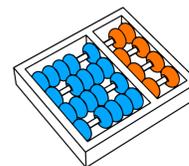


Luiz Fernando de Faria Pereira

“Partições de Digrafos em Caminhos”

CAMPINAS
2013



Universidade Estadual de Campinas
Instituto de Computação

Luiz Fernando de Faria Pereira

“Partições de Digrafos em Caminhos”

Orientador(a): **Prof. Dr. Orlando Lee**

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Computação da Universidade Estadual de Campinas para obtenção do título de Mestre em Ciência da Computação.

ESTE EXEMPLAR CORRESPONDE À VERSÃO FINAL DA DISSERTAÇÃO DEFENDIDA POR LUIZ FERNANDO DE FARIA PEREIRA, SOB ORIENTAÇÃO DE PROF. DR. ORLANDO LEE.

Assinatura do Orientador(a)

CAMPINAS

2013

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca do Instituto de Matemática, Estatística e Computação Científica
Ana Regina Machado - CRB 8/5467

P414p Pereira, Luiz Fernando de Faria, 1986-
Partições de digrafos em caminhos / Luiz Fernando de Faria Pereira. –
Campinas, SP : [s.n.], 2013.

Orientador: Orlando Lee.
Dissertação (mestrado) – Universidade Estadual de Campinas, Instituto de
Computação.

1. Teoria dos grafos. 2. Berge, Conjectura de. I. Lee, Orlando, 1969-. II.
Universidade Estadual de Campinas. Instituto de Computação. III. Título.

Informações para Biblioteca Digital

Título em outro idioma: Path partitions in digraphs

Palavras-chave em inglês:

Graph theory

Berge conjecture

Área de concentração: Ciência da Computação

Titulação: Mestre em Ciência da Computação

Banca examinadora:

Orlando Lee [Orientador]

Daniel Morgato Martin

Eduardo Cândido Xavier

Data de defesa: 10-06-2013

Programa de Pós-Graduação: Ciência da Computação

TERMO DE APROVAÇÃO

Dissertação Defendida e Aprovada em 10 de Junho de 2013, pela
Banca examinadora composta pelos Professores Doutores:



Prof. Dr. Daniel Morgato Martin
CMCC / UFABC



Prof. Dr. Eduardo Cândido Xavier
IC / UNICAMP



Prof. Dr. Orlando Lee
IC / UNICAMP

Partições de Digrafos em Caminhos

Luiz Fernando de Faria Pereira¹

10 de junho de 2013

Banca Examinadora:

- Prof. Dr. Orlando Lee (Orientador)
- Prof. Dr. Daniel Morgato Martin (Membro externo)
CMCC / UFABC
- Prof. Dr. Eduardo Cândido Xavier (Membro interno)
IC / Unicamp
- Prof. Dr. Cid Carvalho de Souza (Suplente interno)
IC / Unicamp
- Prof. Dr. José Coelho da Pina Jr. (Suplente externo)
IME / USP

¹Suporte financeiro do CNPq (2008-2009).

Resumo

Uma partição em caminhos de um grafo dirigido é uma partição do conjunto de vértices deste grafo em caminhos dirigidos. Dada uma métrica sobre partições em caminhos chamada k -norma, o problema de interesse é estabelecer para um dado grafo quais das suas partições em caminhos têm a menor k -norma dentre todas as suas possíveis partições em caminhos. Chamamos estas partições de k -ótimas. Na década de 1980, Claude Berge conjecturou que para toda partição k -ótima, existe um conjunto de k conjuntos independentes disjuntos que, em um certo sentido, interceptam o maior número possível de caminhos desta partição. A validade ou a falsidade desta proposição ainda não foi demonstrada, e ela é conhecida como a conjectura de Berge sobre partições em caminhos.

Nesta dissertação, fizemos um estudo geral sobre a conjectura de Berge, sua história recente, e o trabalho matemático que foi desenvolvido sobre ela. Exibimos demonstrações para diversos casos particulares da conjectura que já foram resolvidos, como para grafos bipartidos, hamiltonianos, acíclicos, partições compostas somente de caminhos curtos, partições compostas somente de caminhos longos, e para valores fixos de k . Uma parte significativa do trabalho foi dedicada à reescrita da demonstração recente do caso particular onde $k = 2$, feita por Eli Berger e Irith Hartman, e uma análise do método usado.

Abstract

A path partition of a directed graph is a partition of its vertex set into directed paths. Given a metric over path partitions called the k -norm, the problem we are interested in is to determine for a given graph which of its path partitions have the smallest k -norm among all possible path partitions. These partitions are called k -optimal. In the 1980's, Claude Berge conjectured that for every k -optimal path partition, there exists a set of k disjoint independent sets which intercepts the maximum number of paths in this partition. The validity of this proposition has not yet been demonstrated, and it is known as Berge's conjecture on path partitions.

In this work, we consider Berge's conjecture, its recent history, and the related mathematical work that has been accomplished. We show proofs for many particular cases of the conjecture, including for acyclic graphs, bipartite graphs, hamiltonian graphs, partitions which include only short paths, partitions which include only long paths, and for fixed values of k . A significant part of this work was dedicated to the rewriting of a recent proof for the particular case where $k = 2$ by Eli Berger and Irith Hartman, and an analysis of their method.

Agradecimentos

Agradeço ao meu orientador Orlando Lee pela paciência, pelas opiniões sensatas e pela leitura cuidadosa e presente das versões deste trabalho.

Agradeço à professora Christiane Neme Campos e ao professor Flávio Keidi Miyazawa por sugestões valiosas dadas durante a apresentação da proposta. Agradeço especialmente à professora Célia Picinin de Mello por ter gentilmente me orientado e aconselhado durante o período de ausência do meu orientador.

Agradeço à professora Irith Ben-Arroyo Hartman, pela presteza durante a correspondência sobre o seu artigo para o caso $k = 2$.

Agradeço também ao CNPq pela ajuda financeira concedida para a realização deste trabalho.

Agradecimentos especiais vão para minha família e meus amigos, pela boa companhia, e pelo suporte dado em momentos de dificuldade.

Sumário

Resumo	ix
Abstract	xi
Agradecimentos	xiii
1 Partições em Caminhos	1
1.1 Notações	1
1.2 Introdução histórica	3
1.2.1 1950 - 1975: Ordens parciais, caminhos e conjuntos independentes	4
1.2.2 1976 - 1982: Algumas generalizações	5
1.2.3 1982 - 2013: Da conjectura de Berge em diante	9
1.3 Teorema de Gallai-Milgram	10
1.4 Complexidade computacional	11
1.5 Colorações e coleções de caminhos	12
2 A Conjectura de Berge	16
2.1 Casos particulares da conjectura	16
2.1.1 Partições sem caminhos longos	16
2.1.2 Grafos com um caminho hamiltoniano	17
2.1.3 Grafos bipartidos	17
2.2 Grafos acíclicos	19
2.2.1 Definição da Rede	19
2.2.2 Rede residual	20
2.2.3 Fluxos e partições	21
2.2.4 Demonstração da conjectura de Berge para grafos acíclicos.	22
3 Uma abordagem construtiva para a conjectura de Berge.	30
3.1 Visão geral da abordagem	30
3.1.1 Trilhas alternantes	31

3.2	O caso $k = 1$.	33
3.2.1	Descrição do algoritmo	33
3.2.2	Prova de correção do algoritmo	35
3.3	O caso das partições longas	38
3.3.1	Descrição do algoritmo	38
3.3.2	Prova de correção do algoritmo	41
4	O Caso $k = 2$.	47
4.1	Introdução	47
4.2	Definições	47
4.2.1	Predecessor	48
4.2.2	Trilha alternante intermediária $Q(v)$	48
4.2.3	Conjunto examinado Y e o conjunto candidato X	49
4.2.4	Remorso	50
4.2.5	Conjunto $W(v)$	51
4.3	Descrição do algoritmo	51
4.4	Análise dos casos de atualização	57
4.5	Invariantes simples do algoritmo	57
4.6	Invariantes principais do algoritmo	59
4.6.1	Tipos de trilhas alternantes	60
4.7	Casos de melhora	63
4.8	Modificações com relação ao artigo original	66
5	Considerações Finais	68
	Referências Bibliográficas	72

Capítulo 1

Partições em Caminhos

Neste capítulo apresentamos algumas definições e notações, e descrevemos o problema que estudamos neste trabalho.

A primeira seção deste capítulo lida somente com notações, e o leitor interessado que já tenha familiaridade com a área pode passar adiante à introdução propriamente dita, consultando esta seção conforme julgar necessário. Já o leitor leigo em teoria dos grafos pode consultar o excelente livro de Bondy e Murty [1] se a notação lhe parecer demasiadamente estéril.

1.1 Notações

Em primeiro lugar, estabelecemos alguns conceitos e notações que serão usadas ao longo deste trabalho. Um *grafo simples* ou *grafo* $G = (V, E)$ é um par ordenado que consiste de um conjunto finito não-vazio V de *vértices* e um conjunto E de pares (distintos) de elementos distintos de V chamados *arestas*. O conjunto de vértices de G é denotado por $V(G)$ e o seu conjunto de arestas por $E(G)$. Se as arestas constituem pares ordenados, trata-se de um grafo *orientado* ou *digrafo*. Senão, estamos lidando com um grafo *não-orientado*. Grafos orientados são o objeto principal de estudo deste trabalho, então ao usarmos a palavra “grafo” neste texto, sem qualquer outra especificação, estamos nos referindo a grafos orientados — enquanto ao tratarmos de grafos não-orientados, a referência será feita de forma explícita.

Se $e = (u, v)$ é uma aresta de um grafo, dizemos que e *incide* em u e v . Mais especificamente, dizemos que e é uma aresta *saindo* de u e *entrando* em v . Para um dado vértice $v \in V(G)$, dizemos que seu *grau de entrada* $d_G^-(v)$ é o número de arestas entrando em v e seu *grau de saída* $d_G^+(v)$ é o número de arestas saindo de v . Se o grafo G está claro pelo contexto, podemos usar a notação $d^-(v)$ e $d^+(v)$ para denotar grau de entrada e grau de saída de v , respectivamente.

Para um grafo $G = (V, E)$, um *subgrafo* de G é um grafo H onde $V(H) \subseteq V(G)$ e $E(H) \subseteq E(G)$. Um *subgrafo gerador* H de G é um subgrafo de G com $V(H) = V(G)$. Dizemos que H é um *subgrafo induzido* de G se $V(H) \subset V(G)$, e H contém todas as arestas $(u, v) \in E(G)$ tais que $u, v \in V(H)$.

Um *passeio* em G é uma sequência de vértices e arestas $P = (v_1, e_1, v_2, e_2, v_3, \dots, e_{k-1}, v_k)$ tal que $v_i \in V(G)$, e $e_i = (v_i, v_{i+1})$ é uma aresta de G para $i = 1, 2, \dots, k-1$. Muitas vezes interpretamos P como um subgrafo de G com conjunto de arestas $E(P)$ e conjunto de vértices $V(P)$. Uma *trilha* é um passeio P onde todas as arestas $e_i \in P$ são distintas. Um *caminho* é um passeio P onde as arestas $e_i \in P$ e os vértices $v_i \in P$ são distintos. Claramente, todo caminho é uma trilha. Podemos descrever um passeio somente por sua sequência de vértices $P = (v_1, v_2, \dots, v_k)$, já que pela nossa definição de grafo as arestas percorridas por um passeio são inequivocamente determinadas por essa sequência de vértices.

Um *circuito* C é uma trilha $C = (v_0, v_1, v_2, \dots, v_l)$ tal que $v_0 = v_l$, $l > 0$, e para qualquer outro par de vértices v_i, v_k tal que $0 \leq i < k \leq l$, temos que $v_i \neq v_k$. Denotamos como *acíclico* um grafo que não contenha circuitos.

A *cardinalidade* de P é definida como $|P| = |V(P)|$. Se $|P| = 1$, P é um passeio *trivial*. Note que a definição de cardinalidade dada aqui difere da definição convencional dada na literatura de teoria dos grafos (onde a cardinalidade de um passeio é definida pelo número de arestas deste).

Para um passeio $P = (v_1, v_2, \dots, v_l)$ em G , denotamos o primeiro vértice (ou *inicial*) de P por $in(P)$ e o último vértice (ou *terminal*) de P por $ter(P)$. Para um passeio P e $v_i \in V(P)$, se $i \geq 2$, denotamos o vértice v_{i-1} como o *antecessor* de v_i em P . Similarmente, se $i < l$, denotamos o vértice v_{i+1} como o *sucessor* de v_i em P . Também usamos a seguinte notação: para um vértice $x \in P$, x_P^- denota seu antecessor, e x_P^+ seu sucessor, se esses existirem. Quando o caminho P está claro pelo contexto, podemos nos referir a x_P^- e x_P^+ como x^- e x^+ respectivamente.

Para passeios $P = (v_1, v_2, \dots, v_s)$ e $Q = (v_s, v_{s+1}, \dots, v_l)$, denominamos como $P * Q = (v_1, \dots, v_s, \dots, v_l)$ a concatenação de P e Q , desde que o último vértice de P e o primeiro vértice de Q coincidam. Para um passeio $P = (v_1, v_2, \dots, v_l)$ e $1 \leq i \leq l$, definimos os passeios $v_i P$ e $P v_i$ como $v_i P = (v_i, v_{i+1}, \dots, v_l)$, e $P v_i = (v_1, v_2, \dots, v_i)$. Todas as definições feitas para passeios também valem para trilhas e caminhos.

Para um circuito C em um grafo G , C é um *circuito hamiltoniano* se $V(G) = V(C)$. Igualmente, chamamos um caminho P de G de *caminho hamiltoniano* se $V(G) = V(P)$. Se um grafo G contém um circuito hamiltoniano, dizemos que G é um *grafo hamiltoniano*.

Um conjunto $I \subseteq V(G)$ é *independente* em G se para todo $u, v \in I$, temos que $(u, v) \notin E(G)$ e $(v, u) \notin E(G)$. Denotamos por $\alpha(G)$ o tamanho do maior conjunto independente em G .

Uma *coloração* $\mathcal{I} = \{I_1, I_2, \dots, I_k\}$ de G é uma partição do conjunto de vértices de G em conjuntos independentes disjuntos entre si. Um grafo é *k-colorível* se ele possui uma coloração de cardinalidade k . A cardinalidade da menor coloração de G é chamada de *número cromático* de G e é denotada por $\chi(G)$. Um grafo G é *bipartido* se $\chi(G) \leq 2$.

Desde que não ocorra ambiguidade, para um conjunto A e um elemento b , podemos usar a notação $A + b$ e $A - b$ para denotar $A \cup \{b\}$ e $A \setminus \{b\}$, respectivamente. Usaremos frequentemente esta convenção na descrição dos algoritmos que aparecem neste trabalho.

1.2 Introdução histórica

Uma *partição em caminhos* (usaremos frequentemente a forma abreviada, *partição*) de G é um conjunto \mathcal{P} de caminhos $P \in \mathcal{P}$ tal que $V(G) = \bigcup_{P \in \mathcal{P}} V(P)$ e os caminhos são disjuntos entre si em relação aos vértices. Podemos estabelecer uma definição alternativa, onde uma partição em caminhos \mathcal{P} é descrita como um subgrafo gerador acíclico de G , onde para todo vértice $v \in V(G)$, $d_{\mathcal{P}}^-(v) \leq 1$ e $d_{\mathcal{P}}^+(v) \leq 1$. É fácil estabelecer uma relação de equivalência entre as duas definições, e podemos usar qualquer uma das duas interpretações conforme o contexto, por conveniência.

Para uma partição em caminhos, definimos $V[\mathcal{P}] = \bigcup\{V(P) : P \in \mathcal{P}\}$ e $E[\mathcal{P}] = \bigcup\{E(P) : P \in \mathcal{P}\}$ como os conjuntos de vértices e de arestas dessa partição. Pela definição de partição em caminhos, claramente $V[\mathcal{P}] = V(G)$. Também definimos $in[\mathcal{P}] = \bigcup\{in(P) : P \in \mathcal{P}\}$, e $ter[\mathcal{P}] = \bigcup\{ter(P) : P \in \mathcal{P}\}$ como os conjuntos dos vértices iniciais e terminais de \mathcal{P} . Por conveniência, definimos também $in^+[\mathcal{P}] = \bigcup\{v \in \mathcal{P} : v^- \in in[\mathcal{P}]\}$ e $ter^-[\mathcal{P}] = \bigcup\{v \in \mathcal{P} : v^+ \in ter[\mathcal{P}]\}$ como os conjuntos dos “segundos” e dos “penúltimos” vértices dos caminhos de \mathcal{P} , respectivamente.

Diversas partições em caminhos distintas podem ser encontradas para um grafo em particular. O problema combinatório de encontrar a menor partição em caminhos para um grafo consiste em tentar encontrar alguma partição, dentre as partições deste grafo com o menor número de caminhos. Denominamos essa partição como *ótima*, e denotamos por $\pi(G)$ a cardinalidade (equivalente ao número de caminhos) desta partição.

A figura 1.1 ilustra um exemplo de uma partição em caminhos. A partição \mathcal{P} deste exemplo não é ótima. Uma outra partição em caminhos $\mathcal{P}' = \{(b, e, h, f, c, a), (g, d)\}$ de G tem um caminho a menos que \mathcal{P} . É fácil mostrar que \mathcal{P}' é ótima, e que $\pi(G) = 2$.

Para qualquer grafo G , existe a partição trivial de tamanho $|V(G)|$ onde cada caminho tem cardinalidade de valor 1, e é formado por um único vértice do grafo. Se $E(G) \neq \emptyset$, é possível encontrar outras partições de G com um número menor de caminhos. A menor partição que um grafo pode ter é o conjunto que contém um único caminho que passa por todos os vértices de G . Claramente, essa partição em caminhos existe se, e só se G tem um caminho hamiltoniano. Podemos, portanto, considerar o problema de encontrar

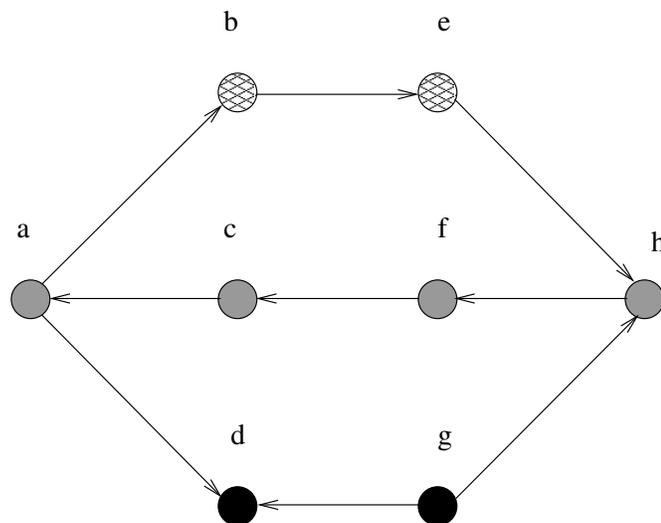


Figura 1.1: Um grafo G , e uma partição em caminhos $\mathcal{P} = \{(b, e), (h, f, c, a), (g, d)\}$ de G .

a partição em caminhos ótima de um grafo como uma generalização sobre o problema de encontrar um caminho hamiltoniano em um dado grafo.

Encontrar uma partição em caminhos ótima de um grafo é um problema computacional de difícil tratamento (provaremos adiante que trata-se de um problema NP-difícil) e não temos pretensões de tratar desse problema em profundidade neste texto. Em vez de abordarmos algoritmos ou soluções computacionais eficientes para o problema, nosso foco de discussão estará na relação entre partições em caminhos e conjuntos independentes, e em alguns problemas em aberto que circundam essa relação.

Segue adiante uma breve introdução à história da pesquisa nessa área:

1.2.1 1950 - 1975: Ordens parciais, caminhos e conjuntos independentes

Três teoremas das décadas de 50 e 60 servem como base para a relação entre caminhos e conjuntos independentes. O primeiro deles é um teorema famoso vindo da teoria de ordens parciais. A linguagem de grafos se mostra apropriada para representar ordens parciais finitas, e podemos representá-las sem perda de generalidade como grafos orientados acíclicos e transitivos. Um dado grafo G é *transitivo* se ele mantém a seguinte propriedade para o seu conjunto de arestas: se $(u, v) \in E(G)$ e $(v, w) \in E(G)$ então, necessariamente, $(u, w) \in E(G)$.

Em 1950, Robert P. Dilworth [10] demonstrou que o tamanho de um conjunto in-

dependente máximo é igual ao tamanho de uma partição mínima em caminhos de um conjunto parcialmente ordenado. Ou seja:

Teorema 1.2.1. (*Dilworth, 1950*)

Para todo grafo orientado transitivo e acíclico G , $\alpha(G) = \pi(G)$.

O segundo teorema pode ser concebido como uma generalização do teorema de Dilworth, e foi proposto e demonstrado por Tibor Gallai e Arthur Milgram em 1960. Ele relaciona conjuntos independentes e partições em caminhos em grafos orientados genéricos.

Teorema 1.2.2. (*Gallai-Milgram, 1960*)

Para todo grafo orientado G , $\alpha(G) \geq \pi(G)$.

O terceiro teorema foi demonstrado nos anos 60 independentemente por quatro autores diferentes: Vitaver [13] em russo, Hasse [14] em alemão, Roy [15] em francês e Gallai [16] em inglês. O teorema é hoje mais conhecido como Teorema de Gallai-Roy — possivelmente pelo fato dos autores ocidentais terem desfrutado de melhor reconhecimento naquela época.

O teorema relaciona o número cromático de um grafo com a cardinalidade de seu caminho mais longo. Seja $\lambda(G)$ o número de vértices do mais longo caminho orientado dentro de um grafo G .

Teorema 1.2.3. (*Gallai-Roy, 1967*)

Para todo grafo orientado G , $\lambda(G) \geq \chi(G)$.

É interessante notar que, enquanto no teorema de Gallai-Milgram é analisada a relação entre a menor partição em caminhos e o maior conjunto independente de um grafo, o teorema de Gallai-Roy explicita a relação entre a menor partição em conjuntos independentes e o maior caminho de um grafo. Essa dualidade entre as duas relações de grandeza também foi explorada; enquanto um corpo teórico surgiu do teorema de Gallai-Milgram, um outro conjunto de problemas teve origem do teorema de Gallai-Roy. Esse conjunto paralelo de problemas (conhecido como os “duais”) resultou em um conjunto de resultados bastante rico, e faremos uma exposição breve sobre eles no final deste capítulo.

1.2.2 1976 - 1982: Algumas generalizações

Em 1976, Curtis Greene e Daniel Kleitman publicaram um artigo contendo uma generalização importante do teorema de Dilworth. Ao invés de comparar conjuntos independentes e caminhos¹, o artigo de Greene e Kleitman tece a relação entre caminhos e certas famílias de conjuntos independentes chamadas colorações parciais.

¹Na verdade, os artigos originais de Dilworth e Greene-Kleitman lidam com conjuntos parcialmente ordenados e usam a notação de “chains” e “antichains”, que são equivalentes, respectivamente, a caminhos e conjuntos independentes na representação matemática de teoria dos grafos.

Uma k -coloração parcial $C^k = \{I_1, I_2, \dots, I_k\}$ é uma coleção de k conjuntos independentes disjuntos entre si. O *peso* $\|C^k\|$ de uma k -coloração parcial C^k é definido como a soma das cardinalidades de cada conjunto independente de C^k . Podemos considerar o problema de maximização onde queremos encontrar a maior coloração parcial possível (segundo o seu peso) de um grafo em particular. Chamamos esta coloração parcial de *ótima* e denotamos como $\alpha_k(G)$ o peso da k -coloração parcial ótima do grafo G . Claramente, $\alpha_1(G) = \alpha(G)$ para qualquer grafo G .

Uma coloração parcial C^k de um grafo G tal que $\|C^k\| = |V(G)|$ também é uma coloração de G (uma partição do conjunto de vértices de G em conjuntos independentes). Na figura 1.2, é dado um exemplo de coloração parcial. A 3-coloração parcial $C^3 = \{\{a, e\}, \{b, c\}, \{d\}\}$ de G também é uma coloração de G .

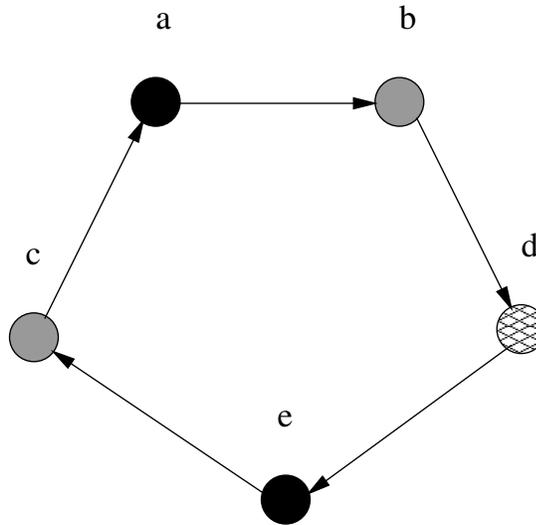


Figura 1.2: Um grafo G , e uma 3-coloração parcial $C^3 = \{\{a, e\}, \{b, c\}, \{d\}\}$ de G .

Definimos também uma métrica distinta para partições em caminhos. A k -norma de uma partição em caminhos é um valor inteiro associado a ela. Denotamos como $|\mathcal{P}|_k$ a k -norma de uma partição \mathcal{P} , calculada por $|\mathcal{P}|_k = \sum_{P \in \mathcal{P}} \min\{|P|, k\}$.

Uma maneira alternativa de calcularmos a k -norma é através da contagem de caminhos longos e curtos: para uma dada partição \mathcal{P} e um valor de k fixo, classificamos um caminho P de \mathcal{P} como *longo* se $|P| \geq k$, ou *curto*, se $|P| < k$. Denotamos por $\mathcal{P}^{\geq k}$ o conjunto dos caminhos longos de \mathcal{P} , e por $\mathcal{P}^{< k}$ o conjunto dos caminhos curtos. Podemos então definir a k -norma como $|\mathcal{P}|_k = \sum_{P \in \mathcal{P}} \min\{|P|, k\} = k|\mathcal{P}^{\geq k}| + |\mathcal{P}^{< k}|$.

Para um grafo G , uma partição \mathcal{P} que minimize a k -norma é chamada de k -ótima. Denotamos como $\pi_k(G)$ a k -norma de uma partição k -ótima em caminhos de G . Para qualquer grafo G , $\pi_1(G) = \pi(G)$.

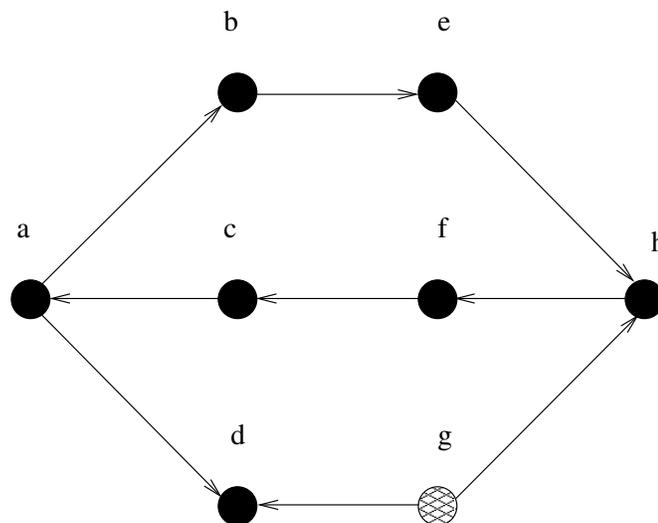


Figura 1.3: Um grafo G , e uma partição em caminhos $\mathcal{P} = \{(b, e, h, f, c, a, d), (g)\}$ de G .

A figura 1.3 ilustra uma nova partição em caminhos $\mathcal{P} = \{(b, e, h, f, c, a, d), (g)\}$ para o grafo G da figura 1.1. Para este exemplo, temos que \mathcal{P} é tanto 1-ótima, 2-ótima e 3-ótima. Ou seja, que $|\mathcal{P}|_1 = \pi_1(G) = 2$, $|\mathcal{P}|_2 = \pi_2(G) = 3$ e $|\mathcal{P}|_3 = \pi_3(G) = 4$.

Dadas essas definições, podemos expor o teorema de Greene-Kleitman.

Teorema 1.2.4. (*Greene-Kleitman, 1976*)

Para todo grafo orientado transitivo e acíclico G , $\alpha_k(G) = \pi_k(G)$.

Ao apresentarmos o teorema de Greene-Kleitman na linguagem de grafos, surge naturalmente a indagação sobre qual seria a relação existente entre colorações parciais e partições em caminhos para grafos orientados genéricos. Em 1981, Nathan Linial publicou um artigo onde é estabelecida a relação entre $\alpha_k(G)$ e $\pi_k(G)$ para um grafo G orientado e acíclico [7].

Teorema 1.2.5. (*Linial, 1981*)

Para todo grafo orientado acíclico G , $\alpha_k(G) \geq \pi_k(G)$.

No final do seu artigo, Linial também questiona se esse resultado poderia ser estendido para grafos orientados genéricos.

Conjectura 1.2.1. (*Linial, 1981*)

Para todo grafo orientado G , $\alpha_k(G) \geq \pi_k(G)$.

Obviamente, para $k = 1$, a conjectura de Linial é equivalente ao teorema de Gallai-Milgram.

Um conjunto de vértices C é *ortogonal* a uma partição em caminhos \mathcal{P} se para cada caminho $P \in \mathcal{P}$, $|V(P) \cap C| \geq 1$ — ou seja, se cada caminho P de \mathcal{P} intercepta pelo menos um vértice do conjunto C . Em geral, trataremos de ortogonalidade entre conjuntos independentes e caminhos (ou partições em caminhos). Uma k -coloração parcial C é *ortogonal* a uma partição em caminhos \mathcal{P} se cada caminho $P \in \mathcal{P}$ intercepta $\min\{|P|, k\}$ conjuntos independentes distintos de C . Na figura² 1.4 temos a 3-coloração parcial C composta dos conjuntos independentes $I_1 = \{a, g, k\}$, $I_2 = \{b, f, i, l\}$, $I_3 = \{c, e, h\}$. Neste exemplo, C é ortogonal à partição $\mathcal{P} = \{(a, b, c, d), (e, f, g), (h, i, j, k), (l)\}$.

Um ano mais tarde, a conjectura de Linial seria revisitada pelo prolífico combinatorialista Claude Berge. Em um artigo de 1982, ele propôs a seguinte generalização da conjectura de Linial [8].

Conjectura 1.2.2. (Berge, 1982)

Sejam G um grafo orientado, e k um inteiro positivo. Então, para toda partição em caminhos k -ótima de G , existe uma k -coloração parcial ortogonal a ela.

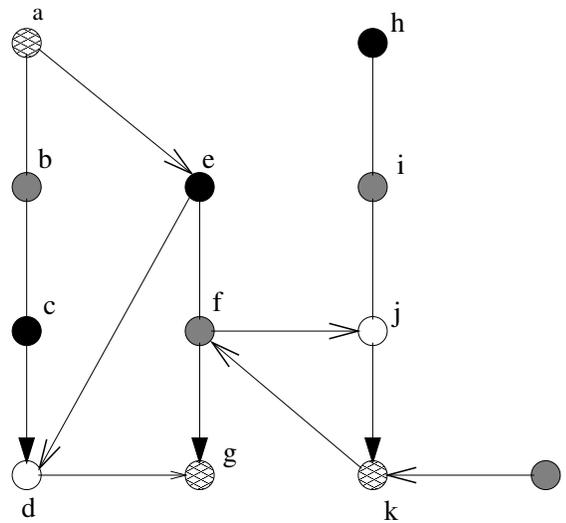


Figura 1.4: Exemplo de ortogonalidade entre uma coloração parcial e uma partição em caminhos.

Podemos mostrar que a conjectura de Berge implica na de Linial: seja \mathcal{P} uma partição

²Neste trabalho, para ilustrações gráficas de partições em caminhos, frequentemente destacamos os caminhos da partição a ser representada. Adotamos a convenção de que os vértices de um mesmo caminho dentro de uma partição estão na mesma linha vertical, e os caminhos sempre seguem o sentido de cima para baixo.

em caminhos k -ótima e C^k uma k -coloração ortogonal a ela. Então:

$$\alpha_k(G) \geq \|C^k\| = \sum_{P \in \mathcal{P}} |V(C^k) \cap V(P)| \geq \sum_{P \in \mathcal{P}} \min\{|P|, k\} = |\mathcal{P}|_k = \pi_k(G) \quad (1.1)$$

Também merece menção o elegante artigo de András Frank [18] onde os teoremas de Greene-Kleitman e de Greene (o teorema de Greene é exposto na seção 1.5) são demonstrados através de uma interpretação como um problema de fluxo em redes de custo mínimo. Esse artigo também merece menção por ser a primeira vez onde é definido e usado o conceito de ortogonalidade.

1.2.3 1982 - 2013: Da conjectura de Berge em diante

Embora as conjecturas de Linial e de Berge tenham recebido certa atenção da comunidade matemática, até a data da conclusão deste texto não foi encontrada uma demonstração (ou um contra-exemplo) para nenhuma delas. Em referências posteriores por estudiosos da área, a conjectura de Linial ficou conhecida como a *conjectura fraca das partições em caminhos* e a de Berge como a *conjectura forte das partições em caminhos*. Especialmente na década de 80 e no começo da década de 90, as conjecturas foram objeto de interesse e diversos de seus casos foram atacados com sucesso.

As primeiras soluções para casos particulares estavam no artigo de Berge que anunciava a conjectura [8]: a conjectura foi demonstrada para grafos hamiltonianos, bipartidos, e para o caso em que o grafo não possui partições que contenham caminhos longos. Em 1993, também foi demonstrada por Ron Aharoni e Irith Ben-Arroyo Hartman para partições que não contenham caminhos curtos [20].

O caso especial para grafos acíclicos tem despertado particular interesse. A demonstração da conjectura de Linial restrita a grafos acíclicos foi escrita pelo próprio Linial em seu artigo de 1981 [7]. Para a conjectura de Berge, existe uma demonstração por programação linear feita por Kathie Cameron em 1986 [9], uma feita por Hartman em 2003 estendendo a noção de partição em caminhos para uma de partição em caminhos e circuitos [22], e outra feita por Eli Berger e Hartman em 2009 utilizando fluxos em rede [24].

Em 2008, Berger e Hartman demonstraram o caso particular da conjectura que estabelece que existe uma 2-coloração parcial ortogonal a uma partição em caminhos ótima segundo a 2-norma (caso $k = 2$) [23].

Nos próximos capítulos, vamos analisar algumas dessas demonstrações, com atenção especial para a recente demonstração do caso $k = 2$ que será exposta no capítulo 4.

1.3 Teorema de Gallai-Milgram

Como adendo à introdução acima, vamos demonstrar e analisar o teorema de Gallai-Milgram; para isso, será provado como verdadeiro o seguinte teorema³:

Teorema 1.3.1. *Seja \mathcal{P} uma partição em caminhos de um grafo G tal que não haja um conjunto independente de G ortogonal a \mathcal{P} . Então, existe uma outra partição em caminhos \mathcal{Q} de G , tal que $|\mathcal{Q}| = |\mathcal{P}| - 1$ e $\text{ter}(\mathcal{Q}) \subset \text{ter}(\mathcal{P})$.*

Demonstração. A demonstração será dada por indução em $|V(G)|$. Para $|V(G)| = 1$, a proposição vale por vacuidade. Suponha então, que $|V(G)| > 1$, e seja \mathcal{P} uma partição em caminhos de G sem conjuntos independentes ortogonais a ela. Como não existem conjuntos independentes ortogonais a \mathcal{P} , $\text{ter}[\mathcal{P}]$ não é um conjunto independente, e existem $u, v \in \text{ter}[\mathcal{P}]$ tal que $(u, v) \in E$. Sejam P o caminho de \mathcal{P} que contém v , e Q o caminho de \mathcal{P} que contém u . Se P é trivial, então podemos obter a nova partição $\mathcal{Q} = \mathcal{P} \setminus \{P, Q\} \cup \{Q * (u, v)\}$, e o teorema vale.

Senão, podemos supor que v é o terminal de um caminho não trivial $P \in \mathcal{P}$. Seja $G' = G - v$ o grafo induzido obtido pela remoção de v em G , $\mathcal{P}' = \mathcal{P} \setminus \{P\}$ e $\mathcal{P}' = (\mathcal{P} \setminus \{P\}) \cup P'$. Como não existem conjuntos independentes ortogonais a \mathcal{P} em G , também não existem conjuntos independentes ortogonais a \mathcal{P}' em G' . Então, pela hipótese de indução, existe uma partição \mathcal{Q}' de G' tal que $|\mathcal{Q}'| = |\mathcal{P}'| - 1$ e $\text{ter}[\mathcal{Q}'] \subset \text{ter}[\mathcal{P}']$.

Seja v^- o vértice que precede v em P . Note que $v^- \in \text{ter}[\mathcal{P}']$. Necessariamente temos que $v^- \in \text{ter}[\mathcal{Q}']$, ou que $u \in \text{ter}[\mathcal{Q}']$. Isso é verdade, já que $\text{ter}(\mathcal{Q}') \subset \text{ter}(\mathcal{P}')$, e se ambos os vértices não pertencessem a $\text{ter}(\mathcal{Q}')$, teríamos que $|\mathcal{Q}'| \leq |\mathcal{P}'| - 2$, o que contradiz a escolha de \mathcal{Q}' . Se $v^- \in \text{ter}(\mathcal{Q}')$, então podemos definir \mathcal{Q} como a partição de G obtida a partir de \mathcal{Q}' , acrescentando a aresta (v^-, v) ao caminho de \mathcal{Q}' que termina em v^- . Se $u \in \text{ter}(\mathcal{Q}')$, podemos obter \mathcal{Q} a partir de \mathcal{Q}' , acrescentando a aresta (u, v) ao caminho de \mathcal{Q}' que termina em u . Em todos os casos, $|\mathcal{Q}| = |\mathcal{P}| - 1$ e $\text{ter}(\mathcal{Q}) \subset \text{ter}(\mathcal{P})$, como queríamos demonstrar.

□

Pela contrapositiva do teorema 1.3.1, se uma partição em caminhos é minimal em relação ao seu conjunto de terminais, então essa tem um conjunto independente ortogonal a ela. Em particular, a partição ótima de G satisfaz essa condição, e, por conseguinte, existe um conjunto independente ortogonal a ela, o que é suficiente para estabelecer a desigualdade do teorema de Gallai-Milgram.

Teorema 1.3.2. *(Gallai-Milgram, 1960)*

Para todo grafo G , $\alpha(G) \geq \pi(G)$.

³O teorema 1.3.1 e sua demonstração são baseados em um artigo de 1978 de Linial [6].

O teorema de Dilworth pode ser facilmente deduzido do teorema de Gallai e Milgram.

Teorema 1.3.3. (*Dilworth, 1950*)

Para todo grafo G transitivo e acíclico, $\alpha(G) = \pi(G)$.

Demonstração. Seja G um grafo transitivo e P um caminho em G . Temos que dois vértices distintos de P não pertencem a um mesmo conjunto independente, pela transitividade de G . Portanto, $\alpha(G) \leq \pi(G)$. Pelo teorema de Gallai-Milgram, temos também que $\alpha(G) \geq \pi(G)$, logo a igualdade vale. □

Mais que a proposição original, a demonstração acima do teorema de Gallai-Milgram pode ser interpretada como a prova de três proposições, todas as três relevantes e pertinentes. Em ordem crescente de força:

1. Para todo G , $\alpha(G) \geq \pi(G)$.
2. Toda partição ótima tem um conjunto independente ortogonal a ela.
3. Toda partição minimal quanto ao conjunto de vértices terminais tem um conjunto independente ortogonal a ela.

Das três proposições acima, a conjectura de Linial surgiu como generalização da primeira, e a conjectura de Berge como generalização da segunda para colorações parciais. A terceira é a contrapositiva do teorema 1.3.1, demonstrado logo acima.

1.4 Complexidade computacional

Definimos como k -PART o problema de, para um dado grafo G , e para $k < |V(G)|$, se encontrar uma partição em caminhos \mathcal{P} mínima quanto à k -norma, e a sua k -norma $|\mathcal{P}|_k = \pi_k(G)$.

Definimos como k -COLOR o problema de, para um dado grafo G , se encontrar uma k -coloração parcial máxima C^k , e o seu peso $|C^k| = \alpha_k(G)$.

Sabe-se (veja [4]) que o problema de decidir se um grafo possui um caminho hamiltoniano e o problema de decidir se um grafo é 3-colorível são problemas NP-completos. Demonstraremos brevemente que tanto k -PART, quanto k -COLOR são problemas NP-difíceis.

Teorema 1.4.1. k -PART é NP-difícil.

Demonstração. É fácil verificar que, para determinado G , G possui um caminho hamiltoniano se e só se $\pi_k(G) = k$. Logo, podemos reduzir o problema de decisão do caminho hamiltoniano para k -PART: caminho hamiltoniano para um dado G devolve “sim” se $\pi_k(G) = k$, e devolve “não” se $\pi_k(G) > k$. □

A demonstração que k -COLOR é NP-difícil é análoga à demonstração anterior:

Teorema 1.4.2. *k -COLOR é NP-difícil.*

Demonstração. É fácil verificar que, para determinado G , G é k -colorível se e só se $\alpha_k(G) = |V(G)|$. Logo, podemos reduzir o problema de decidir se um grafo é 3-colorível para k -COLOR: 3-coloração devolve “sim” se $\alpha_3(G) = |V(G)|$, e devolve “não” se $\alpha_3(G) < |V(G)|$. □

É trivial verificar que k -PART $\in NP$ e k -COLOR $\in NP$. O leitor interessado pode encontrar uma discussão mais aprofundada sobre complexidade computacional e problemas NP-completos em [4].

1.5 Colorações e coleções de caminhos

A conjectura de Berge é um problema em aberto que envolve uma relação entre coleções de conjuntos independentes e partições em caminhos. O que acontece ao intercalarmos os termos da proposição — ao relacionarmos *partições* em conjuntos independentes (ou seja, colorações) e *coleções* de caminhos? Curiosamente, esta nova relação é fortemente análoga à original, e encontramos vários paralelos interessantes ao considerarmos as proposições do problema original na relação entre colorações e coleções de caminhos. Por estas razões, esta relação é conhecida como a “dual” da relação entre partições em caminhos e colorações parciais.

Para um dado inteiro k , a k -norma $|C|_k$ de uma coloração $C = \{I_1, I_2, \dots, I_m\}$ é definida por $|C|_k = \sum_{i=1}^m \min\{|I_i|, k\}$. Chamamos uma coloração que minimiza $|C|_k$ de k -ótima. Denote por $\chi_k(G)$ a k -norma de uma coloração k -ótima em G . Note que uma coloração 1-ótima é uma coloração que minimiza o número de conjuntos independentes desta. Então, $\chi_1(G) = \chi(G)$, o número cromático de G .

Uma k -coleção de caminhos é um conjunto $\mathcal{P}^k = \{P_1, P_2, \dots, P_k\}$ de, no máximo, k caminhos disjuntos em G . O *peso* de uma k -coleção de caminhos $\mathcal{P}^k = \{P_1, P_2, \dots, P_k\}$ é $|\mathcal{P}^k| = \sum_{i=1}^k |P_i|$. Uma k -coleção é *ótima* se ela tem o maior peso possível entre todas as outras k -coleções de caminhos de G . Denotamos por $\lambda_k(G)$ o peso de uma k -coleção ótima em G . Em particular, $\lambda_1(G)$ é o tamanho do maior caminho em G , logo $\lambda_1(G) = \lambda(G)$.

Análogo ao teorema de Greene-Kleitman, Greene[12] demonstrou o seguinte teorema em 1976:

Teorema 1.5.1. *(Greene, 1976)*

Sejam G um grafo orientado transitivo e acíclico, e k um inteiro positivo. Então $\lambda_k(G) = \chi_k(G)$.

No mesmo artigo de Linial onde foi descrita a conjectura fraca das partições em caminhos, também foi feita a conjectura para a relação dual [7].

Conjectura 1.5.1. *(Linial, 1981)*

Seja G um digrafo e k um inteiro positivo. Então $\lambda_k(G) \geq \chi_k(G)$.

Para $k = 1$, a conjectura é equivalente ao teorema de Gallai-Roy (teorema 1.2.3). Para $k \geq \pi(G)$, a conjectura vale trivialmente, já que $\lambda_k(G) = n$ nesse caso. Hoffman [19] e Saks[17] provaram de forma independente que para G acíclico, $\lambda_k(G) \geq \chi_k(G)$ vale.

Podemos definir a noção de ortogonalidade entre coleções de caminhos e colorações: um caminho P é ortogonal a uma coloração $C = \{I_1, I_2, \dots, I_k\}$ se para cada conjunto independente $I \in C$, $|V(P) \cap I| \geq 1$. Uma coleção de caminhos \mathcal{P}^k e uma coloração C são ortogonais entre si se cada conjunto independente $I_i \in C$ intercepta $\min\{|I_i|, k\}$ caminhos diferentes em \mathcal{P}^k . É natural perguntar se podemos estender o teorema de Gallai-Roy tal como estendemos o teorema de Gallai-Milgram para a conjectura de Berge.

Conjectura 1.5.2. *Seja G um grafo orientado e k um inteiro positivo. Então, para toda coloração k -ótima, existe uma k -coleção de caminhos ortogonal a ela.*

Embora ela valha para conjuntos parcialmente ordenados, a conjectura 1.5.2 é falsa; a figura 1.5 descreve um contra-exemplo simples para ela. A seguinte alternativa à conjectura 1.5.2 foi proposta por Aharoni, Hartman e Hoffman:

Conjectura 1.5.3. *(Aharoni, Hartman, Hoffman, 1985) [21]*

Seja G um grafo orientado e k um inteiro positivo. Então, para toda k -coleção de caminhos ótima, existe uma coloração ortogonal a ela.

A conjectura 1.5.3 implica na conjectura 1.5.1. Entre outros casos, a conjectura vale para $k = 1$, grafos acíclicos e para $k \geq \pi(G)$. Nenhuma das conjecturas 1.5.1 ou 1.5.3 foi resolvida até a conclusão deste trabalho. Adiante, mostramos duas tabelas resumindo a diferença entre o problema original e o problema dual mostrado nesta seção.

Problema original	Problema dual
Partição em caminhos $\mathcal{P} = \{P_1, P_2, \dots, P_m\}$	Coloração própria (partição em conjuntos independentes) $C = \{I_1, I_2, \dots, I_m\}$
k -coloração parcial (coleção de k conjuntos independentes) $C^k = \{I_1, I_2, \dots, I_k\}$	k -coleção de caminhos $\mathcal{P}^k = \{P_1, P_2, \dots, P_k\}$
$ \mathcal{P} _k = \sum_{i=1}^m \min\{ P_i , k\}$ k -norma de uma partição em caminhos	$ C _k = \sum_{i=1}^m \min\{ I_i , k\}$ k -norma de uma coloração parcial
$\pi_k(G) = \min\{ \mathcal{P} _k : \mathcal{P} \text{ é uma partição em caminhos}\}$	$\chi_k(G) = \min\{ C _k : C \text{ é uma coloração própria de } G\}$
$\alpha_k(G)$: Número máximo de vértices em uma k -coloração parcial C^k	$\lambda_k(G)$: Número máximo de vértices em uma k -coleção de caminhos \mathcal{P}^k
Ortogonalidade entre \mathcal{P} e C^k : $\forall P \in \mathcal{P}, P$ intercepta $\min\{ P , k\}$ conjuntos independentes distintos de C^k	Ortogonalidade entre C e \mathcal{P}^k : $\forall I \in C, I$ intercepta $\min\{ I , k\}$ caminhos distintos de \mathcal{P}^k

Tabela 1.1: Tabela de comparação de definições entre o problema original e o dual

Problema original	Problema dual
1960 - Teorema de Gallai-Milgram $\forall G : \alpha(G) \geq \pi(G)$	1967 - Teorema de Gallai-Roy $\forall G : \lambda(G) \geq \chi(G)$
1976 - Teorema de Greene-Kleitman $\forall G$ acíclico e transitivo: $\alpha_k(G) = \pi_k(G)$	1976 - Teorema de Greene $\forall G$ acíclico e transitivo: $\lambda_k(G) = \chi_k(G)$
1981 - Conjectura de Linial $\forall G : \alpha_k(G) \geq \pi_k(G)$	1981 - Conjectura de Linial $\forall G : \lambda_k(G) \geq \chi_k(G)$
1982 - Conjectura de Berge $\forall G$: Para toda partição \mathcal{P} tal que $ \mathcal{P} _k = \pi_k(G)$, existe uma k -coloração parcial ortogonal a \mathcal{P} .	Conjectura Falsa $\forall G$: Para toda coloração C tal que $ C _k = \chi_k(G)$, existe uma k -coleção de caminhos ortogonal a C
	1985 - Conjectura de Aharoni-Hartman-Hoffman $\forall G$: Para toda k -coleção de caminhos \mathcal{P}^k tal que $ \mathcal{P}^k = \lambda_k(G)$, existe uma coloração ortogonal a C .

Tabela 1.2: Tabela de comparação de resultados entre o problema original e o dual

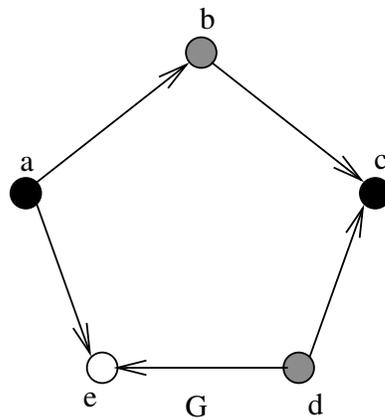


Figura 1.5: Um contra-exemplo para a conjectura 1.5.2. Em G , $C = \{\{a, c\}, \{b, d\}, \{e\}\}$ é uma coloração 1-ótima tal que não existe uma 1-coleção de caminhos ortogonal a ela (já que não existe um caminho hamiltoniano em G).

Capítulo 2

A Conjectura de Berge

As conjecturas de Berge e de Linial foram propostas no começo da década de 80 e, apesar das tentativas de diversos autores, ambas continuam em aberto. Entretanto, a conjectura de Berge (e por conseguinte a de Linial) foi provada como verdadeira para vários casos particulares de grafos; entre elas, podemos citar digrafos acíclicos, bipartidos e hamiltonianos (veja [8] e [9]).

Muitas dessas demonstrações para casos particulares provêm de tentativas feitas para o caso geral. Mas como acontece com frequência na matemática, essas demonstrações tiveram suas idéias devidamente aproveitadas para resolver casos particulares do teorema.

Este capítulo tem um enfoque exploratório, e nele mostramos alguns estudos que foram feitos sobre o problema da partição em caminhos nos últimos 30 anos.

2.1 Casos particulares da conjectura

2.1.1 Partições sem caminhos longos

Um dos casos particulares ocorre quando, para um grafo G e um inteiro k , não existem partições em caminhos que contenham caminhos longos em relação a k . É fácil ver que, sempre que $k \geq \lambda(G)$, qualquer partição em caminhos de G é k -ótima. Neste caso, o problema de otimização da k -norma é trivial. Além disso, a conjectura é válida, e podemos obter uma k -coloração ortogonal a qualquer das partições em caminhos encontradas neste caso.

A demonstração para este caso usa a mesma idéia da demonstração para o teorema de Gallai-Roy usada no livro sobre digrafos de Bang-Jensen e Gutin [5].

Teorema 2.1.1. *Sejam G um grafo orientado, $k \geq \lambda(G)$ um inteiro positivo, e \mathcal{P} uma partição em caminhos de G . Então existe uma coloração $C = \{I_1, I_2, \dots, I_{\lambda(G)}\}$ de G que também é uma k -coloração parcial ortogonal a \mathcal{P} .*

Demonstração. Sejam G um grafo com $\lambda(G) \leq k$, \mathcal{P} uma partição em caminhos qualquer de G , e H um subgrafo gerador de G tal que H é acíclico, maximal quanto ao conjunto de arestas, e tal que $E[\mathcal{P}] \subseteq E[H]$. Para cada $v \in V(G)$, seja $f(v)$ a cardinalidade de um caminho mais longo em H que começa em v . Note que se (x, y) é uma aresta em H , então $f(x) > f(y)$. Além disso, se (x, y) é uma aresta em $G - H$, então existe um caminho de y para x em H , já que H é maximalmente acíclico. Logo, $f(y) > f(x)$.

Como f mapeia um valor para cada vértice tal que cada par de vértices adjacentes em G têm valores distintos, ela descreve a coloração própria $C = \{I_1, I_2, \dots, I_{\lambda(G)}\}$ de G , onde $v \in C_i$ se e somente se $f(v) = i$.

Para todo $v \in G$, $f(v)$ é definida e $1 \leq f(v) \leq \lambda(G) \leq k$, todos os vértices de G recebem uma cor distinta entre 1 e k . Além disso, dentro de um caminho $P \in \mathcal{P}$, $f(v) > f(v_P^+)$. Logo C é uma k -coloração parcial ortogonal a \mathcal{P} . □

O Teorema de Gallai-Roy segue como corolário do teorema anterior.

Teorema 2.1.2. (*Gallai-Roy 1967*)

Para todo grafo G , $\lambda(G) \geq \chi(G)$.

2.1.2 Grafos com um caminho hamiltoniano

Teorema 2.1.3. *Seja G um grafo com um caminho hamiltoniano P (ou seja, $\lambda(G) = |V(G)|$). Então, a conjectura de Berge vale para G .*

Demonstração. Para $\lambda(G) \leq k$, a prova segue pelo teorema 2.1.1. Para $\lambda(G) > k$, a única partição k -ótima é a partição $\mathcal{P} = \{P\}$ pois qualquer outra partição com mais de um caminho tem peso maior. Como encontrar uma coloração ortogonal a \mathcal{P} é trivial, o teorema vale. □

2.1.3 Grafos bipartidos

Inicialmente, note que a conjectura de Linial é trivialmente verdadeira para grafos bipartidos. Isso segue do fato que para um grafo bipartido G , $\chi(G) = 2$ e que para todo $k \geq 2$, $\alpha_k(G) = |V(G)|$. Segue-se que $\alpha_k(G) \geq \pi_k(G)$ para grafos bipartidos.

Resta provar a conjectura de Berge para grafos bipartidos. Definimos $ter_i[\mathcal{P}]$ como o conjunto dos vértices v de algum caminho $P \in \mathcal{P}$ tal que $|vP| = i$. Por exemplo, $ter_1[\mathcal{P}] = ter[\mathcal{P}]$ e $ter_2[\mathcal{P}] = ter^-[\mathcal{P}]$.

Teorema 2.1.4. *Seja G um grafo bipartido com bipartição $\{X, Y\}$. Então, a conjectura de Berge vale para G .*

Demonstração. A prova será dividida em dois casos. No primeiro caso, tratamos o caso em que $k = 2p$ é par. Seja \mathcal{P} uma partição k -ótima de G e denote $r_p = \min\{|P|, 2p\}$, sendo P um caminho de \mathcal{P} . Vamos definir uma coloração parcial $\{I_1, I_{1'}, I_2, I_{2'}, \dots, I_p, I_{p'}\}$ de G . Para cada caminho $P \in \mathcal{P}$, atribuímos sucessivamente cada um dos últimos r_p vértices de P , partindo de $ter(P)$ e percorrendo P no sentido inverso, a um desses conjuntos. Se $v \in X$, v é atribuído ao conjunto independente I_j , onde j é o menor índice tal que nenhum vértice de $X \cap v^+P$ foi atribuído a I_j anteriormente. Se $v \in Y$, v é atribuído ao conjunto independente I'_j , onde j é o menor índice tal que nenhum vértice de $Y \cap v^+P$ foi atribuído a I'_j . Claramente, esta é uma coloração parcial ortogonal a \mathcal{P} .

Agora, vamos considerar o caso onde $k = 2p + 1$ é ímpar. Seja \mathcal{P} uma partição k -ótima de G . Vamos definir agora um subgrafo H determinado a partir de \mathcal{P} . Removemos de G todos os vértices pertencentes a $ter_i[\mathcal{P}]$, para $1 \leq i \leq 2p$. Ou seja, serão removidos de G os últimos r_p vértices de cada caminho $P \in \mathcal{P}$. Se um dado caminho P for curto, todos os seus vértices serão removidos. Seja \mathcal{Q} a partição em caminhos de H obtida a partir de \mathcal{P} , com a exclusão desses vértices. Vamos mostrar que $ter[\mathcal{Q}]$ é minimal entre todas as partições possíveis de H e $\pi(H) = |\mathcal{Q}|$. Caso contrário, existe uma partição \mathcal{Q}' de H , tal que $ter[\mathcal{Q}'] \subset ter[\mathcal{Q}]$ e $|\mathcal{Q}'| < |\mathcal{Q}|$, pelo teorema de Gallai-Milgram.

Neste caso, é possível estender \mathcal{Q}' , e obter uma nova partição \mathcal{P}' de G da seguinte forma: dado um vértice $v \in ter[\mathcal{Q}']$, sejam P o caminho que contém v em \mathcal{P} , Q o caminho que contém v em \mathcal{Q}' , e R o caminho formado pelos últimos $2p$ vértices de P . Inserimos em \mathcal{P}' o caminho $P' = Q * (v, v_p^+) * (v_p^+ R)$. Isto é feito para cada $v \in ter[\mathcal{Q}']$. Feito isso, note que $V(G) - V[\mathcal{P}']$ possui uma partição em caminhos correspondente aos sufixos dos caminhos de \mathcal{P} que não foram concatenados a caminhos de \mathcal{Q}' no passo anterior. Os caminhos desta partição são acrescentados a \mathcal{P}' , resultando em uma nova partição \mathcal{P}' de G .

Como $ter[\mathcal{Q}'] \subset ter[\mathcal{Q}]$, existe algum vértice que pertence a $ter[\mathcal{Q}]$, mas não a $ter[\mathcal{Q}']$. Desta forma, algum dos caminhos de \mathcal{P}' é um caminho novo R composto dos últimos $2p$ vértices do caminho original P . Em \mathcal{P}' , $|R| = 2p$. Em \mathcal{P} , temos que $|P| \geq 2p + 1$. Trata-se da substituição de um caminho longo por um curto. Então $|\mathcal{P}'|_k < |\mathcal{P}|_k$, contradizendo a hipótese inicial de que \mathcal{P} é uma partição k -ótima. Logo, \mathcal{Q}' não existe, e $ter[\mathcal{Q}]$ é minimal.

Como $ter[\mathcal{Q}]$ é minimal, pelo corolário¹ do teorema 1.3.1 existe um conjunto independente I_0 , que é ortogonal a \mathcal{Q} . Este conjunto independente também é ortogonal a \mathcal{P} , e podemos obter uma $(2p + 1)$ -coloração parcial $C = \{I_0, I_1, I_{1'}, I_2, I_{2'}, \dots, I_p, I_{p'}\}$ onde os demais conjuntos independentes são obtidos como no caso anterior da demonstração. \square

¹Descrito informalmente em análise no final da seção 1.3.

2.2 Grafos acíclicos

O caso da conjectura de Berge para grafos acíclicos foi demonstrado por meio de diversos métodos (veja [9], [22] ou [24]). Nesta seção faremos a demonstração segundo um destes métodos, oriunda de um artigo recente de Berger e Hartman [24]. Nesta demonstração, a idéia essencial é a associação entre uma partição de um digrafo acíclico em caminhos e um fluxo de uma rede auxiliar.

2.2.1 Definição da Rede

Seja $G = (V, E)$ um grafo orientado acíclico, e $V(G) = \{v_1, v_2, \dots, v_n\}$. Associada a G , construímos uma rede $R = (\bar{V}, \bar{E}, c, s, t)$ onde:

- $\bar{V} = \{s, t, v'_1, v'_2, \dots, v'_n, v''_1, v''_2, \dots, v''_n\}$,
- $\bar{E} = \{(s, v'_i) : i = 1, 2, \dots, n\} \cup \{(v''_i, t) : i = 1, 2, \dots, n\} \cup \{(v'_i, v''_j) : (v_i, v_j) \in E\} \cup \{(v'_i, v''_i) : 1 \leq i \leq n\} \cup (s, t)$,
- $c(e) = \begin{cases} 1 & \text{se } e = (v'_i, v''_i), \\ k & \text{se } e = (s, t), \\ 0 & \text{caso contrário.} \end{cases}$

Os conjuntos \bar{V} e \bar{E} são, respectivamente, o conjunto de vértices e arestas da rede R , enquanto a função $c : \bar{E} \rightarrow \mathbb{Z}$ representa o custo de cada aresta de \bar{E} . Os vértices $s, t \in \bar{V}$ têm um papel especial e são chamados de *fonte* e *sorvedouro* de R . A utilidade do arco (s, t) ficará clara adiante na seção de demonstrações.

A figura 2.1 ilustra a obtenção de uma rede a partir de um dado grafo, pelo método descrito acima.

Por conveniência, para uma função $f : \bar{E} \rightarrow \mathbb{R}$ e uma aresta $e = (u, v) \in \bar{E}$, usaremos a notação $f(u, v)$ como substituta para a notação $f((u, v))$. Definimos um fluxo viável sobre R como uma função $f : \bar{E} \rightarrow \mathbb{R}^+$ tal que:

1. para uma dada aresta $e \in \bar{E}$, $f(e) \leq 1$,
2. para todo vértice $v \in \bar{V} \setminus \{s, t\}$, $\sum_{(u,v) \in \bar{E}} f(u, v) = \sum_{(v,w) \in \bar{E}} f(v, w)$.

Definimos o valor de um fluxo f como $val(f) = \sum_{(s,v) \in \bar{E}} f(s, v)$ e seu custo como $custo(f) = \sum_{e \in \bar{E}} f(e)c(e)$. Também definimos \mathcal{P}^1 como o conjunto de caminhos triviais de uma partição em caminhos \mathcal{P} .

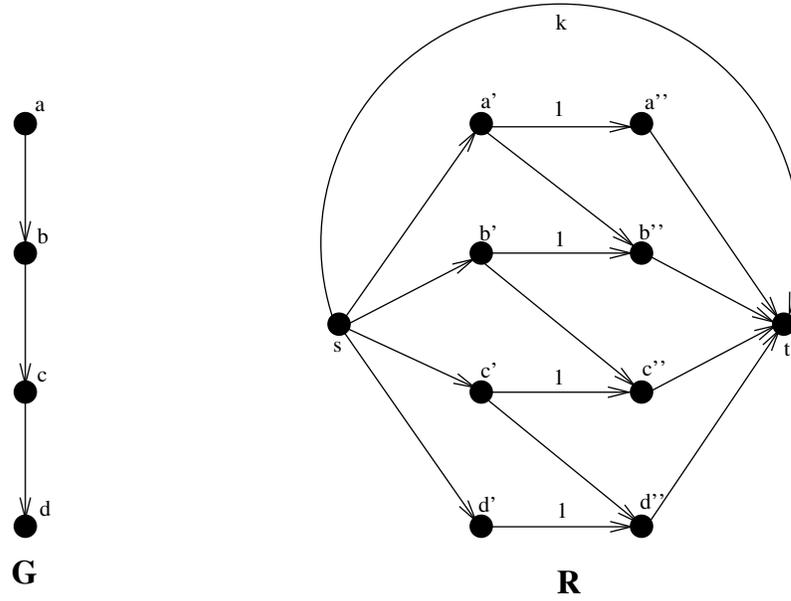


Figura 2.1: Exemplo de uma rede R , construída a partir do grafo G . Os valores descritos nas arestas correspondem aos seus custos.

2.2.2 Rede residual

Para um dado fluxo f em R , definimos a rede residual R_f como:

$$R_f = \{\bar{V}, \bar{E}_f, c_f, s, t\} \quad (2.1)$$

onde $\bar{E}_f := \{e \in \bar{E} : f(e) < 1\} \cup \{\bar{e} : e \in \bar{E} \text{ e } f(e) > 0\}$.

Definimos aqui $\bar{e} = (v, u)$ como a reversa da aresta $e = (u, v)$. O custo residual $c_f : \bar{E}_f \rightarrow \mathbb{Z}$ é definido como $c_f(e) = c(e)$ e $c_f(\bar{e}) = -c(e)$ para todo $e \in \bar{E}$.

Sejam f um fluxo viável em uma rede R , e g um fluxo viável na rede residual R_f . Definimos a função $f + g$ da seguinte forma: $(f + g)(e) = f(e) + g(e) - g(\bar{e})$ para todo $e \in \bar{E}$. Note que $f + g$ é um fluxo na rede R . O seguinte lema segue claramente pela definição dada.

Lema 2.2.1. *Sejam f um fluxo viável em uma rede R , e g um fluxo viável na rede residual R_f . Então $val(f + g) = val(f) + val(g)$ e $custo(f + g) = custo(f) + custo(g)$. \square*

Dentro de rede residual, definimos o valor de um fluxo g como

$$val(g) = \sum_{(s,v) \in \bar{E}_f} g(s, v) - \sum_{(v,s) \in \bar{E}_f} g(v, s) \quad (2.2)$$

e seu custo como $custo(g) = \sum_{e \in \bar{E}} g(e)c_f(e)$.

Seja C um caminho ou um circuito em R_f . Definimos o *custo* de C como $\text{custo}(C) = \sum_{e \in E(C)} c_f(e)$. Seja f um fluxo viável e suponha que a rede residual R_f não contém circuitos de custo negativo. Para cada $u \in \bar{V}$, seja $\rho(u)$ o custo de um caminho mínimo de s a u em R_f . Chamamos ρ de *função potencial* em R_f . Note que pela definição dada, para uma aresta $(u, v) \in \bar{E}_f$, $\rho(v) \leq \rho(u) + c(u, v)$.

2.2.3 Fluxos e partições

Seja f um fluxo em R . Definimos um vértice $v_i \in V(G)$ como *saturado* em f se $f(s, v'_i) = 1$ ou se $f(v''_i, t) = 1$. Um fluxo f é *completo* se todo vértice $v \in V(G)$ é saturado em f . Um fluxo f em R é *inteiro* se $f(e) \in \{0, 1\}$ para todo $e \in \bar{E}$.

Sejam G um grafo acíclico, R a rede correspondente a este grafo segundo a definição dada na seção 2.2.1 e f um fluxo viável inteiro em R . Seja $\mathcal{P}(f) = (V, F)$ o subgrafo gerador de G tal que $(v_i, v_j) \in F$ se e somente se $f(v'_i, v''_j) = 1$, para $i \neq j$. Demonstramos no seguinte lema que $\mathcal{P}(f)$ corresponde a uma partição em caminhos de G .

Lema 2.2.2. *Sejam G um grafo acíclico, R a rede correspondente a este grafo segundo a definição dada na seção 2.2.1 e f um fluxo viável inteiro em R . Então, $\mathcal{P}(f)$ é uma partição em caminhos de G .*

Demonstração. Em R , a única aresta que entra em um dado vértice v'_i de \bar{V} é (s, v'_i) . Igualmente, de um vértice v''_i de \bar{V} , somente uma aresta (v''_i, t) sai. Então, no máximo uma unidade de fluxo sai de v'_i , ou entra em v''_i , e em todo vértice $v_i \in V(\mathcal{P}(f)) = V(G)$, temos que $d^-(v_i) \leq 1$ e $d^+(v_i) \leq 1$. Como G é um grafo acíclico, $\mathcal{P}(f)$ é um subgrafo gerador acíclico que corresponde a uma partição em caminhos. \square

Inversamente, dada uma partição em caminhos \mathcal{P} em G , definimos o seu fluxo correspondente $f = f_{\mathcal{P}}$ da seguinte forma: se $(v_i) \in \mathcal{P}^1$, então $f(s, v'_i) = f(v'_i, v''_i) = f(v''_i, t) = 1$. Se $(v_i, v_j) \in E[\mathcal{P}]$, então $f(s, v'_i) = f(v'_i, v''_j) = f(v''_j, t) = 1$. Para todas as demais arestas $e \in \bar{E}$, temos que $f(e) = 0$.

Sejam f um fluxo completo e $\mathcal{P} = \mathcal{P}(f)$ a partição em caminhos derivada de f . As seguintes equações são equivalentes, e valem pela definição de custo e pelo fato de f ser completo:

$$\begin{aligned} \text{custo}(f) &= |\mathcal{P}^1| + k \cdot f(s, t) \\ |\mathcal{P}^1| &= \text{custo}(f) - k \cdot f(s, t). \end{aligned} \tag{2.3}$$

Para atestarmos a validade das equações em (2.4) (também equivalentes entre si), é suficiente observar que cada caminho trivial em \mathcal{P} contribui com uma unidade em $\text{val}(f)$,

e cada caminho não trivial P contribui com $|P| - 1$ unidades, totalizando $n - |\mathcal{P}^{>1}|$ unidades ao todo:

$$\begin{aligned} \text{val}(f) &= n - |\mathcal{P}^{>1}| + f(s, t) \\ |\mathcal{P}^{>1}| &= n - \text{val}(f) + f(s, t). \end{aligned} \quad (2.4)$$

Logo, temos a seguinte relação entre o fluxo f e a partição $\mathcal{P} = \mathcal{P}(f)$:

$$|\mathcal{P}|_k = k|\mathcal{P}^{\geq k}| + |V[\mathcal{P}^{<k}]| \leq k|\mathcal{P}^{>1}| + |\mathcal{P}^1| = k(n - \text{val}(f)) + \text{custo}(f). \quad (2.5)$$

Além disso, se k é menor ou igual que a cardinalidade do menor caminho não trivial em \mathcal{P} , a equação acima se torna uma igualdade:

$$|\mathcal{P}|_k = k(n - \text{val}(f)) + \text{custo}(f). \quad (2.6)$$

Para um fluxo $f = f_{\mathcal{P}}$ derivado de uma partição \mathcal{P} , é fácil ver que f é um fluxo viável completo com $\text{val}(f) = n - |\mathcal{P}^{>1}|$ e $\text{custo}(f) = |\mathcal{P}^1|$. Portanto, se k é menor ou igual que a cardinalidade do menor caminho não trivial em \mathcal{P} , a equação 2.6 também vale para \mathcal{P} e $f = f_{\mathcal{P}}$.

Se k é maior que a cardinalidade do menor caminho não trivial em \mathcal{P} (ou seja, se \mathcal{P} possui caminhos curtos não triviais), podemos separar artificialmente os caminhos curtos não-triviais de \mathcal{P} em caminhos triviais. Definimos \mathcal{P}^* como a partição em caminhos derivada de \mathcal{P} , onde todo caminho curto não trivial em \mathcal{P} tem seus vértices separados em caminhos triviais em \mathcal{P}^* . Temos que $|\mathcal{P}|_k = |\mathcal{P}^*|_k$, e vale a seguinte equação, equivalente à equação 2.6:

$$|\mathcal{P}|_k = |\mathcal{P}^*|_k = k(n - \text{val}(f)) + \text{custo}(f). \quad (2.7)$$

A figura 2.2 mostra um exemplo da obtenção de uma partição $\mathcal{P}(f)$ a partir de um fluxo f , enquanto a figura 2.3 exemplifica a obtenção de um fluxo $f_{\mathcal{P}}$ a partir de uma partição em caminhos \mathcal{P} .

2.2.4 Demonstração da conjectura de Berge para grafos acíclicos.

Nesta seção, conduzimos a sequência de demonstrações necessárias para a prova do caso acíclico da conjectura de Berge.

Lema 2.2.3. *Seja f um fluxo inteiro e completo em R tal que $\mathcal{P} = \mathcal{P}(f)$ não contenha caminhos não-triviais de tamanho menor que k . Se existe um circuito de custo negativo, ou $\rho(t) < k$ em R_f , então existe uma partição \mathcal{P}' em caminhos de G tal que $|\mathcal{P}'|_k < |\mathcal{P}|_k$.*

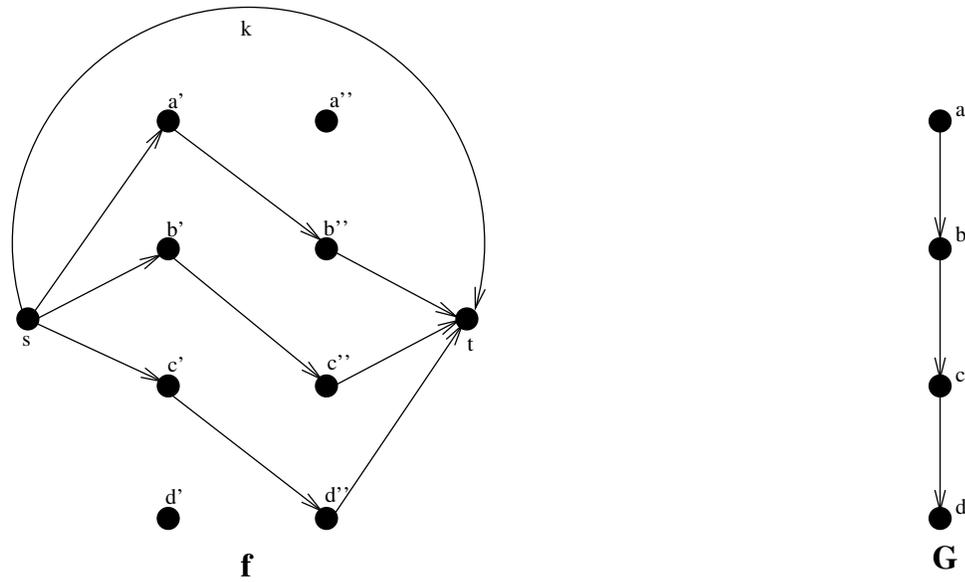


Figura 2.2: Esquerda: Um fluxo inteiro viável f dentro de uma rede R . Na figura, temos que $f(s, t) = k$, e para toda outra aresta e presente, $f(e) = 1$. A presença da aresta (s, t) ou seu valor de fluxo é irrelevante para a definição de $\mathcal{P}(f)$. Direita: Um grafo G e sua partição em caminhos derivada de f , $\mathcal{P}(f) = \{(a, b, c, d)\}$.

Demonstração. Se existe um circuito de custo negativo Q em R_f , então ele corresponde a um fluxo g em R_f de valor 0 e custo negativo. Logo:

$$\begin{aligned}
 |\mathcal{P}|_k &= k.(n - \text{val}(f)) + \text{custo}(f) \\
 &> k.(n - (\text{val}(f) + \text{val}(g))) + \text{custo}(f) + \text{custo}(g) \\
 &= k.(n - \text{val}(f + g)) + \text{custo}(f + g).
 \end{aligned} \tag{2.8}$$

Se não existe um circuito de custo negativo, mas $\rho(t) < k$ em R_f , então existe um caminho Q de s a t em R_f de custo menor que k . Este caminho corresponde a um fluxo g em R_f tal que $\text{val}(g) = 1$ e $\text{custo}(g) = \rho(t) < k$. Logo:

$$\begin{aligned}
 |\mathcal{P}|_k &= k.(n - \text{val}(f)) + \text{custo}(f) \\
 &= k.(n - \text{val}(f) - 1) + \text{custo}(f) + k \\
 &> k.(n - \text{val}(f + g)) + \text{custo}(f + g).
 \end{aligned} \tag{2.9}$$

Em ambos os casos, se $f + g$ determina um fluxo completo em R , segue-se que $\mathcal{P}' = \mathcal{P}(f + g)$ é uma partição em caminhos tal que $|\mathcal{P}'|_k \leq k.(n - \text{val}(f + g)) + \text{custo}(f + g) <$

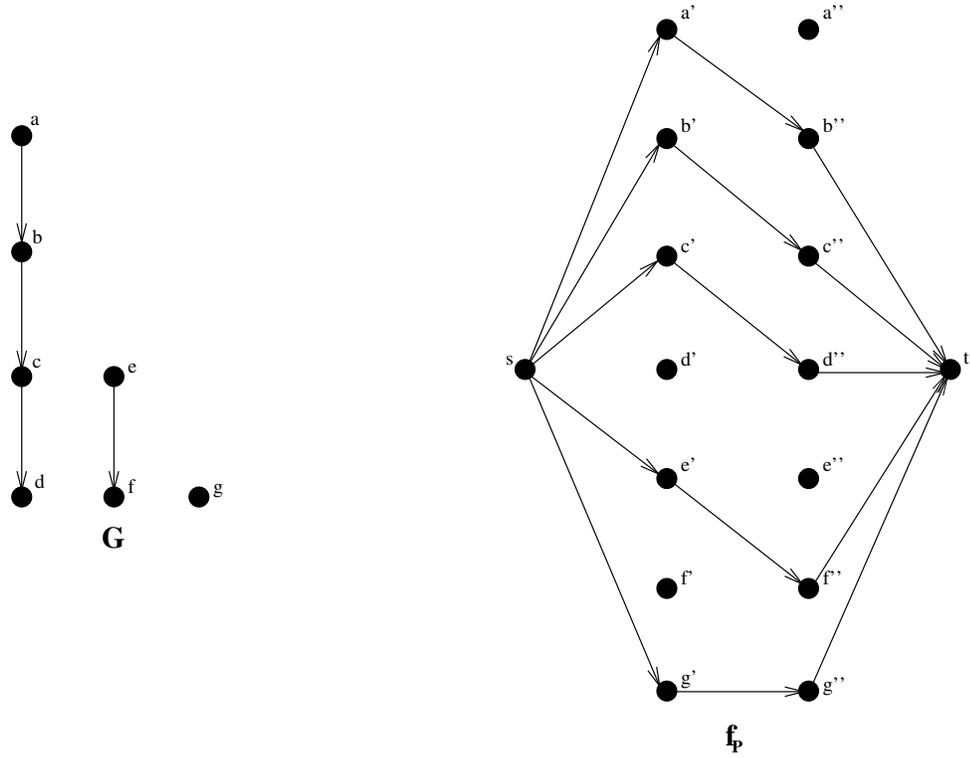


Figura 2.3: Esquerda: Em G , temos a partição em caminhos $\mathcal{P} = \{(a, b, c, d), (e, f), (g)\}$. Direita: A partir de \mathcal{P} , é obtido o fluxo $f_{\mathcal{P}}$. As arestas e explicitadas na figura são aquelas onde $f(e) = 1$.

$|\mathcal{P}|_k$. Caso contrário², definimos um fluxo h da seguinte forma: para todo vértice não saturado v em R , tome $h(s, v') = h(v', v'') = h(v'', t) = 1$. Para todas as outras arestas e de R , tome $h(e) = 0$. Logo, $val(h) = custo(h) = m$, onde m é o número de vértices não-saturados em $f + g$. Logo,

$$\begin{aligned}
 & k.(n - val(f + g)) + custo(f + g) \\
 \geq & k.(n - val(f + g)) + custo(f + g) - k.val(h) + custo(h) \\
 = & k.(n - val(f + g + h)) + custo(f + g + h).
 \end{aligned} \tag{2.10}$$

Portanto, $\mathcal{P}' = \mathcal{P}(f + g + h)$ é uma partição em caminhos tal que

$$|\mathcal{P}'|_k = k.(n - val(f + g + h)) + custo(f + g + h) < |\mathcal{P}|_k. \tag{2.11}$$

²O artigo original assume erroneamente que $f + g$ sempre é um fluxo completo. Deixamos a cargo do leitor verificar que este pode não ser o caso se g é um fluxo oriundo de um circuito de custo negativo.

□

Na figura 2.4, temos um exemplo de um circuito negativo encontrado em uma rede residual. A partir do grafo G descrito na figura 2.1, obtemos uma partição de G em caminhos $\mathcal{P} = \{(a), (b), (c), (d)\}$ composta somente de caminhos triviais, e definimos $k = 2$. A partir de \mathcal{P} , obtemos um fluxo $f = f_{\mathcal{P}}$ viável da rede R (R está descrita na figura 2.1). A partir de f , obtemos a rede residual R_f de R . O circuito $C = (s, t, a'', a', b'', b', c'', c', d'', d', s)$ é um circuito negativo em R_f com $\text{custo}(C) = -2$. Do circuito C , podemos obter um fluxo g sobre a rede residual R_f . Finalmente, a partir do fluxo $h = f + g$, obtemos $\mathcal{P}' = \mathcal{P}(h) = \{(a, b, c, d)\}$, que é uma partição em caminhos com menor k -norma que \mathcal{P} (também, neste exemplo, uma partição k -ótima de G).

Para a sequência de lemas de 2.2.4 a 2.2.7, seja R_f uma rede residual baseada no fluxo f em R que tenha a função potencial definida para todo vértice (que não tenha, portanto, circuitos negativos). Definimos o conjunto $C^k = \{C_1, C_2, \dots, C_k\}$ como um candidato a uma coloração parcial ortogonal a $\mathcal{P} = \mathcal{P}(f)$, onde C_j é definido da seguinte forma:

$$C_j = \{v \in V(G) : j = \rho(v'') = \rho(v') + 1\}. \quad (2.12)$$

Também é útil lembrar da definição de R_f : se $f(u, v) = 0$, então $(u, v) \in \overline{E}_f$, e se $f(u, v) = 1$, então $(v, u) \in \overline{E}_f$.

Lema 2.2.4. C_j é um conjunto independente em G .

Demonstração. Suponha que $u, v \in C_j$, que $(u, v) \in E(G)$ e, portanto, $(u', v'') \in \overline{E}$. Pela definição de C_j , $\rho(u') + 1 = \rho(v') + 1 = \rho(u'') = \rho(v'') = j$. Se $f(u', v'') = 0$, então $\rho(v'') \leq \rho(u')$, contradizendo a suposição que $\rho(v'') = \rho(u') + 1$. Se $f(u', v'') = 1$, então $(v'', u') \in \overline{E}_f$ é a única aresta que entra em u' em R_f . Logo, $\rho(u') = \rho(v'')$. Em ambos os casos, a suposição de que existe uma aresta com extremos em C_j gera uma contradição. □

Lema 2.2.5. Suponha que $\rho(t) = k$, e seja P um caminho em G tal que $P \in \mathcal{P}^{\geq k}$. Então, para todo j tal que $1 \leq j \leq k$, existe um vértice $v \in P$ tal que $v \in C_j$.

Demonstração. Seja $P = (v_1, v_2, \dots, v_l) \in \mathcal{P}^{\geq k}$. Então, $f(v'_i, v''_{i+1}) = 1$ para $i = 1, 2, \dots, l - 1$, $f(v'_i, v''_i) = 0$ para $i = 1, 2, \dots, l$, e $f(s, v'_1) = f(v''_1, t) = 0$. Como $f(s, v'_1) = 0$ e $c_f(s, v'_1) = 0$, temos que $\rho(v'_1) = 0$. Analogamente, como $\rho(t) = k$, $f(v''_1, t) = 0$ e $c_f(v''_1, t) = 0$, necessariamente $\rho(v''_1) \geq k$.

Como $f(v'_i, v''_i) = 0$ e $c_f(v'_i, v''_i) = 1$ para $i = 1, 2, \dots, l$, temos que $\rho(v''_i) \leq \rho(v'_i) + 1$. Como $f(s, v'_i) = f(v'_i, v''_{i+1}) = 1$ para $1 \leq i \leq l - 1$, segue necessariamente que (v''_{i+1}, v'_i) é a única aresta em R_f que entra em v'_i e $\rho(v'_i) = \rho(v''_{i+1})$. Temos então que

$$\rho(v_i'') \leq \rho(v_i') + 1 = \rho(v_{i+1}'') + 1. \quad (2.13)$$

$$\begin{aligned} \rho(v_1'') &\geq k, \\ \rho(v_{i+1}'') &\geq \rho(v_i'') - 1, \\ \rho(v_i'') &\leq 1. \end{aligned} \quad (2.14)$$

Pelo sistema de desigualdades acima, para todo j tal que $1 \leq j \leq k$, existe necessariamente algum v_i tal que $\rho(v_i'') = j$ e $\rho(v_i') = j - 1$. Logo, para $1 \leq j \leq k$, todo conjunto C_j tem um representante em P . \square

Lema 2.2.6. *Suponha que $\rho(t) = k$, e seja P um caminho em G tal que $P \in \mathcal{P}^{<k}$. Então, para todo vértice $v \in P$, necessariamente $v \in C_j$, onde $1 \leq j \leq k$. Além disso, para todo par de vértices $v, w \in P$, temos que se $v \in C_g$ e $w \in C_h$, então $g \neq h$.*

Demonstração. Seja \mathcal{P}^* a partição derivada de \mathcal{P} com a separação dos caminhos curtos não-triviais em caminhos triviais, conforme definido na seção 2.2.3. Temos que $V[\mathcal{P}^{*1}] = V[\mathcal{P}^{<k}]$. Sejam $P = (v_1, v_2, \dots, v_l) \in \mathcal{P}^{<k}$ e $f = f_{\mathcal{P}^*}$. Para cada $v_i \in P$, (v_i) é um caminho trivial em \mathcal{P}^* . Então, para $v_i \in P$, $f(s, v_i') = f(v_i', v_i'') = f(v_i'', t) = 1$. Como $\rho(t) = k$ e $c_f(t, v_i'') = 0$, $\rho(v_i'') \leq k$. Analogamente, como $\rho(s) = 0$ e $c_f(v_i', s) = 0$, $\rho(v_i') \geq 0$. Como $f(v_i', v_i'') = 1$, existe uma aresta (v_i'', v_i') entrando em v_i' em R_f . Como esta aresta é a única que entra em v_i' , e $c_f(v_i'', v_i') = 1$, então $\rho(v_i'') = \rho(v_i') + 1$. Então, necessariamente $v_i \in C_j$ onde $j = \rho(v_i'')$ e $1 \leq j \leq k$.

Como $P \in \mathcal{P}^{<k}$, e para todo $v \in P$, $v \in \mathcal{P}^{*1}$, então temos que $f(v_i', v_{i+1}'') = 0$. Então, $\rho(v_{i+1}'') \leq \rho(v_i') = \rho(v_i'') - 1$. Logo, dentro de P as cores estão em ordem estritamente decrescente, e são, portanto, distintas. \square

Lema 2.2.7. *Se não existem circuitos de custo negativo em R_f e $\rho(t) = k$, então $C^k = \{C_1, C_2, \dots, C_k\}$ é uma k -coloração parcial ortogonal a \mathcal{P} .*

Demonstração. Pelo lema 2.2.4, cada conjunto C_i de C^k é um conjunto independente, e pelos lemas 2.2.5 e 2.2.6, C^k é ortogonal a \mathcal{P} . \square

Teorema 2.2.1. *Seja \mathcal{P} uma partição em caminhos k -ótima em um grafo orientado acíclico G . Então, existe uma k -coloração que seja ortogonal a \mathcal{P} .*

Demonstração. Sejam k um inteiro positivo, e \mathcal{P} uma partição em caminhos k -ótima em um grafo acíclico G . A partir de G , obtemos a rede R , e a partir de \mathcal{P} , obtemos a partição \mathcal{P}^* transformando todos os caminhos curtos de \mathcal{P} em caminhos triviais, e o fluxo $f = f_{\mathcal{P}^*}$. Se existe um circuito negativo, ou se $\rho(t) < k$ em R_f , pelo lema 2.2.3 é possível obter uma

partição \mathcal{P}' tal que $|\mathcal{P}'|_k < |\mathcal{P}|_k$, contradizendo a hipótese que \mathcal{P} é uma partição k -ótima. Senão, $\rho(t) = k$ em R_f , e pelo lema 2.2.7 existe uma k -coloração parcial ortogonal a \mathcal{P} . \square

O exemplo ilustrado na figura 2.5 também tem como referência o grafo e a rede descritos na figura 2.1. O fluxo $f = f_{\mathcal{P}}$ é obtido a partir da partição de G em caminhos $\mathcal{P} = \{(a, b, c, d)\}$, e definimos $k = 2$.³ A partir de f , é construída a rede residual R_f . Como $\rho(t) = k$, é possível obter a 2-coloração parcial $C = \{\{d\}, \{c\}\}$ ortogonal a \mathcal{P} .

³Observe que, embora trate-se da mesma partição obtida no final do exemplo ilustrado pela figura 2.4, o fluxo é diferente ($f(s, t) = 0$, neste exemplo). Este exemplo deixa claro que $f_{\mathcal{P}(f)}$ não é necessariamente igual a f .

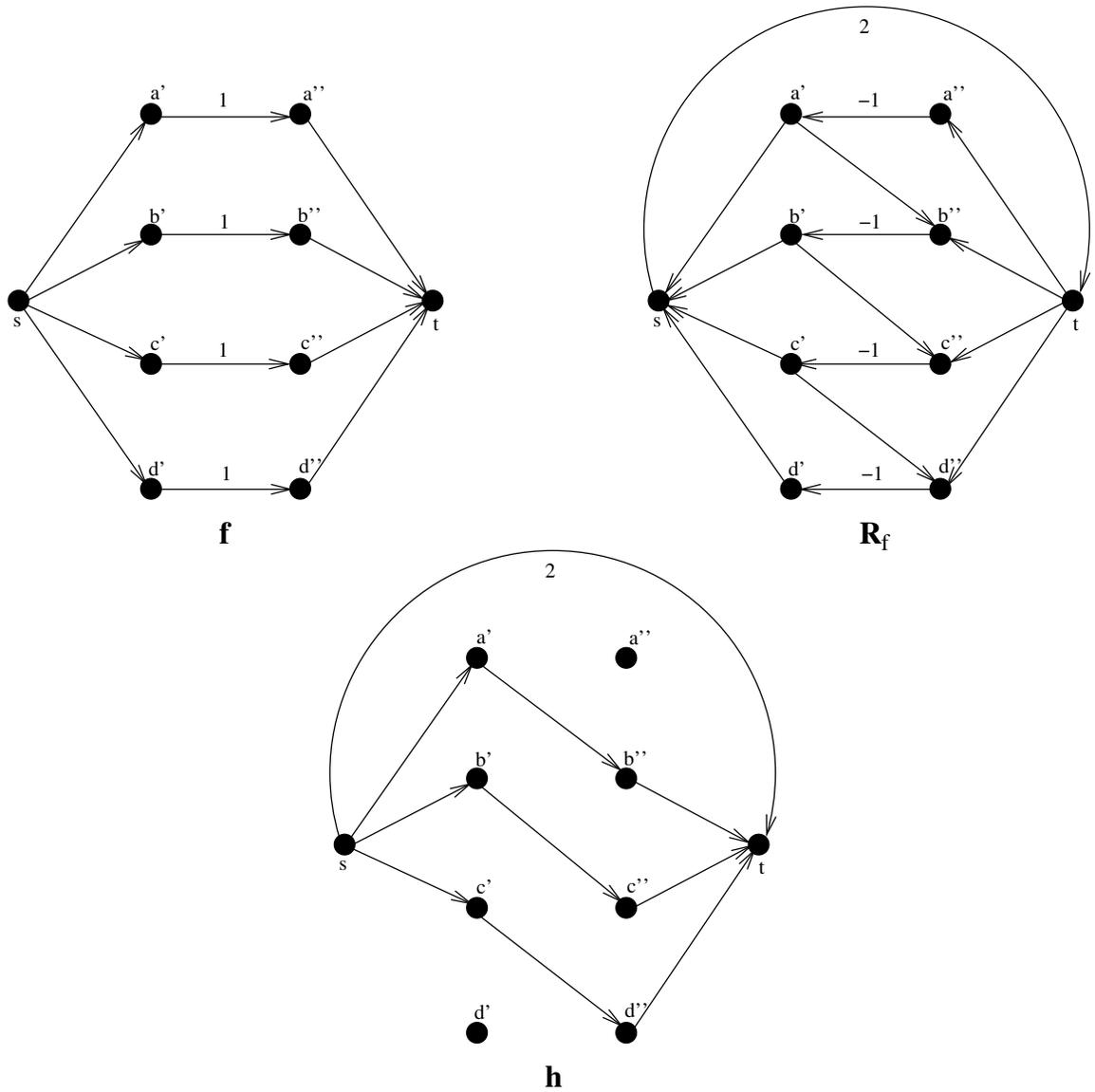


Figura 2.4: Esquerda: Fluxo $f = f_{\mathcal{P}}$ viável de R , proveniente de $\mathcal{P} = \{(a), (b), (c), (d)\}$. Somente as arestas com fluxo não-nulo ($f(e) = 1$) estão presentes na ilustração. Direita: Rede residual R_f . Abaixo: Novo fluxo $h = f + g$, onde g é o fluxo obtido em R_f a partir do circuito $C = (s, t, a'', a', b'', b', c'', c', d'', d', s)$. Os valores descritos nas arestas correspondem aos seus custos.

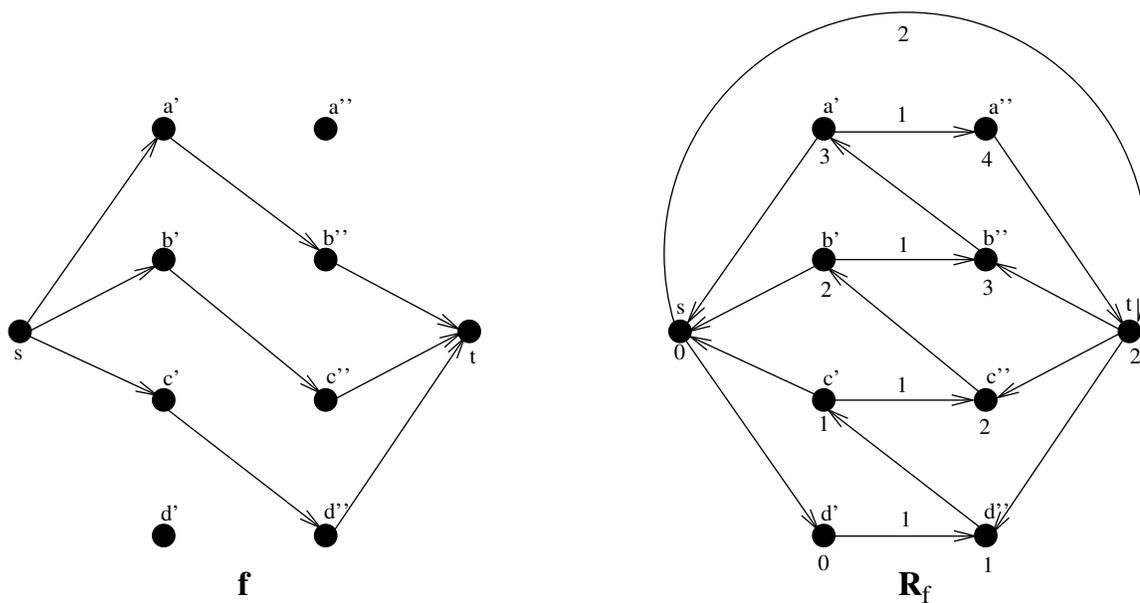


Figura 2.5: Novamente, o grafo e a rede na figura 2.1 foram usados de referência. Esquerda: O fluxo $f = f_{\mathcal{P}}$ é proveniente da partição $\mathcal{P} = \{(a, b, c, d)\}$. Direita: Rede residual R_f . Os valores descritos nas arestas correspondem aos seus custos e os valores descritos nos vértices correspondem aos potenciais.

Capítulo 3

Uma abordagem construtiva para a conjectura de Berge.

Em 2008, Berger e Hartman [23] publicaram uma demonstração construtiva para o caso especial da conjectura de Berge onde $k = 2$. Devotaremos os próximos dois capítulos para estudar a abordagem usada e a demonstração detalhadamente. Neste capítulo, exploraremos e ilustraremos o método envolvido, demonstrando outros dois casos particulares da conjectura: o teorema de Gallai-Milgram (também referenciado pelo texto como o “caso $k = 1$ ”), e o caso onde a partição analisada não contém caminhos curtos. No capítulo 4, o foco estará na demonstração do caso $k = 2$.

3.1 Visão geral da abordagem

Conjectura 3.1.1. (*Berge, 1982*)

Seja G um grafo orientado e k um inteiro positivo. Então, para toda partição em caminhos k -ótima de G , existe uma k -coloração parcial ortogonal a ela.

O artigo de Berger e Hartman [23] descreve uma técnica de demonstração construtiva, e o uso dessa técnica para os casos da conjectura onde $k = 1$ e $k = 2$. Para uma partição em caminhos k -ótima \mathcal{P} qualquer, deseja-se encontrar uma k -coloração parcial ortogonal a \mathcal{P} . Para isso, é descrito um algoritmo tal que, dado um grafo G , e uma partição em caminhos \mathcal{P} de G , devolve uma entre duas possíveis saídas: uma partição em caminhos \mathcal{P}' com norma menor que a de \mathcal{P} , ou uma k -coloração ortogonal a \mathcal{P} .

Note que quando a partição de entrada do algoritmo é k -ótima, este necessariamente devolve uma coloração parcial ortogonal. Logo, ao provarmos a existência e a corretude deste algoritmo, estamos também fornecendo uma demonstração construtiva da conjectura de Berge para o caso particular contemplado.

3.1.1 Trilhas alternantes

Podemos definir um objeto matemático semelhante a uma trilha, mas que ignore orientações de arestas. Uma trilha *não-dirigida* é uma sequência $T = (v_0, e_1, v_1, e_2, v_2, \dots, e_l, v_l)$ tal que, para cada $1 \leq i \leq l$, ou $e_i = (v_{i-1}, v_i) \in E(G)$, ou $e_i = (v_i, v_{i-1}) \in E(G)$, e todas as arestas são distintas. Para uma trilha não-dirigida, definimos uma direção de v_0 para v_l , de forma que se $e_i = (v_{i-1}, v_i)$ então e_i é considerada uma *aresta de avanço*, e se $e_i = (v_i, v_{i-1})$, e_i é considerada uma *aresta de recuo*. Para uma dada trilha T , denominamos o conjunto de arestas de avanço e de recuo de T como A_T e R_T , respectivamente. Se não houver ambiguidade quanto a definição das arestas de T como de avanço ou de recuo (como no caso das trilhas alternantes, descritas a seguir), podemos denotar $T = (v_0, e_1, v_1, e_2, v_2, \dots, e_l, v_l)$ como $T = (v_0, v_1, v_2, \dots, v_l)$.

Para modificarmos nossas partições durante o algoritmo, faremos uso de um tipo bem particular de trilha não-dirigida. Dada uma partição em caminhos \mathcal{P} , uma trilha não-dirigida T é *alternante* em relação a \mathcal{P} se valem as seguintes condições:

1. O primeiro vértice de T está em um terminal de \mathcal{P} .
2. Todas as arestas de avanço de T estão em $E(G) - E[\mathcal{P}]$, todas as arestas de recuo de T estão em $E[\mathcal{P}]$, e toda aresta de avanço (u, v) de T , onde $v \in V[\mathcal{P}^{>1}]$, é seguida por pelo menos uma aresta de recuo de T , a menos que $v \in in[\mathcal{P}^{>1}]$ e $v = ter(T)$.
3. Para cada vértice $v \in V(T)$, existe no máximo uma aresta de avanço de T entrando em v e no máximo uma aresta de avanço de T saindo de v .

Ao tratarmos de arestas de recuo em trilhas alternantes, por convenção usamos o sentido da aresta dentro da trilha, que é inverso ao sentido da aresta dentro do caminho. Por exemplo, podemos nos referir a uma aresta de recuo que percorre no sentido inverso a aresta $(v, v_{\mathcal{P}}^+)$ da partição \mathcal{P} como $(v_{\mathcal{P}}^+, v)$ (ou, na forma simplificada (v^+, v))¹.

A sequência de arestas de uma trilha alternante T pode ser representada pela seguinte estrutura: $T = R_1 a_1 R_2 a_2 \dots R_m a_m R_{m+1}$, onde a_i é uma aresta de avanço para $1 \leq i \leq m$ e R_i é uma sequência de uma ou mais arestas de recuo para $2 \leq i \leq m$, ou de zero ou mais arestas de recuo para $i \in \{1, m+1\}$.

Uma trilha alternante T é *completa* se $ter(T) \in in[\mathcal{P}]$. Um *circuito alternante* é uma trilha alternante completa tal que $in(T) = ter(T) \in \mathcal{P}^1$.

¹Por convenção, ao tratarmos da notação de sucessor (v^+) e antecessor (v^-) de um vértice dentro de uma trilha alternante em relação a uma partição em caminhos, estaremos sempre nos referindo ao antecessor/sucessor do vértice dentro do caminho ao qual esse pertence na partição em caminhos, a menos que seja especificado o contrário.

Dada uma partição em caminhos \mathcal{P} , denotamos por $\mathcal{P} \Delta T$ o subgrafo gerador de G com conjunto de arestas igual a diferença simétrica $E[\mathcal{P}] \Delta E(T) = (E[\mathcal{P}] \cup E(T)) \setminus (E[\mathcal{P}] \cap E(T))$.

Os dois teoremas seguintes servirão de base para as discussões subsequentes sobre os algoritmos.

Teorema 3.1.1. *Sejam \mathcal{P} uma partição em caminhos e T uma trilha alternante em relação a \mathcal{P} . O subgrafo $Q = \mathcal{P} \Delta T$ é um subgrafo gerador de G formado por circuitos e caminhos disjuntos.*

Demonstração. Por definição, $Q = \mathcal{P} \Delta T$ é um subgrafo gerador de G . Mostraremos que para todo vértice $v \in V(G)$, $d_Q^-(v) \leq 1$ e $d_Q^+(v) \leq 1$, de onde o resultado segue.

Se $v \notin V(T)$, então claramente $d_Q^-(v) \leq 1$ e $d_Q^+(v) \leq 1$. Assim, suponha que $v \in V(T)$. Pela definição, sabemos que existe no máximo uma aresta de avanço entrando (saindo) em (de) um dado vértice de T . Seja (u, v) uma aresta de avanço de T entrando em v . Se $v \in \text{in}[\mathcal{P}]$, então $d_{\mathcal{P}}^-(v) = 0$ e $d_Q^-(v) = 1$. Caso contrário, necessariamente (u, v) é seguida de uma aresta (v, v^-) de recuo em T e $d_{\mathcal{P}}^-(v) = d_Q^-(v) \leq 1$.

Um argumento similar vale para arestas de avanço saindo de v . Seja (v, w) uma aresta de avanço saindo de v . Se $v \in \text{ter}[\mathcal{P}]$, então $d_{\mathcal{P}}^+(v) = 0$ e $d_Q^+(v) = 1$. Caso contrário, necessariamente (v, w) foi precedida de uma aresta (v^+, v) de recuo em T e $d_{\mathcal{P}}^+(v) = d_Q^+(v) \leq 1$. \square

Note que não é necessário que o grafo gerado pela diferença simétrica entre uma trilha T e uma partição em caminhos \mathcal{P} também seja uma partição em caminhos. Em particular, $\mathcal{P} \Delta T$ pode conter circuitos. Na figura 3.1 que ilustra esta situação, para $\mathcal{P} = \{(a, b, c, d), (e, f, g, h), (i, j, k)\}$ e $T = (d, c, f, e, j, i, h, g, b, a)$, o circuito (g, b, c, f, g) é formado em $\mathcal{P} \Delta T$.

Teorema 3.1.2. *Sejam \mathcal{P} uma partição em caminhos e T uma trilha alternante. Se $\mathcal{P}' = \mathcal{P} \Delta T$ é acíclico, então \mathcal{P}' é uma partição em caminhos tal que $|\mathcal{P}'| - |\mathcal{P}| = |R_T| - |A_T|$.*

Demonstração. Como $\mathcal{P}' = \mathcal{P} \Delta T$ é acíclico, segue do teorema 3.1.1 que \mathcal{P}' é uma partição em caminhos. A igualdade segue da seguinte equação:

$$\begin{aligned}
|\mathcal{P}'| - |\mathcal{P}| &= (n - |E(\mathcal{P}')|) - (n - |E(\mathcal{P})|) \\
&= |E(\mathcal{P})| - |E(\mathcal{P}')| \\
&= |E(\mathcal{P})| - |E(\mathcal{P}) \Delta E(T)| \\
&= |E(\mathcal{P})| - (|E(\mathcal{P}) \cup E(T)| - |E(\mathcal{P}) \cap E(T)|) \\
&= |E(\mathcal{P})| - (|E(\mathcal{P})| + |A_T| - |R_T|) \\
&= |R_T| - |A_T|.
\end{aligned} \tag{3.1}$$

\square

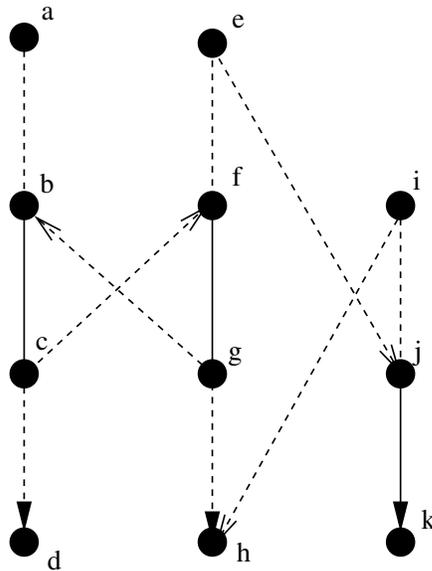


Figura 3.1: Exemplo de uma trilha alternante completa $T = (d, c, f, e, j, i, h, g, b, a)$ sobre uma partição \mathcal{P} , tal que $\mathcal{P} \Delta T$ não é acíclico.

3.2 O caso $k = 1$.

Adiante, descrevemos Part-k-1, um algoritmo que usa trilhas alternantes para demonstrar o teorema de Gallai-Milgram. Dados um grafo G , e uma partição \mathcal{P} de G , Part-k-1 encontra um conjunto independente X ortogonal a \mathcal{P} , ou uma trilha alternante completa T tal que $\mathcal{P}' = \mathcal{P} \Delta T$ é uma partição em caminhos e $|\mathcal{P}'| = |\mathcal{P}| - 1$.

3.2.1 Descrição do algoritmo

O algoritmo recebe como entrada uma partição em caminhos \mathcal{P} de um grafo G . A cada iteração, o algoritmo mantém um conjunto X , que é um candidato a um conjunto independente ortogonal a \mathcal{P} . Inicialmente, $X = \text{ter}[\mathcal{P}]$; se X é independente, o algoritmo devolve X e para. Senão, ele analisa outro possível candidato a X .

Além disso, para cada vértice $v \in X$, o algoritmo mantém uma trilha alternante $Q(v)$. As trilhas alternantes servem essencialmente como planos para a modificação de \mathcal{P} , e o objetivo do algoritmo, não encontrado um conjunto independente ortogonal a \mathcal{P} , é encontrar uma trilha alternante final T tal que $\mathcal{P}' = \mathcal{P} \Delta T$ seja uma partição em caminhos e $|\mathcal{P}'| = |\mathcal{P}| - 1$. As trilhas $Q(v)$ associadas a cada vértice são usadas como trilhas alternantes intermediárias para encontrarmos T , e cada vez que encontramos uma aresta em X , a informação da existência desta é usada para a atualização de uma das

trilhas intermediárias.

Part-k-1(\mathcal{P} , G)

Entrada

Um grafo G , e uma partição em caminhos \mathcal{P} de G .

Saída

Um conjunto independente X ortogonal a \mathcal{P} , ou uma partição \mathcal{P}' com $|\mathcal{P}'| = |\mathcal{P}| - 1$.

Início

$X \leftarrow \text{ter}[\mathcal{P}]$

Para todo vértice $v \in X$, $Q(v) \leftarrow \emptyset$

Repita:

0. Se X é um conjunto independente, então devolva X .
1. Senão, seja $e = (u, v)$ uma aresta incidente em dois vértices em X :
 - a) Se $v \notin \text{in}[\mathcal{P}]$:

$$X \leftarrow X + v^- - v$$

$$Q(v^-) \leftarrow Q(u) * (u, v, v^-)$$
 - b) Se $v \in \text{in}[\mathcal{P}]$:

$$T \leftarrow Q(u) * (u, v)$$
 Devolva $\mathcal{P}' = \mathcal{P} \Delta T$

Fim

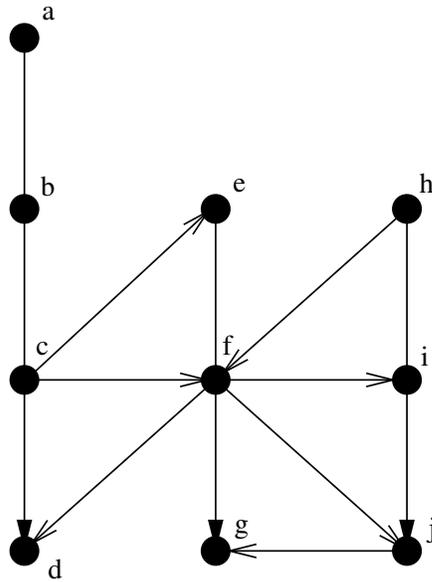


Figura 3.2: Grafo G , e a partição $\mathcal{P} = \{(a, b, c, d), (e, f, g), (h, i, j)\}$, que servem como exemplo para Part-k-1.

Servindo de exemplo, a partição ilustrada na figura 3.2 será usada como entrada para Part-k-1. Inicialmente, $X = \{d, g, j\}$, e a sequência de arestas examinadas é $((j, g), (f, j), (f, d), (f, i), (h, f), (c, e))$. Na primeira iteração, a aresta (j, g) é analisada, X é atualizado para $X = \{d, f, j\}$, e $Q(f)$ é definido como $Q(f) = (j, g, f)$. Na segunda iteração, a aresta (f, j) é analisada, e temos $X = \{d, f, i\}$, e $Q(i) = (j, g, f, j, i)$. Na terceira iteração, a aresta (f, d) é analisada, e temos $X = \{c, f, i\}$, e $Q(c) = (j, g, f, d, c)$. Na quarta iteração, a aresta (f, i) é analisada, e temos $X = \{c, f, h\}$, e $Q(h) = (j, g, f, i, h)$. Na quinta iteração, a aresta (h, f) é analisada, e temos $X = \{c, e, h\}$, e $Q(e) = (j, g, f, i, h, f, e)$. Com o exame da aresta (c, e) , é encontrada a trilha $T = (j, g, f, d, c, e)$. Aplicando a diferença simétrica $\mathcal{P} \Delta T$, é encontrada a partição em caminhos $\mathcal{P}' = \{(a, b, c, e, f, d), (h, i, j, g)\}$, que é devolvida pelo algoritmo.

Note que a ordem de exame das arestas influencia no resultado final do algoritmo. No exemplo, se o algoritmo examinasse a aresta (f, i) antes de (f, d) , a ordem de exame seria necessariamente $((j, g), (f, j), (f, i), (h, f))$, e o algoritmo devolveria $X = \{d, e, h\}$ como um conjunto independente ortogonal a \mathcal{P} .

3.2.2 Prova de correção do algoritmo

Para verificar a correção do algoritmo Part-k-1, mostraremos que este devolve um conjunto independente X ortogonal a \mathcal{P} , ou um subgrafo gerador de G , $\mathcal{P}' = \mathcal{P} \Delta T$, tal

que:

1. T é uma trilha alternante completa,
2. $|A_T| - |R_T| = 1$, e
3. $\mathcal{P}' = \mathcal{P} \Delta T$ é acíclico.

Note que, demonstradas essas propriedades de \mathcal{P}' e T , segue pelos teoremas 3.1.1 e 3.1.2 que \mathcal{P}' é uma partição em caminhos tal que $|\mathcal{P}'| = |\mathcal{P}| - 1$.

Propriedade 3.2.1. *Em Part- $k-1$, o conjunto X é ortogonal a \mathcal{P} .*

Demonstração. Inicialmente, $X = \text{ter}[\mathcal{P}]$ e a propriedade é válida. Para iterações posteriores, a propriedade continua valendo, já que a única operação que altera X ocorre no caso 1a, onde um vértice v de X é substituído pelo seu antecessor v^- , e X permanece como um conjunto ortogonal a \mathcal{P} . \square

Nas próximas demonstrações, mostraremos que \mathcal{P}' devolvido no caso 1b satisfaz as propriedades 1, 2 e 3.

Propriedade 3.2.2. *Para todo $v \in X$, temos que se $Q(v) \neq \emptyset$, então $Q(v)$ é uma trilha alternante que termina em v por uma aresta de recuo. Além disso, $|A_{Q(v)}| = |R_{Q(v)}|$.*

Demonstração. Na inicialização do algoritmo, para todo $v \in \text{ter}[\mathcal{P}]$, $Q(v) = \emptyset$ e a propriedade vale.

Suponha que a propriedade vale em uma iteração qualquer, e mostraremos que ela também vale na próxima. No caso 1a do algoritmo, v^- é colocado no conjunto X e $Q(v^-) = Q(u) * (u, v, v^-)$. Como o trecho (u, v, v^-) é formado por uma aresta de avanço e uma de recuo, e $|A_{Q(u)}| = |R_{Q(u)}|$ (já que $Q(u)$ satisfaz a propriedade), segue que $|A_{Q(v^-)}| = |R_{Q(v^-)}|$. Além disso, claramente $Q(v^-)$ é uma trilha alternante que termina em v^- , e a propriedade é preservada. \square

Lema 3.2.1. *No caso 1b, T é uma trilha alternante completa com $|A_T| - |R_T| = 1$.*

Demonstração. Pela propriedade 3.2.2, $Q(u)$ é uma trilha alternante e $|A_{Q(u)}| = |R_{Q(u)}|$. Pelo algoritmo, $T = Q(u) * (u, v)$. Como (u, v) é uma aresta de avanço, $|A_T| - |R_T| = 1$. Além disso, como $v \in \text{in}[\mathcal{P}]$, T é uma trilha alternante completa. \square

A seguir, provaremos que $\mathcal{P}' = \mathcal{P} \Delta T$ é acíclico (e, conseqüentemente, que \mathcal{P}' é uma partição). Vamos lançar mão da seguinte propriedade, claramente verdadeira pela execução do algoritmo.

Propriedade 3.2.3. *Considere uma iteração qualquer do algoritmo. Sejam u um vértice de X e $e = (w, x)$ uma aresta tal que $x \leq_{\mathcal{P}} u$. Então a aresta e não foi analisada pelo algoritmo até esta iteração, e não faz parte da trilha intermediária $Q(v)$ para nenhum vértice v .*

Propriedade 3.2.4. *Para todo $v \in X$, o subgrafo gerador de G com conjunto de arestas $E[Q(v)] \cup E[\mathcal{P}]$ é acíclico.*

Demonstração. Na inicialização do algoritmo, para todo $v \in X$, $Q(v) = \emptyset$ e a propriedade vale.

Suponha que a propriedade vale em uma iteração qualquer, e mostraremos que ela também vale na próxima. Suponha, por contradição, que ao analisarmos $e = (u, v)$ no caso 1a, um circuito seja formado em $E[Q(v^-)] \cup E[\mathcal{P}]$. Como $Q(v^-) = Q(u) * (u, v, v^-)$, isso implica que logo antes de (u, v) ser analisada, existia um caminho W de v até u com arestas em $E[Q(u)] \cup E[\mathcal{P}]$. Sejam P o caminho que contém u em \mathcal{P} , e (w, x) a última aresta de W que não pertence a Pu . Claramente, $x \leq_{\mathcal{P}} u$, o que entra em contradição com a propriedade 3.2.3. Logo, W não pode existir, e $E[Q(v^-)] \cup E[\mathcal{P}]$ é acíclico. \square

Lema 3.2.2. *Se o algoritmo termina no caso 1b, então o subgrafo $\mathcal{P}' = \mathcal{P} \Delta T$ é acíclico.*

Demonstração. No caso 1b, temos $T = Q(u) * (u, v)$. Pela propriedade 3.2.4, $E[\mathcal{P}] \cup E[Q(u)]$ é acíclico, e usando o mesmo argumento da demonstração deste lema, segue que não existe um caminho de v para u em $Q(u)$ com arestas em $E[Q(u)] \cup E[\mathcal{P}]$. Logo, $E[\mathcal{P}] \cup E[T]$ é acíclico. Como $E[\mathcal{P} \Delta T] \subseteq E[\mathcal{P}] \cup E[T]$, \mathcal{P}' é acíclico. \square

Os lemas 3.2.1 e 3.2.2, além da propriedade 3.2.1, demonstram que Part- $k-1$ é correto. Além disso, o algoritmo termina em tempo finito, já que cada iteração do tipo 1a coloca em X um vértice de G que não havia sido analisado anteriormente. Então, em no máximo $|V(G)|$ iterações, teremos $X = in[\mathcal{P}]$ e o algoritmo necessariamente finaliza na próxima iteração.

Logo, para uma partição 1-ótima, podemos sempre obter um conjunto independente ortogonal a ela, e a conjectura de Berge é verdadeira para $k = 1$.

Teorema 3.2.1. *O algoritmo Part- $k-1$ devolve uma partição em caminhos \mathcal{P}' de G tal que $|\mathcal{P}'| = |\mathcal{P}| - 1$, ou um conjunto independente X ortogonal a \mathcal{P} em tempo finito.*

Teorema 3.2.2. *(Gallai-Milgram, 1960)*

Se \mathcal{P} é uma partição em caminhos 1-ótima de G , então existe um conjunto independente ortogonal a \mathcal{P} .

3.3 O caso das partições longas

Nesta última seção, vamos descrever um algoritmo de nossa autoria que, dada uma partição em caminhos \mathcal{P} de um grafo G e um natural não-nulo k fixo, devolve uma outra partição em caminhos com k -norma menor que a de \mathcal{P} , ou uma k -coloração parcial ortogonal a \mathcal{P} . A única pré-condição para \mathcal{P} é que todos os seus caminhos sejam longos para k . De agora em diante, chamaremos de *longa* uma partição em caminhos cujos caminhos sejam todos longos em relação a k .

Note que se \mathcal{P} é uma partição k -ótima que também é longa, então o algoritmo devolve uma k -coloração parcial que é ortogonal a \mathcal{P} . Provada a corretude deste, também estará provado que a seguinte proposição é verdadeira:

Teorema 3.3.1. *Seja \mathcal{P} uma partição k -ótima de G que é longa. Então, existe uma k -coloração parcial de G que é ortogonal a \mathcal{P} .*

Este é tanto um caso específico da conjectura de Berge, quanto uma generalização do teorema de Gallai-Milgram, já que para $k = 1$, podemos considerar todos os caminhos de \mathcal{P} como longos. No entanto, este teorema não implica na conjectura de Berge, já que claramente existem grafos que admitem partições k -ótimas (para algum k) que não são longas.

3.3.1 Descrição do algoritmo

O algoritmo Part-Longos recebe como entrada um grafo G , uma partição em caminhos \mathcal{P} deste grafo, e $k \in \mathbb{N}^*$. Em cada iteração do algoritmo temos um índice i , tal que $1 \leq i \leq k$. O algoritmo analisa um conjunto de vértices X_i , que é candidato a um conjunto independente ortogonal a \mathcal{P} . Se X_i for de fato independente (caso 0a), armazenamos X_i até o final da execução do algoritmo e tentamos encontrar o próximo conjunto independente X_{i+1} , tomando como primeiro candidato o conjunto dos vértices imediatamente anteriores aos vértices de X_i em \mathcal{P} . Caso consiga encontrar o k -ésimo conjunto independente (caso 0b), o algoritmo devolve a coleção $X = \{X_1, X_2, \dots, X_k\}$ de conjuntos independentes disjuntos ortogonais a \mathcal{P} como uma k -coloração parcial ortogonal a \mathcal{P} .

Concomitante à busca do algoritmo por X_k , esse constrói trilhas alternantes parciais $Q(v)$ para cada vértice analisado, com a finalidade de encontrar uma trilha alternante final T , como em Part-k-1.

O algoritmo pode terminar antes de X alcançar $in[\mathcal{P}]$ porque (como demonstraremos adiante, no lema 3.3.5) esse encontra uma trilha alternante completa T tal que $\mathcal{P}' = \mathcal{P} \Delta T$ é uma partição em caminhos com $|\mathcal{P}'|_k = |\mathcal{P}|_k - 1$.

Part-Longos(\mathcal{P} , k , G)**Entrada**

Um grafo G , um inteiro k , e uma partição em caminhos \mathcal{P} longa em relação a k .

Saída

Uma k -coloração parcial ortogonal a \mathcal{P} , ou uma partição \mathcal{P}' com $|\mathcal{P}'|_k = |\mathcal{P}|_k - 1$.

Início

$X_1 \leftarrow \text{ter}[\mathcal{P}]$.

Para todo $v \in X_1$, $Q(v) \leftarrow \emptyset$.

$i \leftarrow 1$.

Repita:

0. Se X_i é um conjunto independente:

a) Se $i < k$:

$X_{i+1} \leftarrow \emptyset$

Para todo $v \in X_i$, faça:

$X_{i+1} \leftarrow X_{i+1} + v^-$

$Q(v^-) \leftarrow Q(v) * (v, v^-)$

$i \leftarrow i + 1$

b) Senão:

Devolva $X = \{X_1, X_2, \dots, X_k\}$.

1. Senão, seja $e = (u, v)$ uma aresta incidente em dois vértices em X_i , e seja P o caminho de \mathcal{P} que contém v :

a) Se $|Pv^-| \geq k - i + 1$:

$X_i \leftarrow X_i + v^- - v$

$Q(v^-) \leftarrow Q(u) * (u, v, v^-)$

b) Senão:

$T = Q(u) * (u, v) * (v, v^-, \dots, \text{in}(P))$

Devolva $\mathcal{P}' = \mathcal{P} \Delta T$

Fim

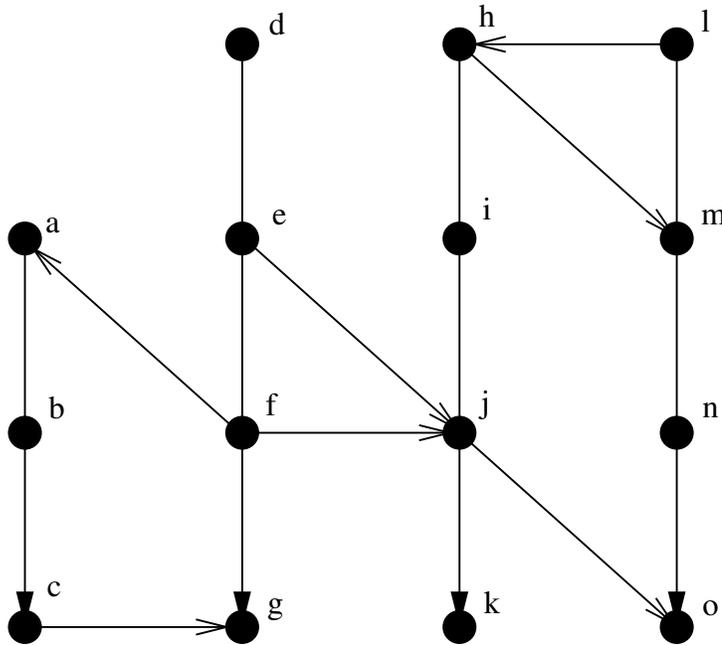


Figura 3.3: Grafo G , e a partição $\mathcal{P} = \{(a, b, c), (d, e, f, g), (h, i, j, k), (l, m, n, o)\}$, que servem como exemplo para Part-Longos.

A partição ilustrada na figura 3.3 serve como exemplo para Part-Longos para $k = 3$. Inicialmente, $i = 1$, e o primeiro candidato a conjunto independente transversal a \mathcal{P} é $X_1 = \{c, g, k, o\}$. A aresta (c, g) é examinada na primeira iteração pelo caso 1a, X_1 é atualizado para $X_1 = \{c, f, k, o\}$ e $Q(f) = (c, g, f)$. Na segunda iteração, X_1 é confirmado como um conjunto independente pelo caso 0a do algoritmo. Além disso, i é atualizado para $i = 2$, e X_2 passa a ser $\{b, e, j, n\}$. Todos os vértices de X_2 têm a sua trilha intermediária definida: $Q(b) = (c, b)$, $Q(e) = (c, g, f, e)$, $Q(j) = (k, j)$ e $Q(n) = (o, n)$. Na terceira iteração, (e, j) é analisada pelo caso 1a, X_2 é atualizado para $\{b, e, i, n\}$ e $Q(i) = (c, g, f, e, j, i)$. Na quarta iteração, $X_2 = \{b, e, i, n\}$ é confirmado como um conjunto independente pelo caso 0a, i é incrementado e $X_3 = \{a, d, h, m\}$ é o novo candidato a conjunto independente, com $Q(a) = (c, b, a)$, $Q(d) = (c, g, f, e, d)$, $Q(h) = (c, g, f, e, j, i, h)$ e $Q(m) = (o, n, m)$. Na quinta iteração, (h, m) é analisada pelo caso 1a, e ao final da iteração, X_3 é atualizado para $X_3 = \{a, d, h, l\}$ e $Q(l) = (c, g, f, e, j, i, h, m, l)$. Na última iteração, é analisada a aresta (l, h) , o algoritmo constrói a trilha $T = (c, g, f, e, j, i, h, m, l, h)$ pelo caso 1b, e devolve $\mathcal{P}' = \mathcal{P} \Delta T = \{(a, b, c, g), (d, e, j, k), (l, h, m, n, o), (f), (i)\}$, composta de um caminho longo a menos, e dois caminhos triviais a mais que \mathcal{P} .

Para uma outra execução de Part-Longos com a mesma partição em caminhos \mathcal{P} , o mesmo k , e um outro grafo de entrada $G' = G - (m, h)$, o algoritmo devolveria a 3-coloração parcial $X = \{c, f, k, o\}, \{b, e, i, n\}, \{a, d, h, l\}$ pelo caso 0b.

3.3.2 Prova de correção do algoritmo

Para provarmos a correção de Part-Longos, devemos provar que no caso $0b$, X é uma k -coloração parcial ortogonal a \mathcal{P} , e que \mathcal{P}' devolvida no caso $1b$ é uma partição em caminhos tal que $|\mathcal{P}'|_k = |\mathcal{P}|_k - 1$.

A seguinte propriedade é claramente verdadeira, já que quando o algoritmo determina que X_i é um conjunto independente, esse permanece inalterado pelo resto da execução do algoritmo.

Propriedade 3.3.1. *Para todo $h < i$, X_h é um conjunto independente.*

Demonstraremos também que esses conjuntos são ortogonais a \mathcal{P} .

Propriedade 3.3.2. *Para todo $h \leq i$, X_h é um conjunto ortogonal a \mathcal{P} .*

Demonstração. Durante a inicialização, $i = 1$, $X_1 = \text{ter}[\mathcal{P}]$, e a propriedade vale. Suponha que a propriedade vale em uma iteração qualquer, e mostraremos que ela também vale para a próxima. Para o caso $0a$, é construído um novo conjunto X_{i+1} , formado de antecessores de elementos de X_i , que é claramente ortogonal a \mathcal{P} . Para o caso $1a$, um vértice v de X_i é substituído por seu antecessor v^- . Para os conjuntos X_g com $g < i$, eles são inalterados. Portanto, a propriedade vale. \square

As propriedades 3.3.1 e 3.3.2 têm como resultado o lema 3.3.1.

Lema 3.3.1. *Ao final do caso $0b$, $X = \{X_1, X_2, \dots, X_k\}$ é uma k -coloração parcial ortogonal a \mathcal{P} .*

Antes de provarmos que $|\mathcal{P}'|_k = |\mathcal{P}|_k - 1$, onde \mathcal{P}' é a partição devolvida ao final do algoritmo no caso $1b$, demonstraremos que a trilha alternante construída por Part-Longos respeita uma série de propriedades.

Propriedade 3.3.3. *Se $v \in X_i$, então $Q(v) = \emptyset$, ou $Q(v)$ é uma trilha alternante que termina em uma aresta de recuo, e $|R_{Q(v)}| - |A_{Q(v)}| = i - 1$.*

Demonstração. Na inicialização do algoritmo, $i = 1$, e $Q(v) = \emptyset$ para todo $v \in X_1$. Logo, a propriedade inicialmente vale para todo $v \in X_1$.

Suponha que a propriedade vale em uma iteração qualquer, e mostraremos que ela também vale para a próxima. Os casos $0a$ e $1a$ são os únicos casos do algoritmo onde para um dado vértice v , $Q(v)$ é atualizado. No caso $1a$, quando v^- é inserido em X_i , se essa propriedade vale para $Q(u)$, ela também vale para $Q(v^-) = Q(u) * (u, v, v^-)$, pois em $Q(v^-)$ há uma aresta de avanço e uma de recuo a mais que em $Q(u)$. No caso $0a$, se o vértice w de X_i preserva essa propriedade, o vértice w^- de X_{i+1} também a preserva, pois $Q(w^-) = Q(w) * (w, w^-)$ tem uma aresta de recuo a mais que $Q(w)$ e $|R_{Q(w^-)}| - |A_{Q(w^-)}| = (i - 1) + 1$. Logo, a propriedade vale. \square

Lema 3.3.2. *Ao ocorrer o caso 1b, $T = Q(u) * (u, v) * (v, v^-, \dots, in[P])$ é uma trilha alternante própria e completa com $|R_T| - |A_T| = k - 2$.*

Demonstração. Como P é um caminho de \mathcal{P} , claramente T é completa. Pela propriedade 3.3.3, temos que $Q(u)$ é uma trilha alternante com $|R_{Q(u)}| - |A_{Q(u)}| = i - 1$, e no trecho $(v, v^-, \dots, in[P])$ existem exatamente $k - i + 1$ vértices (já que o caso 1b tem como hipótese que $|Pv^-| = k - i$), e, portanto, $k - i$ arestas de recuo. Logo, $|R_T| - |A_T| = (i - 1) - 1 + (k - i) = k - 2$. \square

É análoga ao caso $k = 1$ a demonstração que $\mathcal{P}' = \mathcal{P} \Delta T$ não gera circuitos. Os seguintes lemas e invariantes correspondem às propriedades 3.2.3, 3.2.4, ao lema 3.2.2 da demonstração para o caso $k = 1$, e são provados da mesma forma.

Propriedade 3.3.4. *Considere uma iteração qualquer do algoritmo. Seja u um vértice de X_i e $e = (w, x)$ uma aresta tal que $x \leq_p u$. Temos que e não foi analisada pelo algoritmo até esta iteração, e portanto não faz parte da trilha intermediária $Q(v)$ para nenhum vértice $v \in X_i$.*

Propriedade 3.3.5. *Para todo $v \in X_j$, o subgrafo gerador de G com conjunto de arestas $E[Q(v)] \cup E[\mathcal{P}]$ é acíclico.*

Lema 3.3.3. *Sejam T a trilha alternante construída, e $\mathcal{P}' = \mathcal{P} \Delta T$ o subgrafo gerador de G devolvido por Part-Longos no caso 1b. Temos que \mathcal{P}' é acíclico e, portanto (pelo teorema 3.1.1), uma partição em caminhos.*

Antes de demonstrarmos as propriedades seguintes, vamos estabelecer algumas notações. Definimos $\mathcal{P}(v) = \mathcal{P} \Delta Q(v)$ para qualquer vértice v de G com $Q(v) \neq \emptyset$. Para $w, z \in G$, definimos $P(w, z)$ como o caminho da partição $\mathcal{P}(z)$ que contém w .

Propriedade 3.3.6. *Sejam $w \in X_i$ um vértice de G , e P o caminho de \mathcal{P} que contém w . Então, temos que $|Pw| \geq k - i + 1$.*

Demonstração. Inicialmente, a propriedade vale já que para todo $w \in X_1$, temos que $w \in ter[\mathcal{P}]$, e $|Pw| \geq k$ (já que todos os caminhos são longos).

Suponha que a propriedade vale em uma iteração qualquer, e mostraremos que ela também vale para a próxima. Após uma iteração do caso 1a, a propriedade claramente vale, já que $|Pw^-| \geq k - i + 1$ neste caso. Após uma iteração do caso 0a, seja w^- um vértice arbitrário inserido em X e P o caminho de w^- . Então, temos que $|Pw^-| = |Pw| - 1$, e i tem seu valor atualizado para $i + 1$. Como antes da iteração, $|Pw| \geq k - i + 1$, após a iteração, $|Pw^-| \geq k - (i + 1)$, e a propriedade é verdadeira. \square

Propriedade 3.3.7. Para vértices w e z de X_i , temos que $|P(w, z)w| \geq k - i + 1$.

Demonstração. Seja W o caminho de \mathcal{P} que contém w . Para provarmos que $|P(w, z)w| \geq k - i + 1$, é suficiente notar que para qualquer trilha alternante intermediária $Q(z)$ com $z \in X_i$, temos que $P(w, z)w = Ww$. Isso é verdade, já que $E[Q(z)]$ é disjunto de $E[Ww]$ para todo $z \in X_i$. Pela propriedade 3.3.6, $|Ww| \geq k - i + 1$, e a propriedade é verdadeira. \square

Propriedade 3.3.8. Para vértices distintos w e z de X_i , temos que $|wP(w, z)| \geq i$.

Demonstração. Na inicialização a propriedade vale, já que inicialmente $i = 1$.

Suponha que a propriedade vale em uma iteração qualquer, e mostraremos que ela também vale para a próxima. Antes da atualização ela vale para todo $P(w, z)$ tal que $w, z \in X_i$ e $w \neq z$. Ao atualizarmos o algoritmo pelo caso 1a com uma aresta $e = (a, b)$, temos que $Q(b^-) \leftarrow Q(a) * (a, b, b^-)$ e $X_i \leftarrow X_i \cup \{b^-\} \setminus \{b\}$. Após a atualização, para w e z vértices distintos pertencentes à nova instância de X_i , temos que:

- Para todo $P(w, z)$ onde $w \neq b^-$ e $z \neq b^-$, a propriedade continua valendo já que o caminho que contém w em $\mathcal{P}(z)$ permanece inalterado.
- Para $P(b^-, z)$, onde $z \neq b^-$, a partição $\mathcal{P}(z)$ é inalterada em relação à iteração anterior, $b^-P(b^-, z) = (b^-, b) * bP(b, z)$ e $bP(b, z) \geq i$. Logo $|b^-P(b^-, z)| \geq i + 1$.
- Para $P(w, b^-)$ onde $w \notin \{a, b^-\}$, $wP(w, b^-) = wP(w, a)$. Como a propriedade vale para w e a , $|wP(w, b^-)| \geq i$.
- Para $P(a, b^-)$, temos que $aP(a, b^-) = (a, b) * bP(b, a)$ e $bP(b, a) \geq i$. Logo, $|aP(a, b^-)| \geq i + 1$.

Logo, a propriedade é preservada ao ocorrer atualizações do caso 1a. Para o caso 0a, para $P(w, z)$ tal que $w, z \in X_{i+1}$ e $w \neq z$, como $Q(z) \leftarrow Q(z^+) * (z^+, z)$, e z e w pertenciam a caminhos disjuntos antes da atualização, temos que $wP(w, z) = (w, w^+) * w^+P(w^+, z)$. Como $w^+P(w^+, z) \geq i$, $wP(w, z) \geq i + 1$ e a propriedade vale. \square

Note que a propriedade 3.3.8 só se preserva durante a execução do algoritmo quando w e z são distintos. Isso é claro, já que para $v \in X_i$, após um caso de atualização 0a temos que $|v^-P(v^-, v^-)| = 1$.

Propriedade 3.3.9. Para todo $v \in X_i$, $v \in \text{ter}[\mathcal{P}(v)]$.

Demonstração. Para todo $v \in X_i$, necessariamente ou $Q(v) = \emptyset$ e $v \in \text{ter}[\mathcal{P}]$, ou $Q(v)$ termina em uma aresta de recuo (v^+, v) , e pela propriedade 3.3.4, não há arestas de avanço em $Q(v)$ saindo de v . Logo, a propriedade é válida. \square

Propriedade 3.3.10. *Seja $v \in X_i$. Então, todas as arestas de avanço em $Q(v)$ fazem parte de algum caminho longo em $\mathcal{P}(v)$. Além disso, todas as arestas de avanço em T fazem parte de algum caminho longo em $\mathcal{P}' = \mathcal{P} \Delta T$.*

Demonstração. Na inicialização, a propriedade vale, já que $Q(v) = \emptyset$ para $v \in X_1$. Suponha que a propriedade vale em uma iteração qualquer, e mostraremos que ela também vale para a próxima. Ao ocorrer uma iteração do caso 0a, para um vértice v^- que entra em X_{i+1} , nenhuma aresta de avanço nova é adicionada em $Q(v^-) = Q(v) * (v, v^-)$. Quanto às arestas de avanço que estavam originalmente em $Q(v)$, elas continuam pertencendo a algum caminho longo em $\mathcal{P}(v^-)$, já que pela propriedade 3.3.9, $v \in \text{ter}[\mathcal{P}(v)]$, e a única diferença entre $\mathcal{P}(v)$ e $\mathcal{P}(v^-)$ é o particionamento do caminho $P(v, v)$ no caminho longo $P(v, v)v^-$ e no caminho trivial (v) pela remoção da aresta (v^-, v) .

Se a atualização ocorrida for do caso 1a, suponha que a aresta $e = (a, b)$ seja examinada e, portanto, $Q(b^-) \leftarrow Q(a) * (a, b, b^-)$. Pela propriedade 3.3.7 $|P(a, b^-)a| \geq k - i + 1$, e pela propriedade 3.3.8, como $a \neq b$, $|aP(a, b^-)| \geq i$. Então, $|P(a, b^-)| \geq k$, e (a, b) faz parte de um caminho longo em $\mathcal{P}(b^-)$. Para todas as outras arestas de avanço (g, h) de $Q(b^-)$, basta notar que antes da atualização, pela propriedade 3.3.9, $g \in \text{ter}[\mathcal{P}(a)]$. Logo, os caminhos que já incluíam arestas de avanço em $\mathcal{P}(a)$ permanecem inalterados em $\mathcal{P}(b^-)$.

A trilha T claramente satisfaz a propriedade, já que ela vale para todos os vértices de X_i antes da iteração final. O argumento é análogo ao do parágrafo anterior. \square

Lema 3.3.4. *Seja T a trilha alternante encontrada no final do algoritmo. Então a partição $\mathcal{P}' = \mathcal{P} \Delta T$ é composta somente de caminhos longos e caminhos triviais.*

Demonstração. Suponha, por contradição, que exista $P \in \mathcal{P}'$ tal que $2 \leq |P| \leq k - 1$. Em \mathcal{P} , todos os caminhos são longos, e pela propriedade 3.3.10, todas as arestas de avanço de T pertencem a caminhos longos em $\mathcal{P}' = \mathcal{P} \Delta T$. Logo, P deve ser um subcaminho de um caminho $W = (v_1, v_2, \dots, v_l) \in \mathcal{P}$. Usando a mesma numeração dos vértices de W , sejam v_g o vértice inicial, e v_h o vértice final de P para $1 \leq g < h \leq l$. Então, necessariamente existe em T uma aresta de recuo $(v_g, (v_g)_W^-)$, ou uma aresta de recuo $((v_h)_W^+, v_h)^2$.

Como T é completa e $v_h \notin \text{in}[\mathcal{P}]$, $((v_h)_W^+, v_h)$ não pode ser a última aresta de T . Se $((v_h)_W^+, v_h)$ é sucedida em T por uma aresta de recuo $(v_h, (v_h)_W^-)$, então o caminho P não pode terminar em v_h . Se $((v_h)_W^+, v_h)$ é sucedida em T por uma aresta de avanço, então P é um caminho longo pela propriedade 3.3.10. Em ambos os casos obtemos uma contradição.

É análoga a argumentação para o caso em que existe a aresta de recuo $(v_g, (v_g)_W^-)$ em T . Se $(v_g, (v_g)_W^-)$ é precedida por uma aresta de recuo $((v_g)_W^+, v_g)$, então o caminho

²É útil lembrar de que $(v_h)_W^+$ se refere ao sucessor de v_h dentro do caminho W , conforme descrito na seção de notações do capítulo 1.

P não pode começar em v_g . Se $(v_g, (v_g)_W^-)$ é precedida em T por uma aresta de avanço, então pela propriedade 3.3.10 P é um caminho longo. Logo, a suposição de que existe um caminho curto não-trivial em \mathcal{P}' leva a uma contradição e o lema vale. \square

Lema 3.3.5. *Seja T a trilha alternante encontrada no caso 1b do algoritmo. Para $\mathcal{P}' = \mathcal{P} \Delta T$, temos que $|\mathcal{P}'|^{\geq k} = |\mathcal{P}|^{\geq k} - 1$.*

Demonstração. Seja $T = (v_1, v_2, \dots, v_l)$ a trilha obtida pelo algoritmo e $\mathcal{P}' = \mathcal{P} \Delta T$. Mostraremos que $|\text{in}[\mathcal{P}'^{\geq k}]| = |\text{in}[\mathcal{P}^{\geq k}]| - 1$ e $|\text{ter}[\mathcal{P}'^{\geq k}]| = |\text{ter}[\mathcal{P}^{\geq k}]| - 1$.

Como T é uma trilha alternante completa, $v_1 \in \text{ter}[\mathcal{P}]$ e $v_l \in \text{in}[\mathcal{P}]$. É fácil ver que v_1 deixa de ser um terminal de um caminho longo e v_l deixa de ser um inicial de um caminho longo em \mathcal{P}' .

Seja v_i um vértice de T para $2 \leq i \leq l - 1$. Se $v_i \in \text{ter}[\mathcal{P}]$, temos que (v_{i-1}, v_i) é uma aresta de avanço, (v_i, v_{i+1}) uma aresta de recuo, e pela propriedade 3.3.10 v_i é o terminal de um caminho longo em \mathcal{P}' . Analogamente, se $v_i \in \text{in}[\mathcal{P}]$, (v_i, v_{i+1}) é uma aresta de avanço, necessariamente (v_{i-1}, v_i) é uma de recuo, e (também pela propriedade 3.3.10) ele é o inicial de um caminho longo em \mathcal{P}' . Se $v_i \notin \text{ter}[\mathcal{P}] \cup \text{in}[\mathcal{P}]$, então:

- Se (v_{i-1}, v_i) é uma aresta de avanço e (v_i, v_{i+1}) uma de recuo, ou se (v_{i-1}, v_i) é uma aresta de recuo e (v_i, v_{i+1}) uma de avanço, v_i pertence a um caminho longo sem ser um inicial ou terminal em \mathcal{P}' .
- Se (v_{i-1}, v_i) e (v_i, v_{i+1}) forem ambas arestas de recuo, então v_i é um caminho trivial em \mathcal{P}' .

Assim sendo, vértices intermediários de T não alteram sua condição de serem ou não iniciais ou terminais de caminhos longos, e o lema é válido. \square

Os lemas de 3.3.1 a 3.3.5 são suficientes para demonstrarmos a corretude do algoritmo.

Teorema 3.3.2. *O algoritmo Part-Longos devolve em tempo finito uma partição em caminhos \mathcal{P}' de G tal que $|\mathcal{P}'|_k = |\mathcal{P}|_k - 1$, ou uma k -coloração parcial ortogonal a \mathcal{P} .*

Demonstração. Se o algoritmo terminar pelo caso 0b, pelo lema 3.3.1 o conjunto X devolvido é uma k -coloração parcial ortogonal a \mathcal{P} .

Se, ao contrário, o algoritmo terminar pelo caso 1b, para a trilha T calculada ao final do algoritmo, pelo lema 3.3.3 temos que $\mathcal{P}' = \mathcal{P} \Delta T$ é uma partição em caminhos. Para a partição final \mathcal{P}' dada como resultado do algoritmo, podemos calcular $|\mathcal{P}'|_k$ da seguinte forma:

$$\begin{aligned}
|\mathcal{P}'|_k &= k|\mathcal{P}'^{\geq k}| + |V[\mathcal{P}'^{<k}]| \\
&= (k-1)|\mathcal{P}'^{\geq k}| + (|\mathcal{P}'^{\geq k}| + |V[\mathcal{P}'^{<k}]|) \\
&= (k-1)|\mathcal{P}'^{\geq k}| + (|\mathcal{P}'^{\geq k}| + |\mathcal{P}'^{<k}|) \\
&= (k-1)|\mathcal{P}'^{\geq k}| + |\mathcal{P}'|
\end{aligned} \tag{3.2}$$

A terceira linha da equação acima segue pelo lema 3.3.4. Também podemos arranjar o cálculo da k -norma da partição \mathcal{P} dada como entrada ao algoritmo da mesma forma:

$$\begin{aligned}
|\mathcal{P}|_k &= k|\mathcal{P}^{\geq k}| \\
&= (k-1)|\mathcal{P}^{\geq k}| + |\mathcal{P}^{\geq k}| \\
&= (k-1)|\mathcal{P}^{\geq k}| + |\mathcal{P}|
\end{aligned} \tag{3.3}$$

Temos pelo lema 3.3.5 que $|\mathcal{P}'^{\geq k}| - |\mathcal{P}^{\geq k}| = -1$, e pelo lema 3.3.2 em conjunto com o teorema 3.1.2 que $|\mathcal{P}'| - |\mathcal{P}| = k - 2$. Assim sendo, a diferença entre $|\mathcal{P}'|_k$ e $|\mathcal{P}|_k$ pode ser calculada da seguinte forma:

$$\begin{aligned}
|\mathcal{P}'|_k - |\mathcal{P}|_k &= (k-1)(|\mathcal{P}'^{\geq k}| - |\mathcal{P}^{\geq k}|) + (|\mathcal{P}'| - |\mathcal{P}|) \\
&= (-k+1) + (k-2) \\
&= -1
\end{aligned} \tag{3.4}$$

Logo, $|\mathcal{P}'|_k = |\mathcal{P}|_k - 1$. Em uma execução qualquer do algoritmo, supondo que esse não termine antes, teremos $X_k = in[\mathcal{P}]$ em no máximo $V(G)$ iterações. Logo, claramente o algoritmo termina em tempo finito, e ele é correto. \square

Assim sendo, a conjectura de Berge é verdadeira para partições em caminhos que sejam compostas somente de caminhos longos.

Teorema 3.3.3. *Se \mathcal{P} é uma partição em caminhos k -ótima de G , e todos os caminhos de \mathcal{P} são longos para k , então existe uma k -coloração parcial ortogonal a \mathcal{P} .*

Capítulo 4

O Caso $k = 2$.

4.1 Introdução

Neste capítulo estudaremos o seguinte caso particular da conjectura de Berge:

Teorema 4.1.1. (*Berger-Hartman, 2008*)

Se \mathcal{P} é uma partição em caminhos 2-ótima de G , então existe uma 2-coloração parcial ortogonal a \mathcal{P} .

Doravante, por simplicidade, chamamos este caso particular da conjectura de *caso $k = 2$* . A demonstração de Berger e Hartman segue a idéia geral das provas construtivas apresentadas no capítulo anterior. Entretanto, o problema para o caso $k = 2$ mostra uma complexidade combinatória muito maior do que para os outros dois casos, e o principal determinante dessa complexidade é a presença de caminhos curtos e longos na mesma partição. A heterogeneidade do peso entre caminhos curtos e caminhos longos, além do potencial para variação muito maior na estrutura das partições que vem a surgir, tornam muito mais complexa uma possível abordagem para todos os casos genéricos onde $k > 1$, inclusive para este.

Em uma partição em caminhos, são considerados curtos pela 2-norma apenas os caminhos triviais. Para uma partição em caminhos \mathcal{P} de G , denotamos por $\mathcal{P}^{>1}$ o conjunto dos caminhos longos (com cardinalidade maior ou igual a 2) em \mathcal{P} , e por \mathcal{P}^1 o conjunto dos caminhos triviais em \mathcal{P} .

No restante do capítulo, descreveremos o algoritmo Part-k-2 que fornece uma prova construtiva do teorema 4.1.1.

4.2 Definições

Antes de expormos o algoritmo, definiremos alguns conceitos importantes.

4.2.1 Predecessor

Em Part-k-2, não é o suficiente ter somente a informação das trilhas alternantes intermediárias. É necessário também manter informações sobre as alterações feitas nas trilhas intermediárias (assim como nos candidatos a conjunto independente X) ocorridas a cada iteração do algoritmo. O algoritmo guarda essas informações através de um parâmetro intermediário chamado de *predecessor*. Após cada iteração do algoritmo, os predecessores de alguns dos vértices envolvidos na iteração são atualizados. O predecessor é um parâmetro do algoritmo associado a um vértice, e o predecessor de um vértice determina a qual trilha alternante (e a qual vértice) a trilha alternante de v está associada.

Para um dado vértice v , denotamos por $p(v)$ o predecessor de v . Os valores $p(v)$ dependem da sequência de passos do algoritmo. Mais tarde, explicaremos como eles são inicializados e atualizados. Podemos interpretar¹ p como uma função $p : V(G) \rightarrow V(G) \cup \{\lambda\} \cup \{\odot\}$. Se $p(v) = \odot$, dizemos que o predecessor de v é *indefinido*. Se $p(v) = \lambda$, o predecessor de v é definido, porém *vazio*. Definimos a função inversa² de p como $p^{-1}(v) = \{u \in V(G) : p(u) = v\}$.

A partir do predecessor, definimos alguns conjuntos igualmente importantes. Para um vértice v tal que $p(v) \neq \odot$, definimos os *antepassados* de v como o conjunto de vértices acessíveis pela potência da função predecessor. Mais formalmente:

$$antp(v) = \begin{cases} \emptyset & \text{se } p(v) = \lambda, \\ p(v) \cup antp(p(v)) & \text{se } p(v) \neq \lambda. \end{cases}$$

Da mesma forma, os *descendentes* de v são os vértices acessíveis pela aplicação de uma operação análoga à potência da inversa do predecessor. Mais formalmente:

$$desc(v) = \begin{cases} \emptyset & \text{se } p^{-1}(v) = \emptyset, \\ \bigcup_{w \in p^{-1}(v)} \{w\} \cup desc(w) & \text{se } p^{-1}(v) \neq \emptyset. \end{cases}$$

Veremos logo adiante que a relação de predecessores determina a relação de trilhas alternantes e candidatos a conjuntos ortogonais dentro de Part-k-2.

4.2.2 Trilha alternante intermediária $Q(v)$

Para um vértice v com predecessor bem definido ($p(v) \neq \odot$), definimos recursivamente a trilha alternante intermediária $Q(v)$ da seguinte forma:

1. Se $v \in ter[\mathcal{P}]$:

¹Note a diferença entre os conceitos de *antecessor de v* (v^-) e *predecessor de v* ($p(v)$).

²Note que a função *predecessor* não é injetora, e a sua função inversa mapeia um elemento de $V(G)$ a um conjunto.

- Se $p(v) = \lambda$, então $Q(v) = \emptyset$.
- Senão, se $p(v) = u$ para u e v distintos, então $Q(v) = Q(u) * (u, v)$.

2. Se $v \notin \text{ter}[\mathcal{P}]$:

- Se $p(v) = v^+$, então $Q(v) = Q(v^+) * (v^+, v)$.
- Se $p(v) = u$, e u e v pertencem a caminhos distintos em \mathcal{P} , então $Q(v) = Q(u) * (u, v^+, v)$.

Em todos os casos descritos, note que a trilha $Q(v)$ consiste de $Q(p(v))$ concatenado a um pequeno trecho que liga v até $p(v)$. Note também que, para o caso 2 da definição, não foi descrito o subcaso onde $\{p(v) = w : w \neq v^+ \text{ e } w \text{ pertence ao mesmo caminho que } v\}$; esse caso não foi considerado, visto que ele nunca ocorre no algoritmo Part-k-2.

Pela definição acima, a trilha $Q(v)$ pode ser indeterminada se houver um ciclo na função de predecessores; ou seja, se $v \in \text{desc}(v)$. Na propriedade 4.6.1 provaremos que esta situação não ocorre dentro da execução do algoritmo.

4.2.3 Conjunto examinado Y e o conjunto candidato X

O algoritmo Part-k-2, assim como Part-k-1 e Part-Longos, mantém um conjunto $X = \{X_1, X_2\}$ que é um candidato a uma 2-coloração parcial ortogonal a \mathcal{P} , onde X_1 e X_2 são dois conjuntos de vértices ortogonais a \mathcal{P} , e disjuntos entre si. O conjunto X é determinado em função do conjunto de trilhas alternantes intermediárias $Q(v)$ definidas dentro da execução do algoritmo (e portanto, definido indiretamente a partir do conjunto de predecessores do algoritmo).

Definimos a função $c : V(G) \rightarrow \mathbb{N}$ onde $c(v)$ é definido para todo vértice v tal que $p(v) \neq \odot$. Denominamos $c(v)$ como a *cor* de v . Para os vértices v com $Q(v) = \emptyset$, $c(v) = 2$ caso $v \in V[\mathcal{P}^1]$ e $c(v) = 1$ caso contrário. Para os outros vértices (com $Q(v) \neq \emptyset$), definimos:

$$c(v) = \begin{cases} |R_{Q(v)}| - |A_{Q(v)}| + 1 & \text{se } \text{in}[Q(v)] \in \text{ter}[\mathcal{P}^{>1}], \\ |R_{Q(v)}| - |A_{Q(v)}| + 2 & \text{se } \text{in}[Q(v)] \in V[\mathcal{P}^1]. \end{cases}$$

Mostraremos depois que durante a execução do algoritmo, para todo v com predecessor definido, $c(v) \in \{1, 2\}$. Também mostraremos que dentro de um caminho P os vértices v tais que $c(v) = 1$ sempre estão mais próximos de $\text{ter}[P]$ que os vértices tais que $c(v) = 2$. Além disso, pela definição acima, o parâmetro $c(v)$ poderia ser indeterminado se houvesse um ciclo na função de predecessores; ou seja, se $v \in \text{desc}(v)$. Provaremos depois que esta situação não ocorre dentro da execução do algoritmo. Veremos também que $\text{in}[Q(v)] \in \text{ter}[\mathcal{P}]$.

Considere o conjunto $Y = \{Y_1, Y_2\}$ tal que para um vértice v , temos que $v \in Y_1 \cup Y_2$ se e somente se $p(v) \neq \odot$. Ou seja, $Y_1 \cup Y_2$ é o conjunto dos vértices com predecessor definido dentro do algoritmo. O parâmetro $c(v)$ determina a qual dos conjuntos de Y o vértice v pertence: $c(v) = i$ se e somente se $v \in Y_i$. Denominamos Y como o conjunto de vértices examinados pelo algoritmo. A partir do conjunto Y_i , definimos $X_i = \{v : v \in Y_i \text{ e não existe } w \text{ tal que } w \in Y_i \text{ e } w_{<\mathcal{P}}v\}$. Denotamos $X = \{X_1, X_2\}$. Intuitivamente, X_i é o conjunto dos vértices em Y_i que estão mais próximos de $in[\mathcal{P}]$.

4.2.4 Remorso

Em Part-k-2, para um vértice $v \in G$ qualquer, $Q(v)$ é determinado pelo conjunto de predecessores dentro do algoritmo. Uma situação crítica ocorre ao atualizarmos o predecessor de um vértice v , tal que $desc(v) \neq \emptyset$. Pela definição dada, todos os seus descendentes devem ter a trilha alternante, e, possivelmente a sua pertinência a Y atualizados. Esta é uma situação que envolve atualizações ocorridas em iterações anteriores do algoritmo, e deve ser tratada de alguma forma.

Em consonância com a implementação original do algoritmo mostrada no artigo de Berger e Hartman, no algoritmo Part-k-2 evitamos a execução de atualizações que não sejam locais e relativas à iteração corrente. Para isso, foi introduzida a noção de remorso: o remorso é um procedimento que é aplicado a um vértice qualquer v com predecessor definido e remove a lista de descendentes de v , efetivamente desfazendo as operações feitas até então em $desc(v)$.

O remorso somente é aplicado a vértices de Y_2 . Segue adiante a descrição desse procedimento:

Remorso(v)

Para todo $w \in \text{desc}(v)$:

- Se $w^+ \in Y_1$:

$$p(w) \leftarrow w^+$$

$$Q(w) \leftarrow Q(w^+) * (w^+, w)$$

- Senão:

$$p(w) \leftarrow \odot, Q(w) \leftarrow \emptyset$$

$$Y_2 \leftarrow Y_2 \setminus \{w\}$$

Note que o remorso nunca altera predecessores e trilhas alternantes de vértices pertencentes a Y_1 .

4.2.5 Conjunto $W(v)$

Para um vértice v pertencente a Y_2 , $W(v)$ é o primeiro descendente (ou o conjunto dos primeiros descendentes) de v pertencente(s) a Y_1 . É referenciado em alguns casos do algoritmo para encontrar a trilha alternante final T . Mais formalmente,

$$W(v) = \{w \in V(G) : w \in \text{desc}(v) \cap Y_1 \text{ e } \text{antp}(w) \cap Y_1 = \emptyset\}.$$

Veremos na propriedade 4.5.6 que $W(v) \subseteq V[\mathcal{P}^1]$.

4.3 Descrição do algoritmo

Assim como nos algoritmos Part-k-1 e Part-Longos, o algoritmo Part-k-2 consiste em uma sequência de iterações sobre um grafo G e uma partição inicial \mathcal{P} de G , onde em cada iteração, um candidato a conjunto ortogonal (uma 2-coloração parcial para o caso de Part-k-2) a \mathcal{P} é analisado. No caso $k = 1$, só era necessário encontrar um conjunto independente ortogonal a \mathcal{P} . No caso restrito a partições longas, era necessário encontrar k conjuntos independentes disjuntos entre si ortogonais a \mathcal{P} , mas o exame podia ser feito considerando somente um conjunto independente por iteração. No caso $k = 2$, por estarmos lidando com caminhos curtos, este método “guloso” de exame não funciona tão facilmente. Os dois candidatos a conjuntos independentes (que são X_1 e X_2) podem sofrer mudanças em qualquer iteração do algoritmo, e são considerados simultaneamente durante toda a execução do algoritmo. Os conjuntos X_1 e X_2 são disjuntos entre si, e

se em uma dada iteração são ambos independentes, então o algoritmo termina e devolve $X = \{X_1, X_2\}$ como uma 2-coloração parcial ortogonal a \mathcal{P} .

A construção da trilha alternante, como nos algoritmos anteriores, também tem por fim o descobrimento de uma partição em caminhos em G com menor 2-norma que \mathcal{P} . Esta operação se torna muito mais complexa em Part-k-2, e requer maior detalhamento e conceitos adicionais. Nos algoritmos anteriores, a trilha alternante intermediária de um vértice que entra em X em uma dada iteração é baseada na trilha alternante intermediária de um vértice que já se encontra definida antes desta. Em Part-k-2, a construção da trilha alternante final pode envolver partes de trilhas alternantes intermediárias de diferentes vértices. Além disso, diferentemente dos algoritmos anteriores, vértices que têm sua trilha alternante definida em um ponto do algoritmo podem ter sua trilha removida ou modificada. Essas modificações adicionais justificam o uso do predecessor para facilitar a modificação e a decomposição das trilhas alternantes intermediárias.

Um outro aspecto complicador é que, ao incluir caminhos curtos (triviais), o algoritmo necessariamente deve contemplar possibilidades onde a trilha alternante transforme de maneira mais radical a partição dada como entrada. Nas novas partições geradas, caminhos curtos podem se tornar vértices de caminhos longos (incluindo terminais, iniciais e vértices intermediários), e vértices de caminhos longos também podem ser separados em caminhos triviais. Desta forma, o esquema para determinar se uma dada trilha alternante “melhora” a partição atual também é mais detalhado, e é baseado principalmente no cálculo de $c(v)$: ou seja, na diferença entre arestas de avanço e arestas de recuo, além da questão se a trilha alternante se origina de um terminal de caminho longo ou curto.

Para a análise das propriedades e teoremas nas seções seguintes, classificaremos os casos em dois tipos. Os casos em que o algoritmo termina devolvendo uma partição \mathcal{P}' com menor 2-norma que \mathcal{P} a partir de uma trilha alternante T (2c, 3c, 4, 5, 6) são chamados *casos de melhora*. Os outros em que o algoritmo atualiza os predecessores e demais variáveis do algoritmo, e prossegue a sua execução (1, 2a, 2b, 3a e 3c) são chamados *casos de atualização*. O caso 0 do algoritmo é um caso à parte, e ocorre quando o algoritmo encontra uma 2-coloração parcial ortogonal a \mathcal{P} .

Part-k-2 (G, \mathcal{P})**Entrada:**

Um grafo G , e uma partição em caminhos \mathcal{P} de G .

Saída:

Uma 2-coloração parcial ortogonal a \mathcal{P} , ou uma partição \mathcal{P}' de G com $|\mathcal{P}'|_2 < |\mathcal{P}|_2$.

Inicialização:

Para todo vértice $v \in \text{ter}[\mathcal{P}]$:

$$p(v) \leftarrow \lambda, Q(v) \leftarrow \emptyset$$

Para todo vértice $v \in \text{ter}^{-}[\mathcal{P}^{>1}]$:

$$p(v) \leftarrow v^+, Q(v) \leftarrow (v^+, v)$$

Para todo outro vértice v , $p(v) \leftarrow \odot$.

$$Y_1 \leftarrow \text{ter}[\mathcal{P}^{>1}], Y_2 \leftarrow \text{ter}^{-}[\mathcal{P}^{>1}] \cup V[\mathcal{P}^1];$$

Iteração

0. Se X_1 e X_2 são conjuntos independentes, então devolva a 2-coloração parcial $X = \{X_1, X_2\}$.
Senão, seja $e = (u, v)$ uma aresta incidente em dois vértices, ambos em X_1 , ou ambos em X_2 .
1. Se $u, v \in X_2, v \in V[\mathcal{P}^{>1}] \setminus \text{in}[\mathcal{P}^{>1}]$:

$$\begin{aligned} p(v^-) &\leftarrow u \\ Q(v^-) &\leftarrow Q(u) * (u, v, v^-) \\ Y_2 &\leftarrow Y_2 \cup \{v^-\} \end{aligned}$$

2. Se $u, v \in X_2, v \in V[\mathcal{P}^1]$:

- a) Se $\text{desc}(v) = \emptyset$:

$$\begin{aligned} p(v) &\leftarrow u \\ Q(v) &\leftarrow Q(u) * (u, v) \\ Y_1 &\leftarrow Y_1 \cup \{v\} \\ Y_2 &\leftarrow Y_2 \setminus \{v\} \end{aligned}$$

- b) Se $\text{desc}(v) \neq \emptyset$ e $\text{desc}(v) \cap (Y_1 \cup \{u\}) = \emptyset$:

$$\begin{aligned} \text{remorso}(v) \\ p(v) &\leftarrow u \\ Q(v) &\leftarrow Q(u) * (u, v) \\ Y_1 &\leftarrow Y_1 \cup \{v\} \\ Y_2 &\leftarrow Y_2 \setminus \{v\} \end{aligned}$$

- c) Se $\text{desc}(v) \cap (Y_1 \cup \{u\}) \neq \emptyset$:

Se $u \in \text{desc}(v)$:

$$T \leftarrow Q(u) * (u, v)$$

$$\text{Devolva } \mathcal{P}' = \mathcal{P} \Delta T$$

Se $u \notin \text{desc}(v)$:

Seja w um vértice pertencente a $W(v)$:

$$T \leftarrow Q(u) * (u, v) * vQ(w)$$

$$\text{Devolva } \mathcal{P}' = \mathcal{P} \Delta T$$

3. Se $u, v \in X_1, v, v^- \in V[\mathcal{P}^{>1}] \setminus \text{in}[\mathcal{P}^{>1}]$:

a) Se $\text{desc}(v^-) = \emptyset$:

$$p(v^-) \leftarrow u$$

$$Q(v^-) \leftarrow Q(u) * (u, v, v^-)$$

$$Y_1 \leftarrow Y_1 \cup \{v^-\}$$

$$Y_2 \leftarrow Y_2 \setminus \{v^-\}$$

Se $p(v^{--}) = \odot$:

$$p(v^{--}) = v^-$$

$$Q(v^{--}) \leftarrow Q(v^-) * (v^-, v^{--})$$

$$Y_2 \leftarrow Y_2 \cup \{v^{--}\}$$

b) Se $\text{desc}(v^-) \neq \emptyset$ e $\text{desc}(v^-) \cap Y_1 = \emptyset$:

remorso(v^-)

$$p(v^-) \leftarrow u$$

$$Q(v^-) \leftarrow Q(u) * (u, v, v^-)$$

$$Y_1 \leftarrow Y_1 \cup \{v^-\}$$

$$Y_2 \leftarrow Y_2 \setminus \{v^-\}$$

Se $p(v^{--}) = \odot$:

$$p(v^{--}) = v^-$$

$$Q(v^{--}) \leftarrow Q(v^-) * (v^-, v^{--})$$

$$Y_2 \leftarrow Y_2 \cup \{v^{--}\}$$

c) Se $\text{desc}(v^-) \cap Y_1 \neq \emptyset$:

Se $u \in \text{desc}(v^-)$:

$$T \leftarrow v^-Q(u) * (u, v, v^-)$$

$$\text{Devolva } \mathcal{P}' = \mathcal{P} \Delta T$$

Se $u \notin \text{desc}(v^-)$:

Seja w um vértice pertencente a $W(v)$:

$$T \leftarrow Q(u) * (u, v, v^-) * v^-Q(w)$$

$$\text{Devolva } \mathcal{P}' = \mathcal{P} \Delta T$$

4. Se $u, v \in X_1, v \in V[\mathcal{P}^1]$:
- Se $u \in \text{desc}(v)$:
- $T \leftarrow vQ(u) * (u, v)$
- Devolva $\mathcal{P}' = \mathcal{P} \Delta T$
- Se $u \notin \text{desc}(v)$:
- $T \leftarrow Q(u) * (u, v)$
- Devolva $\mathcal{P}' = \mathcal{P} \Delta T$
5. Se $u, v \in X_2, v \in \text{in}[\mathcal{P}^{>1}]$:
- $T \leftarrow Q(u) * (u, v)$
- Devolva $\mathcal{P}' = \mathcal{P} \Delta T$
6. Se $u, v \in X_1, v^- \in \text{in}[\mathcal{P}^{>1}]$:
- $T \leftarrow Q(u) * (u, v, v^-)$
- Devolva $\mathcal{P}' = \mathcal{P} \Delta T$

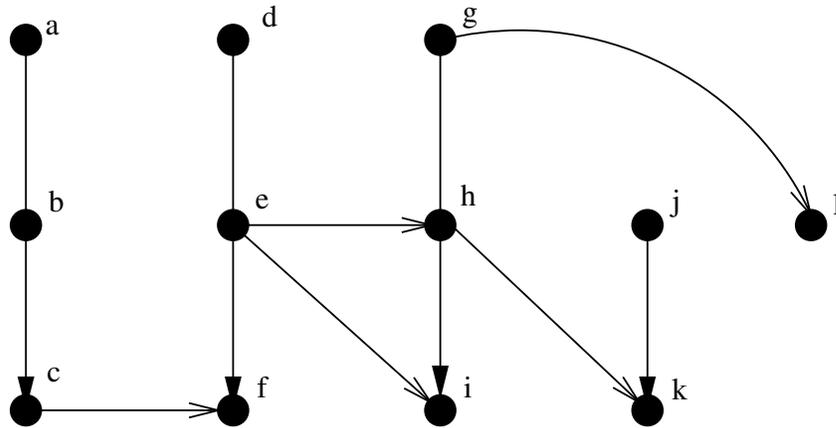


Figura 4.1: Grafo G e a partição em caminhos $\mathcal{P} = \{(a, b, c), (d, e, f), (g, h, i), (j, k), (l)\}$.

Tomemos como exemplo para o algoritmo a partição $\mathcal{P} = \{(a, b, c), (d, e, f), (g, h, i), (j, k), (l)\}$ do grafo G ilustrado na figura 4.1. Durante a inicialização do algoritmo, para todo vértice $v \in \text{ter}[\mathcal{P}] = \{c, f, i, k, l\}$, temos $p(v) = \lambda$, e $Q(v) = \emptyset$. Para os vértices pertencentes a $\text{ter}^-[\mathcal{P}^{>1}]$, temos que $p(b) = c$, $Q(b) = (c, b)$, $p(e) = f$, $Q(e) = (f, e)$, $p(h) = i$, $Q(h) = (i, h)$, $p(j) = k$ e $Q(j) = (k, j)$. Para um vértice $x \in \{a, d, g\}$, temos que $p(x) = \odot$ (e pela definição, $Q(x)$ é indefinido). Pela definição da função $c(v)$ dada na seção 4.2.3, temos que $Y_1 = X_1 = \{c, f, i, k\}$ e $Y_2 = X_2 = \{b, e, h, j, l\}$.

A ordem de exame das arestas em Part-k-2 é importante, e ordens de exame diferentes podem gerar resultados diferentes. Neste exemplo, analisaremos três ordens de exame distintas.

1. Na primeira ordem de exame, a primeira aresta examinada é (c, f) , onde $c, f \in X_1$. Pelo algoritmo, ocorre uma atualização do caso 3a (já que $p^{-1}(e) = \emptyset$ e, portanto, $desc(e) = \emptyset$). Então, temos que $p(e) = c$, $p(d) = e$ (já que $p(d) = \odot$ antes desta iteração), $Q(e) = (c, f, e)$, $Q(d) = (c, f, e, d)$, $Y_1 = \{c, f, i, k, e\}$, $X_1 = \{c, e, i, k\}$ e $Y_2 = X_2 = \{b, d, h, j, l\}$. A segunda aresta examinada é (e, i) , onde $e, i \in X_1$. Novamente, trata-se de uma ocorrência do caso de atualização 3a, e temos que $p(h) = e$, $p(g) = h$, $Q(h) = (c, f, e, i, h)$, $Q(g) = (c, f, e, i, h, g)$, $Y_1 = \{c, f, i, k, e, h\}$, $X_1 = \{c, e, h, k\}$ e $Y_2 = X_2 = \{b, d, g, j, l\}$. A terceira aresta examinada é (h, k) , onde $h, k \in X_1$. Como $j \in in[\mathcal{P}^{>1}]$ (onde j é o antecessor de k), o algoritmo encerra a execução com o caso 6, com $T = (c, f, e, i, h, k, j)$ e devolve $\mathcal{P}' = \{(a, b, c, f), (d, e, i), (g, h, k), (j), (l)\}$.
2. Na segunda ordem de exame, a primeira aresta examinada é (e, h) , onde $e, h \in X_2$. Pelo algoritmo, ocorre uma atualização pelo caso 1, e temos que $p(g) = e$, $Q(g) = (f, e, h, g)$, $Y_1 = X_1 = \{c, f, i, k\}$, $Y_2 = \{b, e, h, j, l, g\}$ e $X_2 = \{b, e, g, j, l\}$. A segunda aresta examinada é (g, l) , onde $g, l \in X_2$. Pelo algoritmo, ocorre uma atualização pelo caso 2a, e temos que $p(l) = g$, $Q(l) = (f, e, h, g, l)$, $Y_1 = X_1 = \{c, f, i, k, l\}$, $Y_2 = \{b, e, h, j, g\}$ e $X_2 = \{b, e, g, j\}$. A terceira aresta examinada é (c, f) , onde $c, f \in X_1$. Como $desc(e) = \{g, l\}$ e $l \in Y_1$, o algoritmo encerra em uma ocorrência do caso 3c. Temos que $W(e) = \{l\}$, e portanto $T = Q(c) * (c, f, e) * eQ(l) = (c, f, e, h, g, l)$. O algoritmo devolve $\mathcal{P}' = \{(a, b, c, f), (d, e, h, i), (g, l), (j, k)\}$ como uma partição com menor 2-norma que \mathcal{P} .
3. Na terceira ordem de exame, a primeira aresta examinada é (e, h) , onde $e, h \in X_2$. Pelo algoritmo, ocorre uma atualização do caso 1, e temos que $p(g) = e$, $Q(g) = (f, e, h, g)$, $Y_1 = X_1 = \{c, f, i, k\}$, $Y_2 = \{b, e, h, j, l, g\}$ e $X_2 = \{b, e, g, j, l\}$. A segunda aresta examinada é (c, f) , onde $c, f \in X_1$. Como $desc(e) = \{g\}$, ocorre uma atualização pelo caso 2b. Inicialmente, é aplicado o procedimento de remorso no vértice e , e após a sua aplicação, temos que $p(g) = \odot$, $Q(g) = \emptyset$ e g não pertence mais a Y . O restante das atualizações relativas ao caso 2b são aplicadas, e após a atualização temos que $p(e) = c$, $p(d) = e$, $Q(e) = (c, f, e)$, $Q(d) = (c, f, e, d)$, $Y_1 = \{c, f, i, k, e\}$, $X_1 = \{c, e, i, k\}$ e $Y_2 = X_2 = \{b, d, h, j, l\}$. As duas próximas arestas a serem analisadas são, nesta ordem, (e, i) e (h, k) . Deixamos a cargo do leitor verificar que o resultado do exame dessas arestas e o resultado final são os mesmos que o exemplificado na primeira ordem de exame.

4.4 Análise dos casos de atualização

Em Part-k-2, várias propriedades do algoritmo são determinadas direta ou indiretamente pela lista de predecessores. Nesta seção, a partir das alterações feitas sobre os predecessores e das aplicações da função remorso obteremos as alterações sobre os demais parâmetros nos casos de atualização do algoritmo, mediante a aplicação das regras de derivação.

Para o caso 1, não há alteração do predecessor ou da trilha alternante para quaisquer vértices que já pertenciam anteriormente a Y . Para v^- , temos que $p(v^-) \leftarrow u$, e $Q(v^-) \leftarrow Q(u) * (u, v, v^-)$. O trecho (u, v, v^-) de $Q(v^-)$ consta de uma aresta de avanço e uma de recuo. Logo, após a atualização, $c(v^-) = c(u) = 2$. Logo, v^- passa a pertencer a Y_2 .

Para o caso 2a, o único vértice que tem seu predecessor e sua trilha alternante alterada é v . Temos que $p(v) \leftarrow u$, e $Q(v) \leftarrow Q(u) * (u, v)$. Como (u, v) é uma aresta de avanço, temos que $c(v) = c(u) - 1 = 1$. Então, v deixa de pertencer a Y_2 , e passa a pertencer a Y_1 .

Para o caso 3a, os únicos vértices com atributos passíveis de alteração são v^- e v^{--} . Para v^- , obtemos $p(v^-) \leftarrow u$, $Q(v^-) \leftarrow Q(u) * (u, v, v^-)$ e $c(v^-) = c(u) = 1$. Caso $p(v^{--})$ seja indefinido antes do começo da iteração, temos que $p(v^{--}) \leftarrow v^-$, $Q(v^{--}) \leftarrow Q(v^-) * (v^-, v^{--})$ e $c(v^{--}) = c(v^-) + 1 = 2$.

Para os casos 2b e 3b, a aplicação de remorso em v garante que $desc(v) = \emptyset$. Logo, as demais atualizações que seguem durante aquela iteração são as mesmas que nos casos 2a ou 3a, respectivamente.

4.5 Invariantes simples do algoritmo

As seguintes propriedades são demonstradas via análise direta dos casos de atualização do algoritmo. São verdadeiras durante a inicialização de Part-k-2, e mantidas após cada iteração subsequente do algoritmo.

Propriedade 4.5.1. *Se em uma dada iteração do algoritmo, um vértice v pertence a Y_1 , para todas as iterações subsequentes temos que $v \in Y_1$.*

A propriedade é claramente verdadeira, já que o remorso somente afeta Y_2 , e em nenhum dos casos de atualização do algoritmo um vértice é removido de Y_1 . Já, a propriedade 4.5.2 segue claramente pela análise do caso 3 do algoritmo.

Propriedade 4.5.2. *Seja v um vértice pertencente a Y_1 . Se em uma dada iteração do algoritmo temos que $p(v) = w$, então para todas as iterações subsequentes temos que $p(v) = w$.*

Temos como corolário da propriedade 4.5.2 a propriedade seguinte:

Propriedade 4.5.3. *Para $v \in \text{ter}[\mathcal{P}^{>1}]$, $p(v) = \lambda$.*

Já a propriedade seguinte segue da discussão feita na seção 4.4.

Propriedade 4.5.4. *Para quaisquer dois vértices $v \in Y_1$ e $w \in Y_2$ em um mesmo caminho P , sabemos que $w_{<P}v$.*

Propriedade 4.5.5. *Se v é um vértice de G tal que $p(v) = \odot$, temos que $p^{-1}(v) = \emptyset$.*

Demonstração. Pela discussão do algoritmo na seção 4.4, em qualquer subcaso de atualização temos que se $p(v) \leftarrow w$, necessariamente $p(w) \neq \odot$. Como a propriedade vale durante a inicialização do algoritmo, ela também vale em qualquer iteração. \square

Propriedade 4.5.6. *Seja v um vértice pertencente a X_2 . Então $W(v) \subseteq V[\mathcal{P}^1]$.*

Demonstração. Durante a inicialização do algoritmo, a propriedade vale, já que para todo $v \in X_2$, temos que $\text{desc}(v) \cap Y_1 = \emptyset$ e, portanto, $W(v) = \emptyset$.

Suponha que a propriedade vale em uma iteração qualquer, e mostraremos que ela também vale na próxima. Ao final do caso 1, para o vértice v^- inserido em X_2 , temos que $\text{desc}(v^-) \cap Y_1 = \emptyset$, pela propriedade 4.5.5. Ao final dos casos 2a e 2b, temos que v passa a fazer parte de $W(u)$, mas como $v \in V[\mathcal{P}^1]$ a propriedade vale. Ao final dos casos 3a e 3b, temos que $\text{desc}(v^-) = \emptyset$ (no caso 3b, graças à aplicação do remorso) e que, se v^{--} tiver predecessor definido nesta iteração, $\text{desc}(v^{--}) = \emptyset$ (novamente, pela propriedade 4.5.5), e $W(v) = \emptyset$. Logo, a propriedade continua valendo. \square

Propriedade 4.5.7. *Os conjuntos X_1 e X_2 são ambos ortogonais a \mathcal{P} e $X_1 \cap X_2 = \emptyset$.*

Demonstração. Pela definição dada na seção 4.2.3, claramente $Y_1 \cap Y_2 = \emptyset$, e como também X_i é um subconjunto de Y_i , temos que $X_1 \cap X_2 = \emptyset$. Também é claro que X_1 e X_2 são sempre dois conjuntos ortogonais a \mathcal{P} , pela análise dos casos feita na seção 4.4. \square

Propriedade 4.5.8. *Para todo $v \in Y$, temos que $Q(v)$ é uma trilha alternante. Além disso, se v não é um caminho trivial em \mathcal{P} , então $Q(v)$ termina em uma aresta de recuo.*

Demonstração. Na inicialização do algoritmo, a propriedade claramente é válida. Suponha então que ela vale em uma iteração qualquer e mostraremos que ela vale na próxima.

Se ocorre o caso 1, somente $Q(v^-)$ é alterado, e temos que $Q(v^-) = Q(u) * (u, v, v^-)$. Como necessariamente u é um caminho curto, ou $Q(u)$ termina em uma aresta de recuo, $Q(v^-)$ é uma trilha alternante que termina em uma aresta de recuo.

Se ocorre o caso $2a$ ou $2b$, somente $Q(v)$ é alterado (desconsiderando o conjunto de vértices removidos pelo remorso, que não mais pertencem a Y), e temos que $Q(v) = Q(u) * (u, v)$. Como necessariamente u é um caminho curto, ou $Q(u)$ termina em uma aresta de recuo, $Q(v)$ é uma trilha alternante.

Se ocorre o caso $3a$ ou $3b$, somente $Q(v^-)$ e $Q(v^{--})$ são passíveis de sofrer qualquer alteração (novamente, desconsiderando os vértices afetados pelo remorso), e temos $Q(v^-) = Q(u) * (u, v, v^-)$, que é uma trilha alternante que termina em uma aresta de recuo. Caso $Q(v^{--})$ seja alterado, temos que $Q(v^{--}) = Q(v^-) * (v^-, v^{--})$ também é uma trilha alternante que termina em uma aresta de recuo. Portanto, para todos os casos de atualização a propriedade continua valendo. \square

4.6 Invariantes principais do algoritmo

Propriedade 4.6.1. *Se um vértice u pertence a Y , então $u \notin \text{desc}(u)$.*

Demonstração. Na inicialização do algoritmo, a propriedade claramente vale. Suponha que esta propriedade vale em uma iteração qualquer, e mostraremos que ela também vale na próxima. No final do caso 1, v^- é inserido em Y , mas como $\text{desc}(v^-) = \emptyset$ (pela propriedade 4.5.5) a propriedade vale. No final dos casos $2a$ e $2b$, temos $\text{desc}(v) = \emptyset$ (no caso $2b$, graças à aplicação do remorso).

No final dos casos $3a$ e $3b$, se v^{--} foi incluído em Y , então $\text{desc}(v^{--}) = \emptyset$. Além disso, $\text{desc}(v^-) = \emptyset$, (para o caso $3b$, devido a aplicação do remorso). Logo, a propriedade vale. \square

Para os próximos lemas, denotamos $Q = \bigcup_{v \in Y} E[Q(v)]$.

Lema 4.6.1. *O subgrafo gerador com conjunto de arestas $E[\mathcal{P}] \cup Q$ é acíclico.*

Demonstração. Na inicialização do algoritmo, para todo v para o qual $Q(v)$ é definido, temos que $A_{Q(v)} = \emptyset$, logo $E[\mathcal{P}] \cup Q = E[\mathcal{P}]$ e a propriedade vale.

Suponha que a propriedade vale no final de uma iteração e provaremos que ela também vale no final da próxima iteração. Para isso, faremos a suposição contraditória de que após a atualização de Q dentro da iteração atual, um circuito C foi formado em $E[\mathcal{P}] \cup Q$. Em todos os casos de atualização do algoritmo, a única aresta de avanço adicionada em qualquer trilha pertencente a Q é uma aresta $e = (u, v)$, analisada por ferir a independência de X . Logo, necessariamente C deve incluir (u, v) .

Como (u, v) pertence a C , e C forma um circuito em $E[\mathcal{P}] \cup Q$, deve existir também uma outra aresta $d = (w, x)$ pertencente a C , tal que $x \leq_{\mathcal{P}} u$. Suponha que x e u ambos pertençam a Y_1 , ou ambos pertençam a Y_2 . Pela definição de X , isso implica que $u \notin X$, contradizendo nossa hipótese inicial. Também não pode ser o caso de que $u \in Y_2$ e

$x \in Y_1$ pela propriedade 4.5.4. Então, $u \in Y_1$ e $x \in Y_2$, e em C deve existir uma aresta $f = (a, b) \in Q$ tal que $a \in Y_1$ e $b \in Y_2$.

Na iteração em que f foi adicionada a Q , a e b pertenciam ambos a X_1 , ou ambos a X_2 . Necessariamente, deviam pertencer ambos a X_2 , já que pela propriedade 4.5.1, um vértice pertencente a X_1 não pode passar a pertencer a X_2 em uma iteração posterior. Também é necessário que a tenha alterado sua pertinência de Y_2 para Y_1 em alguma iteração após f ter sido analisada. Nessa iteração posterior, se $a \in V[\mathcal{P}^1]$, uma aresta (h, a) foi examinada pelo algoritmo em uma iteração na qual ocorria necessariamente o caso 2b (já que pela existência de (a, b) , $desc(a) \neq \emptyset$). Por outro lado, se $a \in V[\mathcal{P}^{>1}]$ (lembrando que $a \notin ter[\mathcal{P}^{>1}]$, ou este vértice seria atribuído a Y_1 pela inicialização do algoritmo), então uma aresta (h, a^+) foi examinada pelo algoritmo em uma iteração na qual ocorria necessariamente o caso 3b. No entanto, em ambos os casos ocorre $remorso(a)$ com a exclusão da aresta (a, b) de Q , e temos uma contradição. □

Lema 4.6.2. *O subgrafo gerado por $\mathcal{P}' = \mathcal{P} \Delta T$ é acíclico.*

Demonstração. Pelo lema anterior, temos que $E[\mathcal{P}] \cup Q$ não contém circuitos. Em todos os casos de melhora do algoritmo, ocorre a inserção de no máximo uma aresta de avanço $e = (u, v)$ não pertencente a Q em T . Pela mesma argumentação do lema anterior, temos portanto que $E[\mathcal{P}] \cup Q \cup \{(u, v)\}$ é acíclico. Como $\mathcal{P} \Delta T \subseteq E[\mathcal{P}] \cup Q \cup \{(u, v)\}$, \mathcal{P}' é necessariamente um grafo acíclico. □

Lema 4.6.3. *Part-k-2 termina em tempo finito.*

Demonstração. Pela propriedade 4.5.1, sabemos que Y_1 nunca decresce durante a execução do algoritmo. No caso 1, Y_2 sempre cresce, e apesar de ser possível que Y_2 decresça nos casos 2 e 3, nestes dois casos Y_1 sempre cresce. Logo, podemos garantir que com um número finito de iterações, o algoritmo acaba encontrando uma 2-coloração ortogonal a \mathcal{P} , uma trilha alternante que pode gerar uma partição em caminhos com menor 2-norma, ou o algoritmo chega a um ponto em que todos os vértices de $in^+[\mathcal{P}^{>1}] \cup \mathcal{P}^1$ estão em X_1 , e o algoritmo termina. □

4.6.1 Tipos de trilhas alternantes

Nos casos de melhora de Part-k-2, queremos encontrar uma partição em caminhos com menor 2-norma que a partição dada como entrada pelo algoritmo. Para uma partição \mathcal{P} qualquer, o seguinte vale:

$$\begin{aligned}
|\mathcal{P}|_2 &= 2|\mathcal{P}^{>1}| + |\mathcal{P}^1| \\
&= |\mathcal{P}^{>1}| + (|\mathcal{P}^{>1}| + |\mathcal{P}^1|) \\
&= |\mathcal{P}^{>1}| + |\mathcal{P}|.
\end{aligned} \tag{4.1}$$

Então, é possível estabelecer a diferença entre a 2-norma de duas partições em caminhos \mathcal{P}' e \mathcal{P} de um mesmo grafo pela seguinte equação:

$$|\mathcal{P}'|_2 - |\mathcal{P}|_2 = |\mathcal{P}'^{>1}| - |\mathcal{P}^{>1}| + |\mathcal{P}'| - |\mathcal{P}|. \tag{4.2}$$

Sejam T uma trilha alternante completa e $\mathcal{P}' = \mathcal{P} \Delta T$. Pelo teorema 3.1.2, podemos calcular $|\mathcal{P}'| - |\mathcal{P}|$ a partir da diferença entre as arestas de avanço e de recuo de T . Felizmente, como T é completa, podemos obter um método de calcular a diferença entre o número de caminhos longos de \mathcal{P} e \mathcal{P}' . Para tal, classificaremos as trilhas alternantes completas nos seguintes tipos:

- $l - l$ (longo-longo) se $in(Q) \in ter[\mathcal{P}^{>1}]$ e $ter(Q) \in in[\mathcal{P}^{>1}]$,
- $l - c$ (longo-curto) se $in(Q) \in ter[\mathcal{P}^{>1}]$ e $ter(Q) \in V[\mathcal{P}^1]$,
- $c - l$ (curto-longo) se $in(Q) \in V[\mathcal{P}^1]$ e $ter(Q) \in in[\mathcal{P}^{>1}]$,
- $c - c$ (curto-curto) se $in(Q) \in V[\mathcal{P}^1]$ e $ter(Q) \in V[\mathcal{P}^1]$.

Dentre as trilhas alternantes completas do tipo $c - c$, também categorizaremos entre as trilhas alternantes completas que também são circuitos alternantes, e as que não são³.

Lema 4.6.4. *Sejam $T = (v_1, v_2, \dots, v_l)$ uma trilha alternante em relação a uma partição \mathcal{P} , $\mathcal{P}' = \mathcal{P} \Delta T$, e v_i um vértice de T para $2 \leq i \leq l-1$. Então, $v_i \in in[\mathcal{P}^{>1}]$ se e somente se $v_i \in in[\mathcal{P}'^{>1}]$, e $v_i \in ter[\mathcal{P}^{>1}]$ se e somente se $v_i \in ter[\mathcal{P}'^{>1}]$.*

Demonstração. Se $v_i \in ter[\mathcal{P}^{>1}]$, então necessariamente (v_{i-1}, v_i) é uma aresta de avanço, (v_i, v_{i+1}) uma de recuo, e v_i continua sendo o terminal de um longo em \mathcal{P}' . Analogamente, se $v_i \in in[\mathcal{P}^{>1}]$, então (v_i, v_{i+1}) é uma aresta de avanço, (v_{i-1}, v_i) é uma aresta de recuo, e v_i permanece como o inicial de um caminho longo em \mathcal{P}' .

Se $v_i \in V[\mathcal{P}^1]$, as arestas (v_{i-1}, v_i) e (v_i, v_{i+1}) de T são ambas de avanço, e $v_i \in \mathcal{P}^{>1} \setminus (in[\mathcal{P}'^{>1}] \cup ter[\mathcal{P}'^{>1}])$. Se $v_i \in \mathcal{P}^{>1} \setminus (in[\mathcal{P}^{>1}] \cup ter[\mathcal{P}^{>1}])$ (ou seja, v_i é o vértice intermediário de um caminho longo em \mathcal{P}) e T passa por v_i uma vez, para o caso em

³Note que, embora a propriedade 4.6.1 previna o surgimento de circuitos alternantes para trilhas alternantes intermediárias, nada impede que um circuito alternante seja obtido ao final de um caso de melhora. Veja a figura 4.2 para um exemplo disso.

que (v_{i-1}, v_i) é uma aresta de avanço e (v_i, v_{i+1}) uma de recuo, ou para o caso em que (v_{i-1}, v_i) é uma aresta de recuo e (v_i, v_{i+1}) uma de avanço, claramente v_i permanece como o intermediário de um longo em \mathcal{P}' . Caso ambas as arestas sejam de recuo, então v_i é isolado, e se torna o vértice de um caminho trivial em \mathcal{P}' .

Finalmente, se $v_i \in \mathcal{P}^{>1} \setminus (in[\mathcal{P}^{>1}] \cup ter[\mathcal{P}^{>1}])$, pode ser o caso que T passe por v_i duas vezes. Uma das vezes deve ser com uma aresta de avanço entrando e uma de recuo saindo, e outra vez deve ser com uma aresta de recuo entrando e uma de avanço saindo de v_i . Neste caso, é claro que v_i também permanece como o intermediário de um caminho longo em \mathcal{P}' . Logo, em todos os casos analisados a propriedade vale. \square

Lema 4.6.5. *Sejam $T = (v_1, v_2, \dots, v_l)$ uma trilha alternante completa e $\mathcal{P}' = \mathcal{P} \triangle T$. Se T é do tipo $l - l$, então $|\mathcal{P}'^{>1}| = |\mathcal{P}^{>1}| - 1$. Similarmente, se T é do tipo $l - c$, $c - l$ ou um circuito alternante do tipo $c - c$, então $|\mathcal{P}'^{>1}| = |\mathcal{P}^{>1}|$. Se T é do tipo $c - c$, e não é um circuito alternante, então $|\mathcal{P}'^{>1}| = |\mathcal{P}^{>1}| + 1$.*

Demonstração. Faremos a contagem pela diferença entre o número de iniciais e terminais de caminhos longos em \mathcal{P}' e \mathcal{P} . Claramente, para um vértice $v \notin T$, v é um inicial/terminal de um caminho não-trivial em \mathcal{P} se e somente se v é um inicial/terminal de um caminho não-trivial em \mathcal{P}' . Além disso, pelo lema 4.6.4, vértices intermediários de T não fazem diferença na contagem. Assim sendo, calcularemos a diferença de iniciais/terminais entre \mathcal{P} e \mathcal{P}' pelas extremidades de T , v_1 e v_l .

Se $v_1 \in ter[\mathcal{P}^{>1}]$, então $v_1 \notin ter[\mathcal{P}'^{>1}]$. Logo, $ter[\mathcal{P}'^{>1}] = ter[\mathcal{P}^{>1}] \setminus \{v_1\}$. Por raciocínio análogo, se $v_l \in in[\mathcal{P}^{>1}]$, $v_l \notin in[\mathcal{P}'^{>1}]$, e, portanto, $ter[\mathcal{P}'^{>1}] = ter[\mathcal{P}^{>1}] \setminus \{v_l\}$.

Como arestas de avanço são as únicas arestas de T que incidem em caminhos curtos, temos que se $v_1 \in V[\mathcal{P}^1]$, então $v_1 \in in[\mathcal{P}'^{>1}]$. Igualmente, se $v_l \in V[\mathcal{P}^1]$, então $v_l \in ter[\mathcal{P}'^{>1}]$. Isso a menos que T seja um circuito alternante, onde $v_1 = v_l \notin ter[\mathcal{P}'^{>1}] \cup in[\mathcal{P}'^{>1}]$.

Logo:

- Se T é do tipo $l - l$, $ter[\mathcal{P}'^{>1}] = ter[\mathcal{P}^{>1}] \setminus \{v_1\}$ e $in[\mathcal{P}'^{>1}] = in[\mathcal{P}^{>1}] \setminus \{v_l\}$.
- Se T é do tipo $l - c$, então $ter[\mathcal{P}'^{>1}] = (ter[\mathcal{P}^{>1}] \setminus \{v_1\}) \cup \{v_l\}$ e $in[\mathcal{P}'^{>1}] = in[\mathcal{P}^{>1}]$.
- Se T é do tipo $c - l$, então $in[\mathcal{P}'^{>1}] = (in[\mathcal{P}^{>1}] \setminus \{v_l\}) \cup \{v_1\}$ e $ter[\mathcal{P}'^{>1}] = ter[\mathcal{P}^{>1}]$.
- Se T é do tipo $c - c$ e não é um circuito alternante, então $ter[\mathcal{P}'^{>1}] = ter[\mathcal{P}^{>1}] \cup \{v_l\}$ e $in[\mathcal{P}'^{>1}] = in[\mathcal{P}^{>1}] \cup \{v_1\}$.
- Se T é um circuito alternante do tipo $c - c$, então $ter[\mathcal{P}'^{>1}] = ter[\mathcal{P}^{>1}]$ e $in[\mathcal{P}'^{>1}] = in[\mathcal{P}^{>1}]$.

A contagem de longos segue trivialmente da contagem de iniciais e terminais feita acima. \square

4.7 Casos de melhora

Teorema 4.7.1. *Ao final dos casos 2c, 3c, 4, 5 ou 6 do algoritmo, temos que $\mathcal{P}' = \mathcal{P} \Delta T$ é uma partição em caminhos de G , tal que $|\mathcal{P}'|_2 = |\mathcal{P}|_2 - 1$.*

Demonstração. Como foi provado no lema 4.6.2, o subgrafo gerador de G representado por $\mathcal{P}' = \mathcal{P} \Delta T$ é acíclico. Portanto, é suficiente provarmos que T é uma trilha alternante própria e que a equação $|\mathcal{P}'|_2 - |\mathcal{P}|_2 = (|\mathcal{P}'^{>1}| - |\mathcal{P}^{>1}|) + (|\mathcal{P}'| - |\mathcal{P}|) = -1$ vale.

Para calcularmos a diferença entre o número de caminhos longos usamos a categorização do lema 4.6.5, e para calcularmos a diferença entre o número de caminhos totais, usamos o resultado exposto no teorema 3.1.2; isto é, que $|\mathcal{P}'| - |\mathcal{P}| = |R_T| - |A_T|$. Especificamente, demonstraremos que a trilha devolvida em cada caso de melhora cai em um dos seguintes casos:

- T é do tipo $l - l$ com $|A_T| - |R_T| = 0$.
- T é do tipo $c - l$ com $|A_T| - |R_T| = 1$.
- T é do tipo $l - c$ com $|A_T| - |R_T| = 1$.
- T é do tipo $c - c$ e não é um circuito alternante, com $|A_T| - |R_T| = 2$.
- T é um circuito alternante com $|A_T| - |R_T| = 1$.

Para todos os casos acima, temos claramente que $|\mathcal{P}'|_2 - |\mathcal{P}|_2 = -1$.

Caso 2c

Como v é um caminho curto tal que $v \in Y_2$, então $p(v) = \lambda$ e $Q(v) = \emptyset$. Se $u \in \text{desc}(v)$, então $\text{in}[Q(u)] = v$, e $|A_{Q(u)}| - |R_{Q(u)}| = 0$. Logo, $T = Q(u) * (u, v)$ é um circuito alternante com $|A_T| - |R_T| = 1$.

Se $u \notin \text{desc}(v)$ e $\text{desc}(v) \cap Y_1 \neq \emptyset$, então, pela propriedade 4.5.6, temos que $w \in V[\mathcal{P}^1]$ (lembrando que w é o primeiro vértice de $\text{desc}(v) \cap Y_1$). Pela propriedade 4.5.8, $T = Q(u) * (u, v) * vQ(w)$ é uma trilha alternante completa. Se $\text{in}[Q(u)] \in V[\mathcal{P}^1]$, T é do tipo $c - c$ com $|A_T| - |R_T| = 2$. Senão, se $\text{in}[Q(u)] \in \text{ter}[\mathcal{P}^{>1}]$, T é do tipo $l - c$ com $|A_T| - |R_T| = 1$.

Caso 3c

Pela propriedade 4.5.6, $w \in V[\mathcal{P}^1]$. Se $u \notin \text{desc}(v^-)$, então $T = Q(u) * (u, v, v^-) * v^-Q(w)$ é uma trilha alternante completa. Se $\text{in}[Q(u)] \in V[\mathcal{P}^1]$, T é do tipo $c - c$ com $|A_T| - |R_T| = 2$. Se $\text{in}[Q(u)] \in \text{ter}[\mathcal{P}^{>1}]$, T é do tipo $l - c$ com $|A_T| - |R_T| = 1$.

Se $u \in \text{desc}(v^-)$, então também $u \in \text{desc}(w)$. Logo, $T = v^-Q(u) * (u, v, v^-) = v^-Q(w) * wQ(u) * (u, v, v^-) = wQ(u) * (u, v, v^-) * v^-Q(w)$ é um circuito alternante do tipo $c - c$ com $|A_T| - |R_T| = 1$.

Veja o exemplo da figura 4.3. A ordem de exame das arestas é $((g, j), (i, m), (a, c), (b, e), (e, h))$. Neste caso, $T = (a, c, b, e) * (e, h) * (h, g, j, i, m)$ e $\mathcal{P}' = \{(a, c, d), (b, e, h), (f, g, j, i), (i, m)\}$. Ao alterarmos a ordem de exame deste exemplo para $((a, c), (b, e), (e, h))$, a 2-coloração parcial $X = \{\{a, b, f, i, m\}, \{d, e, g, l\}\}$ é encontrada.

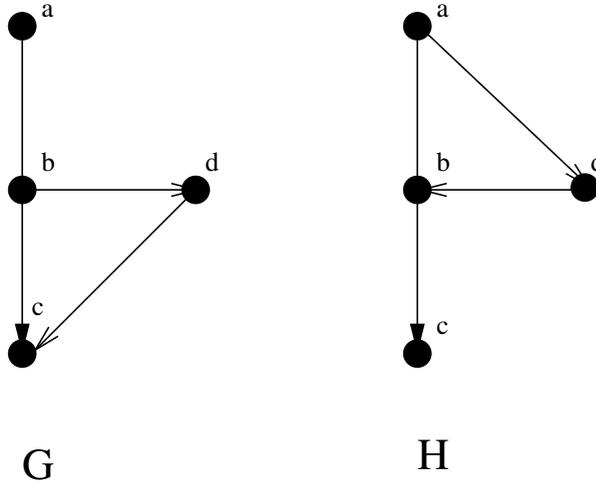


Figura 4.2: Grafo G ilustrando uma ocorrência do caso 3c do algoritmo. O grafo H ilustra uma ocorrência do caso 2c. As trilhas alternantes encontradas ao final de cada caso são (d, c, b, d) para o grafo G , e (d, b, a, d) para o grafo H .

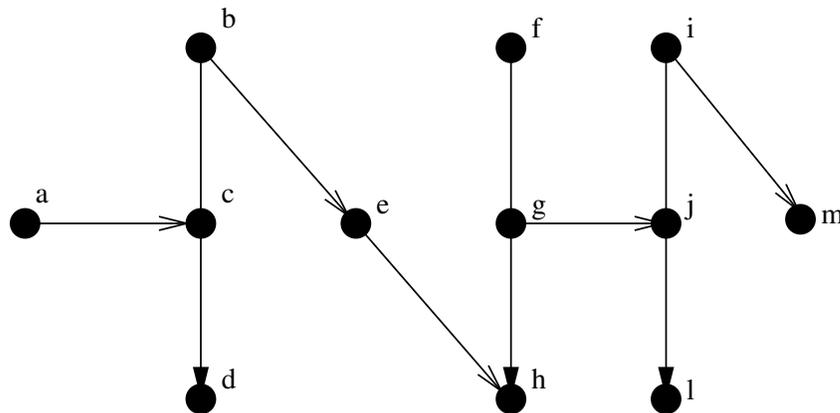


Figura 4.3: Exemplo do caso 3c para o subcaso onde $u \notin \text{desc}(v)$.

Caso 4

Se $u \notin \text{desc}(v)$, construímos a trilha alternante $T = Q(u) * (u, v)$. Caso $\text{in}[Q(u)] \in \text{ter}[\mathcal{P}^{>1}]$, T é do tipo $l - c$, com $|A_T| - |R_T| = 1$. Caso $\text{in}[Q(u)] \in V[\mathcal{P}^1]$, T é do tipo $c - c$ (mas não um circuito alternante), com $|A_T| - |R_T| = 2$.

Se $u \in \text{desc}(v)$, temos que $T = vQ(u) * (u, v)$, e T é um circuito alternante do tipo $c - c$ com $|A_T| - |R_T| = 1$.

Caso 5

Construímos a trilha alternante $T = Q(u) * (u, v)$. Se $\text{in}[Q(u)] \in V[\mathcal{P}^1]$, T é do tipo $c - l$ com $|A_T| - |R_T| = 1$. Senão, se $\text{in}[Q(u)] \in \text{ter}[\mathcal{P}^{>1}]$, T é do tipo $l - l$ com $|A_T| - |R_T| = 0$.

Caso 6

Se $u \notin \text{desc}(v^-)$, construímos $T = Q(u) * (u, v, v^-)$. Se $\text{in}[Q(u)] \in V[\mathcal{P}^1]$, então Q é do tipo $c - l$ com $|A_T| - |R_T| = 1$. Senão, se $\text{in}[Q(u)] \in \text{ter}[\mathcal{P}^{>1}]$, então Q é do tipo $l - l$ com $|A_T| - |R_T| = 0$.

Se $u \in \text{desc}(v^-)$, então, pela mesma justificativa dada no caso 3c, $T = v^-Q(u) * (u, v, v^-)$ é um circuito alternante do tipo $c - c$ com $|A_T| - |R_T| = 1$.

Temos como exemplo do caso 6 a figura 4.4. A ordem de exame das arestas é $((b, e), (c, f), (d, g), (g, c), (b, e))$, o circuito alternante resultante é $T = (b, e, d, g, c, b)$ e $\mathcal{P}' = \{(a, b, e, f), (d, g, c)\}$.

Em todos os casos acima, claramente $\mathcal{P}' = \mathcal{P} \Delta T$ é uma partição em caminhos (já que todas as trilhas alternantes intermediárias envolvidas seguem a propriedade 4.5.8) tal que $|\mathcal{P}'|_2 = |\mathcal{P}|_2 - 1$.

□

O teorema anterior em conjunto com o lema 4.6.3 e a propriedade 4.5.7 provam que o algoritmo funciona corretamente.

Corolário 4.7.1. *Part- $k-2$ devolve uma 2-coloração parcial ortogonal a \mathcal{P} , ou uma partição em caminhos \mathcal{P}' , tal que $|\mathcal{P}'|_2 < |\mathcal{P}|_2$ em tempo finito.*

Teorema 4.7.2. *(Berger-Hartman, 2008)*

Se \mathcal{P} é uma partição em caminhos 2-ótima de G , então existe uma 2-coloração parcial ortogonal a \mathcal{P} .

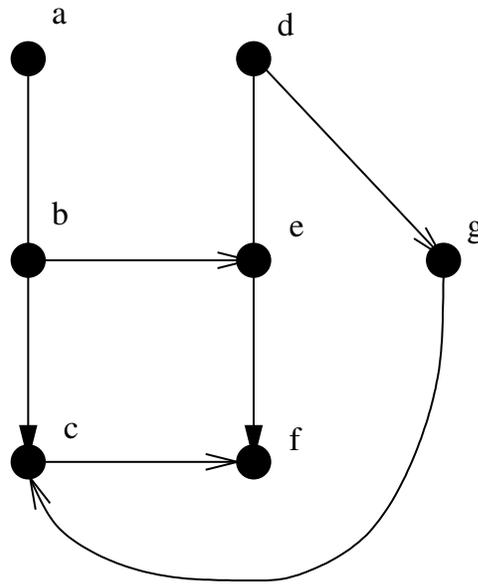


Figura 4.4: Exemplo do caso 6 com geração de circuito alternante.

4.8 Modificações com relação ao artigo original

Ambos os algoritmos para o caso $k = 1$ e para o caso $k = 2$ foram baseados no artigo de 2008 de Berger e Hartman[23]. Para a exposição destes, tomamos a liberdade de efetuar uma série de modificações sobre a demonstração original, especialmente para o caso $k = 2$. Algumas modificações foram feitas de modo a corrigir eventuais erros e reformular, introduzir ou esclarecer conceitos do artigo original. Além disso, várias demonstrações foram reescritas, expandidas ou introduzidas neste texto.

No artigo original, o conceito de fotografia(*snapshot*) é a estrutura matemática central do algoritmo. A fotografia é um conjunto ordenado de arestas do grafo G , que correspondem às arestas examinadas pelo algoritmo, na ordem em que foram examinadas, sendo excluídas algumas arestas em casos especiais determinados (como o nosso caso 3 do algoritmo, e em operações de remorso). Uma execução do algoritmo pode ser resumida pela fotografia, e ela determina de maneira complexa outras estruturas como os predecessores e os *path segments* (estrutura correlacionada ao conceito de *trilhas alternantes intermediárias* deste trabalho). A fotografia tem uma série de propriedades bem definidas. A demonstração de que a execução do algoritmo determina uma sequência de arestas que constitui uma fotografia (ou seja, que preserva essas propriedades) é implícita, e a demonstração de correteude do algoritmo é baseada nas propriedades da fotografia. Ao reescrever a demonstração, preferimos remover completamente o conceito de fotografia, em favor de uma demonstração mais centrada nos próprios algoritmos e suas invariantes.

Algumas correções ao artigo original também se mostraram necessárias para contornar certos casos anômalos que ocorriam na demonstração original.

O caso do algoritmo original que corresponde ao caso 2c deste capítulo não trata de maneira especial quando $u \notin \text{desc}(v)$. Essa condição teve que ser adicionada e tratada à parte como um caso de melhora do algoritmo, já que se $u \in \text{desc}(v)$, a aplicação de remorso pode excluir u de Y , fazendo com que $Q(v)$ seja indeterminado (isto ocorre na partição do grafo H , no exemplo mostrado na figura 4.2).⁴

Nos casos do algoritmo original que correspondem aos casos 3a e 3b, não existe o teste condicional que determina se $p(v^{--}) = \emptyset$; $p(v^{--})$ é sempre atualizado e v^{--} é sempre inserido em Y_2 . No entanto, isso pode levar a uma perda do predecessor de v^{--} , e possivelmente a um “ciclo de predecessão” onde $v^{--} \in \text{desc}(v^{--})$ e $Q(v^{--})$ é indefinido. Veja o exemplo da figura 4.5 — com o exame das arestas (b, e) , (d, g) e (g, f) , $Q(d) = Q(e) = Q(g) = (d, e, f, g, d)$, e não é possível determinar corretamente $c(v)$ para nenhum desses vértices.

Também modificamos a rotina da atualização de Y_2 durante a aplicação do remorso. Na função original, todos os descendentes de v são removidos de Y_2 . No entanto, ao aplicarmos a função de remorso, é possível chegar à situação em que há um caminho $P \in \mathcal{P}$ onde $Y_2 \cap P = \emptyset$ e, portanto, X_2 não é transversal a \mathcal{P} .

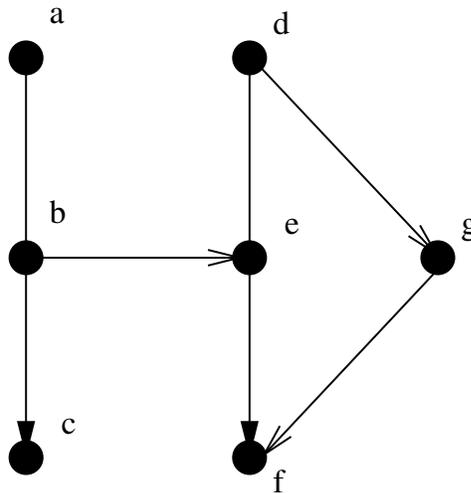


Figura 4.5: Exemplo de um caso anômalo no artigo original.

⁴Em correspondência recente com a profa. Hartman, ela reconheceu o problema e propôs uma solução que é, para todos os efeitos, a mesma que usamos neste texto.

Capítulo 5

Considerações Finais

É digno de nota que as demonstrações dos teoremas 3.2.2, 3.3.3 e 4.7.2 provam, para seus respectivos casos particulares, uma proposição um pouco mais forte que a conjectura de Berge. Nessas demonstrações, as cores das colorações parciais construídas não se “intercalam” entre si.

Essas demonstrações sugerem um fortalecimento da conjectura de Berge. Para um caminho $P = (v_1, v_2, \dots, v_n)$ e $v_i, v_j \in P$, dizemos que v_i precede v_j se $i < j$.

Conjectura 5.0.1. *Sejam G um grafo orientado, k um inteiro positivo, e \mathcal{P} uma partição em caminhos k -ótima de G . Então, existe uma k -coloração parcial $C = \{I_1, I_2, \dots, I_k\}$ ortogonal a \mathcal{P} , onde para todo caminho $P \in \mathcal{P}$, se $u \in I_i$, $v \in I_j$, $u, v \in P$ e v precede u em P , então $j > i$.*

A conjectura 5.0.1 foi proposta por Hartman [24]. Como um exemplo de uma coloração parcial que não segue essa propriedade, temos aquela retratada na figura 1.4 do Capítulo 1.

Pode-se perguntar se a técnica usada no algoritmo para o caso $k = 2$ (isto é, a construção de trilhas alternantes e a obtenção da diferença simétrica entre uma trilha alternante e a partição original) pode ser estendida para uma demonstração do caso geral da conjectura de Berge. Para esta questão, é preciso notar que uma das propriedades mais úteis das trilhas alternantes é aquela dada pelo teorema 3.1.2: isto é, que $|\mathcal{P}'| - |\mathcal{P}| = |R_T| - |A_T|$. Para o caso $k = 1$, esta é claramente uma propriedade interessante, já que é possível obter várias partições em caminhos alternativas candidatas e calcular a 1-norma delas facilmente somente a partir das trilhas alternantes construídas (onde a trilha alternante final representa aquela que tem menor 1-norma que a partição original). Para o caso $k = 2$, também trata-se de uma propriedade útil, já que nesse caso só existem dois tipos de caminhos: caminhos longos e caminhos triviais.

No entanto, para $k \geq 3$, a diferença entre o número de caminhos de duas partições

distintas não é uma informação suficiente para o cálculo da k -norma, e esse fato se reflete na demonstração do teorema 3.3.3 para partições compostas somente de caminhos longos. Para essa demonstração, são exploradas certas propriedades de $\mathcal{P}(v) = \mathcal{P} \Delta Q(v)$, e a demonstração se torna mais complexa. Portanto, as propriedades sobre trilhas alternantes mostradas nos capítulos 3 e 4 provavelmente não são suficientes para uma demonstração do caso geral da conjectura de Berge.

Uma das hipóteses que considerei enquanto estudava o problema era a de simplificar a demonstração abandonando completamente a noção de trilha alternante, e de preservar apenas a abordagem iterativa. Nessa abordagem, o algoritmo se baseia na construção de “partições em caminhos intermediárias” $\mathcal{P}(v)$. A idéia é equivalente à definição dada no Capítulo 3, onde se é definido $\mathcal{P}(v) = \mathcal{P} \Delta Q(v)$. Segue uma versão alternativa do algoritmo Part-k-1 segundo essa idéia.

Part-k-1-Alt(\mathcal{P}, G)

Entrada

Um grafo G , e uma partição em caminhos \mathcal{P} de G .

Saída

Um conjunto independente ortogonal a \mathcal{P} , ou uma partição \mathcal{P}' com $|\mathcal{P}'| < |\mathcal{P}|$.

Início

$X \leftarrow \text{ter}[\mathcal{P}]$.

Para todo vértice $v \in X$, $\mathcal{P}(v) \leftarrow \mathcal{P}$.

Repita:

0. Se X é um conjunto independente, então devolva X :
1. Senão, seja $e = (u, v)$ uma aresta incidente em dois vértices em X :
 - a) Se $v \notin \text{in}[\mathcal{P}]$:

$$X \leftarrow X + v^- - v.$$

$$\mathcal{P}(v^-) \leftarrow \mathcal{P}(u) + (u, v) - (v^-, v).$$
 - b) Se $v \in \text{in}[\mathcal{P}]$:

$$T \leftarrow Q(u) * (u, v).$$
 Devolva $\mathcal{P}' = \mathcal{P}(u) + (u, v)$.

Fim

Os dois lemas seguintes descrevem invariantes claras do algoritmo acima. Lembremos que $P(v)$ se refere ao caminho de \mathcal{P} que contém v , e $P(v, w)$ se refere ao caminho da partição $\mathcal{P}(w)$ que contém v .

Lema 5.0.1. *Para todo $v \in X$, $\mathcal{P}(v)$ é uma partição em caminhos de G tal que $|\mathcal{P}(v)| = |\mathcal{P}|$ e $v \in \text{ter}[\mathcal{P}(v)]$.*

Demonstração. Durante a inicialização do algoritmo, para $v \in \text{ter}[\mathcal{P}]$ a propriedade claramente vale.

Suponha que a propriedade vale em uma iteração qualquer, e mostraremos que ela também vale na próxima. Ao ocorrer o caso 1a, temos que $|\mathcal{P}(u)| = |\mathcal{P}|$ e $u \in \text{ter}[\mathcal{P}(u)]$. Portanto, são substituídos os caminhos $P = (p_1, p_2, \dots, u)$ e $Q = (q_1, \dots, v^-, v, \dots, q_l)$ de $\mathcal{P}(u)$ pelos caminhos $R = (p_1, p_2, \dots, u, v, \dots, q_l)$ e $S = (q_1, \dots, v^-)$ em $\mathcal{P}(v^-)$. Logo, $|\mathcal{P}(v^-)| = |\mathcal{P}|$, e $v^- \in \text{ter}[\mathcal{P}(v^-)]$. \square

Lema 5.0.2. *Para v e w vértices de X , temos que $P(v, w)v = P(v)v$.*

Demonstração. Durante a inicialização do algoritmo, para quaisquer $v, w \in \text{ter}[\mathcal{P}]$, $P(v, w) = P(v)$ e a propriedade vale.

Suponha que a propriedade vale em uma iteração qualquer, e mostraremos que ela também vale na próxima. Assim como na demonstração do lema anterior, ao ocorrer o caso 1a, são substituídos os caminhos $P = (p_1, p_2, \dots, u)$ e $Q = (q_1, \dots, v^-, v, \dots, q_l)$ de $\mathcal{P}(u)$ pelos caminhos $R = (p_1, p_2, \dots, u, v, \dots, q_l)$ e $S = (q_1, \dots, v^-)$ em $\mathcal{P}(v^-)$. Pelo fato que $\mathcal{P}(v, u)v = Qv = \mathcal{P}(v)v$, temos que $\mathcal{P}(v^-, v^-)v^- = S = P(v^-)v^-$. Para $P(j, v^-)j$, onde $j \neq v^-$, temos que $P(j, v^-)j = P(j, u)j$.

Quanto a $P(k, l)k$, onde $l \neq v^-$, basta notar que as outras partições permanecem inalteradas após a iteração. Como a propriedade valia para estes segmentos de caminhos antes da iteração ser executada, ela continua valendo após a iteração. Logo, a propriedade vale durante toda a execução do algoritmo. \square

Assim, segue uma breve demonstração da corretude de Part-k-1-Alt.

Teorema 5.0.1. *\mathcal{P}' é uma partição em caminhos tal que $|\mathcal{P}'| = |\mathcal{P}| - 1$.*

Demonstração. Ao ocorrer o caso 1b pela execução do algoritmo, necessariamente pelo lema 5.0.1, $u \in \text{ter}[\mathcal{P}(u)]$, e pelo lema 5.0.2 temos que $P(v, u)v = P(v)v$. Como $v \in \text{in}[\mathcal{P}]$, então $v \in \text{in}[\mathcal{P}(u)]$. Como u é o terminal de um caminho, e v é o inicial de um outro caminho em $\mathcal{P}(u)$, é possível concatenar estes dois caminhos com a adição da aresta (u, v) a $\mathcal{P}(u)$. Obtemos, portanto \mathcal{P}' , tal que $|\mathcal{P}'| = |\mathcal{P}(u)| - 1 = |\mathcal{P}| - 1$. \square

Acredito que a mesma abordagem funcione para o caso das partições longas. No entanto, para o caso $k = 2$, essa simplificação é insuficiente, e a utilidade das trilhas alternantes mostra-se clara. Em certos casos de melhora (como, por exemplo, no caso 3c), o algoritmo Part-k-2 constrói uma nova partição em caminhos a partir de partes de diversas trilhas alternantes. Para fazer o mesmo para $k = 2$, um algoritmo do mesmo tipo que Part-k-1-Alt, que considera somente partições em caminhos possíveis na busca por uma partição menor, provavelmente teria que bifurcar a execução ao analisar caminhos curtos. Isso poderia acarretar em um número exponencial de partições analisadas, coisa que Part-k-2 engenhosamente evita pela adição de diversos pequenos casos particulares, e pela possibilidade de encontrar partições novas ao combinar partes de partições em caminhos encontradas por meio do uso de trilhas alternantes.

Referências Bibliográficas

- [1] J. A. Bondy, U. S. R. Murty. Graph Theory. 2008.
- [2] Douglas B. West. Introduction to Graph Theory, 2nd edition. 2001.
- [3] Reinhard Diestel. Graph Theory, 2nd edition. 1999.
- [4] M. R. Garey, D. S. Johnson. Computers and Intractability: A Guide to the theory of NP-Completeness. 1979.
- [5] Jorgen Bang-Jensen, Gregory Gutin. Digraphs Theory, Algorithms and Applications. 2007.
- [6] N. Linial. Covering digraphs by paths. *Discrete Math* 23:257-272, 1978.
- [7] N. Linial. Extending the Greene-Kleitman theorem to directed graphs. *J. Combinatorial Theory, Ser. A* 30:331-334, 1981.
- [8] Claude Berge. K-optimal partitions of a directed graph. *Eu. J. of Combinatorics*, 3:97-101, 1982.
- [9] K. Cameron. On K-optimal dipath partitions and partial k-colourings of acyclic digraphs. *Eu. J. of Combinatorics*, 7:115-118, 1986.
- [10] R. P. Dilworth. A decomposition theorem for partially ordered sets. *Annals of Mathematics*, 51:161-166, 1950.
- [11] C. Greene, D. J. Kleitman. The structure of Sperner k -families. *J. Combinatorial Theory, Ser. A* 20:41-68, 1976.
- [12] C. Greene. Some partitions associated with a partially-ordered set. *J. Combinatorial Theory, Ser. A* 20:69-79, 1976.

- [13] L. M. Vitaver. Determination of minimal colouring of vertices of a graph by means of Boolean powers of the incidence matrix.¹ *Dokl. Akad. Nauk SSSR*, 147:758-759, 1962.
- [14] M. Hasse. Zur algebraischen Begründung der Graphentheorie. *Math Nachr.* 28:275-290, 1964/1965.
- [15] B. Roy. Nombre chromatique et plus longs chemins d'un graphe. *Rev. Francaise Informat. Recherche Opérationnelle* 1:129-132, 1967.
- [16] T. Gallai. On directed paths and circuits. *Theory of Graphs (Proc. Coloq., Tihany, 1966)* Academic Press, New York, 115-118, 1968.
- [17] M. Saks. A short proof of the k -saturated partitions. *Adv. in Math* 33:207-211, 1979.
- [18] A. Frank. On Chain and antichain families of a partially ordered set. *J. Combinatorial Theory Ser. B* 29:176-184, 1980.
- [19] A. J. Hoffman. Extending Greene's theorem to directed graphs. *J. Combinatorial Theory Ser A.*, 34:102-107, 1983.
- [20] R. Aharoni, I. Ben-Arroyo Hartman. On Greene-Kleitman's theorem for general digraphs. *Discrete Math.*, 120:13-24, 1993.
- [21] R. Aharoni, I. Ben-Arroyo-Hartman, A. J. Hoffman. Path partitions and packs of acyclic digraphs. *Pacific Journal of Mathematics (2)*, 118:249-259, 1985.
- [22] I. Ben-Arroyo Hartman. An extension of Greene and Greene-Kleitman theorems to all digraphs. Relatório técnico, 2002.
- [23] Eli Berger, I. Ben-Arroyo Hartman. Proof of Berge's strong path partition conjecture for $k = 2$. *Eu. J. of Combinatorics*, 29:179-192, 2008.
- [24] Eli Berger, I. Ben-Arroyo Hartman. A unified approach to known and unknown cases of Berge's conjecture. *J. of Graph Theory*, 2011.

¹Título traduzido para o inglês.