O problema de Steiner com grupos

Fernando Mário de Oliveira Filho

DISSERTAÇÃO APRESENTADA
AO
INSTITUTO DE MATEMÁTICA E ESTATÍSTICA
DA
UNIVERSIDADE DE SÃO PAULO
PARA
A OBTENÇÃO DO GRAU DE MESTRE
EM
CIÊNCIA DA COMPUTAÇÃO

Orientador: Prof. Dr. Carlos Eduardo Ferreira

Durante a elaboração deste trabalho o autor recebeu apoio financeiro da FAPESP através do processo 03/10045-0.

O problema de Steiner com grupos

Este exemplar corresponde à redação final da dissertação devidamente corrigida e defendida por Fernando Mário de Oliveira Filho e aprovada pela comissão julgadora.

São Paulo, 27 de outubro de 2005.

Banca examinadora:

Prof. Dr. Carlos Eduardo Ferreira (orientador)
Prof. Dr. Cid Carvalho de Souza
Profa. Dra. Yoshiko Wakabayashi

IME – USP
IC – UNICAMP
IME – USP

Agradecimentos

Agradeço a minha família pelo apoio, em especial aos meus pais Fernando e Irene pelo carinho durante todos esses anos.

Agradeço ao Carlinhos, meu orientador, pela paciência, apoio, encorajamento e pelo excelente aconselhamento. Agradeço também ao professor Yoshiharu, o Yoshi, pelo sempre presente apoio e encorajamento.

Agradeço aos membros da banca e ao Marcel K. de Carli Silva por terem lido atentamente o texto, apontado inúmeros erros e proposto inúmeras modificações que certamente contribuíram muito para a qualidade deste trabalho.

Agradeço ao Daniel Martin, ao Jozef e à Cris por interessantes discussões, pelo entusiasmo e pelo apoio.

Finalmente, agradeço aos meus amigos, que tornaram esses últimos anos muito divertidos. Sou feliz por conhecê-los.

Resumo

Neste trabalho estudamos o problema de Steiner com grupos. Descrevemos o problema, investigamos suas conexões com outros problemas de otimização combinatória e fazemos uma rápida discussão sobre sua complexidade computacional. Apresentamos também alguns algoritmos de aproximação que foram propostos, introduzimos novas técnicas de redução de instâncias, fazemos um extensivo estudo poliédrico e terminamos por apresentar um algoritmo exato para o problema baseado na estratégia *branch-and-cut*.

Abstract

In this work we study the group Steiner tree problem. We describe the problem, investigate its connections to other combinatorial optimization problems, and survey results about its computational complexity. We also present some approximation algorithms that have been proposed for the problem, introduce new reduction techniques, and present an extensive polyhedral study. Finally, we propose a new branch-and-cut algorithm for the problem.

Sumário

Introdução	1
1 Notação e preliminares	3
1.1 Preliminares gerais	3
1.1.1 Conjuntos de números	3
1.1.2 Vetores e matrizes	3
1.1.3 Álgebra linear	4
1.2 Grafos	5
1.2.1 Grafos não-dirigidos	6
1.2.2 Grafos dirigidos	8
1.2.3 Fluxos máximos em grafos dirigidos	9
1.3 Poliedros	10
1.3.1 Cones, politopos e poliedros	10
1.3.2 Faces, facetas e faces minimais	11
1.3.3 Poliedros inteiros	12
1.4 Um pouco sobre probabilidade	13
2 O problema, aplicações e complexidade	15
2.1 O problema	15
2.2 Aplicação: roteamento em circuitos VLSI	16
2.3 Complexidade computacional	17
2.3.1 Resultados de inaproximabilidade	19
2.3.2 Do problema de Steiner com grupos para o problema de Steiner	20
2.4 Mais sobre o problema de Steiner	20
3 Algoritmos de aproximação	21
3.1 Heurística dos caminhos mínimos	21
3.2 Um algoritmo simples para árvores	23
3.3 O algoritmo de Garg, Konjevod e Ravi	24
3.3.1 Um experimento aleatório	25
3.3.2 O algoritmo	25
3.3.3 Análise do experimento	27
3.4 Observações finais	31

4	Testes de redução	32
4.1	Distâncias de gargalo	32
4.2	Remoção de arestas e não-terminais	33
4.3	Remoção de vértices de grupo	36
4.3.1	Um teste simples	36
4.3.2	Testes de expansão	37
4.3.3	Um teste adicional	40
4.4	Implementações dos testes e resultados	41
5	Uma investigação poliédrica	44
5.1	Os poliedros	44
5.1.1	A dimensão de P_{STG}	44
5.1.2	Adjacência em P_{STG}	46
5.2	Uma primeira formulação para P_{STG}^I	47
5.2.1	O problema da separação	48
5.2.2	A separação em árvores	50
5.3	As facetas de P_{STG}^I	52
5.3.1	Contrações de arestas e facetas de P_{STG}^I	54
5.4	Algumas formulações estendidas para P_{STG}^I	55
5.4.1	Variáveis nos vértices	56
5.4.2	Arestas artificiais	57
5.4.3	Uma formulação dirigida	58
5.4.4	Eliminação de subcircuitos	60
5.5	Descrições completas	64
5.5.1	Dois grupos	64
5.5.2	Árvores 2-grupo restritas	65
5.6	Observações finais	66
6	Um algoritmo <i>branch-and-cut</i>	67
6.1	Estrutura básica de um algoritmo <i>branch-and-cut</i>	67
6.2	O algoritmo <i>branch-and-cut</i> desenvolvido	69
6.2.1	Pré-processamento	69
6.2.2	Formulação	70
6.2.3	Heurística primal	70
6.2.4	Estratégias de ramificação e processamento	71
6.3	Resultados computacionais	71
6.3.1	Os algoritmos de redução	71
6.3.2	O algoritmo <i>branch-and-cut</i>	72
	Conclusão	74
	Referências bibliográficas	76

Introdução

Otimização combinatória é uma das áreas mais importantes da Matemática e Ciência da Computação. De forma bem geral, podemos dizer que um problema de otimização combinatória é um problema do tipo: dada uma função $f: S \rightarrow \mathbb{R}$, sendo S um conjunto finito, encontre um $s \in S$ tal que $f(s)$ seja o menor (ou o maior) possível.

O objetivo é então encontrar um algoritmo eficiente para resolver um tal problema. Do ponto de vista computacional, isso só é interessante quando o conjunto S é dado implicitamente em função, por exemplo, de alguma propriedade que seus elementos devem satisfazer. Nesse caso, o tamanho de S pode ser muito maior do que o tamanho dos objetos usados para defini-lo, de modo que uma simples estratégia de busca exaustiva seria por demais ineficiente.

Para tornar as idéias mais claras, vejamos um exemplo concreto de problema de otimização combinatória, o *problema da árvore geradora mínima*:

- (1) dados: um grafo conexo $G = (V, E)$ e um comprimento $l(e)$ para cada aresta e de G ;
encontre: uma árvore T em G que contenha todos os vértices de G e que tenha comprimento total mínimo.

Acima, o *comprimento* de uma árvore T é $l(ET) := \sum_{e \in ET} l(e)$, sendo ET o conjunto das arestas de T .

No caso do problema da árvore geradora mínima, o conjunto S é o conjunto de todas as árvores geradoras de G (i.e., árvores em G que contêm todos os vértices de G). Não é difícil ver que, em muitos casos, o tamanho de S é exponencial no tamanho de G . Suponha, por exemplo, que G seja completo, i.e., que todo par de vértices de G seja adjacente. Vamos tentar estimar o número de árvores geradoras de G .

Observe que, nesse caso, qualquer caminho em G que contém todos os vértices de G é uma árvore geradora de G . Seja n o número de vértices de G . O número de caminhos em G que contém todos os vértices de G pode ser facilmente determinado. Qualquer permutação dos vértices de G nos fornece um tal caminho. Entretanto, quando contamos todas essas permutações, acabamos contando duas vezes cada caminho. Assim, o número de caminhos geradores é $n!/2$. Para se ter uma idéia, isso significa que, se $n = 50$, o tamanho de S é pelo menos

15 207 046 600 856 689 021 806 304 083 032 384 422 188 820 784 480 256 000 000 000 000.

Mesmo se pudéssemos verificar um trilhão de árvores geradoras por segundo, um algoritmo simples de busca exaustiva levaria pelo menos $4,8 \cdot 10^{43}$ anos para obter a resposta. Claramente, um algoritmo simples de busca exaustiva não é a resposta.

Sabemos, entretanto, que existem algoritmos eficientes para o problema da árvore geradora mínima. Na verdade, esse é um dos problemas “bem resolvidos” da otimização combinatória e surge freqüentemente em aplicações práticas e também na solução de outros problemas.

Para muitos outros problemas de otimização combinatória, entretanto, não são conhecidos algoritmos eficientes. Acredita-se que tais problemas sejam inerentemente difíceis e que tais algoritmos não existam. Esse é o caso do problema de Steiner com grupos, que estudamos neste trabalho.

O problema de Steiner com grupos, como muitos outros problemas de otimização combinatória, surge naturalmente em aplicações práticas. Resolvê-lo é importante, mesmo havendo fortes resultados negativos a seu respeito. Neste trabalho, apresentamos um estudo detalhado do problema. Discutimos sua complexidade computacional, descrevemos algoritmos de aproximação (algoritmos que não necessariamente encontram soluções ótimas, mas que encontram boas soluções para o problema) que para ele foram propostos e terminamos por propor um algoritmo para o problema segundo a estratégia *branch-and-cut*.

Fomos motivados a propor um tal algoritmo devido ao grande sucesso que essa estratégia tem obtido desde que foi proposta. O estudo de poliedros associados ao problema também nos motivou. Além disso, os estudos necessários ao desenvolvimento de um tal algoritmo levaram-nos a propor um novo algoritmo de aproximação para uma restrição do problema (cf. seção 3.2) e novas técnicas de redução de instâncias (cf. capítulo 4) além, é claro, de permitirem um estudo mais detalhado e certamente frutífero (cf. capítulo 5) de poliedros associados ao problema.

O restante do texto está organizado da seguinte forma: no capítulo 1, descrevemos alguns conceitos importantes e a notação utilizada. No capítulo 2, descrevemos o problema e estudamos sua complexidade computacional. No capítulo 3, descrevemos alguns dos algoritmos de aproximação propostos para o problema. No capítulo 4, expomos alguns algoritmos de redução para o problema. No capítulo 5, fazemos um estudo poliédrico e, finalmente, no capítulo 6, descrevemos um algoritmo para o problema segundo a estratégia *branch-and-cut* e apresentamos resultados computacionais.

CAPÍTULO 1

Notação e preliminares

Neste capítulo apresentamos a notação que será utilizada no restante do texto e introduzimos alguns conceitos importantes. O leitor é encorajado a ler este capítulo superficialmente. Durante a leitura do restante do texto, qualquer dúvida sobre a notação pode ser rapidamente resolvida com uma consulta a este capítulo.

1.1. Preliminares gerais

Começamos por apresentar algumas notações básicas para conjuntos de números, vetores, etc. Seguimos basicamente a notação do livro de Schrijver [43].

1.1.1. Conjuntos de números

Denotamos por $\mathbb{Z}$, $\mathbb{Q}$ e $\mathbb{R}$, respectivamente, o conjunto dos números inteiros, racionais e reais. Denotamos por $\mathbb{Z}_+$, $\mathbb{Q}_+$ e $\mathbb{R}_+$, respectivamente, o conjunto dos números inteiros não-negativos, racionais não-negativos e reais não-negativos.

Se x é um número, denotamos por $\lfloor x \rfloor$ o maior inteiro menor ou igual a x . Denotamos por $\lceil x \rceil$ o menor inteiro maior ou igual a x .

1.1.2. Vetores e matrizes

Descrevemos nossa notação para vetores e matrizes em função do conjunto dos números reais, $\mathbb{R}$. Entretanto, tudo aqui pode ser lido trocando-se $\mathbb{R}$ por $\mathbb{Q}$.

Todos os vetores que consideramos são vetores coluna. Se x é um vetor cujos componentes são indexados por elementos de um conjunto finito V , referimo-nos a seus componentes pela notação $x(v)$ ou x_v , para $v \in V$.

Se V é um conjunto finito (ou também uma família finita, caso em que levamos multiplicidades em conta), denotamos por $\mathbb{R}^V$ o conjunto dos vetores reais indexados pelos elementos de V . No texto, confundimos o conceito de um vetor $x \in \mathbb{R}^V$ com o conceito de uma função $x: V \rightarrow \mathbb{R}$.

Se $x, y \in \mathbb{R}^V$ são vetores reais, escrevemos $x \leq y$ quando $x_v \leq y_v$ para todo $v \in V$. Denotamos por 0_V (ou simplesmente por 0 , quando não há perigo de ambigüidade) o vetor em $\mathbb{R}^V$ cujos componentes são todos nulos. Denotamos por 1_V (ou simplesmente por 1) o vetor em $\mathbb{R}^V$ cujos componentes são todos iguais a 1.

Se x é um vetor então o *suporte* de x é o conjunto dos índices que correspondem a componentes não-nulos de x . Denotamos o suporte do vetor x por $\text{supp } x$.

Para qualquer conjunto $U \subseteq V$, o *vetor de incidência* ou *vetor característico* de U (em $\mathbb{R}^V$) é o vetor χ^U definido da seguinte forma:

$$(1.1) \quad \chi^U(s) := \begin{cases} 1, & \text{se } s \in U, \\ 0, & \text{se } s \notin U. \end{cases}$$

Para qualquer $v \in V$ escrevemos

$$(1.2) \quad \chi^v := \chi^{\{v\}}.$$

Esse é o v -ésimo *vetor base unitário*.

Se A é uma matriz, denotamos por A^T a matriz transposta de A . Se x é um vetor então x^T é uma matriz com apenas uma linha. Denotamos por *posto* A o *posto* da matriz A , i.e., a dimensão do espaço vetorial gerado pelas colunas de A (que é igual à dimensão do espaço vetorial gerado pelas colunas de A^T).

Para qualquer função $w: V \rightarrow \mathbb{R}$ e $U \subseteq V$ escrevemos

$$(1.3) \quad w(U) := \sum_{v \in U} w(v).$$

Se U é uma família, levamos as multiplicidades em consideração durante a soma.

Se w é introduzida como sendo uma função “peso”, então $w(v)$ é chamado de *peso* de v . Da mesma forma, para $U \subseteq V$, chamamos $w(U)$ de *peso* de U . O adjetivo “peso” para a palavra “função” não tem significado matemático, ele só serve para que nos possamos referir a $w(U)$ como sendo o peso do conjunto U . O mesmo ocorre quando falamos de uma “função capacidade” ou de uma “função comprimento”.

1.1.3. Álgebra linear

Apresentamos agora alguns conceitos de álgebra linear. Tais conceitos podem ser encontrados no livro de Ferreira e Wakabayashi [16] ou no livro de Schrijver [42].

Sejam $x_1, \dots, x_k$ vetores em $\mathbb{R}^n$ e $\lambda_1, \dots, \lambda_k$ números reais. Dizemos que o vetor

$$(1.4) \quad x := \lambda_1 x_1 + \dots + \lambda_k x_k$$

é uma *combinação linear* dos vetores $x_1, \dots, x_k$. Se $\lambda_1 + \dots + \lambda_k = 1$, x é uma *combinação afim* de $x_1, \dots, x_k$. Se, em vez disso, temos $\lambda_i \geq 0$ para $i = 1, \dots, k$, x é uma *combinação cônica* de $x_1, \dots, x_k$. Dizemos que x é uma *combinação convexa* de $x_1, \dots, x_k$ se $\lambda_i \geq 0$ para $i = 1, \dots, k$ e $\lambda_1 + \dots + \lambda_k = 1$.

Considere um conjunto $X \subseteq \mathbb{R}^n$. Definimos:

$$(1.5) \quad \begin{aligned} \text{fecho.lin } X &:= \{x \in \mathbb{R}^n \mid x \text{ é combinação linear de } x_1, \dots, x_k \in X\}; \\ \text{fecho.afim } X &:= \{x \in \mathbb{R}^n \mid x \text{ é combinação afim de } x_1, \dots, x_k \in X\}; \\ \text{cone } X &:= \{x \in \mathbb{R}^n \mid x \text{ é combinação cônica de } x_1, \dots, x_k \in X\}; \\ \text{fecho.conv } X &:= \{x \in \mathbb{R}^n \mid x \text{ é combinação convexa de } x_1, \dots, x_k \in X\}. \end{aligned}$$

Um conceito análogo ao conceito de independência linear é o conceito de independência afim. Dizemos que os vetores $x_1, \dots, x_k$ são *afim-independentes* se nenhum deles é combinação afim dos demais. Equivalentemente, $x_1, \dots, x_k$ são afim-independentes se e só se

(1.6) não existem números reais $\lambda_1, \dots, \lambda_k$, não todos nulos, tais que $\lambda_1 + \dots + \lambda_k = 0$ e $\lambda_1 x_1 + \dots + \lambda_k x_k = \mathbf{0}$.

Geralmente, é útil utilizar a seguinte relação entre independência afim e independência linear:

(1.7) os vetores $x_0, x_1, \dots, x_k$ são afim-independentes se e somente se os vetores $x_1 - x_0, \dots, x_k - x_0$ são linearmente independentes.

Um conjunto $H \subseteq \mathbb{R}^n$ é um *subespaço afim* se existe uma matriz A e um vetor b tais que

$$(1.8) \quad H = \{x \in \mathbb{R}^n \mid Ax = b\}.$$

Se $b = \mathbf{0}$, H é um *subespaço linear*. Se A é não-nula e tem uma só linha, ou seja, se existe um vetor $c \in \mathbb{R}^n$, $c \neq \mathbf{0}$, e um real δ tais que

$$(1.9) \quad H = \{x \in \mathbb{R}^n \mid c^\top x = \delta\},$$

então H é um *hiperplano afim* (ou simplesmente um *hiperplano*). É fácil provar que:

(1.10) um conjunto $H \subseteq \mathbb{R}^n$ é um subespaço afim se e somente se existe um conjunto finito $X \subseteq \mathbb{R}^n$ tal que $H = \text{fecho.afim } X$.

A *dimensão* do subespaço afim $H = \{x \in \mathbb{R}^n \mid Ax = b\}$ é

$$(1.11) \quad \dim H := n - \text{posto } A,$$

se H é não-vazio. Se H é vazio, sua dimensão é, por definição, igual a -1 . Com isso, temos que:

$$(1.12) \quad \dim H = \max\{|U| \mid U \subseteq H \text{ é afim-independente}\} - 1.$$

Podemos estender o conceito de dimensão a qualquer conjunto $X \subseteq \mathbb{R}^n$. Dado um tal conjunto X , sua *dimensão* é

$$(1.13) \quad \dim X := \dim(\text{fecho.afim } X).$$

Novamente, temos que:

$$(1.14) \quad \dim X = \max\{|U| \mid U \subseteq X \text{ é afim-independente}\} - 1.$$

Um subconjunto X do $\mathbb{R}^n$ cuja dimensão é n tem *dimensão plena*.

1.2. Grafos

Nossa notação para a teoria dos grafos segue, basicamente, a de Schrijver [43]. Para compreender o texto não são necessários grandes conhecimentos de resultados da teoria dos grafos. Um livro sobre grafos que cobre parte desses resultados, principalmente os mais clássicos, é o livro de Diestel [6].

1.2.1. Grafos não-dirigidos

Um *grafo* ou *grafo não-dirigido* é um par $G = (V, E)$, sendo V um conjunto finito e E uma família de pares não-ordenados de elementos de V . Aqui, um par *não-ordenado* de elementos de V é um conjunto $\{u, v\}$, com $u, v \in V$. Chamamos os elementos de V de *vértices* e os elementos de E de *arestas*. Geralmente, denotamos a aresta $\{u, v\}$ simplesmente por uv e dizemos que u e v são suas *pontas*. Também escrevemos

$$(1.15) \quad \begin{aligned} VG &:= \text{conjunto de vértices do grafo } G; \\ EG &:= \text{família de arestas do grafo } G. \end{aligned}$$

É importante observar que E é uma família, logo pode conter elementos repetidos. Uma aresta que ocorre mais de uma vez em E é chamada de aresta *múltipla*. Duas arestas com as mesmas pontas são *paralelas*. Uma aresta da forma $\{v, v\}$ para algum vértice $v \in V$ é chamada de *laço*. Dizemos que um grafo é *simplex* se não possui nem arestas múltiplas nem laços.

Se $e = uv$ é uma aresta do grafo G , dizemos que e *incide* em u e v ou que *liga* u a v . Dizemos que os vértices u e v são *adjacentes* se existe uma aresta que liga u a v . Se v é um vértice de G , o *grau* de v é o número de arestas incidentes em v , sendo que laços contam duas vezes. Um vértice de grau zero é chamado de vértice *isolado*.

Chamamos o grafo G de *completo* se ele for simplex e todo par de vértices de G for adjacente. Em estimativas de complexidade de algoritmos, escrevemos $n := |V|$ e $m := |E|$.

Subgrafos e operações. Seja $G = (V, E)$ um grafo. Dizemos que $H = (V', E')$ é um *subgrafo de G induzido por V'* se $V' \subseteq V$ e E' consiste de todas as arestas de G com ambas as pontas em V' . Para $U \subseteq V$, escrevemos

$$(1.16) \quad \begin{aligned} G[U] &:= \text{subgrafo de } G \text{ induzido por } U; \\ E[U] &:= \{e \in E \mid \exists u, v \in U: e = uv\}. \end{aligned}$$

Dizemos que um grafo $H = (V', E')$ é um *subgrafo* de G se $V' \subseteq V$ e $E' \subseteq E[V']$.

Seja $U \subseteq V$ e $F \subseteq E$. Escrevemos

$$(1.17) \quad \begin{aligned} G - U &:= G[V \setminus U]; \\ G - F &:= (V, E \setminus F). \end{aligned}$$

Se F é uma família de pares não-ordenados, escrevemos

$$(1.18) \quad G + F := (V \cup \bigcup_{e \in F} e, E \cup F).$$

Para $v \in V$, por exemplo, sempre escrevemos $G - v$ em vez de $G - \{v\}$. No caso de arestas, fazemos algo similar.

Seja $e = uv$ uma aresta do grafo G . *Contrair* a aresta e significa remover e de G e identificar suas pontas, como mostra a Figura 1.1. Note que podem surgir laços ou arestas múltiplas no grafo G após a contração de uma aresta. Para $F \subseteq E$, escrevemos

$$(1.19) \quad \begin{aligned} G/e &:= \text{grafo resultante da contração da aresta } e; \\ G/F &:= \text{grafo resultante da contração de todas as arestas em } F. \end{aligned}$$

Figura 1.1: A contração da aresta destacada em cinza no grafo em (a) leva ao grafo em (b). Note que surgiram arestas múltiplas.

A *imagem* de um vértice v de G em G/F é o vértice de G/F para o qual ele é contraído.

Dizemos que um grafo H é um *menor por contração* de G se H pode ser obtido de G a partir de uma série de contrações de arestas. Há uma óbvia correspondência entre as arestas de G/e e as arestas de G . Normalmente, se f é uma aresta de G/e , vamos abusar da notação e considerar f uma aresta de G , mesmo que suas pontas possam ser diferentes. Também podemos fazer o contrário. Nesses casos, é óbvio que nos referimos às arestas correspondentes.

Cortes. Dado um grafo $G = (V, E)$ e um conjunto $U \subseteq V$, escrevemos

$$(1.20) \quad \delta_G(U) := \delta(U) := \{e \in E \mid \exists u \in U, v \in V \setminus U: e = uv\}.$$

Em outras palavras, $\delta(U)$ é a família das arestas que têm uma ponta em U e a outra em $V \setminus U$. Dizemos que $F \subseteq E$ é um *corte* se existe um conjunto $U \subseteq V$ tal que $\delta(U) = F$.

Dado $v \in V$, escrevemos $\delta(v)$ em vez de $\delta(\{v\})$. Dados $s, t \in V$, dizemos que $\delta(U)$ é um s - t *corte* se $s \in U$ e $t \in V \setminus U$. Analogamente, se $S, T \subseteq V$, dizemos que $\delta(U)$ é um S - T *corte* se $S \subseteq U$ e $T \subseteq V \setminus U$.

Caminhos e circuitos. Um *passeio* é uma seqüência

$$(1.21) \quad P = (v_0, e_1, v_1, \dots, e_k, v_k),$$

sendo que $k \geq 0$, $v_0, \dots, v_k$ são vértices e e_i é uma aresta que liga v_{i-1} a v_i , para $i = 1, \dots, k$. Se os vértices v_i são todos distintos, P é um *caminho*.

Dizemos que v_0 e v_k são as *pontas*, *términos* ou *extremos* de P . Dizemos que P *conecta* v_0 a v_k . Se s e t são vértices, chamamos um passeio (caminho) P que conecta s a t de s - t *passeio* (caminho). Se $S, T \subseteq V$ são dois conjuntos disjuntos de vértices, chamamos um passeio (caminho) que conecta um vértice de S a um vértice de T de S - T *passeio* (caminho).

O *comprimento* do passeio P é definido como sendo k . Se introduzimos uma função $l: E \rightarrow \mathbb{R}$ como sendo uma função comprimento, o comprimento do passeio P é $l(e_1) + \dots + l(e_k)$.

Se P é um passeio, escrevemos

$$(1.22) \quad \begin{aligned} VP &:= \text{conjunto de vértices de } P; \\ EP &:= \text{família de arestas de } P. \end{aligned}$$

O passeio em (1.21) é *fechado* se $v_0 = v_k$. Ele é chamado de *circuito* se é fechado, $k \geq 1$, $v_1, \dots, v_k$ são todos distintos e $e_1, \dots, e_k$ são todas distintas.

Conexidade. Seja $G = (V, E)$ um grafo. Dizemos que G é *conexo* se todo par de vértices de G é ligado por um caminho. Um *componente conexo* de G é um subgrafo conexo maximal de G . Aqui, a noção de maximal é a de tomar subgrafos.

Suponha que G seja conexo. Um vértice $v \in V$ é um *vértice de corte* se $G - v$ é desconexo. Dizemos que G é *2-conexo* se G é conexo e não possui vértices de corte.

Um *componente 2-conexo* de um grafo conexo G é um subgrafo 2-conexo maximal de G . Podemos imaginar a seguinte relação sobre as arestas de um grafo conexo G :

$$(1.23) \quad e \sim f \iff e = f \text{ ou } e \text{ e } f \text{ pertencem a um mesmo circuito em } G.$$

Podemos provar que a relação $\sim$ é uma relação de equivalência e que suas classes de equivalência são os conjuntos de arestas dos componentes 2-conexos de G . Com isso, temos que as arestas e e f do grafo G , $e \neq f$, pertencem a um mesmo componente 2-conexo de G se e somente se ocorrem juntas num mesmo circuito de G . Também segue que um grafo é 2-conexo se e somente se quaisquer duas de suas arestas ocorrem juntas em algum circuito.

Árvores, florestas e árvores enraizadas. Um grafo $G = (V, E)$ que não possui circuitos é uma *floresta*. Uma *árvore* é uma floresta conexa. Um vértice v de uma árvore que possui grau menor ou igual a 1 é chamado de *folha*.

Uma *árvore enraizada* é uma árvore $T = (V, E)$ que possui um vértice especial s , chamado de *raiz*. Também chamamos tal árvore de *árvore enraizada em s* ou *árvore de raiz s* .

Seja $T = (V, E)$ uma árvore enraizada em s . A *profundidade* de um vértice v de T é o comprimento do s - v caminho em T . A *altura* da árvore é a profundidade do vértice mais profundo. Se u, v são vértices de T , dizemos que u é *ancestral* de v se $u \neq v$ e u ocorre no s - v caminho em T . O *ancestral comum mais próximo* de dois vértices u, v , $u \neq v$, é o vértice mais profundo que é ancestral de ambos u e v .

Seja u um vértice da árvore enraizada T . Se v é adjacente a u e sua profundidade é maior que a profundidade de u então v é *filho* de u . Dizemos também que u é *pai* de v . A raiz s não possui pai. Para um vértice v da árvore T , escrevemos

$$(1.24) \quad \begin{aligned} \delta_T^{\text{in}}(v) &:= \delta^{\text{in}}(v) := \text{conjunto composto pela aresta que liga } v \text{ ao seu pai;} \\ \delta_T^{\text{out}}(v) &:= \delta^{\text{out}}(v) := \text{conjunto das arestas que ligam } v \text{ aos filhos de } v. \end{aligned}$$

Observe que $\delta^{\text{in}}(s)$ é apenas o conjunto vazio.

1.2.2. Grafos dirigidos

Um *grafo dirigido* é um par $D = (V, A)$, sendo V um conjunto finito e A uma família de pares ordenados de elementos de V . Chamamos os elementos de V de *vértices* e os elementos de A de *arcos*. Escrevemos

$$(1.25) \quad \begin{aligned} VD &:= \text{conjunto de vértices do grafo dirigido } D; \\ AD &:= \text{família de arcos do grafo dirigido } D. \end{aligned}$$

Seja (u, v) um arco de D . Dizemos que (u, v) *sai* de u e *entra* em v . Também dizemos que os vértices u e v são as *pontas* de (u, v) . Chamamos o vértice u de *início* do arco (u, v) e o vértice v de *término* desse arco.

Todo grafo dirigido $D = (V, A)$ possui um *grafo não-dirigido subjacente* $G = (V, E)$, sendo

$$(1.26) \quad E := \{uv \mid (u, v) \in A\}.$$

O grafo não-dirigido subjacente de um grafo dirigido permite-nos traduzir muitos dos conceitos que definimos para grafos não-dirigidos para o caso de grafos dirigidos, como o conceito de passeios e caminhos. O conceito de subgrafos e as operações descritas na seção anterior podem ser estendidos aos grafos dirigidos da forma óbvia. Em particular, se $D = (V, A)$ é um grafo dirigido e $U \subseteq V$, escrevemos

$$(1.27) \quad A[U] := \{a \in A \mid \exists u, v \in U: a = (u, v)\}.$$

Cortes dirigidos. Dado um grafo dirigido $D = (V, A)$ e um conjunto $U \subseteq V$, escrevemos

$$(1.28) \quad \begin{aligned} \delta_D^{\text{in}}(U) &:= \delta^{\text{in}}(U) := \{a \in A \mid \exists u \in V \setminus U, v \in U: a = (u, v)\}; \\ \delta_D^{\text{out}}(U) &:= \delta^{\text{out}}(U) := \{a \in A \mid \exists u \in U, v \in V \setminus U: a = (u, v)\}. \end{aligned}$$

Se v é um vértice de D , escrevemos, por exemplo, $\delta^{\text{out}}(v)$ em vez de $\delta^{\text{out}}(\{v\})$.

Dizemos que $B \subseteq A$ é um *corte dirigido* se existe um conjunto $U \subseteq V$ tal que $B = \delta^{\text{out}}(U)$. Os conceitos de s - t corte dirigido e de S - T corte dirigido são definidos de forma análoga aos mesmos conceitos para grafos não-dirigidos.

Caminhos e circuitos dirigidos. Um passeio, caminho ou circuito num grafo dirigido é simplesmente um passeio, caminho ou circuito no grafo não-dirigido subjacente. Em outras palavras, simplesmente ignoramos a direção dos arcos. Mais precisamente, um *passeio* (*não-dirigido*) num grafo dirigido $D = (V, A)$ é uma seqüência

$$(1.29) \quad P = (v_0, a_1, v_1, \dots, a_k, v_k),$$

sendo que $k \geq 0$, $v_0, \dots, v_k$ são vértices e $a_i = (v_{i-1}, v_i)$ ou $a_i = (v_i, v_{i-1})$ para $i = 1, \dots, k$. Analogamente, podemos definir os conceitos de caminho e circuito.

Já um *passeio dirigido* é uma seqüência

$$(1.30) \quad P = (v_0, a_1, v_1, \dots, a_k, v_k),$$

sendo que $k \geq 0$, $v_0, \dots, v_k$ são vértices e $a_i = (v_{i-1}, v_i)$ para cada $i = 1, \dots, k$. De forma similar podemos definir os conceitos de caminho dirigido e circuito dirigido.

1.2.3. Fluxos máximos em grafos dirigidos

Seja $D = (V, A)$ um grafo dirigido, $c: A \rightarrow \mathbb{R}_+$ uma função capacidade e s, t dois vértices de D . Dizemos que uma função $f: A \rightarrow \mathbb{R}_+$ é um s - t *fluxo sujeito a c* se $f \leq c$ e, para todo vértice $v \in V \setminus \{s, t\}$, temos que

$$(1.31) \quad f(\delta^{\text{in}}(v)) = f(\delta^{\text{out}}(v)).$$

O *valor* do fluxo f é

$$(1.32) \quad f(\delta^{\text{out}}(s)) - f(\delta^{\text{in}}(s)).$$

O famoso teorema do fluxo-máximo corte-mínimo, de Ford e Fulkerson [17] (veja também Schrijver [43], capítulo 10), é o seguinte:

Teorema 1.1 (teorema do fluxo-máximo corte-mínimo). *Se $f: A \rightarrow \mathbb{R}_+$ é um s - t fluxo sujeito a c de valor máximo e $\delta^{\text{out}}(U)$ é um s - t corte de capacidade mínima, então o valor de f é igual à capacidade de $\delta^{\text{out}}(U)$.*

Outro resultado útil é o seguinte “teorema de decomposição” de fluxos:

Teorema 1.2. *Seja $f: A \rightarrow \mathbb{R}_+$ um s - t fluxo sujeito a c de valor ν . Existem s - t caminhos dirigidos $P_1, \dots, P_k$, circuitos dirigidos $C_1, \dots, C_l$ e reais não-negativos $\lambda_1, \dots, \lambda_k$ e $\mu_1, \dots, \mu_l$ tais que $f = \lambda_1 \chi^{AP_1} + \dots + \lambda_k \chi^{AP_k} + \mu_1 \chi^{AC_1} + \dots + \mu_l \chi^{AC_l}$ e $\nu = \lambda_1 + \dots + \lambda_k$.*

Para uma demonstração do teorema acima, veja o livro de Schrijver [43]. Em particular, é fácil ver que se o suporte de f é minimal então $(V, \text{supp } f)$ não contém circuitos dirigidos. Nesse caso podemos desprezar os circuitos na decomposição dada pelo teorema.

1.3. Poliedros

Nesta seção apresentamos algumas definições básicas sobre cones, politopos e poliedros. Tais definições podem ser encontradas em [42].

1.3.1. Cones, politopos e poliedros

Um conjunto $C \subseteq \mathbb{R}^n$ é um *cone poliédrico* (ou simplesmente um *cone*) se existe uma matriz A tal que

$$(1.33) \quad C = \{x \in \mathbb{R}^n \mid Ax \leq 0\}.$$

Podemos provar que:

Teorema 1.3. *Um conjunto $C \subseteq \mathbb{R}^n$ é um cone poliédrico se e somente se existe um conjunto finito $X \subseteq \mathbb{R}^n$ tal que $C = \text{cone } X$.*

Um conjunto $P \subseteq \mathbb{R}^n$ é um *poliedro* se existe uma matriz A e um vetor b tais que

$$(1.34) \quad P = \{x \in \mathbb{R}^n \mid Ax \leq b\}.$$

Há várias formas equivalentes de fornecer um poliedro. Por exemplo, se A é uma matriz e b um vetor, o conjunto $P = \{x \in \mathbb{R}^n \mid x \geq 0, Ax = b\}$ é um poliedro: basta notar que P é o conjunto dos vetores $x \in \mathbb{R}^n$ que satisfazem

$$(1.35) \quad \begin{pmatrix} A \\ -A \\ -I \end{pmatrix} x \leq \begin{pmatrix} b \\ -b \\ 0 \end{pmatrix}.$$

Se $P \subseteq \mathbb{R}^E$ é um poliedro então, para todo $F \subseteq E$, a *projeção de P em $\mathbb{R}^F$* é

$$(1.36) \quad P^F := \{x \in \mathbb{R}^F \mid \exists y \in \mathbb{R}^{E \setminus F}: (x; y) \in P\},$$

sendo que $(x; y)$ denota o vetor $\begin{pmatrix} x \\ y \end{pmatrix}$. Podemos provar que P^F também é um poliedro.

Todo cone é um poliedro. O *cone característico* do poliedro $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ é o conjunto

$$(1.37) \quad \text{cone.car } P := \{x \in \mathbb{R}^n \mid Ax \leq 0\}.$$

Se $P \neq \emptyset$, o cone característico de P é o conjunto dos vetores $y \in \mathbb{R}^n$ tais que $x + y \in P$ para todo vetor $x \in P$.

Um conjunto $P \subseteq \mathbb{R}^n$ é um *politopo* se existe um conjunto finito $X \subseteq \mathbb{R}^n$ tal que

$$(1.38) \quad P = \text{fecho.conv } X.$$

Observe que todo politopo é limitado. Podemos provar o seguinte “teorema de decomposição”:

Teorema 1.4. *Um conjunto $P \subseteq \mathbb{R}^n$ é um poliedro se e somente se existe um politopo Q e um cone C tais que $P = Q + C$.*

Segue trivialmente que todo politopo é um poliedro e que todo poliedro limitado é um politopo. Além disso, segue que P é um poliedro se e somente se existem conjuntos $X, Y \subseteq \mathbb{R}^n$, ambos finitos, tais que

$$(1.39) \quad P = \text{fecho.conv } X + \text{cone } Y.$$

Considere o poliedro $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$. Uma inequação $c^T x \leq \delta$ é *válida* para P se $c^T x \leq \delta$ para todo $x \in P$. Uma inequação $c^T x \leq \delta$ de $Ax \leq b$ é uma *igualdade implícita* se $c^T x = \delta$ para todo x que satisfaz $Ax \leq b$. Podemos então escrever $P = \{x \in \mathbb{R}^n \mid A^=x = b^=, A^+x \leq b^+\}$, sendo $A^=x = b^=$ o subsistema das igualdades implícitas de $Ax \leq b$ e $A^+x \leq b^+$ o sistema composto pelas demais inequações em $Ax \leq b$. Temos que:

$$(1.40) \quad \text{fecho.afim } P = \{x \in \mathbb{R}^n \mid A^=x = b^=\}.$$

Em particular, segue de (1.40) que $\dim P = n - \text{posto } A^=$. Além disso, P tem dimensão plena se e somente se $Ax \leq b$ não possui igualdades implícitas.

1.3.2. Faces, facetas e faces minimais

Seja $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ um poliedro. Se c é um vetor não-nulo e $\delta = \max\{c^T x \mid Ax \leq b\}$, o hiperplano afim $\{x \mid c^T x = \delta\}$ é chamado de *hiperplano suporte* de P . Um subconjunto F de P é uma *face* de P se $F = P$ ou se $F = P \cap H$ para algum hiperplano suporte H de P . Segue que:

$$(1.41) \quad F \text{ é uma face de } P \text{ se e somente se } F \text{ é o conjunto das soluções ótimas do problema } \max\{c^T x \mid Ax \leq b\} \text{ para algum } c \in \mathbb{R}^n.$$

Uma inequação válida $c^T x \leq \delta$ induz uma face F se

$$(1.42) \quad F = \{x \in P \mid c^T x = \delta\}.$$

Temos o seguinte resultado:

Teorema 1.5. *Seja $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ um poliedro. O conjunto F é uma face de P se e somente se*

$$(1.43) \quad F = \{x \in P \mid A'x = b'\}$$

para algum subsistema $A'x \leq b'$ de $Ax \leq b$.

Segue que o número de faces de um poliedro é finito e que toda face de um dado poliedro é também um poliedro.

Uma face F de P é *maximal* se F não está propriamente contida em nenhuma face diferente de P . Uma face maximal é também chamada de *faceta*. Podemos facilmente mostrar que:

$$(1.44) \quad \text{uma face } F \text{ de } P \text{ é uma faceta de } P \text{ se e somente se } \dim F = \dim P - 1.$$

As facetas de um poliedro P são de especial interesse pois são essenciais para sua descrição. Considere um sistema $Ax \leq b$. Dizemos que $Ax \leq b$ é *irredundante* se todo subsistema próprio $A'x \leq b'$ de $Ax \leq b$ possui uma solução x que não satisfaz $Ax \leq b$. Com isso:

Teorema 1.6. *Seja $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ um poliedro. O sistema $Ax \leq b$ é irredundante se e somente se há uma bijeção entre as inequações de $A^+x \leq b^+$ e as facetas de P .*

Em particular, segue que se P tem dimensão plena então as inequações que induzem facetas de P são únicas a menos de multiplicação por escalares positivos.

Uma face F de P é *minimal* se F não contém nenhuma outra face de P . Com isso:

Teorema 1.7. *Um conjunto F é uma face minimal de $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ se e só se*

$$(1.45) \quad F = \{x \in \mathbb{R}^n \mid A'x = b'\}$$

para algum subsistema $A'x \leq b'$ de $Ax \leq b$ com $\text{posto } A' = \text{posto } A$.

1.3.3. Poliedros inteiros

Dizemos que um poliedro P é *inteiro* se cada face de P contém um vetor inteiro. Se P é um poliedro qualquer, denotamos por P_I seu *fecho inteiro*, ou seja,

$$(1.46) \quad P_I := \text{fecho.conv} \{x \in P \mid x \text{ inteiro}\}.$$

É fácil ver que se P é um politopo então P_I é um poliedro. Se P é um poliedro geral, isso nem sempre é verdade. Entretanto, temos o seguinte resultado de Meyer (cf. Schrijver [42]):

Teorema 1.8. *Para qualquer poliedro $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ com A e b racionais temos que P_I também é um poliedro. Além disso, se P_I é não-vazio então $\text{cone.car } P_I = \text{cone.car } P$.*

Figura 1.2: (a) Gráficos das funções $1-x$ e e^{-x} . (b) Gráficos das funções $1-x/2$ e e^{-x} .

1.4. Um pouco sobre probabilidade

Nossa notação para probabilidades é padrão (veja, por exemplo, o livro de Ross [40]). Denotamos por $\Pr[A]$ a probabilidade do evento A . Denotamos por $\wedge$ a conjunção de eventos e por $\vee$ a disjunção de eventos. Denotamos por $\mathbb{E}[X]$ a esperança da variável aleatória X . Lembremos um resultado importante em probabilidade:

Teorema 1.9 (desigualdade de Markov). *Se X é uma variável aleatória que só assume valores não-negativos então*

$$(1.47) \quad \Pr[X \geq a] \leq \frac{\mathbb{E}[X]}{a}.$$

As seguintes desigualdades simples, que não dizem respeito especificamente à teoria das probabilidades, serão úteis mais adiante.

Teorema 1.10. *Temos que:*

- (i) $1-x \leq e^{-x}$ para todo x real;
- (ii) $1-x/2 \geq e^{-x}$ para todo $x \in [0, 1/2]$.

Prova: A Figura 1.2 ilustra as desigualdades apresentadas. Vamos primeiro provar (i). Para tanto, observe que a desigualdade é óbvia para $x \geq 1$. Vamos então demonstrá-la para $x < 1$. Pela expansão de Taylor de e^{-x} sabemos que

$$(1.48) \quad e^{-x} = 1 - x + \underbrace{\frac{x^2}{2!} - \frac{x^3}{3!} + \frac{x^4}{4!} - \cdots}_{R(x)}.$$

Note que, se $x \leq 0$, todos os termos de $R(x)$ são não-negativos, logo $R(x) \geq 0$. Se $0 < x < 1$, observe que para $k \geq 2$ par temos que $x^k > x^{k+1}$, logo $x^k/k! - x^{k+1}/(k+1)! > 0$, e também segue que $R(x) \geq 0$. Em ambos os casos, segue a desigualdade (i).

Para provar (ii), vamos primeiro mostrar que:

(1.49) se $f, g: \mathbb{R} \rightarrow \mathbb{R}$ são duas funções diferenciáveis, $f'(x) \leq g'(x)$ para todo $x \in [a, b]$ e $f(a) \leq g(a)$ então $f(x) \leq g(x)$ para todo $x \in [a, b]$.

Para tanto, fixe $x \in (a, b]$. Pelo teorema do valor médio aplicado à função $g - f$ no intervalo $[a, x]$, sabemos que existe $\xi \in (a, x)$ tal que

$$(1.50) \quad g'(\xi) - f'(\xi) = \frac{g(x) - f(x) - (g(a) - f(a))}{x - a}.$$

O lado esquerdo é não-negativo. Sabemos também que $g(a) - f(a) \geq 0$. Como $x - a > 0$, segue que $g(x) - f(x) \geq 0$, o que prova (1.49).

Agora, é fácil ver que (1.49) implica (ii). De fato, basta tomar $f(x) = e^{-x}$ e $g(x) = 1 - x/2$. Observe que $f(0) = g(0)$. Além disso, $f'(x) = -e^{-x}$ e $g'(x) = -1/2$. Resta apenas observar que $-e^{-x} \leq -1/2$ no intervalo $[0, 1/2]$. De fato, note que e^{-x} é uma função decrescente e que $e^{-1/2} > 1/2$, assim $e^{-x} \geq 1/2$ para $x \in [0, 1/2]$, como queríamos. $\square$

CAPÍTULO 2

O problema, aplicações e complexidade

Descrevemos no presente capítulo o problema de Steiner com grupos, tema deste trabalho, e discutimos sua relação com outros problemas de otimização combinatória. Apresentamos também uma aplicação prática do problema. Terminamos com um rápido resumo de resultados de complexidade computacional.

2.1. O problema

Seja $G = (V, E)$ um grafo e $\mathcal{R}$ uma coleção de subconjuntos de V . Um conjunto $U \subseteq V$ *cobre* $R \in \mathcal{R}$ se U contém pelo menos um vértice de R . Dizemos que U *cobre* $\mathcal{R}$ se U cobre todo conjunto em $\mathcal{R}$. Similarmente, um subgrafo H de G *cobre* $\mathcal{R}$ se VH cobre $\mathcal{R}$. Dizemos que uma subárvore T de G é uma $\mathcal{R}$ -*árvore* se T cobre $\mathcal{R}$. O *problema de Steiner com grupos*, introduzido por Reich e Widmayer [38], é o seguinte:

- (2.1) dados: um grafo $G = (V, E)$, uma coleção $\mathcal{R}$ de subconjuntos de V e uma função comprimento $l: E \rightarrow \mathbb{R}_+$,
encontre: uma $\mathcal{R}$ -árvore em G de comprimento mínimo.

Chamamos os conjuntos em $\mathcal{R}$ de *grupos*. Um vértice de G é um *vértice de grupo* se pertence a algum grupo, caso contrário é um *não-terminal* ou um *vértice de Steiner*. Muitas vezes nos referiremos a $\mathcal{R}$ como sendo uma *coleção de grupos*.

É geralmente útil supor que pelo menos um componente de G cubra $\mathcal{R}$, ou, em outras palavras, que o problema é viável. Vamos implicitamente fazer essa hipótese de agora em diante. Também vamos supor que os grafos considerados sejam simples. Ficará claro no texto quando for importante não fazer tais suposições.

O problema de Steiner com grupos é uma generalização do famoso *problema de Steiner em grafos*, que enunciamos a seguir:

- (2.2) dados: um grafo $G = (V, E)$, um subconjunto Z de V e uma função comprimento $l: E \rightarrow \mathbb{R}_+$,
encontre: uma árvore em G que contenha Z e tenha comprimento mínimo.

Chamamos os vértices em Z de *terminais*. Os demais vértices são *não-terminais* ou *vértices de Steiner*. É fácil ver que o problema de Steiner em grafos corresponde ao caso particular do problema de Steiner com grupos em que cada grupo tem exatamente um vértice.

Grupos dois-a-dois disjuntos. Muitas vezes, convém supor que os grupos sejam dois-a-dois disjuntos. Mostraremos a seguir que não se perde generalidade ao fazê-lo.

Seja $G = (V, E)$ um grafo, $\mathcal{R}$ uma coleção de grupos e $l: E \rightarrow \mathbb{R}_+$ uma função comprimento. Aplique o seguinte algoritmo a G , $\mathcal{R}$ e l :

- (2.3) (i) se os grupos são dois-a-dois disjuntos, pare;
(ii) sejam R, S dois grupos quaisquer tais que $R \cap S \neq \emptyset$. Para cada $v \in R \cap S$, insira em G dois novos vértices, v_R e v_S , ligados ao vértice v por arestas de comprimento zero; remova v de R e S e insira v_R em R e v_S em S . Volte ao passo (i).

A cada aplicação do passo (ii) do algoritmo um par de grupos deixa de ter intersecção. Ademais, o passo (ii) não cria novas intersecções. Assim, podemos concluir que da aplicação do algoritmo resulta um grafo $G' = (V', E')$, uma coleção de grupos $\mathcal{R}'$ e uma função comprimento $l': E' \rightarrow \mathbb{R}_+$ tais que $\mathcal{R}'$ possui grupos dois-a-dois disjuntos. Além disso, é fácil ver que a uma $\mathcal{R}$ -árvore T em G corresponde uma $\mathcal{R}'$ -árvore T' em G' tal que $l(ET) = l'(ET')$, e vice-versa. Em outras palavras, resolver o problema de Steiner com grupos para G' , $\mathcal{R}'$ e l' é o mesmo que o resolver para G , $\mathcal{R}$ e l .

Em geral, não vamos supor que os grupos sejam dois-a-dois disjuntos. Ficará claro no texto quando tal suposição for necessária.

2.2. Aplicação: roteamento em circuitos VLSI

Há duas fases importantes no desenvolvimento de circuitos VLSI¹: a fase de *colocação*, durante a qual se decide o lugar de cada componente do circuito, e a fase de *roteamento*, durante a qual os componentes são conectados por fios condutores. A Figura 2.1a mostra um circuito com seus diversos componentes. Cada componente tem um pino conector; durante a fase de roteamento desejamos ligar os pinos conectores através de uma rede condutora. Podemos usar para tanto os segmentos verticais e horizontais.

A questão é então como obter um roteamento de comprimento mínimo, sendo o comprimento definido de alguma forma razoável dependendo da aplicação que se tem em mente. Podemos usar o problema de Steiner em grafos para modelar o problema de roteamento. Para tanto, associamos um grafo ao circuito da maneira natural: a intersecção de dois segmentos dá origem a um vértice e dois vértices são adjacentes se são ligados por um segmento cujo interior não é intersectado por nenhum outro segmento. Em seguida, consideramos como terminais os pinos conectores. A Figura 2.1b mostra a solução ótima da instância do problema de Steiner associada ao circuito da Figura 2.1a.

Ao modelarmos o problema do roteamento através do problema de Steiner nós desprezamos uma outra operação que pode ser efetuada sobre os componentes do circuito. Após ter tido sua localização determinada no circuito, o componente ainda pode sofrer rotações e espelhamentos. Tais operações mudam a posição do pino conector do componente. Faz então

¹VLSI significa *very large scale integration*. É a tecnologia de circuitos integrados atualmente usada nos computadores.

Figura 2.1: (a) A representação de um circuito. Os componentes estão representados em cinza e os pinos conectores estão destacados. (b) Uma solução ótima para a instância do problema de Steiner em grafos obtida do circuito em (a). Supomos que cada aresta tenha comprimento 1. O comprimento da solução é 24.

sentido perguntar qual o conjunto de rotações e espelhamentos que fornece o roteamento de menor comprimento.

O problema de Steiner com grupos pode ser usado para modelar esse outro aspecto do problema de roteamento. Novamente, associamos um grafo ao circuito da maneira natural. Para cada componente do circuito temos um grupo de vértices que contém os vértices correspondentes às possíveis posições do pino conector do componente, levando-se em conta todas as possíveis rotações e espelhamentos do componente. A Figura 2.2 ilustra essa idéia.

Foi por conta dessa aplicação que Reich e Widmayer [38] propuseram o problema de Steiner com grupos. Rohe e Zachariasen [39] estudam mais a fundo essa aplicação e apresentam um algoritmo exato para a versão retilinear do problema, que está mais diretamente ligada ao problema do roteamento.

2.3. Complexidade computacional

Karp [33] mostrou que a versão de decisão do problema de Steiner em grafos é NP-completa. Como o problema de Steiner com grupos é uma generalização do problema de Steiner em grafos, segue imediatamente que sua versão de decisão é NP-completa e que o problema em si é NP-difícil. Nesta seção, entretanto, mostraremos como reduzir o problema da cobertura mínima por conjuntos ao problema de Steiner com grupos, uma abordagem que fornecerá resultados de complexidade mais fortes.

Seja A um conjunto finito. Uma coleção $\mathcal{S}$ de subconjuntos de A *cobre* A se cada elemento de A pertence a algum conjunto em $\mathcal{S}$. Uma $\mathcal{S}$ -cobertura de A é um conjunto $\mathcal{S}' \subseteq \mathcal{S}$ que cobre A . O *problema da cobertura mínima por conjuntos* é o seguinte:

- (2.4) dados: um conjunto A , uma coleção $\mathcal{S}$ de subconjuntos do conjunto A e uma função peso $w: \mathcal{S} \rightarrow \mathbb{R}_+$,
 encontre: uma $\mathcal{S}$ -cobertura de A que tenha peso mínimo.

Figura 2.2: (a) A instância do problema de Steiner com grupos associada ao problema do roteamento da Figura 2.1b. As possíveis localizações dos conectores de um componente estão destacadas. Conectores num mesmo componente estão num mesmo grupo. (b) Uma solução ótima para a instância em (a). O comprimento da solução é 16.

Muitas vezes nos referiremos ao conjunto A como sendo o *conjunto base*.

Considere agora um conjunto A , uma coleção $\mathcal{S}$ de subconjuntos de A e uma função peso $w: \mathcal{S} \rightarrow \mathbb{R}_+$. Considere também o grafo $G = (V, E)$, a coleção de grupos $\mathcal{R}$ e a função comprimento $l: E \rightarrow \mathbb{R}_+$ dados da seguinte forma:

- (2.5) (i) G contém um vértice r e, para cada $S \in \mathcal{S}$, um vértice v_S ligado ao vértice r por uma aresta de comprimento w_S ;
(ii) para cada $a \in A$ temos um grupo $R_a \in \mathcal{R}$, sendo $R_a := \{v_S \in V \mid a \in S\}$.

Há uma correspondência entre $\mathcal{S}$ -coberturas de A e $\mathcal{R}$ -árvores em G . Mais especificamente:

- (2.6) a cada $\mathcal{S}$ -cobertura S' de A corresponde uma $\mathcal{R}$ -árvore T em G tal que $w(S') = l(ET)$, e vice-versa. A $\mathcal{R}$ -árvore correspondente a uma dada $\mathcal{S}$ -cobertura ou a $\mathcal{S}$ -cobertura correspondente a uma dada $\mathcal{R}$ -árvore podem ser encontradas em tempo polinomial.

De fato, se T é uma $\mathcal{R}$ -árvore em G então, para cada $a \in A$, T possui um vértice $v_S \in R_a$, com $S \in \mathcal{S}$ e $a \in S$. Mas então $S' := \{S \in \mathcal{S} \mid v_S \in VT\}$ é uma $\mathcal{S}$ -cobertura de A tal que $w(S') = l(ET)$. Analogamente, se S' é uma $\mathcal{S}$ -cobertura de A então $T := G[\{r\} \cup \{v_S \in V \mid S \in S'\}]$ é uma $\mathcal{R}$ -árvore em G , com comprimento igual ao peso de S' .

Através de (2.5) podemos reduzir o problema da cobertura mínima por conjuntos ao problema de Steiner com grupos. Esse fato foi observado por Ihler [31] e por Lund e Yannakakis [35].

Teorema 2.1. *A versão de decisão do problema de Steiner com grupos é NP-completa mesmo se restrita a:*

- (i) *estrelas com comprimentos unitários nas arestas;*
(ii) *árvores com comprimentos 0-1 nas arestas, sendo os grupos dois-a-dois disjuntos.*

Prova: O grafo construído a partir de (2.5) é uma estrela. Como a versão de decisão do problema da cobertura mínima por conjuntos é NP-completa mesmo quando os pesos são todos unitários (cf. Garey e Johnson [18]), (i) segue imediatamente.

Para ver (ii), basta observar que podemos aplicar à instância do problema de Steiner com grupos dada por (2.5) o algoritmo (2.3). Na instância resultante, os grupos são dois-a-dois disjuntos e as novas arestas inseridas têm comprimento zero. O grafo não deixa de ser uma árvore. $\square$

Ihler [31] também estudou outras restrições do problema de Steiner com grupos e provou que ele permanece NP-difícil mesmo em diversos casos particulares. Nesse ponto, temos um contraste em relação ao problema de Steiner em grafos, já que vários de seus casos particulares podem ser resolvidos em tempo polinomial (e.g., o problema de Steiner restrito a árvores).

2.3.1. Resultados de inaproximabilidade

Um *algoritmo de aproximação* para um problema de minimização é um algoritmo que, dada uma instância do problema, devolve uma solução viável de custo no máximo α vezes o custo da solução ótima, para algum α que pode ou não depender da instância em questão. O parâmetro α é chamado de *razão de aproximação* do algoritmo. Também nos podemos referir a um tal algoritmo como sendo uma α -*aproximação*. Definições mais precisas desses conceitos podem ser encontradas em [13].

Como foi observado em (2.6), a redução obtida a partir de (2.5) preserva o custo das soluções. Com isso, qualquer algoritmo de aproximação para o problema de Steiner com grupos pode ser utilizado para o problema da cobertura mínima por conjuntos, preservando-se a razão de aproximação.

A dificuldade de aproximar o problema da cobertura mínima por conjuntos foi estudada por diversos autores, entre eles Lund e Yannakakis [35] e Feige [12]. Feige obteve o seguinte resultado:

Teorema 2.2. *Para todo $\epsilon > 0$ não existe uma $(\epsilon \ln n)$ -aproximação polinomial para o problema da cobertura mínima por conjuntos, sendo n o número de elementos do conjunto base, a não ser que $\text{NP} \subseteq \text{DTIME}(n^{O(\log \log n)})$.*

Segue como corolário desse resultado um limitante semelhante para o problema de Steiner com grupos.

Corolário 2.3. *Para todo $\epsilon > 0$ não existe uma $(\epsilon \ln t)$ -aproximação polinomial para o problema de Steiner com grupos restrito a estrelas, sendo t o número de grupos, a não ser que $\text{NP} \subseteq \text{DTIME}(n^{O(\log \log n)})$.*

Prova: Segundo (2.6), a redução dada por (2.5) preserva o custo das soluções. Além disso, o grafo resultante da redução é uma estrela e temos um grupo para cada elemento do conjunto base da instância original do problema da cobertura por conjuntos. O resultado segue então do Teorema 2.2. $\square$

Recentemente, Halperin e Krauthgamer [26] obtiveram um resultado de inaproximabilidade para o problema de Steiner com grupos com um limitante mais forte, embora sob uma hipótese diferente da hipótese de que $\text{NP} \not\subseteq \text{DTIME}(n^{O(\log \log n)})$.

Observe o contraste que temos em relação ao problema de Steiner, para o qual são conhecidos algoritmos de aproximação de razão constante (veja, por exemplo, o livro de Hwang, Richards e Winter [29]).

2.3.2. Do problema de Steiner com grupos para o problema de Steiner

Podemos reduzir o problema de Steiner com grupos ao problema de Steiner. Considere um grafo $G = (V, E)$, uma coleção de grupos $\mathcal{R}$ e uma função comprimento $l: E \rightarrow \mathbb{R}_+$. Escreva $M := l(E) + 1$. Considere também o grafo $G' = (V', E')$, o conjunto de terminais Z e a função comprimento $l': E' \rightarrow \mathbb{R}_+$ dados da seguinte forma:

- (i) G é subgrafo de G' ;
- (ii) l' coincide com l nas arestas de G ;
- (iii) para cada grupo $R \in \mathcal{R}$, G' possui um vértice v_R ligado a todos os vértices em R por arestas de comprimento M ;
- (iv) $Z := \{v_R \in V' \mid R \in \mathcal{R}\}$.

Com isso, devido à escolha de M , uma árvore em G' que conecta todos os terminais em Z e tem comprimento mínimo fornece-nos uma $\mathcal{R}$ -árvore em G de comprimento mínimo.

A redução que acabamos de descrever não preserva os custos das soluções e, dessa forma, não é útil para o desenvolvimento de algoritmos de aproximação. Duin, Volgenant e Voß [10] estudaram a aplicabilidade dessa redução em problemas práticos e concluíram que ela é eficaz. Obviamente, isso depende muito do tipo de instâncias consideradas, em particular do “diâmetro” dos grupos.

Não faremos uso dessa redução. É nossa esperança que técnicas desenvolvidas especificamente para o problema de Steiner com grupos, como aquelas propostas por Rohe e Zachariasen [39], sejam mais eficazes num grupo maior de instâncias.

2.4. Mais sobre o problema de Steiner

O problema de Steiner é um dos problemas mais estudados em otimização combinatória. Vários aspectos do problema foram estudados, como algoritmos de aproximação, formulações e poliedros, relaxações lagrangeanas, etc. O leitor interessado pode encontrar uma resenha do assunto, completa até 1992, no livro de Hwang, Richards e Winter [29]. Uma bibliografia extensa sobre o assunto pode ser encontrada na biblioteca eletrônica sobre o problema de Steiner, a Steinlib, de Koch, Martin e Voß [34].

CAPÍTULO 3

Algoritmos de aproximação

Escolhemos descrever neste capítulo três algoritmos de aproximação propostos para o problema de Steiner com grupos. Incluímos também algumas referências para outros algoritmos propostos para o problema.

3.1. Heurística dos caminhos mínimos

Reich e Widmayer [38] propuseram a primeira heurística para o problema de Steiner com grupos. Ela assemelha-se muito à heurística proposta por Takahashi e Matsuyama [46] para o problema de Steiner em grafos.

Seja $G = (V, E)$ um grafo conexo, $\mathcal{R}$ uma coleção de grupos e $l: E \rightarrow \mathbb{R}_+$ uma função comprimento. A heurística de Reich e Widmayer é a seguinte:

- (3.1)
- (i) fixe um vértice de grupo r_0 de forma arbitrária;
 - (ii) seja T o grafo composto apenas por r_0 ;
 - (iii) se T cobre $\mathcal{R}$, devolva T ;
 - (iv) se T não cobre $\mathcal{R}$, escolha dentre os vértices dos grupos não cobertos por T aquele mais próximo de algum vértice de T . Seja r o vértice de grupo escolhido e v um vértice de T mais próximo dele. Conecte r a T através de um v - r caminho de comprimento mínimo em G de modo a manter T uma árvore. Volte para o passo (iii).

Observe que no passo (iv) podemos de fato manter T uma árvore: basta adicionarmos o menor pedaço possível do caminho de comprimento mínimo considerado. Também é fácil ver que o algoritmo é polinomial: a cada execução do passo (iv) T passa a cobrir pelo menos mais um grupo, logo o passo (iv) é executado no máximo $|\mathcal{R}| - 1$ vezes, já que r_0 é um vértice de grupo. Note que podemos utilizar o algoritmo de Dijkstra [7] (veja também Schrijver [43], capítulo 7) para encontrar o v - r caminho no passo (iv), o que pode ser feito em tempo $O(m \log n)$. Assim, o algoritmo todo consome tempo $O(|\mathcal{R}|m \log n)$.

Está claro que o algoritmo (3.1) devolve uma $\mathcal{R}$ -árvore, já que supomos que G seja conexo. Se todos os grupos são unitários, ou seja, se estamos tratando de uma instância do problema de Steiner em grafos, o algoritmo acima coincide com o algoritmo proposto por Takahashi e Matsuyama [46] para o problema de Steiner em grafos. Sabe-se que nesse caso o comprimento da $\mathcal{R}$ -árvore devolvida é não superior a 2 vezes o comprimento de uma $\mathcal{R}$ -árvore ótima. Em

Figura 3.1: (a) Um exemplo ruim para o algoritmo de Reich e Widmayer. Nesse exemplo, os vértices de mesma cor pertencem a um mesmo grupo. Os comprimentos são dados nas arestas. Uma possível solução usa apenas a aresta de comprimento 1. Outra possível solução usa ambas as arestas. Como podemos escolher M de forma arbitrária, se escolhermos o vértice r_0 no passo (i) do algoritmo, podemos obter uma solução tão distante da ótima quanto desejado. (b) Um exemplo justo para o algoritmo de Reich e Widmayer com a modificação de Ihler. Nesse exemplo temos $k + 1$ grupos, dados por $R_0 := \{r_0\}$ e $R_i := \{u_i, v_i\}$, $i = 1, \dots, k$. Os comprimentos são dados nas arestas, sendo ϵ um número convenientemente pequeno. Se fixarmos o grupo R_0 no algoritmo, vamos obter uma solução de comprimento k , enquanto a solução ótima tem comprimento $1 + \epsilon$.

geral, entretanto, a heurística de Reich e Widmayer pode devolver $\mathcal{R}$ -árvores tão distantes da ótima quanto desejado, como mostra o exemplo da Figura 3.1a.

Uma modificação bastante simples do algoritmo, proposta por Ihler [30], elimina o problema que acabamos de apontar. Ihler propôs o seguinte algoritmo:

- (3.2) fixe um grupo $R_0 \in \mathcal{R}$ e, para cada $r \in R_0$, execute a heurística de Reich e Widmayer com r no lugar de r_0 no passo (i). Devolva a $\mathcal{R}$ -árvore de menor comprimento dentre as encontradas.

Temos o seguinte resultado:

Teorema 3.1. *Seja T a $\mathcal{R}$ -árvore devolvida pelo algoritmo de Ihler e seja T^* uma $\mathcal{R}$ -árvore de comprimento mínimo. Então $l(ET) \leq (|\mathcal{R}| - 1)l(ET^*)$.*

Prova: Observe que, como T^* é uma $\mathcal{R}$ -árvore, algum vértice de R_0 , digamos r_0 , deve ocorrer em T^* . Considere a árvore $\hat{T}$ computada pela heurística de Reich e Widmayer quando tomamos r_0 como o vértice do passo (i). Afirmamos que $l(E\hat{T}) \leq (|\mathcal{R}| - 1)l(ET^*)$.

Para ver isso, sejam $r_1, \dots, r_k$ os vértices de grupos não cobertos escolhidos no passo (iv) do algoritmo e $v_1, \dots, v_k$ os vértices do grafo sendo construído aos quais eles são ligados, sendo que v_i é ligado a r_i , $i = 1, \dots, k$. Como T^* cobre $\mathcal{R}$, para cada $i = 1, \dots, k$, T^* contém um vértice r_i^* do mesmo grupo de r_i . Denote por $T^*(u, v)$ o caminho entre u e v em T^* . Então, quando r_i é escolhido no passo (iv) do algoritmo temos que

$$(3.3) \quad \text{dist}_l(v_i, r_i) \leq \text{dist}_l(r_0, r_i^*) \leq l(E(T^*(r_0, r_i^*))) \leq l(ET^*),$$

sendo $\text{dist}_l(u, v)$ o comprimento, segundo l , de um u - v caminho de comprimento mínimo em G .

Como o passo (iv) é executado no máximo $|\mathcal{R}| - 1$ vezes, temos que

$$(3.4) \quad l(ET) \leq \sum_{i=1}^k \text{dist}_l(v_i, r_i) \leq (|\mathcal{R}| - 1)l(ET^*),$$

e segue a afirmação. Como a $\mathcal{R}$ -árvore devolvida pelo algoritmo de Ihler tem comprimento não superior ao comprimento de $\hat{T}$, segue o teorema. $\square$

A razão de aproximação $|\mathcal{R}| - 1$ é assintoticamente justa, como mostra o exemplo da Figura 3.1b.

3.2. Um algoritmo simples para árvores

Consideramos agora uma variante do problema de Steiner com grupos, o *problema de Steiner com grupos em árvores enraizadas*, que é o seguinte:

- (3.5) dados: uma árvore $T = (V, E)$ de raiz s , uma coleção de grupos $\mathcal{R}$ e uma função comprimento $l: E \rightarrow \mathbb{R}_+$,
 encontre: uma subárvore de T que contenha s , cubra $\mathcal{R}$ e tenha comprimento mínimo.

Obviamente, podemos supor que s não faça parte de nenhum grupo. Uma subárvore de T que contém s e cobre $\mathcal{R}$ é chamada de s - $\mathcal{R}$ -árvore.

Do ponto de vista de algoritmos de aproximação, o problema acima é tão geral quanto o problema de Steiner com grupos em árvores, na medida em que qualquer algoritmo de aproximação para ele também fornece um algoritmo de aproximação de mesma razão para o problema de Steiner com grupos em árvores.

De fato, se dispomos de um algoritmo de aproximação para (3.5) e temos uma árvore $T = (V, E)$, uma coleção de grupos $\mathcal{R}$ e uma função comprimento $l: E \rightarrow \mathbb{R}_+$, basta fixar um grupo R_0 e executar nosso algoritmo de aproximação em T , $\mathcal{R} \setminus \{R_0\}$ e l uma vez para cada vértice de R_0 no lugar da raiz s , escolhendo a melhor solução encontrada.

O algoritmo que vamos apresentar tem grande semelhança com o algoritmo de Hochbaum [28] para o problema da cobertura por conjuntos. Considere $T = (V, E)$, s , $\mathcal{R}$ e l como em (3.5). Nosso algoritmo baseia-se no seguinte programa linear:

$$(3.6) \quad \begin{aligned} &\text{minimizar} && l^\top x \\ & && \text{(i) } x \geq \mathbf{0}, \\ & && \text{(ii) } x(\delta(U)) \geq 1, \text{ para todo } s\text{-}R \text{ corte } \delta(U), R \in \mathcal{R}. \end{aligned}$$

Observe que uma solução ótima inteira de (3.6) fornece uma solução ótima para (3.5). O problema da separação para (3.6) pode ser resolvido em tempo polinomial através de uma sequência de problemas de corte mínimo. Basta observar que um vetor $x \in \mathbb{R}_+^E$ viola uma inequação em (3.6)(ii) se e somente se existe um grupo R tal que o menor s - R corte em T , com capacidades dadas por x , tem capacidade menor do que 1. Podemos encontrar tal corte em tempo polinomial, inserindo um novo vértice t em T e ligando-o a todo vértice de R por arestas de custo infinito. Um s - t corte mínimo nesse novo grafo fornece-nos um s - R corte

mínimo no grafo original, e vice-versa. Repetindo essa idéia para cada grupo em $\mathcal{R}$, podemos determinar se x satisfaz todas as inequações (3.6)(ii), ou encontrar uma delas violada por x .

Pela discussão acima, sabemos que (3.6) pode ser resolvido em tempo polinomial (cf. Grötschel, Lovász e Schrijver [23]). Considere então o seguinte algoritmo:

- (3.7) (i) seja x^* uma solução ótima de (3.6). Seja N o tamanho do maior grupo em $\mathcal{R}$;
(ii) considere o conjunto $F := \{e \in E \mid x^*(e) \geq 1/N\}$;
(iii) devolva o componente de (V, F) que contém s .

Observe que, pela nossa discussão anterior, o algoritmo acima é polinomial. Além disso, temos:

Teorema 3.2. *Seja T^* uma s - $\mathcal{R}$ -árvore de comprimento mínimo. A árvore $\hat{T}$ devolvida pelo algoritmo (3.7) é uma s - $\mathcal{R}$ -árvore tal que $l(E\hat{T}) \leq N \cdot l(ET^*)$.*

Para provarmos esse resultado, faremos uso da seguinte observação, que pode ser demonstrada facilmente:

- (3.8) se $\delta(U)$ é um s - W corte minimal em T para algum $W \subseteq V \setminus \{s\}$, então $\delta(U)$ tem no máximo $|W|$ arestas.

Com isso, podemos provar o Teorema 3.2.

Prova do Teorema 3.2: Que $\hat{T}$ contém s é óbvio, vamos mostrar que $\hat{T}$ cobre $\mathcal{R}$. Para tanto, fixe $R \in \mathcal{R}$ e seja $\delta(U)$ um s - R corte minimal em T . Segue de (3.8) que $\delta(U)$ contém no máximo N arestas.

Com isso, como x^* satisfaz as inequações (3.6)(ii), segue que pelo menos uma aresta e de cada s - R corte é tal que $x^*(e) \geq 1/N$, sendo assim escolhida no passo (ii) do algoritmo. Mas então $\hat{T}$ contém um vértice de R . Como R é arbitrário, $\hat{T}$ cobre $\mathcal{R}$, como queríamos.

Resta provar que $l(E\hat{T}) \leq N \cdot l(ET^*)$. Isso é fácil, visto que

$$(3.9) \quad l(E\hat{T}) \leq \sum_{e \in F} l(e)(Nx^*(e)) \leq N \cdot l^T x^* \leq N \cdot l(ET^*),$$

como queríamos. □

Observe que, quando T é uma estrela e s é seu centro, estamos tratando de uma instância do problema da cobertura mínima por conjuntos (cf. seção 2.3) e, nesse caso, nosso algoritmo coincide com o de Hochbaum [28].

3.3. O algoritmo de Garg, Konjevod e Ravi

Garg, Konjevod e Ravi [19] propuseram um algoritmo de aproximação para o problema de Steiner com grupos em árvores enraizadas, problema esse que foi descrito no início da seção anterior.

O algoritmo de Garg, Konjevod e Ravi arredonda uma solução ótima do programa linear (3.6), como fazia o algoritmo da seção anterior. Entretanto, ele faz esse arredondamento de forma probabilística (cf. Raghavan e Thompson [37]): um determinado experimento

aleatório, baseado numa solução ótima de (3.6), é repetido um certo número de vezes, e garantimos que o resultado de tal experimento é uma solução de comprimento pequeno com alta probabilidade.

Antes de ler esta seção, o leitor talvez queira lembrar nossa notação para árvores enraizadas olhando a seção 1.2.1.

3.3.1. Um experimento aleatório

Começamos por descrever um experimento aleatório que é central ao algoritmo de Garg, Konjevod e Ravi. Mais adiante, faremos uma análise desse experimento.

Seja $T = (V, E)$ uma árvore de raiz s . Se e é uma aresta de T que não incide em s , denote por $\pi(e)$ a aresta que precede e num caminho de s à ponta mais profunda de e . Dizemos que um vetor $x \in \mathbb{R}^E$ é s -decrecente se $x(e) \leq x(\pi(e))$ para toda aresta e de T que não incide em s .

Considere então um vetor s -decrecente x tal que $0 \leq x \leq 1$. Podemos executar o seguinte experimento aleatório sobre T e x :

- (3.10) para cada aresta e de T , escolha e com probabilidade $x(e)$ se e incide em s ou com probabilidade $x(e)/x(\pi(e))$ caso contrário. Denote por F_x o conjunto das arestas escolhidas e por T_x o componente de (V, F_x) que contém s .

Diremos que T_x é a árvore *resultante* do experimento ou a árvore *gerada* pelo experimento.

Mostraremos na seção seguinte que se x é solução básica de (3.6) então x é s -decrecente e $0 \leq x \leq 1$. Dessa forma, o experimento acima pode ser aplicado a soluções básicas ótimas de (3.6), que é o que o algoritmo de Garg, Konjevod e Ravi faz para arredondar tais soluções. Observe, entretanto, que nossa descrição é mais geral. Isso será conveniente para analisarmos o experimento.

A principal propriedade do experimento (3.10) é descrita no seguinte teorema.

Teorema 3.3. *Seja $T = (V, E)$ uma árvore de raiz s e $\mathcal{R}$ uma coleção de grupos dois-a-dois disjuntos. Suponha que todo vértice de grupo seja uma folha de T . Seja x uma solução viável s -decrecente de (3.6) tal que $0 \leq x \leq 1$ e fixe um grupo $R \in \mathcal{R}$. A probabilidade de T_x conter um vértice de R é pelo menos $1/128 \lg N$, sendo N o tamanho do maior grupo em $\mathcal{R}$.*

A demonstração do Teorema 3.3 será feita na seção 3.3.3. Por ora, observe que o teorema nos diz que se repetirmos o experimento (3.10) um número de vezes convenientemente grande, podemos aumentar a probabilidade de uma árvore que cobre um grupo R ser encontrada.

3.3.2. O algoritmo

Seja $T = (V, E)$ uma árvore de raiz s , $\mathcal{R}$ uma coleção de grupos e $l: E \rightarrow \mathbb{R}_+$ uma função comprimento. Vamos supor que os grupos sejam dois-a-dois disjuntos (cf. seção 2.1) e que os vértices de grupo sejam folhas de T . Isso não significa perda de generalidade: se o vértice de grupo v não é folha de T , podemos inserir em T um novo vértice v' , ligá-lo a v por uma aresta de comprimento zero, remover v do grupo ao qual pertence e adicionar v' a esse grupo.

Obtemos uma nova instância equivalente à original. Repetindo o raciocínio, podemos obter uma instância em que os vértices de grupo são folhas de T .

No que segue, ignoramos chãos e tetos por uma questão de estética, mas todos os cálculos podem ser feitos levando-os em consideração. Considere um real ϵ , $0 < \epsilon < 1$. Seja N o tamanho do maior grupo e t o número de grupos. O algoritmo de Garg, Konjevod e Ravi é o seguinte:

- (i) seja x^* uma solução ótima básica de (3.6);
- (ii) para $i = 1, \dots, 4 \ln 1/\epsilon$, repita o experimento (3.10) $128 \lg N \ln 4t$ vezes e denote por T_i a união das árvores geradas em cada execução do experimento;
- (iii) dentre as árvores T_i que cobrem $\mathcal{R}$, devolva a de menor comprimento. Se nenhuma tal árvore cobre $\mathcal{R}$, devolva T .

Para que o passo (i) do algoritmo faça sentido, x^* tem de ser s -decrecente e tal que $0 \leq x^* \leq 1$. Mostraremos mais adiante que esse é o caso.

Obviamente, o algoritmo devolve uma s - $\mathcal{R}$ -árvore e é polinomial. Temos também que:

Teorema 3.4. *Seja $\hat{T}$ a árvore devolvida pelo algoritmo de Garg, Konjevod e Ravi e T^* uma s - $\mathcal{R}$ -árvore de comprimento mínimo. Então $l(\hat{ET}) \leq (512 \lg N \ln 4t)l(ET^*)$ com probabilidade pelo menos $1 - \epsilon$.*

Prova: Seja T' a árvore resultante de uma execução do experimento (3.10) sobre T e x^* . Afirmamos que o comprimento esperado de T' é $l^T x^*$. De fato, se e é uma aresta de T e v é sua ponta mais profunda, incluímos e em T' se e somente se incluímos todas as arestas num caminho de s a v em T' . Sejam $e_0, \dots, e_k = e$ as arestas do s - v caminho de T , na ordem em que ocorrem quando o caminho é percorrido de s a v . Como incluímos as arestas de forma independente, a probabilidade de incluirmos e em T' é $x^*(e_0) \cdot \prod_{i=1}^k (x^*(e_i)/x^*(e_{i-1})) = x^*(e)$. Logo, pela linearidade da esperança, o comprimento esperado de T' é $l^T x^*$, o que prova a afirmação.

Fixe um $i = 1, \dots, 4 \ln 1/\epsilon$. Afirmamos que:

$$(3.11) \quad T_i \text{ cobre } \mathcal{R} \text{ e tem comprimento não superior a } (512 \lg N \ln 4t)l(ET^*) \text{ com probabilidade pelo menos } 1/4.$$

Para ver isso, note primeiro que o comprimento esperado de T_i é no máximo a soma dos comprimentos esperados das $128 \lg N \ln 4t$ árvores que a compõe, ou seja, é no máximo $(128 \lg N \ln 4t)l^T x^*$. Segue então da desigualdade de Markov que

$$(3.12) \quad \Pr[l(ET_i) \geq 4 \cdot (128 \lg N \ln 4t)l^T x^*] \leq 1/4.$$

Como $l^T x^* \leq l(ET^*)$, temos que o comprimento de T_i é no máximo $(512 \lg N \ln 4t)l(ET^*)$ com probabilidade pelo menos $3/4$.

Agora, fixe um grupo $R \in \mathcal{R}$. Vem do Teorema 3.3 que T_i não cobre R com probabilidade no máximo

$$(3.13) \quad \left(1 - \frac{1}{128 \lg N}\right)^{128 \lg N \ln 4t} \leq (e^{-1/128 \lg N})^{128 \lg N \ln 4t} = e^{-\ln 4t} = 1/4t.$$

Acima, usamos que $1 - \lambda \leq e^{-\lambda}$ para todo λ real (cf. Teorema 1.10). Mas então a probabilidade de T_i não conter um vértice de algum grupo é no máximo $1/4$ e, portanto, T_i cobre $\mathcal{R}$ com probabilidade pelo menos $3/4$. Com isso provamos (3.11).

Usando (3.11), a probabilidade de nenhuma das árvores T_i cobrir $\mathcal{R}$ e ter comprimento baixo é no máximo $(1 - 1/4)^{4 \ln 1/\epsilon} \leq \epsilon$. Logo, alguma das árvores T_i cobre $\mathcal{R}$ e tem comprimento baixo com probabilidade pelo menos $1 - \epsilon$, como queríamos. $\square$

Note que, através da escolha de ϵ , podemos aumentar a probabilidade de sucesso arbitrariamente, embora também aumentemos o tempo gasto pelo algoritmo.

Resta apenas mostrar, como prometemos, que $0 \leq x^* \leq 1$ e que x^* é s -decrecente. Mostraremos mais geralmente que isso vale para toda solução básica de (3.6).

Considere uma solução básica x de (3.6). Obviamente temos $0 \leq x \leq 1$. Vamos provar que x é s -decrecente. Suponha que a aresta e de T seja tal que $x(e) > x(\pi(e))$. Sejam $u, v, w \in V$ tais que $\pi(e) = uv$ e $e = vw$. Considere o vetor $\tilde{x} \in \mathbb{R}^E$ tal que $\tilde{x}(f) := x(f)$ para toda aresta $f \in E \setminus \{e\}$ e $\tilde{x}(e) := x(\pi(e))$. Afirmamos que $\tilde{x}$ é solução viável de (3.6).

De fato, considere um s - R corte $\delta(U)$ para algum $R \in \mathcal{R}$. Suponha ainda que $e \in \delta(U)$, caso contrário não há nada a ser feito. Temos dois casos a considerar. Primeiro, se $v \in U$, então denote por X o conjunto de vértices do componente de $T - uv$ que contém v . Escreva $U' := U \setminus X$. Note que $\delta(U')$ também é um s - R corte e que $\delta(U') \subseteq (\delta(U) \setminus \{e\}) \cup \{\pi(e)\}$. Mas então $\tilde{x}(\delta(U)) \geq \tilde{x}(\delta(U')) = x(\delta(U')) \geq 1$.

Suponha agora que $w \in U$. Denote por X o conjunto de vértices do componente de $T - vw$ que contém w e escreva $U' := U \setminus X$. Novamente, $\delta(U')$ é um s - R corte e $\delta(U') \subseteq \delta(U) \setminus \{e\}$. Logo $\tilde{x}(\delta(U)) \geq \tilde{x}(\delta(U')) = x(\delta(U')) \geq 1$ e finalizamos a prova de que $\tilde{x}$ é uma solução viável de (3.6).

Considere agora o vetor $\hat{x}$ que é tal que $\hat{x}(f) := x(f)$ para todo $f \neq e$ e $\hat{x}(e) = 2x(e)$. Como $\tilde{x}(e) < x(e) < \hat{x}(e)$, x é combinação convexa não-trivial de $\tilde{x}$ e $\hat{x}$, uma contradição. Com isso, provamos que x é s -decrecente, como queríamos.

3.3.3. Análise do experimento

Concluimos nossa exposição do algoritmo de Garg, Konjevod e Ravi com a demonstração do Teorema 3.3. Nele, como os vértices de grupo são folhas de T , queremos limitar inferiormente a probabilidade de T_x conter uma aresta em $\delta(R)$. Para cada $e \in \delta(R)$, seja A_e o conjunto das arestas do caminho de s à ponta mais profunda de e . Note que incluímos e em T_x se e somente se $A_e \subseteq ET_x$. Seja B_e o evento $A_e \subseteq ET_x$. Queremos um limitante inferior para

$$(3.14) \quad \Pr \left[\bigvee_{e \in \delta(R)} B_e \right] = 1 - \Pr \left[\bigwedge_{e \in \delta(R)} \overline{B_e} \right].$$

Se os eventos B_e , $e \in \delta(R)$, fossem independentes, teríamos que

$$(3.15) \quad \Pr \left[\bigwedge_{e \in \delta(R)} \overline{B_e} \right] = \prod_{e \in \delta(R)} \Pr[\overline{B_e}].$$

Entretanto, esse não precisa ser o caso. De fato, para $e, f \in \delta(R)$, B_e, B_f são independentes se e somente se $A_e \cap A_f = \emptyset$. Para encontrarmos um limitante para (3.15), vamos fazer uso da desigualdade de Janson [32] (veja também o capítulo 8 do livro de Alon e Spencer [1]).

Para introduzirmos a desigualdade de Janson vamos primeiro introduzir alguma notação. Seja X um subconjunto das folhas de T . Dados $e, f \in \delta(X)$, escrevemos $e \sim f$ se $A_e \cap A_f \neq \emptyset$, i.e., se os eventos B_e, B_f são dependentes. Escreva $\Delta := \sum_{e \sim f} \Pr[B_e \wedge B_f]$, sendo a soma feita sobre todos os pares ordenados, e escreva $\mu := \sum_{e \in \delta(X)} \Pr[B_e]$. A desigualdade de Janson, já adaptada para o nosso caso, é a seguinte:

Teorema 3.5 (desigualdade de Janson). *Sejam $B_e, e \in \delta(X)$, Δ e μ como acima e suponha que $\Pr[B_e] \leq \epsilon$ para todo $e \in \delta(X)$. Então*

$$(3.16) \quad \Pr \left[\bigwedge_{e \in \delta(X)} \overline{B_e} \right] \leq \exp \left(-\mu + \frac{1}{1-\epsilon} \frac{\Delta}{2} \right).$$

Se, adicionalmente, $\Delta \geq \mu(1-\epsilon)$, então

$$(3.17) \quad \Pr \left[\bigwedge_{e \in \delta(X)} \overline{B_e} \right] \leq \exp \left(-\frac{\mu^2(1-\epsilon)}{2\Delta} \right).$$

No que segue, se T é uma árvore, denotamos por LT o conjunto das folhas de T . Antes de provarmos o Teorema 3.3, precisamos de dois resultados intermediários. O primeiro é o seguinte:

Lema 3.6. *Seja $T = (V, E)$ uma árvore de raiz s e W um subconjunto das folhas de T . Seja $x \in \mathbb{R}^E$ um vetor s -decrecente tal que $\mathbf{0} \leq x \leq \mathbf{1}$. Suponha que para algum $\beta \geq 0$ tenhamos que $x(\delta(U)) \geq \beta$ para todo s - W corte $\delta(U)$. Existe um vetor s -decrecente $\tilde{x}$, $\mathbf{0} \leq \tilde{x} \leq x$, tal que*

$$(3.18) \quad \begin{aligned} & \text{(i) } \tilde{x}(\delta^{\text{out}}(s)) = \beta; \\ & \text{(ii) para todo } v \in V \setminus \{s\} \text{ que não é folha de } T \text{ temos que } \tilde{x}(\delta^{\text{in}}(v)) = \tilde{x}(\delta^{\text{out}}(v)); \\ & \text{(iii) se } T' \text{ é a subárvore de } T \text{ enraizada em } v \text{ então } \tilde{x}(\delta^{\text{out}}(v)) = \tilde{x}(\delta(LT')). \end{aligned}$$

Prova: Considere o grafo dirigido $D = (V \cup \{a, b\}, A)$ e a função capacidade $c: A \rightarrow \mathbb{R}_+$ dados da seguinte forma:

- (i) para cada $uv \in E$, sendo v a ponta mais profunda de uv , D possui um arco (u, v) com capacidade $x(uv)$;
- (ii) D possui um arco (a, s) com capacidade β ;
- (iii) para cada $w \in W$, D possui um arco (w, b) com capacidade infinita.

Como $x(\delta(U)) \geq \beta$ para cada s - W corte $\delta(U)$ de T , temos que $c(\delta_D^{\text{out}}(U)) \geq \beta$ para cada a - b corte de D . Além disso, o a - b corte $\delta_D^{\text{out}}(\{a\})$ tem capacidade exatamente β . Logo, pelo teorema do fluxo máximo corte mínimo, temos que se f é um a - b fluxo máximo em D sujeito a c então seu valor é exatamente β . Agora, é fácil ver que um tal fluxo máximo nos fornece um vetor $\tilde{x}$ com as propriedades desejadas. $\square$

O segundo resultado de que precisamos é o seguinte:

Lema 3.7. *Seja $T = (V, E)$ uma árvore de raiz s e W um subconjunto das folhas de T . Sejam $x', x'' \in \mathbb{R}^E$ vetores s -decrecentes tais que $0 \leq x', x'' \leq 1$. Suponha que x', x'' difiram apenas no componente e , de modo que $x'(e) \geq x''(e)$. A probabilidade de $T_{x'}$ conter um vértice de W é maior ou igual à probabilidade de $T_{x''}$ conter um vértice de W .*

Prova: Seja A o evento em que $T_{x'}$ possui um vértice de W na subárvore abaixo da aresta e . Então $\Pr[\bar{A}] = (1 - x'(e)) + x'(e) \Pr[\bar{A} \mid e \text{ é escolhida}]$. Note que o coeficiente de $x'(e)$ nessa expressão é sempre negativo. Logo, se $x'(e)$ é decrescido, $\Pr[\bar{A}]$ aumenta e $\Pr[A]$ decresce. $\square$

Com isso, estamos prontos para provar o Teorema 3.3.

Prova do Teorema 3.3: Se $|R| = 1$, a probabilidade de T_x conter um vértice de R é 1. Suponha então que $|R| \geq 2$.

Atribua 0 a cada componente de x que corresponde a uma aresta de T que não ocorre num caminho de s a algum vértice de R e arredonde os valores resultantes para baixo para a potência de 2 mais próxima. Segundo o Lema 3.7, a probabilidade de T_x conter um vértice de R só decresceu. Note também que agora temos $x(\delta(U)) \geq 1/2$ para cada s - R corte $\delta(U)$ de T .

Escreva $d := \lceil \lg |R| \rceil$ e atribua 0 a cada componente de x cujo valor é menor ou igual a $1/2^{d+2}$. Novamente, a probabilidade de sucesso só diminui. Além disso, usando (3.8), o valor de cada s - R corte minimal diminui no máximo $|R|/2^{d+2} \leq 1/4$, logo $x(\delta(U)) \geq 1/4$ para cada s - R corte $\delta(U)$.

Finalmente, contraia cada aresta e de T que não incida em s e que seja tal que $x(e) = x(\pi(e))$, a não ser que e seja incidente a uma folha. Denote por T' a árvore resultante e por x' a restrição de x às arestas de T' . Note que x' é s -decrecente para T' e que a altura de T' é no máximo $d + 3$. Além disso, a probabilidade de $T_{x'}$ conter um vértice de R é igual à probabilidade de T_x conter um vértice de R . Aplicando o Lema 3.6 a T' e x' com $\beta := 1/4$ e $W := R$, obtemos um vetor s -decrecente $\tilde{x}$, $0 \leq \tilde{x} \leq x'$, que satisfaz (3.18)(i)–(iii).

Vamos calcular Δ para T' e $\tilde{x}$. Dadas arestas $e, f \in \delta_{T'}(R)$ tais que $A_e \cap A_f \neq \emptyset$, denote por $\pi(e, f)$ o ancestral comum mais próximo das pontas mais profundas de e e f . Dada uma aresta $e \in \delta_{T'}(R)$, escreva

$$(3.19) \quad \Delta_e := \sum_{f \sim e} \Pr[B_e \wedge B_f] = \sum_{f \sim e} \frac{\tilde{x}(e)\tilde{x}(f)}{\tilde{x}(\delta^{\text{in}}(\pi(e, f)))}.$$

(Acima, convencionamos que $x(\delta^{\text{in}}(s)) = 1$.)

É fácil ver que Δ_e é a contribuição de e a Δ , de modo que $\Delta = \sum_{e \in \delta_{T'}(R)} \Delta_e$. Vamos estimar Δ_e . Seja $s = v_0, e_0, v_1, e_1, \dots, v_k, e_k = e, v_{k+1}$ o s - v_{k+1} caminho em T' , sendo v_{k+1} a ponta mais profunda de e . Para cada $i = 1, \dots, k$, denote por H_i a subárvore de T' enraizada em v_i e denote por T'_i o componente de $H_i - e_i$ que contém v_i . Escreva $h_i := \tilde{x}(\delta_{T'_i}^{\text{out}}(v_i))$. A Figura 3.2 ilustra essas definições.

Figura 3.2: A figura ilustra as definições dadas na demonstração do Teorema 3.3. Note que o vértice v_i também faz parte da árvore T'_i , marcada em cinza na figura, e que mais de uma aresta pode ligar v_i a vértices de T'_i .

Como $\tilde{x}$ satisfaz (3.18)(ii), $h_i \leq \tilde{x}(e_{i-1})$ para $i = 1, \dots, k$. Além disso, como $\tilde{x}$ satisfaz (3.18)(iii), temos que

$$\begin{aligned}
 (3.20) \quad \Delta_e &= \sum_{i=1}^k \sum_{f \in LT'_i} \frac{\tilde{x}(e)\tilde{x}(f)}{\tilde{x}(e_{i-1})} \\
 &= \sum_{i=1}^k \frac{\tilde{x}(e)}{\tilde{x}(e_{i-1})} \sum_{f \in LT'_i} \tilde{x}(f) \\
 &= \sum_{i=1}^k \frac{\tilde{x}(e)h_i}{\tilde{x}(e_{i-1})} \\
 &\leq \tilde{x}(e)(d+2).
 \end{aligned}$$

Mas então temos

$$(3.21) \quad \Delta = \sum_{e \in \delta_{T'}(R)} \Delta_e \leq \tilde{x}(\delta_{T'}(R))(d+2) = (d+2)/4 \leq \lg |R|,$$

pois $|R| \geq 2$. Vamos agora aplicar a desigualdade de Janson. Note que temos $\mu = 1/4$ e podemos fazer $\epsilon := 1/4$. Se $\Delta \leq \mu(1 - \epsilon) = 3/16$, vem de (3.16) que

$$(3.22) \quad \Pr \left[\bigwedge_{e \in \delta(R)} \overline{B_e} \right] \leq e^{-1/8} \leq .89.$$

Por outro lado, se $\Delta \geq \mu(1 - \epsilon)$, vem de (3.17) que

$$(3.23) \quad \Pr \left[\bigwedge_{e \in \delta(R)} \overline{B_e} \right] \leq e^{-3/2 \cdot 64 \lg |R|} \leq e^{-1/64 \lg |R|} \leq 1 - 1/128 \lg |R|,$$

sendo que a última desigualdade segue do fato de que $e^{-\lambda} \leq 1 - \lambda/2$ para todo real λ no intervalo $(0, 1/2)$ (cf. Teorema 1.10). Mas então, como $1 - 1/128 \lg |R| \geq e^{-1/8}$, a probabilidade de T_x conter um vértice de R é pelo menos $1/128 \lg |R| \geq 1/128 \lg N$. $\square$

3.4. Observações finais

Os algoritmos de aproximação para o problema de Steiner com grupos restrito a árvores podem ser adaptados para o problema em grafos gerais. O processo envolve a aproximação das distâncias no grafo por árvores de forma probabilística. Usando o algoritmo de Fakcharoenphol, Rao e Talwar [11] para fazer uma tal aproximação, as razões de aproximação são multiplicadas por $O(\lg n)$, sendo n o número de vértices do grafo. Para mais informações, veja [2, 11, 19]. Mencionamos também que nossa descrição do algoritmo de Garg, Konjevod e Ravi corrige alguns erros contidos no artigo original e é mais detalhada em alguns pontos.

Slavík [44], em sua tese de doutorado, propôs alguns algoritmos de aproximação para o problema de Steiner com grupos. Helvig, Robins e Zelikovsky [27] também propõem um algoritmo de aproximação para o problema.

CAPÍTULO 4

Testes de redução

Neste capítulo apresentamos diversas técnicas de redução de instâncias do problema de Steiner com grupos. Tais técnicas permitem que vértices e arestas sejam removidos do grafo original, o que potencialmente diminui o tempo gasto por algoritmos que resolvem o problema. Neste capítulo supomos que existam pelo menos dois grupos.

4.1. Distâncias de gargalo

Começamos por apresentar o conceito de distâncias de gargalo, que será usado no restante deste capítulo. Esse conceito foi introduzido por Rohe e Zachariasen [39] e baseia-se numa idéia similar introduzida por Duin e Volgenant [9] para o problema de Steiner em grafos.

Introduzimos primeiro o conceito de gargalo. Seja $G = (V, E)$ um grafo e $l: E \rightarrow \mathbb{R}_+$ uma função comprimento. Dados $u, v \in V$, o *gargalo*, segundo l , entre u e v é

$$(4.1) \quad \text{garg}_l(u, v) := \min_P \max_{e \in EP} l(e),$$

sendo que acima o mínimo é tomado sobre todos os u - v caminhos P em G . Um caminho que atinge o mínimo é chamado de *caminho de gargalo* entre u e v . A Figura 4.1a ilustra essas definições.

Podemos computar o gargalo entre u e v em tempo polinomial (cf. Duin [8]). Na verdade, é um exercício simples provar que, se T é uma árvore geradora de G de comprimento mínimo, então o gargalo entre u e v é igual ao gargalo em T entre u e v . Como há só um u - v caminho em T , é fácil encontrar esse último (veja a Figura 4.1b).

Considere então um grafo $G = (V, E)$ e uma coleção de grupos $\mathcal{R}$. Dizemos que um conjunto $W \subseteq V$ é *completo* para $\mathcal{R}$ se W contém exatamente um vértice de cada grupo em $\mathcal{R}$ e não contém nenhum outro vértice. Denotamos por $\sigma(\mathcal{R})$ a coleção dos conjuntos $W \subseteq V$ que são completos para $\mathcal{R}$.

Considere também uma função comprimento $l: E \rightarrow \mathbb{R}_+$. Para $u, v \in V$, denotamos por $\text{dist}_l(u, v)$ o comprimento, segundo l , de um u - v caminho de comprimento mínimo em G . Denotamos por K_G o grafo completo que possui o mesmo conjunto de vértices de G . Observe que podemos ver dist_l como sendo uma função comprimento definida sobre as arestas de K_G .

Seja $W \subseteq V$ um conjunto completo para $\mathcal{R}$ e considere dois vértices $u, v \in V$. Denote por d a restrição de dist_l às arestas de $K_G[W \cup \{u, v\}]$. O *W -gargalo* entre u e v , segundo l , é

$$(4.2) \quad \text{garg}_l^W(u, v) := \text{garg}_d(u, v),$$

Figura 4.1: (a) No grafo da figura há dois tipos de arestas: as pontilhadas e as cheias. As pontilhadas têm comprimento 1 e as cheias têm comprimento 2. O gargalo entre dois vértices é 1 se eles se encontram numa mesma diagonal pontilhada ou 2 caso contrário. (b) Uma árvore geradora de comprimento mínimo do grafo em (a). Observe como fica explícito o que afirmamos anteriormente: o gargalo entre dois vértices é 1 se eles pertencem à mesma diagonal pontilhada ou 2 caso contrário.

ou seja, o W -gargalo entre u e v é o gargalo entre u e v no grafo $K_G[W \cup \{u, v\}]$. Com isso, o $\mathcal{R}$ -gargalo entre os vértices u e v de G , segundo l , é

$$(4.3) \quad \text{garg}_l^{\mathcal{R}}(u, v) := \max_{W \in \sigma(\mathcal{R})} \text{garg}_l^W(u, v).$$

É fácil ver que $\text{garg}_l^{\mathcal{R}}(u, v) \leq \text{dist}_l(u, v)$ para todo par de vértices $u, v \in V$.

Um caminho de gargalo entre u e v em $K_G[W \cup \{u, v\}]$ induz um passeio em G , obtido da concatenação dos caminhos de comprimento mínimo em G que correspondem às suas arestas. Tal passeio é chamado de *passeio de gargalo para W* entre u e v . Observe que um tal passeio pode ser dividido em caminhos entre vértices consecutivos de W . Tais caminhos não contêm vértices de W como vértices internos; eles serão chamados de caminhos *elementares*. Observe que o comprimento de qualquer caminho elementar de um passeio de gargalo para W entre u e v é no máximo $\text{garg}_l^W(u, v)$. A Figura 4.2 ilustra as definições que acabamos de apresentar.

É fácil ver que, dados vértices $u, v \in V$ e um conjunto completo W , podemos computar $\text{garg}_l^W(u, v)$ em tempo polinomial (já que podemos computar o gargalo entre dois vértices em tempo polinomial, como observamos anteriormente). Entretanto, computar $\text{garg}_l^{\mathcal{R}}(u, v)$ é, em geral, um problema NP-difícil, como mostraram Rohe e Zachariasen [39].

4.2. Remoção de arestas e não-terminais

Usando o conceito de $\mathcal{R}$ -gargalo apresentado na seção anterior, Rohe e Zachariasen [39] mostraram que é possível adaptar para o problema de Steiner com grupos os testes de remoção de arestas e não-terminais que foram propostos por Duin e Volgenant [9] (veja também Duin [8]) para o problema de Steiner em grafos. Nesta seção mostramos como isso é possível.

Figura 4.2: (a) Um grafo G e uma função comprimento definida sobre suas arestas. O conjunto W é o conjunto dos vértices cheios. (b) O grafo $K_G[W]$. Em suas arestas temos a distância em G entre os vértices de W . O caminho destacado em cinza é um caminho de gargalo em $K_G[W]$ entre u e v . Esse caminho fornece o W -gargalo entre u e v no grafo G . (c) O passeio obtido da concatenação dos caminhos mínimos correspondentes às arestas de $K_G[W]$. Abaixo do passeio estão destacados seus caminhos elementares.

Novamente, seja $G = (V, E)$ um grafo, $\mathcal{R}$ uma coleção de grupos e $l: E \rightarrow \mathbb{R}_+$ uma função comprimento. Começamos com um teste para a remoção de arestas:

Teorema 4.1. *Seja $uv \in E$ uma aresta tal que $\text{garg}_l^{\mathcal{R}}(u, v) < l(uv)$. Então uv não pertence a nenhuma $\mathcal{R}$ -árvore de comprimento mínimo em G .*

Prova: Suponha que T seja uma $\mathcal{R}$ -árvore de comprimento mínimo em G que contém uv . Observe que T deve conter um conjunto completo para $\mathcal{R}$, digamos W . Seja P um passeio de gargalo para W entre u e v .

Escreva $T' := T - uv$. Observe que T' possui exatamente dois componentes, um que contém u e o outro que contém v . Como $W \subseteq VT$ e como P é um u - v passeio, algum subcaminho P' de P vai conectar os componentes de T' e esse subcaminho estará contido em algum caminho elementar de P . Mas então temos que $T' + P'$ é uma $\mathcal{R}$ -árvore e que seu comprimento é no máximo

$$(4.4) \quad l(ET) - l(uv) + \text{garg}_l^W(u, v) \leq l(ET) - l(uv) + \text{garg}_l^{\mathcal{R}}(u, v) < l(ET),$$

uma contradição. $\square$

Em outras palavras, qualquer aresta uv que satisfaz as condições do Teorema 4.1 pode ser removida antes de se atacar uma determinada instância do problema.

O próximo teste que apresentaremos irá possibilitar a remoção de vértices de Steiner. Para apresentá-lo, vamos primeiro introduzir alguma notação. Seja $v \in V$ um vértice qualquer. Denotamos por $N_G(v)$ a *vizinhança* de v , i.e., o conjunto dos vértices que são adjacentes a v no grafo G . Se não houver perigo de ambigüidade, escrevemos $N(v)$ em vez de $N_G(v)$.

Para conjuntos disjuntos $A, B \subseteq V$, denotaremos por $\delta(A, B)$ o conjunto das arestas com uma ponta em A e a outra em B . Para $v \in V$, escreveremos $\delta(v, B)$ em vez de $\delta(\{v\}, B)$, por exemplo.

Da mesma forma que podemos ver dist_l como uma função comprimento definida sobre as arestas de K_G , também podemos ver $\text{garg}_l^{\mathcal{R}}$ como uma função comprimento definida sobre

as arestas desse grafo. Dado um conjunto $U \subseteq V$, denotaremos por $\tau(U)$ o comprimento, segundo $\text{garg}_l^{\mathcal{R}}$, de uma árvore geradora mínima de $K_G[U]$. Com isso, temos o seguinte resultado:

Teorema 4.2. *Seja v um vértice não-terminal de G tal que para todo $U \subseteq N(v)$ com $|U| \geq 2$ temos $\tau(U) < l(\delta(v, U))$. Então existe uma $\mathcal{R}$ -árvore de comprimento mínimo que não contém v .*

Na demonstração do Teorema 4.2 faremos uso do seguinte lema, que também nos será útil mais adiante:

Lema 4.3. *Seja F uma floresta em G que contém um conjunto completo W . Seja $U \subseteq V \setminus W$ um conjunto que contém pelo menos um vértice de cada componente de F . Existe uma árvore em G que contém F e cujo comprimento é no máximo $l(EF) + \tau(U)$.*

Prova: Seja H uma árvore geradora de $K_G[U]$ de comprimento mínimo segundo $\text{garg}_l^{\mathcal{R}}$ e seja uv sua aresta de menor comprimento. Considere um passeio de gargalo P para W entre u e v . Percorra esse passeio de u para v . Cada vez que um caminho que conecta componentes diferentes de F é encontrado, as arestas desse caminho são adicionadas a F . Note que tal caminho é subcaminho de algum caminho elementar do passeio P , tendo portanto comprimento no máximo $\text{garg}_l^W(u, v) \leq \text{garg}_l^{\mathcal{R}}(u, v)$. Depois de percorrer P , se F ainda não é conexo, repita o mesmo procedimento, escolhendo a segunda aresta mais curta de H , e assim por diante.

Após adicionarmos no máximo $|U| - 1$ caminhos a F , temos um grafo conexo. Tal grafo contém uma árvore que contém F . Como o comprimento total dos caminhos adicionados é no máximo $\tau(U)$, o comprimento dessa árvore é no máximo $l(EF) + \tau(U)$. $\square$

Com isso, podemos provar o Teorema 4.2.

Prova do Teorema 4.2: Seja T uma $\mathcal{R}$ -árvore de comprimento mínimo que contém v . Como v é um vértice de Steiner, podemos supor que o grau de v em T seja pelo menos 2, caso contrário o resultado segue imediatamente. Escreva $U := N_T(v)$ e $F := T - v$. Observe que F possui exatamente $|U|$ componentes, cada um deles contém um vértice de U . Usando o Lema 4.3, temos que existe uma $\mathcal{R}$ -árvore que contém F e cujo comprimento é no máximo

$$(4.5) \quad l(EF) + \tau(U) = l(ET) - l(\delta(v, U)) + \tau(U) < l(ET),$$

uma contradição. $\square$

Outros testes que foram propostos para o problema de Steiner em grafos para a remoção de vértices de Steiner e arestas, como aqueles propostos por Polzin e Daneshmand [36], podem ser adaptados para o problema de Steiner com grupos de maneira semelhante. Na seção seguinte veremos um exemplo do uso da técnica de Polzin e Daneshmand para a remoção de vértices de grupo.

4.3. Remoção de vértices de grupo

Vimos na seção anterior testes para a remoção de arestas e não-terminais. Vejamos agora alguns testes para a remoção de vértices de grupo, isto é, testes que nos permitirão concluir que um dado vértice de grupo pode ser considerado um vértice de Steiner. Em outras palavras, poderemos concluir que um dado vértice de grupo pode ser removido dos grupos aos quais pertence sem destruir pelo menos uma solução ótima do problema original. As idéias desta seção podem ser encontradas em [14].

Para facilitar nossa exposição, supomos nesta seção que os grupos sejam dois-a-dois disjuntos. Obviamente, como estamos lidando com testes de redução, o uso do algoritmo descrito na seção 2.1 para obter-se uma instância em que os grupos são dois-a-dois disjuntos não faz sentido, visto que tal algoritmo acrescenta vértices e arestas ao grafo. Entretanto, essa restrição faz sentido em boa parte das aplicações práticas do problema. Além disso, os testes aqui apresentados podem ser facilmente modificados para serem utilizados no caso de grupos não disjuntos.

Como os grupos são dois-a-dois disjuntos, podemos denotar por $\text{grupo}(v)$ o grupo ao qual o vértice v pertence.

4.3.1. Um teste simples

Seja $G = (V, E)$ um grafo, $\mathcal{R}$ uma coleção de grupos e $l: E \rightarrow \mathbb{R}_+$ uma função comprimento. Dado um vértice de grupo r e um conjunto $U \subseteq V$ tal que $r \notin U$, escreva

$$(4.6) \quad \pi(r, U) := \min\{\text{dist}_l(u, v) \mid u \in U, v \in \text{grupo}(r) \setminus \{r\}\},$$

ou seja, $\pi(r, U)$ é a distância de U ao vértice mais próximo do mesmo grupo de r que é diferente de r . Com isso, temos:

Teorema 4.4. *Seja $r \in R$, $R \in \mathcal{R}$, um vértice de grupo. Se para cada $U \subseteq N(r) \setminus R$, $|U| \geq 1$, temos que $\tau(U) + \pi(r, U) \leq l(\delta(r, U))$, então r pode ser considerado um não-terminal.*

Prova: Para mostrarmos que r pode ser considerado um não-terminal, mostraremos que existe uma $\mathcal{R}$ -árvore de comprimento mínimo que contém um vértice em $R \setminus \{r\}$.

Para tanto, considere uma $\mathcal{R}$ -árvore T de comprimento mínimo e suponha que r seja o único vértice de R que ocorre em T , caso contrário nada temos a fazer. Escreva $U := N_T(r)$ e $F := T - r$. Observe que $U \subseteq N(r) \setminus R$. Sejam $u \in U$ e $v \in R \setminus \{r\}$ tais que $\text{dist}_l(u, v) = \pi(r, U)$. Conecte v a F usando um subcaminho de um u - v caminho de comprimento mínimo em G de modo que o grafo resultante, F' , continue a ser uma floresta.

Observe que F' cobre $\mathcal{R}$ e, portanto, contém um conjunto completo para $\mathcal{R}$. Além disso, F' tem no máximo $|U|$ componentes e cada um desses componentes contém um vértice de U . Logo, aplicando o Lema 4.3, temos que existe uma $\mathcal{R}$ -árvore T' que contém F' e cujo comprimento é no máximo

$$(4.7) \quad l(EF') + \tau(U) \leq l(ET) - l(\delta(r, U)) + \pi(r, U) + \tau(U) \leq l(ET),$$

e podemos concluir que T' é uma $\mathcal{R}$ -árvore de comprimento mínimo. Mas T' contém um vértice em $R \setminus \{r\}$, mais especificamente o vértice v , como queríamos. $\square$

Figura 4.3: Uma instância do problema de Steiner com grupos. As intersecções dos segmentos do desenho são vértices. Há três grupos: ●, ○ e □. O vértice no centro, que pertence ao grupo ●, pode ser considerado um não-terminal: qualquer $\mathcal{R}$ -árvore que o contém também contém algum outro vértice de seu grupo. Entretanto, um teste que apenas inspeciona seus vizinhos não tem como chegar a essa conclusão.

Observe que o Teorema 4.4 é semelhante ao Teorema 4.2. Um caso particular interessante do Teorema 4.4 ocorre quando todos os vizinhos do vértice de grupo r pertencem ao mesmo grupo que r . Nesse caso, é óbvio que existe uma $\mathcal{R}$ -árvore de comprimento mínimo que contém algum vértice do grupo de r que é diferente de r .

4.3.2. Testes de expansão

O teste apresentado no Teorema 4.4 é um teste *local* pois, para decidir se determinado vértice pode ser considerado um vértice de Steiner, verifica apenas condições locais, que dizem respeito aos vizinhos do vértice em questão. Por esse motivo, é possível que vértices que possam ser considerados vértices de Steiner não sejam identificados, como ilustra a Figura 4.3.

Para eliminar esse problema, Polzin e Daneshmand [36], apoiados num trabalho anterior de Uchoa, de Aragão e Ribeiro [47], propuseram os chamados *testes de expansão*, que analisam não só estruturas locais. Tais testes foram propostos no contexto da remoção de arestas e vértices de Steiner de instâncias do problema de Steiner em grafos. Mostraremos nesta seção como adaptar a idéia de Polzin e Daneshmand para a remoção de vértices de grupo.

Seja $G = (V, E)$ um grafo, $\mathcal{R}$ uma coleção de grupos e $l: E \rightarrow \mathbb{R}_+$ uma função comprimento. Começamos por introduzir algumas definições.

Seja T uma árvore em G . Novamente, denotaremos por LT o conjunto das folhas de T . Considere agora uma subárvore T' de T . O *conjunto de ligação* entre T' e T é o conjunto dos vértices v de T' que são ligados a alguma folha de T por um caminho em T que não possui nenhuma aresta de T' . Dizemos que T' está *periféricamente contida* em T se o conjunto de

Figura 4.4: A subárvore destacada em cinza não está perifericamente contida na árvore maior. O conjunto de ligação, que é o conjunto dos vértices brancos, não é composto apenas por folhas da subárvore cinza.

ligação entre T' e T é LT' . A Figura 4.4 ilustra essas definições.

Se T é uma árvore em G e u é uma folha de T , escrevemos $\mathcal{X}(u, T) := \{uv \in EG \mid v \notin VT\}$. Chamamos qualquer subconjunto não-vazio de $\mathcal{X}(u, T)$ de uma *expansão* de T através de u . Se r é um vértice de grupo, dizemos que a árvore T *respeita* r se T contém r e todo outro vértice de grupo que T contém é folha de T .

O algoritmo que vamos apresentar para a remoção de vértices de grupo é recursivo. Ele recebe uma árvore T que respeita um vértice de grupo r e tenta decidir se r pode ser considerado um não-terminal analisando T . Se não for possível obter essa conclusão, o algoritmo tenta expandir a árvore T e, para cada expansão, executar a si mesmo.

Mais especificamente, seja T^* uma $\mathcal{R}$ -árvore de comprimento mínimo que não possui vértices de Steiner como folhas. É fácil ver que uma tal árvore existe. Há duas versões do nosso algoritmo. A primeira versão recebe uma árvore T que respeita um vértice de grupo r e devolve SIM se a seguinte afirmação for verdadeira:

- (4.8) se T está perifericamente contida em T^* e r não é folha de T^* , então existe uma $\mathcal{R}$ -árvore de comprimento mínimo que contém um vértice em $\text{grupo}(r) \setminus \{r\}$.

A segunda versão do algoritmo recebe uma árvore T que respeita um vértice de grupo r que é folha de T e devolve SIM se a seguinte afirmação for verdadeira:

- (4.9) se T está perifericamente contida em T^* e r é folha de T^* , então existe uma $\mathcal{R}$ -árvore de comprimento mínimo que contém um vértice em $\text{grupo}(r) \setminus \{r\}$.

Inicialmente, a árvore T é composta apenas de r . Nesse caso, dizer que T está perifericamente contida em T^* significa dizer que T^* contém r . Logo, se ambas as versões do algoritmo devolvem SIM quando T é apenas o vértice r , sabemos que r pode ser considerado um não-terminal.

As duas versões do algoritmo são basicamente iguais. Abaixo, esboçamos a idéia de ambas as versões. Comentaremos sobre as diferenças mais adiante.

- (4.10) (i) teste a árvore T . Se for possível concluir que (4.8) (ou (4.9), no caso da segunda versão) vale, devolva SIM;

- (ii) para cada folha u de T faça o seguinte: se u for um não-terminal ou se $u = r$, para cada possível expansão X de T através de u execute o algoritmo recursivamente para $T + X$. Se a resposta, para cada possível expansão, for SIM, devolva SIM;
- (iii) se todas as tentativas de expansão falharam, devolva NÃO.

A diferença entre a versão que testa (4.8) e a versão que testa (4.9) é pequena. No primeiro caso, podemos expandir de qualquer forma no passo (ii). Na verdade, quando T é apenas o vértice r , podemos considerar apenas as expansões com mais de uma aresta. No segundo caso, devemos manter verdadeiro o fato de r ser folha de T , o que limita nossas expansões no passo (ii) da forma óbvia.

Observe que a execução do passo (i) não foi detalhada. Há diversas condições que podemos usar para concluir que (4.8) ou (4.9) valem, como mostra o seguinte teorema:

Teorema 4.5. *Seja T uma árvore que respeita um vértice de grupo $r \in R$, $R \in \mathcal{R}$. Então:*

- (4.11) (i) *se $\tau(LT) + \pi(r, LT \setminus \{r\}) \leq l(ET)$, então se T está periféricamente contida em T^* existe uma $\mathcal{R}$ -árvore de comprimento mínimo que contém um vértice em $R \setminus \{r\}$;*
- (ii) *se $\tau(LT) < l(ET)$ e r é folha de T , então T não está periféricamente contida em T^* ;*
- (iii) *se r é folha de T e $\tau(LT \setminus \{r\}) + \pi(r, LT \setminus \{r\}) \leq l(ET)$, então se T está periféricamente contida em T^* e r é folha de T^* existe uma $\mathcal{R}$ -árvore de comprimento mínimo que contém um vértice de $R \setminus \{r\}$.*

Prova: Para ver (4.11)(i), suponha que T esteja periféricamente contida em T^* . Seja I o conjunto dos vértices internos de T e escreva $F := (T^* - ET) - I$. Sejam $u \in LT \setminus \{r\}$ e $v \in R \setminus \{r\}$ tais que $\text{dist}_l(u, v) = \pi(r, LT \setminus \{r\})$. Conecte v a F usando um subcaminho de um u - v caminho de comprimento mínimo em G de modo que o grafo resultante, F' , continue a ser uma floresta.

Note que, como T respeita r , F' contém um conjunto completo para $\mathcal{R}$. Além disso, como T está periféricamente contida em T^* , F' possui no máximo $|LT|$ componentes, cada um deles contém um vértice de LT . Mas então, usando o Lema 4.3, temos que existe uma $\mathcal{R}$ -árvore $\hat{T}$ em G que contém F' e cujo comprimento é no máximo

$$(4.12) \quad l(EF') + \tau(LT) \leq l(ET^*) - l(ET) + \pi(r, LT \setminus \{r\}) + \tau(LT) \leq l(ET^*),$$

e podemos concluir que $\hat{T}$ também tem comprimento mínimo. Mas $\hat{T}$ possui um vértice em $R \setminus \{r\}$, e segue o resultado.

Podemos provar (4.11)(ii) de maneira análoga. De fato, suponha que T esteja periféricamente contida em T^* . Denote por I o conjunto dos vértices internos de T e escreva $F := (T^* - ET) - I$. Observe que, como T respeita r e como r é folha de T , F contém um conjunto completo para $\mathcal{R}$. Além disso, como T está periféricamente contida em T^* , F tem exatamente $|LT|$ componentes, cada um deles contém um vértice de LT . Mas então, usando o Lema 4.3, temos que existe uma $\mathcal{R}$ -árvore $\hat{T}$ que contém F e cujo comprimento é no máximo

$$(4.13) \quad l(EF) + \tau(LT) \leq l(ET^*) - l(ET) + \tau(LT) < l(ET^*),$$

uma contradição.

A demonstração de (4.11)(iii) pode ser feita de maneira análoga, levando-se em consideração o fato de r ser folha de T^* . $\square$

Observe como foi importante na demonstração do Teorema 4.5 a hipótese de T estar perifericamente contida em T^* ou mesmo a hipótese de T respeitar r . Ambas essas restrições, que podem parecer artificiais num primeiro momento, são essenciais para o desenvolvimento de bons testes.

As condições de teste (4.11)(i) e (4.11)(ii) podem ser usadas com ambas as versões do algoritmo de expansão. Já (4.11)(iii) deve ser usada com a versão do algoritmo que testa (4.9).

Corretude do teste de expansão. Resta mostrar que a estratégia de expansão que apresentamos está correta. Para ver isso observe que, se T' é uma subárvore da árvore T , então:

- (4.14) se T' está perifericamente contida em T então para toda folha u de T' que não é folha de T existe um conjunto não-vazio $X \subseteq \mathcal{X}(u, T')$ tal que $T' + X$ está perifericamente contida em T .

De fato, suponha que T' esteja perifericamente contida em T e seja u uma folha de T' que não é folha de T . Escreva $X := \{uv \in ET \mid v \notin VT'\}$. Observe que $X \subseteq \mathcal{X}(u, T')$ e que, como u não é folha de T , X é não-vazio. Afirmamos que $T' + X$ está perifericamente contida em T .

Para ver isso, considere primeiro um vértice interno v de $T' + X$. Se esse vértice é um vértice interno de T' , como T' está perifericamente contida em T , todo caminho de v a uma folha de T utiliza alguma aresta de T' . Se temos $v = u$, todo caminho de v a alguma folha de T utiliza ou uma aresta de T' ou uma aresta de X , utilizando assim alguma aresta de $T' + X$.

Agora, se v é uma folha de $T' + X$ então é fácil ver que, como $T' + X$ é subárvore de T , existe um caminho de v a alguma folha de T que só usa arestas que não ocorrem em $T' + X$. Com isso, terminamos de provar (4.14).

Usando (4.14) fica fácil ver que sempre que o algoritmo de expansão, quando executado para alguma árvore T e vértice de grupo r , devolve SIM, temos que (4.8) (ou (4.9), no caso da segunda versão do algoritmo) vale. De fato, observe que toda expansão que consideramos respeita o vértice de grupo r . Além disso, só expandimos por vértices que sabemos que não são folhas em T^* . Dessa forma, se para toda possível expansão $T + X$ da árvore T a partir da folha u temos que vale (4.8) (ou (4.9)), a afirmação deve valer para a árvore T , já que se T está perifericamente contida em T^* , alguma expansão também deve estar.

4.3.3. Um teste adicional

O teste que vamos apresentar agora é semelhante ao do Teorema 4.4 e pode ser combinado com o teste de expansão de forma a obter-se mais reduções. Para tanto, considere um grafo $G = (V, E)$, uma coleção de grupos $\mathcal{R}$ e uma função comprimento $l: E \rightarrow \mathbb{R}_+$.

Dados conjuntos disjuntos $A, B \subseteq V$, denotamos por $\underline{\delta}_G(A, B)$ o conjunto que contém, para cada $u \in B$, uma aresta de $\delta_G(A, B)$ incidente em u que tenha comprimento mínimo, se

tal aresta existir. Denote por $\hat{\mathcal{R}}$ o conjunto de todos os vértices de grupo, i.e., a união dos grupos em $\mathcal{R}$. Para todo conjunto $U \subseteq V$ denotamos por $N_G(U)$ a *vizinhança* de U em G , i.e., o conjunto dos vértices em $V \setminus U$ que são adjacentes em G a algum vértice em U . Com isso, temos:

Teorema 4.6. *Seja $r \in R$, $R \in \mathcal{R}$, um vértice de grupo. Suponha que o conjunto $C \subseteq V$ seja tal que $C \cap \hat{\mathcal{R}} = \{r\}$. Se para cada $U \subseteq N(C)$ com $|U| \geq 1$ temos $\tau(U) + \pi(r, U) \leq l(\delta(C, U))$, então existe uma $\mathcal{R}$ -árvore de comprimento mínimo que contém um vértice em $R \setminus \{r\}$.*

Prova: A prova é similiar à prova do Teorema 4.4. Seja T^* uma $\mathcal{R}$ -árvore de comprimento mínimo na qual r ocorre. Escreva $C' := C \cap VT^*$ e $U := N_{T^*}(C')$. Note que $U \subseteq N_G(C)$ e que, como $C \cap \hat{\mathcal{R}} = \{r\}$, U é não-vazio. Seja $F := T^* - C'$. Escolha $u \in U$ e $v \in R \setminus \{r\}$ tais que $\text{dist}_l(u, v) = \pi(r, U)$. Conecte v a F usando um subcaminho de um u - v caminho de comprimento mínimo em G de modo que o grafo resultante, F' , continue sendo uma floresta.

O grafo F' tem no máximo $|U|$ componentes e cada componente contém um vértice de U . Usando o Lema 4.3, podemos concluir que existe uma $\mathcal{R}$ -árvore $\hat{T}$ que contém F' e cujo comprimento é no máximo

$$(4.15) \quad l(EF') + \tau(U) \leq l(ET^*) - l(\delta_{T^*}(C')) + \pi(r, U) + \tau(U).$$

Agora, sabemos que $l(\delta_{T^*}(C')) \geq l(\delta_G(C', U)) \geq l(\delta_G(C, U))$. Segue que o comprimento de $\hat{T}$ é no máximo

$$(4.16) \quad l(ET^*) - l(\delta_G(C, U)) + \pi(r, U) + \tau(U) \leq l(ET^*),$$

e $\hat{T}$ também tem comprimento mínimo. Mas $\hat{T}$ contém um vértice em $R \setminus \{r\}$, o que encerra a demonstração. $\square$

Durante o teste de expansão, consideramos árvores T que são tais que $VT \cap \hat{\mathcal{R}} = \{r\}$. Podemos então usar o teste acima com $C := VT$ para cada uma dessas árvores. Se o teste obtiver sucesso, não há necessidade de continuar com a expansão: podemos encerrar o processo imediatamente e passar a considerar o vértice r como sendo um não-terminal.

4.4. Implementações dos testes e resultados

Implementamos os testes descritos para remoção de arestas e não-terminais, inclusive o teste de expansão de Polzin e Daneshmand [36], adaptado ao problema de Steiner com grupos. Implementamos também os testes de remoção de vértices de grupo.

Os testes de expansão podem consumir muito tempo caso sejam aplicados a vértices de grau alto. Aplicamos tais testes apenas em vértices de grau no máximo 8. Durante a expansão, também tomamos cuidado para que o teste não tome muito tempo. Controlamos a profundidade da recursão para que ela não cresça demais, limitando-a a um valor constante, como 15 ou 16. Também truncamos o teste se as folhas pelas quais tentamos expandir têm grau muito alto.

Outra questão importante para os testes de redução que apresentamos é como calcular limitantes superiores para as distâncias de gargalo das quais precisamos. Aqui optamos por

Instância	Original	Exp.	Adic.	Total	Instância	Original	Exp.	Adic.	Total
wrp4-11	41	1	1	2	wrp4-44	174	10	4	14
wrp4-13	41	9	0	9	wrp4-45	180	3	0	3
wrp4-14	52	1	3	4	wrp4-46	183	0	0	0
wrp4-15	57	3	1	4	wrp4-47	186	2	8	10
wrp4-16	47	6	2	8	wrp4-48	140	9	3	12
wrp4-17	64	8	12	20	wrp4-49	195	11	10	21
wrp4-18	53	5	1	6	wrp4-50	198	13	1	14
wrp4-19	51	3	1	4	wrp4-51	202	4	2	6
wrp4-21	107	7	10	17	wrp4-52	208	3	1	4
wrp4-22	86	3	0	3	wrp4-53	211	4	0	4
wrp4-23	91	0	0	0	wrp4-54	215	0	2	2
wrp4-24	120	8	19	27	wrp4-55	222	7	12	19
wrp4-25	88	5	3	8	wrp4-56	238	9	20	29
wrp4-26	104	4	2	6	wrp4-58	233	1	7	8
wrp4-27	107	0	0	0	wrp4-59	239	3	2	5
wrp4-28	110	0	0	0	wrp4-60	238	5	4	9
wrp4-29	114	0	0	0	wrp4-61	246	2	8	10
wrp4-30	118	3	0	3	wrp4-62	281	13	11	24
wrp4-31	124	2	1	3	wrp4-63	250	7	1	8
wrp4-32	130	4	1	5	wrp4-64	257	6	3	9
wrp4-33	97	4	1	5	wrp4-66	280	9	4	13
wrp4-34	136	0	0	0	wrp4-67	376	41	16	57
wrp4-35	139	0	0	0	wrp4-68	273	0	6	6
wrp4-36	143	0	0	0	wrp4-69	274	7	17	24
wrp4-37	148	0	0	0	wrp4-70	278	6	8	14
wrp4-38	150	0	0	0	wrp4-71	285	4	0	4
wrp4-39	200	31	24	55	wrp4-72	295	10	6	16
wrp4-40	158	0	5	5	wrp4-73	303	2	5	7
wrp4-41	164	0	0	0	wrp4-74	295	6	0	6
wrp4-42	168	0	0	0	wrp4-75	299	3	5	8
wrp4-43	170	1	15	16	wrp4-76	305	6	4	10

Tabela 4.5

usar as técnicas descritas no artigo de Rohe e Zachariasen [39]. Tais técnicas mostraram-se eficientes e forneceram bons limitantes.

Testamos nossos algoritmos de redução com o conjunto de testes WRP4 da Steinlib [34]. Tais instâncias vêm de problemas de roteamento em circuitos. Elas foram propostas por Rohe e Zachariasen [39], que as atacaram utilizando técnicas específicas para problemas retilineares.

Não tivemos acesso às instâncias originais e, portanto, testamos nossos algoritmos de redução sobre instâncias que já haviam passado por um pesado processo de redução (cf. Rohe e Zachariasen [39]).

Nesse contexto, os algoritmos para a remoção de arestas e não-terminais foram ineficazes. Já o algoritmo de expansão que propusemos para remoção de vértices de grupo obteve sucesso em remover uma parcela dos vértices de grupo das instâncias consideradas, não obstante o fato de essas instâncias terem sido previamente submetidas a um processo para a remoção de vértices de grupo que levava em conta o fato de serem instâncias de problemas retilineares (veja Rohe e Zachariasen [39]).

A Tabela 4.5 mostra os resultados que obtivemos. As colunas contêm, respectivamente: o nome da instância, o número de vértices de grupo da instância, a quantidade de vértices de grupo removida pelo algoritmo de expansão, a quantidade de vértices de grupo removida pelo teste adicional e o total de vértices de grupo removidos. Os testes foram feitos num

Pentium III 1.2GHz com 2 gigabytes de memória RAM. Os tempos de execução foram todos inferiores a 1 minuto.

No capítulo 6 vamos descrever um algoritmo *branch-and-cut* que desenvolvemos para o problema de Steiner com grupos e comentar o impacto que as reduções tiveram sobre o funcionamento de tal algoritmo. Mais detalhes sobre resultados obtidos podem ser encontrados em [14].

CAPÍTULO 5

Uma investigação poliédrica

No presente capítulo investigamos o problema de Steiner com grupos do ponto de vista poliédrico. Estudamos dois poliedros associados ao problema e propomos algumas formulações para ele. Investigamos também as relações entre as formulações propostas. Alguns dos resultados que apresentaremos apareceram previamente em [15].

5.1. Os poliedros

Seja $G = (V, E)$ um grafo e $\mathcal{R}$ uma coleção de grupos. O *poliedro do problema de Steiner com grupos* é o conjunto

$$(5.1) \quad P_{\text{STG}}(G, \mathcal{R}) := \text{fecho.conv} \{ \chi^{ET} \in \mathbb{R}^E \mid T \text{ uma } \mathcal{R}\text{-árvore em } G \}.$$

Em outras palavras, $P_{\text{STG}}(G, \mathcal{R})$ é o fecho convexo dos vetores característicos dos conjuntos de arestas das $\mathcal{R}$ -árvores em G ; seus vértices são vetores característicos de conjuntos de arestas de $\mathcal{R}$ -árvores. Quando não houver perigo de ambigüidade, escreveremos P_{STG} em vez de $P_{\text{STG}}(G, \mathcal{R})$.

Nosso estudo vai concentrar-se não em P_{STG} , mas sim em seu dominante, o poliedro

$$(5.2) \quad P_{\text{STG}}^{\dagger}(G, \mathcal{R}) := P_{\text{STG}}(G, \mathcal{R}) + \mathbb{R}_+^E.$$

Faremos isso pois $P_{\text{STG}}^{\dagger}$ possui propriedades que facilitarão nosso trabalho. Por exemplo, $P_{\text{STG}}^{\dagger}$ é do *tipo bloqueador*, ou seja, se $x \in P_{\text{STG}}^{\dagger}$ e $y \in \mathbb{R}_+^E$ então $x + y \in P_{\text{STG}}^{\dagger}$. Equivalentemente, podemos dizer que o cone característico de $P_{\text{STG}}^{\dagger}$ é formado por todos os vetores cujos componentes são todos não-negativos. Segue imediatamente que $P_{\text{STG}}^{\dagger}$ tem dimensão plena. O mesmo não pode ser dito, em geral, de P_{STG} .

Além disso, observe que, para funções comprimento não-negativas, otimizar sobre P_{STG} é o mesmo que otimizar sobre $P_{\text{STG}}^{\dagger}$. Em outras palavras, para $l: E \rightarrow \mathbb{R}_+$ temos $\min\{l^T x \mid x \in P_{\text{STG}}\} = \min\{l^T x \mid x \in P_{\text{STG}}^{\dagger}\}$.

5.1.1. A dimensão de P_{STG}

Determinar a dimensão de P_{STG} é um problema interessante e ilustrativo de algumas técnicas usadas quando lidamos com poliedros.

Seja $G = (V, E)$ um grafo e $\mathcal{R}$ uma coleção de grupos. Dizemos que um vértice $v \in V$ é $\mathcal{R}$ -essencial se não existe $\mathcal{R}$ -árvore em $G - v$. Observe que há dois tipos de vértices $\mathcal{R}$ -essenciais: aqueles que compõem grupos unitários e aqueles que são vértices de corte que

separam dois grupos distintos. Um componente 2-conexo de G é *totalmente $\mathcal{R}$ -essencial* se só contém vértices $\mathcal{R}$ -essenciais.

Teorema 5.1. *Seja $G = (V, E)$ um grafo conexo e $\mathcal{R}$ uma coleção de grupos. Sejam $V_1, \dots, V_k$ os conjuntos de vértices dos componentes 2-conexos de G que são totalmente $\mathcal{R}$ -essenciais. Então*

$$(5.3) \quad \text{fecho.afim } P_{\text{STG}} = \{x \in \mathbb{R}^E \mid x(E[V_i]) = |V_i| - 1, i = 1, \dots, k\}.$$

Prova: Seja T uma $\mathcal{R}$ -árvore em G . Como os conjuntos $V_1, \dots, V_k$ são totalmente $\mathcal{R}$ -essenciais, $V_1, \dots, V_k \subseteq VT$. Como $G[V_1], \dots, G[V_k]$ são componentes 2-conexos de G , $T[V_i]$ é uma árvore para $i = 1, \dots, k$. Mas então $\chi^{ET}(E[V_i]) = |V_i| - 1$, $i = 1, \dots, k$, e segue que todo $x \in P_{\text{STG}}$ satisfaz

$$(5.4) \quad x(E[V_i]) = |V_i| - 1, \text{ para } i = 1, \dots, k.$$

Segue, portanto, a inclusão $\subseteq$ em (5.3).

Para provar a inclusão reversa, suponha que todo $x \in P_{\text{STG}}$ satisfaça $a^\top x = \beta$, para algum vetor a e escalar β . Vamos provar que a equação $a^\top x = \beta$ é uma combinação linear das equações em (5.4). Para tanto, considere duas arestas $e, f \in E$. Afirmamos que:

- (5.5) (i) se e, f ocorrem juntas num circuito então $a(e) = a(f)$;
- (ii) se e pertence a um componente 2-conexo que não é totalmente $\mathcal{R}$ -essencial então $a(e) = 0$.

Para provar (5.5)(i), considere um circuito C que contém e e f e seja T uma árvore geradora de G que contém $C \setminus \{f\}$. Note que T é uma $\mathcal{R}$ -árvore, logo $a^\top \chi^{ET} = \beta$. Mas $T' := T - e + f$ também é uma $\mathcal{R}$ -árvore, logo $a^\top \chi^{ET'} = \beta = a^\top \chi^{ET}$, e segue que $a(e) = a(f)$.

Para ver (5.5)(ii), considere um vértice u do componente ao qual e pertence e suponha que u não seja $\mathcal{R}$ -essencial. Como e pertence ao componente, deve existir um vértice v adjacente a u no mesmo componente. Seja então T uma $\mathcal{R}$ -árvore em $G - u$. Como G é conexo, podemos estender T a uma $\mathcal{R}$ -árvore que contém u ou v , mas não ambos. (Basta notar que u não ocorre em T e adicionar arestas a T até que T contenha u ou v mas não ambos, o que é possível já que G é conexo.) Seja então $T' := T + uv$. Note que T' é uma $\mathcal{R}$ -árvore, logo $a^\top \chi^{ET'} = \beta = a^\top \chi^{ET}$. Segue que $a(uv) = 0$. Como uv está no mesmo componente 2-conexo de e , vem de (5.5)(i) que $a(e) = a(uv) = 0$.

De (5.5) podemos concluir que existem escalares $\lambda_1, \dots, \lambda_k$ tais que $a = \lambda_1 \chi^{E[V_1]} + \dots + \lambda_k \chi^{E[V_k]}$. Agora, se T é uma $\mathcal{R}$ -árvore, sabemos que $\chi^{ET}(E[V_i]) = |V_i| - 1$ para $i = 1, \dots, k$ e, portanto, $\beta = \lambda_1(|V_1| - 1) + \dots + \lambda_k(|V_k| - 1)$. Assim, $a^\top x = \beta$ é combinação linear das equações em (5.4), o que prova a inclusão reversa. $\square$

A dimensão de P_{STG} segue como corolário do resultado acima.

Corolário 5.2. *Seja $G = (V, E)$ um grafo conexo e $\mathcal{R}$ uma coleção de grupos. Seja k o número de componentes 2-conexos de G que são totalmente $\mathcal{R}$ -essenciais. A dimensão de P_{STG} é $|E| - k$.*

Prova: Sejam $V_1, \dots, V_k$ os conjuntos de vértices dos componentes 2-conexos totalmente $\mathcal{R}$ -essenciais de G . Note que o conjunto $\{\chi^{E[V_1]}, \dots, \chi^{E[V_k]}\}$ é linearmente independente. O resultado segue então do Teorema 5.1. $\square$

Uma consequência trivial é que, se G é 2-conexo e todo grupo em $\mathcal{R}$ possui pelo menos dois vértices, então P_{STG} tem dimensão plena.

5.1.2. Adjacência em P_{STG}

Apresentamos nesta seção condições suficientes para que dois vértices de P_{STG} sejam adjacentes (ou seja, estejam contidos numa face limitada de dimensão 1). Essas condições terão consequências para o diâmetro de P_{STG} . Novamente, considere um grafo $G = (V, E)$ e uma coleção de grupos $\mathcal{R}$.

Teorema 5.3. *Sejam T_1, T_2 $\mathcal{R}$ -árvores em G . Se $|ET_1 \triangle ET_2| = 1$ ou se $|ET_1 \triangle ET_2| = 2$ e $VT_1 = VT_2$, então os vértices χ^{ET_1}, χ^{ET_2} de P_{STG} são adjacentes.*

Prova: Suponha primeiro que $|ET_1 \triangle ET_2| = 1$. Sem perda de generalidade, suponha que exista uma aresta $e \in ET_2$ tal que $T_2 = T_1 + e$. Então χ^{ET_1}, χ^{ET_2} satisfazem com igualdade as seguintes $|E| - 1$ inequações linearmente independentes e válidas para P_{STG} :

$$(5.6) \quad \begin{aligned} x(f) &\leq 1, & \text{para } f \in ET_1 \cap ET_2, \\ x(f) &\geq 0, & \text{para } f \in E \setminus (ET_1 \cap ET_2), f \neq e, \end{aligned}$$

e, portanto, são vértices adjacentes de P_{STG} .

Suponha então que $|ET_1 \triangle ET_2| = 2$ e $VT_1 = VT_2$. Como $VT_1 = VT_2$, não podemos ter $ET_1 \subseteq ET_2$ ou vice-versa. Então, seja $\{e_1\} = ET_1 \setminus ET_2$ e $\{e_2\} = ET_2 \setminus ET_1$. Observe que, como $VT_1 = VT_2$ e como $T_2 = T_1 - e_1 + e_2$, $T_1 + e_2 = T_2 + e_1$ possuem um circuito C que contém e_1 e e_2 . Mas então χ^{ET_1}, χ^{ET_2} satisfazem com igualdade as seguintes $|E| - 1$ inequações linearmente independentes e válidas para P_{STG} :

$$(5.7) \quad \begin{aligned} x(f) &\leq 1, & \text{para } f \in ET_1 \cap ET_2, \\ x(f) &\geq 0, & \text{para } f \in E \setminus (ET_1 \cap ET_2), f \neq e_1, e_2, \\ x(EC) &\leq |EC| - 1, \end{aligned}$$

e, portanto, são vértices adjacentes de P_{STG} . $\square$

Para subgrafos G_1, G_2 de G , denote por $\text{dist}(G_1, G_2)$ o comprimento (número de arestas) de um menor caminho que liga um vértice de G_1 a um vértice de G_2 . (Se G é conexo, $\text{dist}(G_1, G_2)$ sempre existe.) Temos o seguinte resultado:

Teorema 5.4. *Suponha que G seja conexo e sejam H_1, H_2 $\mathcal{R}$ -árvores em G . Existe uma sequência $H_1 = T_1, \dots, T_k = H_2$ de $\mathcal{R}$ -árvores em G , $k \leq |EH_1 \triangle EH_2| + 2\text{dist}(H_1, H_2) + 1$, tal que, para $i = 1, \dots, k - 1$:*

$$(5.8) \quad \begin{aligned} &\text{(i) ou } |ET_i \triangle ET_{i+1}| = 1; \\ &\text{(ii) ou } |ET_i \triangle ET_{i+1}| = 2 \text{ e } VT_i = VT_{i+1}. \end{aligned}$$

Prova: Vamos provar a afirmação por indução em $p = p(H_1) := |EH_1 \triangle EH_2| + 2\text{dist}(H_1, H_2)$. Se $p = 0$, $H_1 = H_2$ e segue o resultado. Suponha então que $p > 0$. Temos três casos a considerar:

Caso 1. Suponha primeiro que $\text{dist}(H_1, H_2) > 0$ e seja u, v , $u \in VH_1$ e $v \in VH_2$, um par de vértices de H_1, H_2 mais próximos um do outro. Seja w o vizinho de u num $u-v$ caminho de comprimento mínimo em G . Note que $p(H_1 + uw) < p(H_1)$ e que $H_1 + uw$ é uma $\mathcal{R}$ -árvore. Mas então, por hipótese de indução, existe uma sequência de $\mathcal{R}$ -árvores $H_1 + uw = T_2, \dots, T_k = H_2$ com $k \leq p(H_1 + uw) + 2$, satisfazendo (5.8). Mas então a sequência $H_1 = T_1, T_2, \dots, T_k = H_2$ satisfaz as propriedades necessárias.

Caso 2. Suponha então que $\text{dist}(H_1, H_2) = 0$. Devemos ter $|EH_1 \triangle EH_2| > 0$. Se $EH_2 \subseteq EH_1$, deve existir um vértice v que é folha de H_1 mas não pertence a H_2 . Então $H_1 - v$ é uma $\mathcal{R}$ -árvore e $p(H_1 - v) < p(H_1)$. Como v é folha de H_1 , $|EH_1 \triangle E(H_1 - v)| = 1$, logo a afirmação do teorema segue pela hipótese de indução.

Caso 3. Finalmente, suponha que $\text{dist}(H_1, H_2) = 0$ e que $EH_2 \setminus EH_1 \neq \emptyset$. Como $\text{dist}(H_1, H_2) = 0$, H_1 e H_2 possuem vértices em comum. Podemos então escolher uma aresta $e \in EH_2 \setminus EH_1$ que incide em algum vértice de H_1 .

Se $H_1 + e$ é uma árvore, então $H_1 + e$ é uma $\mathcal{R}$ -árvore e $p(H_1 + e) < p(H_1)$. O resultado segue então da hipótese de indução. Suponha então que $H_1 + e$ possua um circuito C . Como H_2 é uma árvore, C deve conter alguma aresta $f \in EH_1 \setminus EH_2$. Escreva $H'_1 := H_1 + e - f$. Observe que $p(H'_1) < p(H_1)$ e que H_1 e H'_1 satisfazem (5.8)(ii). O teorema segue então da aplicação da hipótese de indução. $\square$

Observe que, se G é conexo, dadas duas $\mathcal{R}$ -árvores quaisquer H_1, H_2 em G , temos

$$(5.9) \quad |EH_1 \triangle EH_2| + 2\text{dist}(H_1, H_2) \leq 2(|V| - 1).$$

É fácil ver isso se $\text{dist}(H_1, H_2) = 0$. Caso contrário, observe que H_1, H_2 são disjuntos nos vértices. Note que o menor caminho entre um vértice de H_1 e outro de H_2 não possui vértices internos em H_1, H_2 . Mas então, se $n_1 := |VH_1|$ e $n_2 := |VH_2|$, temos que

$$(5.10) \quad |EH_1 \triangle EH_2| + 2\text{dist}(H_1, H_2) \leq (n_1 - 1) + (n_2 - 1) + 2(|V| - (n_1 - 1) - (n_2 - 1) - 1),$$

sendo o lado direito no máximo $2(|V| - 1)$, como queríamos. Observe também que, segundo o Teorema 5.3, os vértices de P_{STG} correspondentes a árvores consecutivas na sequência dada no enunciado do Teorema 5.4 são adjacentes em P_{STG} . Em outras palavras:

Corolário 5.5. *Se G é conexo, o diâmetro de $P_{\text{STG}}(G, \mathcal{R})$ é no máximo $2(|V| - 1)$.* $\square$

5.2. Uma primeira formulação para $P_{\text{STG}}^\dagger$

Seja $G = (V, E)$ um grafo e $\mathcal{R}$ uma coleção de grupos. Dizemos que um conjunto $U \subseteq V$ é $\mathcal{R}$ -separador se existem grupos R, S tais que $R \subseteq U$ e $S \subseteq V \setminus U$. Um corte $\delta(U)$ é um $\mathcal{R}$ -corte se U é $\mathcal{R}$ -separador. Obviamente, todo $x \in P_{\text{STG}}^\dagger$ é tal que

$$(5.11) \quad x(\delta(U)) \geq 1, \quad \text{para todo } U \subseteq V \text{ que é } \mathcal{R}\text{-separador}.$$

Figura 5.1: (a) Uma instância do problema de Steiner com grupos com três grupos: A , B e C . O vetor inteiro, indexado pelas arestas do grafo, cujos componentes têm o valor que acompanha cada aresta, satisfaz todas as inequações de $\mathcal{R}$ -cortes, mas não pertence a P_{STG}^I . (b) Uma $\mathcal{R}$ -partição no grafo em (a); as partes estão sombreadas. Note que o vetor inteiro descrito anteriormente não satisfaz a inequação correspondente a essa $\mathcal{R}$ -partição.

Essas inequações, entretanto, não fornecem uma formulação para P_{STG}^I . Mais especificamente, existem vetores inteiros que satisfazem todas as inequações em (5.11) mas que não pertencem a P_{STG}^I , como mostra a Figura 5.1a.

Uma partição $\mathcal{U} = \{V_1, \dots, V_k\}$ de V é uma $\mathcal{R}$ -partição se, para cada $i = 1, \dots, k$, existe um grupo R_i tal que $R_i \cap V_i = \emptyset$. Em outras palavras, nenhum dos V_i cobre $\mathcal{R}$. Note que se $U \subseteq V$ é $\mathcal{R}$ -separador então $\{U, V \setminus U\}$ é uma $\mathcal{R}$ -partição. Se $\mathcal{U}$ é uma $\mathcal{R}$ -partição, denotaremos por $\Delta(\mathcal{U})$ o conjunto das arestas de G com pontas em partes diferentes de $\mathcal{U}$; chamaremos um tal conjunto $\Delta(\mathcal{U})$ de $\mathcal{R}$ -multicorte. Obviamente, todo $x \in P_{\text{STG}}^I$ é tal que

$$(5.12) \quad x(\Delta(\mathcal{U})) \geq 1, \quad \text{para cada } \mathcal{R}\text{-partição } \mathcal{U}.$$

Na verdade, essas inequações fornecem uma formulação para P_{STG}^I . Mais especificamente, denote por $P_{\text{part}}(G, \mathcal{R})$ o poliedro que consiste nos vetores $x \in \mathbb{R}_+^E$ que satisfazem as inequações em (5.12). Então $(P_{\text{part}})_I = P_{\text{STG}}^I$. Para ver isso, note primeiro que a inclusão $P_{\text{STG}}^I \subseteq (P_{\text{part}})_I$ é trivial, de modo que precisamos apenas provar a inclusão reversa. Como P_{STG}^I e P_{part} são ambos do tipo bloqueador, esse fato decorre trivialmente da seguinte observação:

$$(5.13) \quad \text{se } x \in P_{\text{part}} \text{ então existe uma } \mathcal{R}\text{-árvore } T \text{ tal que } ET \subseteq \text{supp } x.$$

Para ver isso, basta observar que, caso contrário, nenhum componente de $(V, \text{supp } x)$ cobre $\mathcal{R}$. Nesse caso, se $V_1, \dots, V_k$ são os conjuntos de vértices dos componentes de $(V, \text{supp } x)$ então $\mathcal{U} = \{V_1, \dots, V_k\}$ é uma $\mathcal{R}$ -partição e $x(\Delta(\mathcal{U})) = 0$, uma contradição. A Figura 5.1b mostra um exemplo de $\mathcal{R}$ -partição.

5.2.1. O problema da separação

O problema da separação para as inequações de $\mathcal{R}$ -cortes pode ser facilmente reduzido ao problema de se encontrar uma série de cortes mínimos em grafos derivados de G , podendo

portanto ser resolvido em tempo polinomial. De fato, suponha que tenhamos um grafo $G = (V, E)$ e uma coleção de grupos $\mathcal{R}$. Suponha também que nos seja dado um vetor $x \in \mathbb{R}_+^E$ e que queiramos verificar se ele satisfaz todas as inequações de $\mathcal{R}$ -cortes. Fixe dois grupos $R, S \in \mathcal{R}$. É fácil ver que x satisfaz $x(\delta(U)) \geq 1$ para todo U tal que $R \subseteq U$ e $S \subseteq V \setminus U$ se e somente se um R - S corte de capacidade mínima em G , sendo as capacidades dadas por x , tem capacidade pelo menos 1.

Dessa forma, podemos encontrar uma inequação de $\mathcal{R}$ -corte mais violada por x repetindo o raciocínio acima para todo par de grupos em $\mathcal{R}$. Observe também que, dados grupos R, S , podemos encontrar um R - S corte de capacidade mínima em G em tempo polinomial: basta adicionar vértices r, s a G , conectando r a todos os vértices de R por arestas de capacidade infinita e s a todos os vértices de S por arestas de capacidade infinita. O problema resume-se então ao problema de se encontrar um r - s corte de capacidade mínima no grafo modificado, o que pode ser feito em tempo polinomial.

Já o problema da separação para as inequações de $\mathcal{R}$ -partições é, em geral, NP-difícil. Esse fato é uma consequência imediata do seguinte teorema:

Teorema 5.6. *O problema:*

$$(5.14) \quad \text{dados um grafo } G = (V, E), \text{ uma coleção de grupos } \mathcal{R}, \text{ uma função peso } w: E \rightarrow \mathbb{Q}_+ \text{ e um número } \beta \in \mathbb{Q}, \beta > 0, \text{ existe um } \mathcal{R}\text{-multicorte } \Delta(\mathcal{U}) \text{ com } w(\Delta(\mathcal{U})) < \beta?$$

é NP-completo mesmo quando temos exatamente três grupos dois-a-dois disjuntos, cada um com exatamente dois vértices.

Prova: Seja $G = (V, E)$ um grafo e $Z \subseteq V$. Um conjunto de arestas $F \subseteq E$ é um Z -multicorte se cada componente de $G - F$ possui no máximo um vértice de Z . Considere o problema:

$$(5.15) \quad \text{dados um grafo } G = (V, E), \text{ um conjunto } Z \subseteq V, \text{ uma função peso } w: E \rightarrow \mathbb{Q}_+ \text{ e um número } \beta \in \mathbb{Q}, \beta > 0, \text{ existe um } Z\text{-multicorte } F \subseteq E \text{ tal que } w(F) < \beta?$$

O problema (5.15) é a versão de decisão do problema do corte multiterminal de peso mínimo. Dahlhaus *et al.* [5] provaram que (5.15) é NP-completo mesmo quando $|Z| = 3$.

Dados G, Z, w e β como em (5.15), considere a coleção de grupos $\mathcal{R} := \{Z \setminus \{v\} \mid v \in Z\}$. Note que nenhum vértice em Z cobre $\mathcal{R}$, mas que quaisquer dois vértices em Z cobrem $\mathcal{R}$. Assim, temos que

$$(5.16) \quad \text{existe um } Z\text{-multicorte } F \subseteq E \text{ tal que } w(F) < \beta \text{ se e somente se existe um } \mathcal{R}\text{-multicorte } \Delta(\mathcal{U}) \text{ tal que } w(\Delta(\mathcal{U})) < \beta.$$

Nossa redução só não está completa pois os grupos em $\mathcal{R}$ têm intersecções. Para contornar esse problema, escreva $W := w(E)$ e aplique o seguinte algoritmo a G, w e $\mathcal{R}$:

$$(5.17) \quad \begin{array}{ll} \text{(i)} & \text{se os grupos são dois-a-dois disjuntos, pare;} \\ \text{(ii)} & \text{sejam } R, S \text{ dois grupos tais que } R \cap S \neq \emptyset. \text{ Para cada } v \in R \cap S, \text{ insira em } G \\ & \text{vértices } v_R \text{ e } v_S \text{ ligados a } v \text{ por arestas de peso } W + 1; \text{ remova } v \text{ de } R \text{ e } S \text{ e} \\ & \text{adicione } v_R \text{ a } R \text{ e } v_S \text{ a } S. \text{ Volte ao passo (i).} \end{array}$$

Sejam G' , w' e $\mathcal{R}'$ o grafo, a função peso e a coleção de grupos resultantes da aplicação do algoritmo, respectivamente. Durante a execução do algoritmo, nunca criamos intersecções entre os grupos e, a cada execução do passo (ii), um par de grupos deixa de ter intersecção. Logo, os grupos em $\mathcal{R}'$ são dois-a-dois disjuntos. Além disso, é imediato que:

$$(5.18) \quad \text{para todo } \beta \in \mathbb{Q}, \beta > 0, \text{ existe um } \mathcal{R}\text{-multicorte } \Delta(\mathcal{U}) \text{ em } G \text{ tal que } w(\Delta(\mathcal{U})) < \beta \\ \text{se e somente se existe um } \mathcal{R}'\text{-multicorte } \Delta(\mathcal{U}') \text{ em } G' \text{ tal que } w'(\Delta(\mathcal{U}')) < \beta.$$

Note também que o tamanho dos grupos em $\mathcal{R}'$ é igual ao tamanho dos grupos em $\mathcal{R}$ e que $|\mathcal{R}| = |\mathcal{R}'|$. Com isso, como o problema (5.15) é NP-completo mesmo quando $|Z| = 3$ e como, por construção, $\mathcal{R}$ possui exatamente $|Z|$ grupos e cada um deles contém exatamente $|Z| - 1$ vértices, o teorema está provado. $\square$

Como corolário, temos:

Corolário 5.7. *O problema:*

$$(5.19) \quad \text{dados um grafo } G = (V, E), \text{ uma coleção de grupos } \mathcal{R} \text{ e um vetor } x \in \mathbb{Q}_+^E, \text{ existe} \\ \text{uma } \mathcal{R}\text{-partição } \mathcal{U} \text{ tal que } x(\Delta(\mathcal{U})) < 1?$$

é NP-completo mesmo se $\mathcal{R}$ possui exatamente três grupos dois-a-dois disjuntos cada um contendo exatamente dois vértices.

Prova: Vamos reduzir o problema (5.14) ao problema (5.19). Dados G , $\mathcal{R}$, w e β como em (5.14), ponha $x := \beta^{-1}w$. É fácil ver que $\mathcal{U}$ é uma $\mathcal{R}$ -partição tal que $x(\Delta(\mathcal{U})) < 1$ se e somente se $\Delta(\mathcal{U})$ é um $\mathcal{R}$ -multicorte tal que $w(\Delta(\mathcal{U})) < \beta$, o que prova o resultado. $\square$

Segue imediatamente que (cf. Grötschel, Lovász e Schrijver [23]):

Corolário 5.8. *Os problemas de separação e de otimização para P_{part} são NP-difíceis, mesmo quando temos exatamente três grupos dois-a-dois disjuntos, cada um contendo exatamente dois vértices.* $\square$

Há casos, entretanto, em que a separação das inequações de $\mathcal{R}$ -partições pode ser feita em tempo polinomial. Se um dos grupos é unitário, por exemplo, a separação das inequações de $\mathcal{R}$ -partições resume-se à separação das inequações de $\mathcal{R}$ -cortes; o mesmo ocorre se temos apenas dois grupos. Como veremos na seção seguinte, o problema da separação também pode ser resolvido em tempo polinomial se o grafo G for uma árvore.

5.2.2. A separação em árvores

Como mencionamos na seção anterior, o problema da separação para $P_{\text{part}}(G, \mathcal{R})$ pode ser resolvido em tempo polinomial quando G é uma árvore. Nesta seção vamos descrever um algoritmo que resolve o seguinte problema:

$$(5.20) \quad \text{dados: uma árvore } T = (V, E), \text{ uma coleção } \mathcal{R} \text{ de grupos de vértices de } T \text{ e uma} \\ \text{função peso } w: E \rightarrow \mathbb{R}_+, \\ \text{encontre: um } \mathcal{R}\text{-multicorte } \Delta(\mathcal{U}) \text{ de peso mínimo.}$$

Note que o problema (5.14) é a versão de decisão do problema (5.20). Um algoritmo que resolve (5.20) permite-nos encontrar uma inequação mais violada no sistema que define P_{part} .

Considere então uma árvore $T = (V, E)$, uma coleção de grupos $\mathcal{R}$ e uma função peso $p: E \rightarrow \mathbb{R}_+$. Seja s um vértice arbitrário de T . Daqui em diante, vamos ver T como uma árvore enraizada em s . Para cada $v \in V$, denote por T_v a subárvore de T enraizada em v .

Seja $F \subseteq E$ um conjunto de peso mínimo tal que não existe $\mathcal{R}$ -árvore em $T - F$. Note que esse conjunto nos fornece um $\mathcal{R}$ -multicorte de peso mínimo: basta considerar a $\mathcal{R}$ -partição $\mathcal{U} = \{V_1, \dots, V_k\}$, sendo $V_1, \dots, V_k$ os conjuntos de vértices dos componentes de $T - F$, e notar que $\Delta(\mathcal{U}) \subseteq F$.

Fixe agora um vértice $v \in V$. Denote por C_v o conjunto dos filhos de v . O conjunto F contém um subconjunto $F_v \subseteq ET_v$ tal que $T_v - F_v$ não contém nenhuma $\mathcal{R}$ -árvore. Deve existir um grupo $R \in \mathcal{R}$ tal que o componente de $T_v - F_v$ que contém v não contém nenhum vértice de R . Mas então, para cada $w \in C_v$, temos duas possibilidades:

- (5.21) (i) se $w \in R$, devemos ter $vw \in F_v$. Além disso, F_v deve conter um conjunto $F_w \subseteq ET_w$, de peso mínimo, tal que $T_w - F_w$ não contém nenhuma $\mathcal{R}$ -árvore;
(ii) se $w \notin R$ então podemos ou não ter $vw \in F_v$. Se temos $vw \in F_v$, F_v deve conter um conjunto F_w como acima. Senão, F_v deve conter um conjunto $F_w \subseteq ET_w$, de peso mínimo, tal que $T_w - F_w$ não contém nenhuma $\mathcal{R}$ -árvore e, além disso, tal que o componente de $T_w - F_w$ que contém w não contém nenhum vértice de R .

Com as observações acima, podemos derivar uma recorrência que nos fornecerá o peso do conjunto F . Para tanto, para cada $v \in V$ e $R \in \mathcal{R}$, denote por $b(v, R)$ o peso de um conjunto $F_v \subseteq ET_v$ tal que

- (i) $T_v - F_v$ não contém nenhuma $\mathcal{R}$ -árvore;
(ii) o componente de $T_v - F_v$ que contém v não contém nenhum vértice de R ;

e tem peso mínimo. Para $v \in V$, escreva $b(v) := \min\{b(v, R) \mid R \in \mathcal{R}\}$. Observe que o peso do conjunto F é exatamente $b(s)$. Note que, se v é uma folha de T , então

$$(5.22) \quad b(v, R) = \begin{cases} 0, & \text{se } v \notin R, \\ \infty, & \text{caso contrário.} \end{cases}$$

Suponha então que v não seja folha de T . Se $v \in R$, $b(v, R) = \infty$. Caso contrário, segue de (5.21) que

$$(5.23) \quad b(v, R) = \sum_{w \in C_v \cap R} (p(vw) + b(w)) + \sum_{w \in C_v \setminus R} \min\{p(vw) + b(w), b(w, R)\}.$$

Agora, é fácil ver que podemos computar $b(v, R)$ de baixo para cima na árvore T . Para tanto, basta usar um esquema simples de programação dinâmica. Mais especificamente, uma tabela que contém para cada $v \in V$ e $R \in \mathcal{R}$ uma entrada para $b(v, R)$. Se armazenarmos $b(v)$ à medida em que computamos $b(v, R)$ para cada vértice v , podemos computar $b(v, R)$ em tempo $O(|C_v|)$, sendo C_v o conjunto dos filhos de v . Com isso, o tempo consumido pelo algoritmo é

$$(5.24) \quad \sum_{v \in V} \sum_{R \in \mathcal{R}} O(|C_v|) = O\left(\sum_{R \in \mathcal{R}} \sum_{v \in V} |C_v|\right) = O(|\mathcal{R}|(n-1)) = O(|\mathcal{R}|n).$$

Obviamente, podemos alterar o algoritmo de forma a que encontre não só o peso do conjunto, mas também o conjunto em si, mantendo a mesma complexidade de tempo.

5.3. As facetas de $P_{\text{STG}}^\dagger$

Faremos agora um estudo das facetas de $P_{\text{STG}}^\dagger$. Novamente, seja $G = (V, E)$ um grafo e $\mathcal{R}$ uma coleção de grupos. Uma primeira observação simples é que a inequação

$$(5.25) \quad x(e) \geq 0, e \in E,$$

induz uma faceta de $P_{\text{STG}}^\dagger$ se e somente se $G - e$ contém uma $\mathcal{R}$ -árvore. A prova desse fato é bem simples. Primeiramente, observe que se $G - e$ não contém nenhuma $\mathcal{R}$ -árvore então toda $\mathcal{R}$ -árvore T é tal que $e \in ET$. Segue que todo $x \in P_{\text{STG}}^\dagger$ é tal que $x(e) \geq 1$, logo (5.25) não induz uma faceta de $P_{\text{STG}}^\dagger$.

Suponha, por outro lado, que $G - e$ contenha uma $\mathcal{R}$ -árvore T . Então os $|E| - 1$ vetores $\chi^{ET} + \chi^f$, $f \in E \setminus \{e\}$, juntamente com o vetor χ^{ET} , pertencem a $P_{\text{STG}}^\dagger$, são afim-independentes e satisfazem (5.25) com igualdade. Logo (5.25) induz uma faceta de $P_{\text{STG}}^\dagger$.

O caso das $\mathcal{R}$ -partições é um pouco mais complicado. No teorema seguinte, caracterizamos quando as inequações de $\mathcal{R}$ -partições induzem facetas de $P_{\text{STG}}^\dagger$.

Teorema 5.9. *Seja $\mathcal{U} = \{V_1, \dots, V_k\}$ uma $\mathcal{R}$ -partição. A inequação*

$$(5.26) \quad x(\Delta(\mathcal{U})) \geq 1$$

induz uma faceta de $P_{\text{STG}}^\dagger$ se e somente se $\Delta(\mathcal{U})$ é um $\mathcal{R}$ -multicorte minimal.

Prova: Para ver a necessidade, suponha que $\Delta(\mathcal{U})$ não seja um $\mathcal{R}$ -multicorte minimal. Então existe um $\mathcal{R}$ -multicorte $\Delta(\mathcal{U}')$ tal que $\Delta(\mathcal{U}')$ é um subconjunto próprio de $\Delta(\mathcal{U})$. Mas então (5.26) é soma das inequações válidas não-triviais $x(\Delta(\mathcal{U}')) \geq 1$ e $x(\Delta(\mathcal{U}) \setminus \Delta(\mathcal{U}')) \geq 0$, não induzindo, portanto, uma faceta de $P_{\text{STG}}^\dagger$.

Para provar a suficiência, suponha que a inequação $a^\top x \geq \beta$ induza uma faceta de $P_{\text{STG}}^\dagger$ e seja tal que

$$(5.27) \quad \{x \in P_{\text{STG}}^\dagger \mid x(\Delta(\mathcal{U})) = 1\} \subseteq \{x \in P_{\text{STG}}^\dagger \mid a^\top x = \beta\}.$$

Afirmamos que $a^\top x \geq \beta$ é um múltiplo não-negativo de (5.26). Para ver isso, observe primeiro que, como $\Delta(\mathcal{U})$ é minimal,

$$(5.28) \quad \text{toda aresta } uv \in \Delta(\mathcal{U}), u \in V_i \text{ e } v \in V_j, \text{ conecta componentes de } G[V_i] \text{ e } G[V_j] \text{ que juntos cobrem } \mathcal{R}.$$

Escreva $F := E[V_1] \cup \dots \cup E[V_k]$. Considere primeiro uma aresta $e \in F$. Afirmamos que $a(e) = 0$. Para ver isso, seja f uma aresta qualquer em $\Delta(\mathcal{U})$. Devido a (5.28), $(V, F \cup \{f\})$ contém uma $\mathcal{R}$ -árvore. Por isso, e como $P_{\text{STG}}^\dagger$ é do tipo bloqueador, temos que $\chi^{F \cup \{f\}} \in P_{\text{STG}}^\dagger$ e $\chi^{F \cup \{f\}} + \chi^e \in P_{\text{STG}}^\dagger$. Mais ainda, ambos esses vetores satisfazem (5.26) com igualdade, logo $a^\top \chi^{F \cup \{f\}} = \beta = a^\top (\chi^{F \cup \{f\}} + \chi^e)$, e segue que $a(e) = 0$.

Considere agora duas arestas $e, f \in \Delta(\mathcal{U})$. Afirmamos que $a(e) = a(f)$. Para ver isso, basta notar que, devido a (5.28) e ao fato de $P_{\text{STG}}^\dagger$ ser do tipo bloqueador, ambos $\chi^{F \cup \{e\}}$ e

$\chi^{F \cup \{f\}}$ pertencem a P_{STG}^1 . Ademais, ambos satisfazem (5.26) com igualdade, logo $a^\top \chi^{F \cup \{e\}} = \beta = a^\top \chi^{F \cup \{f\}}$, e segue que $a(e) = a(f)$.

Com isso, fica fácil ver que $a^\top x \geq \beta$ é um múltiplo não-negativo de (5.26) e que, portanto, induz uma faceta de P_{STG}^1 . $\square$

Uma outra classe interessante de facetas é uma adaptação das inequações de partição de Steiner, de Grötschel e Monma [24]. Uma partição $\mathcal{U} = \{V_1, \dots, V_k\}$ de V é uma *partição de grupos* se, para cada $i = 1, \dots, k$, existe um grupo R_i tal que $R_i \subseteq V_i$. Obviamente, todo $x \in P_{\text{STG}}^1$ é tal que

$$(5.29) \quad x(\Delta(\mathcal{U})) \geq k - 1, \quad \text{para toda partição de grupos } \mathcal{U}.$$

Abaixo fornecemos condições suficientes para que as inequações de partições de grupos induzam facetas de P_{STG}^1 .

Teorema 5.10. *Seja $\mathcal{U} = \{V_1, \dots, V_k\}$ uma partição de grupos. Se*

- (5.30) (i) $G[V_i]$ é conexo para cada $i = 1, \dots, k$;
(ii) o grafo $\hat{G}$, obtido de G contraindo-se cada V_i a um vértice v_i , é 2-conexo;

então a inequação

$$(5.31) \quad x(\Delta(\mathcal{U})) \geq k - 1$$

induz uma faceta de P_{STG}^1 .

Prova: Suponha que a inequação $a^\top x \geq \beta$ induza uma faceta de P_{STG}^1 e seja tal que

$$(5.32) \quad \{x \in P_{\text{STG}}^1 \mid x(\Delta(\mathcal{U})) = k - 1\} \subseteq \{x \in P_{\text{STG}}^1 \mid a^\top x = \beta\}.$$

Vamos provar que $a^\top x \geq \beta$ é um múltiplo não-negativo de (5.31). Para tanto, escreva $F := E[V_1] \cup \dots \cup E[V_k]$. Afirmamos que:

- (5.33) (i) $a(e) = 0$ para todo $e \in F$;
(ii) $a(e) = a(f)$ para todo $e, f \in \Delta(\mathcal{U})$.

Para ver (5.33)(i), considere uma aresta $e \in F$. Seja $\hat{T}$ uma árvore geradora em $\hat{G}$. De (5.30)(i) segue que $(V, F \cup E\hat{T})$ contém uma $\mathcal{R}$ -árvore, logo $\chi^{F \cup E\hat{T}} \in P_{\text{STG}}^1$. Como P_{STG}^1 é do tipo bloqueador, $\chi^{F \cup E\hat{T}} + \chi^e \in P_{\text{STG}}^1$. Note que ambos esses vetores satisfazem (5.31) com igualdade, logo $a^\top \chi^{F \cup E\hat{T}} = \beta = a^\top (\chi^{F \cup E\hat{T}} + \chi^e)$, e segue que $a(e) = 0$.

Para ver (5.33)(ii), considere duas arestas $e, f \in \Delta(\mathcal{U})$. Como $\hat{G}$ é 2-conexo, existe um circuito C em $\hat{G}$ que contém e e f . Podemos então construir uma árvore geradora $\hat{T}$ de $\hat{G}$ que contém $C \setminus \{f\}$. Considere o grafo $H = (V, F \cup E\hat{T})$. Por (5.30)(i) temos que H contém uma $\mathcal{R}$ -árvore. Devido à forma como construímos H , $H' := H - e + f$ também contém uma $\mathcal{R}$ -árvore. Com isso e como P_{STG}^1 é do tipo bloqueador, temos que os vetores $\chi^{EH}, \chi^{EH'}$ pertencem a P_{STG}^1 . Além disso, ambos satisfazem (5.31) com igualdade. Mas então $a^\top \chi^{EH} = \beta = a^\top \chi^{EH'}$, e segue que $a(e) = a(f)$.

Usando (5.33) fica fácil ver que $a^\top x \geq \beta$ é um múltiplo não-negativo de (5.31) e, portanto, que essa inequação induz uma faceta de P_{STG}^1 . $\square$

Como mostra a Figura 5.2, as partições de grupo são mais fortes que as $\mathcal{R}$ -partições.

Finalmente, mencionamos que Grötschel, Monma e Stoer [25] provaram que o problema da separação para as inequações de partição de Steiner é NP-difícil. Isso implica que o problema da separação para as inequações de partição de grupos também é NP-difícil.

Figura 5.2: Nessa instância do problema de Steiner com grupos todos os grupos são unitários. Os vértices de grupo são aqueles pintados de preto. Se $G = (V, E)$ é o grafo da figura, o vetor $x \in \mathbb{R}^E$ tal que $x(e) = 1/2$ para todo $e \in E$ satisfaz todas as inequações de $\mathcal{R}$ -partições, mas não satisfaz a inequação de partição de grupos cujas partes estão destacadas em cinza. Em outras palavras, se $\mathcal{U}$ é a partição de grupos destacada na figura, temos $x(\Delta(\mathcal{U})) = 3/2 < 2$.

5.3.1. Contrações de arestas e facetas de P_{STG}^1

Nesta seção apresentamos brevemente uma técnica que nos permitirá encontrar facetas de P_{STG}^1 a partir de facetas de poliedros associados a menores por contração de G .

Seja $G = (V, E)$ um grafo e $\mathcal{R}$ uma coleção de grupos. Considere uma aresta $e = uv$ de G e seja w a imagem de u (que é igual à imagem de v) em G/e . Para $R \in \mathcal{R}$ escreva

$$(5.34) \quad R/e := \begin{cases} R, & \text{se } R \cap \{u, v\} = \emptyset, \\ (R \setminus \{u, v\}) \cup \{w\}, & \text{caso contrário.} \end{cases}$$

Escreva também $\mathcal{R}/e := \{R/e \mid R \in \mathcal{R}\}$.

Teorema 5.11. *Seja e uma aresta de G e suponha que a inequação $a^\top x \geq \beta$ induza uma faceta de $P_{\text{STG}}^1(G/e, \mathcal{R}/e)$. Considere o vetor $\bar{a} \in \mathbb{R}^E$ tal que $\bar{a}(f) = a(f)$ para $f \neq e$ e $\bar{a}(e) = 0$. Então a inequação $\bar{a}^\top x \geq \beta$ induz uma faceta de $P_{\text{STG}}^1(G, \mathcal{R})$.*

Prova: Observe primeiro que $\bar{a}^\top x \geq \beta$ é válida para $P_{\text{STG}}^1(G, \mathcal{R})$. Como $a^\top x \geq \beta$ induz uma faceta de $P_{\text{STG}}^1(G/e, \mathcal{R}/e)$, existem $m - 1$, $m := |E|$, vetores afim-independentes $x_1, \dots, x_{m-1}$ que pertencem a $P_{\text{STG}}^1(G/e, \mathcal{R}/e)$ e satisfazem $a^\top x \geq \beta$ com igualdade.

Denote por $x'_1, \dots, x'_{m-1}$ os vetores em $\mathbb{R}^E$ tais que $x'_i(f) = x_i(f)$ para $f \neq e$ e $x'_i(e) = 0$. Afirmamos que os vetores $x'_i + \chi^e$, $i = 1, \dots, m - 1$, pertencem a $P_{\text{STG}}^1(G, \mathcal{R})$. Para ver isso, considere o vetor x'_i . Como $x_i \in P_{\text{STG}}^1(G/e, \mathcal{R}/e)$, existem $(\mathcal{R}/e)$ -árvores $T_1, \dots, T_k$, números reais não-negativos $\lambda_1, \dots, \lambda_k$ tais que $\lambda_1 + \dots + \lambda_k = 1$ e um vetor $y \in \mathbb{R}_+^{E \setminus \{e\}}$ tais que

$$(5.35) \quad x_i = \lambda_1 \chi^{ET_1} + \dots + \lambda_k \chi^{ET_k} + y.$$

Como $T_1, \dots, T_k$ são $(\mathcal{R}/e)$ -árvores, os subgrafos $(V, ET_i \cup \{e\})$ de G , para $i = 1, \dots, k$, contêm $\mathcal{R}$ -árvores. Portanto, os vetores $\chi^{ET_i} + \chi^e \in \mathbb{R}^E$, $i = 1, \dots, k$, pertencem a $P_{\text{STG}}^1(G, \mathcal{R})$. Denote por y' o vetor em $\mathbb{R}^E$ que é tal que $y'(f) = y(f)$ para $f \neq e$ e $y'(e) = 0$. Temos que

$$(5.36) \quad x'_i + \chi^e = \lambda_1 (\chi^{ET_1} + \chi^e) + \dots + \lambda_k (\chi^{ET_k} + \chi^e) + y',$$

Figura 5.3: (a) Uma instância do problema de Steiner com grupos com quatro grupos: A , B , C e D . Após a contração das arestas dentro das áreas sombreadas, obtemos o grafo e os grupos em (b). Note que em (b) todos os grupos são unitários, ou seja, temos uma instância do problema de Steiner em grafos.

e $x'_i + \chi^e$ pertence a $P_{\text{STG}}^1(G, \mathcal{R})$, o que prova a afirmação.

Com isso, é fácil ver que $x'_1 + \chi^e, \dots, x'_{m-1} + \chi^e, x'_1 + 2\chi^e$ são m vetores afim-independentes em $P_{\text{STG}}^1(G, \mathcal{R})$ que satisfazem $\bar{a}^T x \geq \beta$ com igualdade. Portanto, $\bar{a}^T x \geq \beta$ induz uma faceta de $P_{\text{STG}}^1(G, \mathcal{R})$. $\square$

Usando esse resultado simples podemos utilizar facetas do bem-estudado dominante do poliedro do problema de Steiner em grafos para o dominante do poliedro do problema de Steiner com grupos. Se, após um determinado número de contrações, todos os grupos forem unitários, temos uma instância do problema de Steiner em grafos. Qualquer faceta do dominante do poliedro associado a essa instância do problema se transforma numa faceta do poliedro associado ao problema original. A Figura 5.3 ilustra essa idéia.

De forma análoga, podemos obter facetas para P_{STG}^1 a partir de facetas do dominante do poliedro do problema da cobertura por conjuntos. De fato, se G é uma árvore e r é um vértice não-terminal de G então podemos contrair arestas de G até obter uma estrela de centro r . Essa estrela, juntamente com a coleção de grupos resultante das contrações, representa uma instância do problema da cobertura por conjuntos (cf. seção 2.3). Nesse caso, P_{STG}^1 corresponde ao dominante do poliedro do problema da cobertura por conjuntos e qualquer faceta desse poliedro corresponde a uma faceta de $P_{\text{STG}}^1(G, \mathcal{R})$.

5.4. Algumas formulações estendidas para P_{STG}^1

Como vimos na seção 5.2, o problema de otimização sobre P_{part} é NP-difícil. Isso torna a formulação apresentada naquela seção inconveniente para o uso num algoritmo que segue a estratégia *branch-and-cut*, por exemplo.

Por isso, apresentamos agora algumas formulações alternativas para o problema. Tais formulações são estendidas no sentido em que usam variáveis “artificiais”.

5.4.1. Variáveis nos vértices

Seja $G = (V, E)$ um grafo e $\mathcal{R}$ uma coleção de grupos. Fixe um grupo $R_0 \in \mathcal{R}$. Chamaremos R_0 de *grupo raiz*. Denote por $P_{\text{vert}}(G, \mathcal{R})$ o conjunto dos vetores $(x; y) \in \mathbb{R}^E \times \mathbb{R}^{R_0}$ que satisfazem¹

$$(5.37) \quad \begin{aligned} & \text{(i)} \quad x, y \geq \mathbf{0}, \\ & \text{(ii)} \quad y(R_0) \geq 1, \\ & \text{(iii)} \quad x(\delta(U)) \geq y(U \cap R_0), \quad \text{para todo } U \subseteq V \text{ que não cobre } \mathcal{R}. \end{aligned}$$

Em palavras, y representa a escolha de um vértice de R_0 para a solução e x representa a solução em si. Denote por P_{vert}^E a projeção de P_{vert} em $\mathbb{R}^E$. É fácil ver que $(P_{\text{vert}}^E)_I = P_{\text{STG}}^I$.

Ao contrário do que ocorre com P_{part} , o problema da separação para P_{vert} pode ser resolvido em tempo polinomial. Para ver isso, suponha que nos seja dado um vetor $(x; y) \in \mathbb{R}_+^E \times \mathbb{R}_+^{R_0}$ que satisfaz $y(R_0) \geq 1$. Vamos mostrar como determinar se $(x; y)$ satisfaz as inequações (5.37)(iii). Fixe um grupo $R \neq R_0$. Mais especificamente, vamos mostrar como encontrar um conjunto $U \subseteq V$ tal que $R \cap U = \emptyset$ e $x(\delta(U)) - y(U \cap R_0)$ é mínimo.

Para tanto, considere o grafo $G' = (V', E')$ e a função capacidade $c: E' \rightarrow \mathbb{R}_+$ dados da seguinte forma:

- (i) $V' := V \cup \{s\}$ e G é subgrafo de G' ;
- (ii) para cada $r \in R_0$, G' possui uma aresta sr com $c(sr) := y(r)$;
- (iii) $c(e) = x(e)$ para toda aresta $e \in E$.

Afirmamos que:

$$(5.38) \quad U \subseteq V \text{ é tal que } R \cap U = \emptyset \text{ e } x(\delta_G(U)) - y(U \cap R_0) \text{ é mínimo se e somente se } \delta_{G'}(U \cup \{s\}) \text{ é um } s\text{-}R \text{ corte de capacidade mínima em } G'.$$

Para provar (5.38), note que $U \subseteq V$ é tal que $R \cap U = \emptyset$ se e somente se $\delta_{G'}(U \cup \{s\})$ é um s - R corte em G' . Além disso, temos que

$$(5.39) \quad c(\delta_{G'}(U \cup \{s\})) = x(\delta_G(U)) + y(R_0 \setminus U)$$

e, subtraindo a constante $y(R_0)$ de ambos os lados, obtemos

$$(5.40) \quad c(\delta_{G'}(U \cup \{s\})) - y(R_0) = x(\delta_G(U)) - y(U \cap R_0).$$

Como $y(R_0)$ é constante, segue que $\delta_{G'}(U \cup \{s\})$ é um s - R corte de capacidade mínima se e somente se $R \cap U = \emptyset$ e $x(\delta_G(U)) - y(U \cap R_0)$ é mínimo. Note que podemos encontrar um s - R corte de capacidade mínima em G' em tempo polinomial: basta adicionar um novo vértice t a G' , ligando-o a todos os vértices de R por arestas de capacidade ∞ , e encontrar no novo grafo um s - t corte de capacidade mínima.

Agora, para encontrar a inequação em (5.37)(iii) mais violada por $(x; y)$, basta repetir o raciocínio descrito acima com cada grupo em $\mathcal{R} \setminus \{R_0\}$ no lugar de R . Segue então que o

¹Denotamos por $(x; y)$ o vetor $\begin{pmatrix} x \\ y \end{pmatrix}$.

Figura 5.4: Uma instância do problema de Steiner com grupos com três grupos, A , B e C , sendo A o grupo raiz R_0 . O vetor descrito na figura através dos números que acompanham as arestas do grafo e os vértices em R_0 pertence a P_{vert} . Entretanto, ele viola a inequação proveniente da $\mathcal{R}$ -partição cujas partes se encontram sombreadas, não pertencendo dessa forma a P_{part} .

problema de otimização sobre P_{vert} pode ser resolvido em tempo polinomial (cf. Grötschel, Lovász e Schrijver [23]).

Uma pergunta óbvia é qual a relação entre P_{vert} e P_{part} . Como mostra o exemplo da Figura 5.4, não é sempre verdade que $P_{\text{vert}}^E \subseteq P_{\text{part}}$. Por outro lado, não sabemos se é sempre verdade que $P_{\text{part}} \subseteq P_{\text{vert}}^E$, embora isso seja verdadeiro quando o grafo em questão é uma estrela. De qualquer forma, P_{vert} não é muito mais fraca do que P_{part} . Para ver isso, considere um vetor $(x; y) \in P_{\text{vert}}$ e uma $\mathcal{R}$ -partição $\mathcal{U} = \{V_1, \dots, V_k\}$. Como nenhum V_i cobre $\mathcal{R}$ e como $(x; y)$ satisfaz as inequações em (5.37), temos que

$$(5.41) \quad y(R_0) = \sum_{i=1}^k y(V_i \cap R_0) \leq \sum_{i=1}^k x(\delta(V_i)) = 2x(\Delta(\mathcal{U})).$$

Logo, como $y(R_0) \geq 1$, segue que $x(\Delta(\mathcal{U})) \geq 1/2$. Como $\mathcal{U}$ é genérico, segue que x satisfaz $x(\Delta(\mathcal{U})) \geq 1/2$ para qualquer $\mathcal{R}$ -partição $\mathcal{U}$. Uma consequência imediata disso é que, se $l: E \rightarrow \mathbb{R}_+$ é uma função comprimento, então

$$(5.42) \quad \min\{l^\top x \mid (x; y) \in P_{\text{vert}}\} \geq (1/2) \min\{l^\top x \mid x \in P_{\text{part}}\},$$

ou seja, há um limite para o quanto P_{vert} pode ser mais fraca do que P_{part} .

5.4.2. Arestas artificiais

A formulação que apresentamos a seguir se baseia numa idéia de Stolfi [45]. Novamente, seja $G = (V, E)$ um grafo e $\mathcal{R}$ uma coleção de grupos. Considere o grafo $G' = (V', E')$ e a coleção de grupos $\mathcal{R}'$ dados da seguinte forma:

- (i) G é subgrafo de G' ;
- (ii) para cada $R \in \mathcal{R}$ e $r \in R$, G' possui um vértice r' , ligado a r através de uma aresta;

(iii) para cada $R \in \mathcal{R}$, $\mathcal{R}'$ possui um grupo $R' := \{r' \in V' \setminus V \mid r \in R\}$.

Em palavras, para cada vértice de grupo r adicionamos a G um vértice r' , que é o *correspondente* de r , e ligamos r a r' por uma aresta. Depois, consideramos os grupos formados pelos correspondentes dos vértices originais. Chamamos o novo grafo e a nova coleção de grupos de G' e $\mathcal{R}'$, respectivamente.

Podemos agora considerar o poliedro $P_{\text{aresta}}(G', \mathcal{R}')$, composto pelos vetores $x \in \mathbb{R}^{E'}$ que satisfazem

$$(5.43) \quad \begin{aligned} & \text{(i)} \quad x, y \geq 0, \\ & \text{(ii)} \quad x(\delta_{G'}(R')) = 1, \quad \text{para todo } R' \in \mathcal{R}', \\ & \text{(iii)} \quad x(\delta_{G'}(U)) \geq 1, \quad \text{para todo } U \subseteq V' \text{ que é } \mathcal{R}'\text{-separador.} \end{aligned}$$

Denote por $P_{\text{aresta}}^E(G', \mathcal{R}')$ a projeção de $P_{\text{aresta}}(G', \mathcal{R}')$ em $\mathbb{R}^E$. Novamente, é fácil ver que $(P_{\text{aresta}}^E(G', \mathcal{R}'))_I = P_{\text{STG}}^I(G, \mathcal{R})$. Note também que o problema da separação pode ser resolvido em tempo polinomial como uma sequência de problemas de corte mínimo em grafos, como explicado no início da seção 5.2.1.

Na verdade, a formulação que acabamos de apresentar é mais forte do que a apresentada na seção 5.4.1, ou seja, $P_{\text{aresta}}^E(G', \mathcal{R}') \subseteq P_{\text{vert}}^E(G, \mathcal{R})$. Para ver isso, escreva $E' = E \cup F$, com $E \cap F = \emptyset$, e considere um vetor $x' := (x_1; x_2) \in P_{\text{aresta}}(G', \mathcal{R}')$, com $x_1 \in \mathbb{R}^E$ e $x_2 \in \mathbb{R}^F$. Considere agora o vetor $y \in \mathbb{R}^{R_0}$ tal que $y(r) := x'(rr')$ para todo $r \in R_0$. Afirmamos que $(x_1; y) \in P_{\text{vert}}(G, \mathcal{R})$. Obviamente, $(x_1; y)$ satisfaz (5.37)(i). Além disso, vem de (5.43)(ii) que $y(R_0) = 1$, e $(x_1; y)$ satisfaz (5.37)(ii).

Resta provar que $(x_1; y)$ satisfaz as inequações (5.37)(iii). Para tanto, considere um conjunto $U \subseteq V$ que não cobre $\mathcal{R} \setminus \{R_0\}$. Denote por $\hat{\mathcal{R}}$ a união dos conjuntos em $\mathcal{R}$, i.e., o conjunto de todos os vértices de grupo em $\mathcal{R}$. Considere o conjunto

$$(5.44) \quad U' := U \cup R'_0 \cup \{r' \in V' \setminus V \mid r \in \hat{\mathcal{R}} \cap U\}.$$

Em outras palavras, adicionamos a U os vértices em R'_0 e, para cada vértice de grupo contido em U , adicionamos a U também seu correspondente no grafo G' . A Figura 5.5 ilustra essa idéia. Note que U' é $\mathcal{R}'$ -separador em G' . Com isso, temos que

$$(5.45) \quad x_1(\delta_G(U)) = x'(\delta_{G'}(U')) - y(R_0 \setminus U) \geq 1 - (1 - y(U \cap R_0)) = y(U \cap R_0),$$

pois, como U' é $\mathcal{R}'$ -separador em G' , $x'(\delta_{G'}(U')) \geq 1$. Segue que $(x_1; y)$ satisfaz as inequações (5.37)(iii).

5.4.3. Uma formulação dirigida

Apresentaremos agora a versão dirigida da formulação apresentada na seção 5.4.1. Como no caso do problema de Steiner (cf. Chopra e Rao [3, 4]), esta formulação será mais forte que as não-dirigidas, como veremos.

Seja $G = (V, E)$ um grafo e $\mathcal{R}$ uma coleção de grupos. Novamente, fixe um grupo raiz $R_0 \in \mathcal{R}$. Considere a versão dirigida do grafo G , ou seja, o grafo $D = (V, A)$ obtido de G substituindo-se

Figura 5.5: Nessa figura, o grupo raiz, R_0 , é composto pelos vértices r_1, r_2, r_3 . Os vértices u, v também são vértices de grupo. Note que para cada vértice de grupo é adicionado um vértice ligado a ele. O conjunto cinza escuro representa o conjunto U ; o conjunto cinza claro (que contém o escuro) representa U' . Observe como construímos U' a partir de U .

$$(5.46) \quad \bullet \text{ --- } \bullet \quad \text{por} \quad \bullet \text{ --- } \bullet$$

Denote por $P_{\text{dir}}(G, \mathcal{R})$ o conjunto dos vetores $(w; x; z) \in \mathbb{R}^A \times \mathbb{R}^E \times \mathbb{R}^{R_0}$ que satisfazem

$$(5.47) \quad \begin{aligned} & \text{(i)} \quad w, x, z \geq 0, \\ & \text{(ii)} \quad z(R_0) \geq 1, \\ & \text{(iii)} \quad x(uv) = w(u, v) + w(v, u), \quad \text{para todo } uv \in E, \\ & \text{(iv)} \quad w(\delta^{\text{out}}(U)) \geq z(U \cap R_0), \quad \text{para todo } U \subseteq V \text{ que não cobre } \mathcal{R}. \end{aligned}$$

Denote por P_{dir}^E a projeção de P_{dir} em $\mathbb{R}^E$. É fácil provar que $(P_{\text{dir}}^E)_I = P_{\text{STG}}^I$.

O leitor já deve ter percebido que (5.47) é uma espécie de versão dirigida de (5.37). O problema da separação para (5.47) pode ser resolvido de forma análoga ao problema da separação para (5.37).

Na seção 5.4.1 mostramos que P_{vert} pode ser mais fraca do que P_{part} . É fácil ver que P_{dir} é mais forte do que P_{vert} . Mostraremos agora que P_{dir} também é mais forte do que P_{part} , ou seja, mostraremos que $P_{\text{dir}}^E \subseteq P_{\text{part}}$. Para tanto, considere um vetor $(w; x; z) \in P_{\text{dir}}$ e seja $\mathcal{U} = \{V_1, \dots, V_k\}$ uma $\mathcal{R}$ -partição qualquer. Como cada V_i não cobre $\mathcal{R}$, temos

$$(5.48) \quad z(R_0) = \sum_{i=1}^k z(V_i \cap R_0) \leq \sum_{i=1}^k w(\delta^{\text{out}}(V_i)) = \sum_{uv \in \Delta(\mathcal{U})} (w(u, v) + w(v, u)) = x(\Delta(\mathcal{U}))$$

e, como $z(R_0) \geq 1$, segue que $x(\Delta(\mathcal{U})) \geq 1$. Mas então, como $\mathcal{U}$ é arbitrário, x satisfaz todas as inequações de $\mathcal{R}$ -partições, pertencendo assim a P_{part} .

Podemos mostrar mais ainda, podemos mostrar que qualquer $x \in P_{\text{dir}}^E$ satisfaz todas as inequações de partição de grupos, i.e., as inequações (5.29). Para ver isso, considere uma

partição de grupos $\mathcal{U} = \{V_1, \dots, V_k\}$. Lembre-se de que para cada V_i existe um grupo R_i com $R_i \subseteq V_i$. Mas então

$$(5.49) \quad x(\Delta(\mathcal{U})) = \sum_{uv \in \Delta(\mathcal{U})} (w(u, v) + w(v, u)) = \sum_{i=1}^k w(\delta^{\text{out}}(V \setminus V_i)) \geq \sum_{i=1}^k z((V \setminus V_i) \cap R_0).$$

Como cada $r \in R_0$ pertence a algum V_i , cada $z(r)$ é somado $k - 1$ vezes na soma da direita. Mas então, como $z(R_0) \geq 1$, temos que

$$(5.50) \quad x(\Delta(\mathcal{U})) \geq \sum_{i=1}^k z((V \setminus V_i) \cap R_0) = (k - 1)z(R_0) \geq k - 1$$

e x satisfaz a inequação de partição de grupos $x(\Delta(\mathcal{U})) \geq k - 1$, como queríamos.

Formulações dirigidas para o problema de Steiner em grafos, que são generalizadas por essa que apresentamos, foram estudadas por diversos autores, entre eles Chopra e Rao [3, 4], Goemans [20] e Goemans e Myung [21]. O problema de encontrar-se um sistema simples para a projeção de tais formulações no espaço $\mathbb{R}^E$, ou seja, no espaço natural das variáveis, foi também muito estudado. Goemans [20] descreve diversas classes de facetas de uma tal projeção, algumas delas bem complicadas, o que de certa forma mostra que não existe um sistema simples que descreva uma tal projeção.

5.4.4. Eliminação de subcircuitos

Nesta seção apresentamos uma formulação para P_{STG}^1 que faz uso das inequações de eliminação de subcircuitos. Tal formulação é próxima à proposta por Salazar [41]. Novamente, considere um grafo $G = (V, E)$, uma coleção de grupos $\mathcal{R}$ e fixe um grupo raiz $R_0 \in \mathcal{R}$. Denote por $P_{\text{circ}}(G, \mathcal{R})$ o poliedro composto pelos vetores $(x; y; z) \in \mathbb{R}^E \times \mathbb{R}^V \times \mathbb{R}^{R_0}$ que satisfazem

$$(5.51) \quad \begin{array}{ll} \text{(i)} & y(R) \geq 1, \\ \text{(ii)} & z(R_0) = 1, \\ \text{(iii)} & x(E) = y(V) - 1, \\ \text{(iv)} & x(E[U]) \leq y(U) - z(U \cap R_0), \quad \text{para todo } U \subseteq V, \\ \text{(v)} & x(E[U]) \leq y(U) - 1, \quad \text{para todo } U \subseteq V \text{ que contém um grupo,} \\ \text{(vi)} & x, y, z \geq 0. \end{array}$$

Intuitivamente, nessa formulação as variáveis y nos dizem quais vértices fazem parte da solução. As variáveis z indicam a raiz da árvore e as variáveis x indicam as arestas que participam da solução.

As inequações (5.51)(iv) e (5.51)(v) são chamadas de *inequações de eliminação de subcircuitos*. Denote por P_{circ}^E a projeção de P_{circ} em $\mathbb{R}^E$. O fato de que $(P_{\text{circ}}^E)_1 = P_{\text{STG}}^1$ vai decorrer da equivalência entre P_{circ} e P_{dir} , como veremos mais adiante. Vamos por ora discutir o problema de separação para as inequações de eliminação de subcircuitos.

Queremos resolver o seguinte problema: dado um vetor $(x; y; z) \in \mathbb{R}_+^E \times \mathbb{R}_+^V \times \mathbb{R}_+^{R_0}$, verifique que $(x; y; z)$ satisfaz as inequações (5.51)(iv)–(v) ou encontre uma dessas inequações violada

por $(x; y; z)$. Note que podemos supor que $z(r) \leq y(r)$ para todo $r \in R_0$, caso contrário a inequação $x(E[\{r\}]) \leq y(r) - z(r)$ está violada por $(x; y; z)$.

Com isso, nosso problema resume-se ao seguinte:

(5.52) dados: vetores $x \in \mathbb{R}_+^E$, $f \in \mathbb{R}_+^V$ e um conjunto $S \subseteq V$,
encontre: um conjunto $U \subseteq V$ tal que $S \subseteq U$ e $f(U) - x(E[U])$ seja mínimo.

De fato, para se encontrar uma inequação em (5.51)(iv) mais violada, basta tomar $S := \emptyset$ e fazer $f(v) := y(v)$ se $v \notin R_0$ e $f(v) := y(v) - z(v)$ caso contrário. Para se encontrar uma inequação em (5.51)(v) mais violada, basta resolver (5.52) para cada $R \in \mathcal{R}$, fazendo $S := R$ e $f(v) := y(v)$ para cada $v \in V$.

Vamos então mostrar como resolver (5.52). Dados x, f e S como em (5.52), considere o grafo dirigido $D = (E \cup V \cup \{s, t\}, A)$ e a função capacidade $c: A \rightarrow \mathbb{R}_+$ dados da seguinte forma:

- (i) para cada $uv \in E$, D possui arcos (uv, u) e (uv, v) , ambos com capacidade infinita;
- (ii) para cada $e \in E$, D possui um arco (s, e) , com capacidade $x(e)$;
- (iii) para cada $v \in V$, D possui um arco (v, t) , com capacidade $f(v)$;
- (iv) para cada $v \in S$, D possui um arco (s, v) , com capacidade infinita.

Afirmamos que:

(5.53) se $\delta^{\text{out}}(W)$ é um s - t corte de capacidade mínima em D então $S \subseteq W \cap V$ e $f(W \cap V) - x(E[W \cap V]) \leq f(U) - x(E[U])$ para todo conjunto $U \subseteq V$ com $S \subseteq U$.

Para ver (5.53), considere primeiro um conjunto $U \subseteq V$ tal que $S \subseteq U$. Escreva $U' := E[U] \cup U \cup \{s\}$. Note que $\delta^{\text{out}}(U')$ é um s - t corte em D e que, devido à maneira como construímos D , $\delta^{\text{out}}(U')$ tem capacidade finita. Mas então

$$(5.54) \quad c(\delta^{\text{out}}(U')) = x(E \setminus E[U]) + f(U) \iff c(\delta^{\text{out}}(U')) - x(E) = f(U) - x(E[U]).$$

Segue de (5.54) que se $\delta^{\text{out}}(W)$ é um s - t corte de capacidade mínima em D então $c(\delta^{\text{out}}(W)) - x(E) \leq f(U) - x(E[U])$ para todo $U \subseteq V$ com $S \subseteq U$.

Agora, observe que se $\delta^{\text{out}}(W)$ é um s - t corte de capacidade mínima em D então sua capacidade é finita (já que $\delta^{\text{out}}(S \cup \{s\})$ é um s - t corte em D de capacidade finita). Com isso, pela construção de D , devemos ter $S \subseteq W \cap V$. Novamente pela construção de D , devemos ter que se $uv \in W \cap E$ então $u, v \in W \cap V$. Além disso, observe também que se para alguma aresta $uv \in E$ temos que $u, v \in W \cap V$ mas $uv \notin W \cap E$ então devemos ter $x(uv) = 0$, caso contrário $\delta^{\text{out}}(W \cup \{uv\})$ seria um s - t corte em D de capacidade inferior à capacidade de $\delta^{\text{out}}(W)$. Mas então, usando (5.54), temos que

$$(5.55) \quad \begin{aligned} f(W \cap V) - x(E[W \cap V]) &= c(\delta^{\text{out}}(E[W \cap V] \cup (W \cap V) \cup \{s\})) - x(E) \\ &= c(\delta^{\text{out}}(W)) - x(E) \\ &\leq f(U) - x(E[U]), \end{aligned}$$

para todo $U \subseteq V$ com $S \subseteq U$, o que prova (5.53).

A equivalência de P_{dir} e P_{circ} . Como dissemos anteriormente, P_{dir} e P_{circ} são equivalentes. Vamos mostrar isso agora. Um resultado semelhante foi obtido para o problema de Steiner em grafos por Goemans e Myung [21]. Considere novamente a versão dirigida $D = (V, A)$ do grafo G , dada como na seção 5.4.3.

Teorema 5.12. *Seja $G = (V, E)$ um grafo e $\mathcal{R}$ uma coleção de grupos. Então $P_{\text{dir}}^E(G, \mathcal{R}) = P_{\text{circ}}^E(G, \mathcal{R})$.*

Prova: Provaremos primeiro que $P_{\text{dir}}^E \subseteq P_{\text{circ}}^E$. Para tanto, seja $(w; x; z) \in P_{\text{dir}}$, sendo $w \in \mathbb{R}^A$, $x \in \mathbb{R}^E$ e $z \in \mathbb{R}^{R_0}$. Vamos supor, por ora, que $z(R_0) = 1$ (mais tarde trataremos do caso mais geral). Considere o vetor $y \in \mathbb{R}^V$ dado da seguinte forma:

$$(5.56) \quad y(v) := \begin{cases} w(\delta^{\text{in}}(v)), & \text{se } v \notin R_0, \\ w(\delta^{\text{in}}(v)) + z(v), & \text{caso contrário.} \end{cases}$$

Vamos mostrar que $(x; y; z) \in P_{\text{circ}}$. Obviamente, $x, y, z \geq 0$ e $z(R_0) = 1$. Além disso, se $R \in \mathcal{R}$ então $w(\delta^{\text{in}}(R)) = w(\delta^{\text{out}}(V \setminus R)) \geq z((V \setminus R) \cap R_0) = z(R_0 \setminus R)$, já que $V \setminus R$ não cobre $\mathcal{R}$. Com isso, temos que

$$(5.57) \quad y(R) \geq w(\delta^{\text{in}}(R)) + z(R \cap R_0) \geq z(R_0 \setminus R) + z(R \cap R_0) = 1,$$

e $(x; y; z)$ satisfaz as inequações (5.51)(i). Observe também que

$$(5.58) \quad x(E) = w(A) = \sum_{v \in V} w(\delta^{\text{in}}(v)) = y(V) - z(V \cap R_0) = y(V) - 1,$$

e $(x; y; z)$ satisfaz (5.51)(iii). Para ver que $(x; y; z)$ satisfaz as inequações (5.51)(iv), considere um conjunto $U \subseteq V$ e observe que

$$(5.59) \quad x(E[U]) = w(A[U]) \leq \sum_{v \in U} w(\delta^{\text{in}}(v)) = y(U) - z(U \cap R_0).$$

Resta mostrar que $(x; y; z)$ satisfaz as inequações (5.51)(v). Para tanto, considere um conjunto $U \subseteq V$ que contém um grupo. Note que, como $V \setminus U$ não cobre $\mathcal{R}$, temos $w(\delta^{\text{in}}(U)) = w(\delta^{\text{out}}(V \setminus U)) \geq z((V \setminus U) \cap R_0) = z(R_0 \setminus U)$. Mas então

$$(5.60) \quad \begin{aligned} x(E[U]) &= w(A[U]) = \sum_{v \in U} w(\delta^{\text{in}}(v)) - w(\delta^{\text{in}}(U)) \\ &\leq y(U) - z(U \cap R_0) - z(R_0 \setminus U) \\ &= y(U) - 1, \end{aligned}$$

e provamos que $(x; y; z) \in P_{\text{circ}}$, o que implica que $x \in P_{\text{circ}}^E$.

Falta tratar do caso em que $z(R_0) > 1$. Vamos mostrar primeiro que P_{circ}^E é do tipo bloqueador. Para tanto, fixe uma aresta $e \in E$ e um vetor qualquer $x \in P_{\text{circ}}^E$. Vamos mostrar que $x + \chi^e$ também pertence a P_{circ}^E . Considere os vetores y, z tais que $(x; y; z) \in P_{\text{circ}}$. Escreva $e = uv$ e considere o vetor y' obtido de y somando-se $1/2$ a $y(u)$ e $y(v)$. É fácil verificar que $(x + \chi^e; y'; z) \in P_{\text{circ}}$, logo $x + \chi^e \in P_{\text{circ}}^E$. Como e e x são arbitrários, segue que, para cada $e \in E$, χ^e pertence ao cone característico de P_{circ}^E , i.e., P_{circ}^E é do tipo bloqueador.

Agora, observe que se $(w; x; z) \in P_{\text{dir}}$ e $z(R_0) > 1$ então o vetor $(\tilde{w}; \tilde{x}; \tilde{z})$, resultado da multiplicação de $(w; x; z)$ por $1/z(R_0)$, também pertence a P_{dir} . Como $\tilde{z}(R_0) = 1$, sabemos que $\tilde{x} \in P_{\text{circ}}^E$. Mas então, como $\tilde{x} \leq x$ e P_{circ}^E é do tipo bloqueador, segue que $x \in P_{\text{circ}}^E$, como queríamos.

Vamos mostrar agora que $P_{\text{circ}}^E \subseteq P_{\text{dir}}^E$. Para tanto, considere um vetor $(x; y; z) \in P_{\text{circ}}$, com $x \in \mathbb{R}^E$, $y \in \mathbb{R}^V$ e $z \in \mathbb{R}^{R_0}$. Vamos mostrar que $x \in P_{\text{dir}}^E$. Suponha que exista um vetor $w \in \mathbb{R}^A$ tal que

$$(5.61) \quad \begin{aligned} w &\geq 0, \\ w(\delta^{\text{in}}(v)) &= y(v), & \text{para todo } v \in V \setminus R_0, \\ w(\delta^{\text{in}}(v)) &= y(v) - z(v), & \text{para } v \in R_0, \\ w(u, v) + w(v, u) &= x(uv), & \text{para toda aresta } uv \in E. \end{aligned}$$

Afirmamos que $(w; x; z) \in P_{\text{dir}}$. É fácil ver que $(w; x; z)$ satisfaz (5.47)(i)–(iii). Resta verificar que $(w; x; z)$ satisfaz as inequações (5.47)(iv). Para ver isso, considere um conjunto $U \subseteq V$ que não cobre $\mathcal{R}$. Então

$$(5.62) \quad w(\delta^{\text{out}}(U)) = w(\delta^{\text{in}}(V \setminus U)) = \sum_{v \in V \setminus U} w(\delta^{\text{in}}(v)) - w(A[V \setminus U]).$$

Como U não cobre $\mathcal{R}$, $V \setminus U$ contém um grupo e, assim, $x(E[V \setminus U]) \leq y(V \setminus U) - 1$. Logo

$$(5.63) \quad w(\delta^{\text{out}}(U)) = y(V \setminus U) - z(R_0 \setminus U) - x(E[V \setminus U]) \geq 1 - z(R_0 \setminus U) = z(U \cap R_0),$$

e $(w; x; p)$ satisfaz as inequações (5.47)(iv).

Resta apenas mostrar que existe um vetor w que satisfaz (5.61). Para tanto, considere o grafo dirigido $H = (E \cup V \cup \{s, t\}, B)$ e a função capacidade $c: B \rightarrow \mathbb{R}_+$ dados da seguinte forma:

- (i) para cada $uv \in E$, H possui arcos (uv, u) e (uv, v) , ambos com capacidade infinita;
- (ii) para cada $uv \in E$, H possui um arco (s, uv) com capacidade $x(uv)$;
- (iii) para cada $v \in V$, H possui um arco (v, t) com capacidade $y(v)$, se $v \notin R_0$, ou $y(v) - z(v)$, se $v \in R_0$.

Todo s – t fluxo sujeito a c em H tem capacidade no máximo $c(\delta_H^{\text{out}}(\{s\})) = x(E)$. É fácil ver que se H admite um s – t fluxo f de valor $x(E)$ então existe um vetor w que satisfaz (5.61). De fato, basta fazer $w(u, v) := f(uv, v)$ para todo $(u, v) \in A$.

Vamos então mostrar que H admite um s – t fluxo de valor $x(E)$. Note que, se esse não for o caso, então pelo teorema do fluxo máximo corte mínimo sabemos que existe um s – t corte $\delta_H^{\text{out}}(W)$ de capacidade menor do que $x(E)$. Escreva $M := W \cap E$ e $U := W \cap V$. Note que, como $\delta_H^{\text{out}}(W)$ tem capacidade finita, pela maneira como construímos H , devemos ter que se $uv \in M$ então $u, v \in U$, i.e., $M \subseteq E[U]$. Mas então temos que

$$(5.64) \quad \begin{aligned} x(E) > c(\delta_H^{\text{out}}(W)) &= x(E \setminus M) + y(U) - z(U \cap R_0) \\ &= x(E) - x(M) + y(U) - z(U \cap R_0) \\ &\geq x(E) - x(E[U]) + y(U) - z(U \cap R_0). \end{aligned}$$

Agora, observe que como $(x; y; z) \in P_{\text{circ}}$ temos $x(E[U]) \leq y(U) - z(U \cap R_0)$, logo

$$(5.65) \quad x(E) > x(E) - x(E[U]) + y(U) - z(U \cap R_0) \geq x(E),$$

uma contradição. □

5.5. Descrições completas

Como o problema de Steiner com grupos é NP-difícil, há pouca esperança de conseguirmos obter boas descrições de poliedros associados ao problema (para uma descrição formal do que queremos dizer e também para obter referências sobre o assunto, consulte o capítulo 5 do livro de Schrijver [43]). Entretanto, quando nos restringimos a casos particulares do problema, casos esses para os quais existem algoritmos polinomiais, podemos encontrar tais descrições. A seguir, fornecemos descrições completas de P_{STG}^1 para dois tais casos particulares.

5.5.1. Dois grupos

Seja $G = (V, E)$ um grafo e $\mathcal{R}$ uma coleção de grupos. Quando temos exatamente dois grupos, o problema de Steiner com grupos pode ser resolvido em tempo polinomial. É fácil ver isso se os grupos possuem intersecção. Se os dois grupos, digamos R e S , são disjuntos, basta observar que toda $\mathcal{R}$ -árvore minimal é um caminho que conecta um vértice de R a um vértice de S . Mas então, para uma função comprimento $l: E \rightarrow \mathbb{R}_+$, a $\mathcal{R}$ -árvore de menor comprimento é o menor R - S caminho em G . Para encontrá-lo, basta encontrar um r - s caminho em G de comprimento mínimo para cada par de vértices $r \in R, s \in S$, o que pode ser feito em tempo polinomial pelo algoritmo de Dijkstra [7] (veja também Schrijver [43], capítulo 7).

No resultado a seguir, caracterizamos completamente o poliedro $P_{\text{STG}}^1(G, \mathcal{R})$ no caso em que $\mathcal{R}$ possui exatamente dois grupos disjuntos.

Teorema 5.13. *Seja $G = (V, E)$ um grafo e $\mathcal{R}$ uma coleção de grupos com exatamente dois grupos R e S disjuntos. Então $P_{\text{STG}}^1(G, \mathcal{R})$ é exatamente o conjunto dos vetores $x \in \mathbb{R}^E$ que satisfazem*

$$(5.66) \quad \begin{aligned} & \text{(i)} \quad x \geq 0, \\ & \text{(ii)} \quad x(\delta(U)) \geq 1, \text{ para todo } U \subseteq V \setminus S \text{ tal que } R \subseteq U. \end{aligned}$$

Prova: É fácil ver que todo $x \in P_{\text{STG}}^1$ satisfaz as inequações em (5.66). Vamos provar a afirmação reversa. Para tanto, suponha que x satisfaça as inequações em (5.66) e que seja um vértice do poliedro que tais inequações definem. Seja $D = (V \cup \{r, s\}, A)$ o grafo dirigido e $c: A \rightarrow \mathbb{R}_+$ a função capacidade dados da seguinte forma:

- (i) para cada $uv \in E$, D possui arcos (u, v) e (v, u) , ambos com capacidade $x(uv)$;
- (ii) para cada $u \in R$, D possui um arco (r, u) de capacidade infinita;
- (iii) para cada $u \in S$, D possui um arco (u, s) de capacidade infinita.

Como x satisfaz (5.66)(ii), se $\delta^{\text{out}}(U)$ é um r - s corte em D então $c(\delta^{\text{out}}(U)) \geq 1$. Agora, como x é um vértice e $R \cap S = \emptyset$, deve haver algum $U \subseteq V \setminus S$ com $R \subseteq U$ tal que $x(\delta(U)) = 1$. Mas então $c(\delta^{\text{out}}(U \cup \{r\})) = 1$, logo podemos concluir que um r - s corte mínimo em D tem capacidade 1.

Seja então $f: A \rightarrow \mathbb{R}_+$ um r - s fluxo máximo em D sujeito a c . Suponha que $|\text{supp } f|$ seja mínimo dentre todos os r - s fluxos máximos em D sujeitos a c . Com isso, temos que:

$$(5.67) \quad \text{para cada } uv \in E, \text{ ou } f(u, v) = 0 \text{ ou } f(v, u) = 0.$$

Observe que, pelo teorema do fluxo-máximo corte-mínimo, f tem valor exatamente 1. Mas então (cf. Teorema 1.2) existem r - s caminhos dirigidos $P_1, \dots, P_k$ em D e reais não-negativos $\lambda_1, \dots, \lambda_k$ tais que $f = \lambda_1 \chi^{AP_1} + \dots + \lambda_k \chi^{AP_k}$ e $\lambda_1 + \dots + \lambda_k = 1$. Cada r - s caminho dirigido P_i dá origem a um R - S caminho Q_i em G da maneira natural. Usando (5.67) e como $f \leq c$, temos que

$$(5.68) \quad \tilde{x} := \lambda_1 \chi^{EQ_1} + \dots + \lambda_k \chi^{EQ_k} \leq x.$$

Note que $\chi^{EQ_i} \in P_{\text{STG}}^I$ para todo i . Segue que $\tilde{x} \in P_{\text{STG}}^I$. Agora, como P_{STG}^I é do tipo bloqueador e $\tilde{x} \leq x$, $x \in P_{\text{STG}}^I$, como queríamos. $\square$

Observe que o problema da separação para (5.66) pode ser resolvido em tempo polinomial. Assim, o método dos elipsóides nos fornece um algoritmo polinomial para o problema de Steiner com grupos quando há exatamente dois grupos (cf. Grötschel, Lovász e Schrijver [23]).

5.5.2. Árvores 2-grupo restritas

Seja $T = (V, E)$ uma árvore e $\mathcal{R}$ uma coleção de grupos. Fixe um vértice $s \in V$ e denote por $\mathcal{R}_s$ o conjunto dos grupos que não contêm s . Dizemos que T é *2-grupo restrita* (em relação a s) se:

- (5.69) (i) $T - s$ não contém nenhuma $\mathcal{R}$ -árvore;
(ii) cada componente de $T - s$ possui vértices de no máximo dois grupos distintos em $\mathcal{R}_s$.

Vamos reduzir o problema de Steiner com grupos em árvores 2-grupo restritas ao problema da cobertura de peso mínimo por arestas. Dado um grafo $G = (V', E')$, um conjunto $C \subseteq E'$ é uma *cobertura por arestas* de G se (V', C) não possui vértices isolados. O *problema da cobertura de peso mínimo por arestas* é o seguinte:

- (5.70) dados: um grafo $G = (V', E')$ e uma função peso $w: E' \rightarrow \mathbb{R}_+$,
encontre: uma cobertura por arestas de G de peso mínimo.

O problema da cobertura de peso mínimo por arestas pode ser resolvido em tempo polinomial (veja a seção 19.3 do livro de Schrijver [43]).

Vamos então mostrar nossa redução. Para tanto, considere os conjuntos de vértices $V_1, \dots, V_k$ dos componentes de $T - s$ e escreva $T_i := T[V_i \cup \{s\}]$, $i = 1, \dots, k$. Denote por $\hat{\mathcal{R}}_i$ o conjunto dos vértices dos grupos em $\mathcal{R}_s$ que pertencem a V_i . Para cada $X \subseteq \hat{\mathcal{R}}_i$ com $|X| \leq 2$, denote por T_X a menor subárvore (em número de arestas) de T_i que contém s e os vértices em X . Escreva $\mathcal{S} := \{X \mid |X| \leq 2 \text{ e } X \subseteq \hat{\mathcal{R}}_i \text{ para algum } i\}$. As condições em (5.69) implicam que:

- (5.71) T' é uma $\mathcal{R}$ -árvore em T se e somente se existem conjuntos $X_1, \dots, X_t$ em $\mathcal{S}$ tais que $X_1 \cup \dots \cup X_t$ cobre $\mathcal{R}_s$ e $T_{X_1}, \dots, T_{X_t} \subseteq T'$.

Considere agora uma função comprimento $l: E \rightarrow \mathbb{R}_+$. Para cada $X \in \mathcal{S}$, denote por $\mathcal{R}_X$ o conjunto dos grupos em $\mathcal{R}_s$ cobertos por X . Note que $|\mathcal{R}_X| \leq 2$ para todo $X \in \mathcal{S}$. Escreva $\mathcal{E} := \{\mathcal{R}_X \mid X \in \mathcal{S}\}$ e considere a função peso $w: \mathcal{E} \rightarrow \mathbb{R}_+$ dada por $w(\mathcal{R}_X) := l(ET_X)$, para todo $X \in \mathcal{S}$. Segue de (5.71) que:

$$(5.72) \quad \text{se } \mathcal{C} \text{ é uma cobertura de peso mínimo por arestas do grafo } \mathcal{G} = (\mathcal{R}_s, \mathcal{E}) \text{ então} \\ \bigcup_{X: \mathcal{R}_X \in \mathcal{C}} T_X \text{ é uma } \mathcal{R}\text{-árvore de comprimento mínimo em } T.$$

(Note que $\mathcal{G}$ pode conter arestas múltiplas e laços.) Isso encerra nossa redução.

Tomando por base a redução apresentada, podemos derivar uma descrição completa de $P_{\text{STG}}^1(T, \mathcal{R})$ de uma descrição do politopo do problema da cobertura por arestas. Nossa descrição, entretanto, fará uso de variáveis artificiais, de modo que P_{STG}^1 será a projeção do poliedro que definiremos em $\mathbb{R}^E$.

Dado um grafo $G = (V', E')$, o *politopo do problema da cobertura por arestas*, denotado por $P_{\text{cob}}(G)$, é o fecho convexo dos vetores característicos de coberturas por arestas de G . Sabe-se que (cf. Schrijver [43], seção 27.2) $P_{\text{cob}}(G)$ é exatamente o conjunto dos vetores $x \in \mathbb{R}^E$ que satisfazem:

$$(5.73) \quad \begin{aligned} &0 \leq x \leq 1, \\ &x(E[U] \cup \delta(U)) \geq \lceil \tfrac{1}{2}|U| \rceil, \quad \text{para cada } U \subseteq V', |U| \text{ ímpar.} \end{aligned}$$

Considere novamente o grafo $\mathcal{G} = (\mathcal{R}_s, \mathcal{E})$. Temos então o seguinte resultado:

Teorema 5.14. *O poliedro $P_{\text{STG}}^1(T, \mathcal{R})$ é a projeção em $\mathbb{R}^E$ do poliedro composto pelos vetores $(x; z) \in \mathbb{R}^E \times \mathbb{R}^{\mathcal{E}}$ que satisfazem:*

$$(5.74) \quad \begin{aligned} &x \geq 0, \\ &0 \leq z \leq 1, \\ &x(e) \geq z(\mathcal{R}_X), \quad \text{para cada } e \in E \text{ e } \mathcal{R}_X \in \mathcal{E} \text{ com } e \in ET_X, \\ &z(\mathcal{E}[U] \cup \delta_{\mathcal{G}}(U)) \geq \lceil \tfrac{1}{2}|U| \rceil, \quad \text{para cada } U \subseteq \mathcal{R}_s, |U| \text{ ímpar.} \end{aligned}$$

Prova: Imediato da discussão anterior. □

5.6. Observações finais

Nesse capítulo estudamos um poliedro relacionado ao problema de Steiner com grupos. Estudamos também várias formulações para o problema. Tal estudo será útil ao desenvolvimento de um algoritmo *branch-and-cut* para o problema, como veremos no capítulo 6.

CAPÍTULO 6

Um algoritmo *branch-and-cut*

Neste capítulo descrevemos o algoritmo *branch-and-cut* que desenvolvemos para o problema de Steiner com grupos. Começamos com uma breve descrição geral do funcionamento de um tal algoritmo e em seguida fornecemos uma descrição mais detalhada do algoritmo desenvolvido. Fornecemos também resultados computacionais que obtivemos.

6.1. Estrutura básica de um algoritmo *branch-and-cut*

Um algoritmo *branch-and-cut* é basicamente um algoritmo de busca exaustiva que tenta resolver um problema de programação inteira, enumerando todas as possíveis soluções numa estrutura de árvore de decisão.

No capítulo 5 vimos diversas formulações inteiras para o problema de Steiner com grupos. Podemos tentar resolver os problemas de programação inteira correspondentes utilizando um algoritmo *branch-and-cut*, encontrando assim a solução ótima da instância do problema de Steiner com grupos correspondente.

Para formalizar esses conceitos, considere um grafo $G = (V, E)$, uma coleção de grupos $\mathcal{R}$ e uma função comprimento $l: E \rightarrow \mathbb{R}_+$. Podemos considerar a formulação da seção 5.4.1. Para tanto, fixamos um grupo raiz R_0 e consideramos o problema de programação inteira abaixo, em que $x \in \mathbb{R}^E$ e $y \in \mathbb{R}^{R_0}$:

$$(6.1) \quad \begin{aligned} &\text{minimizar} && l^\top x \\ & && \text{(i) } x, y \geq 0, x \text{ inteiro,} \\ & && \text{(ii) } y(R_0) \geq 1, \\ & && \text{(iii) } x(\delta(U)) \geq y(U \cap R_0), \text{ para todo } U \subseteq V \text{ que não cobre } \mathcal{R}. \end{aligned}$$

Nosso algoritmo tentaria resolver o problema de programação inteira (6.1). A solução ótima encontrada fornecer-nos-ia uma $\mathcal{R}$ -árvore de comprimento mínimo em G .

A *relaxação linear* do problema (6.1) é o problema de programação linear obtido de (6.1) eliminando-se a restrição de integralidade, i.e., trocando-se a restrição (6.1)(i) pela restrição

$$(6.2) \quad x, y \geq 0.$$

A solução ótima da relaxação linear pode não ser inteira, mas sempre fornece um limitante inferior para o valor da solução ótima do problema de programação inteira original. Esses limitantes serão importantes para o algoritmo *branch-and-cut*, como veremos a seguir.

Figura 6.1: A árvore de decisão que corresponde ao nosso problema. Cada aresta representa a fixação de uma variável do problema.

Em nosso caso, para cada aresta e do grafo G , temos dois subproblemas: aquele em que fixamos $x(e) = 1$ e aquele em que fixamos $x(e) = 0$. Um algoritmo *branch-and-cut* enumera as possíveis soluções do problema numa árvore de decisão, como ilustra a Figura 6.1. Cada nó corresponde a um subproblema em que algumas variáveis estão fixadas e outras permanecem livres. Usamos limitantes inferiores e superiores para tentar eliminar ramos dessa árvore, tornando a busca mais rápida. Vamos descrever isso mais detalhadamente.

Durante o processo de enumeração, o algoritmo mantém um limitante inferior global, $\underline{z}$, e um limitante superior global, $\bar{z}$. O limitante inferior será proveniente de relaxações lineares e o limitante superior, de heurísticas. Mantemos também uma lista de nós *ativos*. Inicialmente, a árvore consiste apenas de um nó raiz, que é ativo e que representa o problema todo, e temos $\underline{z} = -\infty$ e $\bar{z} = \infty$.

A cada iteração, verificamos se $\underline{z} = \bar{z}$, caso em que podemos parar pois temos a solução ótima. Se esse não é o caso, escolhemos um nó ativo para ser processado. Ao *processar* um nó s , executamos o seguinte procedimento:

- (i) resolvemos a relaxação linear para o subproblema do nó, levando em conta as variáveis fixadas. Isso fornece um limitante inferior $\underline{z}_s$ para o valor da solução ótima do problema de programação inteira para o nó s . Tentamos adicionar planos-de-corte à relaxação, de modo a torná-la mais forte;
- (ii) executamos uma heurística primal, como um algoritmo de aproximação, para o subproblema do nó, levando em conta as variáveis fixadas. Isso fornece um limitante superior $\bar{z}_s$ para o valor da solução ótima do problema de programação inteira do nó em questão.

Há quatro possíveis resultados para o processamento de um nó s :

- (i) se o subproblema do nó é inviável, não precisamos mais explorar a subárvore enraizada no nó em questão;
- (ii) se $\underline{z}_s > \bar{z}$, sabemos que a solução ótima não se encontra na subárvore enraizada no nó s e, como no caso anterior, não precisamos mais explorar essa subárvore;
- (iii) se a solução ótima da relaxação linear do subproblema do nó s é inteira, encontramos a solução ótima inteira do subproblema do nó s e não mais precisamos explorar a subárvore enraizada nesse nó. O mesmo ocorre se $\underline{z}_s = \bar{z}_s$;
- (iv) finalmente, se nenhum dos casos anteriores ocorre, escolhemos uma variável do subproblema que não está fixada e ramificamos a árvore através dela.

Figura 6.2: Um possível estado da árvore de decisão durante a execução do algoritmo. Os nós cinzas já foram processados, o branco ainda não. Ao lado do nó s , temos seus limitantes superior e inferior, na forma $\bar{z}_s$. O problema do nó mais à direita é inviável, por isso seus limitantes não são exibidos. Observe que o limitante superior global, nesse caso, é 22. O limitante inferior global é 19.

Após processarmos um nó, ele se torna inativo, sendo removido da lista de nós ativos. Também aproveitamos para atualizar os limitantes globais, como explicaremos a seguir. Um nó que tem seu subproblema resolvido de forma ótima (caso (iii)) é chamado de nó *inteiro*.

Ramificar através de um nó significa escolher uma variável que não está fixada e criar dois novos nós que representam os subproblemas em que essa variável é fixada em zero ou um, respectivamente. Tais nós são filhos do nó pelo qual ramificamos.

Vamos agora mostrar como atualizar os limitantes globais após o processamento de um nó. Para o limitante superior isso é fácil: se processamos o nó s e obtemos $\bar{z}_s < \bar{z}$, podemos mudar o limitante superior global imediatamente.

Mudar o limitante inferior global requer mais cuidado: devemos comparar $\underline{z}_s$ com os limitantes inferiores de todos os nós da árvore que são inteiros ou pelos quais vamos ramificar mas que ainda não tiveram ambos os seus filhos processados. Devemos então escolher o menor de todos os limitantes inferiores desses nós como o novo limitante inferior global. A Figura 6.2 mostra um exemplo.

Obviamente, existem diversas maneiras de implementar cada etapa do algoritmo. A escolha da relaxação é essencial, bem como a escolha de boas heurísticas. Existem diversas estratégias para se escolher a variável pela qual ramificamos, bem como para se escolher o nó a ser processado em seguida. O livro de Ferreira e Wakabayashi [16] cobre boa parte desses tópicos.

6.2. O algoritmo *branch-and-cut* desenvolvido

Nesta seção descrevemos o algoritmo que desenvolvemos para o problema de Steiner com grupos segundo a estratégia *branch-and-cut*.

6.2.1. Pré-processamento

A fase de pré-processamento precede o algoritmo de busca exaustiva. Durante essa fase, tentamos remover vértices e arestas do grafo, de modo a diminuí-lo o máximo possível.

Durante a fase de pré-processamento, utilizamos os algoritmos de redução descritos no capítulo 4.

6.2.2. Formulação

Em nosso algoritmo utilizamos a formulação dirigida da seção 5.4.3. Para resolvê-la, utilizamos o simplex dual do ILOG CPLEX 8.1. O algoritmo de separação foi implementado como uma sequência de execuções do algoritmo de Goldberg e Tarjan [22] (veja também Schrijver [43], capítulo 10) para fluxo máximo em grafos dirigidos. Escolhemos o menor grupo para ser o grupo raiz. Fazemos isso pois testes empíricos indicaram que tal escolha é a melhor em boa parte dos casos.

O uso do simplex dual é interessante pois a introdução de novas restrições no problema primal é equivalente à introdução de novas variáveis no problema dual. Dessa forma, a última base que possuíamos para o problema dual continua viável e a reotimização é consideravelmente mais rápida.

É importante frisar que todos os cortes gerados são válidos para todos os nós da árvore de decisão. Dessa forma, podemos manter apenas um programa linear, em vez de ter de criar um para cada nó que processamos. Além disso, os cortes gerados para um nó podem fortalecer a formulação para outros nós, o que é proveitoso.

Uma questão importante para a solução da relaxação é o número de inequações do sistema. Se o número se torna muito grande, a solução se torna muito lenta, o que prejudica o algoritmo. Por outro lado, muitas inequações que foram adicionadas no início do processo podem não ser mais justas algum tempo depois, ou seja, podem não ser mais satisfeitas com igualdade depois de algum tempo.

Para remediar esse problema, mantemos uma lista (usualmente chamada de *pool* na literatura sobre algoritmos *branch-and-cut*) das inequações que incluímos em nosso problema. Cada inequação é marcada com o momento em que foi criada. Após um certo número de reotimizações, procuramos por inequações que não estão justas há algum tempo e as removemos do programa linear. Observe que a base dual que temos continua viável e ótima após tal remoção, o que significa que não precisamos reotimizar.

As inequações removidas não são descartadas, mas sim guardadas em uma outra lista. Antes de procurarmos novos cortes, verificamos se alguma inequação dessa lista está violada. Nesse caso, adicionamos tal inequação ao programa linear.

6.2.3. Heurística primal

A heurística primal utilizada foi a heurística dos caminhos mínimos, descrita na seção 3.1. Nós apenas a aplicamos quando a solução da relaxação linear fornece uma solução ótima que não é inteira. Caso contrário, sabemos que o problema foi resolvido e não há necessidade de executar-se a heurística. É importante observar que durante a aplicação da heurística respeitamos a fixação das variáveis.

6.2.4. Estratégias de ramificação e processamento

Finalmente, descrevemos rapidamente algumas das estratégias que utilizamos para ramificação e escolha do nó a ser processado. Se a solução ótima da relaxação não é inteira, escolhemos ramificar pela variável que tem valor mais próximo de $1/2$, ou seja, a variável “mais fracionária”.

A escolha do nó a ser processado é simples: escolhemos o primeiro nó da lista, que é o que há mais tempo aguarda para ser processado. Outras estratégias que testamos foram a escolha do nó que tem o menor limitante superior ou o maior limitante inferior, mas nenhuma diferença foi notada.

6.3. Resultados computacionais

Testamos nosso algoritmo com o conjunto de testes WRP4 da biblioteca Steinlib de Koch, Martin e Voß [34]. Tais instâncias vêm de problemas de roteamento em circuitos. Elas foram propostas por Rohe e Zachariasen [39], que as atacaram utilizando técnicas específicas para problemas retilineares.

6.3.1. Os algoritmos de redução

O desempenho dos algoritmos de redução que utilizamos encontra-se descrito no capítulo 4. Nesta seção vamos comentar sobre o impacto de tais algoritmos sobre o algoritmo *branch-and-cut* que desenvolvemos.

Para analisar tal impacto, desenvolvemos o seguinte teste: executamos o algoritmo *branch-and-cut* sobre o problema original e sobre o reduzido por no máximo 500 segundos de tempo de CPU. Ao final desse tempo o algoritmo é interrompido e o melhor limitante inferior computado é anotado.

Observamos dois fenômenos separados ao executar esse teste. Primeiro, nas instâncias em que obtivemos poucas reduções quase não houve diferença entre os resultados obtidos. Já nas instâncias para as quais conseguimos boas reduções, o impacto foi significativo, como ilustra a Tabela 6.3. Tal tabela traz, para cada instância considerada, o melhor limitante inferior obtido para a instância original e o melhor limitante inferior obtido para a instância reduzida, ambos após 500 segundos de processamento. A tabela contém apenas as instâncias das quais removemos pelo menos 8% do total de vértices de grupo.

Na tabela, uma linha marcada com uma estrela (*) indica um caso em que o algoritmo foi melhor na instância reduzida do que na original. Em algumas das instâncias consideradas, isso não ocorreu. Isso pode ser atribuído à escolha do grupo raiz feita pelo algoritmo, que pode ter mudado graças às reduções. Outra possibilidade é a ordem de geração dos planos-de-corte, sobre a qual não temos controle. De qualquer forma, observe que nesses casos a diferença entre os limitantes é pequena. O mais provável é que o algoritmo tenha sido executado por um pouco mais de tempo em um caso do que no outro (talvez 1 ou 2 segundos a mais), o que resulta no limitante melhor.

Instância	Original	Reduzido
wrp4-16	1 187,52	1 182,07
* wrp4-21	2 705,36	2 951,69
wrp4-24	3 060,55	3 036,30
wrp4-25	804,04	798,90
* wrp4-39	2 673,20	2 887,45
* wrp4-43	826,72	828,81
* wrp4-44	1 199,60	1 422,56
* wrp4-48	2 199,17	2 214,55
* wrp4-49	1 526,83	1 548,23
* wrp4-55	1 508,24	1 549,87
wrp4-56	1 876,36	1 840,27
* wrp4-62	1 057,26	1 152,83
* wrp4-67	1 396,15	1 719,89
* wrp4-69	1 899,47	2 057,63

Tabela 6.3

6.3.2. O algoritmo *branch-and-cut*

O algoritmo *branch-and-cut* que desenvolvemos apresentou excelentes resultados no conjunto de testes considerado. Conseguimos resolver de forma ótima a maioria das instâncias. Além disso, na grande maioria dos casos, a formulação dirigida mostrou-se forte o bastante a ponto de fornecer a solução inteira do problema sem a necessidade de busca exaustiva.

Mesmo quando houve necessidade de ramificar a árvore, o número de nós manteve-se pequeno. A heurística primal que utilizamos também se mostrou boa, fornecendo resultados próximos da solução ótima.

A Tabela 6.4 fornece os resultados que obtivemos. Na tabela, n é o número de vértices da instância, m o número de arestas e $|\mathcal{R}|$ o número de grupos. Fornecemos também o limitante inferior global obtido, $\underline{z}$, e o limitante superior global, $\bar{z}$. Quando ambos são iguais, obtivemos uma solução inteira ótima. Em seguida, fornecemos o número de nós da árvore de decisão, o número de cortes gerados e o maior número de cortes a compor o programa linear em qualquer momento da execução. O tempo fornecido corresponde apenas ao tempo de CPU do programa (i.e., o *user time*). Para algumas instâncias, aquelas marcadas com uma estrela (*), não foi possível encontrar uma solução ótima. Nesses casos, o programa foi interrompido após 4 horas de execução.

Os testes foram executados num Pentium III 1.2GHz com 2 gigabytes de memória RAM.

Instância	n	m	$ \mathcal{R} $	z	$\bar{z}$	# de nós	# de cortes	Máx. cortes	Tempo
wrp4-11	112	192	11	179	179	1	1 618	880	2,87s
wrp4-13	97	147	13	798	798	1	1 327	771	0,72s
wrp4-14	131	231	14	290	290	1	2 238	938	7,71s
wrp4-15	178	312	15	405	405	1	3 193	1 214	10,50s
wrp4-17	206	340	17	525	525	1	4 746	1 435	41,96s
wrp4-18	193	327	18	1 464	1 464	1	4 276	1 680	28,33s
wrp4-19	100	155	19	1 446	1 446	1	2 867	1 099	3,97s
wrp4-22	272	482	22	394	394	1	9 215	2 320	7m23,49s
wrp4-23	234	424	23	376	376	3	5 371	1 658	53,36s
wrp4-24	469	843	24	3 332	3 332	1	21 295	3 610	16m06,72s
wrp4-25	397	720	25	828	828	3	36 403	5 594	59m06,09s
wrp4-26	370	677	26	443	443	7	13 399	3 519	27m42,32s
wrp4-27	216	390	27	441	441	1	6 253	1 973	45,47s
wrp4-28	244	435	28	466	466	1	6 208	2 131	1m29,12s
wrp4-29	218	391	29	484	484	3	4 806	2 054	27,42s
wrp4-30	331	606	30	526	526	15	14 720	4 386	23m18,01s
wrp4-31	359	662	31	526	526	7	20 941	3 533	43m33,88s
wrp4-32	279	502	32	554	554	7	11 523	2 525	8m22,27s
wrp4-33	271	474	33	655	655	1	8 882	3 072	2m36,71s
wrp4-34	280	514	34	525	525	1	8 217	2 622	2m23,31s
wrp4-36	327	607	36	596	596	1	9 680	2 744	5m15,43s
wrp4-37	485	906	37	647	647	1	21 349	3 892	1h18m21,10s
wrp4-38	256	468	38	606	606	1	6 283	2 387	44,27s
wrp4-39	763	1 353	39	3 734	3 734	3	50 733	7 669	1h46m09,05s
wrp4-41	424	791	41	695	695	3	21 326	3 936	40m21,40s
wrp4-42	510	963	42	701	701	7	32 822	4 803	2h55m17,02s
wrp4-43	553	978	43	1 508	1 508	7	21 475	5 704	22m10,60s
wrp4-44	354	614	44	1 504	1 504	423	25 525	7 689	1h16m29,48s
wrp4-45	343	635	45	728	728	1	14 080	3 371	8m38,90s
wrp4-46	586	1 104	46	756	756	1	23 580	4 601	1h19m56,40s
wrp4-47	508	912	47	1 318	1 318	1	14 336	3 534	10m47,17s
wrp4-48	403	685	48	2 220	2 220	3	12 099	3 474	3m20,38s
wrp4-49	508	885	49	1 968	1 968	3	23 092	3 851	22m11,83s
wrp4-50	514	914	50	1 625	1 625	1	35 031	5 152	1h04m16,80s
wrp4-51	617	1 104	51	1 616	1 616	3	32 041	5 379	45m28,67s
wrp4-52	495	907	52	1 081	1 081	1	34 281	5 928	30m09,30s
wrp4-54	634	1 173	54	1 534	1 534	7	32 351	5 989	1h28m34,31s
wrp4-55	555	979	55	1 952	1 952	7	24 315	4 467	42m44,38s
wrp4-56	783	1 379	56	2 299	2 299	1	38 475	5 341	2h11m21,77s
wrp4-58	699	1 260	58	1 466	1 466	1	28 190	4 199	2h46m02,75s
* wrp4-59	845	1567	59	1 591,89	1 694	18	40 403	8 137	—
wrp4-60	633	1 132	60	1 782	1 782	1	24 270	5 831	1h13m02,53s
wrp4-61	714	1 292	61	2 210	2 210	1	28 880	6 774	1h01m26,23s
* wrp4-62	1 221	2 212	62	2 049,25	2 215	1	40 210	7 887	—
* wrp4-63	1 058	1 977	63	1 234,45	1 634	1	29 067	6 367	—
wrp4-64	568	1 024	64	1 996	1 996	3	20 120	5 283	14m33,37s
wrp4-66	778	1 411	66	2 931	2 931	1	56 538	6 784	2h57m22,93s
* wrp4-67	1 451	2 684	67	2 576,26	2 906	1	57 141	9 382	—
wrp4-68	849	1 577	68	1 753	1 753	1	37 271	13 833	—
wrp4-69	505	891	69	2 328	2 328	3	25 213	3 902	31m15,80s
wrp4-70	567	991	70	3 022	3 022	1	18 347	4 806	5m48,33s
wrp4-71	731	1 324	71	2 320	2 320	3	34 689	6 593	1h03m03,26s
* wrp4-72	1 079	1 979	72	2 693,05	3 025	1	50 574	6 661	—
* wrp4-73	1 825	3 313	73	2 338,98	2 802	1	49 364	16 957	—
wrp4-74	728	1 325	74	2 046	2 046	1	25 664	7 529	1h08m23,60s
wrp4-75	863	1 570	75	1 712	1 712	7	30 883	9 955	1h17m14,26s
wrp4-76	690	1 230	76	2 040	2 040	15	24 453	5 661	34m42,96s

Tabela 6.4

Conclusão

Nesse trabalho fizemos um estudo do problema de Steiner com grupos: estudamos sua complexidade computacional, algoritmos de aproximação para a obtenção de boas soluções, técnicas de redução do tamanho das instâncias tratadas, formulações baseadas em programação linear inteira para o problema e, finalmente, desenvolvemos um algoritmo baseado na estratégia *branch-and-cut* para resolvê-lo.

Os resultados obtidos deram-se em vários dos temas estudados:

- desenvolvemos um algoritmo de aproximação de razão N para o problema restrito a árvores, sendo N o tamanho do maior grupo;
- estudamos as técnicas de redução de Duin e Volgenant [9], Polzin e Daneshmand [36] e Rohe e Zachariasen [39] para o problema de Steiner em grafos e o problema de Steiner com grupos, obtendo resultados que foram descritos no artigo [14];
- estudamos formulações para o problema utilizando programação linear inteira. Os resultados encontram-se no artigo [15];
- estudamos um poliedro relacionado ao problema e obtivemos resultados a respeito de propriedades desse poliedro, como dimensão, adjacência, facetas, entre outras;
- implementamos um algoritmo baseado no método *branch-and-cut* para atacar instâncias do problema presentes na Steinlib [34]. Os resultados computacionais foram muito bons.

Alguns dos resultados obtidos deram origem a perguntas que não exploramos nesse trabalho, mas que poderiam ser tratadas em trabalhos futuros:

- qual seria o comportamento empírico dos algoritmos de aproximação descritos?
- como utilizar os resultados sobre adjacência de vértices de P_{STG} numa heurística de busca local para o problema?
- como obter um algoritmo combinatório para resolver o problema de programação linear (3.6)?
- que outras classes de facetas existem para P_{STG}^I ? Como obtê-las e separá-las eficientemente?
- é sempre verdade que $P_{\text{part}} \subseteq P_{\text{vert}}^E$?

Usualmente, o problema de Steiner com grupos é atacado usando-se a redução apresentada na seção 2.3.2. Nosso algoritmo não usa tal redução. Uma comparação de nossa abordagem com as demais presentes na literatura também é um interessante trabalho a ser desenvolvido.

Referências Bibliográficas

- [1] N. Alon e J. Spencer. *The Probabilistic Method*. Wiley Interscience, 1992.
- [2] Y. Bartal. On approximating arbitrary metrics by tree metrics. In *Proceedings of the 30th Annual ACM Symposium on Theory of Computing*, p. 161–168, 1998.
- [3] S. Chopra e M. R. Rao. The Steiner tree problem I: Formulations, compositions and extension of facets. *Mathematical Programming*, 64:209–229, 1994.
- [4] S. Chopra e M. R. Rao. The Steiner tree problem II: Properties and classes of facets. *Mathematical Programming*, 64:231–246, 1994.
- [5] E. Dahlhaus, D. S. Johnson, C. H. Papadimitriou, P. D. Seymour e M. Yannakakis. The complexity of multiterminal cuts. *SIAM Journal on Computing*, 23:864–894, 1994.
- [6] R. Diestel. *Graph Theory*. Springer, New York, 1997.
- [7] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959.
- [8] C. W. Duin. *Steiner's Problem in Graphs*. Tese de doutorado, Universiteit van Amsterdam, 1994.
- [9] C. W. Duin e A. Volgenant. Reduction tests for the Steiner problem in graphs. *Networks*, 19:549–567, 1989.
- [10] C. W. Duin, A. Volgenant e S. Voß. Solving group Steiner problems as Steiner problems. *European Journal of Operational Research*, 154:323–329, 2004.
- [11] J. Fakcharoenphol, S. Rao e K. Talwar. A tight bound on approximating arbitrary metrics by tree metrics. *FOCS*, 2003.
- [12] U. Feige. A threshold of $\ln n$ for approximating set cover. *Journal of the ACM*, 45:634–652, 1998.
- [13] C. G. Fernandes, F. K. Miyazawa, M. C. Cerioli e P. Feofiloff, editores. *Uma Introdução Sucinta a Algoritmos de Aproximação*. 23^o Colóquio Brasileiro de Matemática, IMPA, 2001.

- [14] C. E. Ferreira e F. M. de Oliveira Filho. New reduction techniques for the group Steiner tree problem. Submetido, 2004.
- [15] C. E. Ferreira e F. M. de Oliveira Filho. Some formulations for the group Steiner tree problem. A ser publicado em *Discrete Applied Mathematics*, 2004.
- [16] C. E. Ferreira e Y. Wakabayashi. *Combinatória Poliédrica e Planos-de-Corte Faciais*. X Escola de Computação, IC-UNICAMP, 1996.
- [17] L. R. Ford Jr. e D. R. Fulkerson. Maximal flow through a network. *Canadian Journal of Mathematics*, 8:399–404, 1956.
- [18] M. R. Garey e D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, San Francisco, 1979.
- [19] N. Garg, G. Konjevod e R. Ravi. A polylogarithmic approximation algorithm for the group Steiner tree problem. *Journal of Algorithms*, 37(1):66–84, 2000.
- [20] M. X. Goemans. The Steiner tree polytope and related polyhedra. *Mathematical Programming*, 63:157–182, 1994.
- [21] M. X. Goemans e Y. S. Myung. A catalog of Steiner tree formulations. *Networks*, 23:19–28, 1993.
- [22] A. V. Goldberg e R. E. Tarjan. A new approach to the maximum-flow problem. *Journal of the Association for Computing Machinery*, 35:921–940, 1988.
- [23] M. Grötschel, L. Lovász e A. Schrijver. *Geometric Algorithms and Combinatorial Optimization*. Springer-Verlag, 1988.
- [24] M. Grötschel e C. L. Monma. Integer polyhedra associated with certain network design problems with connectivity constraints. *SIAM Journal on Discrete Mathematics*, 3:502–523, 1990.
- [25] M. Grötschel, C. L. Monma e M. Stoer. Facets for polyhedra arising in the design of communication networks with low connectivity constraints. *SIAM Journal on Optimization*, 2:125–128, 1992.
- [26] E. Halperin e R. Krauthgamer. Polylogarithmic inapproximability. In *STOC '03: Proceedings of the thirty-fifth annual ACM symposium on Theory of computing*, p. 585–594. ACM Press, 2003.
- [27] C. H. Helving, R. Robins e A. Zelikovsky. An improved approximation scheme for the group Steiner tree problem. *Networks*, 37(1):8–20, 2001.
- [28] D. S. Hochbaum. Approximation algorithms for the set-covering and vertex cover problems. *SIAM Journal on Computing*, 15(1):35–40, 1982.

- [29] F. K. Hwang, D. S. Richards e P. Winter. *The Steiner Tree Problem*, volume 53 of *Annals of Discrete Mathematics*. North-Holland, 1992.
- [30] E. Ihler. Bounds on the quality of approximate solutions to the group Steiner problem. In *Graph-Theoretic Concepts in Computer Science WG90*, volume 484 of *Lecture Notes in Computer Science*, p. 109–118. Springer, 1991.
- [31] E. Ihler. The complexity of approximating the class Steiner tree problem. In *Graph-Theoretic Concepts in Computer Science WG91*, volume 570 of *Lecture Notes in Computer Science*, p. 85–96. Springer, 1992.
- [32] S. Janson. Poisson approximations for large deviations. *Random Structures and Applications*, 1:221–230, 1990.
- [33] R. M. Karp. Reducibility among combinatorial problems. In Miller e Thatcher, editores, *Complexity of Computer Computations*, p. 85–103. Plenum Press, New York, 1972.
- [34] T. Koch, A. Martin e S. Voß. SteinLib: An updated library on Steiner tree problems in graphs. Relatório Técnico ZIB-Report 00-37, Konrad-Zuse-Zentrum für Informationstechnik Berlin, Takustr. 7, Berlin, 2000. URL: <http://elib.zib.de/steinlib>.
- [35] C. Lund e M. Yannakakis. On the hardness of approximating minimization problems. *Journal of the ACM*, 41(5):960–981, 1994.
- [36] T. Polzin e S. V. Daneshmand. Extending reduction techniques for the Steiner tree problem: A combination of alternative and bound-based approaches. Relatório Técnico MPI-I-2001-1-007, Max-Planck-Institut für Informatik, 2001.
- [37] P. Raghavan e C. D. Thompson. Randomized rounding. *Combinatorica*, 7:365–374, 1987.
- [38] G. Reich e P. Widmayer. Beyond Steiner’s problem: A VLSI oriented generalization. In *Graph-Theoretic Concepts in Computer Science WG89*, volume 411 of *Lecture Notes in Computer Science*, p. 196–210. Springer, 1990.
- [39] A. Rohe e M. Zachariasen. Rectilinear group Steiner trees and applications in VLSI design. *Mathematical Programming*, 94(2–3):407–433, 2003.
- [40] S. M. Ross. *Introduction to Probability Models*. Academic Press, 1997.
- [41] J. J. Salazar. A note on the generalized Steiner tree polytope. *Discrete Applied Mathematics*, 100:137–144, 2000.
- [42] A. Schrijver. *Theory of Linear and Integer Programming*. Wiley Interscience, 1986.
- [43] A. Schrijver. *Combinatorial Optimization — Volume A*. Springer-Verlag, 2003.
- [44] P. Slavík. *Approximation Algorithms for Set Cover and Related Problems*. Tese de doutorado, University of New York at Buffalo, 1998.

- [45] J. Stolfi, 2004. Comunicação pessoal.
- [46] H. Takahashi e A. Matsuyama. An approximate solution for the Steiner problem in graphs. *Math. Japonica*, 24:573–577, 1980.
- [47] E. Uchoa, M. de Aragão e C. Ribeiro. Preprocessing Steiner problems from VLSI layout. *Networks*, 40:38–50, 2002.