \subseteq V, |S| \geq 2$$

Se o conjunto S não incluir nenhum nodo presente na solução (o segundo membro da desigualdade será nulo), então estas desigualdades impedem que qualquer arco com ambos os nodos no conjunto S esteja na solução. Por outro lado, se todos os nodos de S vierem a estar presentes na solução, as respectivas restrições *RESG* resumem-se à versão clássica, com o segundo membro igual a $|S| - 1$. Para qualquer conjunto de nodos em que p deles estejam presentes na solução o segundo membro será $\min(p, p - 1) = p - 1$, ou seja, no conjunto S não podem existir circuitos que envolvam apenas estes p nodos.

⁹Note-se que a raiz fictícia pertence ao conjunto \overline{S} .

Analogamente ao que se passava com as restrições RC e RES também com as restrições RCG e $RESG$ se verifica a equivalência dos respectivos modelos em termos da relaxação linear. De facto, somando as restrições de grau interno (1.13) para todos os nodos $i \in S$, $S \subset V$, $|S| \geq 2$ obtém-se:

$$X(A_0(\bar{S}, S)) + X(A(S)) = Z(S) = Z(S \setminus \{k\}) + z_k \quad \forall k \in S$$

Daqui resulta que as restrições RCG são válidas para um dado conjunto S e um dado nodo $k \in S$, se e só se, as respectivas restrições $RESG$ forem válidas para os mesmos S e k .

1.1.3.3 Sistema de fluxos com comodidade única

Para obter versões dos sistemas de fluxos com comodidade única e com múltiplas comodidades apresentados anteriormente, há que notar que apenas é necessário garantir a chegada de uma unidade de fluxo, a cada um dos nodos de V_c que esteja na solução. Nos restantes nodos apenas é necessário garantir a conservação de fluxo. Existe assim uma economia em termos de restrições de fluxo em relação ao caso em que se garante a chegada de uma unidade de fluxo a todos os nodos que estiverem presentes na solução. Assim, no sistema de fluxos SC (ver Figura 1.5) o fluxo total que sai da raiz fictícia é $Z(V_c)$ e em cada nodo de V_c fica retida uma unidade de fluxo, se e só se, o nodo cliente estiver na solução.

As restrições de ligação entre as variáveis f_{ij} e as variáveis x_{ij} podem ainda ser apertadas para arcos (i, j) , $i \neq 0$ tendo em conta que uma unidade de fluxo é imediatamente consumida por um nodo de V_c , à saída da raiz fictícia.

$$F(A_0^+(i)) - F(A_0^-(i)) = \begin{cases} Z(V_c) & i = 0 \\ -z_i & i \in V_c \\ 0 & i \in V_{nc} \end{cases} \quad \begin{array}{l} (1.20a) \\ (1.20b) \\ (1.20c) \end{array}$$

$$x_{ij} \leq f_{ij} \leq |V_c| \cdot x_{ij} \quad (i, j) \in A_0 \quad (1.20d)$$

$$f_{ij} \geq 0 \quad (i, j) \in A_0 \quad (1.20e)$$

Figura 1.5: Sistema de fluxos SC para o (AStRP).

Assim, para esses arcos podem-se substituir as restrições (1.20d) por:

$$x_{ij} \leq f_{ij} \leq (|V_c| - 1) \cdot x_{ij} \quad (i, j) \in A$$

Estas restrições em conjunto com as restrições de *grau interno* garantem ainda que a passagem de fluxo num dado arco (i, j) , $j \in V_{nc}$, implica a presença do nodo não cliente, j , na solução.

1.1.3.4 Sistema de fluxos com múltiplas comodidades

Analogamente ao sistema de fluxos anterior, no sistema de fluxos com múltiplas comodidades, MC , apresentado na Figura 1.6, apenas se definem comodidades para os nodos de V_c . Tendo em conta que este sistema é uma versão desagregada do sistema anterior, também fica garantida a presença na solução de qualquer nodo não cliente onde haja passagem de fluxo, para qualquer comodidade, $k \in V_c$. Pelas mesmas razões apresentadas para os sistemas de fluxos para o (ASup), também para o (AStRP) o sistema de fluxos MC é mais forte que o sistema de fluxos SC .

$$F^k(A_0^+(i)) - F^k(A_0^-(i)) = \begin{cases} z_k & k \in V_c, i = 0 \\ 0 & k \in V_c, i \in V \setminus \{k\} \\ -z_k & k \in V_c, i = k \end{cases} \quad \begin{array}{l} (1.21a) \\ (1.21b) \\ (1.21c) \end{array}$$

$$f_{ij}^k \leq x_{ij} \quad (i, j) \in A_0, k \in V_c, k \neq i, j \quad (1.21d)$$

$$f_{ij}^j = x_{ij} \quad (i, j) \in A_0, j \in V_c \quad (1.21e)$$

$$f_{ij}^k \geq 0 \quad (i, j) \in A_0, k \in V_c \quad (1.21f)$$

Figura 1.6: Sistema de fluxos *MC* para o (AStRP).

Para a implementação dos modelos descritos no Capítulo 4 apenas foram considerados os sistemas de fluxos das Figuras 1.5 e 1.6 para descrever as restrições (1.14). As razões desta escolha são as mesmas da escolha feita para o (ASup). Em relação aos modelos que utilizam o sistema de fluxos *SC*, utilizaram-se ainda as desigualdades *RESG* para conjuntos com 2 nodos:

$$\begin{aligned} x_{ij} + x_{ji} &\leq z_i & (i, j) \in A, i < j, i, j \neq r \\ x_{ij} + x_{ji} &\leq z_j & (i, j) \in A, i < j, i, j \neq r \end{aligned}$$

por forma a fortalecer as respectivas relaxações lineares. O sistema de fluxos assim obtido foi novamente designado por *SC**.

1.2 Restrições em Problemas de Árvores

Diversas variantes "difíceis" podem ser obtidas do (PASup) quando são introduzidas restrições adicionais à topologia da solução. Entre estas encontra-se o problema da *Árvore de Suporte com restrição de Diâmetro*¹⁰ (ver por exem-

¹⁰*Bounded Diameter Minimum Spanning Tree Problem* ou *Diameter-constrained Minimum Spanning Tree Problem* na terminologia anglo-saxónica.

plo [34, 54]) onde se impõe que a solução seja uma árvore de suporte em que o número de arestas no caminho entre qualquer par de nodos não excede um dado valor inteiro.

No problema da *Árvore de Suporte com restrição de Salto*¹¹ (ver por exemplo [35]) impõe-se um limite superior ao número de arcos no caminho desde um nodo raiz pré-definido até qualquer um dos outros nodos do grafo.

O problema da *Árvore de Suporte com restrição de Capacidade* (ver [32, 48, 56]) é um problema clássico na área de desenho de redes e consiste na determinação de uma árvore de suporte de custo mínimo em que a procura total dos nodos de qualquer subárvore, orientada a partir de um dado nodo raiz, não pode exceder um dado valor.

As restrições de grau em problemas de árvore podem ser aplicadas a um único nodo, a um subconjunto de nodos ou à totalidade dos nodos. Podem além disso limitar inferiormente e/ou superiormente o grau de cada nodo, podendo este limite ser ou não o mesmo para todos os nodos.

As restrições que limitam superiormente o grau dos nodos estão geralmente associadas a problemas em redes de telecomunicações. Dizem respeito à necessidade de impor um número máximo de ligações em cada nodo, de tal forma que seja minimizado o tráfego/interferências de sinais em cada nodo. Um limite inferior no grau de cada nodo (ver [2]) está associado a problemas onde seja necessário distinguir entre nodos "centrais" (centro de distribuição) e nodos "periféricos" (clientes ou consumidores individuais). Aqui um nodo será considerado "central" se estiver afecto a um número mínimo de nodos, caso contrário será um nodo "periférico" (com grau 1).

¹¹*Hop-constrained Minimum Spanning Tree Problem* na terminologia anglo-saxónica.

Quanto aos problemas com restrição de grau máximo, Gabow [24] em 1978 considera o problema em que apenas um nodo (raiz) tem um grau determinado *a priori*, um problema que foi primeiramente estudado por Glover & Klingman [28] e apresenta um algoritmo polinomial ($\mathcal{O}(|E| \cdot \log(\log |V|) + |V| \cdot \log |V|)$) para a sua resolução. Cunha e Lucena [15] em 2005, consideram um limite superior diferente para cada nodo da rede. Quando o grau máximo é igual a 2 para todos os nodos da rede, a solução será um *caminho Hamiltoniano de custo mínimo*, que está relacionado com o problema do *Caixeiro Viajante* e é \mathcal{NP} -difícil [26].

Quando o conjunto de nodos é composto por pontos no plano e os custos das arestas são definidos como a distância euclidiana, pode-se provar (ver [46]) que qualquer árvore de suporte de custo mínimo tem um grau máximo igual a 5 em qualquer nodo. Papadimitriou & Vazirani [49] provaram que, encontrar uma árvore de suporte com um grau máximo em todos os nodos igual a 3, é \mathcal{NP} -difícil, além de terem deixado em aberto que se mantém \mathcal{NP} -difícil mesmo quando o grau máximo é 4. Khuller *et al.* [37], provaram mais tarde que esta suposição estava correcta. Ainda no plano euclidiano, vários autores [39, 47] descreveram algoritmos exactos de *branch- \mathcal{E} -bound* e apresentam [21, 52] heurísticas polinomiais eficientes para o problema no espaço euclidiano.

Os problemas estudados nesta dissertação aplicam-se ao desenho de redes de telecomunicações onde é natural considerar uma restrição que limite superiormente o número de ligações em cada nodo da rede. Esta restrição é motivada pela necessidade de limitar o número de *portas de acesso* em cada nodo devido às limitações de equipamento instalado. Em redes do tipo *wireless*, em que não existem ligações físicas, esta restrição é ainda mais pertinente.

Neste tipo de redes, cada ligação entre dois nodos é estabelecida através de um sistema constituído por um par de antenas emissora/receptora (uma antena em cada nodo da ligação). Estas antenas utilizam um canal de frequência específico escolhido de entre um conjunto limitado de canais. Em cada nodo, por cada ligação a estabelecer existe um sistema destes, por conseguinte, um canal de frequência será utilizado. Para evitar interferências entre os sinais emitidos/recebidos, que poderiam provocar eventuais erros de transmissão, é necessário escolher, para o mesmo nodo, canais de frequência relativamente distintos uns dos outros. A restrição do número de ligações por nodo é consequência do número limitado de canais de frequência diferentes existentes.

1.3 Técnicas de Reformulação

1.3.1 Reformulação por discretização

A técnica de reformulação por discretização pode ser aplicada quando no modelo há um conjunto de variáveis com domínio inteiro limitado e um outro conjunto de variáveis binárias ambas relacionadas com a mesma entidade. O segundo conjunto indica se a entidade está presente na solução e o primeiro conjunto indica um valor associado à entidade caso esta esteja presente na solução. Esta técnica permite substituir estes dois conjuntos de variáveis por um outro conjunto de variáveis binárias que através de um índice extra incorpora toda a informação, todos os possíveis valores que a variável inteira possa tomar. Como exemplo, as formulações *time-dependent* para o problema do Caixeiro Viajante (ver [22]) podem ser vistas como formulações discretizadas para o problema original.

Para melhor entender esta técnica considere-se, genericamente, uma variável inteira x que, caso a entidade a ela associada esteja presente na solução, toma valores no domínio inteiro positivo $\{A, \dots, B\}$, caso contrário será nula. Considere-se ainda a variável binária y tal que, $y = 1$, significa que a entidade está presente na solução, caso contrário, $y = 0$. O seguinte sistema permite modelar o domínio destas variáveis:

$$A \cdot y \leq x \leq B \cdot y \quad (1.22a)$$

$$y \in \{0, 1\} \quad (1.22b)$$

$$x \in \mathbb{N}_0 \quad (1.22c)$$

Criando variáveis binárias "discretizadas" designadas por z_q ($q = A, \dots, B$) é possível substituir as anteriores variáveis por estas últimas através das seguintes relações lineares:

$$x = \sum_{q=A}^B q \cdot z_q \quad (1.23a)$$

$$y = \sum_{q=A}^B z_q \quad (1.23b)$$

$$\sum_{q=A}^B z_q \leq 1 \quad (1.23c)$$

$$z_q \in \{0, 1\} \quad q = A, \dots, B \quad (1.23d)$$

As anteriores desigualdades (1.22a), que relacionam as variáveis x e as variáveis y , tornam-se redundantes depois da substituição usando as restrições (1.23a) e (1.23b) (visto $A \leq q \leq B$), podendo ser removidas do modelo. No fundo, as restrições (1.23a), (1.23c) e (1.23d) garantem o domínio inteiro $\{A, \dots, B\}$ da variável x , caso a entidade esteja presente na solução.

As restrições (1.23b), (1.23c) e (1.23d) garantem o domínio binário da variável y . Além disso, a igualdade (1.23b) mantém a consistência da transformação, garantindo que cada valor da variável x está associado a apenas uma variável binária z_q , exactamente aquela cujo índice é igual ao valor de x .

Este tipo de técnica faz aumentar polinomialmente o número de variáveis do modelo, no entanto permite lidar com funções/restrições "complicadas" em termos lineares, como nos seguintes exemplos.

Exemplo 1.1: Restrições disjuntivas

Considere-se a seguinte restrição disjuntiva na variável inteira x :

$$2 \leq x \leq 3 \quad \text{ou} \quad 5 \leq x \leq 8$$

Esta restrição pode ser modelada recorrendo a uma variável binária ξ e a dois conjuntos de restrições, da seguinte forma ¹²:

$$\begin{aligned} 2 \cdot \xi \leq x \leq 3 \cdot \xi + \mathcal{M} \cdot (1 - \xi) \\ 5 \cdot (1 - \xi) \leq x \leq 8 \cdot (1 - \xi) + \mathcal{M} \cdot \xi \end{aligned}$$

Em alternativa, utilizando a técnica de discretização, substitui-se a variável x recorrendo a um conjunto de variáveis binárias, reescrevendo aquela restrição disjuntiva da seguinte forma:

$$\begin{aligned} x &= \sum_{q=2}^8 q \cdot z_q & \text{e} & \quad \sum_{q=2}^8 z_q = 1 \\ z_q &= 0 & & \quad q = 4, 5 \\ z_q &\in \{0, 1\} & & \quad q = 2, \dots, 8 \end{aligned}$$

¹²Onde \mathcal{M} é um inteiro "suficientemente grande".

Ou seja, basta escrever x em função de z_q através das equações (1.23a), anulando todas as variáveis z_q cujo índice corresponda aos valores que a variável x não pode tomar.

Exemplo 1.2: Funções quadráticas

Seja, $\varphi(x) = a \cdot x^2 + b \cdot x + c$, uma função quadrática em x com domínio inteiro $\{1, \dots, Q\}$. A linearização desta função passa pela seguinte discretização da variável x :

$$\begin{aligned} \varphi(x) &= \sum_{q=1}^Q (a \cdot q^2 + b \cdot q) \cdot z_q + c & \text{e} & \quad \sum_{q=1}^Q z_q = 1 \\ z_q &\in \{0, 1\} & & \quad q = 1, \dots, Q \end{aligned}$$

No fundo, discretizam-se as variáveis inteiras, x^2 e x , recorrendo às mesmas variáveis binárias, z_q .

Exemplo 1.3: Funções lineares por segmentos em x

Considere-se uma partição do domínio I da variável inteira x em L subconjuntos, $I = \bigcup_{i=1}^L I_i$, $I_i \cap I_j = \emptyset$, $i, j = 1, \dots, L, i \neq j$. Seja $\phi(x)$ uma função tal que $\phi(x) = a_i$ se $x \in I_i$. Utilizando a técnica de discretização, a função anterior é reescrita como:

$$\begin{aligned} \phi(x) &= \sum_{i=1}^L a_i \cdot \sum_{q \in I_i} z_q & \text{e} & \quad \sum_{q \in I} z_q = 1 \\ z_q &\in \{0, 1\} & & \quad q \in I \end{aligned}$$

Este último exemplo ilustra como modelar funções de custos do género da que será motivada no próximo capítulo. Outra aplicação importante desta técnica no contexto dos modelos apresentados nos Capítulos 3 e 4 é a possibilidade de criar desigualdades válidas que permitem fortalecer os modelos, em termos da respectiva relaxação linear. Estas desigualdades são bastante intuitivas no espaço das variáveis discretizadas mas difíceis ou mesmo impossíveis de escrever no espaço das variáveis originais.

Esta técnica de discretização foi originalmente utilizada por Gouveia [32] que demonstrou como transformar um modelo de fluxos com comodidade única num modelo discretizado equivalente em termos das respectivas relaxações lineares. Posteriormente esta técnica surgiu aplicada a outros problemas (ver, por exemplo, [8, 9, 18, 33, 56]). Nestes trabalhos, a razão principal para aplicar esta técnica é a criação de novos conjuntos de desigualdades válidas. Porém, em [32] e mais tarde em [18] e [33], o uso de variáveis discretizadas permite ainda modelar versões do problema original com custos não lineares. Os problemas em estudo nesta dissertação enquadram-se neste caso onde a função de custos inclui uma componente não linear, como será descrito no Capítulo 2.

1.3.2 Reformulação por caminhos

Esta técnica é usada em casos em que o modelo (ou parte dele) pode ser reformulado como um problema de *Caminho mais Curto sem restrições* entre dois pares de nodos num grafo expandido, desde que, a dimensão deste grafo seja uma função polinomial da dimensão do grafo original (ver [45] para uma abordagem mais geral desta técnica). Os problemas de caminho mais curto com restrições são bons candidatos à aplicação desta técnica.

Como exemplo, Dahl *et. al.* [16] aplicaram esta técnica ao problema da *Árvore de Suporte de Custo Mínimo com Restrição de Salto* que consiste em determinar a árvore de suporte com custo mínimo onde o único caminho entre um dado nodo raiz, $i = 0$, e qualquer um dos restantes nodos tem no máximo H saltos (arestas). Para cada um destes nodos, $k \in V \setminus \{0\}$, os autores consideram uma relaxação do subproblema da determinação do caminho com restrição de número máximo de saltos, assumindo que o caminho do nodo raiz ao nodo k pode ser não elementar (como o custo de cada arco é não negativo fica garantida a existência de pelo menos uma solução ótima em que os caminhos são elementares). Mostram de seguida que este subproblema relaxado pode ser reformulado como um *problema de caminho mais curto sem restrições* num grafo expandido acíclico. Este grafo é construído usando "níveis", onde cada nodo do grafo original é replicado $H - 1$ vezes (H vezes para o nodo k), uma para cada salto e onde o conjunto de arcos é tal que, qualquer caminho entre o nodo origem e o nodo destino corresponde a um caminho entre o nodo $i = 0$ e o nodo k no grafo original, com não mais do que H saltos.

Exemplo 1.4: Construção de um grafo expandido

A Figura 1.7 mostra um exemplo da expansão de um grafo original com 5 nodos e um número máximo de saltos igual a 3 para um dado nodo $k = 5$ (o grafo expandido pode ainda ser simplificado eliminando todos os arcos (a tracejado) que não façam parte de algum caminho do nodo 0_0 ao nodo 5_3). Qualquer caminho no grafo expandido do nodo 0_0 ao nodo 5_3 representa um caminho do nodo 0 ao nodo 5 com no máximo 3 saltos no grafo original.

No grafo expandido, qualquer caminho que contenha um dos arcos $(5_h, 5_{h+1})$, $h = 1, 2$, representa um caminho com um número de saltos estritamente inferior a 3 no grafo original. Aqueles arcos representariam lacetes no nodo 5 no grafo original.

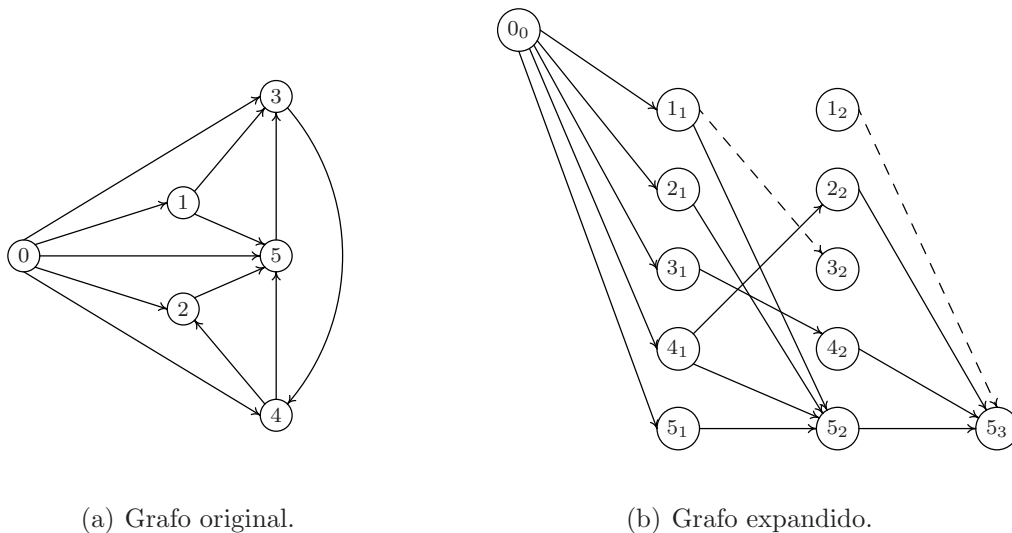


Figura 1.7: Reformulação por caminhos: construção do grafo expandido associado a um grafo original com 5 nodos e $H = 3$.

Utilizando a tradicional formulação de fluxos para o problema da determinação do caminho mais curto no grafo expandido consegue-se obter uma formulação exacta para o problema original obtendo assim uma descrição completa do envolvente convexo do subproblema relaxado. Esta formulação envolve um novo conjunto de variáveis, associadas aos arcos do grafo expandido e como tal fornece uma representação estendida do envolvente convexo do subproblema relaxado.

Além disso é compacta visto o número de novas variáveis e restrições ser uma função polinomial da dimensão do grafo original. Através de uma combinação adequada entre esta descrição estendida e a estrutura de árvore do problema, os autores conseguem obter uma formulação compacta para o problema em questão.

A utilização desta técnica na Secção 3.5 permite não só obter uma descrição completa do envolvente convexo de um subproblema mas também resolver a não linearidade da função objectivo.

Capítulo 2

Custos dependentes do grau: motivação tecnológica

A maior parte dos trabalhos na área de problemas de árvore com restrição de grau nos nodos não consideram custos associados ao grau de cada nodo (ver [15], e as referências nele contidas). Nos dois problemas abordados nesta dissertação, além desta restrição de grau nos nodos, consideram-se também custos associados ao valor do grau dos nodos na solução. A função objectivo torna-se assim mais abrangente, considerando além do custo associado à construção da rede em si (das ligações entre os nodos), um custo associado ao número de ligações que cada nodo tem na solução. Para estabelecer as ligações entre os nodos é necessário instalar equipamento especial nos nodos. É natural que nodos com apenas uma ligação necessitem de menor tecnologia instalada e, conseqüentemente, tenham um custo mais baixo do que nodos com mais do que uma ligação; além disso, quanto maior o número de ligações maior deve ser o custo associado. Este tipo de custos é motivado, por exemplo, no contexto de redes de telecomunicações *wireless* onde estes custos geralmente dominam os custos associados às ligações em si, *i.e.*, o custo

total associado ao equipamento instalado nos nodos é uma parte significativa do custo total da rede.

Em redes por cabo, o custo das ligações tem-se tornado cada vez mais baixo em comparação com os custos do equipamento, pois muitas vezes as ligações são estabelecidas aproveitando cabos já existentes (eventualmente de outro tipo de redes). Quando estes cabos não existem, em vez de construir novos cabos, os operadores preferem, cada vez mais, utilizar tecnologia *wireless*, com um custo de instalação mais baixo.

O custo associado ao grau dos nodos está associado a equipamento *routing* que necessita de ser instalado em nodos ligados a mais do que um nodo na rede. Este equipamento consiste numa matriz de *routing* e num *interface* constituídos por módulos. Cada matriz de *routing* instalada num nodo é responsável pela decisão de reencaminhamento do tráfego que chega a esse nodo. Assim, sempre que um pacote¹ chega a um nodo nestas condições, a matriz de *routing*, através da leitura da informação no cabeçalho do pacote, decide por qual das portas dos módulos de *interface* instalados, este pacote deve ser reenviado (ou se é efectivamente para o nodo em causa). O pacote é então colocado na respectiva fila de espera. Cada módulo de *interface* tem um número fixo de portas podendo ser instalado em qualquer nodo mais do que um módulo de acordo com as necessidades expressas pelo número de ligações no nodo. Como consequência, nem todas as portas dos módulos instalados num dado nodo têm de estar a ser utilizadas. Um nodo que na solução só tenha uma ligação (designado por *nodo folha*) não precisa deste tipo de tecnologia instalada, pois limita-se a receber pacotes cujo destino é o próprio nodo, daí o *custo tecnológico* só ser considerado para nodos com grau superior ou igual a 2 na solução.

¹*Packet* na terminologia anglo-saxónica.

Numa abordagem do (PASTRP) é possível considerar adicionalmente que os nodos com prémio nulo, apenas representam pontos de ligação na rede e portanto limitam-se a receber e a enviar pacotes não sendo eles próprios destino de nenhum pacote enviado pela rede. Assim sendo, um nodo deste tipo só estará presente em qualquer solução, se o seu grau for superior ou igual a 2. Acresce ainda que, no caso do seu grau ser efectivamente igual a 2, não deverá ter *custo tecnológico* associado, visto qualquer pacote que chegue por uma das duas ligações terá de ser obrigatoriamente reencaminhado através da outra ligação, não havendo necessidade da instalação da matriz de *routing*. Além disso, também não é necessário instalar um módulo neste nodo: as duas ligações podem ser ligadas directamente, virtualmente eliminando o nodo. O *custo tecnológico* destes nodos só é então considerado quando o seu grau é superior ou igual a 3 na solução.

O custo associado ao número de ligações de cada nodo não folha pode então ser dado em função do número de módulos instalados nesse nodo, ou seja, para cada nodo:

$$\phi^m = K_1 + m \cdot K_2, \forall m = 1, \dots, M \quad (2.1)$$

Os parâmetros K_1 e K_2 representam respectivamente, o custo da matriz de *routing* e o custo do módulo de *interface*; m representa o número de módulos necessários para estabelecer todas as ligações de recepção/reenvio no nodo em causa.

Como o número de módulos instalados depende do grau do nodo, a função (2.1) deve ser definida em termos de $g(i)$, o grau do nodo i em qualquer solução admissível. Considere-se, para tal, o parâmetro Q como sendo a capacidade de qualquer módulo (o número fixo de portas em qualquer módulo

de *interface*), e seja D o número máximo de ligações que qualquer nodo pode ter, *i.e.*, $g(i) \leq D$ para qualquer nodo i . Sendo assim, é possível definir o *maior número de módulos* que necessitam de ser instalados em qualquer nodo como sendo² $M = \left\lceil \frac{D}{Q} \right\rceil$.

O custo associado ao número de ligações de qualquer nodo i , definido em função do grau do nodo na solução, é dado explicitamente pela função (2.2):

$$\Phi(g(i)) = \begin{cases} 0 & g(i) = 0, 1 \\ \phi^1 & 2 \leq g(i) \leq Q \\ \phi^m & (m-1) \cdot Q + 1 \leq g(i) \leq m \cdot Q, \quad m = 2, \dots, M-1 \\ \phi^M & (M-1) \cdot Q + 1 \leq g(i) \leq D \end{cases} \quad (2.2)$$

No Capítulo 4 esta função aparece redefinida para os nodos cujo prémio é nulo devido às razões explicadas anteriormente. Na Figura 2.1 é apresentado um exemplo de uma função $\Phi(g(i))$, para uma instância com $Q = 3$, $(K_1, K_2) = (100, 20)$ e $D = 8$, onde se pode observar que a função é côncava e linear por segmentos.

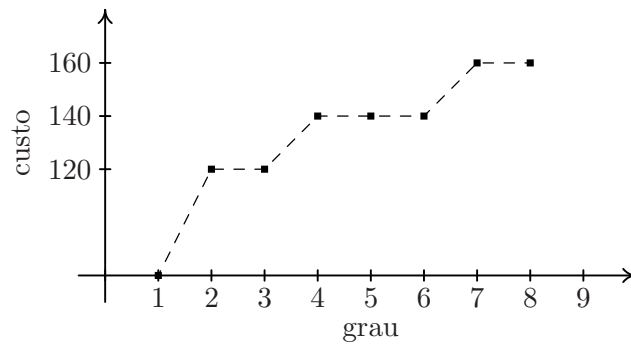
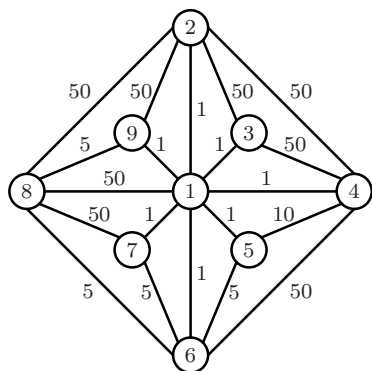


Figura 2.1: Função de custos $\Phi(\cdot)$ com $(K_1, K_2) = (100, 20)$, $Q = 3$ e $D = 8$.

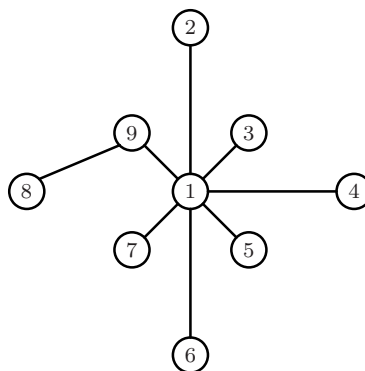
² $\lceil p \rceil = \min\{x \in \mathbb{Z} : x \geq p\}$

Exemplo 2.1: Efeito da restrição de grau e função de custos de grau

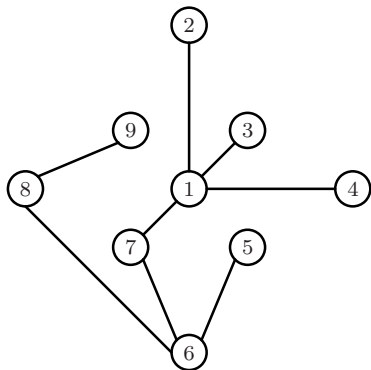
Na Figura 2.2 apresenta-se um exemplo de como se comporta o (ASup) face à inserção de restrições de grau e custos de grau.



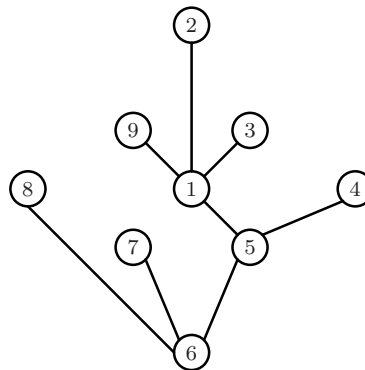
(a) Rede inicial.



(b) Árvore de suporte de custo mínimo.



(c) Árvore de suporte de custo mínimo com restrição de grau.



(d) Árvore de suporte de custo mínimo com restrição de grau e função de custos de grau.

Figura 2.2: Aplicação da restrição de grau máximo ($D = 4$) e de uma função de custos de grau.

Na Figura 2.2(b) é apresentada a solução do (ASup) – sem a adição de qualquer restrição ou custos extra – com custo 12.

Ao introduzir um grau máximo igual a 4, esta solução deixa de ser admissível, obtendo-se agora a solução apresentada na Figura 2.2(c), com custo 24. Porém, quando se consideram módulos de capacidade $Q = 3$ e custos de equipamento, $K_1 = 100$ e $K_2 = 20$, ou seja, ao introduzir a função de custos de grau:

$$\Phi(g(i)) = \begin{cases} 0 & \text{se } g(i) = 1 \\ 120 & \text{se } g(i) = 2, 3 \\ 140 & \text{se } g(i) = 4 \end{cases}$$

a solução da Figura 2.2(c) deixa de ser ótima (o seu custo total passa a ser de 524). A nova solução ótima passa agora a ser a solução da Figura 2.2(d) com custo igual a 409 (o custo total de arestas é de 29).

Este exemplo ilustra o efeito que a função de custos de grau desempenha no (re)desenhar da solução/árvore de suporte.

Nos dois capítulos seguintes serão abordados dois problemas de determinação de árvores que, para além de incorporarem uma restrição de grau máximo em todos os nodos como a descrita na secção anterior, incorporam também uma função de custos do tipo da função (2.2). Assim, no Capítulo 3 é abordado o problema da *Árvore de Suporte de custo mínimo com restrição de grau e Custos dependentes do Grau* (designado por (ASupCG)) enquanto no Capítulo 4 é abordado o problema da *Árvore de Steiner com Recolha de Prémios com restrição de grau e Custos dependentes do Grau* (designado por (AStRPCG)).

Capítulo 3

O problema (ASupCG)

Neste capítulo analisam-se vários modelos para o (ASupCG), comparando-os em termos das respectivas relaxações lineares. Estes modelos são baseados no modelo genérico enunciado na Secção 1.1.1 para o (ASup) (ver Figura 1.1) e serão também aqui apresentados de uma forma genérica no que diz respeito às restrições de conectividade. Sendo assim, aquando da apresentação de cada modelo será omitida a designação "genérico", salvo caso em contrário.

Na Secção 3.1 apresenta-se um primeiro modelo básico, recorrendo a dois conjuntos de variáveis: o primeiro associado aos arcos do grafo e o segundo associado aos nodos do grafo. No entanto este modelo é não linear, no que diz respeito à função objectivo. Aplicando neste modelo as duas técnicas de reformulação introduzidas no Capítulo 1 (Secção 1.3) obtêm-se os modelos lineares das secções seguintes.

Na Secção 3.2 apresentam-se dois modelos usuais na literatura para problemas de *Network Loading*. Estes modelos utilizam variáveis binárias relacionadas com o número de módulos instalados em cada nodo e são equivalentes no que diz respeito às respectivas relaxações lineares.

Na Secção 3.3 introduz-se um modelo que utiliza a técnica de discretização aplicada às variáveis-nodo do modelo não linear e mostra-se que é equivalente aos dois modelos anteriores no que diz respeito à relaxação linear.

Na Secção 3.4 apresenta-se um modelo linear onde, para além das variáveis-nodo discretizadas, as variáveis-arco são também discretizadas, dando origem a um novo conjunto de variáveis-arco discretizadas. Com este modelo estendido é possível obter desigualdades válidas que permitam fortalecer o modelo anterior. Na Secção 3.4.1 mostra-se que estas desigualdades permitem ainda obter, por projecção, desigualdades válidas no espaço das variáveis do modelo da Secção 3.3.

Os dois últimos modelos lineares, apresentados na Secção 3.5, aproveitam uma estrutura de Saco Mochila, implícita no modelo não linear para, através da técnica de reformulação por caminhos enunciada na Secção 1.3.2, linearizar a função objectivo. O segundo destes modelos utiliza ainda a técnica de discretização o que permite fazer a ligação entre os vários modelos apresentados. A estrutura de Saco Mochila motiva também um conjunto de desigualdades válidas apresentadas no final da secção.

No final de cada uma das secções 3.3, 3.4, 3.5 e 3.6 apresentam-se resultados computacionais para avaliar a qualidade dos limites inferiores obtidos com a relaxação linear de cada um dos modelos inteiros apresentados.

3.1 Modelo não linear

Neste primeiro modelo para o problema utilizam-se as mesmas variáveis topológicas, x_{ij} , definidas na Secção 1.1. Para modelar as restrições de grau e escrever a componente de custos de equipamento definida no Capítulo 2 é necessário criar variáveis para indicar o grau de cada nodo.

Note-se porém que, na variante *Árvore de Suporte*, todos os nodos se encontram presentes na solução e como tal o grau de qualquer nodo é sempre não inferior a 1. Sendo assim, basta considerar variáveis que representem o grau do nodo i , para além da unidade. Considerem-se então as variáveis inteiras U_i tal que, na solução o grau do nodo i é, $g(i) = U_i + 1, \forall i \in V$. O modelo genérico encontra-se descrito na Figura 3.1.

$$(SUP_{NL}) \min \sum_{(i,j) \in A} c_{ij} x_{ij} + \sum_{i \in V} \Phi(U_i + 1) \quad (3.1)$$

$$s.a : X(A^-(i)) = 1 \quad i \in V \setminus \{r\} \quad (1.2)$$

$$\{ (i, j) \in A : x_{ij} = 1 \} \text{ é conexo} \quad (1.3)$$

$$X(A^+(i)) = \begin{cases} U_i & i \in V \setminus \{r\} \\ U_r + 1 & i = r \end{cases} \quad (3.2a)$$

$$U_r + 1 \quad i = r \quad (3.2b)$$

$$U_i \leq D - 1 \quad i \in V \quad (3.3)$$

$$x_{ij} \in \{0, 1\} \quad (i, j) \in A \quad (1.4)$$

$$U_i \in \mathbb{N}_0 \quad i \in V \quad (3.4)$$

Figura 3.1: Modelo não linear para o (ASupCG).

A função objectivo (3.1) é composta por duas componentes: a primeira diz respeito ao custo das ligações entre os nodos da solução; a segunda refere-se ao custo associado ao grau dos nodos e é definida à custa da função (2.2), descrita no Capítulo 2. As restrições (1.3), apresentadas de uma forma genérica, podem ser substituídas por qualquer um dos conjuntos apresentados na Secção 1.1.1. As restrições (3.2a) e (3.2b), designadas por *restrições de grau externo*, definem o número de arcos divergentes no nodo i .

Para qualquer nodo $i \in V \setminus \{r\}$, o seu grau externo na solução é $g^+(i) = X(A^+(i)) = U_i$ o que, em conjunto com a respectiva *restrição de grau interno*, (1.2) (que garante que $g^-(i) = 1$), implica que na solução o grau do nodo i é de facto $U_i + 1$. No caso da raiz, já que o seu grau interno na solução é nulo, o seu grau na solução, $U_r + 1$, será mesmo igual ao seu grau externo na solução, $g^+(r) = X(A^+(r))$.

As restrições (3.3) definem, para qualquer nodo de V , o seu grau máximo na solução, $g(i) = U_i + 1 \leq D$ e as restrições (1.4) e (3.4) definem o domínio das variáveis envolvidas no modelo.

Atendendo à descrição dos custos de equipamento feita no Capítulo 2, a função objectivo (3.1) do modelo (SUP_{NL}) é não linear. Nas secções seguintes através das técnicas enunciadas na Secção 1.3 essa questão será resolvida.

3.2 Modelos com variáveis-módulo

Uma forma de linearizar a função objectivo (3.1) consiste em recorrer a uma ideia utilizada em modelos para o problema de *Network Loading* (ver [4, 13, 14]) onde a função objectivo comporta, geralmente, uma parcela linear por segmentos, como é o caso da função objectivo (3.1). Nestes modelos utilizam-se variáveis binárias associadas ao número de equipamentos a instalar em cada nodo por forma a garantir certas restrições de capacidade de tráfego. Estas variáveis estão associadas a cada salto de uma função de custos do género da função (2.2) apresentada na Figura 2.1.

Apresentam-se de seguida dois tipos de modelos: o modelo de Escolha Múltipla e o modelo Incremental. No primeiro, cada variável binária associada ao equipamento indica a escolha de *quantos* módulos devem ser instalados em cada nodo.

Estas variáveis podem resultar, em certa medida, da aplicação da técnica de discretização ao seguinte conjunto de variáveis: variáveis inteiras que representam o *número de módulos a instalar* em cada nodo i e variáveis binárias que indicam se o nodo i é ou não folha. No segundo modelo as variáveis binárias indicam se o m -ésimo módulo é ou não instalado (em caso afirmativo, implicam a instalação do h -ésimo módulo, $\forall h < m$).

3.2.1 Modelo de Escolha Múltipla

Neste primeiro modelo as variáveis binárias associadas ao número de módulos instalados em cada nodo são definidas, para qualquer nodo $i \in V$, como:

$$v_i^m = \begin{cases} 1 & , \text{ se } m \text{ módulos são instalados em } i \\ 0 & , \text{ caso contrário} \end{cases} \quad 1 \leq m \leq M$$

Por questões de consistência, para cada nodo i , no máximo uma destas novas variáveis pode ser igual a 1 em qualquer solução admissível e como tal é necessário acrescentar ao modelo as restrições $\sum_{m=1}^M v_i^m \leq 1$, $\forall i \in V$. Como o grau de cada nodo na solução depende do número de módulos instalados nesse nodo, as variáveis v_i^m encontram-se relacionadas com as variáveis inteiras U_i da seguinte forma, para qualquer nodo $i \in V$:

$$\begin{aligned} v_i^1 &\leq U_i \leq (Q - 1) \cdot v_i^1 \\ (m - 1)Q \cdot v_i^m &\leq U_i \leq (mQ - 1) \cdot v_i^m \quad m \geq 2 \end{aligned} \quad (3.5)$$

Assim, quando $v_i^m = 1$, exactamente m módulos são instalados no nodo i , o que implica (ver definição da função $\Phi(U_i + 1)$) um custo igual a ϕ^m .

Se para um dado nodo i , estas variáveis forem todas nulas então, na solução óptima $U_i = 0$ (o grau é um, o nodo é uma folha e não é necessário instalar tecnologia especial). O modelo linear (SUP_{EM}), utilizando estas variáveis associadas ao número de módulos, é apresentado na Figura 3.2.

$$(SUP_{EM}) \min \sum_{(i,j) \in A} c_{ij} x_{ij} + \sum_{i \in V} \sum_{m=1}^M \phi^m v_i^m \quad (3.6)$$

$$s.a : X(A^-(i)) = 1 \quad i \in V \setminus \{r\} \quad (1.2)$$

$$\{ (i,j) \in A : x_{ij} = 1 \} \text{ é conexo} \quad (1.3)$$

$$X(A^+(i)) = \begin{cases} U_i & i \in V \setminus \{r\} \\ U_r + 1 & i = r \end{cases} \quad (3.2a)$$

$$U_i \leq D - 1 \quad i \in V \quad (3.3)$$

$$U_i \leq \sum_{m=1}^M (mQ - 1) \cdot v_i^m \quad i \in V \quad (3.7)$$

$$\sum_{m=1}^M v_i^m \leq 1 \quad i \in V \quad (3.8)$$

$$x_{ij} \in \{0, 1\} \quad (i, j) \in A \quad (1.4)$$

$$U_i \in \mathbb{N}_0 \quad i \in V \quad (3.4)$$

$$v_i^m \in \{0, 1\} \quad i \in V, 1 \leq m \leq M \quad (3.9)$$

Figura 3.2: Modelo Linear com variáveis-módulo discretizadas para o (ASupCG): Modelo de Escolha Múltipla.

Agora, ao contrário da função objectivo do modelo (SUP_{NL}), a parte referente aos custos de grau é linear, visto ser dada em função das novas variáveis binárias.

As restrições (3.7), limitam superiormente o grau do nodo i , em termos das variáveis-módulo e em conjunto com as restrições de consistência (3.8) garantem que o número necessário de módulos é instalado no nodo i .

Assim, se forem instalados m módulos no nodo i ($U_i \leq mQ - 1$), o grau do nodo i na solução será no máximo mQ , ou seja, a capacidade total dos módulos instalados no nodo i .

Por outro lado, tendo em conta que a sequência de custos $\{\phi^1, \phi^2, \dots, \phi^M\}$ é estritamente crescente, não mais do que o número necessário de módulos será instalado no nodo i . Fica assim garantido que, em qualquer solução óptima para o problema, $U_i \geq (m - 1) \cdot Q$ ($U_i \geq 1$ no caso de $m = 1$), caso sejam instalados m módulos no nodo i ; caso contrário, existiria uma solução admissível com custo inferior e com $l < m$ módulos instalados no nodo i . Sendo assim, não é necessário incluir no modelo os limites inferiores para as variáveis U_i , apresentados nas inequações (3.5).

Caso nenhum módulo seja instalado no nodo i , $v_i^m = 0, \forall m = 1, \dots, M$, as restrições (3.7) garantem que esse nodo é uma folha. Note-se ainda que, estas restrições podem ser "apertadas" tendo em conta que se $v_i^M = 1$ então, pelo facto de que $M = \lceil \frac{D}{Q} \rceil$, se tem $U_i \leq D - 1$. Assim sendo, as restrições (3.7) podem ser substituídas no modelo (SUP_{EM}) por:

$$U_i \leq \sum_{m=1}^{M-1} (mQ - 1) \cdot v_i^m + (D - 1) \cdot v_i^M \quad i \in V \quad (3.7')$$

Estas restrições, em conjunto com as restrições (3.8), tornam redundantes as restrições de grau máximo, (3.3) e assim, o modelo (SUP_{EM}) pode ser redefinido como:

$$(SUP_{EM}) \min \sum_{(i,j) \in A} c_{ij} x_{ij} + \sum_{i \in V} \sum_{m=1}^M \phi^m v_i^m \quad (3.6)$$

$$s.a : (1.2), (1.3), (1.4), (3.2a), (3.2b), (3.4), (3.7'), (3.8) \text{ e } (3.9)$$

O modelo (SUP_{EM}) apresentado na Figura 3.2 pode ser considerado como um modelo "discretizado" obtido a partir do modelo não linear, (SUP_{NL}) , através da aplicação da técnica apresentada na Secção 1.3.1. Para isso é necessário considerar as seguintes variáveis inteiras extra: $V_i \in \{1, \dots, M\}$ representa o número de módulos instalados no nodo i ; $b_i \in \{0, 1\}$ toma valor 1 caso o grau do nodo i seja estritamente superior a 1. As restrições que ligam estas variáveis às variáveis U_i do modelo não linear são as seguintes:

$$b_i \leq V_i \leq M \cdot b_i \quad (3.10)$$

$$U_i \leq Q \cdot V_i - b_i \quad (3.11)$$

A interpretação destas desigualdades é a seguinte: se o nodo i for uma folha na solução, $b_i = 0$, o que implica que $V_i = U_i = 0$, nenhum módulo é instalado no nodo i e o seu grau é 1. Se o nodo i tem grau superior a 1, $b_i = 1$, o número de módulos instalado no nodo é $V_i \in \{1, \dots, M\}$ e o seu grau, $U_i + 1$, não excede a capacidade total dos módulos instalados, $Q \cdot V_i$.

Ao introduzir estas variáveis e as desigualdades (3.10) e (3.11) no modelo não linear (SUP_{NL}) , obtém-se novamente um modelo válido para o problema e a componente não linear da função objectivo pode ser reescrita como:

$$\sum_{i \in V} (K_1 \cdot b_i + K_2 \cdot V_i)$$

o que permite obter um modelo linear.

Aplicando de seguida a técnica de discretização às variáveis V_i e b_i , utilizando para isso as variáveis binárias anteriormente definidas, tem-se:

$$\begin{aligned} V_i &= \sum_{m=1}^M m \cdot v_i^m \\ b_i &= \sum_{m=1}^M v_i^m \\ \sum_{m=1}^M v_i^m &\leq 1 \\ v_i^m &\in \{0, 1\} \quad m = 1, \dots, M \end{aligned}$$

Estas relações entre as variáveis V_i e b_i e as variáveis v_i^m permitem remover as restrições (3.10) por se tornarem redundantes e ao reescrever as restrições (3.11), substituindo as variáveis V_i e b_i , obtêm-se as restrições (3.7) do modelo linear, (SUP_{EM}), apresentado na Figura 3.2. Além disso, a parte da função objectivo escrita nas variáveis V_i e b_i , passa a ficar definida como:

$$\sum_{i \in V} (K_1 \cdot b_i + K_2 \cdot V_i) = \sum_{i \in V} \left(\sum_{m=1}^M (K_1 + K_2 \cdot m) v_i^m \right) = \sum_{i \in V} \sum_{m=1}^M \phi^m v_i^m.$$

ou seja, a parte da função objectivo (3.6), do modelo (SUP_{EM}), associada às variáveis v_i^m .

3.2.2 Modelo Incremental

Neste segundo modelo com variáveis associadas ao número de módulos instalados em cada nodo, as variáveis binárias a introduzir são definidas como:

$$t_i^m = \begin{cases} 1 & , \text{ se o } m\text{-ésimo módulo é instalado em } i \\ 0 & , \text{ caso contrário} \end{cases} \quad 1 \leq m \leq M$$

Caso o m -ésimo módulo seja instalado, o grau do nodo deve ser pelo menos igual a $(m - 1) \cdot Q$, as variáveis grau U_i relacionam-se linearmente com as variáveis t_i^m da seguinte forma:

$$\begin{aligned} t_i^1 &\leq U_i \\ (m - 1)Q \cdot t_i^m &\leq U_i \quad m \geq 2 \end{aligned} \tag{3.12}$$

Pela definição das variáveis, é necessário garantir que se tem $t_i^h = 1$ para todo $h < m$, sempre que $t_i^m = 1$, o que é conseguido através das restrições $t_i^m \leq t_i^{m-1}$, $m = 2, \dots, M$. Assim, cada variável t_i^m é responsável por parte do custo de instalação de equipamento tecnológico especial no nodo i , a parte associada ao m -ésimo módulo. Desta forma, a variável t_i^1 é responsável pela instalação do primeiro módulo e o seu coeficiente incorpora o custo fixo e o custo do módulo, $\phi^1 = K_1 + K_2$; o coeficiente das restantes variáveis t_i^m tem associado apenas o custo do m -ésimo módulo a ser instalado, $\phi^m - \phi^{m-1} = K_2$. O modelo linear com estas variáveis-módulo *incrementais* é apresentada na Figura 3.3.

A função objectivo (3.13), (considerando por comodidade de escrita, $\phi^0 \equiv 0$), reflecte as últimas considerações feitas em relação aos custos de grau a atribuir a cada variável t_i^m . As restrições (3.15) expressam a natureza *incremental* das variáveis t_i^m e garantem por transitividade que, se o m -ésimo módulo for instalado, então os primeiros $m - 1$ módulos têm de estar todos instalados. As restrições (3.14) garantem que o m -ésimo módulo instalado permite um aumento de Q ligações no nodo i , para além das $(m - 1) \cdot Q$ ligações permitidas (e efectivamente estabelecidas) pelos primeiros $m - 1$ módulos já instalados.

$$(SUP_I) \min \sum_{(i,j) \in A} c_{ij} x_{ij} + \sum_{i \in V} \sum_{m=1}^M (\phi^m - \phi^{m-1}) \cdot t_i^m \quad (3.13)$$

$$s.a : X(A^-(i)) = 1 \quad i \in V \setminus \{r\} \quad (1.2)$$

$$\{ (i, j) \in A : x_{ij} = 1 \} \text{ é conexo} \quad (1.3)$$

$$X(A^+(i)) = \begin{cases} U_i & i \in V \setminus \{r\} \\ U_r + 1 & i = r \end{cases} \quad (3.2a)$$

$$U_r + 1 \quad i = r \quad (3.2b)$$

$$U_i \leq D - 1 \quad i \in V \quad (3.3)$$

$$U_i \leq (Q - 1) \cdot t_i^1 + Q \sum_{m=2}^M t_i^m \quad i \in V \quad (3.14)$$

$$t_i^m \leq t_i^{m-1} \quad i \in V, 2 \leq m \leq M \quad (3.15)$$

$$x_{ij} \in \{0, 1\} \quad (i, j) \in A \quad (1.4)$$

$$U_i \in \mathbb{N}_0 \quad i \in V \quad (3.4)$$

$$t_i^m \in \{0, 1\} \quad i \in V, 1 \leq m \leq M \quad (3.16)$$

Figura 3.3: Modelo Linear com variáveis-módulo discretizadas para o (ASupCG): Modelo Incremental.

Tendo em conta o coeficiente estritamente positivo das variáveis t_i^m na função objectivo, em qualquer solução óptima deste modelo, a instalação do m -ésimo módulo no nodo i pressupõe que as ligações dos $m - 1$ módulos já instalados estão totalmente utilizadas *i.e.*, $U_i + 1 > (m - 1) \cdot Q$. Sendo assim, não é necessário incluir restrições que garantam o limite inferior das variáveis U_i expressos em (3.12). Como para o modelo anterior e pela mesma razão, também as restrições (3.14) podem ser "apertadas", tendo em conta que se o M -ésimo módulo for instalado, a sua capacidade realmente "utilizável" será, $Q - (M \cdot Q - D)$.

As restrições (3.14) podem então ser substituídas pelas restrições:

$$U_i \leq (Q - 1) \cdot t_i^1 + Q \sum_{m=2}^{M-1} t_i^m + (D - (M - 1) \cdot Q) \cdot t_i^M \quad i \in V \quad (3.14')$$

Novamente, a presença destas restrições no modelo (SUP_I), em substituição das restrições (3.14), permite eliminar as restrições (3.3) que se tornam redundantes. Assim sendo, o modelo (SUP_I) pode ser redefinido como:

$$(SUP_I) \min \sum_{(i,j) \in A} c_{ij} x_{ij} + \sum_{i \in V} \sum_{m=1}^M (\phi^m - \phi^{m-1}) t_i^m \quad (3.13)$$

$$s.a : (1.2), (1.3), (1.4), (3.2a), (3.2b), (3.4), (3.14'), (3.15) \text{ e } (3.16)$$

Em ambos os modelos lineares (SUP_{EM}) e (SUP_I) a utilização das variáveis binárias, v_i^m e t_i^m respectivamente, não permite a eliminação das variáveis inteiras, U_i . No modelo apresentado na Secção 3.3, ao aplicar a técnica de discretização directamente nas variáveis U_i é possível eliminá-las do modelo e ao mesmo tempo resolver a não linearidade da função objectivo (3.1).

3.2.3 Comparação de relaxações lineares: Parte I

Para comparar os dois anteriores modelos com variáveis-módulo consideram-se, sem perda de generalidade, aqueles que se obtêm depois de "apertar" as restrições (3.7) e (3.14), respectivamente, e eliminar as restrições (3.3). Em termos das respectivas relaxações lineares os dois modelos (SUP_{EM}) e (SUP_I) são equivalentes.

Proposição 3.2.1. *Se as restrições (1.3) forem modeladas da mesma forma para os modelos (SUP_{EM}) e (SUP_I) então:*

$$V(LSUP_{EM}) = V(LSUP_I)$$

Demonstração. No fundo basta mostrar que qualquer solução admissível de $(LSUP_{EM})$ pode ser transformada numa solução admissível de $(LSUP_I)$ com o mesmo custo, através de uma transformação linear e vice-versa.

1. Seja $(\hat{x}, \hat{U}, \hat{v})$ uma solução admissível para $(LSUP_{EM})$. Considere-se ainda uma solução $(\check{x}, \check{U}, \check{t})$ onde $\check{x} = \hat{x}$, $\check{U} = \hat{U}$ e o valor das variáveis t é obtido da seguinte forma:

$$\check{t}_i^m = \sum_{h=m}^M \hat{v}_i^h \quad \forall i \in V, 1 \leq m \leq M \quad (3.17)$$

A ideia é mostrar que a solução $(\check{x}, \check{U}, \check{t})$ é admissível para $(LSUP_I)$.

Para qualquer nodo $i \in V$, as parcelas do segundo membro nas restrições (3.7') do modelo de escolha múltipla, podem ser decompostas da seguinte forma¹:

$$\begin{aligned} \hat{U}_i &\leq (Q-1) \cdot \hat{v}_i^1 + ((Q-1) \cdot \hat{v}_i^2 + Q\hat{v}_i^2) + ((Q-1) \cdot \hat{v}_i^3 + Q\hat{v}_i^3 + Q\hat{v}_i^3) + \\ &\dots + ((Q-1) \cdot \hat{v}_i^{M-1} + \underbrace{Q\hat{v}_i^{M-1} + \dots + Q\hat{v}_i^{M-1}}_{M-2 \text{ vezes}}) + \\ &\quad + ((Q-1) \cdot \hat{v}_i^M + \underbrace{Q\hat{v}_i^M + \dots + Q\hat{v}_i^M}_{M-2 \text{ vezes}}) + (D - (M-1) \cdot Q) \cdot \hat{v}_i^M = \\ &= (Q-1) \cdot \check{t}_i^1 + Q\check{t}_i^2 + Q\check{t}_i^3 + \dots + Q\check{t}_i^{M-1} + (D - (M-1) \cdot Q) \cdot \check{t}_i^M \end{aligned}$$

ou seja, as restrições (3.14') são verificadas pela solução $(\check{x}, \check{U}, \check{t})$.

¹Note-se que $D = (M-1) \cdot Q + (D - (M-1) \cdot Q)$.

Somando a ambos os membros da restrição de não negatividade para a variável v_i^{m-1} , a soma $\sum_{h=m}^M v_i^h$, obtém-se:

$$\sum_{h=m}^M \hat{v}_i^h \leq \sum_{h=m-1}^M \hat{v}_i^h \Leftrightarrow \check{t}_i^m \leq \check{t}_i^{m-1} \quad \forall i \in V, 2 \leq m \leq M$$

Finalmente, usando as equações (3.17), as restrições (3.8) e as restrições de domínio das variáveis v_i^m , conclui-se que $0 \leq \check{t}_i^m \leq 1, \forall i \in V, m = 1, \dots, M$. Portanto, a solução $(\check{x}, \check{U}, \check{t})$ é admissível para $(LSUP_I)$.

O custo desta solução é obtido pela soma do custo associado às variáveis x com:

$$\begin{aligned} \sum_{i \in V} \sum_{m=1}^M (\phi^m - \phi^{m-1}) \cdot \check{t}_i^m &= \sum_{i \in V} \sum_{m=1}^M (\phi^m - \phi^{m-1}) \cdot \sum_{h=m}^M \hat{v}_i^h = \\ &= \sum_{i \in V} \sum_{h=1}^M \hat{v}_i^h \cdot \sum_{m=1}^h (\phi^m - \phi^{m-1}) = \sum_{i \in V} \sum_{h=1}^M \phi^h \cdot \hat{v}_i^h \end{aligned}$$

que é o custo da solução $(\hat{x}, \hat{U}, \hat{v})$ ou seja, as duas soluções têm o mesmo custo, provando-se assim, que $V(LSUP_{EM}) \geq V(LSUP_I)$.

2. Considere-se agora uma solução, $(\check{x}, \check{U}, \check{t})$, admissível para $(LSUP_I)$ e uma outra solução $(\hat{x}, \hat{U}, \hat{v})$ em que $\hat{x} = \check{x}, \hat{U} = \check{U}$ e o valor das variáveis v é obtido da seguinte forma, para $\forall i \in V$:

$$\hat{v}_i^m = \check{t}_i^m - \check{t}_i^{m+1} \quad 1 \leq m \leq M - 1 \quad (3.18a)$$

$$\hat{v}_i^M = \check{t}_i^M \quad (3.18b)$$

Para provar que $(\hat{x}, \hat{U}, \hat{v})$ é uma solução admissível para $(LSUP_{EM})$ é necessário reorganizar as restrições (3.14') do modelo incremental da seguinte forma, para $\forall i \in V$:

$$\begin{aligned}
\check{U}_i &\leq (Q - 1) \cdot \check{t}_i^1 + \sum_{m=2}^{M-1} ((mQ - 1) - ((m - 1) \cdot Q - 1)) \cdot \check{t}_i^m + \\
&\quad + \left((D - 1) - ((M - 1) \cdot Q - 1) \right) \cdot \check{t}_i^M = \\
&= \sum_{m=1}^{M-1} (mQ - 1) \cdot \check{t}_i^m - \sum_{m=2}^M ((m - 1) \cdot Q - 1) \cdot \check{t}_i^m + (D - 1) \cdot \check{t}_i^M = \\
&= \sum_{m=1}^{M-1} (mQ - 1) \cdot \check{t}_i^m - \sum_{m=1}^{M-1} (mQ - 1) \cdot \check{t}_i^{m+1} + (D - 1) \cdot \check{t}_i^M = \\
&= \sum_{m=1}^{M-1} (mQ - 1) \cdot \hat{v}_i^m + (D - 1) \cdot \hat{v}_i^M
\end{aligned}$$

ou seja, as restrições (3.7') são verificadas pela solução $(\hat{x}, \hat{U}, \hat{v})$.

Além disso, esta solução também verifica as restrições (3.8), visto que $\check{t}_i^1 \leq 1$, para qualquer $i \in V$, implica que:

$$\begin{aligned}
1 &\geq \check{t}_i^1 + \sum_{m=2}^M \check{t}_i^m - \sum_{m=2}^M \check{t}_i^m = \sum_{m=1}^M \check{t}_i^m - \sum_{m=1}^{M-1} \check{t}_i^{m+1} = \\
&= \sum_{m=1}^{M-1} (\check{t}_i^m - \check{t}_i^{m+1}) + \check{t}_i^M = \sum_{m=1}^M \hat{v}_i^m
\end{aligned}$$

Finalmente, pelas restrições de domínio das variáveis t_i^m , pelas restrições (3.15) e usando as igualdades (3.18a) e (3.18b), conclui-se que $0 \leq \hat{v}_i^m \leq 1, \forall i \in V, m = 1, \dots, M$ e portanto a solução $(\hat{x}, \hat{U}, \hat{v})$ é admissível para $(LSUP_{EM})$.

O seu custo, além da parcela associada às variáveis x , é:

$$\begin{aligned}
\sum_{i \in V} \sum_{m=1}^M \phi^m \cdot \hat{v}_i^m &= \sum_{i \in V} \left(\sum_{m=1}^{M-1} \phi^m \cdot (\check{t}_i^m - \check{t}_i^{m+1}) + \phi^M \check{t}_i^M \right) = \\
&= \sum_{i \in V} \left(\sum_{m=1}^M \phi^m \check{t}_i^m - \sum_{m=1}^{M-1} \phi^m \check{t}_i^{m+1} \right) \stackrel{\phi^0=0}{=} \\
&= \sum_{i \in V} \left(\sum_{m=1}^M \phi^m \check{t}_i^m - \sum_{m=0}^{M-1} \phi^m \check{t}_i^{m+1} \right) = \\
&= \sum_{i \in V} \sum_{m=1}^M (\phi^m - \phi^{m-1}) \cdot \check{t}_i^m
\end{aligned}$$

que é o custo da solução $(\check{x}, \check{U}, \check{t})$, ou seja, as duas soluções têm o mesmo custo, concluindo-se assim, que $V(LSUP_I) \geq V(LSUP_{EM})$.

Assim se conclui que existem transformações lineares, (3.17), (3.18a) e (3.18b), que a partir de qualquer solução admissível de um dos modelos permitem construir uma solução admissível para o outro modelo e com o mesmo custo, ou seja:

$$V(LSUP_I) = V(LSUP_{EM})$$

□

3.3 Modelo com variáveis-nodo discretizadas

O modelo (SUP_{NL}) apresentado na Secção 3.1 apresenta uma função linear por segmentos nas variáveis U_i . Como tal, é possível aplicar a este modelo a técnica de discretização apresentada na Secção 1.3.1, criando um novo conjunto de variáveis binárias, u_i^d com o seguinte significado:

$$u_i^d = \begin{cases} 1 & \text{se na solução o nodo } i \text{ tem grau } d + 1 \\ 0 & \text{caso contrário} \end{cases} \quad \forall i \in V, d = 1, \dots, D - 1$$

com $\sum_{d=1}^{D-1} u_i^d \leq 1$ para garantir a consistência da substituição.

Estas novas variáveis estão relacionadas com as variáveis inteiras anteriores U_i da seguinte forma:

$$U_i = \sum_{d=1}^{D-1} d \cdot u_i^d, \quad \forall i \in V \quad (3.19)$$

Se o nodo i estiver na solução com grau superior a 1 ($U_i > 0$) então exactamente uma das variáveis u_i^d terá valor igual a um. Essa variável será aquela que estiver associada ao índice d correspondente ao valor da variável inteira $U_i = d$. Caso contrário, o nodo é uma folha ($U_i = 0$) e então $u_i^d = 0$, $d = 1, \dots, D - 1$.

Para cada nodo i , o termo não linear da função objectivo (3.1), $\Phi(U_i + 1)$, tem domínio $\{0, 1, \dots, D - 1\}$. Este domínio pode ser particionado nos subconjuntos $\{0\} \cup \{1, \dots, Q - 1\} \cup \{Q, \dots, 2Q - 1\} \cup \dots \cup \{(M - 1)Q, \dots, D - 1\}$ para os quais a função toma os valores $0, \phi^1, \phi^2, \dots, \phi^M$, respectivamente. Tomando como referência o Exemplo 1.3 da Secção 1.3.1, a função $\Phi(U_i + 1)$ ficará então escrita como:

$$\Phi(U_i + 1) = \phi^1 \cdot \sum_{d=1}^{Q-1} u_i^d + \sum_{m=2}^{M-1} \phi^m \cdot \sum_{d=(m-1)Q}^{mQ-1} u_i^d + \phi^M \cdot \sum_{d=(M-1)Q}^{D-1} u_i^d$$

Definindo os coeficientes $\lambda^d = \Phi(d+1)$ para $d = 1, \dots, D-1$, a função linear por segmentos em U_i dá origem, através da técnica de discretização, à função linear nas variáveis u_i^d :

$$\Phi(U_i + 1) = \sum_{d=1}^{D-1} \lambda^d \cdot u_i^d$$

Finalmente, utilizando (3.19) para substituir as variáveis U_i no modelo não linear, obtém-se a formulação linear *discretizada* apresentada na Figura 3.4.

$$(SUP_D) \min \sum_{(i,j) \in A} c_{ij} x_{ij} + \sum_{i \in V} \sum_{d=1}^{D-1} \lambda^d u_i^d \quad (3.20)$$

$$s.a.: \quad X(A^-(i)) = 1 \quad i \in V \setminus \{r\} \quad (1.2)$$

$$\{ (i, j) \in A : x_{ij} = 1 \} \text{ é conexo} \quad (1.3)$$

$$X(A^+(i)) = \begin{cases} \sum_{d=1}^{D-1} d \cdot u_i^d & i \in V \setminus \{r\} \\ \sum_{d=1}^{D-1} d \cdot u_r^d + 1 & i = r \end{cases} \quad (3.21a)$$

$$\sum_{d=1}^{D-1} d \cdot u_r^d + 1 \quad i = r \quad (3.21b)$$

$$\sum_{d=1}^{D-1} u_i^d \leq 1 \quad i \in V \quad (3.22)$$

$$x_{ij} \in \{0, 1\} \quad (i, j) \in A \quad (1.4)$$

$$u_i^d \in \{0, 1\} \quad i \in V, 1 \leq d \leq D-1 \quad (3.23)$$

Figura 3.4: Modelo Linear com variáveis-grau discretizadas para o (ASupCG).

Note-se que, ao substituir as variáveis U_i pelas variáveis u_i^d , as restrições (3.3) do modelo não linear tornam-se redundantes.

As novas restrições de *grau externo* (3.21a) (3.21b) em conjunto com as restrições de consistência (3.22), garantem que o número de arcos divergentes do nodo i , incluídos na solução, é igual ao índice d ($d + 1$ no caso do nodo raiz) da única variável u_i^d com valor igual a 1.

Neste modelo é possível ainda eliminar certas variáveis u_i^d . Isto acontece sempre que, numa instância do (ASupCG), o número de arestas incidentes num dado nodo seja inferior (ou igual) ao grau máximo estabelecido:

$$\begin{aligned} u_i^d &= 0 & d &= |A^+(i)|, \dots, D - 1 \\ u_i^{|A^+(i)|} &= 0 & i &\neq r \text{ e } \nexists (r, i) \in A \text{ ou } i = r \end{aligned}$$

3.3.1 Comparação de relaxações lineares: Parte II

O modelo discretizado (SUP_D) é equivalente em termos das respectivas relaxações lineares, aos modelos (SUP_{EM}) e (SUP_I), apresentados na Secção 3.2, admitindo que as restrições genéricas (1.3) são modeladas da mesma forma. A prova será feita comparando os modelos (SUP_D) e (SUP_{EM}).

Proposição 3.3.1. *Se as restrições (1.3) forem modeladas da mesma forma para os modelos (SUP_{EM}) e (SUP_D) então:*

$$V(LSUP_{EM}) = V(LSUP_D)$$

Demonstração. A demonstração consiste em duas partes: na primeira parte mostra-se que qualquer solução admissível do modelo ($LSUP_D$) pode ser transformada linearmente numa solução admissível do modelo ($LSUP_{EM}$) com o mesmo custo.

Na segunda parte, mostra-se que a partir de qualquer solução óptima do modelo ($LSUP_{EM}$) se pode obter, através de uma transformação linear, uma solução admissível para o modelo ($LSUP_D$) com o mesmo custo.

1. Seja (\hat{x}, \hat{u}) uma solução admissível para o modelo ($LSUP_D$). Considere-se ainda uma solução $(\check{x}, \check{U}, \check{v})$, em que $\check{x} = \hat{x}$ e o valor das variáveis U_i é obtido a partir do valores das variáveis u_i^d utilizando as equações de discretização já apresentadas, (3.19). O valor das variáveis v_i^m é obtido através do seguinte sistema de equações lineares:

$$\check{v}_i^1 = \sum_{d=1}^{Q-1} \hat{u}_i^d \quad (3.24a)$$

$$\check{v}_i^m = \sum_{d=(m-1)Q}^{mQ-1} \hat{u}_i^d \quad 2 \leq m \leq M-1 \quad (3.24b)$$

$$\check{v}_i^M = \sum_{d=(M-1)Q}^{D-1} \hat{u}_i^d \quad (3.24c)$$

A solução $(\check{x}, \check{U}, \check{v})$ assim obtida, satisfaz as restrições (3.2a) e (3.2b) devido às igualdades (3.19). Além disso, a partir destas mesmas igualdades obtém-se:

$$\begin{aligned} \check{U}_i &= \sum_{d=1}^{D-1} d \cdot \hat{u}_i^d = \sum_{d=1}^{Q-1} d \cdot \hat{u}_i^d + \sum_{m=2}^{M-1} \sum_{d=(m-1)Q}^{mQ-1} d \cdot \hat{u}_i^d + \sum_{d=(M-1)Q}^{D-1} d \cdot \hat{u}_i^d \leq \\ &\leq \sum_{d=1}^{Q-1} (Q-1) \cdot \hat{u}_i^d + \sum_{m=2}^{M-1} \sum_{d=(m-1)Q}^{mQ-1} (mQ-1) \cdot \hat{u}_i^d \\ &\quad + \sum_{d=(M-1)Q}^{D-1} (D-1) \cdot \hat{u}_i^d = \\ &= (Q-1) \cdot \check{v}_i^1 + \sum_{m=2}^{M-1} (mQ-1) \cdot \check{v}_i^m + (D-1) \cdot \check{v}_i^M \end{aligned}$$

ou seja, as restrições (3.7') são verificadas.

Por outro lado, para cada $i \in V$, utilizando o sistema (3.24a) – (3.24c), obtêm-se as restrições (3.8):

$$\sum_{m=1}^M \check{v}_i^m = \sum_{d=1}^{Q-1} \hat{u}_i^d + \sum_{m=2}^{M-1} \sum_{d=(m-1)Q}^{mQ-1} \hat{u}_i^d + \sum_{d=(M-1)Q}^{D-1} \hat{u}_i^d = \sum_{d=1}^D \hat{u}_i^d \leq 1$$

Finalmente, o domínio das variáveis \hat{u}_i^d e as restrições (3.22) permitem concluir que $0 \leq \check{v}_i^m \leq 1, \check{U}_i \geq 0, \forall i, m$. Conclui-se assim que a solução $(\check{x}, \check{U}, \check{v})$ é admissível para o modelo $(LSUP_{EM})$. À parte do custo associado às variáveis x e atendendo à definição da função $\Phi(d)$ (ver Capítulo 2), o custo desta solução é:

$$\begin{aligned} \sum_{i \in V} \sum_{m=1}^M \phi^m \check{v}_i^m &= \sum_{i \in V} \left(\sum_{d=1}^{Q-1} \phi^1 \hat{u}_i^d + \sum_{m=2}^{M-1} \sum_{d=(m-1)Q}^{mQ-1} \phi^m \hat{u}_i^d + \sum_{d=(M-1)Q}^{D-1} \phi^M \hat{u}_i^d \right) = \\ &= \sum_{i \in V} \sum_{d=1}^{D-1} \Phi(d+1) \cdot \hat{u}_i^d = \sum_{i \in V} \sum_{d=1}^{D-1} \lambda^d \cdot \hat{u}_i^d \end{aligned}$$

Ou seja, as duas soluções têm o mesmo custo e prova-se assim que, $V(LSUP_D) \geq V(LSUP_{EM})$.

2. Para estabelecer $V(LSUP_D) \leq V(LSUP_{EM})$ começa-se por considerar uma solução óptima $(\bar{x}, \bar{U}, \bar{v})$ para $(LSUP_{EM})$. Em qualquer solução óptima deste modelo, as restrições (3.7') são satisfeitas como igualdades, visto que, para cada nodo $i \in V$, as variáveis do conjunto $\{v_i^1, v_i^2, \dots, v_i^M\}$ estão envolvidas apenas na restrição (3.7') e na restrição (3.8), para o nodo i . De facto, suponha-se por absurdo que existe uma solução óptima em que, para um dado nodo $i \in V$ a restrição (3.7') é verificada como desigualdade estrita.

A partir desta solução, é possível diminuir o valor de uma ou mais dessas variáveis $v_i^1, v_i^2, \dots, v_i^M$, mantendo-as não negativas, até que o segundo membro da restrição (3.7') seja igual ao primeiro membro, U_i . Construída desta forma, esta nova solução não viola a restrição (3.8) e além disso, tendo em conta o coeficiente positivo das variáveis v_i^m na função objectivo, tem um custo inferior ao da solução óptima, o que é absurdo.

A partir da solução óptima, $(\bar{x}, \bar{U}, \bar{v})$, constrói-se uma solução (\hat{x}, \hat{u}) em que o valor das variáveis u_i^d é obtido da seguinte forma (o valor das variáveis x é o mesmo, $\hat{x} \equiv \bar{x}$), para cada i e d :

$$\hat{u}_i^d = \begin{cases} \bar{v}_i^m & , \text{ se } d = mQ - 1 \quad \forall m = 1, \dots, M - 1 \\ \bar{v}_i^M & , \text{ se } d = D - 1 \\ 0 & , \text{ caso contrário} \end{cases} \quad (3.25)$$

ou seja, para um dado número de módulos instalados no nodo i , m , apenas será não nula a variável u_i^d , tal que $d + 1$ corresponde ao número máximo de ligações permitidas por esse número de módulos instalados², mQ ; todas as restantes variáveis u_i^d serão nulas.

Pelo domínio das variáveis v é imediato observar que $0 \leq \hat{u}_i^d \leq 1$, $\forall i \in V, d = 1, \dots, D - 1$. Por outro lado, as restrições (3.22) são verificadas para qualquer nodo $i \in V$:

$$\sum_{d=1}^{D-1} \hat{u}_i^d = \sum_{m=1}^{M-1} \hat{u}_i^{mQ-1} + \hat{u}_i^{D-1} = \sum_{m=1}^{M-1} \bar{v}_i^m + \bar{v}_i^M \leq 1$$

²No caso do número de módulos instalados ser M , a única variável não nula é u_i^{D-1} .

Para qualquer nodo $i \in V$, a partir das restrições (3.7') satisfeitas como igualdade e usando (3.25), obtêm-se as igualdades (3.19):

$$\begin{aligned}
 \bar{U}_i &= \sum_{m=1}^M (mQ - 1) \cdot \bar{v}_i^m + (D - 1) \cdot \bar{v}_i^M = \sum_{m=1}^{M-1} (mQ - 1) \cdot \hat{u}_i^{mQ-1} + \\
 &+ (D - 1) \cdot \hat{u}_i^{D-1} + \underbrace{\sum_{d=1}^{Q-2} d \cdot \hat{u}_i^d + \sum_{m=2}^{M-1} \sum_{d=(m-1)Q}^{mQ-2} d \cdot \hat{u}_i^d}_{=0} + \\
 &+ \underbrace{\sum_{d=(M-1)Q}^{D-2} d \cdot \hat{u}_i^d}_{=0} = \sum_{m=1}^{M-1} \sum_{d=(m-1)Q+1}^{mQ-1} d \cdot \hat{u}_i^d + \sum_{d=(M-1)Q}^{D-1} d \cdot \hat{u}_i^d = \\
 &= \sum_{d=1}^{D-1} d \cdot \hat{u}_i^d
 \end{aligned}$$

Este facto, em conjunto com as restrições (3.2a) e (3.2b) do modelo ($LSUP_{EM}$), permite concluir que a solução (\hat{x}, \hat{u}) verifica as restrições (3.21a) e (3.21b) do modelo ($LSUP_D$) e portanto, a solução (\hat{x}, \hat{u}) é admissível para este modelo.

Finalmente, a parte do custo desta solução, associada às variáveis u é dada por ³:

$$\begin{aligned}
 \sum_{i \in V} \sum_{d=1}^{D-1} \lambda^d \hat{u}_i^d &\stackrel{(3.25)}{=} \sum_{i \in V} \left(\sum_{m=1}^{M-1} \lambda^{mQ-1} \cdot \hat{u}_i^{mQ-1} + \lambda^{D-1} \cdot \hat{u}_i^{D-1} \right) = \\
 &= \sum_{i \in V} \left(\sum_{m=1}^{M-1} \Phi(mQ) \cdot \bar{v}_i^m + \Phi(D) \cdot \bar{v}_i^M \right) = \\
 &= \sum_{i \in V} \left(\sum_{m=1}^{M-1} \phi^m \cdot \bar{v}_i^m + \phi^M \cdot \bar{v}_i^M \right) = \sum_{i \in V} \sum_{m=1}^M \phi^m \bar{v}_i^m
 \end{aligned}$$

³Atente-se à definição da função $\Phi(\cdot)$ no Capítulo 2 e λ^d na Secção 3.3.

ou seja, as duas soluções $(\bar{x}, \bar{U}, \bar{v})$ e (\hat{x}, \hat{u}) têm o mesmo custo, concluindo-se assim que:

$$V(LSUP_{EM}) \geq V(LSUP_D)$$

□

3.3.2 Resultados Computacionais: Modelo (SUP_D)

Nesta secção avalia-se a qualidade dos limites fornecidos pela relaxação linear do modelo discretizado (SUP_D) aplicado a um conjunto de instâncias de teste com 25 e 50 nodos. No Apêndice B é descrita a forma como as instâncias foram geradas. Todos os resultados apresentados foram obtidos num computador INTEL CORE 2 - 2.4 GHz com 3.327 GB de memória RAM, utilizando o software CPLEX 11.0/Concert Technology 2.5 da ILOG. Nas tabelas seguintes a primeira coluna, "R", identifica o tipo de Rede: "W" para redes *Wireless* e "C" para redes *Cabo*, de acordo com o tipo de custos das ligações entre os nodos (ver Apêndice B). As colunas $|E|$, (K_1, K_2) e D indicam respectivamente, a densidade da instância em termos de número de arestas, o par de custos tecnológicos que define a função de custos nos nodos e o grau máximo em cada nodo. De seguida as tabelas encontram-se divididas em duas partes, a primeira referente aos desvios relativos do valor óptimo do modelo linear $(LSUP_D)$ em relação ao valor óptimo do problema inteiro e a segunda referente aos tempos de CPU para resolver o modelo linear $(LSUP_D)$. O desvio relativo é calculado como habitualmente:

$$desvio = \frac{V(P) - V(LP)}{V(P)}$$

para um modelo genérico P .

Um valor de desvio relativo igual a 0^* significa que o valor óptimo da relaxação linear do modelo ($LSUP_D$) é igual ao valor óptimo do problema inteiro. Quanto à escolha do modelo para obter a solução óptima inteira do problema apresenta-se no Apêndice C um estudo comparativo de alguns modelos (os mais relevantes) apresentados ao longo deste capítulo. Os resultados obtidos com o modelo escolhido serão analisados após a sua apresentação.

As colunas SC , SC^* e MC indicam os resultados obtidos utilizando os diferentes sistemas de fluxos para garantir a conexidade da solução, respectivamente, com comodidade única, com comodidade única fortalecido com a adição das desigualdades (1.10) (ver Secção 1.1.1) e com múltiplas comodidades. Por clareza de apresentação, não se incluem as colunas referentes aos tempos de CPU para o modelo ($LSUP_D$) com sistema de fluxos SC , com ou sem desigualdades (1.10) adicionadas, visto estes serem inferiores a 1 segundo (*cf.* Apêndice D).

Comparando sistemas de fluxos.

Comparando os resultados obtidos com os diferentes sistemas de fluxos, observa-se que os desvios do modelo utilizando o sistema SC melhoram consideravelmente quando se utilizam as desigualdades (1.10). Mesmo assim, a utilização do sistema MC ainda consegue reduzir os desvios obtidos com a utilização do sistema SC^* . Note-se ainda que, com o sistema de fluxos MC , o modelo ($LSUP_D$) conseguiu reduzir o desvio a zero, para uma instância *Wireless* com 50 nodos e 150 arestas (para $D = 3$). Em termos de tempo de CPU, o sistema MC que envolve mais variáveis e restrições do que o sistema SC (e SC^*) regista valores mais elevados no entanto são consideravelmente baixos (inferiores a 30 segundos).

R	$ E $	(K_1, K_2)	D	$(LSUP_D)$			CPU
				SC	desvio SC^*	MC	
75	(100, 10)	3	4,2	3,9	3,7	0	
		4	4,0	3,7	3,4	0	
		5	6,4	5,9	5,6	0	
		6	19,9	19,5	19,2	0	
	(100, 50)	3	4,2	4,0	3,8	0	
		4	1,8	1,6	1,4	0	
		5	5,6	5,3	5,1	0	
		6	16,2	15,9	15,7	0	
150	(100, 10)	3	3,8	3,8	3,8	1	
		4	3,5	3,5	3,5	0	
		5	4,9	4,9	4,9	0	
		6	9,0	9,0	9,0	0	
	(100, 50)	3	3,9	3,9	3,9	1	
		4	1,5	1,5	1,5	0	
		5	4,6	4,6	4,6	0	
		6	8,7	8,7	8,7	0	
75	(100, 10)	3	7,5	6,2	5,7	0	
		4	12,9	11,1	10,6	0	
		5	21,6	19,9	19,4	0	
		6	26,2	24,4	23,9	0	
	(100, 50)	3	6,9	5,8	5,5	0	
		4	9,7	8,3	7,9	0	
		5	18,5	17,1	16,7	0	
		6	24,0	22,4	22,0	0	
150	(100, 10)	3	8,4	5,9	5,3	0	
		4	12,4	9,6	9,3	0	
		5	16,9	14,3	13,9	0	
		6	23,1	20,4	20,0	0	
	(100, 50)	3	7,7	5,6	5,1	0	
		4	9,1	6,9	6,7	0	
		5	14,3	12,2	11,9	0	
		6	20,4	18,1	17,9	0	

Tabela 3.1: Resultados para instâncias com 25 nodos: Modelo $(LSUP_D)$.*Densidade e Tipo de rede.*

Em relação ao tipo de rede, os desvios são consideravelmente maiores para as redes *Cabo* do que para as redes *Wireless*. Quanto à densidade da rede, os comportamentos são distintos para as instâncias com 25 e 50 nodos. Nas primeiras os desvios tendem a diminuir quando a rede se torna mais densa,

quer para redes *Cabo*, quer para redes *Wireless* embora nestas últimas a diminuição seja maior. Nas instâncias com 50 nodos sucede o contrário: os desvios tendem a aumentar quando a rede se torna mais densa, quer para redes *Cabo*, quer para redes *Wireless*, sendo maior o aumento nestas últimas.

R	$ E $	(K_1, K_2)	D	$(LSUP_D)$			CPU MC
				SC	desvio SC^*	MC	
W	150	(100, 10)	3	0,7	0,1	0*	9
			4	1,3	0,5	0,3	1
			5	2,8	1,8	1,6	1
			6	7,4	6,2	6,0	1
	(100, 50)	3	0,6	0,1	0*	9	
		4	0,8	0,3	0,2	2	
		5	1,9	1,2	1,1	1	
		6	6,3	5,5	5,4	1	
	300	(100, 10)	3	1,0	0,1	0,1	27
			4	1,9	1,1	1,0	2
			5	3,5	2,4	2,3	2
			6	7,9	6,7	4,0	1
(100, 50)	3	0,7	0,1	0,1	23		
	4	1,2	0,7	0,6	2		
	5	2,3	1,6	1,5	2		
	6	6,6	5,8	4,0	1		
C	150	(100, 10)	3	8,0	2,8	2,7	1
			4	12,8	7,6	7,6	1
			5	17,8	13,2	12,9	1
			6	21,0	16,4	16,2	1
	(100, 50)	3	7,0	2,4	2,4	1	
		4	10,0	5,7	5,7	1	
		5	16,1	12,1	11,9	1	
		6	19,9	15,9	15,8	1	
	300	(100, 10)	3	11,6	5,8	5,7	2
			4	19,2	12,3	12,2	2
			5	25,5	18,3	18,1	2
			6	31,7	24,4	24,2	2
(100, 50)	3	9,7	4,8	4,8	2		
	4	13,6	8,1	8,0	2		
	5	22,0	16,2	16,0	2		
	6	29,0	23,0	22,8	2		

Tabela 3.2: Resultados para instâncias com 50 nodos: Modelo $(LSUP_D)$.

Efeito dos parâmetros.

O efeito do parâmetro D nos desvios não parece ser o mesmo para os diferentes cenários. Aparentemente os desvios aumentam com o parâmetro D , havendo no entanto exceções: nas redes *Wireless* com 25 nodos decresce para $D = 4$. Para as redes *Wireless* regista-se um grande aumento nos desvios quando o grau máximo é 6.

Quanto ao par de custos tecnológicos (K_1, K_2) os desvios são mais baixos quando os custos nos nodos são mais elevados, $(K_1, K_2) = (100, 50)$. As diferenças são maiores nas redes *Cabo* do que nas redes *Wireless*.

Como foi referido no início da análise de resultados, a utilização das desigualdades (1.10) em conjunto com o sistema de fluxos SC , revelou-se útil, em termos da qualidade dos limites inferiores. Sendo assim, nos resultados das secções seguintes, apenas serão apresentados resultados para modelos que utilizam um sistema de fluxos SC^* ou MC .

3.4 Modelo com variáveis-arco discretizadas

Os resultados computacionais da secção anterior mostram que, a qualidade dos limites inferiores fornecidos pela relaxação linear do modelo (SUP_D) é em geral fraca como, principalmente para as instâncias *Cabo*. Uma forma de os melhorar consiste em adicionar desigualdades válidas ao modelo (SUP_D) , que podem ser escritas em termos das variáveis já existentes no modelo $(x_{ij}$ e $u_i^d)$ ou, de uma forma alternativa, utilizando novas variáveis através dum modelo estendido obtido a partir do modelo (SUP_D) . Nesta secção discute-se a segunda alternativa e considera-se um modelo estendido onde as variáveis-arco, x_{ij} , são desagregadas em novas variáveis- -arco discretizadas, x_{ij}^d , de modo a

introduzir desigualdades válidas do tipo $x_{ij}^d \leq u_i^d$. Estas desigualdades são semelhantes na forma, a desigualdades utilizadas para reduzir a diferença, em termos de relaxação linear, entre modelos de fluxos de múltiplas comodidades e modelos de fluxos de comodidade única, reforçando estes últimos (ver [51]). Desigualdades semelhantes a estas, mas onde as variáveis têm uma interpretação diferente, foram também utilizadas para reforçar a relaxação linear de formulações básicas para um problema de *Network Loading* [23] ou para um problema de *Network Flow* com uma função de custos linear por segmentos [14]. As novas variáveis-arco discretizadas são definidas da seguinte forma:

$$x_{ij}^d = \begin{cases} 1 & , \text{ se o arco } (i, j) \text{ está na solução e o grau do nodo } i \text{ é } d + 1 \\ 0 & , \text{ caso contrário} \end{cases} \quad (i, j) \in A; \quad d = 1, \dots, D - 1$$

Como o nodo raiz tem apenas arcos divergentes é necessário definir ainda as variáveis x_{rj}^0 para salvaguardar o caso em que o grau do nodo raiz é 1:

$$x_{rj}^0 = \begin{cases} 1 & , \text{ se o arco } (r, j) \text{ está na solução e o grau do nodo } r \text{ é } 1 \\ 0 & , \text{ caso contrário} \end{cases} \quad (r, j) \in A$$

A introdução das novas desigualdades válidas referidas anteriormente, só tem o efeito pretendido se conjuntamente se introduzirem restrições de ligação entre as novas variáveis x_{ij}^d e as variáveis x_{ij} e u_i^d . Estas restrições de ligação⁴ permitem também dar um significado mais preciso às variáveis x_{ij}^d :

⁴ $X^d(A^+(i)) = \sum_{(i,j) \in A^+(i)} x_{ij}^d$.

$$x_{ij} = \sum_{d=1}^{D-1} x_{ij}^d \quad (i, j) \in A, i \neq r \quad (3.26a)$$

$$x_{rj} = \sum_{d=0}^{D-1} x_{rj}^d \quad (r, j) \in A \quad (3.26b)$$

$$X^d(A^+(i)) = d \cdot u_i^d \quad i \in V \setminus \{r\}; d = 1, \dots, D-1 \quad (3.27a)$$

$$X^d(A^+(r)) = (d+1) \cdot u_r^d \quad d = 1, \dots, D-1 \quad (3.27b)$$

$$X^0(A^+(r)) = 1 - \sum_{d=1}^{D-1} u_r^d \quad (3.27c)$$

Figura 3.5: Sistema linear de definição das variáveis x_{ij}^d para o (ASupCG).

As igualdades (3.26a) e (3.26b) fazem a ligação entre as variáveis-arco e as variáveis-arco discretizadas e garantem que se um arco (i, j) estiver presente na solução, então (exactamente) uma variável x_{ij}^d , $d = 1, \dots, D-1$ ($d = 0, \dots, D-1$ no caso de $i = r$) terá valor igual a 1, ou seja, o arco (i, j) está na solução, qualquer que seja o grau do nodo i (desde que o nodo não seja folha no caso de $i \neq r$).

As igualdades (3.27a) - (3.27c) fazem a ligação entre as variáveis-nodo discretizadas e as variáveis-arco discretizadas. Garantem que no caso de um nodo $i \neq r$ ser uma folha na solução, então as variáveis x_{ij}^d serão nulas para todos os arcos divergentes do nodo i e para todo o $d = 1, \dots, D-1$ o que, pelas igualdades (3.26a) implica que nenhum arco divergente de i estará presente na solução. Por outro lado, se o nodo i tiver grau $g(i) = d' + 1$ na solução, então exactamente d' variáveis $x_{ij}^{d'}$ serão iguais a 1; pelas igualdades (3.26a) exactamente d' arcos divergentes no nodo i , estarão presentes na solução. Analogamente, no caso do nodo raiz ser uma folha, pelas igualdades (3.27b)

tem-se $x_{rj}^d = 0$ para todo o arco (r, j) e $d = 1, \dots, D - 1$, o que implica, pelas igualdades (3.26b), que $x_{rj} = x_{rj}^0$ para todo o arco (r, j) . Pela igualdade (3.27c) fica garantido que apenas uma das variáveis x_{rj}^0 será igual a 1 e portanto, apenas um dos arcos divergentes na raiz estará presente na solução. No caso da raiz ter grau $g(r) = d' + 1$, as variáveis x_{rj}^0 , $(r, j) \in A$, serão todas nulas e pelas igualdades (3.27b) exactamente $d' + 1$ variáveis $x_{rj}^{d'}$ serão iguais a 1. Isto implica, pelas igualdades (3.26b), que exactamente $d' + 1$ arcos divergentes na raiz estarão presentes na solução.

As restrições de ligação (3.27a), (3.27b) e (3.27c) podem ser vistas como uma versão desagregada das restrições de ligação entre as variáveis x_{ij} e u_i^d no modelo (SUP_D), respectivamente (3.21a) e (3.21b). De facto, para cada nodo $i \neq r$, somando as restrições de ligação (3.27a) para todo o $d = 1, \dots, D - 1$ e usando a igualdade (3.26a) obtém-se a restrição de ligação (3.21a) para o mesmo nodo i . Analogamente para o nodo raiz, somando as restrições de ligação (3.27b) para todo o $d = 1, \dots, D - 1$, com a restrição de ligação para $d = 0$, (3.27c) e usando (3.26b), obtém-se a restrição de ligação (3.21b).

O modelo discretizado estendido obtém-se ao adicionar o sistema linear da Figura 3.5 ao modelo discretizado (SUP_D). Pelo que foi dito antes, as restrições (3.21a) e (3.21b) podem ser ainda eliminadas do modelo estendido. A adição do sistema linear da Figura 3.5 por si só não chega para fortalecer a relaxação linear do modelo já que as restrições, por enquanto incluídas, apenas permitem definir as novas variáveis.

Por forma a motivar o conjunto de desigualdades válidas mencionadas no início desta secção, considere-se para um dado nodo $i \neq r$ e um valor fixo $d \in \{1, \dots, D-1\}$, o poliedro $\mathcal{P}_{i,d}$, definido por ⁵:

$$\begin{aligned} X^d(A^+(i)) &= d \cdot u_i^d \\ 0 \leq x_{ij}^d &\leq 1 & (i, j) \in A^+(i) \\ 0 \leq u_i^d &\leq 1 \end{aligned}$$

Este poliedro surge como subestrutura em vários modelos de Optimização Combinatória, como por exemplo Modelos de Localização (ver [33]) onde o sinal de igualdade é geralmente substituído por um sinal de \leq . Se forem adicionadas as desigualdades válidas:

$$x_{ij}^d \leq u_i^d \quad (i, j) \in A^+(i)$$

ao poliedro anterior, obtém-se uma descrição linear completa do envolvente convexo definido pelas soluções inteiras do poliedro $\mathcal{P}_{i,d}$ (ver demonstração no Apêndice A).

Este facto permite concluir que não é possível encontrar outras desigualdades válidas envolvendo as variáveis x_{ij}^d e as variáveis u_i^d , $(i, j) \in A^+(i)$, que não sejam dominadas pelas desigualdades válidas apresentadas. Portanto, é natural assumir que se obtém um modelo com uma relaxação linear mais forte se forem introduzidas no modelo estendido, obtido do modelo (SUP_D), as seguintes desigualdades designadas por *desigualdades Arco Desagregadas* ⁶:

$$\boxed{AD(i, j; d): \quad x_{ij}^d \leq u_i^d \quad (i, j) \in A; \quad d = 1, \dots, D-1}$$

⁵Para $i = r$ o poliedro é semelhante mas o segundo membro da igualdade será $(d+1) \cdot u_r^d$.

⁶A sigla AD corresponde às iniciais do nome das desigualdades.

O facto de, para cada $i \in V$, $d = 1, \dots, D - 1$, estas desigualdades definirem facetas do poliedro $\mathcal{P}_{i,d}$, garante a sua validade. No entanto, no modelo estendido, as desigualdades válidas $AD(i, j; 1)$ são redundantes para qualquer $(i, j) \in A$, $i \neq r$ pois, pela não negatividade das variáveis e utilizando a restrição de ligação (3.27a) resulta:

$$x_{ij}^1 \leq X^1(A^+(i)) = u_i^1$$

Como conclusão, as desigualdades válidas desagregadas a introduzir no modelo para fortalecer a respectiva relaxação linear são:

$$AD(i, j; d) \quad , \quad i \in V \setminus \{r\}; \quad d = 2, \dots, D - 1 \quad (3.28a)$$

$$AD(r, j; d) \quad , \quad d = 1, \dots, D - 1 \quad (3.28b)$$

Designa-se por $(P+AD)$ o modelo fortalecido obtido ao adicionar todas as desigualdades válidas (3.28a) e (3.28b) e o sistema de restrições de definição (3.26a) - (3.27c) a um modelo genérico (P) . Este tipo de fortalecimento do modelo discretizado será também aplicado a um outro modelo linear apresentado na Secção 3.5.

Termina-se esta secção com um exemplo da aplicação das desigualdades $AD(i, j; d)$ ao modelo discretizado. Nos resultados apresentados na Secção 3.4.2 comprovar-se-á que, em quase todos os casos, o modelo fortalecido com estas desigualdades permite obter limites inferiores, consideravelmente melhores do que o modelo (SUP_D) .

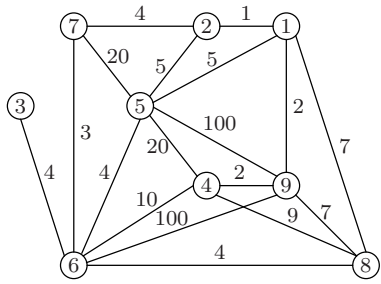
Exemplo 3.1: Modelo (SUP_D) vs modelo $(SUP_D + AD)$ ⁷

Na Figura 3.6(a) apresenta-se uma instância com 9 nodos, 17 arestas, onde o grau máximo é igual a 5 e a dimensão de cada interface é igual a 3. O valor junto a cada aresta representa o respectivo custo de utilização. O custo da matriz de *routing* e de cada interface é, respectivamente, 100 e 10, dando origem ao vector de custos de grau, $\Phi(d)$.

O valor óptimo do modelo $(LSUP_D)$ é igual a 234 (*cf.* Figura 3.6(c)) a que corresponde um desvio de 37.3% (*cf.* Figura 3.6(b)), enquanto o valor óptimo do modelo fortalecido, $(LSUP_D + AD)$ é igual a 308 (ver Figura 3.6(d)) a que corresponde um desvio de 17.4%. Note-se que, embora a solução do modelo $(LSUP_D)$, seja inteira nas variáveis x_{ij} , não corresponde à configuração da solução óptima do modelo (SUP_D) , nem sequer tem o mesmo custo (o valor é 24 no modelo $(LSUP_D)$ e 33 no modelo (SUP_D)).

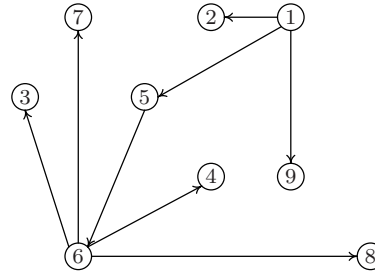
Note-se também que, qualquer solução "desagregada" obtida da solução óptima de $(LSUP_D)$, utilizando as restrições de ligação (3.26a)–(3.27c), resulta numa solução no espaço das variáveis $(x_{ij}, x_{ij}^d, u_i^d)$ que viola pelo menos uma das desigualdades $AD(i, j; d)$. Basta considerar, por exemplo, o arco (6, 8) para o qual as restrições de ligação implicam que $x_{68}^5 = x_{68} = 1$ e $x_{68}^d = 0$ para $d \neq 5$, o que claramente viola a desigualdade $AD(6, 8; 5)$, visto $1 = x_{68}^5 > u_6^5 = \frac{3}{5}$.

⁷Utilizando o sistema de fluxos *MC*.



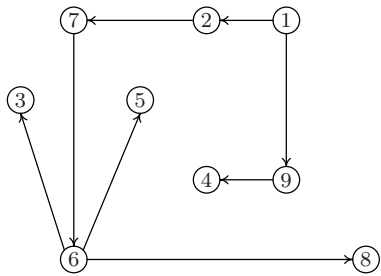
$$\Phi(d) = [110 \ 110 \ 120 \ 120]$$

(a) Grafo original.



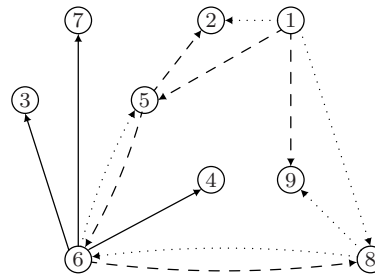
$$u_1^2 = u_5^1 = u_6^4 = 1, \\ u_i^d = 0 \text{ para as restantes}$$

(b) Solução ótima (inteira) com valor 373.



$$u_1^4 = u_2^4 = u_7^4 = u_9^4 = \frac{1}{4}, u_6^4 = \frac{3}{4} \\ u_i^d = 0 \text{ para as restantes}$$

(c) $V(LSUP_D) = 234$.



$$u_1^3 = u_2^8 = \frac{1}{3}, u_5^2 = \frac{2}{3}, u_6^4 = 1, \\ u_i^d = 0 \text{ para as restantes}$$

(d) $V(LSUP_D + AD) = 308$.

Figura 3.6: Comparação dos modelos (SUP_D) e $(SUP_D + AD)$ (arcos a cheio representam $x_{ij} = 1$; arcos a tracejado, $x_{ij} = \frac{2}{3}$; arcos a ponteados, $x_{ij} = \frac{1}{3}$).

3.4.1 Desigualdades *Arco Desagregadas* no espaço das variáveis (x, u)

Muito embora o modelo $(LSUP_D + AD)$ possa dar origem a bons limites inferiores pra o problema, tem uma possível desvantagem: faz aumentar o número de variáveis e restrições no modelo.

Nesta secção retoma-se a primeira alternativa enunciada no início da Secção 3.4 que consiste em encontrar desigualdades válidas para o modelo (SUP_D) escritas em termos das variáveis originais do modelo. Uma forma de o fazer consiste em encontrar desigualdades válidas no espaço das variáveis x_{ij} e u_i^d que sejam implicadas pela relaxação linear do modelo estendido (SUP_{D+AD}).

Assim, para qualquer nodo $i \neq r$ e um qualquer subconjunto de arcos, $H(i) \subseteq A^+(i)$, tal que $1 \leq h \equiv |H(i)| \leq D - 2$, considerem-se as seguintes desigualdades, designadas por *desigualdades Arco* ⁸:

$$A_h(i, H(i)) \quad X(H(i)) \leq \sum_{d=1}^{h-1} d \cdot u_i^d + h \cdot \sum_{d=h}^{D-1} u_i^d, \quad H(i) \subseteq A^+(i), i \neq r$$

É fácil verificar que para subconjuntos de arcos divergentes no nodo $i \neq r$, com dimensão $h \geq D - 1$, as desigualdades válidas $A_h(i, H(i))$ são implicadas pelas restrições de ligação (3.21a) do modelo discretizado.

Para entender melhor o significado destas desigualdades considere-se o caso particular, para um conjunto com dimensão $h = 1$, $H(i) = \{(i, j)\}$:

$$x_{ij} \leq \sum_{d=1}^{D-1} u_i^d$$

⁸A sigla A corresponde à inicial do nome das desigualdades.

Esta desigualdade indica que a presença do arco (i, j) ($i \neq r$) na solução, só é possível se no nodo i for instalado equipamento (já que neste caso o nodo i terá na solução um grau, $g(i) \geq 2$).

Para o caso mais genérico, as desigualdades $A_h(i, H(i))$ implicam que, se todos os arcos de $H(i)$ estiverem presentes na solução, o primeiro membro da desigualdade toma valor h , o que implica que $\sum_{d=1}^{h-1} u_i^d = 0$ e $\sum_{d=h}^{D-1} u_i^d = 1$ já que apenas uma destas variáveis poderá tomar valor 1 (pelas restrições (3.22)). Por outras palavras, o grau do nodo i na solução tem de ser pelo menos igual a $h + 1$ e neste caso a desigualdade $A_h(i, H(i))$ é satisfeita como igualdade.

Pela mesma razão, se apenas $k < h$ arcos de $H(i)$ estiverem presentes na solução então uma das variáveis u_i^d com $d \geq k$ terá um valor igual a 1. Mas neste caso, se for uma das variáveis u_i^d com $d \in \{k + 1, \dots, D - 1\}$, a desigualdade $A_h(i, H(i))$ será satisfeita como desigualdade estrita.

Para o nodo raiz, e para qualquer subconjunto de arcos, $H(r) \subseteq A^+(r)$, tal que $2 \leq h \equiv |H(r)| \leq D - 1$, as *desigualdades Arco* são:

$$A_h(r, H(r)) \quad X(H(r)) \leq 1 + \sum_{d=1}^{h-2} d \cdot u_r^d + (h-1) \cdot \sum_{d=h-1}^{D-1} u_r^d \quad , \quad H(r) \subseteq A^+(r)$$

Também para subconjuntos de arcos com dimensão $h \geq D$, divergentes do nodo raiz, as desigualdades $A_h(r, H(r))$ são dominadas pelas restrições (3.21b). No caso de subconjuntos com dimensão $h = 1$, as desigualdades $A_1(r, \{(r, j)\})$, $x_{rj} \leq 1$, $(r, j) \in A^+(r)$, são dominadas pelas próprias restrições de domínio das variáveis x_{rj} .

Para um dado subconjunto de arcos $H(r) = \{(r, j), (r, k)\}$, a desigualdade $A_2(r, H(r))$ é

$$x_{rj} + x_{rk} \leq 1 + \sum_{d=1}^{D-1} u_r^d$$

que garante que, se dois arcos divergentes da raiz estiverem presentes na solução, então o grau da raiz na solução tem de ser pelo menos igual a 2 (*i.e.*, $\sum_{d=1}^{D-1} u_r^d = 1$).

Para o caso genérico e fazendo uma descrição análoga à dos nodos $i \neq r$, se todos os arcos de $H(r)$ estiverem presentes na solução (o primeiro membro da desigualdade é igual a h), então uma das variáveis u_r^d com $d \geq h - 1$ terá de ser 1, *i.e.*, o grau do nodo r na solução tem de ser pelo menos h e neste caso a desigualdade $A_h(r, H(r))$ é satisfeita como igualdade. Se apenas $k < h$ arcos de $H(r)$ estiverem presentes na solução e $u_r^d = 1$ para algum $d \geq k + 1$, a desigualdade $A_h(r, H(r))$ será satisfeita como desigualdade estrita.

Para mostrar que as desigualdades $A_h(i, H(i))$, $i \in V$, são válidas para o modelo discretizado (SUP_D), mostra-se de seguida que são implicadas pelo modelo (SUP_D+AD). Para tal considere-se em primeiro lugar, para qualquer nodo $i \in V \setminus \{r\}$, os seguintes poliedros: o poliedro \mathcal{P}_i definido por:

$$\begin{aligned} X^d(A^+(i)) &= d \cdot u_i^d & d = 1, \dots, D-1 \\ x_{ij} &= \sum_{d=1}^{D-1} x_{ij}^d & (i, j) \in A^+(i) \\ x_{ij}^d &\leq u_i^d & (i, j) \in A^+(i); d = 1, \dots, D-1 \\ 0 &\leq x_{ij}^d \leq 1 & (i, j) \in A^+(i); d = 1, \dots, D-1 \\ 0 &\leq x_{ij} \leq 1 & (i, j) \in A^+(i) \\ 0 &\leq u_i^d \leq 1 & d = 1, \dots, D-1 \end{aligned}$$

e o poliedro \mathcal{Q}_i definido por ($h = |H(i)|$):

$$\begin{aligned} X(A^+(i)) &= \sum_{d=1}^{D-1} d \cdot u_i^d \\ X(H(i)) &\leq \sum_{d=1}^{h-1} d \cdot u_i^d + h \cdot \sum_{d=h}^{D-1} u_i^d & H(i) \subseteq A^+(i), 1 \leq h \leq D-2 \\ 0 \leq x_{ij} &\leq 1 & (i, j) \in A^+(i) \\ 0 \leq u_i^d &\leq 1 & d = 1, \dots, D-1 \end{aligned}$$

Entre estes dois poliedros verifica-se o seguinte resultado:

Proposição 3.4.1. *Para qualquer nodo $i \in V \setminus \{r\}$*

$$\text{proj}_{x,u}(\mathcal{P}_i) \subseteq \mathcal{Q}_i$$

Demonstração. A prova consiste em mostrar que, para qualquer subconjunto de arcos divergentes em i com dimensão $h \in \{1, \dots, D-2\}$, as desigualdades $A_h(i, H(i))$ são obtidas a partir das desigualdades válidas $AD(i, j; d)$ (e das restrições de ligação, (3.26a) e (3.27a)). Por um lado, as restrições de ligação (3.27a) implicam que:

$$X^d(H(i)) \leq d \cdot u_i^d \quad d = 1, \dots, D-1 \quad (3.29)$$

Por outro lado, somando as desigualdades válidas $AD(i, j; d)$ para todos os arcos de $H(i)$ (fixando o índice d), obtém-se:

$$X^d(H(i)) \leq h \cdot u_i^d \quad d = 1, \dots, D-1 \quad (3.30)$$

Somando agora as inequações (3.29) para todo o $d = 1, \dots, h - 1$ e as inequações (3.30) para todo o $d = h, \dots, D - 1$ e somando o resultado, obtém-se:

$$\sum_{d=1}^{h-1} X^d(H(i)) + \sum_{d=h}^{D-1} X^d(H(i)) \leq \sum_{d=1}^{h-1} d \cdot u_i^d + h \cdot \sum_{d=h}^{D-1} u_i^d$$

Utilizando as restrições de ligação (3.26a), $\forall (i, j) \in H(i)$, no primeiro membro deste último resultado, obtém-se a desigualdade válida $A_h(i, H(i))$.

Pelo que foi dito aquando da apresentação das variáveis x_{ij}^d , o primeiro conjunto de igualdades do poliedro \mathcal{P}_i em conjunto com as restrições de ligação (3.26a), implicam a primeira igualdade do poliedro \mathcal{Q}_i , ficando assim provado que

$$\text{proj}_{x,u}(\mathcal{P}_i) \subseteq \mathcal{Q}_i \quad \square$$

Para o nodo raiz é possível definir poliedros semelhantes. O poliedro \mathcal{P}_r é definido por:

$$\begin{aligned} X^d(A^+(r)) &= (d+1) \cdot u_r^d & d = 1, \dots, D-1 \\ x_{rj} &= \sum_{d=0}^{D-1} x_{rj}^d & (r, j) \in A^+(r) \\ X^0(A^+(r)) &= 1 - \sum_{d=1}^{D-1} u_r^d \\ x_{rj}^d &\leq u_r^d & (r, j) \in A^+(r); d = 1, \dots, D-1 \\ 0 &\leq x_{rj}^d \leq 1 & (r, j) \in A^+(r); d = 1, \dots, D-1 \\ 0 &\leq x_{rj} \leq 1 & (r, j) \in A^+(r) \\ 0 &\leq u_r^d \leq 1 & d = 1, \dots, D-1 \end{aligned}$$

e o poliedro \mathcal{Q}_r é definido por ($h = |H(r)|$):

$$\begin{aligned} X(A^+(r)) &= \sum_{d=1}^{D-1} d \cdot u_r^d + 1 \\ X(H(r)) &\leq 1 + \sum_{d=1}^{h-2} d \cdot u_r^d + (h-1) \cdot \sum_{d=h-1}^{D-1} u_r^d & H(r) \subseteq A^+(r), 2 \leq h \leq D-1 \\ 0 \leq x_{rj} &\leq 1 & (r, j) \in A^+(r) \\ 0 \leq u_r^d &\leq 1 & d = 1, \dots, D-1 \end{aligned}$$

Entre os dois poliedros, \mathcal{P}_r e \mathcal{Q}_r , verifica-se um resultado análogo à Proposição 3.4.1:

Proposição 3.4.2. *Para o nodo raiz*

$$\text{proj}_{x,u}(\mathcal{P}_r) \subseteq \mathcal{Q}_r$$

Demonstração. As restrições de ligação (3.27b) entre as variáveis x_{rj}^d e u_r^d , implicam que $X^d(H(r)) \leq (d+1) \cdot u_r^d$ para todo $d = 1, \dots, D-1$. Somando estas desigualdades para $d = 1, \dots, h-2$ obtém-se:

$$\sum_{d=1}^{h-2} X^d(H(r)) \leq \sum_{d=1}^{h-2} (d+1) \cdot u_r^d \quad (3.31)$$

Somando as desigualdades válidas $AD(r, j; d)$ para todo o arco em $H(r)$ e para todo o $d = h-1, \dots, D-1$, obtém-se:

$$\sum_{d=h-1}^{D-1} X^d(H(r)) \leq h \cdot \sum_{d=h-1}^{D-1} u_r^d \quad (3.32)$$

Somando as duas desigualdades (3.31) e (3.32) obtém-se:

$$\sum_{d=1}^{D-1} X^d(H(r)) = \sum_{d=1}^{h-2} X^d(H(r)) + \sum_{d=h-1}^{D-1} X^d(H(r)) \leq \sum_{d=1}^{h-2} (d+1) \cdot u_r^d + h \cdot \sum_{d=h-1}^{D-1} u_r^d$$

Por outro lado, a restrição de ligação (3.27c) entre as variáveis x_{rj}^0 e u_r^d implica que $X^0(H(r)) \leq 1 - \sum_{d=1}^{D-1} u_r^d$, o que somando à anterior desigualdade permite obter:

$$X^0(H(r)) + \sum_{d=1}^{D-1} X^d(H(r)) \leq \sum_{d=1}^{h-2} (d+1) \cdot u_r^d + h \cdot \sum_{d=h-1}^{D-1} u_r^d + 1 - \sum_{d=1}^{D-1} u_r^d$$

Reorganizando o segundo membro desta desigualdade e utilizando as restrições de ligação (3.26b), $\forall (r, j) \in H(r)$, no primeiro membro da desigualdade obtém-se a desigualdade válida $A_h(r, H(r))$.

Novamente, o primeiro conjunto de igualdades do poliedro \mathcal{P}_r em conjunto com as restrições de ligação (3.26b) implicam a primeira igualdade do poliedro \mathcal{Q}_r ficando assim provado o resultado. \square

Para um modelo genérico (P), designe-se por ($P+A$) o modelo obtido a partir do modelo (P) ao adicionar as desigualdades $A_h(i, H(i))$, $\forall i \in V$, para todos os valores de h para os quais as desigualdades válidas estão definidas.

Como consequência dos dois anteriores resultados conclui-se que:

Proposição 3.4.3. *Se as restrições (1.3) forem modeladas da mesma forma para os modelos (SUP_D+AD) e (SUP_D+A) então:*

$$proj_{x,u}(Adm(LSUP_D+AD)) \subseteq Adm(LSUP_D+A)$$

Demonstração. O resultado é consequência directa das Proposições 3.4.1 e 3.4.2 e tendo em conta que os poliedros \mathcal{P}_r e \mathcal{P}_i , $\forall i \in V \setminus \{r\}$, estão contidos no conjunto $Adm(LSUP_D + AD)$, assim como, os poliedros \mathcal{Q}_r e \mathcal{Q}_i , $\forall i \in V \setminus \{r\}$, estão contidos no conjunto de soluções $Adm(LSUP_D + A)$ (as restantes restrições de ambos os modelos são as mesmas). \square

E assim, como conclusão final tem-se:

Corolário 3.4.4. *Se as restrições (1.3) forem modeladas da mesma forma para os modelos $(SUP_D + AD)$ e $(SUP_D + A)$ então:*

$$V(LSUP_D) \leq V(LSUP_D + A) \leq V(LSUP_D + AD)$$

Retomando a instância apresentada na Figura 3.6(a), compara-se de seguida o limite inferior obtido pela relaxação linear do modelo $(SUP_D + AD)$, apresentado na Secção 3.4, com o limite inferior obtido pela relaxação linear do modelo $(SUP_D + A)$. Para avaliar melhor o papel destas, considerou-se a introdução "incremental" destas desigualdades no modelo (SUP_D) . Assim, o modelo $(SUP_D + A_t)$ representa o modelo (SUP_D) fortalecido com todas as desigualdades $A_h(i, H(i))$ com $h \leq t$, para o qual as desigualdades estão definidas ⁹.

⁹ $(SUP_D + A) \equiv (SUP_D + A_D - 1)$.

Exemplo 3.2: Modelo $(SUP_D + AD)$ vs modelo $(SUP_D + A)$ ¹⁰

Para a instância apresentada na Figura 3.6(a), o desvio do modelo discretizado sem a adição de desigualdades válidas é de 37.3% (cf. Figura 3.6(c)). Este desvio é reduzido para 17.4% com o modelo fortalecido ($LSUP_D + AD$) (cf. Figura 3.6(d)).

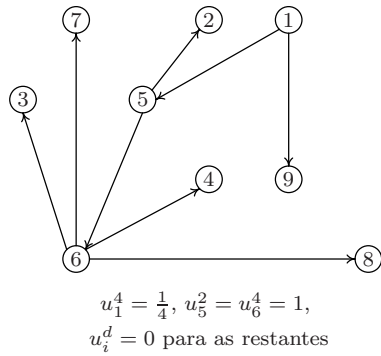
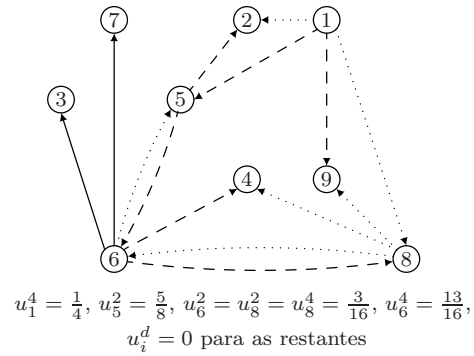
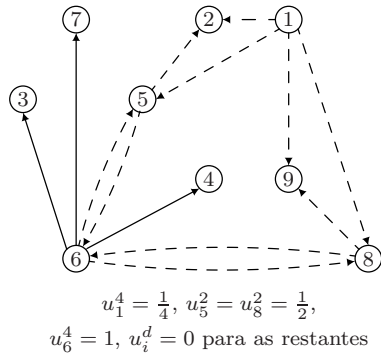
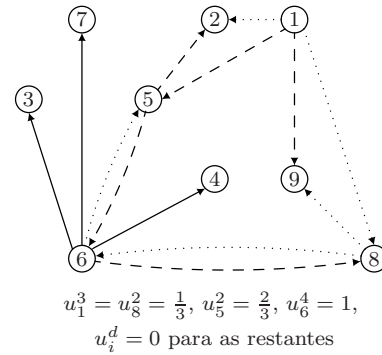
(a) $V(LSUP_D + A_1) = 297$.(b) $V(LSUP_D + A_2) = 297.75$ (arcos a traçado, $x_{ij} = \frac{5}{8}$; arcos a ponteadado, $x_{ij} = \frac{3}{8}$).(c) $V(LSUP_D + A_3) = 298.5$ (arcos a traçado, $x_{ij} = \frac{1}{2}$).(d) $V(LSUP_D + A_4) = 308$ (arcos a traçado, $x_{ij} = \frac{2}{3}$; arcos a ponteadado, $x_{ij} = \frac{1}{3}$).

Figura 3.7: Comparação dos modelos $(SUP_D + A_t)$, $t = 1, \dots, 4$, e $(SUP_D + AD)$ (arcos a cheio, $x_{ij} = 1$).

¹⁰Utilizando o sistema de fluxos *MC*.

O valor óptimo do modelo $(LSUP_D + A_1)$ é igual 297 (*cf.* Figura 3.7(a)), a que corresponde um desvio de 20.4%. Note-se que, na solução óptima deste modelo, o único nodo para o qual as desigualdades $A_2(i, H(i))$ não se verificam é o nodo raiz. De facto ($r \equiv 1$):

$$2 = x_{15} + x_{19} > 1 + \sum_{d=1}^4 u_1^d = 1 + \frac{1}{4}$$

No modelo $(LSUP_D + A_2)$ o desvio diminui para 20.2% (*cf.* Figura 3.7(b)) ao passo que no modelo $(LSUP_D + A_3)$ é igual a 19.9% (*cf.* Figura 3.7(c)). Finalmente, quando todas as desigualdades válidas $A_h(i, H(i))$, $h \leq 4$ são introduzidas, a solução óptima do modelo $(LSUP_D + A_4)$ é a mesma que a solução óptima do modelo $(LSUP_D + AD)$ (*cf.* Figura 3.6(d)).

No exemplo anterior, nenhuma das desigualdades válidas $A_h(i, H(i))$ é dominada por outra desigualdade válida $A_t(i, H(i))$ com $t < h$. Além disso, o exemplo mostra que a qualidade dos limites inferiores para o valor óptimo do problema, fornecidos pelas relaxações lineares do modelo, melhora à medida que se vão introduzindo as desigualdades válidas para conjuntos com um maior número de arcos. Note-se ainda que, a relação entre os modelos $(LSUP_D + A)$ e $(LSUP_D + AD)$, expressa no Corolário 3.4.4 verifica-se como igualdade para o problema apresentado. Dada a qualidade do limite inferior fornecido pelo modelo $(LSUP_D + A)$ face ao limite inferior fornecido pelo modelo $(LSUP_D + AD)$, poderia ter interesse implementar o modelo $(LSUP_D + A)$.

No entanto, embora o modelo $(LSUP_D + AD)$ contenha um conjunto extra de variáveis (que, assim como o sistema de restrições de ligação associado, é de dimensão polinomial), o número de desigualdades válidas no modelo $(LSUP_D + A)$ cresce exponencialmente em função de h ao passo que, no modelo $(LSUP_D + AD)$ as desigualdades válidas introduzidas são em número polinomial.

Assim sendo, na secção seguinte apenas se avalia a qualidade dos limites inferiores fornecidos pelos modelos $(LSUP_D + A_1)$ e $(LSUP_D + A_2)$ (este último permite considerar desigualdades válidas *Arco* escritas também para o nodo raiz).

3.4.2 Resultados Computacionais:

Modelos Discretizados fortalecidos

Nesta secção avalia-se a qualidade dos limites fornecidos pela relaxação linear dos modelos fortalecidos obtidos a partir do modelo discretizado (SUP_D) quando se introduzem as desigualdades válidas descritas nas secções anteriores. Nas tabelas seguintes (estrutura idêntica à das Tabelas 3.1 e 3.2) compara-se o modelo $(LSUP_D)$ com os modelos fortalecidos $(LSUP_D + A_1)$, $(LSUP_D + A_2)$ e $(LSUP_D + AD)$. Novamente, não são apresentados tempos de CPU para os modelos com sistema de fluxo SC^* , por estes serem inferiores ou iguais a 1 segundo.

A análise dos resultados para o modelo $(LSUP_D)$ (ver Secção 3.3.2) quanto à dimensão, tipo de rede e efeito do par de custo tecnológicos, mantém-se para os modelos $(LSUP_D + A_1)$, $(LSUP_D + A_2)$ e $(LSUP_D + AD)$.

No entanto, quanto ao parâmetro D , os desvios não se comportam de uma maneira uniforme para todas as instâncias. Para as instâncias *Wireless* com 50 nodos o desvio aumenta com o grau máximo dos nodos. Para as restantes instâncias, o desvio aumenta para os maiores valores do grau máximo ($D = 5, 6$), tendo um comportamento oscilante para os valores mais baixos.

Comparação de Modelos fortalecidos.

À excepção do caso das redes *Wireless* para $D = 3$ (25 e 50 nodos) onde não houve alteração, o modelo $(LSUP_D + A_1)$ consegue sempre reduzir o desvio obtido com o modelo sem desigualdades válidas adicionadas. Esta redução é mais eficaz à medida que o grau máximo aumenta e é consideravelmente mais notória para as redes *Cabo*. Os tempos de CPU do modelo $(LSUP_D + A_1)$ com sistema de fluxos *MC* aumentam mas continuam relativamente baixos (inferiores a 37 segundos), sendo mais elevados para as redes *Wireless*.

A introdução das desigualdades válidas $A_2(i, H(i))$ no modelo fortalecido $(LSUP_D + A_1)$ não parece provocar uma redução significativa nos desvios obtidos com este modelo. Em quase todos os casos testados, os desvios obtidos com os modelos $(LSUP_D + A_1)$ e $(LSUP_D + A_2)$ são iguais¹¹. Isto deve-se, quase na maioria dos casos, ao facto de que o limite obtido com o modelo $(LSUP_D + A_1)$ já é igual ao limite obtido com o modelo com desigualdades *Arco Desagregadas*, $(LSUP_D + AD)$, e daí a introdução das desigualdades $A_2(i, H(i))$ não vir a melhorar a qualidades dos limites inferiores. É o que se passa, por exemplo, em todas as instâncias *Wireless* com maior número de arestas.

¹¹Existem diferenças significativas (mas inferiores a 1%) para as redes *Cabo* com 50 nodos.

R	$ E $	(K_1, K_2)	D	$(LSUP_D)$			$(LSUP_D + A_1)$			$(LSUP_D + A_2)$			$(LSUP_D + AD)$			
				desvio	CPU		desvio	CPU		desvio	CPU		desvio	CPU		
				SC^*	MC	MC	SC^*	MC	MC	SC^*	MC	MC	SC^*	MC	MC	
W	(100, 10)	75	3	3,9	3,7	0	3,9	3,7	0	3,9	3,7	0	3,9	3,7	0	
			4	3,7	3,4	0	3,2	3,0	0	3,2	2,9	0	3,0	2,9	1	
			5	5,9	5,6	0	4,3	4,2	0	4,3	4,1	1	4,0	3,8	1	
			6	19,5	19,2	0	15,5	15,4	0	15,5	15,4	1	15,3	15,2	1	
	(100, 50)	75	3	4,0	3,8	0	4,0	3,8	0	4,0	3,8	0	4,0	3,8	0	
			4	1,6	1,4	0	1,3	1,2	0	1,2	1,1	0	1,2	1,1	0	
			5	5,3	5,1	0	4,3	4,2	0	4,2	4,1	1	4,1	3,9	1	
			6	15,9	15,7	0	13,0	13,0	0	13,0	13,0	1	12,6	12,5	1	
	C	(100, 10)	150	3	3,8	3,8	1	3,8	3,8	1	3,8	3,8	1	3,8	3,8	1
				4	3,5	3,5	0	3,2	3,1	1	3,2	3,1	2	3,2	3,1	2
				5	4,9	4,9	0	3,8	3,6	1	3,8	3,6	2	3,8	3,6	2
				6	9,0	9,0	0	6,9	6,8	1	6,9	6,8	2	6,9	6,8	2
(100, 50)		150	3	3,9	3,9	1	3,9	3,9	1	3,9	3,9	1	3,9	3,9	1	
			4	1,5	1,5	0	1,3	1,2	1	1,3	1,2	2	1,3	1,2	1	
			5	4,6	4,6	0	3,9	3,8	1	3,9	3,8	2	3,9	3,8	1	
			6	8,7	8,7	0	7,3	7,2	1	7,3	7,2	2	7,3	7,2	2	
C		(100, 10)	75	3	6,2	5,7	0	3,0	2,8	0	3,0	2,8	0	3,0	2,8	0
				4	11,1	10,6	0	2,8	2,1	0	2,8	2,1	0	2,8	2,1	0
				5	19,9	19,4	0	6,9	6,1	0	6,9	6,1	0	6,6	5,8	0
				6	24,4	23,9	0	6,9	6,2	0	6,9	6,2	0	6,0	5,3	0
	(100, 50)	75	3	5,8	5,5	0	3,2	3,0	0	3,2	3,0	0	3,2	3,0	0	
			4	8,3	7,9	0	2,3	1,9	0	2,3	1,9	0	2,1	1,8	0	
			5	17,1	16,7	0	6,4	5,8	1	6,4	5,8	0	6,1	5,5	0	
			6	22,4	22,0	0	6,9	6,5	1	6,9	6,5	0	4,7	4,3	1	
	(100, 10)	150	3	5,9	5,3	0	4,1	3,4	1	4,1	3,4	1	4,1	3,4	1	
			4	9,6	9,3	0	3,8	3,0	1	3,8	3,0	1	3,8	3,0	1	
			5	14,3	13,9	0	3,0	2,1	1	3,0	2,1	1	2,9	2,1	2	
			6	20,4	20,0	0	4,4	3,6	1	4,4	3,6	2	4,2	3,3	2	
(100, 50)		150	3	5,6	5,1	0	4,1	3,5	0	4,1	3,5	1	4,1	3,5	1	
			4	6,9	6,7	0	2,9	2,2	1	2,9	2,2	1	2,9	2,2	1	
			5	12,2	11,9	0	3,3	2,6	1	3,3	2,6	2	3,2	2,5	1	
			6	18,1	17,9	0	4,8	4,1	1	4,8	4,1	1	4,2	3,5	2	

Tabela 3.3: Resultados para instâncias com 25 nodos: Modelos fortalecidos.

Comparando agora os modelos $(LSUP_D + A_1)$ e $(LSUP_D + AD)$, o modelo com desigualdades *Arco Desagregadas* consegue reduzir ainda o desvio obtido com o modelo $(LSUP_D + A_1)$ para os valores mais elevados do grau máximo, principalmente para as redes *Cabo* com 50 nodos.

R	$ E $	(K_1, K_2)	D	$(LSUP_D)$			$(LSUP_D + A_1)$			$(LSUP_D + A_2)$			$(LSUP_D + AD)$			
				desvio	CPU		desvio	CPU		desvio	CPU		desvio	CPU		
				SC^*	MC	MC	SC^*	MC	MC	SC^*	MC	MC	SC^*	MC	MC	
150	(100, 10)		3	0,1	0*	9	0,1	0*	9	0,1	0*	10	0,1	0*	10	
			4	0,5	0,3	1	0,2	0,2	2	0,2	0,2	4	0,2	0,2	4	
			5	1,8	1,6	1	0,7	0,7	2	0,6	0,6	4	0,6	0,6	5	
			6	6,2	6,0	1	3,2	3,2	2	3,1	3,0	6	3,0	3,0	13	
	(100, 50)		3	0,1	0*	9	0,1	0*	9	0,1	0*	10	0,1	0*	22	
			4	0,3	0,2	2	0,2	0,1	2	0,1	0,1	5	0,1	0,1	5	
			5	1,2	1,1	1	0,5	0,5	3	0,4	0,4	6	0,4	0,4	7	
			6	5,5	5,4	1	3,4	3,4	3	3,3	3,3	7	3,2	3,2	12	
	300	(100, 10)		3	0,1	0,1	27	0,1	0,1	36	0,1	0,1	136	0,1	0,1	48
				4	1,1	1,0	2	0,1	0,1	16	0,1	0,1	29	0,1	0,1	17
				5	2,4	2,3	2	0,2	0,2	9	0,2	0,2	25	0,2	0,2	21
				6	6,7	4,0	1	3,2	3,1	14	3,2	3,1	30	3,2	3,1	35
(100, 50)			3	0,1	0,1	23	0,1	0,1	35	0,1	0,1	134	0,1	0,1	38	
			4	0,7	0,6	2	0,1	0,1	20	0,1	0,1	37	0,1	0,1	33	
			5	1,6	1,5	2	0,2	0,2	9	0,2	0,2	20	0,2	0,2	16	
			6	5,8	4,0	1	3,5	3,4	22	3,5	3,4	38	3,5	3,4	35	
C		(100, 10)		3	2,8	2,7	1	0,7	0,5	2	0,7	0,5	4	0,7	0,5	2
				4	7,6	7,6	1	0,9	0,8	2	0,5	0,4	3	0,5	0,4	2
				5	13,2	12,9	1	2,3	2,2	2	1,5	1,3	4	1,3	1,2	5
				6	16,4	16,2	1	2,1	2,1	1	1,3	1,2	2	1,0	0,9	2
	(100, 50)		3	2,4	2,4	1	0,6	0,5	3	0,6	0,5	4	0,6	0,5	3	
			4	5,7	5,7	1	0,9	0,8	2	0,6	0,6	4	0,6	0,6	3	
			5	12,1	11,9	1	2,8	2,7	2	2,1	2,0	3	1,9	1,8	3	
			6	15,9	15,8	1	2,8	2,8	1	2,0	2,0	4	1,2	1,1	4	
	(100, 10)		3	5,8	5,7	2	2,3	1,8	7	2,1	1,7	17	2,1	1,7	11	
			4	12,3	12,2	2	3,0	2,6	7	3,0	2,6	12	3,0	2,6	8	
			5	18,3	18,1	2	2,3	1,9	9	2,3	1,9	18	1,8	1,5	15	
			6	24,4	24,2	2	3,4	3,1	8	3,4	3,1	18	2,4	2,2	18	
(100, 50)		3	4,8	4,8	2	2,0	1,5	6	1,8	1,4	15	1,8	1,4	9		
		4	8,1	8,0	2	2,2	1,8	6	2,2	1,8	17	2,2	1,8	7		
		5	16,2	16,0	2	3,2	3,0	8	3,2	3,0	13	2,1	2,0	14		
		6	23,0	22,8	2	4,7	4,4	8	4,7	4,4	20	2,1	2,1	19		

Tabela 3.4: Resultados para instâncias com 50 nodos: Modelos fortalecidos.

No caso das redes *Cabo*, onde os desvios obtidos pelo modelo básico ($LSUP_D$) são maiores, os modelos fortalecidos conseguem uma maior redução nas instâncias com 50 nodos. Os tempos de CPU para os modelos fortalecidos, que utilizam o sistema de fluxos MC , atingem valores mais elevados para o modelo ($LSUP_D + A_2$), sendo mais elevados para as instâncias com maior número de arestas, como seria de esperar. Por este motivo, em conjunto

com a baixa eficácia da introdução das desigualdades válidas $A_2(i, H(i))$ na redução dos desvios, nos próximos resultados computacionais apenas será considerada a introdução das desigualdades válidas $A_1(i, H(i))$ e das desigualdades válidas $AD(i, j; d)$.

3.5 Modelos Árvore de Suporte/Saco Mochila

Nesta secção apresenta-se uma perspectiva diferente de analisar e modelar o problema (ASupCG), realçando uma estrutura de Saco Mochila implícita no modelo não linear (SUP_{NL}), à qual se aplica a técnica de reformulação por caminhos apresentada na Secção 1.3.2, obtendo-se assim um novo modelo linear. Posteriormente, aplica-se também a técnica de discretização a este novo modelo obtendo-se assim um modelo que permitirá fazer a ligação entre os modelos apresentados nesta secção e os modelos lineares apresentados nas secções anteriores. No final desta secção apresenta-se um conjunto de desigualdades válidas motivadas pelo subproblema de Saco Mochila.

3.5.1 Parte I: Reformulação por caminhos

Para fazer surgir a estrutura de Saco Mochila, começa-se por somar a restrição (3.2b) com as restrições (3.2a), $\forall i \in V \setminus \{r\}$, obtendo-se assim ¹²:

$$\sum_{i \in V} U_i = \sum_{i \in V} X(A^+(i)) - 1 = X(A) - 1 = n - 2 \quad (3.33)$$

¹²O número total de arcos em qualquer solução é $n - 1$.

Ao adicionar esta igualdade ao modelo (SUP_{NL}) é possível identificar dois poliedros ligados entre si através das restrições (3.2a) e (3.2b). O primeiro, nas variáveis x , definido como $\mathcal{AS}_r \equiv \{x_{ij} : (1.2), (1.3) \text{ e } (1.4)\}$, corresponde ao poliedro das arborescências de suporte orientadas a partir da raiz r no grafo $G = (V, A, c_{ij})$. O segundo poliedro, nas variáveis U , definido como $\mathcal{SM} \equiv \{U_i : (3.3), (3.33) \text{ e } (3.4)\}$, corresponde ao poliedro de um *Problema de Saco Mochila Limitado* com uma restrição de igualdade (ver [36, 44]).

A Figura 3.5.1 mostra o modelo não linear reescrito de modo a dar ênfase aos dois poliedros.

$$(SUP/SM_{NL}) \min \sum_{(i,j) \in A} c_{ij} x_{ij} + \sum_{i \in V} \Phi(U_i + 1) \quad (3.1)$$

$$s.a : \quad X(A^+(i)) = \begin{cases} U_i & i \in V \setminus \{r\} \\ U_r + 1 & i = r \end{cases} \quad (3.2a)$$

$$\mathbf{x} \in \mathcal{AS}_r$$

$$\mathbf{U} \in \mathcal{SM}$$

Figura 3.8: Modelo Árvore de Suporte/Saco Mochila não linear para o (ASupCG).

Uma maneira de obter uma descrição completa do envolvente convexo do poliedro \mathcal{SM} consiste em usar a técnica de reformulação por caminhos apresentada na Seção 1.3.2. Assim, substituem-se as restrições do poliedro \mathcal{SM} por uma formulação estendida, pseudo-polinomial, que corresponde ao sistema de equações de caminho associado à resolução do subproblema de Saco Mochila através de Programação Dinâmica. Para tal, considere-se $G_E = (V_E, A_E)$, o grafo expandido acíclico, associado a esse sistema.

Cada nodo deste grafo é representado por i_s e tem associado dois parâmetros inteiros i e s , onde:

- i , representa cada um dos nodos do grafo original ($i = 1 \dots, n$);
- $s = s(i)$, representa a soma inteira $\sum_{j=1}^i U_j$, ($s = 0, \dots, n - 2$).

Adicionalmente é criado um nodo inicial 0_0 . Pela definição do parâmetro s , conclui-se que o seu valor depende exclusivamente do nodo original ao qual se refere e da posição que este ocupa na sequência de nodos $\{1, \dots, n\}$. Assim sendo, como $U_j \in \{0, \dots, D - 1\}$ para cada $j \in V$, tem-se:

$$s(i) = \sum_{j=1}^i U_j \in \{0, \dots, i \cdot (D - 1)\}$$

mas também

$$\sum_{j=i+1}^n U_j \in \{0, \dots, (n - i) \cdot (D - 1)\}$$

Este último resultado, em conjunto com a equação $\sum_{i \in V} U_i = n - 2$, implica também que

$$s(i) \in \{(n - 2) - (n - i) \cdot (D - 1), \dots, n - 2\}$$

Ou seja, o conjunto de nodos do grafo expandido é definido como:

$$V_E = \{i_s : i = 0, \dots, n; s = a(i), \dots, b(i)\}$$

onde, para cada i , os valores mínimo e máximo para o parâmetro s são definidos, respectivamente, como

$$a_i = \max(0, (n-2) - (n-i) \cdot (D-1))$$

$$b_i = \min(i \cdot (D-1), n-2)$$

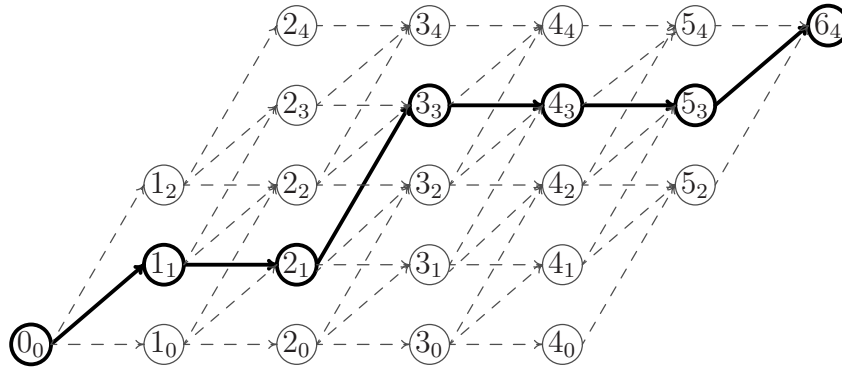
Os arcos em A_E são da forma $(i-1_s, i_t)$, serão representados sem perda de generalidade como $(i_{s,t})$ e representam a decisão de atribuir o grau $t-s+1$ ao nodo i no grafo original. O conjunto de arcos fica então definido como:

$$A_E = \{ (i_{s,t}) : i-1_s, i_t \in V_E, 0 \leq t-s \leq D-1 \}$$

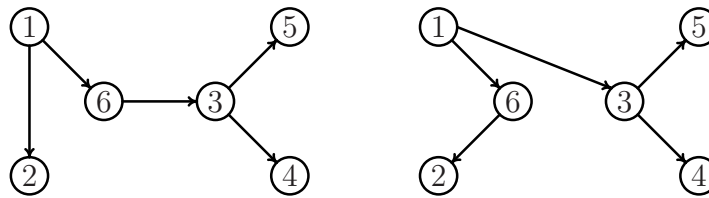
No grafo expandido, qualquer caminho do nodo 0_0 ao nodo n_{n-2} representa uma única atribuição admissível de graus aos nodos do grafo original, como é ilustrado pelo exemplo seguinte.

Exemplo 3.3: Construção de um grafo expandido

O exemplo seguinte ilustra a construção de um grafo expandido associado ao poliedro \mathcal{SM} com $n = 6$ nodos e grau máximo nos nodos igual a 3. Para cada um dos 6 nodos do grafo original criam-se $n-1$ cópias. Depois de criados os arcos no grafo expandido eliminam-se todos os nodos-cópias que não façam parte de caminhos do nodo 0_0 ao nodo 6_4 obtendo-se o grafo expandido da Figura 3.9(a). A traço cheio está representado um caminho admissível de 0_0 a 6_4 para o qual se apresentam, na Figura 3.9(b), duas possíveis soluções admissíveis no grafo original (assumindo que os respectivos arcos existem no grafo original).



(a) Caminho admissível no grafo expandido (arcos a cheio).



(b) Soluções admissíveis no grafo original, associadas ao caminho indicado no grafo expandido.

Figura 3.9: Grafo expandido associado a uma instância do problema de Saco Mochila com $n = 6$ e $D - 1 = 2$.

Este exemplo permite verificar que a dimensão do grafo expandido é uma função polinomial da dimensão do grafo original. O número de nodos no grafo expandido é dado por

$$1 + \sum_{i=1}^n (b_i - a_i + 1) = 2 \cdot \sum_{k=0}^{k_{max}} (k \cdot (D - 1) + 1) + (n - 1) \cdot (n - 2 \cdot k_{max} - 1)$$

onde $k_{max} = \lfloor \frac{n-2}{D-1} \rfloor$.

Para definir caminhos no grafo expandido, é necessário criar *variáveis-caminho*, w_i^{st} , que indicam se o arco $(i_{s,t}) \in A_E$ está no caminho mais curto do nodo 0_0 para o nodo n_{n-2} . A Figura 3.10 apresenta as equações¹³ que modelam qualquer caminho neste grafo expandido (a notação $W(\Theta)$ com $\Theta \subset A_E$ é análoga à utilizada para as variáveis x).

$$W(A_E^+(0_0)) = 1 \quad (3.34a)$$

$$W(A_E^-(i_s)) = W(A_E^+(i_s)) \quad \forall i_s \in V_E : i = 1, \dots, n-1 \quad (3.34b)$$

$$W(A_E^-(n_{n-2})) = 1 \quad (3.34c)$$

$$w_i^{st} \in \{0, 1\} \quad \forall (i_{s,t}) \in A_E \quad (3.34d)$$

Figura 3.10: Sistema de equações de caminho no grafo expandido.

Para poder usar este sistema linear em substituição das restrições do Saco Mochila, (3.3), (3.33) e (3.4), é necessário ainda ligar os dois conjuntos de variáveis, o que é feito através das seguintes igualdades:

$$U_i = \sum_{(i_{s,t}) \in A_E} (t-s) \cdot w_i^{st} \quad , \forall i \in V \quad (3.35)$$

Se $U_i = d'$ para algum nodo i , então apenas uma variável w_i^{st} (garantido pelas restrições de caminho) será igual a 1, tal que $t-s = d'$, ou seja $\sum_{s=a(i)}^{b(i)} w_i^{s,s+d'} = 1$. Caso $U_i = 0$ para algum nodo i então todas as variáveis w_i^{st} com $t-s \neq 0$ serão nulas e $\sum_{s=a(i)}^{b(i)} w_i^{s,s} = 1$.

¹³Neste sistema, uma das equações (3.34a) ou (3.34c) pode ser eliminada por redundância.

O facto de o sistema (3.34a) – (3.34d) satisfazer a propriedade da integridade (ver [1]) garante que a projecção do conjunto $\{U_i, w_i^{st} : (3.34a) - (3.34c), (3.35), (3.4) \text{ e } w_i^{st} \geq 0, \forall (i,s,t) \in A_E\}$ no espaço definido pelas variáveis U_i , é dada por $conv(\mathcal{SM})$.

Assim, é possível obter uma nova formulação para o (ASupCG) onde o conjunto de restrições em \mathcal{SM} é substituído pelo conjunto

$$\mathcal{SM}/Cam_0 \equiv \{U_i, w_i^{st} : (3.34a), (3.34b), (3.34c), (3.34d), (3.4) \text{ e } (3.35)\}$$

O principal interesse na aplicação da técnica de reformulação por caminhos deve-se ao facto de, utilizando as restrições de ligação (3.35), se tornar possível reescrever o termo não linear da função objectivo (3.1) de uma forma linear, (tendo em conta que, para cada $i \in V$, $\sum_{(i,s,t) \in A_E} w_i^{st} = 1$):

$$\sum_{i \in V} \Phi(U_i + 1) = \sum_{i \in V} \Phi \left(\sum_{(i,s,t) \in A_E} (t - s) \cdot w_i^{st} + 1 \right) = \sum_{i \in V} \sum_{\substack{(i,s,t) \in A_E \\ t-s \geq 1}} \lambda^{t-s} \cdot w_i^{st}$$

onde, como foi definido na Secção 3.3, $\lambda^{t-s} = \Phi(t - s + 1)$.

Finalmente, o primeiro modelo linear Árvore de Suporte/Saco Mochila, utilizando a técnica de reformulação por caminhos é apresentado na Figura 3.11 (o índice "C" indica a utilização da técnica de reformulação por caminhos).

As variáveis U_i podem ser eliminadas do modelo, (através das restrições (3.35)) no entanto são mantidas com o propósito de facilitar a introdução do próximo modelo. Em termos da relaxação linear, o modelo (SUP/SM_C) é mais forte do que o modelo discretizado, (SUP_D) apresentado na Secção 3.3,

$$(SUP/SM_C) \min \sum_{(i,j) \in A} c_{ij} x_{ij} + \sum_{i \in V} \sum_{\substack{(i,s,t) \in A_E \\ t-s \geq 1}} \lambda^{t-s} \cdot w_i^{st} \quad (3.36)$$

$$s.a : X(A^+(i)) = \begin{cases} U_i & i \in V \setminus \{r\} \\ U_r + 1 & i = r \end{cases} \quad (3.2a)$$

$$\mathbf{x} \in AS_r$$

$$(\mathbf{U}, \mathbf{w}) \in SM/Cam_0 \quad (3.2b)$$

Figura 3.11: Modelo Árvore de Suporte/Saco Mochila para o (ASupCG).

como os resultados computacionais no final desta secção mostram. Para provar este resultado é necessário primeiro contextualizar o modelo (SUP/SM_C) em termos das variáveis discretizadas, aplicando-lhe a técnica de discretização.

3.5.2 Parte II: Reformulação por caminhos e por discretização

A partir do modelo (SUP/SM_C) é possível obter um modelo equivalente em termos da relaxação linear, aplicando a técnica de discretização às variáveis inteiras U_i . Para tal, utilizando as restrições de ligação (3.19) entre as variáveis U_i e u_i^d , substituem-se as restrições de grau externo (3.2a) e (3.2b) no modelo (SUP/SM_C) pelas restrições (3.21a) e (3.21b), respectivamente.

No conjunto SM/Cam_0 substituem-se as restrições de domínio (3.4) por (3.23) e as restrições de ligação anteriores, (3.35), pelas novas restrições de ligação:

$$\sum_{d=1}^{D-1} d \cdot u_i^d = \sum_{(i,s,t) \in A_E} (t-s) \cdot w_i^{st} \quad , i \in V \quad (3.37)$$

obtendo-se assim o novo conjunto

$$\mathcal{SM}/Cam_1 \equiv \{u_i^d, w_i^{st} : (3.34a), (3.34b), (3.34c), (3.34d), (3.23) \text{ e } (3.37)\}$$

A Figura 3.12 apresenta o modelo assim obtido (o índice "CD" indica a utilização das duas técnicas de reformulação apresentadas na Secção 1.3).

$$(SUP/SM_{CD1}) \min \sum_{(i,j) \in A} c_{ij} x_{ij} + \sum_{i \in V} \sum_{\substack{(i,s,t) \in A_E \\ t-s \geq 1}} \lambda^{t-s} \cdot w_i^{st} \quad (3.36)$$

$$s.a : \quad X(A^+(i)) = \begin{cases} \sum_{d=1}^{D-1} d \cdot u_i^d & i \in V \setminus \{r\} \\ \sum_{d=1}^{D-1} d \cdot u_r^d + 1 & i = r \end{cases} \quad (3.21a)$$

$$\mathbf{x} \in \mathcal{AS}_r$$

$$(\mathbf{u}, \mathbf{w}) \in \mathcal{SM}/Cam_1$$

Figura 3.12: Modelo Árvore de Suporte/Saco Mochila nas variáveis discretizadas para o (ASupCG).

A relaxação linear do modelo (SUP/SM_{CD1}) produz exactamente o mesmo limite inferior para o problema que a relaxação linear do modelo (SUP/SM_C) pois apenas foi aplicada a técnica de discretização a este modelo o que, por si só, não altera a qualidade dos limites obtidos (como pode ser observado nos trabalhos anteriores onde esta técnica foi utilizada, ver [9, 33], por exemplo).

Proposição 3.5.1.

$$V(LSUP/SM_C) = V(LSUP/SM_{CD1})$$

Para tentar obter um modelo Árvore de Suporte/Saco Mochila mais forte do que o modelo (SUP/SM_{CD1}) , pode-se considerar a seguinte versão desagregada das igualdades (3.37):

$$u_i^d = \sum_{(i,s,s+d) \in A_E} w_i^{s,s+d} \quad i \in V, \quad d = 1, \dots, D-1 \quad (3.38)$$

Estas igualdades são de facto uma desagregação das anteriores (3.37) pois, para cada nodo $i \in V$, ao multiplicar cada uma das equações (3.38) pelo respectivo coeficiente d e somando-as de seguida para todo $d = 1, \dots, D-1$ obtém-se a igualdade (3.37) para o nodo i (pela definição do conjunto A_E , $0 \leq t-s \leq D-1, \forall (i,s,t) \in A_E$):

$$\sum_{d=1}^{D-1} d \cdot u_i^d = \sum_{d=1}^{D-1} d \cdot \sum_{(i,s,s+d) \in A_E} w_i^{s,s+d} \stackrel{(d=t-s)}{=} \sum_{\substack{(i,s,t) \in A_E \\ 1 \leq t-s \leq D-1}} (t-s) \cdot w_i^{st} = \sum_{(i,s,t) \in A_E} (t-s) \cdot w_i^{st}$$

Designa-se então por (SUP/SM_{CD2}) (ver Figura 3.13) o modelo Árvore de Suporte/Saco Mochila obtido do modelo (SUP/SM_{CD1}) onde o conjunto SM/Cam_1 é substituído pelo conjunto

$$SM/Cam_2 \equiv \{u_i^d, w_i^{st} : (3.34a), (3.34b), (3.34c), (3.34d), (3.23) \text{ e } (3.38)\}$$

$$(SUP/SM_{CD2}) \min \sum_{(i,j) \in A} c_{ij} x_{ij} + \sum_{i \in V} \sum_{\substack{(i,s,t) \in A_E \\ t-s \geq 1}} \lambda^{t-s} \cdot w_i^{st} \quad (3.36)$$

$$s.a : \quad X(A^+(i)) = \begin{cases} \sum_{d=1}^{D-1} d \cdot u_i^d & i \in V \setminus \{r\} \\ \sum_{d=1}^{D-1} d \cdot u_r^d + 1 & i = r \end{cases} \quad (3.21a)$$

$$\mathbf{x} \in \mathcal{AS}_r$$

$$(\mathbf{u}, \mathbf{w}) \in \mathcal{SM}/Cam_2$$

Figura 3.13: Modelo desagregado Árvore de Suporte/Saco Mochila nas variáveis discretizadas para o (ASupCG).

Porém, ao contrário do que seria de esperar, este modelo "desagregado", não consegue ser mais forte que o anterior modelo "agregado" (SUP/SM_{CD1}): a relaxação linear de ambos produz exactamente o mesmo limite, como se provará de seguida.

Proposição 3.5.2.

$$V(LSUP/SM_{CD1}) = V(LSUP/SM_{CD2})$$

Demonstração. O gráfico da Figura 3.14 ilustra o que na realidade se passa entre os dois modelos em termos das respectivas relaxações lineares: existe sempre uma solução óptima do modelo "agregado" ($LSUP/SM_{CD1}$) que satisfaz as restrições de ligação "desagregadas" (3.38).

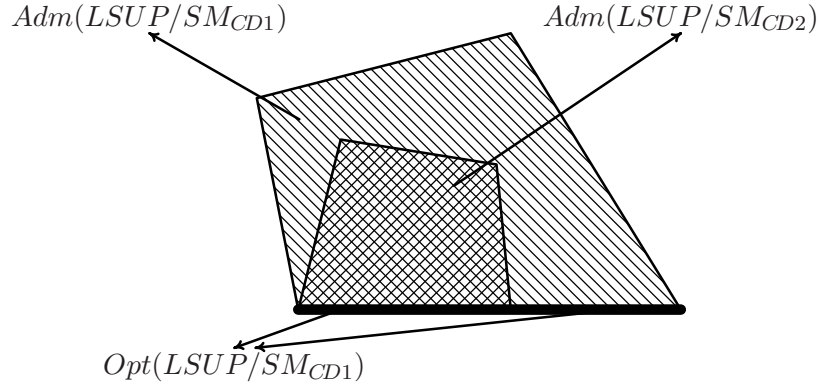


Figura 3.14: Relação entre as relaxações lineares dos modelos (SUP/SM_{CD1}) e (SUP/SM_{CD2}) (O conjunto de soluções óptimas de $(LSUP/SM_{CD1})$ está indicado a traço grosso).

1. Em primeiro lugar é intuitivo que

$$Adm(LSUP/SM_{CD2}) \subseteq Adm(LSUP/SM_{CD1})$$

Como já foi referido, as restrições de ligação (3.38), presentes no modelo (SUP/SM_{CD2}) , implicam as restrições de ligação (3.37) do modelo (SUP/SM_{CD1}) . Como a função objectivo é a mesma conclui-se que:

$$V(LSUP/SM_{CD2}) \geq V(LSUP/SM_{CD1})$$

2. Resta assim provar que

$$Adm(LSUP/SM_{CD2}) \cap Opt(LSUP/SM_{CD1}) \neq \emptyset$$

ou seja, uma solução óptima para a relaxação linear do modelo (SUP/SM_{CD1}) que não satisfaça algumas das restrições (3.38), pode ser sempre transformada linearmente numa outra solução com o mesmo custo e que já satisfaça todas as restrições (3.38).

Seja $(\check{x}, \check{u}, \check{w})$ uma solução óptima de (SUP/SM_{CD1}) e admita-se que alguma das restrições (3.38) não é satisfeita por esta solução. Considere-se também uma solução $(\hat{x}, \hat{u}, \hat{w})$ tal que, $\hat{x} = \check{x}$, $\hat{w} = \check{w}$ e o valor das variáveis u_i^d é obtido da seguinte forma:

$$\hat{u}_i^d = \sum_{(i,s,s+d) \in A_E} \check{w}_i^{s,s+d} \quad , i \in V, d = 1, \dots, D-1$$

Assim sendo, a solução $(\hat{x}, \hat{u}, \hat{w})$ satisfaz, obviamente, todas as restrições (3.38), ao contrário da solução óptima $(\check{x}, \check{u}, \check{w})$.

Falta averiguar se também as restantes restrições do modelo $(LSUP/SM_{CD2})$ são verificadas pela solução $(\hat{x}, \hat{u}, \hat{w})$, nomeadamente as restrições de ligação entre as variáveis u e x e as restrições de consistência e de domínio das variáveis u . De facto, em relação às restrições de ligação (3.21a) e (3.21b), elas são verificadas para qualquer $i \in V \setminus \{r\}$ (para $i = r$ a demonstração é análoga) visto:

$$\begin{aligned} \sum_{d=1}^{D-1} d \cdot \hat{u}_i^d &= \sum_{d=1}^{D-1} d \cdot \sum_{(i,s,s+d) \in A_E} \check{w}_i^{s,s+d} \stackrel{d=t-s}{=} \sum_{\substack{(i,s,t) \in A_E \\ t-s \geq 1}} (t-s) \cdot \check{w}_i^{st} = \\ &= \sum_{d=1}^{D-1} d \cdot \check{u}_i^d = \check{X}(A^+(i)) = \hat{X}(A^+(i)) \end{aligned}$$

Pela maneira como o grafo expandido foi construído, as restrições de caminho garantem que $\sum_{(i,s,t) \in A_E} \check{w}_i^{st} = 1$, para cada $i \in V$.

Ficam assim garantidas as restrições (3.22) bem como as restrições de domínio (3.23) ¹⁴:

$$\sum_{d=1}^{D-1} \hat{u}_i^d = \sum_{d=1}^{D-1} \sum_{(i,s,s+d) \in A_E} \check{w}_i^{s,s+d} \leq 1$$

$$0 \leq \hat{u}_i^d = \sum_{(i,s,s+d) \in A_E} \check{w}_i^{s,s+d} \leq 1$$

Ou seja, a solução $(\hat{x}, \hat{u}, \hat{w})$ é admissível para o modelo $(LSUP/SM_{CD2})$ e como esta solução apenas difere da solução $(\check{x}, \check{u}, \check{w})$ no valor das variáveis u , as duas soluções têm o mesmo valor, já que os dois modelos têm a mesma função objectivo, escrita nas variáveis x e w . Como tal, a solução $(\hat{x}, \hat{u}, \hat{w})$ é solução óptima para o modelo (SUP/SM_{CD1}) , e admissível para o modelo $(LSUP/SM_{CD2})$, e assim

$$V(LSUP/SM_{CD2}) \leq V(LSUP/SM_{CD1})$$

Fica assim provado que os dois modelos Árvore de Suporte/Saco Mochila com variáveis discretizadas são equivalentes em termos das respectivas relaxações lineares. \square

3.5.3 Comparação de relaxações lineares: Parte III

É este último modelo, (SUP/SM_{CD2}) , que faz a ligação entre o modelo discretizado (SUP_D) , apresentado na Secção 3.3, e o modelo (SUP/SM_C) , apresentado na Secção 3.5.1. A proposição seguinte faz a ligação entre estes dois modelos, em termos das respectivas relaxações lineares:

¹⁴A variável $\check{w}_i^{s,s}$ não entra no somatório, daí $\sum_{\substack{(i,s,t) \in A_E \\ t-s \geq 1}} \check{w}_i^{s,t} \leq 1$.

Proposição 3.5.3. $V(LSUP_D) \leq V(LSUP/SM_{CD2})$

Demonstração. Para mostrar que, quanto à qualidade dos limites inferiores fornecidos, a relaxação linear do modelo (SUP/SM_{CD2}) é pelo menos tão boa como a relaxação linear do modelo (SUP_D) , note-se que esta última pode ser "fortalecida" de forma a obter a primeira. Para tal, comece-se por adicionar a restrição redundante:

$$\sum_{d=1}^{D-1} d \cdot \sum_{i \in V} u_i^d = n - 2 \quad (3.39)$$

ao modelo (SUP_D) . Esta igualdade não é mais do que a igualdade (3.33), descrita no início da Secção 3.5.1, depois de fazer a substituição das variáveis U_i pelas variáveis u_i^d , utilizando as igualdades (3.19). Como é natural, também pode ser obtida adicionado as restrições (3.21a) para todo $i \in V \setminus \{r\}$ com a restrição (3.21b).

Depois da inclusão no modelo (SUP_D) das igualdades (3.39), o problema pode ser visto novamente, como sendo composto por dois subproblemas ligados pelas restrições (3.21a) e (3.21b). O primeiro subproblema é novamente um problema de *Arborescência de Suporte* nas variáveis x e o segundo é desta vez, um problema de *Saco Mochila de Escolha Múltipla* com uma restrição de igualdade, nas variáveis u (ver [36]):

$$\sum_{d=1}^{D-1} u_i^d \leq 1 \quad i \in V \quad (3.22)$$

$$\sum_{d=1}^{D-1} d \cdot \sum_{i \in V} u_i^d = n - 2 \quad (3.39)$$

$$u_i^d \in \{0, 1\} \quad i \in V, 1 \leq d \leq D - 1 \quad (3.23)$$

Este tipo de decomposição é em tudo semelhante à que foi feita na Secção 3.5.1, mas desta vez o subproblema de Saco Mochila é ligeiramente diferente devido à presença das variáveis discretizadas.

Também para este subproblema se pode aplicar a técnica de reformulação por caminhos, substituindo as restrições de Saco Mochila por restrições que modelem o problema de caminho associado à resolução do subproblema de Saco Mochila através de Programação Dinâmica. O grafo associado a esta representação pode ser reduzido até se tornar num grafo expandido idêntico ao grafo expandido para o Problema de Saco Mochila Limitado do modelo (SUP/SM_C) (o exemplo da Figura 3.9 serve de igual forma como exemplo de um grafo expandido deste novo problema). As variáveis de caminho, w_i^{st} mantêm o seu significado e portanto $w_i^{st} = 1$ implica que $u_i^{t-s} = 1$. Sendo assim, as restrições de ligação entre as variáveis u_i^d e as variáveis de caminho w_i^{st} são as restrições (3.38) apresentadas anteriormente e incluídas no modelo (SUP/SM_{CD2}).

Além disso, graças à presença destas últimas igualdades, a função objectivo (3.20) do modelo (SUP_D) pode ser reescrita como a função objectivo (3.36) do modelo (SUP/SM_{CD2}). Assim sendo, aplicando a técnica de reformulação por caminhos ao modelo (SUP_D) obtém-se o modelo (SUP/SM_{CD2}), já apresentado. No entanto, é fácil de provar através de um simples exemplo (ver Figura 3.15) que alguns dos pontos extremos do conjunto de soluções admissíveis da relaxação linear do problema de Saco Mochila definido pelas restrições (3.22), (3.39) e (3.23) são fraccionários.

Novamente, o facto de o sistema (3.34a) – (3.34d) satisfazer a propriedade da integralidade garante que a projecção do conjunto $\{U_i, w_i^{st} : (3.34a) - (3.34c), (3.38), (3.23) \text{ e } w_i^{st} \geq 0, \forall (i,s,t) \in A_E\}$ no espaço definido pelas variáveis discretizadas u_i^d é dada pelo envolvente convexo do conjunto $\{u_i^d : (3.22), (3.39) \text{ e } (3.23)\}$. Por outro lado, o exemplo anterior permite mostrar que este envolvente convexo está estritamente contido no conjunto $\{u_i^d : (3.22), (3.39) \text{ e } u_i^d \geq 0, \forall i \in V, 1 \leq d \leq D - 1\}$ e assim se conclui que:

$$V(LSUP_D) \leq V(LSUP/SM_{CD2})$$

□

Exemplo 3.4: Problema de Saco Mochila nas variáveis u_i^d

Na Figura 3.15 são apresentadas soluções para uma instância com 5 nodos, grau máximo igual a 3, capacidade unitária para cada módulo, custo da matriz de *routing* igual a 7 e custo por módulo igual a 3.

$$\mathbf{u}^I = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

(a) Solução inteira com custo 30.

$$\mathbf{u}^{II} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

(b) Solução inteira com custo 23.

$$\mathbf{u}^{III} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ \frac{1}{2} & 1 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{u}^{IV} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ \frac{3}{10} & \frac{3}{10} & \frac{3}{10} & \frac{3}{10} & \frac{3}{10} \end{bmatrix}$$

(c) Soluções fraccionárias com custo 19.5.

Figura 3.15: Pontos extremos fraccionários na relaxação linear do problema de Saco Mochila de Escolha Múltipla.

As soluções do problema são representadas através de matrizes¹⁵. Só existem dois tipos de soluções admissíveis inteiras que são representadas, à parte de uma troca de colunas, pelas matrizes das Figuras 3.15(a) e 3.15(b) (o valor ótimo 23 corresponde a matrizes do tipo \mathbf{u}^{II}). No entanto, o valor ótimo da relaxação linear do problema tem valor 19.5 e as matrizes da Figura 3.15(c) representam apenas algumas das soluções ótimas. Note-se que a solução fraccionária \mathbf{u}^{III} corresponde à mesma atribuição de graus aos nodos que a solução inteira \mathbf{u}^{II} .

Nos resultados apresentados a seguir, mostra-se que a dominância descrita na Proposição 3.5.3 é estrita para algumas instâncias.

Os modelos Árvore de Suporte/Saco Mochila, $(LSUP/SM_{CD1})$ e $(LSUP/SM_{CD2})$ ¹⁶, podem ser fortalecidos com a adição de versões das desigualdades válidas $AD(i, j; d)$ e $A_h(i, H(i))$, escritas em termos das variáveis de caminho, w_i^{st} .

Assim, as desigualdades *Arco Desagregadas*, $AD(i, j; d)$, são escritas como:

$$x_{ij}^d \leq \sum_{(i,s+s+d) \in A_E} w_i^{s,s+d} \quad (i, j) \in A; \quad d = 1, \dots, D - 1$$

¹⁵O elemento da linha d e coluna i representa o valor da variável u_i^d .

¹⁶Não se considera o modelo $(LSUP/SM_C)$, pois para as desigualdades $AD(i, j; d)$ é preciso primeiro discretizar as variáveis-nodo, U_i .

As desigualdades $Arco$, $A_h(i, H(i))$, são escritas como:

$$X(H(i)) \leq \sum_{\substack{(i,s,t) \in A_E \\ 1 \leq t-s \leq h-1}} (t-s) \cdot w_i^{st} + h \cdot \sum_{\substack{(i,s,t) \in A_E \\ h \leq t-s \leq D-1}} w_i^{st}, \quad H(i) \subseteq A^+(i), i \neq r$$

$$X(H(r)) \leq 1 + \sum_{\substack{(r,s,t) \in A_E \\ 1 \leq t-s \leq h-2}} (t-s) \cdot w_r^{st} + (h-1) \cdot \sum_{\substack{(r,s,t) \in A_E \\ h-1 \leq t-s \leq D-1}} w_r^{st}, \quad H(r) \subseteq A^+(r)$$

Estas novas versões podem ser obtidas a partir das versões originais utilizando para isso as restrições de ligação "desagregadas" (3.38) entre as variáveis u_i^d e w_i^{st} . Como tal, a introdução das desigualdades $AD(i, j; d)$ e $A_h(i, H(i))$ pode ser feita, quer em termos das variáveis u_i^d , quer em termos das variáveis w_i^{st} , apenas para o modelo ($LSUP/SM_{CD2}$), onde as restrições de ligação (3.38) estão presentes. Considere-se o modelo fortalecido, ($LSUP/SM_{CD2} + A_t$) obtido a partir do modelo ($LSUP/SM_{CD2}$) ao introduzir as desigualdades válidas $A_h(i, H(i))$, $i \in V$ para todo o valor $h \leq t$. De igual forma, o modelo fortalecido ($LSUP/SM_{CD2} + AD$) é obtido a partir do modelo ($LSUP/SM_{CD2}$) ao introduzir as desigualdades válidas $AD(i, j; d)$, $i \in V$.

Exemplo 3.5: Modelos Discretizados vs Modelos Árvore de Suporte/Saco Mochila.

Retomando a instância da Figura 3.6(a), apresentam-se na Tabela 3.5 os valores óptimos do modelo ($LSUP/SM_{CD2}$) e dos modelos fortalecidos associados, comparando-os com os correspondentes valores óptimos do modelo discretizado ($LSUP_D$) e dos modelos fortalecidos associados (*cf.* Figuras 3.6 e 3.7).

	$(LSUP_D)$	$(LSUP/SM_{CD2})$
sem Des. Válidas	234	264
$+A_1(i, H(i))$	297	303
$+A_t(i, H(i)), t = 1, 2$	297.75	304.167
$+AD(i, j; d)$	308	311.667

Tabela 3.5: Comparação entre modelos Discretizados e modelos Árvore de Suporte/Saco Mochila.

Para a instância apresentada, os modelos Árvore de Suporte/Saco Mochila fornecem limites inferiores melhores do que os obtidos com os correspondentes modelos Discretizados. O melhor limite inferior, obtido anteriormente com o modelo $(LSUP_D + AD)$ (17,4%), é agora reduzido para 16,4%, correspondendo à solução óptima do modelo $(LSUP/SM_{CD2} + AD)$.

3.5.4 Resultados Computacionais:

Modelo (SUP/SM_{CD2}) vs Modelo (SUP_D)

Nesta secção apresentam-se os resultados obtidos com a relaxação linear do modelo Árvore de Suporte/Saco Mochila, (SUP/SM_{CD2}) . Tendo em conta a redução pouco relevante dos desvios obtidos com o modelo $(LSUP_D + A_2)$ em relação ao modelo $(LSUP_D + A_1)$ (ver Secção 3.4.2) nesta secção apenas se considera o modelo fortalecido $(LSUP/SM_{CD2} + A_1)$ bem como o modelo $(LSUP/SM_{CD2} + AD)$.

Para comparação, apresentam-se de novo os resultados para a relaxação linear do modelo discretizado (SUP_D) e dos modelos fortalecidos ($SUP_D + A_1$) e ($SUP_D + AD$) associados. Os tempos de CPU para os modelos Árvore de Suporte/Saco Mochila com sistema de fluxos SC^* são mais elevados do que os tempos de CPU para os modelos discretizados com o mesmo sistema de fluxos, porém, são quase na totalidade inferiores a 10 segundos (*cf.* Apêndice D), razão pela qual também não se apresentam as respectivas colunas ¹⁷. Nas tabelas seguintes apresentam-se (na última coluna) os tempos de CPU gastos pelo modelo ($SUP/SM_C + A_1$) com o sistema de fluxos SC^* , para obter a solução ótima inteira. Foi este o modelo que melhor se comportou em termos de tempo de CPU gasto (ver Apêndice C).

Modelos Árvore de Suporte/Saco Mochila vs. Modelos Discretizados.

O resultado mais importante a salientar, nas instâncias com 25 nodos, onde mesmo com os modelos Discretizados fortalecidos ($LSUP_D + A_1$) e ($LSUP_D + AD$) se obtinham desvios ainda relativamente elevados. Consegue-se agora obter uma redução considerável nos desvios, com os modelos Árvore de Suporte/Saco Mochila (fortalecidos ou não). Com estes modelos consegue-se mesmo atingir o valor ótimo da solução inteira na maioria das instâncias *Wireless* com grau máximo $D = 3$. Para as instâncias com 50 nodos, o modelo ($LSUP/SM_{CD2}$) (fortalecido ou não) só consegue reduzir o desvio obtido com o correspondente modelo Discretizado para o valor de grau máximo, $D = 6$. É interessante notar que, quando se compara um modelo Discretizado com o correspondente modelo Árvore de Suporte/Saco Mochila (com o mesmo tipo de desigualdades válidas adicionadas), a redução no desvio é praticamente a mesma, quer se use o sistema de fluxos SC^* ou MC .

¹⁷Para as instâncias com 25 nodos estes tempos são mesmo inferiores a 2 segundos.

T	$ E $	(K_1, K_2)	D	$(LSUP_D)$			$(LSUP/SM_{CD2})$			$+A_1(i, H(i))$			$+AD(i, j; d)$			Int						
				desvio		CPU	desvio		CPU	desvio		CPU	desvio		CPU		CPU					
				SC^*	MC	MC	SC^*	MC	MC	SC^*	MC	MC	SC^*	MC	MC			SC^*	MC	MC		
W	(100, 10)	3	3,9	3,7	0	0,2	0*	1	3,9	3,7	0	0,2	0*	1	3,9	3,7	0	0,2	0*	0	1	
		4	3,7	3,4	0	1,0	0,7	0	3,2	3,0	0	0,6	0,4	1	3,0	2,9	0	0,5	0,3	1	1	
		5	5,9	5,6	0	2,6	2,2	0	4,3	4,2	0	1,2	1,1	3	4,0	3,8	1	0,9	0,8	2	2	
		6	19,5	19,2	0	14,0	13,7	1	15,5	15,4	0	11,7	11,7	2	15,3	15,2	1	11,4	11,4	3	23	
	(100, 50)	3	4,0	3,8	0	0,2	0*	0	4,0	3,8	0	0,2	0*	1	4,0	3,8	0	0,2	0*	0	2	
		4	1,6	1,4	0	0,6	0,5	1	1,3	1,2	0	0,4	0,3	1	1,2	1,1	0	0,3	0,2	1	1	
		5	5,3	5,1	0	1,7	1,5	0	4,3	4,2	0	0,8	0,7	2	4,1	3,9	1	0,6	0,5	2	3	
		6	15,9	15,7	0	9,6	9,4	1	13,0	13,0	0	8,0	8,0	3	12,6	12,5	1	7,8	7,8	5	31	
	C	(100, 10)	3	3,8	3,8	1	0*	0*	1	3,8	3,8	1	0*	0*	2	3,8	3,8	1	0*	0*	1	3
			4	3,5	3,5	0	0,7	0,7	1	3,2	3,1	1	0,6	0,5	4	3,2	3,1	2	0,6	0,5	2	8
			5	4,9	4,9	0	1,4	1,4	1	3,8	3,6	1	0,5	0,4	5	3,8	3,6	2	0,5	0,4	5	3
			6	9,0	9,0	0	2,7	2,7	1	6,9	6,8	1	1,0	0,8	7	6,9	6,8	2	1,0	0,8	7	9
(100, 50)		3	3,9	3,9	1	0*	0*	1	3,9	3,9	1	0*	0*	1	3,9	3,9	1	0*	0*	1	15	
		4	1,5	1,5	0	0,5	0,5	1	1,3	1,2	1	0,4	0,3	4	1,3	1,2	2	0,4	0,3	3	10	
		5	4,6	4,6	0	0,9	0,9	1	3,9	3,8	1	0,3	0,3	4	3,9	3,8	2	0,3	0,3	4	3	
		6	8,7	8,7	0	1,7	1,7	1	7,3	7,2	1	0,6	0,6	7	7,3	7,2	2	0,6	0,6	9	12	
T		(100, 10)	3	6,2	5,7	0	4,0	3,5	0	3,0	2,8	0	2,0	1,7	0	3,0	2,8	0	2,0	1,6	0	1
			4	11,1	10,6	0	9,7	9,3	0	2,8	2,1	0	2,7	2,1	1	2,8	2,1	0	2,6	2,0	1	1
			5	19,9	19,4	0	18,5	18,0	0	6,9	6,1	0	6,7	5,9	0	6,6	5,8	0	6,6	5,8	1	14
			6	24,4	23,9	0	22,2	21,7	0	6,9	6,2	0	6,8	6,1	1	6,0	5,3	0	6,0	5,3	1	12
	(100, 50)	3	5,8	5,5	0	3,4	3,0	0	3,2	3,0	0	1,7	1,4	0	3,2	3,0	0	1,7	1,3	0	0	
		4	8,3	7,9	0	7,7	7,3	0	2,3	1,9	0	2,3	1,9	1	2,1	1,8	0	2,1	1,8	0	1	
		5	17,1	16,7	0	15,2	14,8	0	6,4	5,8	1	5,5	4,9	1	6,1	5,5	0	5,4	4,8	1	19	
		6	22,4	22,0	0	19,4	18,9	0	6,9	6,5	1	6,5	5,9	1	4,7	4,3	1	4,7	4,3	1	18	
	(100, 10)	3	5,9	5,3	0	3,5	3,0	1	4,1	3,4	1	2,4	1,7	1	4,1	3,4	1	2,4	1,7	0	6	
		4	9,6	9,3	0	8,2	7,8	0	3,8	3,0	1	3,7	2,8	1	3,8	3,0	1	3,7	2,8	1	8	
		5	14,3	13,9	0	12,6	12,3	0	3,0	2,1	1	2,8	1,9	2	2,9	2,1	2	2,8	1,9	2	5	
		6	20,4	20,0	0	17,7	17,4	0	4,4	3,6	1	4,1	3,3	2	4,2	3,3	2	4,1	3,2	4	19	
(100, 50)	3	5,6	5,1	0	2,9	2,5	1	4,1	3,5	0	2,0	1,4	1	4,1	3,5	1	2,0	1,4	1	6		
	4	6,9	6,7	0	6,3	6,1	1	2,9	2,2	1	2,9	2,2	1	2,9	2,2	1	2,9	2,2	1	20		
	5	12,2	11,9	0	10,1	9,8	0	3,3	2,6	1	2,2	1,5	3	3,2	2,5	1	2,2	1,5	2	7		
	6	18,1	17,9	0	14,6	14,3	0	4,8	4,1	1	3,4	2,7	3	4,2	3,5	2	3,4	2,6	4	17		

Tabela 3.6: Resultados para instâncias com 25 nodos: Modelos (SUP/SM_{CD2}) vs Modelos (SUP_D) .

Em termos de tempo de CPU, os modelos Discretizados com o sistema MC são mais rápidos a obter um limite inferior para o problema do que os correspondentes modelos Árvore de Suporte/Saco Mochila; estes últimos têm um maior número de variáveis e restrições, associadas ao sistema de caminhos (3.34a) – (3.34d), número esse que aumenta com o valor do parâmetro D .

T	$ E $	(K_1, K_2)	D	$+A_1(i, H(i))$									$+AD(i, j; d)$									Int CPU SC*	
				$(LSUP_D)$			$(LSUP/SM_{CD2})$			$(LSUP_D)$			$(LSUP/SM_{CD2})$			$(LSUP_D)$			$(LSUP/SM_{CD2})$				
				desvio	CPU		desvio	CPU		desvio	CPU		desvio	CPU		desvio	CPU		desvio	CPU			
SC*	MC	MC	SC*	MC	MC	SC*	MC	MC	SC*	MC	MC	SC*	MC	MC	SC*	MC	MC						
W	(100, 10)	3	3	0,1	0*	9	0,1	0*	15	0,1	0*	9	0,1	0*	20	0,1	0*	10	0,1	0*	15	4	
		4	4	0,5	0,3	1	0,5	0,3	5	0,2	0,2	2	0,2	0,2	13	0,2	0,2	4	0,2	0,2	11	6	
		5	5	1,8	1,6	1	1,8	1,6	3	0,7	0,7	2	0,7	0,7	30	0,6	0,6	5	0,6	0,6	23	6	
		6	6	6,2	6,0	1	3,3	3,1	109	3,2	3,2	2	1,5	1,5	67	3,0	3,0	13	1,3	1,3	44	15	
	(100, 50)	3	3	0,1	0*	9	0,1	0*	16	0,1	0*	9	0,1	0*	16	0,1	0*	22	0,1	0*	15	3	
		4	4	0,3	0,2	2	0,3	0,2	4	0,2	0,1	2	0,2	0,1	13	0,1	0,1	5	0,1	0,1	13	6	
		5	5	1,2	1,1	1	1,2	1,1	3	0,5	0,5	3	0,5	0,5	36	0,4	0,4	7	0,4	0,4	26	6	
		6	6	5,5	5,4	1	2,2	2,1	96	3,4	3,4	3	1,0	1,0	80	3,2	3,2	12	0,9	0,9	59	14	
	C	(100, 10)	3	3	0,1	0,1	27	0,1	0,1	61	0,1	0,1	36	0,1	0,1	61	0,1	0,1	48	0,1	0,1	46	46
			4	4	1,1	1,0	2	1,1	1,0	25	0,1	0,1	16	0,1	0,1	89	0,1	0,1	17	0,1	0,1	39	8
			5	5	2,4	2,3	2	2,4	2,3	4	0,2	0,2	9	0,2	0,2	128	0,2	0,2	21	0,2	0,2	57	43
			6	6	6,7	4,0	1	3,6	3,4	183	3,2	3,1	14	0,4	0,3	283	3,2	3,1	35	0,4	0,3	252	80
(100, 50)		3	3	0,1	0,1	23	0,1	0,1	57	0,1	0,1	35	0,1	0,1	62	0,1	0,1	38	0,1	0,1	48	21	
		4	4	0,7	0,6	2	0,7	0,6	25	0,1	0,1	20	0,1	0,1	66	0,1	0,1	33	0,1	0,1	50	7	
		5	5	1,6	1,5	2	1,6	1,5	14	0,2	0,2	9	0,2	0,2	75	0,2	0,2	16	0,2	0,2	81	21	
		6	6	5,8	4,0	1	2,4	2,3	130	3,5	3,4	22	0,2	0,2	216	3,5	3,4	35	0,2	0,2	267	137	
C		(100, 10)	3	3	2,8	2,7	1	2,8	2,7	1	0,7	0,5	2	0,7	0,5	7	0,7	0,5	2	0,7	0,5	5	2
			4	4	7,6	7,6	1	7,6	7,6	3	0,9	0,8	2	0,9	0,8	7	0,5	0,4	2	0,5	0,4	5	8
			5	5	13,2	12,9	1	13,2	12,9	2	2,3	2,2	2	2,3	2,2	4	1,3	1,2	5	1,3	1,2	8	30
			6	6	16,4	16,2	1	15,6	15,4	2	2,1	2,1	1	2,1	2,1	5	1,0	0,9	2	1,0	0,9	10	17
	(100, 50)	3	3	2,4	2,4	1	2,4	2,4	2	0,6	0,5	3	0,6	0,5	6	0,6	0,5	3	0,6	0,5	3	2	
		4	4	5,7	5,7	1	5,7	5,7	2	0,9	0,8	2	0,9	0,8	10	0,6	0,6	3	0,6	0,6	5	11	
		5	5	12,1	11,9	1	12,1	11,9	2	2,8	2,7	2	2,8	2,7	6	1,9	1,8	3	1,9	1,8	7	47	
		6	6	15,9	15,8	1	14,8	14,6	2	2,8	2,8	1	2,8	2,8	7	1,2	1,1	4	1,2	1,1	13	18	
	C	(100, 10)	3	3	5,8	5,7	2	5,8	5,7	4	2,3	1,8	7	2,3	1,8	9	2,1	1,7	11	2,1	1,7	14	255
			4	4	12,3	12,2	2	12,3	12,2	3	3,0	2,6	7	3,0	2,6	22	3,0	2,6	8	3,0	2,6	18	327
			5	5	18,3	18,1	2	18,3	18,1	4	2,3	1,9	9	2,3	1,9	19	1,8	1,5	15	1,8	1,5	31	141
			6	6	24,4	24,2	2	23,3	23,1	6	3,4	3,1	8	3,4	3,1	23	2,4	2,2	18	2,4	2,2	69	902
(100, 50)		3	3	4,8	4,8	2	4,8	4,8	3	2,0	1,5	6	2,0	1,5	21	1,8	1,4	9	1,8	1,4	13	323	
		4	4	8,1	8,0	2	8,1	8,0	3	2,2	1,8	6	2,2	1,8	22	2,2	1,8	7	2,2	1,8	12	317	
		5	5	16,2	16,0	2	16,2	16,0	4	3,2	3,0	8	3,2	3,0	31	2,1	2,0	14	2,1	2,0	38	1706	
		6	6	23,0	22,8	2	21,5	21,3	6	4,7	4,4	8	4,7	4,4	32	2,1	2,1	19	2,1	2,1	45	10093	

Tabela 3.7: Resultados para instâncias com 50 nodos: Modelos (SUP/SM_{CD2}) vs Modelos (SUP_D) .

Efeitos das desigualdades válidas no modelo $(LSUP/SM_{CD2})$.

Nas instâncias *Cabo*, onde o modelo $(LSUP/SM_{CD2})$ atinge os desvios mais elevados, a introdução de desigualdades válidas *Arco* ou *Arco Desagregadas* neste modelo provoca uma maior redução nos desvios do que nas instâncias *Wireless*.

Nas redes *Cabo*, o modelo ($LSUP/SM_{CD2} + AD$) consegue ainda melhorar os desvios obtidos com o modelo ($LSUP/SM_{CD2} + A_1$), de uma forma mais relevante.

Obtenção da solução ótima inteira.

Nas instâncias com 25 nodos o tempo gasto para obter a solução ótima inteira foi relativamente baixo. Quanto às instâncias de 50 nodos, embora os tempos sejam maiores nas redes *Cabo* do que nas redes *Wireless*, apenas uma excedeu o tempo limite de 2 horas.

3.6 Desigualdades de *Arredondamento*

Aquando da formulação dos modelos Árvore de Suporte/Saco Mochila foi necessário incluir no modelo a igualdade (3.39) (ou a igualdade (3.33) na versão escrita com as variáveis inteiras). Esta igualdade, redundante no modelo, permite construir desigualdades válidas escritas apenas com recurso às variáveis discretizadas. O modo de as obter consiste em dividir ambos os membros da igualdade (3.39) por um número inteiro entre 2 e $D - 1$ obtendo-se assim um conjunto de igualdades equivalentes:

$$\sum_{d=1}^{D-1} \frac{d}{p} \cdot \sum_{i \in V} u_i^d = \frac{n-2}{p} \quad p = 2, \dots, D-1$$

O processo de obtenção das desigualdades válidas consiste em arredondar "para baixo" o coeficiente do primeiro membro de cada uma destas últimas igualdades e de seguida arredondar "para baixo" o termo do segundo membro (o que pode ser feito dada a natureza inteira das variáveis u_i^d).

Obtém-se assim o seguinte conjunto de *desigualdades de Arredondamento para Baixo*¹⁸:

$$B_p \quad \sum_{i \in V} \sum_{d=p}^{D-1} \left\lfloor \frac{d}{p} \right\rfloor \cdot u_i^d \leq \left\lfloor \frac{n-2}{p} \right\rfloor \quad p = 2, \dots, D-1$$

Utilizando um raciocínio semelhante mas desta vez arredondando "para cima" cada coeficiente do primeiro membro e de seguida arredondando "para cima" o termo do segundo membro nas anteriores igualdades, obtém-se o seguinte conjunto de *desigualdades de Arredondamento para Cima*:

$$C_p \quad \sum_{i \in V} \sum_{d=1}^{D-1} \left\lceil \frac{d}{p} \right\rceil \cdot u_i^d \geq \left\lceil \frac{n-2}{p} \right\rceil \quad p = 2, \dots, D-1$$

Este tipo de desigualdades foi anteriormente utilizado em trabalhos onde a técnica de discretização foi abordada, nomeadamente aplicada ao problema de *Localização de Concentradores com Capacidade* (ver [33]) e ao problema de *Empacotamento com Dimensões Variáveis* (ver [9]).

Casos Particulares:

1. A desigualdade válida B_{D-1} :

$$\sum_{i \in V} u_i^{D-1} \leq \left\lfloor \frac{n-2}{D-1} \right\rfloor$$

impõe um limite máximo ao número de nodos na solução com o maior grau permitido, D .

¹⁸ $\left\lfloor \frac{d}{p} \right\rfloor = 0, \forall d < p$.

2. A desigualdade válida C_{D-1} :

$$\sum_{i \in V} \sum_{d=1}^{D-1} u_i^d \geq \left\lceil \frac{n-2}{D-1} \right\rceil$$

impõe um limite mínimo ao número de nodos não-folha na solução, ou seja, um limite mínimo ao número de nodos onde será instalado equipamento.

O processo descrito para a obtenção das desigualdades B_p e C_p , a partir da igualdade redundante (3.39), prova que são válidas para qualquer um dos modelos apresentados nas secções anteriores (desde que incluam as variáveis discretizadas u_i^d).

Embora o número de *Desigualdades de Arredondamento* a introduzir no modelo seja em número polinomial ($D - 2$ para cada conjunto) pode ainda ser reduzido. Por um lado, para qualquer número de nodos as desigualdades B_2 e C_2 são equivalentes (ver Resultado 2 do Apêndice A). Por outro lado, em cada conjunto de desigualdades, B_p ou C_p , algumas delas poderão ter o mesmo segundo membro e como tal a mais fraca pode ser eliminada. Tome-se como exemplo o caso em que $n - 2 = 9$ e $D - 1 = 4$. A desigualdade C_3 ,

$$\sum_{i \in V} (u_i^1 + u_i^2 + u_i^3 + 2 \cdot u_i^4) \geq \left\lceil \frac{9}{3} \right\rceil = 3$$

é implicada pela desigualdade C_4 ,

$$\sum_{i \in V} (u_i^1 + u_i^2 + u_i^3 + u_i^4) \geq \left\lceil \frac{9}{4} \right\rceil = 3$$

O resultado seguinte mostra que estas desigualdades de *Arredondamento* são ainda implicadas pelo modelo (*SUP/SM_{CD2}*) definido na Secção 3.5.2.

Proposição 3.6.1. *As desigualdades B_p e C_p , $p = 1, \dots, D - 1$, são implicadas pelo modelo (*SUP/SM_{CD2}*).*

Demonstração. Considere-se o conjunto *SM/Cam₂* (ver Secção 3.5.2) definido pelas restrições de caminho nas variáveis w_i^{st} , (3.34a), (3.34b), (3.34c), (3.34d), pelas restrições de domínio das variáveis u_i^d , (3.23) e pelas restrições de ligação, (3.38). Como já foi dito antes, o sistema (3.34a) – (3.34d) goza da propriedade da integralidade, que não é alterada pela introdução das restrições de ligação (3.38). Assim, qualquer ponto extremo do seguinte poliedro (designado por *LSM/Cam₂*) tem todas as coordenadas inteiras:

$$\begin{aligned}
 W(A_E^+(0_0)) &= 1 \\
 W(A_E^-(i_s)) &= W(A_E^+(i_s)) && i_s \in V_E : i = 1, \dots, n - 1 \\
 W(A_E^-(n_{n-2})) &= 1 \\
 u_i^d &= \sum_{(i_s, s+d) \in A_E} w_i^{s, s+d} && i \in V, d = 1, \dots, D - 1 \\
 0 \leq w_i^{st} &\leq 1 && (i_s, t) \in A_E \\
 0 \leq u_i^d &\leq 1 && i \in V; d = 1, \dots, D - 1
 \end{aligned}$$

Nomeadamente, todas as coordenadas associadas às variáveis u_i^d são inteiras e como tal verificam as desigualdades de *Arredondamento*. Assim sendo, a projecção deste poliedro no espaço das variáveis u_i^d está contida no envolvente convexo definido pelas soluções inteiras do seguinte conjunto:

$$\begin{aligned}
\sum_{d=1}^{D-1} d \cdot \sum_{i \in V} u_i^d &= n - 2 \\
\sum_{d=1}^{D-1} u_i^d &\leq 1 && i \in V \\
\sum_{i \in V} \sum_{d=p}^{D-1} \left\lfloor \frac{d}{p} \right\rfloor \cdot u_i^d &\leq \left\lfloor \frac{n-2}{p} \right\rfloor && p = 2, \dots, D-1 \\
\sum_{i \in V} \sum_{d=1}^{D-1} \left\lceil \frac{d}{p} \right\rceil \cdot u_i^d &\geq \left\lceil \frac{n-2}{p} \right\rceil && p = 2, \dots, D-1 \\
0 \leq u_i^d &\leq 1 && i \in V; d = 1, \dots, D-1
\end{aligned}$$

□

A partir dum modelo genérico (P), obtém-se o modelo fortalecido ($P + BC$) ao introduzir as desigualdades de *Arredondamento*, B_p e C_p , $p = 2, \dots, D-1$. No exemplo apresentado anteriormente na Figura 3.6, a introdução destas desigualdades permite melhorar os limites inferiores para o problema, obtidos com os modelos discretizados. Comparam-se também estes resultados com os resultados obtidos com o modelo ($LSUP/SM_{CD2}$) e respectivos modelos fortalecidos ilustrando assim o resultado enunciado na Proposição 3.6.1.

Exemplo 3.6: Modelos Discretizados com desigualdades de *Arredondamento vs Modelos Árvore de Suporte/Saco Mochila*.

Retomando a instância da Figura 3.6(a) compara-se a introdução das desigualdades de arredondamento B_p e $C_p \forall p$, no modelo discretizado, (SUP_D) e nos modelos fortalecidos com as desigualdades *Arco* e *Arco Desagregadas*.

	$(LSUP_D)$	$(LSUP_D+BC)$	$(LSUP/SM_{CD2})$
sem Desigualdades Válidas	234	264	264
+ $A_1(i, H(i))$	297	302.5	303
+ $A_t(i, H(i)), t = 1, 2$	297.75	303	304.167
+ $AD(i, j; d)$	308	311	311.667

Tabela 3.8: Comparação entre modelos fortalecidos com desigualdades de *Arredondamento* e modelos Árvore de Suporte/Saco Mochila.

Como se pode observar, os limites inferiores obtidos com o modelo $(LSUP_D)$ são melhorados ao introduzir as desigualdades de *Arredondamento* no modelo. Isto verifica-se, quer para os modelos sem desigualdades válidas, quer para os modelos fortalecidos com as desigualdades *Arco* ou *Arco Desagregadas*. Estes limites por sua vez são inferiores aos obtidos com o modelo $(LSUP/SM_{CD2})$, o que permite concluir que, mesmo com a introdução das desigualdades de *Arredondamento* no modelo discretizado, o resultado da Proposição 3.5.3 pode ser verificado no sentido estrito.

3.6.1 Resultados Computacionais:

Modelo discretizados fortalecidos com desigualdades de *Arredondamento*

Nesta última secção de resultados computacionais para o problema (ASupCG) apresentam-se os resultados obtidos com a relaxação linear dos modelos discretizados (SUP_D), ($SUP_D + A_1$), e ($SUP_D + AD$) fortalecidos com as desigualdades de *Arredondamento* apresentadas na secção anterior. Não se fez distinção entre a adição de um dos dois tipos de desigualdades, optando-se pela adição conjunta dos dois tipos de desigualdades de *Arredondamento*.

A adição das desigualdades de *Arredondamento* nos modelos discretizados (fortalecidos ou não com as desigualdades *Arco* ou *Arco Desagregadas*) resulta, em quase todos os casos, de uma forma tão eficaz como a utilização dos modelos Árvore de Suporte/Saco Mochila. A única excepção regista-se nos casos em que o grau máximo nos nodos é $D = 6$, onde os modelos Árvore de Suporte/Saco Mochila dão origem a desvios mais baixos (*cf.* Tabelas 3.6 e 3.7). Estas diferenças são maiores no caso em que os custos tecnológicos são mais caros, $(K_1, K_2) = (100, 50)$.

Além disso, os tempos de CPU do modelo ($LSUP_D + BC$) que utiliza o sistema de fluxos *MC*, são mais baixos do que os correspondentes tempos de CPU do modelo ($LSUP/SM_{CD2}$). E isto verifica-se quer se adicionem ou não as desigualdades *Arco* ou *Arco Desagregadas* ao modelo ($LSUP_D + BC$).

R	$ E $	(K_1, K_2)	D	$+A_1(i, H(i))$												$+AD(i, j; d)$					
				$(LSUP_D)$			$(LSUP_D+BC)$			$(LSUP_D)$			$(LSUP_D+BC)$			$(LSUP_D)$			$(LSUP_D+BC)$		
				desvio	CPU		desvio	CPU		desvio	CPU		desvio	CPU		desvio	CPU		desvio	CPU	
SC^*	MC	MC	SC^*	MC	MC	SC^*	MC	MC	SC^*	MC	MC	SC^*	MC	MC	SC^*	MC	MC	SC^*	MC	MC	
W	(100, 10)	3	3,9	3,7	0	0,2	0*	0	3,9	3,7	0	0,2	0*	0	3,9	3,7	0	0,2	0*	0	
		4	3,7	3,4	0	1,0	0,7	0	3,2	3,0	0	0,6	0,4	0	3,0	2,9	0	0,5	0,3	1	
		5	5,9	5,6	0	2,6	2,2	0	4,3	4,2	0	1,2	1,1	0	4,0	3,8	1	0,9	0,8	1	
		6	19,5	19,2	0	14,6	14,2	0	15,5	15,4	0	12,3	12,3	0	15,3	15,2	1	11,9	11,8	1	
	(100, 50)	3	4,0	3,8	0	0,2	0*	0	4,0	3,8	0	0,2	0*	0	4,0	3,8	0	0,2	0*	0	
		4	1,6	1,4	0	0,6	0,5	0	1,3	1,2	0	0,4	0,3	0	1,2	1,1	0	0,3	0,2	0	
		5	5,3	5,1	0	1,7	1,5	0	4,3	4,2	0	0,8	0,7	0	4,1	3,9	1	0,6	0,5	1	
		6	15,9	15,7	0	11,6	11,3	0	13,0	13,0	0	10,0	10,0	0	12,6	12,5	1	9,7	9,7	1	
	150	(100, 10)	3	3,8	3,8	1	0*	0*	0	3,8	3,8	1	0*	0*	1	3,8	3,8	1	0*	0*	1
			4	3,5	3,5	0	0,7	0,7	0	3,2	3,1	1	0,6	0,5	1	3,2	3,1	2	0,6	0,5	2
			5	4,9	4,9	0	1,4	1,4	0	3,8	3,6	1	0,5	0,4	1	3,8	3,6	2	0,5	0,4	2
			6	9,0	9,0	0	3,3	3,3	0	6,9	6,8	1	1,6	1,5	1	6,9	6,8	2	1,6	1,5	4
(100, 50)	3	3,9	3,9	1	0*	0*	0	3,9	3,9	1	0*	0*	1	3,9	3,9	1	0*	0*	1		
	4	1,5	1,5	0	0,5	0,5	0	1,3	1,2	1	0,4	0,3	1	1,3	1,2	2	0,4	0,3	1		
	5	4,6	4,6	0	0,9	0,9	0	3,9	3,8	1	0,3	0,3	1	3,9	3,8	2	0,3	0,3	2		
	6	8,7	8,7	0	3,9	3,9	0	7,3	7,2	1	2,8	2,7	1	7,3	7,2	2	2,8	2,7	3		
C	(100, 10)	3	6,2	5,7	0	4,0	3,5	0	3,0	2,8	0	2,0	1,7	0	3,0	2,8	0	2,0	1,6	0	
		4	11,1	10,6	0	9,7	9,3	0	2,8	2,1	0	2,7	2,1	0	2,8	2,1	0	2,6	2,0	0	
		5	19,9	19,4	0	18,5	18,0	0	6,9	6,1	0	6,7	5,9	0	6,6	5,8	0	6,6	5,8	1	
		6	24,4	23,9	0	22,4	21,9	0	6,9	6,2	0	6,9	6,2	0	6,0	5,3	0	6,0	5,3	0	
	(100, 50)	3	5,8	5,5	0	3,4	3,0	0	3,2	3,0	0	1,7	1,4	0	3,2	3,0	0	1,7	1,3	0	
		4	8,3	7,9	0	7,7	7,3	0	2,3	1,9	0	2,3	1,9	0	2,1	1,8	0	2,1	1,8	0	
		5	17,1	16,7	0	15,2	14,8	0	6,4	5,8	1	5,5	4,9	0	6,1	5,5	0	5,4	4,8	0	
		6	22,4	22,0	0	20,3	19,9	0	6,9	6,5	1	6,9	6,5	0	4,7	4,3	1	4,7	4,3	1	
	150	(100, 10)	3	5,9	5,3	0	3,5	3,0	0	4,1	3,4	1	2,4	1,7	0	4,1	3,4	1	2,4	1,7	0
			4	9,6	9,3	0	8,2	7,8	0	3,8	3,0	1	3,7	2,8	1	3,8	3,0	1	3,7	2,8	1
			5	14,3	13,9	0	12,6	12,3	0	3,0	2,1	1	2,8	1,9	1	2,9	2,1	2	2,8	1,9	1
			6	20,4	20,0	0	18,0	17,7	0	4,4	3,6	1	4,4	3,6	1	4,2	3,3	2	4,2	3,3	2
(100, 50)	3	5,6	5,1	0	2,9	2,5	0	4,1	3,5	0	2,0	1,4	0	4,1	3,5	1	2,0	1,4	1		
	4	6,9	6,7	0	6,3	6,1	0	2,9	2,2	1	2,9	2,2	1	2,9	2,2	1	2,9	2,2	1		
	5	12,2	11,9	0	10,1	9,8	0	3,3	2,6	1	2,2	1,5	1	3,2	2,5	1	2,2	1,5	1		
	6	18,1	17,9	0	15,7	15,4	0	4,8	4,1	1	4,5	3,8	1	4,2	3,5	2	4,2	3,4	2		

Tabela 3.9: Resultados para instâncias com 25 nodos: Modelos fortalecidos com desigualdades de *Arredondamento*.

R	E	(K ₁ , K ₂)	D	+A ₁ (i, H(i))												+AD(i, j; d)					
				(LSUP _D)			(LSUP _D +BC)			(LSUP _D)			(LSUP _D +BC)			(LSUP _D)			(LSUP _D +BC)		
				desvio	CPU		desvio	CPU		desvio	CPU		desvio	CPU		desvio	CPU		desvio	CPU	
SC*	MC	MC	SC*	MC	MC	SC*	MC	MC	SC*	MC	MC	SC*	MC	MC	SC*	MC	MC	SC*	MC	MC	
W	(100, 10)	3	0,1	0*	9	0,1	0*	9	0,1	0*	9	0,1	0*	13	0,1	0*	10	0,1	0*	8	
		4	0,5	0,3	1	0,5	0,3	3	0,2	0,2	2	0,2	0,2	2	0,2	0,2	4	0,2	0,2	16	
		5	1,8	1,6	1	1,8	1,6	1	0,7	0,7	2	0,7	0,7	3	0,6	0,6	5	0,6	0,6	9	
		6	6,2	6,0	1	3,6	3,4	1	3,2	3,2	2	1,8	1,8	2	3,0	3,0	13	1,4	1,4	12	
	(100, 50)	3	0,1	0*	9	0,1	0*	9	0,1	0*	9	0,1	0*	13	0,1	0*	22	0,1	0*	8	
		4	0,3	0,2	2	0,3	0,2	2	0,2	0,1	2	0,2	0,1	5	0,1	0,1	5	0,1	0,1	10	
		5	1,2	1,1	1	1,2	1,1	1	0,5	0,5	3	0,5	0,5	2	0,4	0,4	7	0,4	0,4	9	
		6	5,5	5,4	1	3,2	3,1	0	3,4	3,4	3	2,0	2,0	3	3,2	3,2	12	1,9	1,8	10	
	300	(100, 10)	3	0,1	0,1	27	0,1	0,1	27	0,1	0,1	36	0,1	0,1	49	0,1	0,1	48	0,1	0,1	88
			4	1,1	1,0	2	1,1	1,0	3	0,1	0,1	16	0,1	0,1	20	0,1	0,1	17	0,1	0,1	34
			5	2,4	2,3	2	2,4	2,3	2	0,2	0,2	9	0,2	0,2	10	0,2	0,2	21	0,2	0,2	31
			6	6,7	4,0	1	3,9	3,8	2	3,2	3,1	14	0,7	0,6	34	3,2	3,1	35	0,7	0,6	41
(100, 50)		3	0,1	0,1	23	0,1	0,1	40	0,1	0,1	35	0,1	0,1	54	0,1	0,1	38	0,1	0,1	69	
		4	0,7	0,6	2	0,7	0,6	4	0,1	0,1	20	0,1	0,1	18	0,1	0,1	33	0,1	0,1	31	
		5	1,6	1,5	2	1,6	1,5	2	0,2	0,2	9	0,2	0,2	14	0,2	0,2	16	0,2	0,2	23	
		6	5,8	4,0	1	3,4	3,3	1	3,5	3,4	22	1,3	1,3	14	3,5	3,4	35	1,3	1,3	39	
C	(100, 10)	3	2,8	2,7	1	2,8	2,7	2	0,7	0,5	2	0,7	0,5	3	0,7	0,5	2	0,7	0,5	4	
		4	7,6	7,6	1	7,6	7,6	1	0,9	0,8	2	0,9	0,8	2	0,5	0,4	2	0,5	0,4	5	
		5	13,2	12,9	1	13,2	12,9	1	2,3	2,2	2	2,3	2,2	2	1,3	1,2	5	1,3	1,2	3	
		6	16,4	16,2	1	15,7	15,5	0	2,1	2,1	1	2,1	2,1	1	1,0	0,9	2	1,0	0,9	3	
	(100, 50)	3	2,4	2,4	1	2,4	2,4	2	0,6	0,5	3	0,6	0,5	4	0,6	0,5	3	0,6	0,5	3	
		4	5,7	5,7	1	5,7	5,7	1	0,9	0,8	2	0,9	0,8	2	0,6	0,6	3	0,6	0,6	4	
		5	12,1	11,9	1	12,1	11,9	1	2,8	2,7	2	2,8	2,7	1	1,9	1,8	3	1,9	1,8	4	
		6	15,9	15,8	1	15,1	15,0	0	2,8	2,8	1	2,8	2,8	1	1,2	1,1	4	1,2	1,1	5	
	300	(100, 10)	3	5,8	5,7	2	5,8	5,7	2	2,3	1,8	7	2,3	1,8	8	2,1	1,7	11	2,1	1,7	7
			4	12,3	12,2	2	12,3	12,2	2	3,0	2,6	7	3,0	2,6	5	3,0	2,6	8	3,0	2,6	10
			5	18,3	18,1	2	18,3	18,1	2	2,3	1,9	9	2,3	1,9	8	1,8	1,5	15	1,8	1,5	16
			6	24,4	24,2	2	23,4	23,2	2	3,4	3,1	8	3,4	3,1	7	2,4	2,2	18	2,4	2,2	20
(100, 50)		3	4,8	4,8	2	4,8	4,8	2	2,0	1,5	6	2,0	1,5	5	1,8	1,4	9	1,8	1,4	10	
		4	8,1	8,0	2	8,1	8,0	1	2,2	1,8	6	2,2	1,8	6	2,2	1,8	7	2,2	1,8	7	
		5	16,2	16,0	2	16,2	16,0	2	3,2	3,0	8	3,2	3,0	7	2,1	2,0	14	2,1	2,0	12	
		6	23,0	22,8	2	21,9	21,8	2	4,7	4,4	8	4,7	4,4	8	2,1	2,1	19	2,1	2,1	19	

Tabela 3.10: Resultados para instâncias com 50 nodos: Modelos fortalecidos com desigualdades de *Arredondamento*.

3.7 Conclusões

Com base nos resultados teóricos apresentados é possível estabelecer relações (ver Figura 3.16) entre todos os modelos apresentados neste capítulo, em termos das respectivas relaxações lineares. Os resultados computacionais mostraram que muitas das relações de dominância descritas na figura podem verificar-se mesmo em sentido estrito.

Os modelos testados tiveram melhores resultados nas redes do tipo *Wireless*, onde os desvios obtidos com as respectivas relaxações lineares foram mais baixos do que nas instâncias *Cabo*. Foi nestas instâncias que a relaxação linear do melhor modelo, (SUP/SM_{CD2+AD}) , conseguiu as maiores reduções aos desvios obtidos com a relaxação linear do pior modelo (SUP_D) .

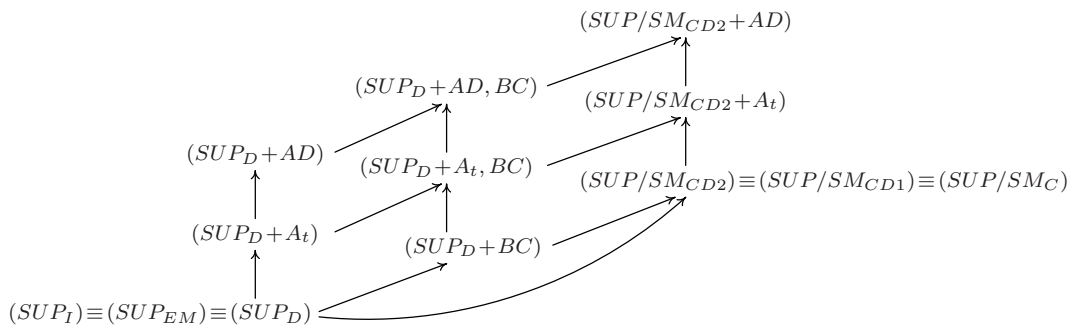


Figura 3.16: Comparação dos modelos apresentados. $A \rightarrow B$ ($A \equiv B$) significa que o modelo B é mais forte do que (equivalente a) o modelo A em termos das respectivas relaxações lineares.

Capítulo 4

O problema (AStRPCG)

Os modelos apresentados neste capítulo, para o (AStRPCG) são baseados no modelo genérico apresentado na Secção 1.1.3 para o (PASTRP). Na Secção 4.1 introduz-se neste modelo genérico, a estrutura referente ao grau de cada nodo na solução e a respectiva função de custos, obtendo-se assim, um modelo não linear que servirá de base aos modelos seguintes. Na Secção 4.2 constrói-se um modelo linear aplicando a técnica de discretização (apresentada na Secção 1.3.1) ao modelo não linear. Este modelo discretizado é semelhante ao modelo (SUP_D) apresentado no capítulo anterior. De igual modo o modelo linear apresentado na Secção 4.3 é semelhante ao modelo com variáveis-arco discretizadas apresentado no capítulo anterior para a variante *Árvore de Suporte*. Na Secção 4.3.1 mostra-se como fortalecer o modelo da Secção 4.2 através de desigualdades válidas implicadas pelo modelo linear da Secção 4.3. No final das Secções 4.2 e 4.3 apresentam-se resultados computacionais para avaliar a qualidade dos limites inferiores fornecidos pela relaxação linear dos modelos.

4.1 Modelo não linear

Recordando a reformulação do (AStRP) num grafo orientado, feita na Secção 1.1.3, também o (AStRPCG) é modelado num grafo orientado $G_0 = (V_0, A_0, c_{ij})$ onde $V_0 = V \cup \{0\}$ (0 é a raiz fictícia de qualquer solução/arborescência) e $A_0 = A \cup \{(0, j) : j \in V_c\}$, sendo $V_c = \{i \in V : p_i > 0\}$ o conjunto dos *nodos clientes*. Nesta versão do (AStRP) a parte topológica da solução é definida recorrendo novamente às variáveis apresentadas na Secção 1.1.3. Assim, as variáveis x_{ij} e z_i têm o mesmo significado, recordado aqui:

$$\forall (i, j) \in A_0, \quad x_{ij} = \begin{cases} 1 & \text{se o arco } (i, j) \text{ está na solução} \\ 0 & \text{caso contrário} \end{cases}$$

e

$$\forall i \in V, \quad z_i = \begin{cases} 1 & \text{se o nodo } i \text{ está na solução} \\ 0 & \text{caso contrário} \end{cases}$$

Quanto às variáveis-grau, e ao contrário da variante *Árvore de Suporte*, o grau de qualquer nodo (à excepção de $i = 0$) pode ser nulo, indicando que esse nodo não se encontra presente na solução. Sendo assim, é necessário considerar variáveis que representem efectivamente o grau do nodo i , $g(i)$. Essas variáveis serão designadas por Y_i para as distinguir das variáveis U_i , que representavam $g(i) - 1$, utilizadas no modelo (SUP_{NL}). À partida, o grau de qualquer nodo i na solução é $g(i) = Y_i \in \mathbb{N}_0, \forall i \in V$, no entanto, em qualquer solução óptima, nenhum nodo com $p_i = 0$ terá grau unitário. Se por absurdo isso acontecesse, bastava remover o único arco ligado a esse nodo e obter-se-ia uma solução admissível com menor custo (descontando apenas o custo do arco removido já que $p_i = 0$).

Sendo assim, em qualquer solução óptima $g(i) = Y_i \in \mathbb{N}_0 \setminus \{1\}$, $\forall i \in V_{nc}$. Além disso, para cada um dos nodos com prémio nulo a função de custos nos nodos, $\Phi(\cdot)$, é redefinida da seguinte forma (ver Capítulo 2):

$$\Phi^*(Y_i) = \begin{cases} 0 & Y_i = 0, 1, 2 \\ \phi^1 & 3 \leq Y_i \leq Q \\ \phi^m & (m-1) \cdot Q + 1 \leq Y_i \leq m \cdot Q \\ \phi^M & (M-1) \cdot Q + 1 \leq Y_i \leq D \end{cases}, \quad m = 2, \dots, M-1 \quad (4.1)$$

ou seja, $\Phi^*(Y_i) = \Phi(Y_i)$ para $Y_i \neq 2$. Para os nodos $i \in V_c$, a definição da função de custos (2.2) mantém-se.

O modelo básico não linear para o (AStRPCG) é apresentado na Figura 4.1. Na função objectivo (4.2) a parcela referente ao custo associado ao grau dos nodos na solução é não linear, definida com recurso à função (2.2) para os nodos $i \in V_c$ e à função (4.1) para os nodos $i \in V_{nc}$. Além disso, a função objectivo contém uma parcela constante — $\sum_{i \in V_c} p_i$ — que, sem perda de generalidade, será omitida nos modelos seguintes.

As restrições (1.13) - (1.17) foram já definidas e descritas na Secção 1.1.3; as restrições (1.14) estão escritas de uma forma genérica, no entanto, nos resultados computacionais apresentados nas secções seguintes comparam-se novamente os efeitos da utilização de um sistema de fluxos *SC* ou *MC* para garantir a conexidade da solução.

As restrições de *grau externo*, (4.3a) e (4.3b), definem o número de arcos divergentes¹ de qualquer nodo $i \in V$, quer este esteja ou não presente na solução.

¹Utiliza-se $A^+(i)$ em vez de $A_0^+(i)$ visto as restrições serem definidas apenas para os nodos de V .

$$(StP_{NL}) \min \sum_{(i,j) \in A} c_{ij} x_{ij} + \sum_{i \in V_c} p_i(1 - z_i) + \sum_{i \in V_c} \Phi(Y_i) + \sum_{i \in V_{nc}} \Phi^*(Y_i) \quad (4.2)$$

$$s.a : X(A_0^-(i)) = z_i \quad i \in V \quad (1.13)$$

$$\{ (i, j) \in A_0 : x_{ij} = 1 \} \text{ é conexo} \quad (1.14)$$

$$X(A^+(i)) = Y_i - z_i + \begin{cases} x_{0i} & i \in V_c \\ 0 & i \in V_{nc} \end{cases} \quad (4.3a)$$

$$0 & i \in V_{nc} \quad (4.3b)$$

$$Y_i \leq D \cdot z_i \quad i \in V \quad (4.4)$$

$$X(A_0^+(0)) = 1 \quad (1.15)$$

$$Y_i \geq 2 \quad \text{ou} \quad Y_i = 0 \quad i \in V_{nc} \quad (4.5)$$

$$x_{ij} \in \{0, 1\} \quad (i, j) \in A_0 \quad (1.16)$$

$$z_i \in \{0, 1\} \quad i \in V \quad (1.17)$$

$$Y_i \in \mathbb{N}_0 \quad i \in V_c \quad (4.6)$$

Figura 4.1: Modelo genérico não linear para o (AStRPCG).

Para nodos que não venham a fazer parte da solução tem-se $z_i = 0$, o que implica, pelas restrições (1.13) e (4.4), que $x_{0i} = Y_i = 0$. Sendo assim, o número de arcos divergentes do nodo i será efectivamente nulo, $X(A^+(i)) = 0$. Para os nodos presentes na solução, $z_i = 1$ e como tal, o número de arcos divergentes no nodo i é determinado pela diferença entre o seu grau na solução, Y_i , e o número de arcos convergentes em i , $z_i = 1$ (pelas restrições (1.13)). Quanto aos nodos em V_c , é ainda necessário distinguir o caso do nodo que será efectivamente a raiz da solução no grafo $G = (V, A, c_{ij})$. Para este nodo tem-se $x_{0i} = z_i = 1$ e assim, o número de arcos divergentes será exactamente igual ao seu grau, Y_i .

As restrições (4.4) definem o grau máximo de qualquer nodo de V presente na solução; apenas os nodos presentes na solução poderão ter um grau superior a 0. Não é necessário acrescentar restrições do tipo $Y_i \geq z_i, i \in V_c$, por forma a assegurar o grau mínimo dos nodos $i \in V_c$ (clientes potenciais) presentes na solução, visto estas restrições serem implicitamente verificadas pela conexidade da solução. Para os nodos em V_{nc} as restrições disjuntivas (4.5) garantem que o seu grau, caso estejam presentes na solução, não seja unitário. Finalmente, as restrições (4.6) definem o domínio das novas variáveis.

4.2 Modelo com variáveis-nodo discretizadas

O modelo anterior, além da parte não linear da função objectivo, possui também uma restrição disjuntiva nas variáveis Y_i . Os exemplos dados na Secção 1.3.1 mostram que a técnica de reformulação por discretização pode ser utilizada para lidar com ambos os casos. Para tal define-se o seguinte conjunto de variáveis binárias:

$$y_i^d = \begin{cases} 1 & \text{se o nodo } i \text{ tem grau } d \text{ na solução} \\ 0 & \text{caso contrário} \end{cases} \quad \forall i \in V, d = 1, \dots, D$$

Ao contrário das variáveis u_i^d do modelo (SUP_D) definidas na Secção 3.3, agora o índice d representa efectivamente o grau do nodo i . No caso de um nodo não estar presente na solução ter-se-á $\sum_{d=1}^D y_i^d = 0$, caso contrário $\sum_{d=1}^D y_i^d = 1$. Com estas novas variáveis é possível agora reescrever as restrições disjuntivas (4.5), bastando para tal fazer $y_i^1 = 0, \forall i \in V_{nc}$.

As novas variáveis e as anteriores Y_i e z_i relacionam-se da seguinte forma:

$$Y_i = \sum_{d=1}^D d \cdot y_i^d \quad \forall i \in V \quad (4.7)$$

$$z_i = \sum_{d=1}^D y_i^d \quad \forall i \in V \quad (4.8)$$

Analogamente ao tratamento dado à função não linear do modelo (SUP_{NL}) apresentado no Capítulo dedicado ao (ASupCG), também neste caso o domínio da função não linear (4.2) pode ser particionado em subconjuntos (para um qualquer nodo em V_c ²): $\{0, 1\} \cup \{2, \dots, Q\} \cup \{Q + 1, \dots, 2Q\} \cup \dots \cup \{(M - 1)Q, \dots, D\}$. Assim, a função $\Phi(Y_i)$ para nodos em V_c é reescrita como:

$$\Phi(Y_i) = \phi^1 \cdot \sum_{d=2}^Q y_i^d + \sum_{m=2}^{M-1} \phi^m \cdot \sum_{d=(m-1)Q+1}^{mQ} y_i^d + \phi^M \cdot \sum_{d=(M-1)Q+1}^D y_i^d$$

Definindo $\gamma^d = \Phi(d)$ para $d = 1, \dots, D$, é possível ainda reescrevê-la como, $\Phi(Y_i) = \sum_{d=1}^D \gamma^d \cdot y_i^d$. Para os nodos em V_{nc} o raciocínio é semelhante e a função $\Phi^*(Y_i)$ é reescrita como, $\Phi^*(Y_i) = \sum_{d=3}^D \gamma^d \cdot y_i^d$, onde $\gamma^d = \Phi(d) = \Phi^*(d)$, para $d = 3, \dots, D$.

Na Figura 4.2 é apresentado o modelo discretizado para o (AStRPCG) (note-se a semelhança com o modelo (SUP_D) apresentado na Figura 3.4 para o (ASupCG)).

²Para os nodos de V_{nc} a partição só é diferente nos dois primeiros conjuntos: $\{0, 1, 2\}$ e $\{3, \dots, Q\}$.

$$(StP_D) \min \sum_{(i,j) \in A} c_{ij} x_{ij} + \sum_{i \in V_c} \sum_{d=1}^D (\gamma^d - p_i) \cdot y_i^d + \sum_{i \in V_{nc}} \sum_{d=3}^D \gamma^d \cdot y_i^d \quad (4.9)$$

$$s.a : X(A_0^-(i)) = \sum_{d=1}^D y_i^d \quad i \in V \quad (4.10)$$

$$\{ (i, j) \in A_0 : x_{ij} = 1 \} \text{ é conexo} \quad (1.14)$$

$$X(A^+(i)) = \sum_{d=2}^D (d-1) \cdot y_i^d + \begin{cases} x_{0i} & i \in V_c \\ 0 & i \in V_{nc} \end{cases} \quad (4.11a)$$

$$(4.11b)$$

$$\sum_{d=1}^D y_i^d \leq 1 \quad i \in V_c \quad (4.12)$$

$$X(A_0^+(0)) = 1 \quad (1.17)$$

$$y_i^1 = 0 \quad i \in V_{nc} \quad (4.13)$$

$$x_{ij} \in \{0, 1\} \quad (i, j) \in A_0 \quad (1.16)$$

$$y_i^d \in \{0, 1\} \quad i \in V, d = 1, \dots, D \quad (4.14)$$

Figura 4.2: Modelo Linear com variáveis-grau discretizadas para o (PASTRPCG).

Para os nodos clientes, o coeficiente da variável y_i^d na função objectivo, $\gamma^d - p_i$, significa que, caso o nodo $i \in V_c$ esteja presente na solução, duas situações ocorrem: deve-se descontar a "penalidade" p_i e pagar um custo de acordo com o grau que o nodo tem na solução (se este nodo for uma folha na solução então o seu "custo" será $\gamma^1 - p_i = -p_i$). Para os nodos de V_{nc} o índice do somatório varia entre 3 e D visto $\gamma^1 = \gamma^2 = 0$. Pelas equações de substituição (4.7) e (4.8) as restrições (4.4) do modelo não linear tornam-se redundantes. As restrições de grau externo, (4.11a) e (4.11b), correspondem às anteriores restrições (4.3a) e (4.3b) após a substituição de variáveis. Note-se que, devido a esta substituição as variáveis y_i^1 não aparecem nas restrições de grau externo.

De facto, se existirem arcos divergentes de um nodo i , que não esteja ligado à raiz fictícia, então o seu grau deve ser superior ou igual a 2, ou seja, uma das variáveis y_i^d com $d \geq 2$ tem de ser não nula. As restrições (4.12) representam a consistência das variáveis y_i^d devido à aplicação da técnica de discretização. As restrições (4.13) substituem as restrições disjuntivas (4.5) do anterior modelo (*cf.* Secção 1.3.1).

4.2.1 Resultados Computacionais: Modelo (StP_D)

A descrição da geração das instâncias utilizadas nesta secção encontra-se no Apêndice B. Nas tabelas seguintes as primeiras colunas são idênticas às apresentadas nas tabelas para o (ASupCG) com excepção da coluna "nc" que indica o número de nodos clientes na instância. Nas colunas seguintes apresentam-se os desvios obtidos com o modelo ($LStP_D$), consoante se utiliza um sistema de fluxos SC ou MC . O modelo ($LStP_D$) com sistema de fluxos SC é ainda fortalecido com as desigualdades válidas (1.22) (ver Secção 1.1.3) – colunas SC^* e SC_a^* – e com as desigualdades de *Assimetria* (1.18) (ver Secção 1.1.3) – colunas SC_a e SC_a^* . Estas últimas desigualdades são também utilizadas para fortalecer o modelo com o sistema de fluxos MC – coluna MC_a . Não se apresentam os tempos de CPU gastos na resolução dos modelos lineares, visto estes serem inferiores a 5 segundos (*cf.* Apêndice D).

Comparando sistemas de fluxos.

O modelo ($LStP_D$) com o sistema de fluxos SC ganha consideravelmente com a adição das desigualdades (1.22), principalmente nas redes *Cabo* onde o modelo ($LStP_D$) chega a registar desvios superiores a 40%.

Ainda assim, registam-se algumas reduções nos desvios entre a utilização dos sistemas SC^* e MC , principalmente nas redes *Cabo* mais densas. Em algumas instâncias (nas redes *Wireless* com $D = 3$) o modelo ($LStP_D$) com o sistema de fluxos MC consegue mesmo reduzir o desvio a zero.

As desigualdades de Assimetria.

A adição das desigualdades de assimetria é mais eficaz no modelo ($LStP_D$) com o sistema de fluxos SC (onde se atingem os maiores desvios). Neste caso, as reduções nos desvios são maiores nas redes *Cabo*. No modelo ($LStP_D$) com o sistema SC^* as desigualdades de *Assimetria* (1.18) conseguem ainda reduzir os desvios, principalmente nas redes *Cabo*. Já no modelo que utiliza o sistema MC a utilização das desigualdades (1.18) não tem qualquer efeito no limite obtido por este modelo. No entanto, isso já não acontecerá nos modelos fortalecidos apresentados na secção seguinte.

Dimensão e Tipo de rede.

Os desvios são maiores nas redes *Cabo* do que nas redes *Wireless* e para estas últimas os desvios geralmente são menores para as redes mais densas.

Efeito dos parâmetros.

Em quase todas as instâncias os desvios aumentam com o grau máximo, D , independentemente do sistema de fluxos que está a ser utilizado e com a adição ou não das desigualdades (1.22) e/ou (1.18); existem no entanto algumas excepções nas instâncias com 25 nodos. Nas redes *Wireless*, quando o custo de cada módulo, K_2 , aumenta, os desvios diminuem. Este comportamento regista-se também na maioria das instâncias *Cabo*.

R	$ E $	nc	(K_1, K_2)	D	$(LStP_D)$ desvio					
					SC	SC_α	SC^*	SC_α^*	MC	MC_α
75	(100, 10)	7		3	2,5	1,9	1,5	1,5	1,5	1,5
				4	11,3	10,4	10,2	10,2	10,2	10,2
				5	12,6	11,5	11,3	11,3	11,3	11,3
				6	9,1	8,1	6,4	6,4	6,4	6,4
	(100, 50)	7		3	1,2	0*	0*	0*	0*	0*
				4	2,8	2,0	0,8	0,8	0,8	0,8
				5	4,4	3,9	3,9	3,9	3,9	3,9
				6	5,6	4,7	4,5	4,5	4,5	4,5
	(100, 10)	13		3	2,0	1,5	0,6	0,6	0,6	0,6
				4	8,1	7,4	6,4	6,4	6,4	6,4
				5	13,3	12,5	11,3	11,3	11,3	11,3
				6	19,8	19,0	17,6	17,6	17,6	17,6
(100, 50)	13		3	1,9	1,4	1,1	0,9	0,9	0,9	
			4	3,2	2,8	2,4	2,4	2,4	2,4	
			5	7,3	6,7	5,5	5,5	5,5	5,5	
			6	11,7	11,1	10,1	10,1	10,1	10,1	
W	(100, 10)	7		3	1,2	1,0	1,1	1,0	1,0	1,0
				4	7,6	6,7	6,7	6,7	6,7	6,7
				5	10,1	9,1	9,1	9,1	9,1	9,1
				6	6,3	5,0	5,0	5,0	5,0	5,0
	(100, 50)	7		3	1,2	0,4	0*	0*	0*	0*
				4	1,2	0,5	0,1	0,1	0,1	0,1
				5	2,5	1,7	1,5	1,3	1,3	1,3
				6	4,4	3,5	3,5	3,5	3,5	3,5
	(100, 10)	150		3	1,2	1,1	0,8	0,6	0,6	0,6
				4	5,6	5,6	5,0	5,0	4,9	4,9
				5	9,6	9,6	8,9	8,9	8,7	8,7
				6	10,9	10,9	10,0	10,0	9,8	9,8
(100, 50)	13		3	0,5	0,2	0*	0*	0*	0*	
			4	1,9	1,7	1,1	0,8	0,8	0,8	
			5	4,7	4,6	4,2	4,1	4,1	4,1	
			6	4,5	4,5	3,9	3,9	3,8	3,8	
C	(100, 10)	7		3	19,5	12,3	2,9	2,9	2,9	2,9
				4	27,6	19,3	8,9	8,9	8,9	8,9
				5	36,9	27,6	16,5	16,5	16,5	16,5
				6	42,3	32,4	20,8	20,8	20,8	20,8
	(100, 50)	7		3	16,9	10,5	2,8	2,8	2,8	2,8
				4	20,3	13,6	4,8	4,8	4,8	4,8
				5	32,9	25,3	15,8	15,8	15,8	15,8
				6	40,3	31,9	22,0	22,0	22,0	22,0
	(100, 10)	75		3	13,4	10,7	7,1	7,1	7,1	7,1
				4	15,9	13,3	9,9	9,9	9,9	9,9
				5	25,2	22,6	18,3	18,3	18,3	18,3
				6	30,8	28,3	23,4	23,4	23,4	23,4
(100, 50)	13		3	12,7	10,4	7,3	7,3	7,3	7,3	
			4	11,8	9,7	7,3	7,3	7,3	7,3	
			5	23,0	20,8	17,8	17,8	17,8	17,8	
			6	30,7	28,5	24,8	24,8	24,8	24,8	
RB2.0ex(100, 10)	(100, 10)	7		3	22,9	17,8	7,2	7,2	7,2	7,2
				4	29,2	23,3	10,1	8,7	8,7	8,7
				5	41,2	35,1	18,8	16,9	15,9	15,9
				6	47,9	41,7	24,6	22,8	21,4	21,4
	(100, 50)	7		3	22,1	17,6	8,5	8,5	8,5	8,5
				4	22,3	17,6	7,3	6,9	6,9	6,9
				5	35,7	30,7	18,5	17,2	17,1	17,1
				6	41,2	35,6	20,8	19,0	18,2	18,2
	(100, 10)	150		3	12,7	9,3	6,3	6,2	5,7	5,7
				4	16,1	12,3	9,1	8,7	8,0	8,0
				5	25,2	20,7	15,0	14,5	13,8	13,8
				6	33,1	28,1	21,2	20,7	20,0	20,0
(100, 50)	13		3	11,9	9,2	6,6	6,6	6,1	6,1	
			4	11,5	8,5	6,1	5,7	5,2	5,2	
			5	20,2	16,9	13,6	13,3	12,7	12,7	
			6	29,5	25,7	20,9	20,6	20,0	20,0	

Tabela 4.1: Resultados para instâncias com 25 nodos: Modelo $(LStP_D)$.

R	$ E $	nc	(K_1, K_2)	D	$(LStP_D)$					
					desvio					
					SC	SC_a	SC^*	SC_a^*	MC	MC_a
W	(100, 10)	13	3	3,6	2,9	1,3	1,3	1,1	1,1	
			4	9,0	7,9	5,9	5,8	5,7	5,7	
			5	12,8	11,5	9,1	8,9	8,8	8,8	
			6	17,8	16,3	13,7	13,6	13,4	13,4	
	(100, 50)	150	3	1,2	0,8	0,4	0,4	0,4	0,4	
			4	3,0	2,6	1,6	1,4	1,3	1,3	
			5	5,9	5,0	3,6	3,4	3,3	3,3	
			6	10,5	9,4	7,5	7,3	7,2	7,2	
	(100, 10)	25	3	2,9	2,6	0,7	0,7	0,7	0,7	
			4	5,2	5,1	3,5	3,5	3,5	3,5	
			5	7,9	7,8	6,1	6,1	6,1	6,1	
			6	12,6	12,4	10,6	10,6	10,6	10,6	
	(100, 50)	300	3	2,2	1,8	0,8	0,7	0,7	0,7	
			4	3,5	3,2	2,2	2,2	2,2	2,2	
			5	4,6	4,4	2,5	2,5	2,5	2,5	
			6	7,2	7,1	5,8	5,8	5,8	5,8	
	(100, 10)	13	3	2,2	1,6	0,6	0,6	0,6	0,6	
			4	6,6	6,0	4,1	4,1	4,1	4,1	
			5	12,3	11,5	9,3	9,3	9,3	9,3	
			6	14,7	13,8	11,3	11,3	11,3	11,3	
	(100, 50)	300	3	0*	0*	0*	0*	0*	0*	
			4	1,8	1,2	1,2	1,2	1,2	1,2	
			5	3,9	3,4	1,7	1,7	1,7	1,7	
			6	7,7	7,1	5,3	5,3	5,3	5,3	
(100, 10)	25	3	1,5	1,2	0,5	0,5	0,5	0,5		
		4	4,9	4,8	3,8	3,8	3,8	3,8		
		5	7,7	7,6	6,3	6,3	6,3	6,3		
		6	11,8	11,7	10,2	10,2	10,2	10,2		
(100, 50)	300	3	0,5	0,2	0,1	0,1	0,1	0,1		
		4	1,6	1,3	0,7	0,7	0,7	0,7		
		5	3,6	3,4	2,3	2,3	2,3	2,3		
		6	7,9	7,8	6,8	6,8	6,8	6,8		
C	(100, 10)	13	3	19,1	15,2	1,7	1,6	1,6	1,6	
			4	26,5	21,4	6,7	6,7	6,2	6,2	
			5	34,7	29,1	12,9	12,9	12,9	12,9	
			6	39,9	34,0	17,0	17,0	17,0	17,0	
	(100, 50)	150	3	15,6	12,8	1,2	0,9	0,9	0,9	
			4	20,3	16,7	3,1	3,0	2,7	2,7	
			5	30,9	26,5	12,2	12,1	11,8	11,8	
			6	37,3	32,4	17,2	17,1	17,1	17,1	
	(100, 10)	25	3	18,0	16,6	5,2	4,7	4,7	4,7	
			4	25,8	24,3	11,8	11,2	11,1	11,1	
			5	32,3	30,6	16,4	15,8	15,8	15,8	
			6	36,9	35,2	20,1	19,4	19,4	19,4	
	(100, 50)	300	3	15,9	14,7	4,8	4,4	4,2	4,2	
			4	19,6	18,4	7,9	7,5	7,4	7,4	
			5	28,6	27,3	15,4	14,9	14,8	14,8	
			6	34,9	33,4	20,5	20,0	20,0	20,0	
	(100, 10)	13	3	20,6	18,9	7,0	7,0	7,0	7,0	
			4	30,7	28,2	16,2	16,1	16,1	16,1	
			5	38,6	35,9	22,2	22,1	21,7	21,7	
			6	45,0	42,0	27,4	27,3	26,8	26,8	
	(100, 50)	150	3	18,9	17,5	7,2	7,2	7,2	7,2	
			4	23,6	21,4	12,0	11,9	11,9	11,9	
			5	32,8	30,6	19,5	19,4	19,3	19,3	
			6	39,8	37,3	25,1	25,0	24,7	24,7	
(100, 10)	25	3	9,7	9,0	3,7	3,7	3,7	3,7		
		4	18,1	17,1	9,9	9,9	9,9	9,9		
		5	24,2	23,2	14,5	14,4	14,0	14,0		
		6	27,9	26,8	17,6	17,5	16,7	16,7		
(100, 50)	300	3	8,5	7,9	3,7	3,6	3,6	3,6		
		4	12,7	11,8	6,7	6,7	6,7	6,7		
		5	21,0	20,2	13,2	13,2	13,2	13,2		
		6	24,6	23,7	15,7	15,6	15,2	15,2		

Tabela 4.2: Resultados para instâncias com 50 nodos: Modelo $(LStP_D)$.

4.3 Modelo com variáveis-arco discretizadas

Nesta secção considera-se uma estratégia análoga à utilizada para o (ASupCG) para melhorar os resultados obtidos com a relaxação linear do modelo (StP_D). Para isso considere-se de novo o mesmo tipo de discretização das variáveis arco, x_{ij} , $(i, j) \in A$ (para os nodos de V_{nc} basta considerar variáveis x_{ij}^d com $d = 2, \dots, D$):

$$x_{ij}^d = \begin{cases} 1 & , \text{ se o arco } (i, j) \text{ está na solução e o grau do nodo } i \text{ é } d \\ 0 & , \text{ caso contrário} \end{cases} \quad (i, j) \in A; \quad d = 1, \dots, D$$

Adicionalmente, por uma questão de consistência das soluções do novo modelo, é necessário discretizar também as variáveis x_{0j} , $j \in V_c$, mas onde o índice extra está associado ao nodo final do arco:

$$x_{0j}^d = \begin{cases} 1 & , \text{ se o arco } (0, j) \text{ está na solução e o grau do nodo } j \text{ é } d \\ 0 & , \text{ caso contrário} \end{cases} \quad j \in V_c, \quad d = 1, \dots, D$$

As restrições de ligação entre estas novas variáveis e as variáveis x_{ij} e y_i^d são apresentadas na Figura 4.3. As restrições (4.15a) e (4.15b) têm uma interpretação directa: se um arco (i, j) estiver presente na solução então uma das variáveis x_{ij}^d será igual a 1, para um dos valores possíveis para o grau do nodo i . A interpretação das restrições (4.15c) é análoga. As restrições (4.16a) e (4.16b) podem ser vistas como uma versão desagregada das restrições de ligação (4.11a) e (4.11b) do modelo (StP_D), respectivamente. De facto, para um dado nodo $i \in V_c$, somando as restrições (4.16a) para todo o $d = 1, \dots, D$ e usando as restrições (4.15b) e (4.15c) obtém-se a respectiva restrição (4.11a). Para os nodos $i \in V_{nc}$ o raciocínio é análogo.

$$x_{ij} = \sum_{d=2}^D x_{ij}^d \quad (i, j) \in A, i \in V_{nc} \quad (4.15a)$$

$$x_{ij} = \sum_{d=1}^D x_{ij}^d \quad (i, j) \in A, i \in V_c \quad (4.15b)$$

$$x_{0i} = \sum_{d=1}^D x_{0i}^d \quad (0, i) \in A_0 \quad (4.15c)$$

$$X^d(A^+(i)) = (d-1) \cdot y_i^d + x_{0i}^d \quad i \in V_c, d = 1, \dots, D \quad (4.16a)$$

$$X^d(A^+(i)) = (d-1) \cdot y_i^d \quad i \in V_{nc}, d = 2, \dots, D \quad (4.16b)$$

$$x_{0i}^d \leq y_i^d \quad i \in V_c, d = 1, \dots, D \quad (4.16c)$$

Figura 4.3: Sistema linear de definição das variáveis x_{ij}^d para o (ASTRPCG).

Finalmente, as restrições (4.16c) estabelecem a ligação directa entre as variáveis associadas aos arcos fictícios, $(0, i)$ e as variáveis y_i^d . São necessárias para manter a consistência da solução. Para compreender melhor estas restrições, considere-se o caso de um nodo $i \in V_c$ ligado à raiz fictícia e cujo grau é $\delta \in \{1, \dots, D\}$, *i.e.*, $\sum_{d=1}^D x_{0i}^d = 1$, $y_i^\delta = 1$ e $y_i^d = 0$, $d \neq \delta$. Então, as restrições (4.16a) ficam escritas como:

$$\begin{aligned} X^d(A^+(i)) &= x_{0i}^d & d \neq \delta \\ X^\delta(A^+(i)) &= (\delta - 1) + x_{0i}^\delta \end{aligned}$$

Para evitar soluções em que $x_{ij}^d = 1$ para algum $d \neq \delta$ (pois o grau do nodo i é δ), basta garantir que $x_{0i}^d = 0$ para todo o valor $d \neq \delta$. O objectivo das restrições de ligação (4.16c) é exactamente esse e assim tem-se $x_{0i}^d \leq 0$ para todo o valor $d \neq \delta$.

Para os nodos $i \in V_c$ que não estejam ligados à raiz fictícia e cujo grau seja $\delta \in \{1, \dots, D\}$, tem-se $x_{ij}^d = 0$, $d \neq \delta$ pois $x_{0i}^d = y_i^d = 0$ para todo $d \neq \delta$.

Aplicando o mesmo raciocínio que foi utilizado anteriormente para o problema (ASupCG), para o (AStRPCG) o modelo estendido obtém-se adicionando ao modelo (StP_D) as restrições de ligação (4.15a) – (4.16c) e as seguintes desigualdades válidas, novamente designadas por *desigualdades Arco Desagregadas*:

$$AD(i, j; d): \quad x_{ij}^d \leq y_i^d \quad (i, j) \in A; \quad d = 1, \dots, D$$

Para nodos $i \in V_{nc}$, as desigualdades $AD(i, j; d)$ são idênticas mas apenas estão definidas para $d \geq 2$.

Assim como para o (ASupCG) também no caso do (AStRPCG) não é possível encontrar mais desigualdades válidas, para além das desigualdades $AD(i, j; d)$, relacionando as variáveis x_{ij}^d e as variáveis y_i^d , que permitam melhorar os limites inferiores obtidos pela relaxação do modelo estendido. De facto, para um dado nodo $i \in V_{nc}$ e um dado valor $d \in \{2, \dots, D\}$, considere-se o poliedro $\mathcal{P}_{i,d}^{nc}$ definido por:

$$\begin{aligned} X^d(A^+(i)) &= (d-1) \cdot y_i^d \\ 0 \leq x_{ij}^d &\leq 1 && (i, j) \in A^+(i) \\ 0 \leq y_i^d &\leq 1 \end{aligned}$$

Este poliedro é idêntico ao poliedro definido na Secção 3.4 para o (ASupCG) (apenas o coeficiente no segundo membro da igualdade é diferente). Como tal, ao adicionar a este poliedro as desigualdades $AD(i, j; d)$ para todos os arcos $(i, j) \in A^+(i)$ e para o mesmo valor de d , obtém-se a descrição linear completa do envolvente convexo definido pelas soluções inteiras de $\mathcal{P}_{i,d}^{nc}$ (ver a demonstração no Apêndice A).

Para um dado nodo $i \in V_c$ e para um dado valor $d \in \{1, \dots, D\}$ o poliedro $\mathcal{P}_{i,d}^c$ é definido por:

$$\begin{aligned} X^d(A^+(i)) &= (d-1) \cdot y_i^d + x_{0i}^d \\ x_{0i}^d &\leq y_i^d \\ 0 &\leq x_{ij}^d \leq 1 && (i, j) \in A^+(i) \\ 0 &\leq x_{0i}^d \leq 1 \\ 0 &\leq y_i^d \leq 1 \end{aligned}$$

Este poliedro é ligeiramente diferente do anterior, mas também é possível provar (ver Apêndice A) que ao introduzir no poliedro $\mathcal{P}_{i,d}^c$, as desigualdades válidas $AD(i, j; d)$, $\forall (i, j) \in A^+(i)$ e para o mesmo valor de d , se obtém a descrição linear completa do envolvente convexo definido pelas soluções inteiras de $\mathcal{P}_{i,d}^c$.

Designa-se o modelo fortalecido ($P + AD$), obtido ao adicionar todas as desigualdades válidas $AD(i, j; d)$ bem como as restrições de ligação (4.15a) – (4.16c) a um modelo genérico (P).

4.3.1 Desigualdades *Arco Desagregadas* no espaço das variáveis (x, u)

Analogamente ao problema (ASupCG), também para o problema (AStRPCG) é possível fortalecer o modelo discretizado (StP_D), apresentado na Figura 4.2, com a introdução de desigualdades válidas implicadas pelas desigualdades válidas $AD(i, j; d)$.

Assim sendo, para um qualquer nodo $i \in V$ e um qualquer subconjunto de arcos $H(i) \subseteq A^+(i)$, tal que $1 \leq h \equiv |H(i)| \leq D - 1$, considerem-se as seguintes desigualdades *Arco*:

$$A_h(i, H(i)) \quad X(H(i)) \leq \sum_{d=2}^h (d-1) \cdot y_i^d + h \cdot \sum_{d=h+1}^D y_i^d + \begin{cases} x_{0i} & i \in V_c \\ 0 & i \in V_{nc} \end{cases} \\ H(i) \subseteq A^+(i)$$

Para subconjuntos $H(i)$ com mais do que $D - 1$ arcos as desigualdades $A_h(i, H(i))$ são dominadas pelas restrições de ligação (4.11a) e (4.11b).

A interpretação destas desigualdades é semelhante à interpretação feita para as desigualdades $A_h(i, H(i))$ do problema (ASupCG): se todos os arcos de $H(i)$ estiverem presentes na solução então o primeiro membro da desigualdade é igual a h . Para os nodo $i \in V_{nc}$ isto implica que (exactamente) uma das variáveis y_i^d , $d \geq h + 1$, seja igual a 1, *i.e.*, o grau do nodo i tem de ser no mínimo igual a $h + 1$ (além dos h arcos divergentes tem de existir um arco convergente) sendo a desigualdade satisfeita como igualdade. O mesmo se passa com os nodos $i \in V_c$ que não estejam ligados à raiz fictícia.

Para o nodo $i \in V_c$ ligado à raiz fictícia existem duas hipóteses: ou o grau do nodo é igual a h ($y_i^h = 1$) e a desigualdade é satisfeita como igualdade, ou o grau do nodo é maior do que h ($\sum_{d=h+1}^D y_i^d = 1$) e a desigualdade é satisfeita como desigualdade estrita.

Para mostrar que as desigualdades $A_h(i, H(i))$, para $i \in V$ e $1 \leq h \leq D - 1$, são válidas para o modelo discretizado (StP_D) mostra-se que são implicadas pelo modelo ($StP_D + AD$).

Para tal considere-se para um qualquer nodo $i \in V_{nc}$, o poliedro \mathcal{P}_i^{nc} definido por:

$$\begin{aligned}
X^d(A^+(i)) &= (d-1) \cdot y_i^d & d = 2, \dots, D \\
x_{ij} &= \sum_{d=2}^D x_{ij}^d & (i, j) \in A^+(i) \\
x_{ij}^d &\leq y_i^d & (i, j) \in A^+(i); d = 2, \dots, D \\
0 \leq x_{ij}^d &\leq 1 & (i, j) \in A^+(i); d = 2, \dots, D \\
0 \leq x_{ij} &\leq 1 & (i, j) \in A^+(i) \\
0 \leq y_i^d &\leq 1 & d = 2, \dots, D
\end{aligned}$$

e o poliedro \mathcal{Q}_i^{nc} definido por ($h = |H(i)|$):

$$\begin{aligned}
X(A^+(i)) &= \sum_{d=2}^D (d-1) \cdot y_i^d \\
X(H(i)) &\leq \sum_{d=2}^h (d-1) \cdot y_i^d + h \cdot \sum_{d=h+1}^D y_i^d & H(i) \subseteq A^+(i), 1 \leq h \leq D-1 \\
0 \leq x_{ij} &\leq 1 & (i, j) \in A^+(i) \\
0 \leq y_i^d &\leq 1 & d = 2, \dots, D
\end{aligned}$$

Entre estes dois poliedros verifica-se a seguinte relação:

Proposição 4.3.1. *Para qualquer nodo $i \in V_{nc}$*

$$\text{proj}_{x,u}(\mathcal{P}_i^{nc}) \subseteq \mathcal{Q}_i^{nc}$$

Demonstração. A prova é análoga à da Proposição 3.4.1. □

Para os nodos de V_c é possível definir poliedros semelhantes. Assim, o poliedro \mathcal{P}_i^c é definido por:

$$\begin{aligned}
X^d(A^+(i)) &= (d-1) \cdot y_i^d + x_{0i}^d & d = 1, \dots, D \\
x_{ij} &= \sum_{d=1}^D x_{ij}^d & (i, j) \in A^+(i) \\
x_{0i} &= \sum_{d=1}^D x_{0i}^d \\
x_{0i}^d &\leq y_i^d & d = 1, \dots, D \\
x_{ij}^d &\leq y_i^d & (i, j) \in A^+(i); d = 1, \dots, D \\
0 \leq x_{ij}^d &\leq 1 & (i, j) \in A^+(i); d = 1, \dots, D \\
0 \leq x_{ij} &\leq 1 & (i, j) \in A^+(i) \\
0 \leq y_i^d &\leq 1 & d = 1, \dots, D
\end{aligned}$$

e o poliedro \mathcal{Q}_i^c é definido por ($h = |H(i)|$):

$$\begin{aligned}
X(A^+(i)) &= \sum_{d=2}^D (d-1) \cdot y_i^d + x_{0i} \\
X(H(i)) &\leq \sum_{d=2}^h (d-1) \cdot y_i^d + h \cdot \sum_{d=h+1}^D y_i^d + x_{0i} & H(i) \subseteq A^+(i), 1 \leq h \leq D-1 \\
0 \leq x_{ij} &\leq 1 & (i, j) \in A^+(i) \\
0 \leq y_i^d &\leq 1 & d = 1, \dots, D
\end{aligned}$$

Também entre estes dois poliedros se verifica um resultado análogo.

Proposição 4.3.2. *Para qualquer nodo $i \in V_c$*

$$proj_{x,u}(\mathcal{P}_i^c) \subseteq \mathcal{Q}_i^c$$

Demonstração. As restrições de ligação (4.16a) implicam que:

$$X^d(H(i)) \leq (d-1) \cdot y_i^d + x_{0i}^d \quad d = 1, \dots, D \quad (4.17)$$

Por outro lado, somando as desigualdades válidas $AD(i, j; d)$ para todos os arcos de $H(i)$ obtém-se:

$$X^d(H(i)) \leq h \cdot y_i^d \quad d = 1, \dots, D \quad (4.18)$$

Somando agora as inequações (4.17) para todo o $d = 2, \dots, h$ com a soma das inequações (4.18) para todo o $d = h+1, \dots, D$ obtém-se o seguinte resultado:

$$\sum_{d=2}^D X^d(H(i)) \leq \sum_{d=2}^h (d-1) \cdot y_i^d + h \cdot \sum_{d=h+1}^D y_i^d + \sum_{d=2}^h x_{0i}^d$$

Note-se ainda que, as inequações (4.17) para $d = 1$ implicam a desigualdade $X^1(H(i)) \leq x_{0i}^1$, que somada ao último resultado permite obter:

$$\sum_{d=1}^D X^d(H(i)) \leq \sum_{d=2}^h (d-1) \cdot y_i^d + h \cdot \sum_{d=h+1}^D y_i^d + \sum_{d=1}^h x_{0i}^d$$

Após aplicação das restrições de ligação (4.15b) e (4.15c), respectivamente ao primeiro membro e ao segundo membro deste último resultado obtém-se a desigualdade válida $A_h(i, H(i))$.

Como as restrições de ligação (4.15b), (4.15c), (4.16a) e (4.16c) implicam as restrições (4.11a) conclui-se que

$$proj_{x,u}(\mathcal{P}_i^c) \subseteq \mathcal{Q}_i^c$$

□

Considere-se de novo, $(P+A)$ o modelo obtido a partir de um modelo genérico (P) ao adicionar as desigualdades $A_h(i, H(i)), \forall i \in V$, para todos os valores de h para os quais as desigualdades válidas estão definidas. Entre os dois modelos fortalecidos $(StP_D + AD)$ e $(StP_D + A)$ é possível estabelecer o seguinte resultado:

Proposição 4.3.3. *Se as restrições (1.14) forem modeladas da mesma forma para os modelos $(StP_D + AD)$ e $(StP_D + A)$ então:*

$$proj_{x,u}(Adm(LStP_D + AD)) \subseteq Adm(LStP_D + A)$$

Demonstração. Os poliedros $\mathcal{P}_i^{nc}, i \in V_{nc}$, e $\mathcal{P}_i^c, i \in V_c$, estão contidos em $ADM(LStP_D + AD)$. De igual forma, os poliedros $\mathcal{Q}_i^{nc}, i \in V_{nc}$, e $\mathcal{Q}_i^c, i \in V_c$, estão contidos em $ADM(LStP_D + A)$. Então, o resultado fica provado como consequência das Proposições 4.3.1 e 4.3.2 visto as restantes restrições de ambos os modelos serem as mesmas. \square

Como conclusão final, a relação entre os valores ótimos das relaxações lineares dos três modelos apresentados é a seguinte:

Corolário 4.3.4. *Se as restrições (1.14) forem modeladas da mesma forma para os modelos $(StP_D + AD)$ e $(StP_D + A)$, então:*

$$V(LStP_D) \leq V(LStP_D + A) \leq V(LStP_D + AD)$$

4.3.2 Resultados Computacionais:

Modelos $(StP_D + A_1)$ e $(StP_D + AD)$.

Nesta secção comparam-se os limites obtidos com o modelo discretizado ($LStP_D$) e com os modelos fortalecidos, $(LStP_D + AD)$ e $(LStP_D + A_1)$, onde este último é obtido ao adicionar as desigualdades válidas $A_1(i, H(i))$ ao modelo $(LStP_D)$. Para estes três modelos apenas se apresentam os resultados utilizando os sistemas de fluxos SC^* e MC com e sem as desigualdades (1.18). Como nas Tabelas 4.1 e 4.2, também para os modelos lineares fortalecidos não se apresentam os tempos de CPU, pois embora tenham sido mais elevados para os modelos que utilizam o sistema de fluxos MC , continuam a ser relativamente baixos (inferiores a 16 segundos *cf.* Apêndice D). As três últimas colunas indicam, respectivamente, o número de nodos clientes (VS_c), nodos não clientes (VS_{nc}) na solução óptima inteira e o tempo de CPU para resolver o problema inteiro $(StP_D + A_1)$ com o sistema de fluxos SC fortalecido com as desigualdades (1.18) e (1.22). A escolha deste modelo para obter a solução óptima inteira, prende-se basicamente com um critério: a rapidez de execução do algoritmo de *Branch-ℳ-Bound* na obtenção da solução óptima inteira (ver Apêndice C).

Comparação de Modelos fortalecidos.

A adição das desigualdades válidas, *Arco* ou *Arco Desagregadas*, no modelo $(LStP_D)$ provoca uma maior redução nos desvios nas redes *Cabo*³ do que nas redes *Wireless*. Esta redução é tanto maior quanto maior o valor de D e menor a densidade da rede. Além disso, em algumas instâncias, quando combinadas com as desigualdades de Assimetria (1.18) chegam mesmo a reduzir a zero o desvio, obtendo-se assim o valor óptimo do problema inteiro.

³Onde se chegam a registar reduções superiores a 15% com o modelo $(LStP_D + AD)$.

Quando se comparam os desvios obtidos com os modelos fortalecidos, $(LStP_D + A_1)$ e $(LStP_D + AD)$, este último consegue diminuir os desvios de uma forma mais acentuada nas redes *Cabo* onde consegue, em algumas instâncias, reduzir o desvio a zero.

As desigualdades de Assimetria.

Ao contrário do modelo $(LStP_D)$, as desigualdades (1.18) produzem agora uma diminuição nos desvios obtidos com os modelos fortalecidos que utilizam o sistema *MC*. De igual forma, estas desigualdades produzem uma redução mais acentuada nos modelos $(LStP_D + A_1)$ e $(LStP_D + AD)$ do que no modelo $(LStP_D)$, com o sistema de fluxos *SC**.

Solução ótima inteira.

Os tempos de CPU gastos pelo modelo $(LStP_D + A_1)$ utilizando o sistema de fluxos *SC* fortalecido com as desigualdades (1.18) e (1.22) são relativamente baixos na maioria dos casos. São obviamente maiores nas redes de 50 nodos e nas redes mais densas. À exceção de 4 instâncias *Wireless* de 50 nodos, para todas as restantes instâncias o algoritmo de *Branch-é-Bound* demorou menos de 2 horas a obter a solução ótima inteira.

O número de nodos clientes na solução tende a aumentar com o grau máximo nos nodos e a diminuir quando o custo de cada módulo (K_2) aumenta. Além disso, nas redes *Cabo* a percentagem de nodos clientes na solução é maior do que nas redes *Wireless*.

Capítulo 5

Conclusões

Esta dissertação centrou-se numa variante de um problema clássico de árvores em grafos: o problema da *Árvore de Suporte de custo mínimo*. Nesta variante assume-se que, para além dos custos associados às ligações entre os nodos, existem também custos associados aos nodos que dependem do número de ligações que cada nodo terá na solução. Obteve-se assim um novo problema, o problema da *Árvore de Suporte de custo mínimo com Custos dependentes do Grau* – (ASupCG). Esta variante foi também aplicada a um outro problema de árvores em grafos, o problema da *Árvore de Steiner com Recolha de Prémios*, obtendo-se assim o problema da *Árvore de Steiner com Recolha de Prémios e Custos dependentes do Grau* – (AStRPCG).

No Capítulo 2 apresentou-se a motivação para este tipo de custos associados aos nodos, no campo das redes de telecomunicações onde estes custos surgem relacionados com equipamento de *routing*. É necessário instalar este equipamento em qualquer nodo que na solução tenha mais do que uma ligação e onde, portanto, há necessidade de redireccionamento/retransmissão de tráfego.

É ainda habitual neste tipo de problemas, restringir o número máximo de ligações na solução em cada nodo, devido a possíveis interferências do sinal de transmissão.

No Capítulo 1 foram apresentados os modelos que viriam a servir de base para os dois novos problemas. Nestes modelos, para garantir a conexidade da solução, considerou-se um sistema de fluxos de múltiplas comodidades, sistema MC , e um sistema de comodidade única fortalecido com a adição de restrições de eliminação de subcircuitos para conjuntos de dois nodos, sistema SC^* . Estes modelos, nos quais se incluíram as restrições de grau nos nodos e a componente dos custos referentes aos graus dos nodos, foram então estudados nos Capítulos 3 e 4.

A estes modelos aplicaram-se duas técnicas de reformulação que permitiram resolver a não negatividade da função objectivo: a técnica de *Reformulação por Discretização* e a técnica de *Reformulação por Caminhos*, apresentadas na Secção 1.3. A aplicação da primeira técnica permitiu, além disso, construir desigualdades válidas para o respectivo modelo em ambos os problemas, (ASupCG) e (AStRPCG). A segunda técnica só pôde ser aplicada ao problema (ASupCG) em virtude de se basear numa estrutura presente apenas neste problema.

Assim, para o problema (ASupCG) foram apresentados dois tipos de modelos: modelos *Discretizados* e modelos *Árvore de Suporte/Saco Mochila*. Com base nas variáveis introduzidas pela técnica de discretização, construíram-se dois tipos de desigualdades válidas, as desigualdades *Arco* e as desigualdades *Arco Desagregadas*, que foram utilizadas para fortalecer qualquer um dos tipos de modelos.

O modelo Árvore de Suporte/Saco Mochila permitiu ainda induzir um terceiro conjunto de desigualdades válidas, as desigualdades de *Arredondamento*, que serviram para fortalecer os modelos discretizados. Estes modelos foram comparados em termos dos limites inferiores obtidos com as respectivas relaxações lineares, utilizando para isso um conjunto de instâncias com 25 e 50 nodos, considerando dois tipos de redes, *Wireless* e *Cabo*. Estas redes distinguem-se pelo tipo de custos das ligações entre os nodos.

O modelo que se revelou mais rápido na obtenção da solução ótima inteira foi o modelo discretizado, fortalecido com as desigualdades *Arco* e as desigualdades de *Arredondamento* e utilizando o sistema de fluxos *SC**. Quanto às relaxações lineares dos modelos testados, obtiveram-se limites inferiores com melhor qualidade com o modelos Árvore de Suporte/Saco Mochila fortalecido quer com as desigualdades *Arco* quer com as desigualdades *Arco Desagregadas* e utilizando o sistema de fluxos *MC* (em média obtiveram-se desvios na ordem dos 2.1% e 0.8% nas instâncias de 25 nodos e de 50 nodos, respectivamente).

Para o problema (AStRPCG), pelas razões enunciadas, apenas se apresentou um modelo *Discretizado*, semelhante ao apresentado para o problema (ASupCG). Para fortalecer este modelo foi utilizado o mesmo tipo de desigualdades válidas *Arco* e *Arco Desagregadas* apresentadas para o problema (ASupCG). Adicionalmente, considerou-se a adição de um conjunto de desigualdades válidas de *Assimetria*, que não só aceleraram a execução do algoritmo de *Branch-&-Bound* na obtenção da solução ótima inteira, como permitiram melhorar a qualidade dos limites inferiores obtidos com a relaxação linear dos modelos discretizados fortalecidos com as desigualdades *Arco* ou *Arco Desagregadas*.

Para o problema (AStRPCG) o modelo discretizado fortalecido com as desigualdades *Arco Desagregadas* e de *Assimetria*, utilizando o sistema de fluxos SC^* , produziu os melhores limites inferiores assim como foi o mais rápido na obtenção da solução óptima inteira.

Apêndice A

Demonstrações Suplementares

Resultado 1. Para quaisquer inteiros $a \in \mathbb{N}$ e $g \in \{2, 3, \dots\}$ verifica-se

$$\left\lfloor \frac{a}{g} \right\rfloor = \left\lfloor \frac{a - \left\lfloor \frac{a}{g} \right\rfloor}{g-1} \right\rfloor$$

Demonstração. Sendo $a \in \mathbb{N}$ e $g \in \{2, 3, \dots\}$ então é possível escrever a como:

$$a = g \cdot k + \delta \quad , \quad k \in \mathbb{N}_0, \delta \in \{0, 1, \dots, g-1\}$$

Então:

$$\begin{aligned} \left\lfloor \frac{a - \left\lfloor \frac{a}{g} \right\rfloor}{g-1} \right\rfloor &= \left\lfloor \frac{gk + \delta - \left\lfloor \frac{gk + \delta}{g} \right\rfloor}{g-1} \right\rfloor = \left\lfloor \frac{gk + \delta - \left\lfloor k + \frac{\delta}{g} \right\rfloor}{g-1} \right\rfloor \stackrel{\frac{\delta}{g} < 1}{=} \\ &= \left\lfloor \frac{gk + \delta - k}{g-1} \right\rfloor = \left\lfloor k + \frac{\delta}{g-1} \right\rfloor \stackrel{\frac{\delta}{g-1} \leq 1}{=} \left\lfloor k + \frac{\delta}{g} \right\rfloor = \left\lfloor \frac{a}{g} \right\rfloor \end{aligned}$$

□

Corolário 1. Para qualquer inteiro $a \in \mathbb{N}$ tem-se

$$\left\lceil \frac{a}{2} \right\rceil + \left\lfloor \frac{a}{2} \right\rfloor = a \quad , \quad a \in \mathbb{N}$$

Demonstração. A partir do resultado anterior tem-se para qualquer $a \in \mathbb{N}$:

$$\left\lceil \frac{a}{2} \right\rceil = \left\lceil a - \left\lfloor \frac{a}{2} \right\rfloor \right\rceil$$

Como o a e $\left\lfloor \frac{a}{2} \right\rfloor$ são inteiros deixa de ser necessária a notação $\lceil \cdot \rceil$:

$$\left\lceil \frac{a}{2} \right\rceil = a - \left\lfloor \frac{a}{2} \right\rfloor$$

e a prova fica concluída. □

Resultado 2. As desigualdades de Arredondamento, B_2 e C_2 , são equivalentes.

Demonstração. As desigualdades B_2 e C_2 são, respectivamente,

$$B_2 \quad \boxed{\sum_{i \in V} \sum_{d=2}^{D-1} \left\lfloor \frac{d}{2} \right\rfloor \cdot u_i^d \leq \left\lfloor \frac{n-2}{2} \right\rfloor}$$

e

$$C_2 \quad \boxed{\sum_{i \in V} \sum_{d=1}^{D-1} \left\lceil \frac{d}{2} \right\rceil \cdot u_i^d \geq \left\lceil \frac{n-2}{2} \right\rceil}$$

Partindo de B_2 e tendo em conta que $\left\lfloor \frac{1}{2} \right\rfloor = 0$ tem-se:

$$\sum_{i \in V} \sum_{d=1}^{D-1} \left\lfloor \frac{d}{2} \right\rfloor \cdot u_i^d \leq \left\lfloor \frac{n-2}{2} \right\rfloor \stackrel{\text{Cor.1}}{\Leftrightarrow} \sum_{i \in V} \sum_{d=1}^{D-1} \left(d - \left\lceil \frac{d}{2} \right\rceil \right) \cdot u_i^d \leq n-2 - \left\lceil \frac{n-2}{2} \right\rceil$$

Utilizando a igualdade (3.39), $\sum_{d=1}^{D-1} d \cdot \sum_{i \in V} u_i^d = n - 2$, a anterior desigualdade é equivalente a C_2 . O raciocínio inverso permite obter B_2 a partir de C_2 , concluindo-se assim a prova. \square

As seguintes demonstrações referem-se a resultados enunciados nas Secções 3.4 e 4.3. Para tal defina-se genericamente o conjunto (para $m \geq b + 1$)

$$\mathcal{X} = \{(\xi, \mu, \nu) \in \mathbb{B}^{m+2} : \sum_{j=1}^m \xi_j = b \cdot \mu + \nu, \nu \leq \mu\}$$

e o poliedro

$$\mathcal{P} = \{(\xi, \mu, \nu) \in [0, 1]^{m+2} : \sum_{j=1}^m \xi_j = b \cdot \mu + \nu, \nu \leq \mu, \xi_j \leq \mu, \forall j = 1, \dots, m\}$$

Qualquer um dos poliedros das secções referidas pode ser obtido a partir do poliedro genérico \mathcal{P} , através da seguinte substituição de parâmetros e variáveis:

- $m = |A^+(i)|$;
- $\xi_j \equiv x_{ij}^d, \forall j \in A^+(i)$;
- Para os poliedros $\mathcal{P}_{i,d}$ da Secção 3.4, $\mu \equiv u_i^d$ e:
 - ★ $\nu \equiv 0$ e $b = d$ para $i \neq r$;
 - ★ $\nu \equiv 0$ e $b = d + 1$ para $i = r$;
- Para os poliedros $\mathcal{P}_{i,d}^{nc}$ e $\mathcal{P}_{i,d}^c$ da Secção 4.3, $\mu \equiv y_i^d$ e:
 - ★ $\nu \equiv 0$ e $b = d - 1$ para $i \in V_{nc}$;
 - ★ $\nu \equiv x_{0i}^d$ e $b = d - 1$ para $i \in V_c$.

Resultado 3. *As seguintes desigualdades são válidas para \mathcal{X} e definem facetas do poliedro $\text{conv}(\mathcal{X})$:*

$$\xi_j \leq \mu, j = 1, \dots, m$$

Demonstração. As desigualdades são válidas para \mathcal{X} pois caso contrário existe um ponto tal que, para pelo menos uma das suas coordenadas $j = 1, \dots, m$, se verifica $1 = \xi_j > \mu = 0$. Nesse caso, tem-se, $\nu = 0$ o que implica que a igualdade $1 \leq \sum_{j=1}^m \xi_j = b \cdot \mu + \nu = 0$ é falsa. Note-se ainda que, o conjunto $\text{conv}(\mathcal{X})$ não é de dimensão completa ¹ devido à presença da igualdade $\sum_{j=1}^m \xi_j = b \cdot \mu + \nu$. De facto $\dim(\text{conv}(\mathcal{X})) = m + 1$ e como tal o número máximo de pontos *independentes afim* é $m + 2$.

Para mostrar que uma desigualdade válida arbitraria, $\xi_{j^*} \leq \mu$, define uma faceta ² do poliedro $\text{conv}(\mathcal{X})$, basta determinar $m + 1$ pontos de \mathcal{X} que satisfaçam $\xi_{j^*} = \mu$ e provar que são independentes afim. Em alternativa, uma forma de verificar a independência afim consiste em seleccionar ($t \geq m + 1$) pontos de \mathcal{X} que satisfaçam $\xi_{j^*} = \mu$, colocá-los num hiperplano genérico definido em \mathbb{B}^{m+2} , $a \cdot (\xi, \mu, \nu) = a_0$, e verificar que o único hiperplano que os contém é, à parte de uma constante positiva, o hiperplano definido por $\xi_{j^*} = \mu$.

Pelo facto de $\dim(\text{conv}(\mathcal{X})) = m + 1$, pelo menos um dos coeficientes do hiperplano genérico deve ser nulo, já que as facetas do poliedro $\text{conv}(\mathcal{X})$ têm dimensão m ; sem perda de generalidade, seja ele o coeficiente associado à variável ξ_{j^o} , ($j^o \neq j^*$).

¹Para o conjunto $\text{conv}(\mathcal{X})$ ser de dimensão completa, *i.e.* $\dim(\text{conv}(\mathcal{X})) = m + 2$, não pode existir nenhum hiperplano que contenha todos os pontos de $\text{conv}(\mathcal{X})$.

²Por definição, a dimensão de qualquer faceta de $\text{conv}(\mathcal{X})$ é $\dim(\text{conv}(\mathcal{X})) - 1 = m$.

Considerem-se agora os subconjuntos de $(b - 1)$ índices escolhidos de entre os índices $\{1, \dots, m\} \setminus \{j^*\}$. Designem-se genericamente estes subconjuntos por I_k , $k = 1, \dots, \binom{m-1}{b-1}$. Como estes conjuntos definem todas as diferentes combinações de $(b - 1)$ índices escolhidos entre $(m - 1)$, para cada par de conjuntos I_k e I_l com $k \neq l$ tem-se, $|I_k \cap I_l| \leq b - 2$. De facto, para qualquer conjunto I_k existe pelo menos um conjunto I_l ($k \neq l$) com exactamente $(b - 2)$ índices em comum. Recorrendo aos subconjuntos I_k é possível afirmar que os seguintes pontos pertencem a \mathcal{X} e satisfazem a igualdade $\xi_{j^*} = \mu$:

$$(\xi, \mu, \nu)^k = \begin{cases} (\mathbf{0}_m, 0, 0) & k = 0 \\ (e_{j^*} + \sum_{i \in I_k} e_i, 1, 0) & k = 1, \dots, \binom{m-1}{b-1} \end{cases}$$

onde $\mathbf{0}_m$ é o vector nulo de \mathbb{B}^m e e_i é o i -ésimo vector indicatriz ³ de \mathbb{B}^m . De facto, para os pontos $(\xi, \mu, \nu)^k$, $k \neq 0$, a soma $e_{j^*} + \sum_{i \in I_k} e_i$ é um vector de B^m com um "1" nas b posições, $j \in I_k \cup \{j^*\}$, e um "0" nas restantes.

Considere-se de novo o hiperplano genérico $a \cdot (\xi, \mu) = a_0$, com $a_{j^*} = 0$. Para que o ponto $(\xi, \mu, \nu)^0$ pertença a este hiperplano então $a_0 = 0$. Para que os restantes pontos $(\xi, \mu, \nu)^k$, $k = 1, \dots, \binom{m-1}{b-1}$, pertençam ao hiperplano, tem de se verificar (note-se que para estes pontos, o valor da variável ν é nulo):

$$a_{j^*} + \sum_{\substack{i \in I_k \\ i \neq j^*}} a_i + a_{m+1} + a_{m+2} \cdot 0 = a_0 = 0 \quad k = 1, \dots, \binom{m-1}{b-1} \quad (1.1)$$

Sejam I_p e I_q os conjuntos de índices associados a dois pontos quaisquer, $(\xi, \mu, \nu)^p$ e $(\xi, \mu, \nu)^q$, respectivamente, verificando as seguintes características:

³O vector $e_i = [(e_i)_j]$ é vector indicatriz, se e só se, $(e_i)_i = 1$ e $(e_i)_j = 0$, $j \neq i$.

- $|I_p \cap I_q| = b - 2$, $j^o \in I_p$;
- $I_q = I_p \setminus \{j^o\} \cup \{j^+\}$, para algum $j^+ \notin I_p$ e $j^+ \neq j^*$.

Os dois conjuntos I_p e I_q só diferem num índice: $j^o \in I_p \setminus I_q$ e $j^+ \in I_q \setminus I_p$. Para os dois pontos $(\xi, \mu, \nu)^p$ e $(\xi, \mu, \nu)^q$ considerados, as equações (1.1) são respectivamente:

$$a_{j^*} + \sum_{i \in I_p \setminus \{j^o\}} a_i + a_{m+1} = 0$$

$$a_{j^*} + \sum_{i \in I_q} a_i + a_{m+1} = a_{j^*} + \sum_{i \in I_p \setminus \{j^o\}} a_i + a_{j^+} + a_{m+1} = 0$$

o que permite concluir que $a_{j^+} = 0$. Como para qualquer conjunto I_k existe pelo menos um conjunto I_l com exactamente $(b - 2)$ índices em comum, diferindo apenas num índice, pode-se concluir que:

$$a_j = 0 \quad \forall j \neq j^*, m + 1$$

e portanto $a_{j^*} + a_{m+1} = 0$, o que faz com que o hiperplano seja:

$$a_{j^*} \xi_{j^*} - a_{j^*} \mu + a_{m+2} \nu = 0$$

Por outro lado, também o ponto $(e_{j^*} + e_{j^+} + \sum_{i \in I_1} e_i, 1, 1)$, com $j^+ \notin I_1 \cup \{j^*\}$, pertence a \mathcal{X} e satisfaz a igualdade $\xi_{j^*} = \mu$. Para que este ponto pertença ao hiperplano anterior então:

$$a_{j^*} - a_{j^*} + a_{m+2} = 0$$

Assim, o único hiperplano que contém todos os pontos considerados é:

$$a_{j^*} \xi_{j^*} - a_{j^*} \mu = 0$$

que é equivalente a $\xi_{j^*} = \mu$. Então a desigualdade $\xi_{j^*} \leq \mu$ define uma faceta do poliedro $\text{conv}(\mathcal{X})$. \square

Resultado 4. *O poliedro \mathcal{P} define o envolvente convexo do conjunto \mathcal{X} , i.e., $\text{conv}(\mathcal{X}) = \mathcal{P}$.*

Demonstração. Como as desigualdades $\xi_j \leq \mu, \forall j = 1, \dots, m$ são válidas para qualquer elemento de \mathcal{X} , então $\mathcal{X} \subset \text{conv}(\mathcal{X}) \subset \mathcal{P}$. Tendo em conta o domínio da variável μ , o conjunto \mathcal{X} pode ser particionado em dois conjuntos:

$$\mathcal{X} = \{(\mathbf{0}_m, 0, 0)\} \cup \mathcal{X}_1 \cup \mathcal{X}_2$$

em que os conjuntos \mathcal{X}_1 e \mathcal{X}_2 são definidos da seguinte forma:

$$\begin{aligned} \mathcal{X}_1 &= \{(\xi, 1, 1) : \xi \in \mathbb{B}^m, \sum_{j=1}^m \xi_j = b + 1\} \\ \mathcal{X}_2 &= \{(\xi, 1, 0) : \xi \in \mathbb{B}^m, \sum_{j=1}^m \xi_j = b\} \end{aligned}$$

Considerem-se ainda os poliedros $\mathcal{Q}_i, i = 1, 2$, onde se relaxa a condição de integralidade das variáveis ξ , i.e.:

$$\begin{aligned} \mathcal{Q}_1 &= \{(\xi, 1, 1) : \xi \in [0, 1]^m, \sum_{j=1}^m \xi_j = b + 1\} \\ \mathcal{Q}_2 &= \{(\xi, 1, 0) : \xi \in [0, 1]^m, \sum_{j=1}^m \xi_j = b\} \end{aligned}$$

Como a matriz das restrições de cada um destes poliedros é *totalmente unimodular* e o termo independente é inteiro, tem-se:

$$\text{conv}(\mathcal{X}_i) = \mathcal{Q}_i \quad i = 1, 2$$

Assim, o envolvente convexo de \mathcal{X} é dado por

$$\begin{aligned} \text{conv}(\mathcal{X}) &= \text{conv}\left(\{(\mathbf{0}_m, 0, 0)\} \cup \mathcal{X}_1 \cup \mathcal{X}_2\right) = \text{conv}\left(\{(\mathbf{0}_m, 0, 0)\} \cup \text{conv}(\mathcal{X}_1) \cup \right. \\ &\quad \left. \cup \text{conv}(\mathcal{X}_2)\right) = \text{conv}\left(\{(\mathbf{0}_m, 0, 0)\} \cup \mathcal{Q}_1 \cup \mathcal{Q}_2\right) = \\ &= \text{conv}\left(\{(\xi, \mu, \nu) \in [0, 1]^m \times \mathbb{B}^2 : \sum_{j=1}^m \xi_j = b \cdot \mu + \nu, \nu \leq \mu\}\right) \end{aligned}$$

Então, se o poliedro \mathcal{P} define o envolvente convexo de \mathcal{X} , também define o envolvente convexo de:

$$\{(\xi, \mu, \nu) \in [0, 1]^m \times \mathbb{B}^2 : \sum_{j=1}^m \xi_j = b \cdot \mu + \nu, \nu \leq \mu\}$$

Para o provar basta então mostrar que todos os pontos extremos de \mathcal{P} têm coordenadas inteiras associadas às variáveis μ e ν . Ou seja, qualquer ponto de \mathcal{P} com coordenada fraccionária associada a estas variáveis não pode ser ponto extremo de \mathcal{P} . Considere-se então um ponto arbitrário $(\check{\xi}, \check{\mu}, \check{\nu}) \in \mathcal{P}$ tal que ambas as coordenadas $\check{\mu}, \check{\nu}$ são fraccionárias. Note-se que o ponto $(\mathbf{0}_m, 0, 0) \in \mathcal{P}$. Além disso, pelas restrições do poliedro \mathcal{P} :

- $\sum_{j=1}^m \check{\xi}_j = b \cdot \check{\mu} + \check{\nu} \Rightarrow \sum_{j=1}^m \frac{\check{\xi}_j}{\check{\mu}} = b \cdot 1 + \frac{\check{\nu}}{\check{\mu}}$
- $\check{\nu}_j \leq \check{\mu} \Rightarrow 0 \leq \frac{\check{\nu}_j}{\check{\mu}} \leq 1$
- $0 \leq \check{\xi}_j \leq \check{\mu}, \forall j = 1, \dots, m \Rightarrow 0 \leq \frac{\check{\xi}_j}{\check{\mu}} \leq 1, \forall j = 1, \dots, m$

E assim, também o ponto $\left(\frac{\check{\xi}_1}{\check{\mu}}, \frac{\check{\xi}_2}{\check{\mu}}, \dots, \frac{\check{\xi}_m}{\check{\mu}}, 1, \frac{\check{\nu}}{\check{\mu}}\right)$ pertence a \mathcal{P} .

Mas sendo assim, tem-se:

$$(\check{\xi}, \check{\mu}, \check{\nu}) = (1 - \check{\mu}) \cdot (\mathbf{0}_m, 0, 0) + \check{\mu} \cdot \left(\frac{\check{\xi}_1}{\check{\mu}}, \frac{\check{\xi}_2}{\check{\mu}}, \dots, \frac{\check{\xi}_m}{\check{\mu}}, 1, \frac{\check{\nu}}{\check{\mu}} \right)$$

ou seja, o ponto $(\check{\xi}, \check{\mu}, \check{\nu})$ é uma combinação linear convexa de dois pontos de \mathcal{P} , logo não poderá ser ponto extremo de \mathcal{P} .

Para um ponto arbitrário com $\check{\nu} = 0$ e $\check{\mu} \in]0, 1[$ o raciocínio anterior mantém-se. Se $\check{\nu} = 1$ então $\check{\mu} = 1$ e o ponto tem coordenadas inteiras nas variáveis μ e ν , o mesmo acontecendo se $\check{\mu} = 0$ pois nesse caso tem-se $\check{\nu} = 0$. Finalmente se $\check{\mu} = 1$ e $\check{\nu} \in]0, 1[$, pode-se demonstrar de uma forma análoga que o ponto arbitrário $(\check{\xi}, 1, \check{\nu})$ é combinação linear convexa dos pontos $(\mathbf{0}_m, 0, 0)$ e $\left(\frac{\check{\xi}_1}{\check{\nu}}, \frac{\check{\xi}_2}{\check{\nu}}, \dots, \frac{\check{\xi}_m}{\check{\nu}}, \frac{1}{\check{\nu}}, 1 \right)$, ambos pertencentes a \mathcal{P} .

Assim, fica provado que todos os pontos extremos de \mathcal{P} , tem coordenadas inteiras nas variáveis μ e ν . Então:

$$\begin{aligned} \mathcal{P} &= \text{conv} \left(\left\{ (\xi, \mu, \nu) \in [0, 1]^m \times \mathbb{B}^2 : \sum_{j=1}^m \xi_j = b \cdot \mu + \nu, n\mu \leq \mu \right\} \right) = \\ &= \text{conv}(\mathcal{X}) \end{aligned}$$

Uma prova alternativa consiste em mostrar que, para qualquer vector de custos $(c, f, g) \in \mathbb{R}^{m+2}$ o programa linear $\max \{ c \cdot \xi + f \cdot \mu + g \cdot \nu : (\xi, \mu, \nu) \in \mathcal{P} \}$ tem uma solução óptima (ξ^*, μ^*, ν^*) em \mathcal{X} . Para tal ordene-se o vector de custos c_j por ordem decrescente e considere-se $c_{j:m}$ o j -ésimo custo no vector ordenado. Caso $\mu = 0$ então $\nu = 0$, $\xi_j = 0$, $\forall j = 1, \dots, m$ e o custo da solução será 0.

Caso $\mu > 0$, a solução óptima do programa linear consiste em fazer:

$$\begin{aligned}\xi_{j:m} &= \mu & j &= 1, \dots, b \\ \xi_{b+1:m} &= \nu \\ \xi_{j:m} &= 0 & j &= b+2, \dots, m\end{aligned}$$

e o custo desta solução será $\left(\sum_{j=1}^b c_{j:m} + f\right) \cdot \mu + \left(c_{b+1:m} + g\right) \cdot \nu$. Caso $c_{b+1:m} + g > 0$, o valor da variável ν é otimizado fazendo $\nu^* = \mu^*$. Caso contrário, $\nu^* = 0$. Assim sendo, o valor de qualquer solução do programa linear é dado por, $\left(\sum_{j=1}^b c_{j:m} + f + \max(0, c_{b+1:m} + g)\right) \cdot \mu$.

Se $\left(\sum_{j=1}^b c_{j:m} + f + \max(0, c_{b+1:m} + g)\right) > 0$ o valor da solução é otimizado fazendo $\mu^* = 1$; caso contrário $\mu^* = \nu^* = 0$. No caso em que $\mu^* = 1$, se adicionalmente se tiver $(c_{b+1:m} + g) > 0$, ter-se-á $\nu^* = 1$; caso contrário $\nu^* = 0$. Conclusão, para qualquer vector de custos $(c, f, g) \in \mathbb{R}^{m+2}$, existe sempre uma solução óptima (ξ^*, μ^*, ν^*) em que o valor das variáveis μ e ν é inteiro e o valor óptimo do programa linear é dado por

$$\max \left\{ 0, \sum_{j=1}^b c_{j:m} + f + \max(0, c_{b+1:m} + g) \right\}$$

Note-se que, sendo a solução óptima inteira nas variáveis μ e ν também o será nas variáveis $\xi_j, \forall j = 1, \dots, m$. \square

Nos dois últimos resultados assumiu-se que $m \geq b + 1$. Caso $m = b$, então só existem duas soluções inteiras, $\mathcal{X} = \{(\mathbf{0}_m, 0, 0), (\mathbf{1}_m, 1, 0)\}$, em que $\mathbf{1}_m$ representa o vector unitário de \mathbb{B}^m . Caso $m < b$, então só existe uma solução inteira $\mathcal{X} = \text{conv}(\mathcal{X}) = \mathcal{P} = \{(\mathbf{0}_m, 0, 0)\}$. Para qualquer um destes conjuntos as provas anteriores mantêm-se .

Apêndice B

Geração de instâncias

Nesta secção descreve-se a geração das instâncias para os problemas (ASupCG) e (AStRPCG). A geração das segundas é em tudo semelhante à geração das primeiras, excepto quanto à geração de nodos cliente e respectivos prémios.

B.1 A distribuição espacial dos nodos

Testaram-se instâncias com 25 e 50 nodos dispersos aleatoriamente numa grelha quadrada, cuja dimensão depende do número de nodos nela contidos. Para compreender melhor a escolha da dimensão da grelha em função do número de nodos considere-se esta definida num plano ortonormado, onde dois dos nodos opostos da grelha são os pontos com coordenadas $(0, 0)$ e (L, L) ($L \in \mathbb{N}$). Assumindo que os nodos do grafo apenas podem ocupar pontos com coordenadas inteiras, então existem $(L + 1)^2$ pontos, dentro da grelha, onde podem ser dispostos os nodos do grafo. Considere-se ainda uma medida de concentração, *conc*, que permite avaliar a concentração/dispersão do número de nodos na respectiva grelha, ou seja $conc = \frac{|V|}{(L+1)^2}$.

Assim, consoante o parâmetro *conc* escolhido, a abcissa e ordenada máximas de qualquer ponto na grelha será dado por:

$$L = \left\lceil \sqrt{\frac{|V|}{conc}} - 1 \right\rceil$$

Após a definição da dimensão da grelha, os nodos foram distribuídos aleatoriamente na grelha, garantindo apenas que não existem nodos coincidentes, *i.e.*, a distância entre os nodos, $dist_{ij}$, é estritamente positiva.

B.2 A densidade do grafo

Em termos da densidade de cada grafo gerado, optou-se por testar diferentes valores consoante o número de nodos do grafo. Isto prende-se com o facto de que, se por um lado uma densidade alta numa rede com um grande número de nodos, não reflecte a maioria das situações reais, por outro lado quanto menor o número de arestas existentes no grafo, menos provável será que qualquer nodo tenha o máximo número de ligações permitidas, D . Ou seja, corre-se o risco de a própria densidade do grafo actuar como uma restrição ao número de ligações de cada nodo. Para poder avaliar a influência do parâmetro *grau máximo*, garantindo uma influência mínima indirecta causada pelo parâmetro densidade, cada nodo no grafo deverá ter em média, um grau pelo menos igual ao maior valor do parâmetro *grau máximo* testado. Como será descrito mais à frente, este valor será $D = 6$. O grau médio dos nodos no grafo é dado pela soma do grau de todos os nodos a dividir pelo número de nodos, *i.e.*, $\sum_{i \in V} d(i)/|V| = 2 \cdot |E|/|V|$. Assim, para garantir pelo menos um grau médio igual a 6 em todos os nodos o número de arestas geradas deve ser $|E| \geq 3 \cdot |V|$.

Nas instâncias utilizadas para os problemas (ASupCG) e (AStRPCG) testaram-se dois valores para o número de arestas geradas, $3 \cdot |V|$ e $6 \cdot |V|$, apresentando-se na Tabela B.1 as várias combinações. Além disso, testaram-se duas dimensões para o conjunto de nodos clientes, 25% e 50% do número total de nodos na rede (valor arredondado para cima). Assim, para as instâncias de 25 nodos, o número de nodos clientes é 7 ou 13 e para as instâncias de 50 nodos o número de nodos clientes é 13 ou 25.

$ V $	L	$dist_{ij} \in [,]$	$ E $	$dens$	Grau médio
25	15	[1, 21]	75	25%	6
			150	50%	12
50	22	[1, 31]	150	13%	6
			300	25%	12

Tabela B.1: Dimensão das grelhas e densidades utilizadas na geração das instâncias

B.3 Custos das arestas

Após gerada uma instância na respectiva grelha, é calculada a distância euclidiana entre os nodos extremos de cada aresta. O custo da aresta é então definido em função do tipo de rede a testar: *Wireless* ou *Cabo*. Nas redes *Wireless*, o custo de cada aresta depende da proximidade dos nodos que a definem, visto depender da força que é necessário imprimir ao sinal de transmissão. Como tal, para ligar nodos mais afastados será necessário usar uma maior potência de sinal utilizando para isso equipamento de transmissão mais caro. Assim, para um nodo i , o custo associado a qualquer aresta $e = \{i, j\}$ será o mesmo para todos os nodos j cuja distância ao nodo i se encontre dentro do mesmo intervalo.

Nas instâncias geradas o custo de cada aresta é definido como:

$$c_e = \begin{cases} 5 \cdot dist_{ij} & \text{se } dist_{ij} \leq 5 \\ 9 \cdot dist_{ij} & \text{se } 5 < dist_{ij} \leq 15 \\ 12 \cdot dist_{ij} & \text{se } dist_{ij} > 15 \end{cases}$$

No caso das redes *Cabo*, onde se assume que é necessário construir um cabo de ligação física entre os dois nodos, o custo de cada aresta é directamente proporcional à distância euclidiana, sendo calculado como ¹:

$$c_e = 10 \cdot dist_{ij}$$

B.4 Prémios dos nodos clientes

No problema (AStRPCG) os prémios dos nodos clientes são gerados uniformemente no intervalo [51, 100] para as redes *Wireless*. Para as redes *Cabo*, os custos das ligações são mais caros e como tal os prémios terão de ser mais elevados para evitar soluções "vazias". Assim, para estas redes os prémios são gerados uniformemente no intervalo [251, 300].

B.5 Os parâmetros Q , D e (K_1, K_2)

Escolheu-se o valor $Q = 3$ para a dimensão dos módulos por este ser um valor usual para a dimensão de módulos. Para o grau máximo em cada nodo, testaram-se 4 valores: $D = 3, 4, 5, 6$. Quanto aos custos tecnológicos, foram testados dois pares de valores onde o custo fixo (da matriz de *routing*) é o mesmo, variando apenas o custo de cada módulo, reflectindo um *ratio* de 10 e 2, respectivamente, entre estes dois custos: $(K_1, K_2) = (100, 10), (100, 50)$.

¹O valor 10 representa o custo do cabo por unidade de comprimento.

Apêndice C

Escolha do modelo Inteiro

Todos os modelos testados nos resultados computacionais apresentados ao longo da dissertação foram implementados em *C++*. As respectivas soluções óptimas foram obtidas utilizando o *software* CPLEX 11.0/Concert Technology 2.5 da ILOG num computador com um processador INTEL CORE 2 - 2.4GHz com 3.327 Gb de memória RAM. No caso dos modelos inteiros, o tempo máximo de execução do algoritmo de *Branch-&Bound* é de 2 horas.

C.1 O problema (ASupCG)

Na escolha do modelo a utilizar na determinação da solução óptima inteira de todas as instâncias do (ASupCG), seleccionaram-se alguns modelos para comparação, em termos dos tempos de CPU gastos na obtenção da solução. Escolheram-se para esse efeito as duas instâncias *Wireless* com 25 nodos e 75 e 150 arestas, respectivamente (para todas as combinações dos parâmetros D e (K_1, K_2) descritos no Apêndice B).

Para esta comparação, de entre todos os modelos implementados, foram escolhidos os 4 melhores em termos do limite inferior fornecido pela respectiva relaxação linear: dois modelos Discretizados, $(SUP_D + A_1, BC)$ e $(SUP_D + AD, BC)$, e dois modelos Árvore de Suporte/Saco Mochila, $(SUP/SM_{CD2} + A_1)$ e $(SUP/SM_{CD2} + AD)$. Utilizou-se ainda o primeiro modelo linear Árvore de Suporte/Saco Mochila apresentado na Secção 3.5.1, (SUP/SM_C) . Este modelo utiliza as variáveis inteiras, U_i , e não as variáveis discretizadas, u_i^d . Em virtude de ter menos variáveis que os outros modelos Árvore de Suporte/Saco Mochila foi também eleito para comparação. Os 5 modelos inteiros foram então implementados utilizando qualquer uma das duas versões do sistema de fluxos: SC fortalecido com as desigualdades (1.10) e o sistema MC .

Por forma a acelerar a execução do algoritmo de *Branch-é-Bound* forneceram-se ao *software* CPLEX (através do parâmetro `IloCplex::CutUp`) limites superiores ao valor óptimo do problema. Estes limites superiores correspondem ao valor de soluções admissíveis para o (ASupCG) determinadas através de uma meta-heurística GRASP para o problema implementada por Duhamel e Souza [19].

A Tabela C.1 apresenta os tempos de CPU gastos por cada um dos modelos. As primeiras 4 colunas indicam, respectivamente, o tipo de rede, o densidades da rede em termos do número de arestas, o par de custos tecnológicos e o grau máximo em cada nodo. Como seria de esperar, a utilização do sistema de fluxos MC origina maiores tempos de CPU (em algumas instâncias com grau máximo $D = 6$ atinge-se mesmo o tempo máximo estipulado). A razão de tal facto prende-se com o maior número de variáveis e restrições associadas ao sistema de fluxos, quando comparado com o sistema SC^* .

R	$ E $	(K_1, K_2)	D	$(SUP_D + A_1, BC)$		$(SUP_D + AD, BC)$		$(SUP/SM_{CD2} + A_1)$		$(SUP/SM_{CD2} + AD)$		$(SUP/SM_C + A_1)$	
				SC^*	MC	SC^*	MC	SC^*	MC	SC^*	MC	SC^*	MC
	(100, 10)		3	0	1	0	0	1	5	4	2	1	4
			4	0	5	1	3	1	13	2	9	1	12
			5	1	5	1	12	2	26	3	38	2	7
			6	5	70	13	669	43	257	61	959	23	109
75	(100, 50)		3	0	0	0	0	1	5	3	2	2	3
			4	1	3	1	3	1	11	3	6	1	13
			5	1	6	1	14	5	86	4	114	3	38
			6	6	395	40	254	37	215	50	438	31	140
W	(100, 10)		3	0	2	1	2	1	11	3	5	3	2
			4	5	59	4	84	35	136	8	107	8	136
			5	1	25	2	116	9	49	3	29	3	30
			6	28	6652	329	MAX	15	125	20	135	9	121
150	(100, 50)		3	0	2	0	2	3	7	3	4	15	2
			4	2	54	4	132	12	292	9	119	10	335
			5	1	40	2	40	4	38	13	43	3	40
			6	MAX	MAX	138	MAX	12	197	20	142	12	128

Tabela C.1: Comparação dos tempos de CPU de modelos inteiros para o (ASupCG).

Quer entre os modelos Discretizados, quer entre os modelos Árvore de Suporte/Saco Mochila, o modelo com desigualdades *Arco* adicionadas é, na maioria dos casos, mais rápido do que o modelo com desigualdades *Arco Desagregadas*. Pela rapidez de execução, escolheu-se o modelo (SUP/SM_C) com o sistema de fluxos SC^* para determinar a solução óptima inteira de todas as instâncias de teste utilizadas nos resultados do Capítulo 3.

C.2 O problema (AStRPCG)

Para o problema (AStRPCG) seleccionaram-se 2 modelos, o modelo $(StP_D + A_1)$ e o modelo $(StP_D + AD)$. Estes dois modelos foram comparados com a respectiva versão fortalecida com as desigualdades de *Assimetria* apresentadas na Secção 1.1.3.

Cada um dos 4 modelos assim obtido foi implementado utilizando qualquer uma das duas versões do sistema de fluxos: SC fortalecido com as desigualdades (1.10) e o sistema MC . A Tabela C.2 apresenta os resultados obtidos com duas instâncias de 25 nodos com 13 nodos clientes.

R	$ E $	nc	(K_1, K_2)	D	$(StP_D + A_1)$				$(StP_D + AD)$				
					SC^*	SC_a^*	MC	MC_a	SC^*	SC_a^*	MC	MC_a	
	(100, 10)	75	13	3	0	0	0	0	0	0	0	0	0
				4	1	1	7	1	2	1	7	4	
				5	1	1	2	1	6	2	9	2	
				6	0	0	1	0	0	0	1	0	
	(100, 50)	75	13	3	0	0	0	0	0	0	0	0	0
				4	0	0	0	0	0	0	0	0	0
				5	0	0	0	0	0	0	1	0	
				6	0	0	1	0	0	0	0	0	
W	(100, 10)	150	13	3	0	0	1	0	0	0	4	0	
				4	41	7	513	32	68	8	477	85	
				5	11	5	291	15	58	16	486	36	
				6	49	4	328	30	108	9	612	51	
	(100, 50)	150	13	3	0	0	1	0	0	0	1	0	
				4	1	0	2	1	1	0	2	0	
				5	6	2	20	12	7	2	51	5	
				6	3	1	20	3	5	1	41	4	

Tabela C.2: Comparação dos tempos de CPU de modelos inteiros para o (AStRPCG).

A adição das desigualdades de *Assimetria* claramente reduz o tempo de execução do algoritmo de *Branch-&-Bound*, assim como a utilização do sistema de fluxos SC^* . Entre os modelos $(StP_D + A_1)$ e $(StP_D + AD)$ não parece haver diferença significativa em termos de tempos de CPU. No entanto escolheu-se o modelo $(StP_D + A_1)$ fortalecido com as desigualdades de *Assimetria* e utilizando o sistema de fluxos SC^* para obter a solução óptima de todas as instâncias de teste utilizadas nos resultados do Capítulo 4.

Apêndice D

Resultados Completos

Neste Apêndice reúnem-se os valores óptimos e tempos de CPU de todos os modelos testados ao longo da dissertação, quer sejam modelos inteiros ou relaxações lineares.

Todos os resultados apresentados são obtidos num computador com um processador INTEL CORE 2 - 2.4GHz com 3.327 Gb de memória RAM, utilizando o software CPLEX 11.0/Concert Technology 2.5 da ILOG. O tempo máximo de execução do algoritmo de *Branch-ℳ-Bound* é de 2 horas.

Nas Tabelas D.1 e D.2 apresentam-se os valores associados às instâncias do (ASupCG), para 25 e 50 nodos respectivamente. Nas Tabelas D.3 e D.4 apresentam-se os valores associados às instâncias do (AStRPCG), para 25 e 50 nodos respectivamente.

		(LSUP_D)			(LSUP_D+A_1)			(LSUP_D+A_2)			(LSUP_D+AD)			(LSUP_D+BC)			(LSUP_D+A_1BC)			(LSUP_D+AD,BC)			(LSUP/SM_CD2)			(LSUP/SM_CD2+A_1)			(LSUP/SM_CD2+AD)			(SUP/SM_CD2+A_1)								
T	IEI	(K_1,K_2)	V		T		V		T		V		T		V		T		V		T		V		T		V		T		V		T							
W	75	(100,10)	3	1410,348	0	1414	0	1417	0	1414,474	0	1417	0	1414,483	0	1417	0	1414,483	0	1417	0	1469	0	1472	0	1469,021	0	1472	0	1469,021	0	1472	0	1469,021	0	1472	0	1		
			150	(100,10)	3	1870,348	0	1874	0	1877	0	1874,474	0	1877	0	1874,483	0	1877	0	1874,483	0	1877	0	1949	0	1952	0	1949,021	0	1952	0	1949,021	0	1952	0	1949,021	0	1952	0	2

Figura D.1: Resultados de todos os modelos para as instâncias de 25 nodos para o (ASupCG).

Table with 4 columns: T, IEI, (K,C), and a grid of values for various models (LSUP_D, LSUP_D+A,1, LSUP_D+A,2, LSUP_D+AD, LSUP_D+BC, LSUP_D+AD,1,BC, LSUP_D+AD,BC, LSUP/SM_CD2, LSUP/SM_CD2+A,1, LSUP/SM_CD2+AD, LSUP/SM_CD2+A,1). Rows are grouped by W (150, 300) and C (150, 300).

Figura D.2: Resultados de todos os modelos para as instâncias de 50 nodos para o (ASupCG).

Lista de Figuras

1.1	(ASup): Modelo genérico.	3
1.2	(ASup): Sistema de fluxos <i>SC</i>	8
1.3	(ASup): Sistema de fluxos <i>MC</i>	10
1.4	(AStRP): Modelo genérico.	19
1.5	(AStRP): Sistema de fluxos <i>SC</i>	23
1.6	(AStRP): Sistema de fluxos <i>MC</i>	24
1.7	Reformulação por caminhos: construção do grafo expandido.	33
2.1	Função de custos de grau: exemplo.	38
2.2	Aplicação da restrição de grau e dos custos de grau: exemplo.	39
3.1	(ASupCG): Modelo não linear.	43
3.2	(ASupCG): Modelo Linear com variáveis-módulo discretizadas: Modelo de Escolha Múltipla.	46
3.3	(ASupCG): Modelo Linear com variáveis-módulo discretizadas: Modelo Incremental.	51
3.4	(ASupCG): Modelo Linear com variáveis-grau discretizadas.	58
3.5	(ASupCG): Sistema linear de definição das variáveis x_{ij}^d	70
3.6	Comparação do modelo discretizado com o modelo fortalecido com desigualdades <i>Arco Desagregadas</i>	75

3.7	Comparação dos modelos fortalecidos com desigualdades <i>Arco</i> e desigualdades <i>Arco Desagregadas</i>	84
3.8	(ASupCG): Modelo Árvore de Suporte/Saco Mochila não linear.	91
3.9	Exemplo de um grafo expandido.	94
3.10	Sistema de equações de caminho no grafo expandido.	95
3.11	(ASupCG): Modelo Árvore de Suporte/Saco Mochila.	97
3.12	(ASupCG): Modelo Árvore de Suporte/Saco Mochila nas variáveis discretizadas.	98
3.13	(ASupCG): Modelo desagregado Árvore de Suporte/Saco Mochila nas variáveis discretizadas.	100
3.14	Relação entre as relaxações lineares dos modelos (SUP/SM_{CD1}) e (SUP/SM_{CD2}).	101
3.15	Pontos extremos fracionários na relaxação linear do problema de Saco Mochila de Escolha Múltipla.	106
3.16	Comparação dos modelos apresentados.	122
4.1	(AStRPCG): Modelo genérico não linear.	126
4.2	(PAStPRCG): Modelo Linear com variáveis-grau discretizadas.	129
4.3	(AStRPCG): Sistema linear de definição das variáveis x_{ij}^d	135
D.1	Resultados completos para as instâncias de 25 nodos para o (ASupCG).	170
D.2	Resultados completos para as instâncias de 50 nodos para o (ASupCG).	171
D.3	Resultados completos para as instâncias de 25 nodos para o (AStRPCG).	172
D.4	Resultados completos para as instâncias de 50 nodos para o (AStRPCG).	173

Lista de Tabelas

3.1	Resultados para instâncias com 25 nodos: Modelo ($LSUP_D$). . .	66
3.2	Resultados para instâncias com 50 nodos: Modelo ($LSUP_D$). . .	67
3.3	Resultados para instâncias com 25 nodos: Modelos fortalecidos.	88
3.4	Resultados para instâncias com 50 nodos: Modelos fortalecidos.	89
3.5	Comparação entre modelos Discretizados e modelos Árvore de Suporte/Saco Mochila: exemplo.	109
3.6	Resultados para instâncias com 25 nodos: Modelos (SUP/SM_{CD2}) vs Modelos (SUP_D).	111
3.7	Resultados para instâncias com 50 nodos: Modelos (SUP/SM_{CD2}) vs Modelos (SUP_D).	112
3.8	Comparação entre modelos fortalecidos com desigualdades de <i>Arredondamento</i> e modelo (SUP/SM_{CD2}): exemplo.	118
3.9	Resultados para instâncias com 25 nodos: Modelos fortalecidos com desigualdades de <i>Arredondamento</i>	120
3.10	Resultados para instâncias com 50 nodos: Modelos fortalecidos com desigualdades de <i>Arredondamento</i>	121
4.1	Resultados para instâncias com 25 nodos: Modelo ($LStP_D$). . .	132
4.2	Resultados para instâncias com 50 nodos: Modelo ($LStP_D$). . .	133
4.3	Resultados para instâncias com 25 nodos: Modelos fortalecidos.	145

4.4	Resultados para instâncias com 50 nodos: Modelos fortalecidos.	146
B.1	Dimensão das grelhas e densidades utilizadas na geração das instâncias.	163
C.1	Comparação dos tempos de CPU de modelos inteiros para o (ASupCG).	167
C.2	Comparação dos tempos de CPU de modelos inteiros para o (AStRPCG).	168

Bibliografia

- [1] R. Ahuja, T. Magnanti, and J. Orlin. *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall, New York, 1993.
- [2] A. Almeida, P. Martins, and M. Souza. Min-degree constrained minimum spanning tree problem: complexity, proprieties and formulations. Working Paper 6, Centro de Investigação Operacional, 2006.
- [3] E. Balas. The prize collecting traveling salesman problem. *Networks*, 19:621–636, 1989.
- [4] P. Belotti, L. Brunetta, and F. Malucelli. Multicommodity network design with discrete node costs. *Networks*, 49:90–99, 2007.
- [5] S. Canuto, M. Resende, and C. Ribeiro. Local search with perturbations for the prize-collecting steiner tree problem in graphs. *Networks*, 38(1):50–58, 2001.
- [6] O. Chapovska and A. Punnen. Variations of the prize-collecting steiner tree problem. *Networks*, 47(4):199–205, 2006.
- [7] M. Chimani, M. Kandyba, I. Ljubic, and P. Mutzel. Obtaining optimal k-cardinality trees fast. In *ALLENEX, Workshop on Algorithm Engineering & Experiments*, pages 27–36, 2008.

-
- [8] M. Constantino and L. Gouveia. Reformulation by discretization: application to economic lot sizing. *to appear in OR letters*, 2007.
- [9] I. Correia, L. Gouveia, and F. S. da Gama. Solving the variable size bin packing problem with discretized formulations. Working paper 5, Centro de Investigação Operacional, 2006.
- [10] A. M. Costa, J.-F. Cordeau, and G. Laporte. Steiner tree problems with profits. *INFOR*, 44(2):99–115, 2006.
- [11] A. M. Costa, J.-F. Cordeau, and G. Laporte. Models and branch-and-cut algorithms for the steiner tree problem with revenues, budget and hop constraints. *Networks*, 53(2):141–159, 2009.
- [12] R. Courant, H. Robbins, and I. Stewart. *What is mathematics?: an elementary approach to ideas and methods*. Oxford University Press US, 2nd edition, 1996.
- [13] K. Croxton, B. Gendron, and T. Magnanti. A comparison of mixed-integer programming models for non-convex piecewise linear cost minimization problems. *Management Science*, 49:1268–1273, 2003.
- [14] K. Croxton, B. Gendron, and T. Magnanti. Variable disaggregation in network flow problems with piecewise linear costs. *Operations Research*, 55:146–157, 2007.
- [15] A. Cunha and A. Lucena. Algorithms for the degree-constrained minimum spanning tree problem. *Electronic Notes in Discrete Applied Mathematics*, 19:403–409, 2005.
- [16] G. Dahl, L. Gouveia, and C. Requejo. *Handbook of Optimization in Telecommunications*, chapter On formulations and methods for

- the hop-constrained minimum spanning tree problem, pages 493–515. Birkhäuser, 2006.
- [17] S. Dreyfus and R. Wagner. The steiner problem in graphs. *Networks*, 1:195–207, 1972.
- [18] C. Duhamel, L. Gouveia, P. Moura, and M. Souza. Minimum cost degree constrained spanning trees with node-dependent costs. Working paper 8/2007, CIO - Centro de Investigação Operacional, Faculdade de Ciências da universidade de Lisboa, 2007.
- [19] C. Duhamel and M. Souza. A grasp heuristic for the node degree constrained minimum spanning tree problem. In *International Network Optimization Conference*, Spa, Bélgica, 2007.
- [20] E. Estrela, M. A. Soares, and M. J. Leitão. *Saber escrever uma tese e outros textos*. Dom Quitoxe, 5ª edition, Julho 2007.
- [21] S. Fekete, S. Khuller, M. Klemstein, B. Raghavachari, and N. Young. A network-flow technique for finding low-wieght bounded-degree spanning trees. *Journal of Algorithms*, 24:310–324, 1997.
- [22] K. Fox, B. Gavish, and S. Graves. A n-constraint formulation for the (time-dependent) traveling salesman problem. *Operations Research*, 28:1018–1021, 1980.
- [23] A. Frangioni and B. Gendron. 0-1 reformulations of the network loading problem. In *INOC 2005*, 2005.
- [24] H. Gabow. A good algorithm for smallest spanning trees with a degree constraint. *Networks*, 8:201–208, 1978.

-
- [25] M. Garey, R. Graham, and D. Johnson. The complexity of computing steiner minimal trees. *SIAM Journal on Applied Mathematics*, 32:835–859, 1977.
- [26] M. Garey and D. Johnson. *Computers and Intractability: A guide to the theory of \mathcal{NP} -completeness*. W.H. Freeman, New York, 1979.
- [27] E. Gilbert and H. Pollak. Steiner minimal trees. *SIAM Journal on Applied Mathematics*, 16:1–29, 1968.
- [28] F. Glover and D. Klingman. Finding minimum spanning trees with a fixed number of links at a node. Research report 74-5, Center for Cybernetic Studies, University of Texas at Austin, Austin, Texas, 1974.
- [29] M. Goemans. The steiner tree polytope and related polyhedra. *Mathematical Programming*, 63:157–182, 1994.
- [30] M. Goemans and Y. Myung. A catalog of steiner tree formulations. *Networks*, 23:19–28, 1993.
- [31] M. Goemans and D. Williamson. A general approximation technique for constrained forest problems. *SIAM Journal on Computing*, 24(2):296–317, 1995.
- [32] L. Gouveia. A $2n$ constraint formulation for the capacitated minimal spanning tree problem. *Operations Research*, 43:130–141, 1995.
- [33] L. Gouveia and F. S. da Gama. On the capacitated concentrator location problem: a reformulation by discretization. *Computers and Operations Research*, 33:1242–1258, 2006.

-
- [34] L. Gouveia and T. L. Magnanti. Network flow models for designing diameter-constrained minimum-spanning and steiner trees. *Networks*, 41(3):159–173, 2003.
- [35] L. Gouveia and C. Requejo. A new lagrangean relaxation approach for the hop-constrained minimum spanning tree problem. *European Journal of Operational Research*, 132:539–552, 2001.
- [36] H. Kellerer, U. Pferschy, and D. Pisinger. *Knapsack Problems*. Springer, Berlin, Germany, 2004.
- [37] S. Khuller, B. Raghavachari, and N. Young. Low-degree spanning trees of small weight. *SIAM J. Comput.*, 25(2), 1996.
- [38] T. Koch and A. Martin. Solving steiner tree problems in graphs to optimality. *Networks*, 32:207–232, 1998.
- [39] M. Krishnamoorthy, A. Ernst, and Y. Sharaiha. Comparison of algorithms for the degree constrained minimum spanning tree. *Journal of Heuristics*, 7(6):587–611, 2001.
- [40] J. Kruskal. On the shortest spanning subtree of a graph and the travelling salesman problem. In *Proceedings of the American Mathematics Society*, volume 7, pages 48–50, 1956.
- [41] I. Ljubic, R. Weiskircher, U. Pferschy, G. Klau, P. Mutzel, and M. Fischetti. An algorithmic framework for the exact solution of the prize-collecting steiner tree problem. *Mathematical Programming*, 105(2-3):427–449, 2006.

-
- [42] A. Lucena and M. Resende. Strong lower bounds for the prize-collecting steiner problem in graphs. *Discrete Applied Mathematics*, 141:277–294, 2004.
- [43] T. Magnanti and L. Wolsey. *Handbooks in Operational Research and Management Science*, chapter Optimal Trees. M.O. Ball and T.L. Magnanti and C.L. Monma and G.L. Nemhauser, 1995.
- [44] S. Martello and P. Toth. *Knapsack Problems: Algorithms and Computer Implementations*. John Wiley & Sons, 1990.
- [45] R. Martin. *Large scale linear and integer optimization: a unified approach*. Kluwer Academic Publishers, 1999.
- [46] C. Monma and S. Suri. Transitions in geometric minimum spanning trees. *Discrete & Computational Geometry*, 8(3):265–293, 1992.
- [47] S. Narula and C. Ho. Degree-constrained minimum spanning trees. *Computers and Operations Research*, 7:239–249, 1980.
- [48] C. Papadimitriou. The complexity of the capacitated tree problem. *Networks*, 8:217–230, 1978.
- [49] C. Papadimitriou and U. Vazirani. On two geometric problems related to the traveling salesman problem. *Journal of Algorithms*, 5(2):231–246, 1984.
- [50] R. Prim. Shortest connection networks and some generalizations. *Bell System Technical Journal*, 36:1389–1401, 1957.
- [51] R. L. Rardin and U. Choe. Tighter relaxations of fixed charge network flow problems. Technical report, School of Industrial and Systems Engineering, Georgia Institute of Technology, 1979.

-
- [52] M. Savelsbergh and T. Volgenant. Edge exchanges in the degree-constrained minimum spanning tree problem. *Computers and Operations Research*, 12(4):341–348, 1985.
- [53] A. Segev. The node-weighted steiner tree problem. *Networks*, 17(1):1–17, 1987.
- [54] A. Singh and A. K. Gupta. Improved heuristics for the bounded-diameter minimum spanning tree problem. *Soft Computing - A Fusion of Foundations, Methodologies and Applications*, 11(10):911–921, August 2007.
- [55] E. Uchoa. Reduction tests for the prize-collecting steiner problem. *Operations Research Letters*, 34:437–444, 2006.
- [56] E. Uchoa, R. Fukasawa, J. Lysgaard, A. Pessoa, M. P. de Aragão, and D. Andrade. Robust branch-cut-and-price for the capacitated minimum spanning tree problem over a large extended formulation. *Mathematical Programming series A*, 112(2):443–472, 2008.