**HOMEOMORFISMO EM GRAFOS:
ALGORITMOS E COMPLEXIDADE COMPUTACIONAL**

Haroldo Gonçalves Benatti

DISSERTAÇÃO APRESENTADA
AO
INSTITUTO DE MATEMÁTICA E ESTATÍSTICA
DA
UNIVERSIDADE DE SÃO PAULO

PARA OBTENÇÃO DO GRAU DE MESTRE
EM
MATEMÁTICA APLICADA

Área de Concentração: **Ciência da Computação**
Orientador: **Prof^a. Dr^a. Yoshiko Wakabayashi**

Durante a elaboração deste trabalho, o autor recebeu apoio financeiro do CNPq.

-São Paulo, 15 de setembro de 1993-

HOMEOMORFISMO EM GRAFOS:
ALGORITMOS E COMPLEXIDADE COMPUTACIONAL

Este exemplar corresponde à redação
final da dissertação devidamente corrigida
e defendida por Haroldo Gonçalves Benatti e aprovada
pela comissão julgadora.

São Paulo, 4 de novembro de 1992.

Banca examinadora:

- Prof^a. Dr^a. Yoshiko Wakabayashi (orientadora) - IME-USP
- Prof. Dr. Yoshiharu Kohayakawa - IME-USP
- Prof^a. Dr^a. Celina Miraglia Herrera de Figueiredo - COPPE-UFRJ

aos meus pais

Adalberto e Ivone

e a Luciene

Agradecimentos

Gostaria de agradecer a todos que colaboraram para que este trabalho assumisse esta forma final, mas em especial:

À professora Yoshiko Wakabayashi não só pela excelente orientação, mas também pela paciência, apoio e compreensão que foram tão importantes durante todo o mestrado.

Ao Coelho pelas valiosas dicas que me ajudaram a entender o algoritmo do Schrijver.

Ao Flávio, ao Ronaldo, à professora Maria Ângela Gurgel e ao Álvaro pelas discussões durante os nossos seminários, que foram muito importantes para a realização deste trabalho.

A todo o pessoal do Departamento de Ciência da Computação, mesmo os mais distantes (Karina, Carlinhos, ...) pela colaboração durante todo o mestrado.

Aos amigos que incentivaram na elaboração deste trabalho.

A Sandrinha que não se intimidou com a distância, se fazendo o mais presente possível e me dando a maior força.

A Regina e a Rosângela pelas sugestões (até mesmo aquelas mais “venenosas”) com relação aos meus grafos.

Finalmente, a Luciene e à minha família, por tudo ...

MUITO OBRIGADO !

Resumo

Neste trabalho estudamos vários problemas que envolvem homeomorfismo de grafos procurando responder questões referentes à sua complexidade computacional e à existência de algoritmos polinomiais para resolvê-los.

Dados dois grafos P e G , ambos orientados ou ambos não-orientados, um homeomorfismo de P em G é um par de funções injetoras (m, g) , onde m associa cada vértice de P a um vértice de G e g associa cada aresta $e = uv$ em P a um caminho de $m(u)$ a $m(v)$ em G . Se os caminhos que estão na imagem de g são dois a dois disjuntos nos vértices (nas arestas) então (m, g) é um vértice-homeomorfismo (aresta-homeomorfismo) de P em G . Dizemos que a função m é o vértice-mapeamento e que P é o grafo padrão. Os problemas estudados consistem em decidir se existe um vértice-homeomorfismo (aresta-homeomorfismo) de P em G . Os problemas diferem em função dos dados de entrada, que podem conter o vértice-mapeamento e/ou o grafo padrão ou nenhum deles. Quando o grafo padrão não faz parte da entrada dizemos que o mesmo é fixo.

Nesta dissertação estudamos relações entre aresta-homeomorfismo e vértice-homeomorfismo. Algumas destas relações são utilizadas para provar a $\mathcal{NP}$ -completude de vários problemas de homeomorfismo sem padrão fixo. No caso orientado, para os problemas com padrão fixo são conhecidos algoritmos polinomiais para apenas alguns padrões. Já no caso não-orientado, existe um algoritmo polinomial que resolve o problema para qualquer padrão, mas este algoritmo não é suscetível a uma implementação eficiente. Isto motivou o estudo destes problemas com entradas restritas a classes particulares de grafos. Apresentamos algoritmos polinomiais, em alguns casos bem eficientes, para a classe dos grafos planares.

Por último estudamos a complexidade de alguns problemas de modularidade de caminhos e circuitos em grafos orientados. Estes resultados foram obtidos usando problemas de homeomorfismo com padrão fixo. Analisamos a complexidade desses mesmos problemas quando restritos à classe dos grafos bipartidos e constatamos que esta restrição, em geral, não altera a complexidade destes problemas.

A maioria dos resultados que mencionamos foram encontrados na literatura. Algumas das demonstrações foram simplificadas, outras foram detalhadas provendo lemas auxiliares e provas de resultados apenas mencionados. Os estudos sobre a complexidade computacional dos problemas de modularidade de caminhos e circuitos em grafos bipartidos não foram encontrados na literatura.

Abstract

In this work we study many problems concerning homeomorphisms of graphs, establishing their computational complexity status and exhibiting polynomial algorithms for some of them.

Given two graphs P and G , both directed or both undirected, a homeomorphism of P in G is a pair of injective functions (m, g) , where m maps each vertex of P to a vertex of G and g maps each edge $e = uv$ in P to a path from $m(u)$ to $m(v)$ in G . When the paths in the image of g are pairwise vertex-disjoint (edge-disjoint) then (m, g) is a vertex-homeomorphism (edge-homeomorphism) of P in G . We say that m is the vertex-mapping and P is the pattern graph. The problems studied here consist in deciding whether P is homeomorphic to G . We consider all the variants which arise according to what we consider to be part of the input: the vertex-mapping and/or the pattern graph may be fixed or not, or both may be left unspecified.

In this dissertation we study the relationship between vertex-homeomorphism and edge-homeomorphism problems. Some of these relations are used to prove the $\mathcal{NP}$ -completeness of a number of problems for which the pattern graph is not fixed. When the pattern graph is fixed, in the directed case, polynomial algorithms are known for only a few patterns; in the undirected case, there exists a polynomial algorithm to solve the problem for any pattern graph, but the algorithm is not susceptible of an efficient implementation. This fact motivated the study of these problems with input restricted to special classes of graphs. We present polynomial algorithms, in some cases very efficient, for the class of planar graphs.

Finally we study the complexity of problems concerning the existence of paths and circuits of prescribed modularity in directed graphs. These results are proved using homeomorphism of graphs. We also analyse the complexity of these problems when the input graph is bipartite. We prove that this restriction, in general, does not change the complexity of these problems.

Most of the results presented here were found in the literature. Some proofs were simplified; others were detailed providing auxiliary lemmas and proofs of claimed results. The computational complexity status of the problems on paths and cycles of prescribed modularity in bipartite graphs were not found in the literature.

Introdução

O conceito de homeomorfismo em grafos já existe há bastante tempo. Em 1930, *K. Kuratowski* já usava este termo em seu famoso artigo “*On the topological problem of non-planar curves*”. Entretanto, apenas recentemente problemas envolvendo homeomorfismo de grafos têm sido mais estudados. Desde 1978, diversos problemas deste tipo têm surgido na literatura. Tais problemas podem ser vistos como uma generalização do problema de encontrar caminhos em um dado grafo. Daí o interesse por tais problemas. Dados dois grafos P e G , ambos orientados ou ambos não-orientados, dizemos que um par de funções injetoras (m, g) , onde m associa cada vértice de P a um vértice de G e g associa cada aresta $e = uv$ em P a um caminho em G de $m(u)$ a $m(v)$, é um homeomorfismo de P em G . Se os caminhos que estão na imagem de g são dois a dois disjuntos nos vértices (nas arestas) então (m, g) é um vértice-homeomorfismo (aresta-homeomorfismo) de P em G . Dizemos que a função m é um vértice-mapeamento e que P é o grafo padrão. Os problemas que estudaremos consistem, de modo geral, em decidir se um grafo P é vértice-homeomorfo (aresta-homeomorfo) a um subgrafo de G . Estes problemas diferem em função dos dados que são fornecidos na entrada do problema. Um problema possui vértice-mapeamento fixo se o vértice-mapeamento é fornecido na entrada do problema. Quando o grafo padrão P não faz parte da entrada do problema dizemos que o grafo padrão é fixo. Nesta dissertação, estudamos vários problemas que envolvem homeomorfismo de grafos procurando responder questões referentes à sua complexidade computacional e à existência de algoritmos polinomiais para resolvê-los.

A primeira pergunta que abordamos neste trabalho é a seguinte: Existe alguma relação entre os problemas que envolvem aresta-homeomorfismo e vértice-homeomorfismo? No **primeiro capítulo**, logo após definir os problemas, analisamos estas relações. Tais resultados foram originalmente apresentados por *LaPaugh e Rivest*.

No **segundo capítulo** desta dissertação analisamos a complexidade dos problemas de homeomorfismo sem padrão fixo, tanto no caso em que o vértice-mapeamento é fixo como no caso em que este não é fixo, nas versões para grafos orientados e grafos não-orientados. No caso em que o mapeamento é fixo, para grafos orientados, apresentamos uma demonstração da $\mathcal{NP}$ -*completude* do problema que é mais direta do que a de *LaPaugh e Rivest*, embora as idéias utilizadas tenham sido as mesmas. Também demonstramos que todos os problemas de homeomorfismo sem padrão fixo são $\mathcal{NP}$ -*completos*. Estes resultados são devidos a *LaPaugh e Rivest*. Procuramos elaborar uma exposição mais detalhada e didática dos mesmos.

Após saber que os problemas de homeomorfismo sem padrão fixo são $\mathcal{NP}$ -completos, pareceu-nos interessante estudar os problemas de homeomorfismo com padrão fixo, buscando caracterizar os padrões para os quais o problema é fácil. Fazemos isto no **terceiro capítulo** deste trabalho. Como veremos, no caso do problema do vértice-homeomorfismo com padrão e mapeamento fixo para grafos orientados existe uma tal caracterização. *Fortune, Hopcroft e Wyllie* usaram idéias e construções bastantes sofisticadas no artigo em que apresentaram esta caracterização. Procuramos deixar mais claras as idéias utilizadas por eles. A caracterização obtida por *Fortune, Hopcroft e Wyllie* mostra que o problema do homeomorfismo com padrão e mapeamento fixos para grafos orientados é difícil para a maioria dos grafos padrões. Isto nos forneceu motivação para estudarmos a complexidade deste problema quando restringimos a sua entrada a certas classes de grafos orientados. Apresentamos alguns resultados a este respeito. A restrição a algumas destas classes já havia sido investigada por *Fortune, Hopcroft e Wyllie*. Recentemente, *Robertson e Seymour* conseguiram um algoritmo polinomial para o problema do vértice-homeomorfismo com padrão e mapeamento fixos para grafos não-orientados. Este resultado é bastante interessante, do ponto de vista teórico, pois não se sabia quase nada a respeito deste problema até então. Discutiremos os principais resultados envolvendo esta classe de problemas.

No **quarto capítulo** estudamos problemas de homeomorfismo com mapeamento fixo, para certos padrões fixos, com entrada restrita a grafos planares. Inicialmente, estudamos o problema no qual o padrão consiste de k arcos disjuntos. Recentemente, *Schrijver* apresentou um algoritmo polinomial para este problema. Tal algoritmo é muito sofisticado e seu interesse é mais a nível teórico, do que prático, pois é difícil conceber uma boa implementação para o mesmo. O artigo em que *Schrijver* apresentou este algoritmo é extremamente sucinto. Talvez devido à dificuldade para explicar com detalhes as idéias um tanto engenhosas que foram utilizadas. Isto nos levou a introduzir alguns lemas com o objetivo de elucidar algumas das idéias utilizadas. Também procuramos demonstrar alguns dos resultados mencionados sem prova. Visando, ainda, facilitar a compreensão do algoritmo proposto, recorreremos a um exemplo ilustrativo da execução de uma parte do algoritmo. Este artigo consumiu-nos um grande tempo para sua compreensão, tendo sido um desafio para nós, apresentar as idéias nele contidas de forma que o leitor as entendesse e apreciasse a sua engenhosidade. Em seguida, estudamos o problema no qual o padrão consiste de k arestas múltiplas entre dois vértices, na versão para grafos não-orientados. *Suzuki, Akama, Nishizeki* apresentaram em 1989 um algoritmo $O(n \log n)$ para encontrar o número máximo de caminhos disjuntos entre dois vértices fixos em um grafo não-orientado planar de ordem n . Recentemente, *Ripphausen-Lipa, Wagner e Weihe* apresentaram um algoritmo linear para encontrar o número máximo de caminhos disjuntos entre dois vértices fixos em um grafo planar. Estes dois algoritmos utilizam idéias diferentes e bastante interessantes. Além disso, parecem possuir implementação muito eficientes, principalmente o segundo. Por estes motivos decidimos apresentá-los em detalhes neste nosso trabalho.

No **quinto capítulo**, os problemas de homeomorfismo com padrão e mapeamentos fixos na versão orientada, para certos padrões em que o problema é $\mathcal{NP}$ -*completo*, foram utilizados para demonstrar que certos problemas de modularidade de caminhos e circuitos em grafos orientados são $\mathcal{NP}$ -*completos*. Os resultados básicos deste capítulo são devidos a *Arkin, Papadimitriou e Yannakakis*. Além destes resultados já conhecidos, apresentamos alguns resultados não encontrados na literatura que obtivemos sobre a complexidade de certos problemas de modularidade de caminhos e circuitos quando restringimos a entrada do problema a grafos orientados bipartidos.

Conteúdo

1	Conceitos básicos e definições dos problemas	1
1.1	Conceitos básicos e notações	1
1.2	Homeomorfismo em grafos	5
1.3	Os problemas	7
1.4	Vértice-homeomorfismo $\times$ aresta-homeomorfismo	9
2	Homeomorfismo sem padrão fixo	15
2.1	Grafos orientados	15
2.2	Grafos não-orientados	21
3	Homeomorfismo com padrão fixo	29
3.1	Mapeamento fixo: caso orientado	29
3.1.1	PHF-O com padrão fixo	29
3.1.2	PHF-O com padrão fixo e entrada específica	41
3.1.3	PAHF-O com padrão fixo	47
3.2	Mapeamento fixo: caso não-orientado	47

3.3	Mapeamento livre	49
4	Homeomorfismo em grafos planares	51
4.1	PHF-O com padrão fixo genérico	52
4.1.1	Preliminares e notação	52
4.1.2	Funções homólogas	56
4.1.3	O algoritmo de Schrijver	57
4.1.4	Algoritmo para o problema da viabilidade cohomológica	73
4.2	PHF com padrão fixo específico	82
4.2.1	Algoritmos baseado em divisão-e-conquista	82
4.2.2	Algoritmo baseado em busca em profundidade	106
5	Caminhos e circuitos $p \bmod q$ em grafos orientados	117
5.1	Caso geral	118
5.2	Grafos bipartidos	123

Capítulo 1

Conceitos básicos e definições dos problemas

Neste capítulo apresentamos primeiramente alguns conceitos básicos da Teoria dos Grafos e estabelecemos a notação a ser utilizada. Definimos a seguir os vários tipos de homeomorfismo e os problemas que serão objeto de estudo desta dissertação. Nesta segunda parte também convencionamos algumas notações, sobretudo quanto aos nomes dos problemas que serão referenciados no decorrer desta dissertação. Na última seção, apresentamos alguns resultados que relacionam problemas de aresta-homeomorfismo e de vértice-homeomorfismo.

Neste trabalho faremos uso também de conceitos básicos da Teoria da Complexidade Computacional. Não incluímos aqui nenhuma seção sobre tais conceitos já que a terminologia utilizada segue exatamente a de Garey e Johnson [9].

1.1 Conceitos básicos e notações

A finalidade desta seção é fixar a terminologia e a notação relativas aos conceitos da Teoria dos Grafos que serão utilizados nesta dissertação. A terminologia adotada se aproxima mais a de Bondy e Murty [5] e, a menos de pequenas variações, coincide com a de outros textos introdutórios sobre grafos ([4, 13]).

Um grafo **não-orientado** é uma tripla $G = (VG, AG, \psi_G)$, onde VG é um conjunto finito de elementos chamados **vértices**, AG é um conjunto finito de elementos chamados **arestas** e ψ_G é uma função de incidência que associa cada aresta e de G a um par não-ordenado $\{v, w\}$ de vértices. Uma tal aresta e será indicada, quando não houver arestas múltiplas, por $\{v, w\}$ ou vw . Com esta convenção, a função de incidência fica implicitamente definida. As arestas associadas a um mesmo par de vértices são chamadas **arestas múltiplas**. Um **laço** é uma aresta da forma $\{v, v\}$. Um grafo sem laços e sem arestas múltiplas é um **grafo simples**. Um grafo G em que $VG = \emptyset$ é dito **grafo vazio**.

Os vértices v e w são os **extremos** da aresta vw . Dizemos que a aresta vw **incide** nos vértices v e w . Dois vértices u e v são **adjacentes** se existe uma aresta com extremos u e v . Chamamos **grau** de um vértice v em G , e indicamos $gr_G(v)$ ou, simplesmente, $gr(v)$, o número de arestas em G que incidem em v , contando os laços duas vezes.

Um **grafo orientado** $D = (VD, AD, \psi_D)$ é definido da mesma forma que um grafo não-orientado, exceto que a função de incidência ψ_D associa cada aresta a um par ordenado (v, w) de vértices. Geralmente, chamaremos as arestas de um grafo orientado de **arcos**. Um arco (v, w) será indicado, também, por vw . Dizemos que um arco $e = vw$ é um arco **de v para w** e que e **sai** de v e **entra** em w . Neste caso, o vértice v é a **cauda** de e e w é a **ponta** de e ; tanto um como outro são ditos **extremos** de e . Indicaremos a cauda de e por $c(e)$ e a ponta de e por $p(e)$. O arco wv é chamado **reverso** de vw . Chamamos **grau de saída** (**grau de entrada**) de um vértice v em D e indicamos por $gr_D^+(v)$ ($gr_D^-(v)$) ou, simplesmente $gr^+(v)$ ($gr^-(v)$), ao número de arcos em D que saem de v (entram em v). O **grau** de v em D , denotado por $gr_D(v)$ ou, simplesmente $gr(v)$, é o número inteiro $gr_D^+(v) + gr_D^-(v)$.

Eventualmente, dizemos grafo quando queremos nos referir indistintamente a grafo orientado ou grafo não-orientado ou quando o contexto permite distinguir se estamos nos referindo a grafo orientado ou grafo não-orientado.

Um grafo $H = (VH, AH, \psi_H)$ é um **subgrafo** de G , e indicamos $H \subseteq G$, se $VH \subseteq VG$, $AH \subseteq AG$ e ψ_H é a restrição de ψ_G a AH . Seja V' um subconjunto não-vazio de VG . O grafo $G - V'$ é o subgrafo de G obtido retirando de G os vértices que estão em V' juntamente com as arestas incidentes nestes vértices. O **grafo gerado por V'** , denotado por $G[V']$, é o subgrafo $G - (VG/V')$. Seja A' um subconjunto não-vazio de AG . O grafo $G - A'$ é o subgrafo de G obtido retirando de G as arestas que estão em A' . O

grafo gerado por A' , denotado por $G[A']$ é o subgrafo $G - (AG/A')$. Sejam G_1 e G_2 dois grafos, ambos orientados ou não-orientados, então $G_1 + G_2$ é o grafo cujo conjunto de vértices é $VG_1 \cup VG_2$ e cujo conjunto de arestas é $AG_1 \cup AG_2$. Seja G um grafo e $e = uv$ uma aresta de G . O grafo não-orientado obtido a partir de G pela **contração** de uma aresta $e = uv$ é o grafo H em que $VH = VG - \{v\}$ e $AH = (AG - \{xv : x \text{ é adjacente a } v \text{ em } G\}) \cup \{ux : x \text{ é adjacente a } v \text{ em } G \text{ e não é adjacente a } u; x \neq u\}$. Dado um grafo G , a operação de **inserir k vértices** (novos) $\{v_1, \dots, v_k\}$ em uma aresta $e = uv$ de G consiste em obter um novo grafo G' tal que $VG' = VG \cup \{v_1, \dots, v_k\}$ e $AG' = AG/\{uv\} \cup \{uv_1, v_1v_2, \dots, v_{k-1}v_k, v_kv\}$.

Dois grafos não-orientados (orientados) H e G são **isomorfos** se existem funções bi-jetoras $\phi : AH \rightarrow AG$ e $\psi : VH \rightarrow VG$ tais que $\phi(uv)$ tem extremos $\psi(u)$ e $\psi(v)$ (tem $\psi(u)$ como cauda e $\psi(v)$ como ponta), para cada aresta uv em AH .

Um **passeio** em um grafo G é uma seqüência $W = (v_0, e_1, v_1, \dots, e_k, v_k)$, onde $v_i \in VG$, para $0 \leq i \leq k$, e $e_j = v_{j-1}v_j \in AG$, para $1 \leq j \leq k$. Geralmente, indicaremos W apenas pela seqüência de seus vértices, ou seja, $W = (v_0, v_1, \dots, v_k)$. Denotaremos por VW o conjunto dos vértices de W e AW o conjunto das arestas de W . Dizemos que W é um passeio de v_0 a v_k ; que v_0 é o **início** e v_k é o **fim** de W . Os vértices v_0 e v_k são os **extremos** de W . Os vértices $v_1, \dots, v_{k-1}$ são os **vértices internos** de W . Dizemos que o passeio W **passa** pelo vértice v_i , $0 \leq i \leq k$, e pela aresta e_j , $1 \leq j \leq k$. O **comprimento** de um passeio W , denotado por $|W|$, é o número de arestas em W . Um passeio de comprimento nulo é chamado **degenerado**. Um passeio **mínimo** entre u e w é um passeio de menor comprimento possível entre u e w .

Uma **trilha** é um passeio cujas arestas são distintas. Um **caminho** é um passeio cujos vértices são distintos. Um passeio é dito **fechado** se tem início e fim iguais. Um **circuito** é uma trilha fechada de comprimento não nulo cujo início e vértices internos são todos distintos.

Cabe aqui ressaltar que a definição de **passeio** (caminho, circuito) num grafo orientado, como acabamos de apresentar, não segue a terminologia usualmente adotada. Para muitos autores, esta é a definição de *passeio (caminho, circuito) orientado*.

Em um grafo orientado D , um **passeio não-orientado** é uma seqüência da forma $W = (v_0, e_1, v_1, \dots, e_k, v_k)$, onde cada v_i é um vértice de D e cada e_j é um arco de D cujos

extremos são v_j e v_{j-1} . Note que podemos ter $e_j = v_{j-1}v_j$ ou $e_j = v_jv_{j-1}$. Dizemos que W passa por e_j no **sentido correto** se $e_j = v_{j-1}v_j$. Analogamente ao caso de passeio, define-se **caminho não-orientado** e **circuito não-orientado**.

Uma **seção** de um passeio (passeio não-orientado) $W = (v_0, e_1, v_1, \dots, e_k, v_k)$ é uma subsequência $(v_i, e_{i+1}, v_{i+1}, \dots, e_j, v_j)$, com $0 \leq i \leq j \leq k$. Indicaremos a seção de um passeio W entre os vértices u e v por $W[u, v]$. Sejam $W_1 = (v_0, e_1, v_1, \dots, e_k, v_k)$ e $W_2 = (u_0, a_1, u_1, \dots, a_l, u_l)$ dois passeios (passeios não-orientados) em G tais que $v_k = u_0$. Chamamos **concatenação** de W_1 e W_2 o passeio (passeio não-orientado) $W = (v_0, e_1, v_1, \dots, e_k, v_k = u_0, a_1, u_1, \dots, a_l, u_l)$ e o indicamos por $W_1 \cdot W_2$.

Um grafo orientado sem circuitos é chamado **acíclico**. Note que tal grafo pode ter circuitos não-orientados. Um grafo orientado D é **transitivo** se para quaisquer três vértices distintos u, v, w em D , sempre que $uv \in AD$ e $vw \in AD$ tem-se que $uw \in AD$.

Dizemos que dois passeios W_1 e W_2 são **disjuntos nos vértices** ou, simplesmente, **disjuntos** se os vértices de W_1 e W_2 são dois a dois distintos, com exceção, eventualmente, de seus extremos. Os passeios W_1 e W_2 são **disjuntos nas arestas** se $AW_1 \cap AW_2 = \emptyset$. Uma coleção de passeios disjuntos nos vértices (nas arestas) é uma coleção de passeios em G que são dois a dois disjuntos nos vértices (nas arestas).

Se G é um grafo e s e t são vértices distintos em uma mesma componente de G , dizemos que um subconjunto $X \subset \vec{V}G/\{s, t\}$ é um **corte** que separa s e t se em $G - X$ não existe nenhum caminho de s a t .

Dois vértices v e w em um grafo não-orientado (orientado) estão **ligados** se existe um passeio (passeio não-orientado) de v a w . A relação de ligação é uma relação de equivalência sobre os vértices do grafo e os subgrafos gerados pelas classes de equivalência são chamadas **componentes** do grafo. Um grafo que possui uma única componente é chamado **conexo**; um grafo com mais de uma componente é chamado **desconexo**.

Uma **floresta** é um grafo não-orientado (orientado) sem circuitos (sem circuitos não-orientados). Uma **árvore** é uma floresta conexa. Dada uma árvore T podemos fixar um vértice r como sendo a **raiz** de T . Neste caso, dizemos que T é uma **árvore com raiz** r . Se v é um vértice de uma árvore T com raiz r , então os vértices de T que estão no caminho (caminho não-orientado) de r a v , exceto v , são chamados **ancestrais** de v . Os

vértices que têm grau um em uma floresta são chamados **folhas** e os demais vértices são chamados **vértices internos** da floresta.

Um grafo G é **bipartido** se VG pode ser particionado em dois conjuntos V_1 e V_2 tais que toda aresta de G tem um extremo em V_1 e outro em V_2 . Denotaremos um tal grafo por $G(V_1, V_2)$. Se $|V_1| = n$, $|V_2| = m$ e $G(V_1, V_2)$ é um grafo não-orientado simples com nm arestas, então $G(V_1, V_2)$ será denotado por $K_{m,n}$.

Um grafo G é dito **planar** se pode ser desenhado no plano (representando vértices por pontos e arestas por curvas ligando seus extremos) de modo que suas arestas se interceptem apenas em seus extremos. Quando um tal desenho é dado, dizemos que o grafo G está **imerso** no plano. Uma **curva de Jordan** é uma curva contínua que não se intercepta. Dado um grafo G imerso no plano, dois pontos x e y do plano são ditos **conectados** se não estão em G e podem ser ligados por uma curva de Jordan cujos pontos não estão em G . Para um tal grafo G , a relação de conexão é uma relação de equivalência sobre os pontos no plano não pertencentes a G ; suas classes de equivalências são as **faces** de G . Note que uma das faces é ilimitada. Esta face é chamada **face externa**. Denotamos por FG o conjunto das faces de G . O passeio fechado que “contorna” uma face F em G é chamado **fronteira** de F . Indicamos por VF (AF) o conjunto dos vértices (arestas) que estão na fronteira de F . Note que cada arco $a \in AD$ separa duas faces de D : convencionaremos que a **face à esquerda (direita)** de a é a face que fica à esquerda (direita) de a quando caminhamos sobre o arco a indo de $c(a)$ para $p(a)$. Dado um grafo não-orientado (orientado) G , podemos definir um outro grafo G^* , chamado **dual** de G , da seguinte maneira: para cada face F de G existe um vértice F^* de G^* e para cada aresta e de G existe uma aresta e^* de G^* ; existe uma aresta com extremos F_1^* e F_2^* (um arco de F_1^* para F_2^*) em G^* se, e somente se, as faces F_1 e F_2 são separadas pela aresta e em G (F_1 e F_2 ficam à esquerda e à direita de e em G , respectivamente).

1.2 Homeomorfismo em grafos

Dados dois grafos P e G , ambos orientados ou ambos não-orientados, dizemos que P é **homeomorfo** a G se existe um par de funções injetoras (m, g) tais que m associa cada vértice de P a um vértice de G e g associa cada aresta $e = uv$ em P a um caminho em G de

$m(u)$ a $m(v)$, não-degenerado. Dizemos, neste caso, que (m, g) é um **homeomorfismo** de P em G . Usualmente, nos referiremos à função m por **vértice-mapeamento** ou, simplesmente, **mapeamento**. O grafo P será chamado **grafo padrão**.

Sejam P e G dois grafos, ambos orientados ou ambos não-orientados. Dizemos que P é **vértice-homeomorfo** (**aresta-homeomorfo**) a G se existe um homeomorfismo (m, g) de P em G tal que os caminhos que estão na imagem de g são dois a dois disjuntos nos vértices (nas arestas). Neste caso, dizemos que (m, g) é um **vértice-homeomorfismo** (**aresta-homeomorfismo**) de P em G . Usaremos a notação $P <_V G$ ($P <_A G$) para indicar que P é vértice-homeomorfo (aresta-homeomorfo) a G .

Figura 1.1: Exemplo de grafos P e G tais que $P <_V G$ e $P <_A G$.

Sejam P e G dois grafos, ambos orientados ou ambos não-orientados, e m uma função injetora que associa cada vértice de P a um vértice de G . Dizemos que P é **m -vértice-homeomorfo** (**m -aresta-homeomorfo**) a G se existe uma função injetora g que associa cada aresta de P a um caminho de G tal que (m, g) é um vértice-homeomorfismo (aresta-homeomorfismo) de P em G . Neste caso, dizemos que g é um **m -vértice-homeomorfismo** (**m -aresta-homeomorfismo**) de P em G . Usaremos a notação $P <_{V,m} G$ ($P <_{A,m} G$) para indicar que P é m -vértice-homeomorfo (m -aresta-homeomorfo) a G . Cabe aqui notar que, a rigor, deveríamos dizer P é (aresta) vértice-homeomorfo a um subgrafo de G mas, por simplicidade, omitiremos propositadamente o termo subgrafo.

Vale observar que uma definição alternativa, porém equivalente, para o conceito de vértice-homeomorfismo tem sido usada na literatura. Por esta definição alternativa, um grafo P é **vértice-homeomorfo** a G se for possível obter, a partir de P , um novo grafo P'

isomorfo a um subgrafo de G apenas inserindo vértices ao longo das arestas de P . Nós preferimos a definição em termos de caminhos de G pois assim o problema do vértice-homeomorfismo pode ser visto como uma generalização do problema de encontrar caminhos em G e assim poderemos naturalmente usar diversos algoritmos já elaborados para este último problema.

1.3 Os problemas

São vários os problemas na Teoria dos Grafos que envolvem o conceito de homeomorfismo. Estes têm sido investigado desde 1978, sendo vasta a literatura a respeito. Passaremos, neste momento, a definir alguns destes problemas. No que segue, ambos os grafos P e G são não-orientados.

Problema do aresta-homeomorfismo (PAH)

Entrada: Dois grafos P e G .

Pergunta: P é aresta-homeomorfo a G ?

Problema do vértice-homeomorfismo (PH)

Entrada: Dois grafos P e G .

Pergunta: P é vértice-homeomorfo a G ?

Problema do aresta-homeomorfismo com vértice-mapeamento fixo (PAHF)

Entrada: Dois grafos P e G e uma função injetora m de VP em VG .

Pergunta: P é m -aresta-homeomorfo a G ?

Problema do vértice-homeomorfismo com vértice-mapeamento fixo (PHF)

Entrada: Dois grafos P e G e uma função injetora m de VP em VG .

Pergunta: P é m -vértice-homeomorfo a G ?

Problema do aresta-homeomorfismo com padrão fixo P

(PAH com padrão P)

Entrada: Um grafo G .

Pergunta: P é aresta-homeomorfo a G ?

Problema do vértice-homeomorfismo com padrão fixo P

(PH com padrão P)

Entrada: Um grafo G .

Pergunta: P é vértice-homeomorfo a G ?

Problema do aresta-homeomorfismo com padrão fixo P e vértice-mapeamento fixo (PAHF com padrão P)

Entrada: Um grafo G e uma função injetora m de VP em VG .

Pergunta: P é m -aresta-homeomorfo a G ?

Problema do vértice-homeomorfismo com padrão fixo P e vértice-mapeamento fixo (PHF com padrão P)

Entrada: Um grafo G e uma função injetora m de VP em VG .

Pergunta: P é m -vértice-homeomorfo a G ?

Todos os problemas enunciados acima possuem a versão correspondente para grafos orientados. Tal versão, onde os grafos P e G são ambos orientados, será denotada pelo mesmo nome mnemônico seguido da letra O . Assim, por exemplo, **PAH-O** denota o PAH para grafos orientados.

O grafo G que faz parte das entradas dos problemas acima será chamado **grafo de entrada**. Observe que o grafo P , também, faz parte da entrada de alguns problemas. Porém nós o chamaremos, genericamente, **grafo padrão**.

Note que no caso em que o grafo padrão é fixo, a cada tal padrão corresponde um problema distinto.

Tendo introduzido os vários conceitos e problemas sobre homeomorfismo, passaremos

a responder, na próxima seção, a seguinte pergunta: existe alguma relação entre os problemas que envolvem aresta-homeomorfismo e vértice-homeomorfismo ?

1.4 Vértice-homeomorfismo $\times$ aresta-homeomorfismo

LaPaugh e Rivest [12] estudaram as relações entre os problemas que envolvem aresta-homeomorfismo e os que envolvem vértice-homeomorfismo. Nesta seção estudaremos estas relações. Veremos que no caso em que o padrão P não é fixo, existem reduções polinomiais diretas dos problemas de aresta-homeomorfismo para os problemas de vértice-homeomorfismo. Isto tanto no caso de grafos orientados ou não-orientados e com ou sem vértice-mapeamento fixo. Para dar uma visão global dos resultados desta parte, no final desta seção apresentamos um resumo esquemático dos resultados, indicado na Figura 1.3. Nesta figura as setas tracejadas indicam as reduções (de um problema A para um problema B) que faremos nesta seção. Para facilitar uma consulta posterior, seria interessante que o leitor indicasse nas setas os números dos teoremas correspondentes à redução.

O primeiro resultado que veremos é sobre a redutibilidade do PHF-O para o PAHF-O. O leitor que já estudou o teorema de Menger reconhecerá o “truque” utilizado. As reduções seguintes serão de problemas de aresta-homeomorfismo para problemas de vértice-homeomorfismo.

Teorema 1.4.1 *O PHF-O pode ser polinomialmente reduzido ao PAHF-O.*

Dem: Sejam P e G os grafos de entrada para o PHF-O e seja m o vértice-mapeamento de VP a VG que foi fixado.

Inicialmente, faça $G' = G$ e substitua cada vértice x em G' pelos vértices $c(x)$ e $p(x)$. Acrescente em G' um arco indo de $c(x)$ para $p(x)$. Finalmente, faça os arcos que entravam em x entrarem em $c(x)$ e os que saíam de x saírem de $p(x)$. De modo análogo, construa P' . Se $m(x) = y$ então faça $m'(p(x)) = p(y)$ e $m'(c(x)) = c(y)$.

Claramente, $P <_{V,m} G$ se, e somente se, $P' <_{A,m'} G'$.

□

Figura 1.2: Introdução de “switchs” em G .

Antes de prosseguirmos com o estudo das relações entre vértice-homeomorfismo e aresta-homeomorfismo vamos definir uma operação sobre um grafo (orientado ou não-orientado) para tornar mais clara a demonstração do próximo teorema.

Seja $e = uv$ uma aresta em um grafo (orientado ou não-orientado) G . Se $k_1, k_2 > 0$ a operação de **inserir em e k_1 vértices próximos a u e k_2 vértices próximos a v** consiste em substituir em G a aresta e pelo caminho $(u, u_1, \dots, u_{k_1}, v_1, \dots, v_{k_2}, v)$. Na Figura 1.2 são inseridos em e dois vértices próximos a u (u_1 e u_2) e dois vértices próximos a v (v_1 e v_2). Se $k_1 = 0$ e $k_2 > 0$ então tal operação consiste em substituir em G a aresta e pelo caminho $(u, v_1, \dots, v_{k_2}, v)$. Se $k_1 > 0$ e $k_2 = 0$ então tal operação consiste em substituir em G a aresta e pelo caminho $(u, u_1, \dots, u_{k_1}, v)$.

A próxima demonstração é bastante interessante, pois mostra que, com um pouco de engenhosidade, podemos fazer com que caminhos disjuntos nas arestas em um grafo correspondam a caminhos disjuntos nos vértices em um outro grafo.

Teorema 1.4.2 *O PAH-O (PAHF-O) é polinomialmente redutível ao PH-O (PHF-O).*

Dem: Sejam P e G os grafos de entrada para o PAH-O (PAHF-O).

Suponha inicialmente que o vértice-mapeamento não é fixo.

Iremos modificar o grafo G criando um “switch” para cada vértice de modo que caminhos disjuntos nas arestas contendo um vértice interno em comum possam ser transformados em caminhos disjuntos nos vértices usando caminhos existentes no “switch” que evitam este vértice. Considere um vértice y do grafo G . Insira $gr^+(y)$ vértices próximos a y em cada arco que entra em y e $gr^-(y)$ vértices próximos a y em cada arco que sai deste vértice. Veja a Figura 1.2. No novo grafo $\hat{G}$, cada arco original entrando em y é “conectado” a todos arcos originais saindo de y adicionando um arco, que chamaremos **arco de conexão**, de cada vértice inserido neste arco para um dos vértices inseridos em um dos arcos que saem de y . Em cada vértice inserido nos arcos que saem de y entra um arco de conexão. Note que a partir de um caminho no grafo original G que tem y como vértice interno podemos conseguir, no novo grafo $\hat{G}$, um caminho que evita y usando um arco de conexão.

Para ter certeza de que os vértices do grafo padrão não serão associados aos vértices de $V\hat{G} - VG$ construímos P' a partir de P , e G' a partir de $\hat{G}$ fazendo o seguinte: para cada vértice v de P (do grafo original G) adicionamos três novos vértices v_1 , v_2 e v_3 ao grafo P (ao grafo $\hat{G}$) e acrescentamos a P (a $\hat{G}$) os arcos (v, v_1) , (v, v_2) e (v, v_3) . Em G' , todos os vértices em $V\hat{G} - VG$ que foram inseridos em G para construir os “switchs” têm grau de saída igual a dois enquanto no novo grafo P' todos os vértices originais têm grau de saída no mínimo três. Logo, os vértices do grafo padrão original não poderão ser associados aos vértices de $V\hat{G} - VG$.

É fácil ver que P é aresta-homeomorfo a G se, e somente se, P' é vértice-homeomorfo a G' .

Suponha, agora, que o vértice-mapeamento seja fixo. Seja m o vértice-mapeamento fornecido como entrada para o PAHF-O.

Neste caso o “switch” que criaremos pode ser mais simples. Considere um vértice y do grafo G . Para conectar os arcos que entram em y com os arcos que saem de y nós não

precisamos inserir toda aquela quantidade de vértices que foi inserida no caso em que não tínhamos vértice-mapeamento fixo. Anteriormente, precisávamos porque tínhamos que ter um controle sobre o grau dos vértices que estávamos inserindo. Agora basta inserir um vértice em cada arco que entra ou sai de y e, como antes, adicionar um arco indo de cada vértice inserido nos arcos que entram em y para os vértices inseridos nos arcos que saem de y .

Claramente, P é m -aresta-homeomorfo a G se, e somente se, P é m -vértice-homeomorfo a G' .

□

De maneira similar pode-se provar o seguinte

Teorema 1.4.3 *O PAH (PAHF) é polinomialmente redutível ao PH (PHF).*

□

É interessante observar que no caso do PAHF-O e do PHF-O foram apresentadas reduções nos dois sentidos (Teoremas 1.4.1 e 1.4.2). Nos demais casos, a redução apresentada foi apenas do problema de aresta-homeomorfismo para o de vértice-homeomorfismo.

Como já mencionamos, na Figura 1.3 os resultados apresentados nesta seção estão indicados através de setas tracejadas. No próximo capítulo, estudaremos a complexidade dos problemas de homeomorfismo sem padrão fixo, e apresentaremos as reduções indicadas na Figura 1.3 através de setas contínuas.

Figura 1.3: Reduções entre os problemas de homeomorfismo sem padrão fixo.

Capítulo 2

Homeomorfismo sem padrão fixo

Neste capítulo estudaremos a complexidade computacional de vários problemas de homeomorfismo sem padrão fixo. Veremos que os problemas PAHF-O, PAH-O, PH-O, PHF-O, PAHF, PAH, PH, PHF são todos $\mathcal{NP}$ -completos. Como já dissemos no capítulo anterior, na Figura 1.3 as setas contínuas indicam as reduções que faremos neste capítulo.

Na primeira seção apresentamos os resultados para grafos orientados, ficando a seção seguinte para grafos não-orientados.

2.1 Grafos orientados

Claramente os problemas PAHF-O, PAH-O, PH-O, PHF-O, PAHF, PAH, PH, PHF estão em $\mathcal{NP}$. Talvez não seja tão surpreendente o fato de que os problemas de homeomorfismo sem padrão fixo e sem vértice-mapeamento fixo para grafos orientados sejam $\mathcal{NP}$ -completos. Mas, como veremos, mesmo com vértice-mapeamento fixo, estes continuam sendo $\mathcal{NP}$ -completos.

Even, Itai e Shamir [6] provaram que o problema do dois-fluxo inteiro é $\mathcal{NP}$ -completo, fazendo uma redução do problema da satisfatibilidade para este problema. LaPaugh e Rivest [12] usaram este resultado para provar que o PAHF-O é $\mathcal{NP}$ -completo. Demonstraremos que o PAHF-O é $\mathcal{NP}$ -completo reduzindo o problema da satisfatibilidade,

diretamente, ao PAHF-O, sem falar do problema do dois-fluxo inteiro. As idéias usadas na demonstração que apresentaremos são, essencialmente, as mesmas que foram usadas por Even, Itai e Shamir.

Apresentaremos a seguir o problema da satisfatibilidade que será usado na próxima redução. Antes disso, definiremos alguns termos.

Seja $U = \{x_1, \dots, x_l\}$ um conjunto de **variáveis** lógicas. Uma **atribuição** de valores para U é uma função $f : U \rightarrow \{V, F\}$. Se $f(x) = V$ dizemos que a variável x é **verdadeira** para a atribuição de valores f ; se $f(x) = F$ então dizemos que x é **falsa**. Se x é uma variável em U , então x e $\bar{x}$ são **literais** sobre U . O literal x é verdadeiro se, e somente se, a variável x é verdadeira; o literal $\bar{x}$ é verdadeiro se, e somente se, a variável x é falsa. Uma **cláusula** D é um conjunto finito de literais sobre U . Uma cláusula representa a disjunção dos literais que a formam e é **satisfeita** por uma atribuição de valores se, e somente se, um dos seus literais é verdadeiro para esta atribuição de valores. Uma **fórmula** $\mathcal{F}$ é uma coleção finita de cláusulas. Uma fórmula $\mathcal{F}$ sobre U é **satisfatível** se, e somente se, existe uma atribuição de valores para U que satisfaz cada uma das cláusulas na fórmula $\mathcal{F}$. Neste caso, dizemos que esta atribuição de valores **satisfaz** $\mathcal{F}$. Podemos, agora, definir:

Problema da satisfatibilidade (SAT)

Entrada: Uma fórmula $\mathcal{F}$ sobre um conjunto U de variáveis.

Pergunta: Existe uma atribuição de valores para U que satisfaz $\mathcal{F}$?

É um fato bastante conhecido que o problema da satisfatibilidade é $\mathcal{NP}$ -completo (veja [15]). Usaremos este fato na demonstração do próximo teorema.

Figura 2.1: Bifurcação B_i (associada à variável x_i).

Teorema 2.1.1 *O PAHF-O é $\mathcal{NP}$ -completo.*

Dem: É suficiente mostrar que o SAT é polinomialmente redutível ao PAHF-O.

Seja $\mathcal{F}$ a fórmula fornecida como entrada para o SAT. Sejam $D_1, D_2, \dots, D_k$ as cláusulas da fórmula $\mathcal{F}$ e $x_1, x_2, \dots, x_l$ as variáveis de $\mathcal{F}$. Usaremos a fórmula $\mathcal{F}$ para construir os grafos orientados P e $D_{\mathcal{F}}$ que serão os grafos fornecidos como entrada para o PAHF-O.

Primeiro, construiremos o grafo de entrada $D_{\mathcal{F}}$. Inicialmente, $D_{\mathcal{F}}$ é o grafo formado por l **bifurcações** $B_1, B_2, \dots, B_l$, sendo cada bifurcação B_i um grafo associado à variável x_i , definido de acordo com a Figura 2.1. Nesta figura, p_i denota o número de ocorrências do literal x_i na fórmula $\mathcal{F}$ e q_i denota o número de ocorrências do literal $\bar{x}_i$ em $\mathcal{F}$. Se $p_i = 0$ ($q_i = 0$) então na bifurcação B_i temos o arco (v_s^i, v_i^i) no lugar do caminho $(v_s^i, v_1^i, v_2^i, \dots, v_{2p_i}^i, v_i^i)$ $((v_s^i, \bar{v}_1^i, \bar{v}_2^i, \dots, \bar{v}_{2q_i}^i, v_i^i))$.

Na bifurcação B_i os arcos $(v_s^i, v_1^i), (v_1^i, v_2^i), \dots, (v_{2p_i-1}^i, v_{2p_i}^i), (v_{2p_i}^i, v_i^i)$ são chamados **arcos superiores** da bifurcação B_i e os vértices $v_1^i, \dots, v_{2p_i}^i$ são chamados **vértices superiores** da bifurcação B_i . Os arcos $(v_s^i, \bar{v}_1^i), (\bar{v}_1^i, \bar{v}_2^i), \dots, (\bar{v}_{2q_i}^i, v_i^i)$ são chamados **arcos inferiores** da bifurcação B_i e os vértices $\bar{v}_1^i, \dots, \bar{v}_{2q_i}^i$ são chamados **vértices inferiores** da bifurcação B_i .

A seguir acrescentamos a $D_{\mathcal{F}}$ os vértices s'_1, t'_1 e os arcos (v_i^i, v_s^{i+1}) , para $i = 1, \dots, l-1$; (s'_1, v_s^1) ; (v_l^l, t'_1) e estes arcos são chamados **arcos de conexão**.

Acrescentamos, ainda, a $D_{\mathcal{F}}$ os vértices s'_2, t'_2 e os arcos $(s'_2, v_j^i), (s'_2, \bar{v}_j^i)$, onde $i = 1, \dots, l$ e j é ímpar. Além disso, acrescentamos a $D_{\mathcal{F}}$ os vértices $D_1, \dots, D_k$, associados às cláusulas de $\mathcal{F}$, e um arco indo de cada um destes vértices para o vértice t'_2 . Para a j -ésima ocorrência do literal x_i ($\bar{x}_i$) na fórmula $\mathcal{F}$ acrescentamos a $D_{\mathcal{F}}$ um arco indo de v_{2j}^i ($\bar{v}_{2j}^i$) para o vértice D_r associado à cláusula r em que acontece a j -ésima ocorrência deste literal.

Agora, construiremos o grafo padrão P . O grafo P terá quatro vértices, digamos s_1, t_1, s_2, t_2 , e terá um arco de s_1 para t_1 e k arcos múltiplos de s_2 para t_2 .

O vértice-mapeamento m que será fornecido na entrada para o PAHF-O será o seguinte:

$$m(s_1) = s'_1, m(t_1) = t'_1, m(s_2) = s'_2, m(t_2) = t'_2.$$

Mostraremos a seguir que P é m -aresta-homeomorfo a $D_{\mathcal{F}}$ se, e somente se, a fórmula $\mathcal{F}$ é satisfatível.

Suponha que $P <_{A,m} D_{\mathcal{F}}$. Seja g um m -aresta-homeomorfismo de P em $D_{\mathcal{F}}$. Considere os caminhos em $D_{\mathcal{F}}$ definidos por g . O caminho que vai de s'_1 a t'_1 , que chamaremos de C , passa por todos os arcos de conexão e em cada bifurcação passa ou por todos arcos superiores ou por todos arcos inferiores. Cada caminho que vai de s'_2 a t'_2 passa por exatamente uma bifurcação, pois os arcos de conexão são usados pelo caminho C . Além disso, sabemos que existem k destes caminhos, que estes caminhos são dois a dois disjuntos nos arcos e que existem exatamente k arcos entrando em t'_2 , que são aqueles que saem dos vértices D_j , com $j = 1, \dots, k$. Logo cada um destes vértices está exatamente em um dos k caminhos de s'_2 a t'_2 . Chamaremos de C_j o caminho de s'_2 a t'_2 que passa pelo vértice D_j , para $j = 1, \dots, k$.

Cada um dos caminhos C_j é de um dos seguintes tipos : vai de s'_2 para algum vértice superior (inferior) de alguma bifurcação, digamos B_i , passa pelo arco superior (inferior) que sai deste vértice (pois este é o único arco que sai deste vértice) e possivelmente mais alguns arcos superiores (inferiores) da bifurcação B_i . Passa por um arco que vai de um vértice superior (inferior) de B_i para o vértice D_j e, finalmente, alcança o vértice t'_2 .

Deste modo, se um caminho C_j usa um arco superior da bifurcação B_i então a cláusula D_j contém o literal x_i , pois em $D_{\mathcal{F}}$ existe um arco de um vértice superior da bifurcação B_i para o vértice D_j . Se um caminho C_j usa um arco inferior de B_i , a cláusula D_j contém o literal $\bar{x}_i$.

Podemos então fazer a seguinte atribuição de valores às variáveis de $\mathcal{F}$: x_i será verdadeira se, e somente se, o caminho C usar os arcos inferiores da bifurcação B_i .

Considere o caminho C_j , onde $j \in \{1, \dots, k\}$.

Se C_j usa um arco superior de B_i então D_j contém o literal x_i . Como C_j e C são disjuntos nas arestas então, certamente, o caminho C passa pelos arcos inferiores de B_i . Logo, x_i é verdadeira e a cláusula D_j é satisfeita.

Se C_j usa um arco inferior de B_i então D_j contém o literal $\bar{x}_i$. Neste caso o caminho C passa pelos arcos superiores de B_i . Logo, x_i é falsa e a cláusula D_j é satisfeita. Portanto, a atribuição de valores definida satisfaz $\mathcal{F}$.

Agora, suponha que a fórmula $\mathcal{F}$ é satisfatível. Considere uma atribuição de valores às variáveis $x_1, \dots, x_l$ que satisfaz a fórmula $\mathcal{F}$.

Para mostrar que $P <_{A,m} D_{\mathcal{F}}$ é suficiente mostrar que existem k caminhos de s'_2 a t'_2 e um caminho de s'_1 a t'_1 que são dois a dois disjuntos nos arcos.

Considere o caminho de s'_1 a t'_1 que na bifurcação B_i , para $i = 1, \dots, l$, passa pelos arcos inferiores se, e somente se, x_i é verdadeira. Seja C este caminho. Agora mostraremos como construir os caminhos de s'_2 a t'_2 . Como a atribuição de valores fixada para as variáveis satisfaz $\mathcal{F}$, então cada cláusula D_j possui um literal que é verdadeiro. Fixe j e escolha um literal na cláusula D_j que seja verdadeiro.

Suponha que o literal escolhido tenha sido x_i . Suponha que este literal ocorre pela p -ésima vez em $\mathcal{F}$ na cláusula D_j . Seja p a ordem desta ocorrência do literal x_i na fórmula $\mathcal{F}$. Neste caso, tome $C_j = (s'_2, v_{2p-1}^i, v_{2p}^i, D_j, t'_2)$. Observe que C e C_j não possuem arcos em comum, pois na bifurcação B_i o caminho C passa pelos arcos inferiores.

Agora, suponha que o literal escolhido tenha sido $\bar{x}_i$. Seja q a ordem desta ocorrência do literal $\bar{x}_i$ na fórmula $\mathcal{F}$. Tome, então, $C_j = (s'_2, \bar{v}_{2q-1}^i, \bar{v}_{2q}^i, D_j, t'_2)$. Observe que C e C_j não possuem arcos em comum, pois na bifurcação B_i o caminho C passa pelos arcos superiores.

Executando o procedimento descrito acima para $j = 1, \dots, k$ obteremos os caminhos $C_1, \dots, C_k$ que não possuem arcos em comum com C . Além disso, estes caminhos são dois a dois disjuntos nos arcos. Portanto, P é m -aresta-homeomorfo a $D_{\mathcal{F}}$.

□

No que segue nos referiremos à seguinte classe $\mathcal{P}$ de grafos:

$$\mathcal{P} = \{P_j: j \in \mathbb{Z}_+\},$$

onde P_j é o grafo indicado na Figura 2.2.

O corolário seguinte, embora pareça um tanto restrito, é a base para as demonstrações de vários resultados que apresentaremos neste capítulo.

Figura 2.2: O grafo P_j .

Corolário 2.1.2 *O PAHF-O com grafo padrão restrito à classe $\mathcal{P}$ é $\mathcal{NP}$ -completo.*

□

No que segue veremos que os problemas de homeomorfismo sem vértice-mapeamento fixo são tão difíceis quanto os seus correspondentes com vértice-mapeamento fixo. As demonstrações são devidas a LaPaugh e Rivest [12].

Teorema 2.1.3 *O PAH-O é $\mathcal{NP}$ -completo.*

Dem: É suficiente mostrar que o PAHF-O com grafo padrão restrito à classe $\mathcal{P}$ pode ser reduzido ao PAH-O. Sejam os grafos P (com $P \in \mathcal{P}$) e D e a função $m : VP \rightarrow VD$ definida por $m(s_1) = s'_1$, $m(t_1) = t'_1$, $m(s_2) = s'_2$, $m(t_2) = t'_2$ a entrada para o PAHF-O.

Seja $n - 1$ o grau de entrada máximo em D . Inicialmente, faça $P' = P$ e $D' = D$. Acrescente a P' (D') $4n$ novos vértices e conecte-os ao vértice s_1 (s'_1) usando arcos que saem destes novos vértices e entram em s_1 (s'_1). De modo similar conecte $3n$ novos vértices a t_1 (t'_1), $2n$ novos vértices a s_2 (s'_2) e n novos vértices a t_2 (t'_2).

Vamos mostrar que $P <_{A,m} D$ se, e somente se, $P' <_A D'$.

Se P é m -aresta-homeomorfo a D então é imediato que P' é aresta-homeomorfo a D' . Suponha que P' é aresta-homeomorfo a D' . Seja (m', g') um aresta-homeomorfismo de P' em D' . Como o único vértice de D' que possui grau de entrada no mínimo $4n$ é s'_1 então o vértice s_1 deve estar associado a s'_1 . Como o único vértice de D' , distinto de s'_1 , que possui grau de entrada no mínimo $3n$ é t'_1 então o vértice t_1 deve estar associado a t'_1 . Continuando assim obtemos que s_2 deve estar associado a s'_2 e que t_2 deve estar associado a t'_2 . Seja g a função g' restrita a AP . Claramente, (m, g) é um aresta-homeomorfismo de P em D . Isto completa a demonstração de que o PAH-O é $\mathcal{NP}$ -completo.

□

Como os caminhos construídos na demonstração do Teorema 2.1.1 são disjuntos nos vértices então vale o seguinte teorema.

Teorema 2.1.4 *O PHF-O é $\mathcal{NP}$ -completo.*

□

O Teorema 2.1.1 e a redução no Teorema 1.4.2 fornecem que o PHF-O com padrão restrito à classe $\mathcal{P}$ é $\mathcal{NP}$ -completo. Assim, com uma demonstração análoga à do Teorema 2.1.3 pode-se provar o próximo teorema.

Teorema 2.1.5 *O PH-O é $\mathcal{NP}$ -completo.*

□

2.2 Grafos não-orientados

Veremos, nesta seção, que todos os problemas de homeomorfismo sem padrão fixo para grafos não-orientados são $\mathcal{NP}$ -completos. Os resultados desta seção são devidos a Even, Itai e Shamir [6].

Teorema 2.2.1 *O PAHF é $\mathcal{NP}$ -completo.*

Dem: Pelo Corolário 2.1.2, é suficiente mostrar que o PAHF-O com grafo padrão restrito à classe $\mathcal{P}$ pode ser reduzido ao PAHF.

Sejam os grafos orientados P (com $P \in \mathcal{P}$) e D os grafos fornecidos na entrada do PAHF-O. Seja $m(s_1) = \bar{s}_1$, $m(t_1) = \bar{t}_1$, $m(s_2) = \bar{s}_2$, $m(t_2) = \bar{t}_2$ a função de VP em VD fornecida na entrada do PAHF-O. Seja k o número de arcos múltiplos em P com extremos s_2 e t_2 . Assim $P = P_k$ (cf. Figura 2.2).

Seja D' o grafo orientado obtido ao acrescentarmos a D quatro novos vértices, digamos s'_1, s'_2, t'_1, t'_2 , um arco de s'_1 a $\bar{s}_1$, um arco de $\bar{t}_1$ a t'_1 , k arcos múltiplos de s'_2 a $\bar{s}_2$ e k arcos múltiplos de $\bar{t}_2$ a t'_2 .

Seja G o grafo não-orientado obtido de D' da seguinte maneira:

Inicialmente, faça $G = D'$. A seguir substitua cada arco uv em G pela estrutura B_{uv} apresentada na Figura 2.3 (os vértices u ou v podem ser um dos vértices s'_1, s'_2, t'_1, t'_2). Nos referiremos a esta estrutura como **bloco**. Vale observar que somente os vértices não-rotulados em cada bloco são novos e estes vértices são todos distintos.

Seja H o grafo não-orientado formado por quatro vértices s_1, s_2, t_1, t_2 , $1 + |AD'|$ arestas múltiplas com extremos s_1 e t_1 e $k + |AD'|$ arestas múltiplas com extremos s_2 e t_2 .

Seja $m' : \{s_1, s_2, t_1, t_2\} \rightarrow VG$ definida por: $m'(s_1) = s'_1$, $m'(t_1) = t'_1$, $m'(s_2) = s'_2$, $m'(t_2) = t'_2$.

Claramente, $P <_{A,m} D$ se, e somente se, $P <_{A,m'} D'$. Logo, é suficiente mostrar que $P <_{A,m'} D'$ se, e somente se, $H <_{A,m'} G$.

Suponha, inicialmente, que P é m' -aresta-homeomorfo a D' . É fácil obter $|AD'|$ caminhos em G de s'_i a t'_i , para $i = 1, 2$. Basta escolher em cada bloco os caminhos (com arestas escuras) mostrados na Figura 2.4. Para provar que $H <_{A,m'} G$ é suficiente mostrar que no grafo G existem $1 + |AD'|$ caminhos de s'_1 a t'_1 e $k + |AD'|$ caminhos de s'_2 a t'_2 que são dois a dois disjuntos nas arestas. A seguir mostraremos como obtê-los a partir dos $|AD'|$ caminhos selecionados inicialmente.

Figura 2.3: Bloco B_{uv} (associado ao arco uv).

Figura 2.4: Escolha inicial das arestas em B_{uv} .

Figura 2.5: Escolha das arestas no bloco B_{uv} , quando um dos caminhos de g' passa por uv e vai de s'_1 a t'_1 .

Seja g' um m' -aresta-homeomorfismo de P em D' . Considere os caminhos de D' que estão na imagem de g' . Se um arco $uv \in AD'$ está em um destes caminhos, nós substituímos o caminho selecionado no bloco B_{uv} em G (correspondente ao arco uv) pelos dois caminhos apresentados na Figura 2.5, se o caminho que passa por uv vai de s'_1 a t'_1 ; ou pelos dois caminhos apresentados na Figura 2.6, se o caminho que passa por uv vai de s'_2 a t'_2 .

Repetindo o procedimento descrito acima para cada arco de D' e conectando os caminhos que têm um vértice em comum obteremos $1 + |AD'|$ caminhos em G de s'_1 a t'_1 e $k + |AD'|$ caminhos em G de s'_2 a t'_2 , que são dois a dois disjuntos nas arestas. Vejamos porque isto acontece.

Considere o caminho de s'_1 a t'_1 que fixamos em D' . Seja C este caminho. Fazendo as substituições de caminhos nos blocos associados aos arcos que fazem parte do caminho C aumentamos de uma unidade a quantidade de caminhos de s'_1 a t'_1 selecionados. Esta afirmação é válida, pois no bloco correspondente ao arco de C que sai de s'_1 existe um caminho de s'_1 a t'_1 , no bloco correspondente ao arco de C que entra em t'_1 existe mais um caminho deste tipo e para cada par de arcos consecutivos em C existe um caminho de s'_1

Figura 2.6: Escolha das arestas no bloco B_{uv} , quando um dos caminhos de g' passa por uv e vai de s'_2 a t'_2 .

a t'_1 começando no bloco correspondente ao primeiro arco do par e terminando no bloco correspondente ao segundo arco.

De modo similar podemos mostrar que ao executarmos o procedimento descrito acima para os k caminhos de s'_2 a t'_2 que fixamos em D' obteremos $k + |AD'|$ caminhos de s'_2 a t'_2 em G . Claramente, estes caminhos são dois a dois disjuntos nas arestas.

Agora, suponha que H é m' -aresta-homeomorfo a G . No grafo G escolha $1 + |AD'|$ caminhos de s'_1 a t'_1 e $k + |AD'|$ caminhos de s'_2 a t'_2 que são dois a dois disjuntos nas arestas. Tais caminhos existem pois $H <_{A,m'} G$. Como o número de arestas em G incidentes em cada um dos vértices s'_1 e t'_1 é $1 + |AD'|$ e o número de arestas incidentes em cada um dos vértices s'_2 e t'_2 é $k + |AD'|$, então cada uma das arestas incidentes em s'_1 , t'_1 , s'_2 ou t'_2 está em um dos caminhos que escolhemos.

Assim em cada bloco as arestas que estão nos caminhos seleccionados podem ser de um dos seguintes tipos:

- (a) arestas marcadas na Figura 2.4 ou
- (b) arestas marcadas na Figura 2.5 ou
- (c) arestas marcadas na Figura 2.6.

Para mostrar que $P <_{A,m'} D'$ vamos mostrar que em D' existem k caminhos de s'_2 a t'_2 e um caminho de s'_1 a t'_1 que são dois a dois disjuntos nos arcos. Considere os arcos em D' associados aos blocos em G cujas arestas que estão nos caminhos selecionados em G são do tipo marcadas na Figura 2.5. Estes arcos geram em D' um caminho de s'_1 a t'_1 . Os arcos de D' associados aos blocos em G cujas arestas que estão nos caminhos escolhidos em G são do tipo marcadas na Figura 2.6 geram k caminhos de s'_2 a t'_2 . Claramente, estes caminhos são dois a dois disjuntos nos arcos. Isto completa a demonstração de que o PAHF é $\mathcal{NP}$ -completo.

□

De modo similar àquele feito na demonstração do Teorema 2.1.3 podemos provar o seguinte teorema.

Teorema 2.2.2 *O PAH é $\mathcal{NP}$ -completo.*

□

Segue do Teorema 1.4.3 e do Teorema 2.2.1 o seguinte corolário.

Corolário 2.2.3 *O PHF é $\mathcal{NP}$ -completo.*

□

Segue dos Teoremas 1.4.3 e 2.2.2 o seguinte teorema.

Teorema 2.2.4 *O PH é $\mathcal{NP}$ -completo.*

□

Finalmente, se usarmos a designação de **problemas gerais de homeomorfismo** para os problemas PAHF, PAH, PHF, PH, nas versões para grafos orientados e não-orientados, podemos formular o seguinte

Teorema 2.2.5 *Todos os problemas gerais de homeomorfismo são $\mathcal{NP}$ -completos.*

□

No próximo capítulo estudaremos os problemas de homeomorfismo com padrão fixo para grafos orientados e não-orientados. Vale observar que o fato do PHF ser $\mathcal{NP}$ -completo não implica que o PHF com padrão P , para algum P , seja $\mathcal{NP}$ -completo. Além disso, a prova de que os problemas PHF-O e PAHF-O são equivalentes, também não implica que o PHF-O com padrão P seja equivalente ao PAHF-O com o mesmo padrão P , pois na demonstração do Teorema 1.4.1 os grafos padrões ali tratados são distintos.

Capítulo 3

Homeomorfismo com padrão fixo

Neste capítulo estudaremos os problemas de homeomorfismo com padrão fixo. Primeiramente, veremos o caso em que o vértice-mapeamento é fixo e, em seguida, o caso em que o vértice-mapeamento é livre.

3.1 Mapeamento fixo: caso orientado

Nesta seção estudaremos o PHF-O com padrão fixo. Fortune, Hopcroft e Wyllie [7] caracterizaram os grafos padrões para os quais o problema é fácil. Este resultado será apresentado na primeira parte desta seção. Em seguida, estudaremos o PHF-O com padrão fixo com entrada restrita a algumas classes de grafos, com o objetivo de verificar se o problema em questão se torna mais fácil quando impomos estas restrições. No final desta seção analisaremos a complexidade do PAHF-O.

3.1.1 PHF-O com padrão fixo

Seja $\mathcal{T}$ o conjunto dos grafos orientados que possuem um nó distinto, chamado **raiz**, que é a ponta de todos os arcos ou é a cauda de todos os arcos. Os grafos que estão em $\mathcal{T}$ são conhecidos como **in-tree** e **out-tree**. Veja exemplos destes grafos na seguinte figura:

Figura 3.1: Exemplo de out-tree e in-tree.

É bastante conhecido o fato de que o problema do fluxo máximo pode ser resolvido em tempo polinomial (veja Papadimitriou e Steiglitz [15]). Este fato será usado na demonstração do próximo teorema, que é devido a Fortune, Hopcroft e Wyllie [7].

Teorema 3.1.1 *Para cada grafo orientado P em $\mathcal{T}$ existe um algoritmo polinomial para o PHF-O com padrão P .*

Dem: Seja $P \in \mathcal{T}$. Vamos considerar o caso em que todos arcos de P saem da raiz. O outro caso pode ser provado de maneira similar. Seja D o grafo orientado e m o vértice-mapeamento fornecidos como entrada para o PHF-O com padrão P . Para todo v em P , faremos $v' = m(v)$.

Inicialmente, observamos que se existem laços na raiz r de P , então podemos obter um PHF-O com um novo grafo padrão que não possui laços e que é equivalente ao primeiro. Para obter este novo grafo padrão acrescentamos a P um novo vértice, que chamaremos f , e substituímos os laços existentes em P por arcos que vão de r para f . Seja P' este novo grafo (veja Figura 3.2). No grafo D acrescentamos um novo vértice f' e substituímos todos os arcos ur' que entravam no vértice r' por arcos uf' . Seja D' este novo grafo. Seja m' o mapeamento dos vértices de P' em D' que associa f ao vértice f' e que para os demais vértices faz a mesma associação que m . Claramente, P é m -vértice-homeomorfo a D se, e somente se, P' é m' -vértice-homeomorfo a D' .

Seja D'' o grafo obtido de D' substituindo cada vértice v por dois vértices v_1 e v_2 e um arco (v_1, v_2) (veja Figura 3.3). Faça entrar em v_1 todos os arcos que entravam em v e

Figura 3.2: Construção dos grafos D' e P' .

Os arcos tracejados correspondem a vértices de D' .
Os arcos não-rotulados possuem capacidade unitária.

Figura 3.3: Grafo D'' .

sair de v_2 todos os arcos que saíam de v . Atribua ao arco (r'_1, r'_2) capacidade igual ao grau de saída da raiz r' de P' . Para cada arco (u'_1, u'_2) em D'' , em que u' é a imagem de um vértice de P' distinto de r , atribua capacidade igual ao grau de entrada deste vértice em P' . Atribua capacidade um aos demais arcos de D'' . Considere r'_1 como origem e todos os vértices u'_2 ($u \in VP - \{r\}$) como destinos. Claramente, P' é m' -vértice-homeomorfo a D' se, e somente se, existe um fluxo em D'' (da origem aos destinos) de valor igual ao grau de saída de r' em P' .

□

Seja D o grafo de entrada e m o vértice-mapeamento fornecidos como entrada para o PHF-O com padrão P , com $P \in \mathcal{T}$. Note que se $P <_{V,m} D$ então usando o algoritmo descrito na demonstração do Teorema 3.1.1 podemos encontrar (e não apenas decidir a existência de) um vértice-homeomorfismo de P em D .

A seguir mostraremos que para cada grafo padrão P que não está em $\mathcal{T}$ o PHF-O com padrão P é $\mathcal{NP}$ -completo. Antes, porém, provaremos alguns lemas.

Lema 3.1.2 *Se P é um subgrafo de Q e o PHF-O com padrão P é $\mathcal{NP}$ -completo então o PHF-O com padrão Q é $\mathcal{NP}$ -completo.*

Dem: Dado um grafo G juntamente com um mapeamento m dos vértices de P nos vértices de G podemos construir em tempo polinomial um grafo H e um mapeamento l dos vértices de Q nos vértices de H tais que P é m -vértice-homeomorfo a G se, e somente se, Q é l -vértice-homeomorfo a H .

Seja $Q' = Q[AQ - AP]$ e seja H o grafo obtido de G acrescentando uma cópia de Q' . No grafo H um vértice v em Q' que também está em P é identificado com o vértice $m(v)$ (veja Figura 3.4). Estenda o mapeamento m para um mapeamento l dos vértices de Q nos vértices de H da maneira natural. Se a é um arco em Q' denotaremos por a' o seu arco correspondente em H .

Claramente, se P é m -vértice-homeomorfo a G então Q é l -vértice-homeomorfo a H (basta estender, da maneira natural, um m -vértice-homeomorfismo de P em G a um l -vértice-homeomorfismo de Q em H).

Figura 3.4: Construção do grafo H . Figura 3.5: Construção do l -vértice-homeomorfismo h' de Q em H .

Para mostrar a recíproca usaremos indução no número de arcos em Q' . Se Q' é vazio então, trivialmente, P é m -vértice-homeomorfo a G . Suponha que Q' não é vazio e que Q é l -vértice-homeomorfo a H . Seja h um l -vértice-homeomorfismo de Q em H . Se todo arco de P tem imagem em G , então P é m -vértice-homeomorfo a G (h restrita a AP é um m -vértice-homeomorfismo de P em G). Suponha que o caminho em H associado a algum arco $p \in AP$, pela função h , contém um arco q' que está em H mas não está em G . Associado ao arco q existe, em H , um caminho da cauda de q' à ponta de q' (veja Figura 3.5). Logo, a imagem de p é um caminho de comprimento um cujo único arco é q' ; caso contrário, os caminhos associados a q e p não seriam disjuntos nos vértices. O mapeamento h' formado trocando os valores de $h(p)$ e $h(q)$ é um l -vértice-homeomorfismo de Q em H . Restringindo o domínio de h' a $Q - \{q\}$, nós temos que $Q - \{q\}$ é vértice-

homeomorfo a $H - \{q'\}$. Segue pela hipótese de indução que P é m -vértice-homeomorfo a G .

□

A estrutura apresentada na Figura 3.6 será denominada **switch**. O lema seguinte prova uma propriedade dos switches que será fundamental na demonstração de que o PHF-O com padrão consistindo de dois arcos disjuntos é $\mathcal{NP}$ -completo (Lema 3.1.4).

Lema 3.1.3 *Considere um switch, como indicado na Figura 3.6. Se existem dois caminhos disjuntos passando pelo switch, um que sai pelo vértice a e outro que entra pelo vértice b , então o caminho que sai por a deve entrar por c e o caminho que entra por b deve sair por d . Além disso, existe exatamente um caminho que “passa” pelo switch e é disjunto dos anteriores que é ou $(5, 6, 7, 4, 3, 8)$ ou $(5', 6', 7', 4', 3', 8')$ dependendo do caminho que entra por a .*

Dem: Suponha que existam dois caminhos disjuntos passando pelo switch, um que sai pelo vértice a e outro que entra pelo vértice b . O caminho que sai por a deve passar pelo vértice 1 ou pelo vértice 1'. Suponha que este caminho passe por 1. Este caminho deve passar também pelos vértices 2, 3 e 4. O caminho que entra por b não pode passar pelo vértice 2, pois o caminho que sai por a passa por este vértice, e então deve passar pelo vértice 2'. Este caminho deve, também, passar pelos vértices 1', 6 e 7. Como este caminho não pode passar pelo vértice 4 então este passa por 7 e sai por d . O caminho que sai por a não pode passar por 7 e então vem de c . O caminho $(5, 6, 7, 4, 3, 8)$ está bloqueado, pois um de seus vértices já foi utilizado pelos outros caminhos, e o caminho $(5', 6', 7', 4', 3', 8')$ é livre, ou seja é disjunto dos caminhos considerados. Observe que se um caminho entra por 5' então deve sair por 8' pois o vértice 2' já foi usado pelo caminho que entra por b .

Se o caminho que sai por a passa por 1' então, devido à simetria do grafo, podemos mostrar, de modo análogo ao que foi feito acima, que o caminho que sai por a deve entrar por c , o caminho que entra por b deve sair por d e neste caso o caminho $(5, 6, 7, 4, 3, 8)$ é livre.

□

Observe que se **empilharmos** vários switches identificando o arco que entra em c (o arco que sai de d) de um switch com o arco que sai de a (o arco que entra em b) do

Figura 3.6: *Exemplo de um switch.*

próximo switch, a primeira parte da afirmação do lema anterior continua válida. Este fato será usado na demonstração do Lema 3.1.4.

Um switch como o da Figura 3.6 será representado esquematicamente como indicado na Figura 3.7. As flechas verticais, que não são arcos, representam os caminhos $(5, 6, 7, 4, 3, 8)$ e $(5', 6', 7', 4', 3', 8')$ e a linha horizontal indica que no máximo um destes caminhos pode ser usado. Os caminhos que saem por a e entram por b neste switch ficam implícitos nesta representação.

Por simplicidade, durante a demonstração do próximo lema chamaremos os caminhos $(c, 4, 3, 2, 1, a)$ e $(b, 2', 1', 6, 7, d)$ de **caminhos internos diretos** (do switch), os caminhos $(c, 4', 3', 2', 1', a)$ e $(b, 2, 1, 6', 7', d)$ de **caminhos internos esquerdos**. Estes quatro caminhos serão também chamados, indistintamente, de **caminhos internos**. O cami-

Figura 3.7: Representação esquemática de um switch.

inho $(5, 6, 7, 4, 3, 8)$ será chamado **caminho adicional direito** e o $(5', 6', 7', 4', 3', 8')$ será chamado **caminho adicional esquerdo**. Estes dois caminhos serão chamados, indistintamente, de **caminhos adicionais**. Usando esta nomenclatura, o lema anterior afirma que existem dois caminhos internos disjuntos nos vértices passando pelo switch, que são ou esquerdos ou direitos, e que existe um caminho adicional passando pelo switch que é disjunto nos vértices com cada um dos caminhos internos que passa pelo switch. Além disso, tal caminho adicional é direito se os caminhos internos forem esquerdos ou é esquerdo se os caminhos internos forem direitos.

Usando a nomenclatura fixada no capítulo anterior vamos definir um caso especial do problema da satisfatibilidade, também $\mathcal{NP}$ -completo, que será usado na demonstração do próximo lema.

Problema da 3-satisfatibilidade (3-SAT)

Entrada: Uma fórmula $\mathcal{F}$ sobre um conjunto U de variáveis,
onde cada cláusula possui exatamente três literais.

Pergunta: Existe uma atribuição de valores para U que satisfaz $\mathcal{F}$?

O lema que provaremos a seguir, apesar de um tanto restrito, é a chave para a obtenção do resultado principal desta seção, que caracteriza a classe dos grafos padrões P para os quais o PHF-O com padrão P é $\mathcal{NP}$ -completo. Neste lema nos referiremos ao seguinte grafo T :

Figura 3.8: Grafo padrão T .

Lema 3.1.4 *O PHF-O com padrão T é $\mathcal{NP}$ -completo.*

Dem: É suficiente mostrar que o 3-SAT pode ser reduzido ao PHF-O com padrão T . Seja $\mathcal{F}$ a fórmula fornecida como entrada para o 3-SAT. Sejam $D_1, \dots, D_l$ as cláusulas da fórmula $\mathcal{F}$ e $x_1, \dots, x_k$ as variáveis de $\mathcal{F}$. Usaremos a fórmula $\mathcal{F}$ para construir um grafo orientado $D_{\mathcal{F}}$ e um mapeamento m dos vértices de T nos vértices de $D_{\mathcal{F}}$ tais que T é m -vértice-homeomorfo a $D_{\mathcal{F}}$ se, e somente se, $\mathcal{F}$ é satisfatível.

Inicialmente, considere $D_{\mathcal{F}}$ como sendo o grafo vazio. Para cada variável x_i de $\mathcal{F}$ acrescente a $D_{\mathcal{F}}$ um subgrafo como o indicado na Figura 3.9. Seja t o vértice do topo do subgrafo associado a x_1 . Associe uma das colunas de arcos verticais com o literal x_i e a outra coluna com o literal $\bar{x}_i$. O número de arcos em cada coluna é o número de ocorrências do literal associado a esta coluna na fórmula $\mathcal{F}$. Estes subgrafos são conectados por um arco que vai do vértice da base do subgrafo associado à variável x_i para o vértice do topo do subgrafo associado à variável x_{i+1} , para $i = 1, \dots, k-1$. Acrescente a $D_{\mathcal{F}}$ os vértices $n_0, \dots, n_l$ e três arcos paralelos de n_{i-1} para n_i , para $i = 1, \dots, l$, que correspondem aos literais que estão em D_i . Acrescente um arco do vértice da base do subgrafo associado a x_k para o vértice n_0 . Para cada literal x que está na cláusula D_i troque um dos arcos que vai de n_{i-1} a n_i e um dos arcos na coluna associada a x por um switch, identificando os vértices 8 e $8'$ do switch com os vértices que são as pontas dos arcos retirados e identificando os vértices

Figura 3.9: Subgrafo associado à variável x_i .

5 e 5' com os vértices que são as caudas dos arcos retirados. Os switches são empilhados da maneira que discutimos após o Lema 3.1.3 (numa ordem arbitrária). Acrescente a $D_{\mathcal{F}}$ os vértices u' , v' , r' e s' . Faça o arco que entra no vértice b do primeiro switch (da pilha) sair de r' , o arco que sai do vértice a do primeiro switch entrar em v' , o arco que entra em c no último switch sair de u' e o arco que sai de d do último switch (da pilha) entrar em t . Finalmente, acrescente um arco de n_l para s' completando assim a construção de $D_{\mathcal{F}}$. Um exemplo esquemático de um grafo $D_{\mathcal{F}}$ é mostrado na Figura 3.10.

Seja m o mapeamento dos vértices de T nos vértices de $D_{\mathcal{F}}$ definido por $m(u) = u'$, $m(v) = v'$, $m(r) = r'$, $m(s) = s'$.

Vamos mostrar que em $D_{\mathcal{F}}$ existem um caminho de r' a s' e um caminho de u' a v' , disjuntos nos vértices, se e somente se a fórmula $\mathcal{F}$ é satisfatível.

Suponha que a fórmula $\mathcal{F}$ é satisfatível e fixe uma atribuição de valores às variáveis $x_1, \dots, x_k$ que satisfaz $\mathcal{F}$. Observando esta atribuição de valores construiremos um caminho de u' a v' e um caminho de r' a s' que são disjuntos nos vértices. O caminho de u' a v' e a seção do caminho de r' a s' que vai de r' a t são obtidos concatenando-se caminhos internos (um em cada switch da pilha). Ao passar por um switch associado ao literal x estes caminhos serão os caminhos internos direitos se, e somente se, x for verdadeiro. A seção do caminho de r' a s' que vai de t a n_0 consiste de uma concatenação de caminhos adicionais de alguns switches. Os caminhos adicionais direitos nos switches associados a literais verdadeiros estão bloqueados pois os caminhos internos direitos de cada um destes switches já foram usados. Deste modo, o caminho de r' a s' passará, agora, pelo caminho

adicional direito de cada um dos switches associados a literais falsos. Como no mínimo um literal em cada cláusula é verdadeiro, e nos switches associados a literais verdadeiros foram usados apenas os caminhos internos direitos, então pelo menos um dos caminhos adicionais esquerdos de n_i a n_{i+1} está livre. Portanto, existe um caminho de r' a s' e um caminho de u' a v' em $D_{\mathcal{F}}$ que são disjuntos nos vértices.

Suponha, agora, que exista um caminho de u' a v' e um caminho de r' a s' que são disjuntos nos vértices. Fixe um tal par de caminhos. Suponha que a seção do caminho de r' a s' compreendida entre os vértices n_i a n_{i+1} seja o caminho adicional esquerdo de um switch associado a um literal x e a seção deste caminho compreendida entre os vértices n_j e n_{j+1} ($i \neq j$) seja o caminho adicional esquerdo de um switch associado a $\bar{x}$. O caminho de u' a v' e a seção do caminho de r' a s' compreendida entre os vértices r' e t passa pelos caminhos internos de cada um destes switches. Nestes switches os caminhos internos usados são os direitos, ficando, desta forma, bloqueados os caminhos adicionais direitos deste switches. Mas isto é um absurdo, pois o caminho de r' a s' ou passa pelos caminhos adicionais direitos de todos os switches associados ao literal x ou passa pelos caminhos adicionais direitos de todos os switches associados ao literal $\bar{x}$.

Podemos então fazer a seguinte atribuição de valores às variáveis de $\mathcal{F}$: x_i é verdadeira se, e somente se, o caminho de r' a s' passa pelo caminho adicional esquerdo de algum switch associado ao literal x_i . Claramente, esta atribuição de valores satisfaz a fórmula $\mathcal{F}$.

□

Teorema 3.1.5 *Para cada grafo orientado P que não está em $\mathcal{T}$ o PHF-O com padrão P é $\mathcal{NP}$ -completo.*

Dem: É fácil ver que um grafo orientado P não está em $\mathcal{T}$ se, e somente se, P contém um dos seguintes subgrafos (veja Figura 3.11):

- (a) um grafo gerado por dois arcos disjuntos (que podem ser laços),
- (b) um caminho de comprimento dois, ou
- (c) um circuito de comprimento dois.

Figura 3.10: Exemplo esquemático de um grafo $D_{\mathcal{F}}$.

Figura 3.11: Subgrafos minimais que não estão em $\mathcal{T}$.

O Lema 3.1.4 prova que o PHF-O com grafo padrão (a), onde os arcos não são laços, é $\mathcal{NP}$ -completo. Os demais casos são provados analogamente, mudando apenas a construção do grafo $D_{\mathcal{F}}$. No caso (a), quando há laços identificamos no grafo $D_{\mathcal{F}}$ construído no Lema 3.1.4, os vértices u' com v' e/ou r' com s' . No caso (b) identificamos v' com r' , e no caso (c) identificamos u' com s' e v' com r' . Para concluir a prova do teorema basta observar que se P é um grafo que não está em $\mathcal{T}$ então P contém um dos subgrafos descritos acima e portanto, pelo Lema 3.1.2, o PHF-O com padrão P é $\mathcal{NP}$ -completo. $\square$

Os Teoremas 3.1.1 e 3.1.5 fornecem uma caracterização simples dos grafos padrões P para os quais o PHF-O com padrão P é fácil. Em linhas gerais, podemos afirmar que para a maioria dos grafos padrões P o PHF-O com padrão P é $\mathcal{NP}$ -completo. Isto fornece motivação para estudar o PHF-O com padrão fixo com entrada restrita a algumas classes de grafos, objetivando verificar se o problema se torna fácil ou continua $\mathcal{NP}$ -completo.

3.1.2 PHF-O com padrão fixo e entrada específica

Nesta parte estudaremos o PHF-O com padrão fixo e entrada restrita a certas classes de grafos. Analisaremos o problema para as seguintes classes de grafos orientados: (a) grafos orientados com grau no máximo três, (b) grafos bipartidos, (c) grafos orientados transitivos e (d) grafos orientados acíclicos.

(a) *Grafos orientados com grau no máximo três.*

Fortune, Hopcroft e Wyllie [7] observaram que o PHF-O com padrão P e entrada restrita a grafos com grau no máximo três possui a mesma complexidade que o problema sem esta restrição, qualquer que seja o grafo padrão P . Veremos, a seguir, este resultado.

O grafo $D_{\mathcal{F}}$ construído no Lema 3.1.4 pode ser modificado de maneira a obtermos um novo grafo $D'_{\mathcal{F}}$ em que o grau é no máximo três. Observe que os vértices que estão nos switches (veja Figura 3.6) possuem grau no máximo três. Desta forma, os únicos vértices do grafo $D_{\mathcal{F}}$ que possuem grau maior do que três são os vértices $n_0, n_1, \dots, n_l$. Na Figura 3.12 indicamos como deve ser feita a modificação no grafo $D_{\mathcal{F}}$ de modo a obter um grafo com grau no máximo três.

Seja P o grafo que consiste de dois arcos disjuntos. Substituindo na demonstração do Lema 3.1.4 o grafo $D_{\mathcal{F}}$ pelo grafo modificado $D'_{\mathcal{F}}$ podemos provar que o PHF-O com padrão P e entrada restrita a grafos com grau no máximo três é $\mathcal{NP}$ -completo. Construções análogas às definidas na prova do Teorema 3.1.5 garantem que para cada grafo P que não é um in-tree ou um out-tree o PHF-O com padrão P e entrada restrita a grafos com grau no máximo três é $\mathcal{NP}$ -completo. Desta forma, a caracterização dos grafos padrões P para os quais o PHF-O com padrão P e entrada restrita a grafos com grau no máximo três tem algoritmo polinomial é a mesma que vimos para o problema sem restrição na entrada.

(b) *Grafos bipartidos.*

O grafo $D_{\mathcal{F}}$ construído no Lema 3.1.4 pode ser modificado de maneira a obtermos um novo grafo $D'_{\mathcal{F}}$ que é bipartido. Basta inserir um vértice em cada arco $e = uv$ em $D_{\mathcal{F}}$. Por uma argumentação análoga à feita para o item (a) concluímos que a caracterização dos grafos P para os quais o PHF-O com padrão P e entrada restrita a grafos bipartidos é a mesma que vimos para o problema sem restrição na entrada.

(c) *Grafos orientados transitivos.*

Veremos que o PHF-O com padrão fixo com entrada restrita a grafos orientados transitivos é polinomial para qualquer grafo padrão. Seja P um grafo orientado. Seja D o grafo orientado transitivo fornecido na entrada do PHF-O com padrão P e seja m o mapeamento dos vértices de P nos vértices de D , fornecido juntamente com D . É fácil decidir se P é vértice-homeomorfo a D . Basta proceder da seguinte maneira: para cada

Figura 3.12: Grafo $D_{\mathcal{F}}$ modificado com grau no máximo três.

arco $e = uv \in AP$ verifique se o arco $(m(u), m(v))$ está em D . Claramente, P é vértice-homeomorfo a D se, e somente se, existir um tal arco para cada arco de P (note que como D é transitivo, existe um caminho de $m(u)$ a $m(v)$ se, e somente se, o arco $(m(u), m(v))$ está em D).

(d) *Grafos orientados acíclicos.*

Seja P o grafo orientado que consiste de dois arcos disjuntos e os quatro vértices em que estes arcos incidem. Perl e Shiloach [16] apresentaram um algoritmo polinomial para resolver o PHF-O com padrão P e entrada restrita a grafos orientados acíclicos. Ou seja, apresentaram um algoritmo para encontrar em um grafo acíclico D um caminho de s_1 a t_1 e um caminho de s_2 a t_2 disjuntos nos vértices, onde s_1, s_2, t_1 e t_2 são vértices fixos de D . Fortune, Hopcroft e Wyllie [7] observaram que este algoritmo podia ser generalizado para resolver o PHF-O com um padrão fixo qualquer e apresentaram esta generalização em termos de um jogo de pedrinhas. Perl e Shiloach construíam a partir de D um novo grafo orientado e procuravam um caminho entre dois vértices específicos deste novo grafo. O uso do jogo de pedrinhas é muito interessante, principalmente por deixar clara a idéia existente por trás do algoritmo. Por este motivo decidimos apresentar o algoritmo para P arbitrário e D acíclico usando o jogo de pedrinhas.

Fixe um grafo padrão P . Seja D o grafo orientado acíclico e m o vértice-mapeamento fornecidos como entrada para o PHF-O com padrão P .

Definimos o nível de um vértice v no grafo de entrada D denotado por $l(v)$, como sendo o comprimento de um caminho mais longo que começa em v . Claramente, se existe um caminho de v a w , ($v \neq w$) então $l(v) > l(w)$.

No jogo, as pedrinhas indicarão a escolha dos arcos do grafo de entrada. O jogo será feito movendo as pedrinhas sobre o grafo D . Se P for vértice-homeomorfo a D então o caminho percorrido pela pedrinha p_i , associada ao arco $a_i = uv$, será o caminho em D associado ao arco a .

As regras do jogo de pedrinhas são as seguintes:

1. Para cada arco a_i no grafo padrão existe no jogo uma pedrinha p_i . No início do jogo, para cada vértice s do grafo padrão P , as pedrinhas associadas aos arcos saindo de

s serão colocadas no vértice $m(s)$ em D .

2. Em qualquer momento uma pedrinha p_i ($a_i \in AP$), que está em um vértice u , pode ser movida para um vértice v , onde $uv \in AD$, se:
 - (a) u tem o maior nível entre os vértices que estão com pedrinhas (se duas ou mais pedrinhas estão em vértices que têm nível máximo, qualquer uma das pedrinhas pode ser movida).
 - (b) não existe pedrinha em v ou existe pedrinha em v e v é a imagem, por m , da ponta de a_i e da cauda de algum outro arco de P .
 - (c) v não é imagem por m de nenhum vértice do grafo padrão, exceto, possivelmente da ponta de a_i .
3. A pedrinha p_i pode ser retirada do grafo quando é colocada na imagem da ponta de a_i .
4. O jogador vence se todas as pedrinhas forem retiradas do grafo de entrada.

Lema 3.1.6 *O jogo de pedrinhas pode ser vencido se, e somente se, o grafo padrão é vértice-homeomorfo ao grafo de entrada.*

Dem: Suponha, inicialmente, que o jogador possui uma estratégia para vencer o jogo. Claramente, o caminho percorrido pela pedrinha p_i no grafo de entrada é um caminho da imagem da cauda de a_i à imagem da ponta de a_i . Vamos mostrar que estes caminhos são disjuntos nos vértices. Suponha que os caminhos percorridos pelas pedrinhas p_i e p_j se encontram em um vértice w e que as pontas de a_i e a_j não sejam coincidentes. Como a pedrinha p_i é colocada no vértice w então sabemos pela regra (2c) que w não é imagem de nenhum vértice do grafo padrão, exceto, possivelmente, a ponta de a_i . Mas, a pedrinha p_j também é colocada no vértice w . E então w não pode ser imagem de nenhum vértice do grafo padrão, exceto, possivelmente, a ponta de a_j . Logo, w não é imagem de nenhum vértice do grafo padrão. Podemos assumir, sem perda de generalidade, que p_j visita w primeiro. Pela regra (2b), a pedrinha p_j deve sair de w antes de p_i chegar. Mas isto contradiz (2a), pois o nível do vértice em que a pedrinha p_i está é maior do que o nível do vértice em que a pedrinha p_j está, pois existe um caminho do vértice em que está p_i ao vértice em que está p_j . Logo, os caminhos percorridos pelas pedrinhas no jogo são disjuntos nos vértices. Portanto, o grafo padrão é vértice-homeomorfo ao grafo de entrada.

Suponha que P é vértice-homeomorfo a D . Fixe um tal vértice-homeomorfismo. É fácil ver que o jogador vence o jogo se adotar a seguinte estratégia: sempre que for movimentar uma pedrinha p_i deslocá-la ao longo de arcos que estejam no caminho associado (pelo vértice-homeomorfismo fixado) ao arco a_i .

□

Chamaremos **configuração** a qualquer disposição possível das pedrinhas no grafo em um dado instante do jogo.

Lema 3.1.7 *A partir do grafo padrão P e do grafo de entrada D , pode ser construído em tempo polinomial no tamanho de D , um grafo D' cujos vértices correspondem às configurações e cujos arcos correspondem aos movimentos possíveis no jogo.*

Dem: Seja $\hat{D}$ o grafo obtido a partir de D substituindo cada vértice v , que é imagem de um vértice do grafo padrão, por $gr_D(v)$ novos vértices e substituindo cada arco que entra (sai) de v por um arco que entra (sai) de um desses vértices novos.

Seja $k = |AP|$ e seja D' o grafo orientado simples com

$$VD' = \{\langle u_1, \dots, u_k \rangle : u_i \in V\hat{D}, u_i \neq u_j, i = 1, \dots, k\} \text{ e}$$

$$AD' = \{(\langle u_1, \dots, u_i, \dots, u_k \rangle, \langle u_1, \dots, u_{i'}, \dots, u_k \rangle) : \\ (u_i, u_{i'}) \in AD \text{ e } l(u_i) \geq l(u_{i'}), j = 1, \dots, k\}.$$

Note que uma k -upla de vértices indica em que vértice está cada uma das k pedrinhas. Note que D' tem $O(|VD|^k)$ vértices.

É fácil ver que D' pode ser construído em tempo $O(|AD|)$.

□

Teorema 3.1.8 *Qualquer que seja o grafo padrão P existe um algoritmo polinomial para resolver o PHF-O com padrão P com entrada restrita a grafos orientados acíclicos.*

Dem: Seja P o grafo padrão dado e seja D o grafo fornecido na entrada do PHF-O com padrão P . Construa o grafo D' citado no lema anterior. No grafo D' os vértices correspondem às configurações do jogo de pedrinhas. Um único vértice de D' está associado

à configuração inicial do jogo e um único vértice está associado à configuração final no caso em que o jogo é vencido. Execute um algoritmo para procurar em D' um caminho do vértice correspondente à configuração inicial ao vértice correspondente à configuração final. Claramente, existe um tal caminho em D' se, e somente se, o jogo de pedrinhas pode ser vencido. Segue pelo Lema 3.1.6 que P é vértice-homeomorfo a D se, e somente se, existir um tal caminho. Se P for vértice-homeomorfo a D , usando o caminho devolvido pelo algoritmo podemos determinar um vértice-homeomorfismo de P em D . Como $|AD'| \leq k(|V\hat{D}| - 2)^{k-1}|AD|$, onde k é o número de arcos de P , e um caminho em D' pode ser encontrado por um algoritmo $O(|AD'|)$ então o algoritmo para o PHF-O com padrão P com entrada restrita a grafos acíclicos é polinomial (já que k é fixo).

□

Vimos que para diversos padrões P o PHF-O com padrão P é $\mathcal{NP}$ -completo. Alguns destes problemas podem ser utilizados para mostrar que certos problemas envolvendo modularidade de caminhos e circuitos em grafos orientados são $\mathcal{NP}$ -completos. Estudaremos alguns destes problemas no Capítulo 5.

3.1.3 PAHF-O com padrão fixo

Vimos na Seção 3.1.1 que existe uma caracterização simples dos grafos padrões P para os quais o PHF-O com padrão P é fácil. Entretanto, não se conhece uma tal caracterização para o PAHF-O com padrão fixo. Segue diretamente da demonstração do Teorema 1.4.2 que se P é um grafo para o qual existe um algoritmo polinomial para o PHF-O com padrão P então existe também um algoritmo polinomial para o PAHF-O com padrão P . Observe que na demonstração do Teorema 1.4.1 o grafo padrão é modificado. Assim, este resultado não garante que se o PHF-O com padrão P é $\mathcal{NP}$ -completo então o PAHF-O com padrão P também é $\mathcal{NP}$ -completo.

3.2 Mapeamento fixo: caso não-orientado

Discutiremos nesta seção os resultados existentes a respeito da versão não-orientada dos problemas de homeomorfismo com padrão e mapeamento fixos.

LaPaugh e Rivest [12] conseguiram, em 1980, um algoritmo linear para o seguinte problema:

Problema do triângulo:

Entrada: Um grafo G não-orientado e três vértices distintos v_1, v_2 e v_3 em G .

Pergunta: Existe um circuito em G que passa por v_1, v_2 e v_3 ?

O algoritmo de LaPaugh e Rivest é baseado em análise exaustiva de casos e não utiliza nenhuma técnica muito interessante.

Em 1980, Shiloach [20] obteve um algoritmo de complexidade $O(|VG| \cdot |AG|)$ para o seguinte problema, no caso em que $k = 2$:

Problema dos k -caminhos:

Entrada: Um grafo G não-orientado e k pares de vértices $s_1, t_1, \dots, s_k, t_k$ em G .

Pergunta: Existem caminhos $P_1, \dots, P_k$ disjuntos, onde $P_i, i = 1, \dots, k$,
é um caminho de s_i a t_i em G ?

O algoritmo desenvolvido por Shiloach faz uso de vários outros algoritmos. Um destes algoritmos é o de Perl e Shiloach [16], que resolve o problema com a entrada restrita a grafos planares. O algoritmo é bastante extenso e não utiliza nenhuma técnica muito especial.

Até há pouco tempo atrás estes eram os únicos problemas de homeomorfismo com padrão e mapeamentos fixos para grafos não-orientados que sabíamos serem polinomialmente solúveis. Não se conhecia nenhum padrão para o qual o problema fosse $\mathcal{NP}$ -completo. Recentemente, Robertson e Seymour [18] provaram que **para qualquer padrão P , existe um algoritmo polinomial para o PHF com padrão P , e conseqüentemente também para o PAHF com padrão P** (veja Teorema 1.4.3). Estes resultados são de grande interesse teórico pois, como comentamos acima, pouco se sabia a respeito destes problemas. Porém, o algoritmo de Robertson e Seymour não apresenta muito interesse na prática, pois as constantes envolvidas crescem muito rapidamente com o número de vértices do grafo padrão. O artigo em que este algoritmo é apresentado é bastante extenso (aproximadamente 50 páginas) e utiliza um lema na prova da corretude do algoritmo cuja demonstração não está contida no artigo por ser muito longa (mais de 300 páginas).

Pelos motivos expostos ao longo desta seção não incluiremos neste trabalho os algoritmos citados. O leitor interessado pode recorrer diretamente aos artigos originais.

3.3 Mapeamento livre

No caso de problemas de homeomorfismo com padrão fixo, na versão orientada ou não-orientada, os problemas com vértice-mapeamento livre podem ser polinomialmente reduzidos a problemas com vértice-mapeamento fixo. Basta considerar todos os vértice-mapeamentos possíveis. Note que esta quantidade é polinomial pois P não faz parte da entrada do problema. Assim, lembrando o resultado devido a Robertson e Seymour que mencionamos na seção anterior, temos que tanto o PH como o PAH com padrão fixo podem ser resolvidos em tempo polinomial. Vale aqui ressaltar que para o caso em que o padrão é o $K_{3,3}$ Asano [3] provou que existe um algoritmo linear para o problema. Quanto à redução recíproca (isto é, de mapeamento livre para mapeamento fixo), não se pode fazer uma afirmação análoga, já que em geral, o padrão é modificado. Isto é o que ocorre, por exemplo, na demonstração do Teorema 2.1.3. Logo, demonstrações análogas à feita para este teorema não provariam que problemas de homeomorfismo com padrão P e vértice-mapeamento fixo podem ser reduzidos a problemas de homeomorfismo com (o mesmo) padrão P e vértice-mapeamento livre.

Capítulo 4

Homeomorfismo em grafos planares

Vimos nos capítulos anteriores que o PHF-O com padrão fixo é $\mathcal{NP}$ -completo para a maioria dos grafos padrões, e que o PHF com um padrão fixo qualquer possui algoritmo de complexidade polinomial mas bastante sofisticado. Isto nos motivou a estudar estes problemas com entrada restrita a grafos planares. Neste capítulo denotaremos por n o número de vértices do grafo de entrada.

Na primeira seção estudaremos o PHF-O com um padrão fixo qualquer. Veremos um algoritmo polinomial para este caso que foi obtido recentemente por Schrijver [19]. Trata-se de um algoritmo bastante engenhoso que, apesar de polinomial, não possui grande interesse de ordem prática, devido a dificuldades de implementação.

Na segunda seção estudaremos o PHF com padrão fixo P , onde P é o grafo que consiste de dois vértices e k arestas múltiplas. Apresentaremos dois algoritmos: o primeiro, devido a Suzuki, Akama e Nishizeki [21], utiliza divisão-e-conquista e possui complexidade $O(n \log n)$, e o segundo, proposto por Ripphausen-Lipa, Wagner e Weihe [17], é um algoritmo linear que faz uso de busca em profundidade. Ambos podem ser eficientemente implementados.

4.1 PHF-O com padrão fixo genérico

Nesta seção estudaremos o PHF-O com padrão fixo P e entrada restrita a grafos planares, onde P é um grafo orientado com k arcos. Este problema pode ser enunciado da seguinte forma:

Problema dos k caminhos disjuntos - $\text{PkC}(D, r_1, s_1, \dots, r_k, s_k)$

Entrada: Um grafo orientado planar D , imerso no plano, e k pares de vértices.

Pergunta: Decidir se em D existem k caminhos disjuntos nos vértices $(r_1, s_1), \dots, (r_k, s_k)$
 $P_1, \dots, P_k$, onde P_i é um caminho de r_i a s_i , para $i = 1, \dots, k$.

Note que não há nenhuma restrição quanto aos vértices r_i e s_j serem distintos.

O problema será estudado considerando o grafo D conexo. Em 1992, A. Schrijver [19] obteve um algoritmo polinomial para o PkC . Os resultados que apresentaremos nesta seção são baseados neste seu trabalho. Vale observar que se não consideramos o parâmetro k fixo o problema é $\mathcal{NP}$ -completo (veja [14]).

4.1.1 Preliminares e notação

No que segue enunciaremos alguns resultados básicos que utilizaremos para resolver o PkC . Aproveitaremos, também, para fixar parte da notação que será utilizada.

O algoritmo que apresentaremos para o PkC faz uso de conceitos da teoria dos grupos para expressar mais facilmente as posições relativas de certos caminhos nos grafos com que trabalharemos. Apesar de serem bastante conhecidas as definições em teoria dos grupos que apresentaremos agora, elas serão úteis para fixarmos algumas das notações que usaremos nesta seção.

Seja $\mathcal{G}$ um conjunto e $*$: $\mathcal{G} \times \mathcal{G} \rightarrow \mathcal{G}$ uma operação binária definida em $\mathcal{G}$. Dizemos que $(\mathcal{G}, *)$ é um **grupo** se os seguintes axiomas são satisfeitos:

- (Propriedade associativa) Para quaisquer $x, y, z \in \mathcal{G}$ tem-se $(x * y) * z = x * (y * z)$.
- (Existência do elemento neutro) Existe em $\mathcal{G}$ um elemento neutro 1 tal que $x * 1 = x = 1 * x$, para todo $x \in \mathcal{G}$.
- (Existência do inverso) Para qualquer $x \in \mathcal{G}$, existe um elemento em $\mathcal{G}$, que denotaremos por x^{-1} , tal que $x * x^{-1} = 1 = x^{-1} * x$.

Se qualquer elemento de $\mathcal{G}$ puder ser obtido através de operações $*$ sobre $x_1, x_1^{-1}, \dots, x_k, x_k^{-1}$ diremos que $(\mathcal{G}, *)$ é **gerado** por $x_1, \dots, x_k$.

Denotaremos por G_k o grupo $(G_k, \cdot)$ gerado por $g_1, \dots, g_k$ que consiste de todas as palavras $b_1 \dots b_t$, onde $t \geq 0$ e $b_1, \dots, b_t \in \{g_1, g_1^{-1}, \dots, g_k, g_k^{-1}\}$ e tais que $b_i b_{i+1} \neq g_j g_j^{-1}$ e $b_i b_{i+1} \neq g_j^{-1} g_j$, para $i = 1, \dots, t-1$ e $j = 1, \dots, k$. O produto $x \cdot y$, onde x e y são palavras de G_k , é obtido concatenando-se x e y (xy) e retirando-se iterativamente todas as ocorrências de $g_j g_j^{-1}$ e $g_j^{-1} g_j$, para $j = 1, \dots, k$. Note que, geralmente, $x \cdot y \neq xy$. É fácil ver que o elemento neutro de $(G_k, \cdot)$ é a palavra vazia. Chamaremos $g_1, g_1^{-1}, \dots, g_k, g_k^{-1}$ de **símbolos**. O **tamanho** de uma palavra x , denotado por $|x|$, é o número de símbolos em x , contando as suas multiplicidades.

Sejam $y, w \in G_k$. Diremos que y é um **segmento** de w se $w = xyz$ para certos $x, z \in G_k$. Se $w = yz$, para algum $z \in G_k$, então diremos que y é um **segmento inicial** de w , e denotaremos $y \preceq w$. A relação binária $\preceq$ determina uma ordem parcial sobre G_k . Estendendo G_k a $G_k \cup \{\infty\}$ temos um reticulado. Neste reticulado, indicaremos a operação **inf** por $\wedge$ e a operação **sup** por $\vee$. Se $x \preceq y$ temos que $x \wedge y = x$ e $x \vee y = y$. Se x e y não são comparáveis então $x \wedge y = 1$ e $x \vee y = \infty$. Um subconjunto Γ de G_k é **hereditário** se para cada palavra $y \in \Gamma$, cada segmento de y pertence a Γ .

Seja D um grafo orientado conexo e r um vértice de D . Sejam ϕ e ψ duas funções de AD em G_k . Dizemos que ψ é **r -cohomóloga** a ϕ se existe uma função $f : VD \rightarrow G_k$ tal que

$$(a) f(r) = 1;$$

$$(b) \psi(a) = f(u)^{-1} \cdot \phi(a) \cdot f(w), \text{ para cada arco } a = uw \in AD.$$

Proposição 4.1.1 *Seja $\mathcal{F} = \{\phi : \phi \text{ é uma função de } AD \text{ em } G_k\}$. A relação $\mathcal{R} = \{(\phi_1, \phi_2) \in \mathcal{F} \times \mathcal{F} : \phi_1 \text{ é } r\text{-cohomóloga a } \phi_2\}$ é de equivalência.*

Dem: Seja $\phi_1 \in \mathcal{F}$. É imediato que ϕ_1 é r -cohomóloga a ϕ_1 (basta tomar $f : VD \rightarrow G_k$ definida por $f(v) = 1$, para cada $v \in VD$). Logo, $\mathcal{R}$ é reflexiva.

Sejam $\phi_1, \phi_2 \in \mathcal{F}$. Suponha que $(\phi_1, \phi_2) \in \mathcal{R}$. Neste caso, existe $f : VD \rightarrow G_k$ tal que $\phi_1(a) = f(u)^{-1} \cdot \phi_2(a) \cdot f(w)$, para cada $a = uw \in AD$. Seja $g : VD \rightarrow G_k$ definida por $g(v) = f(v)^{-1}$, para cada $v \in VD$. Assim, $\phi_2(a) = f(u) \cdot \phi_1(a) \cdot f(w)^{-1} = g(u)^{-1} \cdot \phi_2(a) \cdot g(w)$. Além disso, $g(r) = 1$. Logo, $(\phi_2, \phi_1) \in \mathcal{R}$ e, portanto, $\mathcal{R}$ é simétrica.

Sejam $\phi_1, \phi_2, \phi_3 \in \mathcal{F}$. Suponha que (ϕ_1, ϕ_2) e (ϕ_2, ϕ_3) pertencem a $\mathcal{R}$. Neste caso, existem funções $f, g : VD \rightarrow G_k$ tais que

$$(a) \phi_1(a) = f(u)^{-1} \cdot \phi_2(a) \cdot f(w), \text{ para cada } a = uw \in AD,$$

$$(b) \phi_2(a) = g(u)^{-1} \cdot \phi_3(a) \cdot g(w), \text{ para cada } a = uw \in AD \text{ e}$$

$$(c) f(r) = g(r) = 1.$$

Seja $h : VD \rightarrow G_k$ definida por $h(v) = g(v) \cdot f(v)$, para cada vértice $v \in VD$. De (a) e (b) segue que $\phi_1(a) = f(u)^{-1} \cdot g(u)^{-1} \cdot \phi_3(a) \cdot g(w) \cdot f(w)$, para cada $a = uw \in AD$. Ou seja, $\phi_1(a) = h(u)^{-1} \cdot \phi_3(a) \cdot h(w)$, para cada $a = uw \in AD$. Além disso, $h(r) = 1$. Logo, $(\phi_1, \phi_3) \in \mathcal{R}$ e, portanto, $\mathcal{R}$ é transitiva. Isto completa a demonstração de que $\mathcal{R}$ é uma relação de equivalência.

□

A proposição seguinte nos fornece uma interpretação para cohomologia de funções.

Proposição 4.1.2 *Sejam $\phi, \psi : AD \rightarrow G_k$ duas funções r -cohomólogas. Seja $f : VD \rightarrow G_k$ uma função que satisfaz as condições (a) e (b) da definição de r -cohomologia. Se $C = (s = v_0, e_1, v_1, \dots, v_{q-1}, e_q, v_q = s)$ é um passeio fechado não-orientado em D então $\psi(e_1)^{\varepsilon_1} \dots \psi(e_q)^{\varepsilon_q} = f(s)^{-1} \cdot \phi(e_1)^{\varepsilon_1} \dots \phi(e_q)^{\varepsilon_q} \cdot f(s)$, onde $\varepsilon_i = 1$ se C passa pelo arco e_i no sentido correto, e $\varepsilon_i = -1$ caso contrário. No caso particular em que $s = r$ temos $\psi(e_1)^{\varepsilon_1} \dots \psi(e_q)^{\varepsilon_q} = \phi(e_1)^{\varepsilon_1} \dots \phi(e_q)^{\varepsilon_q}$.*

Dem: Considere um arco e_i que está no passeio fechado não-orientado C , $1 \leq i \leq q$. Seja u a cauda de e_i e w a ponta de e_i . Observe que:

- Se $\varepsilon_i = 1$ então $\psi(e_i)^{\varepsilon_i} = [f(u)^{-1} \cdot \phi(e_i) \cdot f(w)]^{\varepsilon_i} = f(u)^{-1} \cdot \phi(e_i)^{\varepsilon_i} \cdot f(w) = f(v_{i-1})^{-1} \cdot \phi(e_i)^{\varepsilon_i} \cdot f(v_i)$.
- Se $\varepsilon_i = -1$ então $\psi(e_i)^{\varepsilon_i} = [f(u)^{-1} \cdot \phi(e_i) \cdot f(w)]^{\varepsilon_i} = f(w)^{-1} \cdot \phi(e_i)^{\varepsilon_i} \cdot f(u) = f(v_{i-1})^{-1} \cdot \phi(e_i)^{\varepsilon_i} \cdot f(v_i)$.

Assim, $\psi(e_1)^{\varepsilon_1} \dots \psi(e_q)^{\varepsilon_q} = f(v_0)^{-1} \cdot \phi(e_1)^{\varepsilon_1} \cdot f(v_1) \cdot f(v_1)^{-1} \cdot \phi(e_2)^{\varepsilon_2} \cdot f(v_2) \cdot f(v_2)^{-1} \cdot \phi(e_3)^{\varepsilon_3} \cdot f(v_3) \dots f(v_{q-1})^{-1} \cdot \phi(e_q)^{\varepsilon_q} \cdot f(v_q) = f(s)^{-1} \cdot \phi(e_1)^{\varepsilon_1} \cdot 1 \cdot \phi(e_2)^{\varepsilon_2} \cdot 1 \cdot \phi(e_3)^{\varepsilon_3} \cdot 1 \dots \phi(e_q)^{\varepsilon_q} \cdot f(s) = f(s)^{-1} \cdot \phi(e_1)^{\varepsilon_1} \cdot \phi(e_2)^{\varepsilon_2} \cdot \phi(e_3)^{\varepsilon_3} \dots \phi(e_q)^{\varepsilon_q} \cdot f(s)$. No caso particular em que $s = r$ temos $\psi(e_1)^{\varepsilon_1} \dots \psi(e_q)^{\varepsilon_q} = \phi(e_1)^{\varepsilon_1} \dots \phi(e_q)^{\varepsilon_q}$, pois $f(s) = 1$. □

Considere agora o seguinte problema sobre funções r -cohomólogas.

Problema da Viabilidade Cohomológica – PVC(D, r, ϕ, Γ)

Entrada: Um grafo orientado conexo D , um vértice $r \in VD$, uma função

$\phi : AD \rightarrow G_k$ e para cada arco $a \in AD$ um subconjunto hereditário

$\Gamma(a)$ de G_k .

Pergunta: Existe alguma função $\psi : AD \rightarrow G_k$ tal que ψ é r -cohomóloga a

ϕ e $\psi(a) \in \Gamma(a)$, para cada arco $a \in AD$? Em caso afirmativo, encontrar uma tal função.

Mais tarde, apresentaremos um algoritmo polinomial para resolver o PVC. Este algoritmo é relativamente complicado e sua inclusão aqui poderá causar perda de continuidade. Por ora, assumiremos que existe um tal algoritmo e o usaremos na construção de um algoritmo para o PkC.

4.1.2 Funções homólogas

Nesta parte vamos supor que D seja um grafo orientado imerso no plano e R seja uma das faces de D . Lembramos também que FD denota o conjunto das faces de D , e que se $F \in FD$ então F^* é o vértice no dual D^* que corresponde a F .

Sejam ϕ e ψ duas funções de AD em G_k . Dizemos que ψ é **R-homóloga** a ϕ se existe uma função $f : FD \rightarrow G_k$ tal que:

- (a) $f(R) = 1$;
- (b) $\psi(a) = f(F_E)^{-1} \cdot \phi(a) \cdot f(F_D)$, para cada arco $a \in AD$, onde F_E e F_D são as faces de D que ficam à esquerda e à direita de a , respectivamente.

Note que para grafos planares as funções homólogas e cohomólogas estão relacionadas por dualidade. Seja D^* o grafo dual de D . Para qualquer função $\gamma : AD \rightarrow G_k$ definimos $\gamma^* : AD^* \rightarrow G_k$ por $\gamma^*(a^*) = \gamma(a)$, para cada arco $a \in AD$. Sejam ϕ , ψ duas funções definidas de AD em G_k . Claramente, ϕ e ψ são R -homólogas (em D) se, e somente se, ϕ^* e ψ^* são R^* -cohomólogas (em D^*).

Sejam ϕ , $\psi : AD \rightarrow G_k$ duas funções R -homólogas. Como observamos acima, ϕ^* e ψ^* são R^* -cohomólogas. Seja $f : VD^* \rightarrow G_k$ tal que:

- (a) $f(R^*) = 1$;
- (b) $\psi^*(a^*) = f(F_E^*)^{-1} \cdot \phi^*(a^*) \cdot f(F_D^*)$, para todo $a^* \in AD^*$.

Seja $C^* = (S^* = F_0^*, a_1^*, F_1^*, \dots, F_{q-1}^*, a_q^*, F_q^* = S^*)$ um passeio fechado não-orientado em D^* . Segue pela Proposição 4.1.2 que

$$\psi^*(a_1^*)^{\varepsilon_1} \dots \psi^*(a_q^*)^{\varepsilon_q} = f(S^*)^{-1} \cdot \phi^*(a_1^*)^{\varepsilon_1} \dots \phi^*(a_q^*)^{\varepsilon_q} \cdot f(S^*),$$

onde $\varepsilon_i = 1$ se C^* passa por a_i^* no sentido correto, e $\varepsilon_i = -1$ caso contrário.

Seja $g : FD \rightarrow G_k$ a função definida da seguinte maneira: $g(F) = f(F^*)$, para todo $F \in FD$.

Temos que $\psi(a_1)^{\varepsilon_1} \dots \psi(a_q)^{\varepsilon_q} = g(S)^{-1} \cdot \phi(a_1)^{\varepsilon_1} \dots \phi(a_q)^{\varepsilon_q} \cdot g(S)$. No caso particular em

que $S = R$ temos $\psi(a_1)^{\varepsilon_1} \dots \psi(a_q)^{\varepsilon_q} = \phi(a_1)^{\varepsilon_1} \dots \phi(a_q)^{\varepsilon_q}$. Estas igualdades nos fornecem uma interpretação para homologia de funções.

Finalizamos esta seção observando que o conceito de funções homólogas só se aplica para grafos orientados imersos no plano. Já o conceito de funções cohomólogas se aplica também para grafos não planares.

4.1.3 O algoritmo de Schrijver

Nesta parte apresentaremos um algoritmo polinomial para resolver o $\text{PkC}(D, r_1, s_1, \dots, r_k, s_k)$. Vamos considerar D já imerso no plano e chamaremos de R a face externa de D . Sem perda de generalidade, podemos supor que os vértices $r_1, s_1, \dots, r_k, s_k$ são dois a dois distintos e que cada um destes vértices tem grau um. Não apresentaremos estas reduções por serem relativamente simples.

Sejam $P = (u_0, \dots, u_i = v, \dots, u_p)$ e $Q = (w_0, \dots, w_j = v, \dots, w_q)$ dois passeios (passeios não-orientados) em um grafo não-orientado (orientado) D . Dizemos que P e Q **se cruzam em** v se u_{i-1} e u_{i+1} ficam em “lados opostos” em relação ao passeio (passeio não-orientado) Q . Se para cada vértice u_i , $1 < i < p$, P e Q não se cruzam em u_i então dizemos que P e Q não se **cruzam**.

Dizemos que $\{P_1, \dots, P_l\}$ é uma coleção de passeios (passeios não-orientados) que não se cruzam se P_i e P_j não se cruzam, para quaisquer $1 \leq i < j \leq l$.

Para qualquer solução $\Pi = (P_1, \dots, P_k)$ para o $\text{PkC}(D, r_1, s_1, \dots, r_k, s_k)$ considere a função $\phi_\Pi : AD \rightarrow G_k$ assim definida:

- $\phi_\Pi(a) = g_i$, se o caminho P_i passa pelo arco a ($1 \leq i \leq k$);
- $\phi_\Pi(a) = 1$, se nenhum dos caminhos P_j ($1 \leq j \leq k$) passa pelo arco a .

Iremos descrever como obter uma lista de funções $\phi_1, \dots, \phi_N$, tal que para qualquer solução Π para o $\text{PkC}(D, r_1, s_1, \dots, r_k, s_k)$, a função ϕ_Π é R -homóloga a no mínimo uma das funções ϕ_j , $j = 1, \dots, N$. Diremos que uma lista deste tipo é uma **lista-solução-homóloga**.

Figura 4.1: Modificação a ser feita no grafo D .

Primeiramente, modifique o grafo D substituindo cada vértice v com $gr(v) > 3$ por um circuito cujo comprimento é $gr(v)$. Faça os arcos que incidiam em v no grafo original incidir nos novos vértices (veja Figura 4.1), mantendo os arcos dispostos segundo a imersão do grafo D e fazendo cada vértice novo ter grau três. Chame de D o grafo assim obtido.

É fácil ver que um subgrafo de D , minimal, que contém um caminho não-orientado entre quaisquer dois dos vértices $r_1, s_1, \dots, r_k, s_k$ é uma árvore. Diremos que um tal subgrafo é uma **árvore de conexão** para $(D, r_1, s_1, \dots, r_k, s_k)$. Claramente, em uma tal árvore de conexão os vértices $r_1, s_1, \dots, r_k, s_k$ têm grau um (pois estes vértices têm grau um em D) e os demais vértices têm grau no máximo três. Os arcos que estão em uma árvore de conexão para $(D, r_1, s_1, \dots, r_k, s_k)$ podem ser particionados em $4k - 3$ conjuntos, conforme ilustramos na Figura 4.2. Chamaremos cada um destes conjuntos de **série**.

Proposição 4.1.3 *Seja T uma árvore de conexão para $(D, r_1, s_1, \dots, r_k, s_k)$ e seja $\Pi = (P_1, \dots, P_k)$ uma solução para o $PkC(D, r_1, s_1, \dots, r_k, s_k)$. Existe uma única função $\psi_\Pi : AD \rightarrow G_k$ que é R -homóloga a ϕ_Π e $\psi_\Pi(a) = 1$, para cada arco $a \notin AT$.*

Dem: Precisamos mostrar que existe uma única função $f : FD \rightarrow G_k$ tal que

- $f(R) = 1$;
- $f(F_E)^{-1} \cdot \phi_\Pi(a) \cdot f(F_D) = 1$, para cada $a \notin AT$, onde F_E e F_D são as faces de D que ficam à esquerda e à direita de a , respectivamente.

Figura 4.2: Esquema de uma árvore de conexão para $(D, r_1, s_1, r_2, s_2, r_3, s_3)$.

Isto é equivalente a mostrar que existe uma única função $g : VD^* \rightarrow G_k$ tal que

- $g(R^*) = 1$;
- $g(F_E^*)^{-1} \cdot \phi_\Pi^*(a^*) \cdot g(F_D^*) = 1$, para cada $a \notin AT$.

Claramente, qualquer vértice F^* de D^* pode ser alcançado a partir de R^* percorrendo-se um caminho não-orientado $C = (R^*, a_1^*, F_1^*, \dots, a_l^*, F_l^* = F^*)$, onde $a_1, \dots, a_l$ são arcos de D que não estão em T . Como $g(R^*) = 1$ e $\psi_\Pi^*(a_i^*) = 1$, para $i = 1, \dots, l$, então $g(F^*) = \phi_\Pi^*(a_l^*)^{\varepsilon_l} \cdot \dots \cdot \phi_\Pi^*(a_1^*)^{\varepsilon_1}$, onde $\varepsilon_i = -1$, se C passa por a_i^* no sentido correto e $\varepsilon_i = 1$, caso contrário.

Como os caminhos $P_1, \dots, P_k$ não se cruzam então $g(F^*)$ é independente do caminho não-orientado C . Logo, existe uma única função $\psi_\Pi : AD \rightarrow G_k$ que é R -homóloga a ϕ_Π e tal que $\psi_\Pi(a) = 1$, para cada $a \in AD - AT$.

□

Considere o grafo D da Figura 4.3. Na Figura 4.4 mostramos uma árvore de conexão T para $(D, r_1, s_1, r_2, s_2, r_3, s_3)$. Observe que $\Pi = (P_1, P_2, P_3)$ é uma solução para

Figura 4.3: Um grafo D e uma coleção $\Pi = (P_1, P_2, P_3)$ que é uma solução para o $PkC(D, r_1, s_1, r_2, s_2, r_3, s_3)$.

Figura 4.4: Árvore de conexão T para $(D, r_1, s_1, r_2, s_2, r_3, s_3)$.

o $\text{PkC}(D, r_1, s_1, r_2, s_2, r_3, s_3)$. Vamos mostrar a idéia que está por trás do algoritmo que encontra a função ψ_Π que é R -homóloga a ϕ_Π e tal que $\psi_\Pi(a) = 1$, para cada $a \notin AT$. Considere o grafo obtido a partir de D , retirando-se todos os arcos que não estão em T ou em um dos caminhos P_i , $i = 1, 2, 3$. Em seguida, afaste ao longo do plano as curvas que representam os caminhos P_1, P_2, P_3 (e deforme-as se preciso) sem cruzar uma com a outra e de forma que elas fiquem sempre “paralelas” às séries da árvore T , só incidindo nos vértices de T . Seja $\vec{T}_G$ este novo grafo. Defina *série* em $\vec{T}_G$ de modo análogo ao que foi feito para T . Retire de $\vec{T}_G$ os arcos de T que não estão em nenhum dos caminhos P_i , $i = 1, 2, 3$. Substitua cada arco em $\vec{T}_G$ por uma aresta com os mesmos extremos que o arco, obtendo o grafo não-orientado T_G (veja a Figura 4.5). Note que as arestas de T_G podem ser particionados em três passeios Q_1, Q_2, Q_3 tais que:

- Q_i é um passeio de r_i a s_i , $i = 1, 2, 3$;
- Q_i e Q_j não se cruzam, $1 \leq i, j \leq 3$;
- $\bigcup_{i=1}^3 AQ_i = AT_G$.

Observe que o passeio Q_i resulta do caminho P_i . Substitua cada aresta em T_G por arcos que possuem o mesmo sentido do passeio que passa pela respectiva aresta. Seja $\phi' : AT_G \rightarrow G_k$ a função assim definida:

- $\phi'(a) = g_i$, se o passeio Q_i passa por a ($1 \leq i \leq 3$);
- $\phi'(a) = 1$, se nenhum dos passeios Q_j ($1 \leq j \leq 3$) passa pelo arco a .

Em seguida, determine ψ' que é R -homóloga a ϕ' e tal que $\psi'(a) = 1$, para cada arco a que não está no centro de uma série em $\vec{T}_G$ (conforme descrito na Proposição 4.1.3).

Para cada arco a em T , defina $\psi_\Pi(a)$ como o valor (reverso do valor) que ψ' assume em um arco central na série em $\vec{T}_G$ correspondente à série de T em que está o arco a , se o sentido destes arcos coincidirem (forem diferentes). Faça $\psi_\Pi(a) = 1$ para todo $a \notin AT$. A função ψ_Π é R -homóloga a ϕ_Π . Na Figura 4.4, apresentamos a árvore de conexão T com o valor de ψ_Π indicado ao lado de certos arcos em T .

Figura 4.5: O grafo T_G obtido a partir de T e de $\Pi = (P_1, P_2, P_3)$.

Observe que:

1. Se tivermos os passeios Q_i , $i = 1, 2, 3$, então podemos encontrar a função ψ_Π que é R -homóloga a ϕ_Π .
2. Para obtermos os Q_i 's precisamos encontrar um T_G que seja particionável em k passeios Q_i 's de r_i a s_i que não se cruzam. Um tal T_G será chamado **válido**.
3. Para encontrar todos os T_G 's que são válidos, partimos da árvore T e atribuímos a cada série uma certa quantidade de arestas paralelas, quantidade esta inferior a $|AD|$. Isto equivale a atribuir pesos inteiros entre 0 e $|AD|$ a cada uma das séries da árvore T (veja Figura 4.6). Para cada atribuição de peso W às séries de T temos um grafo que denotaremos por $\hat{T}_G(W)$. Note que temos no máximo $|AD|^{4k-3}$ tais grafos.

Suponha agora que temos um grafo $\hat{T}_G(W)$. Veremos a seguir como podemos descobrir se tal $\hat{T}_G(W)$ corresponde a um T_G válido; e se for, como encontrar os passeios Q_i 's.

Vamos explicar a idéia através de um exemplo. Para isso, considere um grafo $\hat{T}_G(W)$, como indicado na Figura 4.6. Então, conforme mostramos na Figura 4.7(a), por δ deve passar

$$\begin{aligned} x & \text{ caminhos entre } r_3 \text{ e } \Upsilon, \\ y & \text{ caminhos entre } r_3 \text{ e } s_3 \text{ e} \\ z & \text{ caminhos entre } s_3 \text{ e } \Upsilon, \end{aligned}$$

devendo x, y, z satisfazer o sistema:

$$\begin{aligned} x + y &= 5 \\ x + z &= 6 \\ y + z &= 5 \end{aligned}$$

A única solução deste sistema é $x = 3$, $y = 2$, $z = 3$ (veja Figura 4.7(b)).

Digamos que por Υ passem

Figura 4.6: Um grafo $\hat{T}_G(W)$, obtido de T atribuindo a cada série um valor inferior a $|AD|$.

p caminhos entre r_2 e δ ,
 q caminhos entre r_2 e β e
 w caminhos entre β e δ .

Neste caso, temos o seguinte sistema:

$$\begin{aligned} p + q &= 3 \\ q + w &= 9 \\ p + w &= 6 \end{aligned}$$

A única solução deste sistema é $p = 0$, $q = 3$, $w = 6$ (veja Figura 4.8(b)). Fazendo o mesmo para os vértices β e α , obtemos as correspondentes soluções.

Tendo tais soluções tentamos formar passeios que não se cruzam, fazendo “colagens” (se possível), como mostramos na Figura 4.9. Após fazer as “colagens”, concatenamos os passeios obtidos nos vértices r_i e s_i , $i = 1, 2, 3$, que estão em posições simétricas em relação à aresta do meio que incide em r_i ou s_i , como é esboçado na Figura 4.10. Desta

Figura 4.7: Caminhos no grafo T que passam pelo vértice δ .

Figura 4.8: Caminhos no grafo T que passam pelo vértice Υ .

Figura 4.9: Colagem dos passeios indicados nas Figuras 4.7 e 4.8.

forma encontramos, caso existam, os passeios Q_1, Q_2, Q_3 . Se para uma escolha W de pesos não for possível obter k passeios Q_i 's como desejamos, então descartamos tal W .

Se executarmos o procedimento descrito no exemplo para cada $\hat{T}_G(W)$, desprezando aqueles que não correspondem a um T_G válido, obteremos uma lista de funções $\phi_1', \dots, \phi_N'$, tal que para qualquer solução Π para o $\text{PkC}(D, r_1, s_1, \dots, r_k, s_k)$, ϕ_Π é R -homóloga a exatamente uma das funções ϕ_j' , $j = 1, \dots, N$. Considere a partir de agora o grafo D original (que pode ter grau máximo maior do que três). Seja ϕ_j a restrição de ϕ_j' aos arcos do grafo D original. É fácil ver que $\phi_1, \dots, \phi_N$ é uma lista-solução-homóloga para o $\text{PkC}(D, r_1, s_1, \dots, r_k, s_k)$.

Como o número de $\hat{T}_G(W)$ possíveis é no máximo $|AD|^{4k-3}$ e a solução de um sistema linear pode ser obtida em tempo polinomial temos o seguinte lema.

Figura 4.10: Concatenação dos passeios indicados na Figura 4.9, mantendo a simetria em relação à aresta do meio e contornando r_2 , r_3 e s_3 .

Lema 4.1.4 *O algoritmo para encontrar uma lista-solução-homóloga é polinomial.*

□

Veremos, agora, como os resultados anteriores podem ser usados para resolver o $\text{PkC}(D, r_1, s_1, \dots, r_k, s_k)$.

Considere o grafo orientado D^* que é o dual de D . Construa, a partir de D^* , um **grafo dual cordal** D^+ adicionando em cada face de D^* todas as cordas possíveis. Mais precisamente, para quaisquer vértices F_1^* e F_2^* de D^* para os quais existe um caminho não-orientado π de F_1^* a F_2^* na fronteira de alguma face de D^* , acrescentamos um arco a_π de F_1^* a F_2^* . Note que D^+ não é necessariamente planar e nem único, já que o sentido dos arcos a_π não está univocamente determinado.

Para qualquer $\phi : AD \rightarrow G_k$ defina $\phi^+ : AD^+ \rightarrow G_k$ da seguinte maneira:

- $\phi^+(a^*) = \phi(a)$, para cada arco $a \in AD$;
- $\phi^+(a_\pi) = \phi(a_1)^{\varepsilon_1} \cdots \phi(a_t)^{\varepsilon_t}$, para qualquer caminho $\pi = ((a_1^*)^{\varepsilon_1}, \dots, (a_t^*)^{\varepsilon_t})$ na fronteira de alguma face de D^* , com $\varepsilon_1, \dots, \varepsilon_t \in \{+1, -1\}$.

Lema 4.1.5 *Seja D^+ um grafo dual cordal obtido de D . Seja $\Gamma : AD^+ \rightarrow G_k$ definido por $\Gamma(a^*) = \{1, g_1, \dots, g_k\}$, para cada arco $a \in AD$, e $\Gamma(a_\pi) = \{1, g_1, g_1^{-1}, \dots, g_k, g_k^{-1}\}$, para cada arco a_π que foi acrescentado a D durante a construção de D^+ . Se ψ^+ é uma solução para o $\text{PVC}(D^+, R^*, \phi^+, \Gamma)$ e e_1 e e_2 são dois arcos em D que incidem em um mesmo vértice v com $\psi(e_i) \neq 1$, para $i = 1, 2$, então $\psi(e_1) = \psi(e_2)^\varepsilon$, onde $\varepsilon = -1$ se e_1 e e_2 são arcos que saem de v ou e_1 e e_2 são arcos que entram em v e $\varepsilon = 1$ caso contrário.*

Dem: Sejam v um vértice de D e $a_1, a_2, \dots, a_{q-1}, a_q$ arcos que incidem em v e tais que $\psi(a_1) \neq 1$, $\psi(a_i) = 1$, para $i = 2, \dots, q-1$ e $\psi(a_q) \neq 1$. Seja π o caminho não-orientado em D^* formado pelos arcos $a_1^*, a_2^*, \dots, a_q^*$. Seja $\varepsilon_i = 1$ se a_i é um arco que sai de v , e $\varepsilon_i = -1$ caso contrário. Considere o grafo D^+ e seja F^* o vértice de D^* de onde sai o arco a_π . Como $\phi^+(a_\pi) \cdot \phi^+(a_q^*)^{\varepsilon_q} \cdots \phi^+(a_1^*)^{\varepsilon_1} = \phi^+(a_1^*)^{-\varepsilon_1} \cdots \phi^+(a_q^*)^{-\varepsilon_q} \cdot \phi^+(a_q^*)^{\varepsilon_q} \cdots \phi^+(a_1^*)^{\varepsilon_1} = 1$, então

$$\psi^+(a_\pi) \cdot \psi^+(a_q^*)^{\varepsilon_q} \dots \psi^+(a_1^*)^{\varepsilon_1} = f(F^*)^{-1} \cdot \phi^+(a_\pi) \cdot \phi^+(a_q^*)^{\varepsilon_q} \dots \phi^+(a_1^*)^{\varepsilon_1} \cdot f(F^*) = f(F^*)^{-1} \cdot 1 \cdot f(F^*) = 1.$$

Assim, $\psi^+(a_\pi) = \psi^+(a_1^*)^{-\varepsilon_1} \cdot \psi^+(a_q^*)^{-\varepsilon_q}$. Como $|\psi^+(a_\pi)| \leq 1$ e $\psi(a_i^*) \neq 1$, $i = 1, q$, então $\varepsilon_1 = -\varepsilon_q$ e $\psi^+(a_1^*) = \psi^+(a_q^*)^{-1}$. Logo, $\psi(a_1) = \psi(a_q)^{-1}$.

Portanto, se ψ^+ é uma solução para o $PVC(D^+, R^*, \phi^+, \Gamma)$ e e_1 e e_2 são dois arcos em D que incidem em um mesmo vértice v com $\psi(e_i) \neq 1$, para $i = 1, 2$ então $\psi(e_1) = \psi(e_2)^\varepsilon$.

□

Lema 4.1.6 *Seja Π uma solução para o $PkC(D, r_1, s_1, \dots, r_k, s_k)$. Se $\psi^+ : AD^+ \rightarrow G_k$ é R^* -cohomóloga a ϕ_Π^+ e $\psi^+(a^*) \in \{1, g_1, \dots, g_k\}$, para todo $a \in AD$, então existe em D , para cada $i = 1, \dots, k$, um passeio orientado P_i de r_i a s_i que passa somente por arcos $a \in AD$ para os quais $\psi(a) = g_i$.*

Dem: Seja a_{i1} o arco em D que sai de r_i . Mostraremos que $\psi(a_{i1}) = g_i$. Seja F_{i1} a face de D adjacente a a_{i1} . Então $\psi^+(a_{i1}^*) = f(F_{i1}^*)^{-1} \cdot \phi_\Pi^+(a_{i1}^*) \cdot f(F_{i1}^*) = f(F_{i1}^*)^{-1} \cdot g_i \cdot f(F_{i1}^*)$.

Como $\psi^+(a_{i1}^*) \in \Gamma(a) = \{1, g_1, \dots, g_k\}$ então $f(F_{i1}^*) = 1$ ou $f(F_{i1}^*) = g_i$. Em qualquer destes casos $\psi^+(a_{i1}^*) = g_i$.

De forma análoga prova-se que o valor de ψ para o arco que está entrando em s_i é g_i .

Seja $a_{ij} = u_{j-1}u_j$, $u_j \neq s_i$, um arco em D para o qual $\psi^+(a_{ij}^*) = g_i$. Sejam $a_1, a_2, \dots, a_p$ os outros arcos de D que entram ou saem de u_j , no sentido anti-horário a partir de a_{ij} . Se fizermos $\varepsilon_l = 1$ quando o arco a_l entra em u_j e $\varepsilon_l = -1$ quando a_l sai de u_j , então

$$\psi^+(a_{ij}^*) \cdot \psi^+(a_1^*)^{\varepsilon_1} \dots \psi^+(a_p^*)^{\varepsilon_p} = f(F^*)^{-1} \cdot \phi_\Pi^+(a_{ij}^*) \cdot \phi_\Pi^+(a_1^*)^{\varepsilon_1} \dots \phi_\Pi^+(a_p^*)^{\varepsilon_p} \cdot f(F^*) = f(F^*)^{-1} \cdot 1 \cdot f(F^*) = 1.$$

Como $\psi^+(a_{ij}^*) = g_i$ então $\psi^+(a_t^*)^{\varepsilon_t} = g_i^{-1}$, para algum t , $1 \leq t \leq p$. Como $\psi^+(a_t^*) \in \{1, g_1, \dots, g_k\}$ então $\psi^+(a_t^*) = g_i$ e $\varepsilon_t = -1$. Logo, a_t é um arco em D que sai de u_j e $\psi^+(a_t^*) = g_i$.

Portanto, existe um passeio orientado P_i que vai de r_i a s_i , $i = 1, \dots, k$, que passa somente por arcos $a \in AD$ para os quais $\psi(a) = g_i$.

□

Teorema 4.1.7 *Existe algoritmo polinomial para o $\text{PkC}(D, r_1, s_1, \dots, r_k, s_k)$.*

Dem: Segue pelo que explicamos anteriormente que podemos encontrar, em tempo polinomial, uma lista de funções $\phi_1, \dots, \phi_N : AD \rightarrow G_k$ tal que para cada solução Π para o $\text{PkC}(D, r_1, s_1, \dots, r_k, s_k)$, ϕ_Π é R -homóloga a no mínimo um dos ϕ_j .

Construa D^+ a partir de D e faça $\Gamma(a^*) = \{1, g_1, \dots, g_k\}$, para cada arco $a \in AD$, e $\Gamma(a_\pi) = \{1, g_1, g_1^{-1}, \dots, g_k, g_k^{-1}\}$, para cada arco a_π que foi acrescentado a D^* para construir D^+ .

Pelo Teorema 4.1.15 (no final desta seção) podemos encontrar, em tempo polinomial, para cada $j = 1, \dots, N$, uma função $\psi^+ : AD^+ \rightarrow G_k$ que é R^* -cohomóloga a ϕ_j^+ em D^+ , com $\psi^+(b) \in \Gamma(b)$ para cada arco b em D^+ , desde que exista um tal ψ^+ .

Se nós encontramos um tal ψ^+ , então considere (se existir), para cada i , $1 \leq i \leq k$, um caminho P_i de r_i a s_i que passa apenas por arcos $a \in AD$ nos quais $\psi^+(a^*) = g_i$. Se tais caminhos existem então pelo Lema 4.1.5 eles são disjuntos e, portanto, constituem uma solução para o $\text{PkC}(D, r_1, s_1, \dots, r_k, s_k)$.

Precisamos mostrar que se para nenhum j em $\{1, \dots, N\}$ encontramos tais caminhos, então o $\text{PkC}(D, r_1, s_1, \dots, r_k, s_k)$ não tem solução. Suponha que $\Pi = (C_1, \dots, C_k)$ é uma solução para este problema. Então para algum $j \in \{1, \dots, N\}$ as funções ϕ_Π e ϕ_j são R -homólogas. Ou seja, ϕ_Π^+ e ϕ_j^+ são R^* -cohomólogas. A Proposição 4.1.1 nos garante que qualquer ψ' R^* -cohomóloga a ϕ_j^+ é, também, R^* -cohomóloga a ϕ_Π^+ . Segue pelo Lema 4.1.6 que existem em D caminhos P'_i , $i = 1, \dots, k$, que vão de r_i a s_i usando apenas arcos $a \in AD$ para os quais $\psi'(a) = g_i$.

□

4.1.4 Algoritmo para o problema da viabilidade cohomológica

No que segue apresentaremos um algoritmo polinomial para resolver o problema da viabilidade cohomológica, PVC, introduzido na Seção 4.1.1. Inicialmente, provaremos um resultado que será muito utilizado nesta parte.

Lema 4.1.8 *Sejam x, z palavras de G_k e b um símbolo de G_k . Se $x \preceq b \cdot z$ e $z \preceq b^{-1} \cdot x$ então $x^{-1} \cdot b \cdot z = 1$ ou $x = z = 1$.*

Dem: Seja $y = x^{-1} \cdot b \cdot z$. Suponha que $y \neq 1$. Como $x \preceq b \cdot z$, então $b \cdot z = xy$. (I)

Note que $y = x^{-1} \cdot b \cdot z$ implica que $x \cdot y = b \cdot z$. (II)

De (I) e (II) segue que $xy = x \cdot y$.

Como $z \preceq b^{-1} \cdot x$ e $y^{-1} = z^{-1} \cdot b^{-1} \cdot x$, então $b^{-1} \cdot x = zy^{-1}$. Disto segue que $x^{-1} \cdot b = yz^{-1}$. (III)

Note que $y = x^{-1} \cdot b \cdot z$ implica que $y \cdot z^{-1} = x^{-1} \cdot b$. (IV)

De (III) e (IV) segue que $yz^{-1} = y \cdot z^{-1}$.

Logo, $xyz^{-1} = x \cdot yz^{-1} = x \cdot y \cdot z^{-1} = b$. (pois $yz^{-1} = y \cdot z^{-1}$ e $y = x^{-1} \cdot b \cdot z$).

Mas $y \neq 1$ e $xyz^{-1} = b$ implicam que $x = z = 1$.

□

É imediato que o $PVC(D, r, \phi, \Gamma)$ pode ser formulado da seguinte maneira:

Encontrar uma função $f : VD \rightarrow G_k$ que satisfaça:

- $f(r) = 1$;
- $f(u)^{-1} \cdot \phi(a) \cdot f(w) \in \Gamma(a)$, para cada arco $a = uw \in AD$.

Uma tal função f é chamada **viável**. Em outras palavras, o problema da viabilidade cohomológica consiste em encontrar uma função viável f , dados D , r , ϕ e Γ .

Lema 4.1.9 *Para resolver o problema da viabilidade cohomológica podemos supor, sem perda de generalidade, que $|\phi(a)| \leq 1$ para cada arco $a \in AD$.*

Dem: Suponha que $|\phi(e)| > 1$, para algum $e = uw \in AD$. Seja $\phi(e) = b_1 \dots b_t$, $t \geq 2$. Seja D' o grafo orientado obtido a partir de D substituindo e pelo caminho $(u = u_0, e_1, u_1, \dots, e_t, u_t = w)$. Seja $\phi' : AD' \rightarrow G_k$ definida por:

$$(a) \phi'(a) = \phi(a), \text{ para cada } a \in AD' - \{e_1, \dots, e_t\}.$$

$$(b) \phi'(e_i) = b_i, \text{ para } i = 1, \dots, t.$$

Faça $\Gamma'(e_1) = \Gamma(e)$, $\Gamma'(e_i) = \{1\}$, para $i = 2, \dots, t$ e $\Gamma'(a) = \Gamma(a)$, para cada $a \in AD' - \{e_1, \dots, e_t\}$.

Suponha que $f : VD \rightarrow G_k$ seja uma função viável para o problema $PVC(D, r, \phi, \Gamma)$. Seja $f' : VD' \rightarrow G_k$ definida por:

$$(a) f'(v) = f(v), \text{ para cada } v \in VD' - \{u_1, \dots, u_{t-1}\}.$$

$$(b) f'(u_{i-1}) = b_i \cdot f'(u_i), \text{ para } i = t, \dots, 2.$$

Provamos a seguir que f' é viável para o problema $PVC(D', r, \phi', \Gamma')$.

- $f'(r) = f(r) = 1$, pois $r \notin \{u_1, \dots, u_{t-1}\}$.
- Para $i = 2, \dots, t$ temos: $f'(u_{i-1})^{-1} \cdot \phi'(e_i) \cdot f'(u_i) = f'(u_{i-1})^{-1} \cdot f'(u_{i-1}) = 1 \in \Gamma'(e_i)$.
- Como $f'(u_{i-1})^{-1} \cdot b_i \cdot f'(u_i) = 1$, para $i = 2, \dots, t$ então $f'(u_0)^{-1} \cdot \phi'(e_1) \cdot f'(u_1) = f'(u_0)^{-1} \cdot \phi'(e_1) \cdot f'(u_1) \cdot 1 \dots 1 = f'(u_0)^{-1} \cdot b_1 \cdot f'(u_1) \cdot f'(u_1)^{-1} \cdot b_2 \cdot f'(u_2) \cdot f'(u_2)^{-1} \cdot b_3 \cdot f'(u_3) \dots f'(u_{t-1}) \cdot b_t \cdot f'(u_t) = f'(u_0) \cdot b_1 \cdot b_2 \dots b_t \cdot f'(u_t) = f(u_0) \cdot \phi(e) \cdot f(u_t) \in \Gamma(e) = \Gamma'(e_1)$.
- $f'(u)^{-1} \cdot \phi'(a) \cdot f'(w) = f(u)^{-1} \cdot \phi(a) \cdot f(w) \in \Gamma(a) = \Gamma'(a)$, para cada $a = uw \in AD' - \{e_1, \dots, e_t\}$.

Considere, agora, uma função viável, digamos g , para o $\text{PVC}(D', r, \phi', \Gamma')$. Como g é viável para o $\text{PVC}(D', r, \phi', \Gamma')$ e $\Gamma'(e_i) = 1$, para $i = 2, \dots, t$, então $g(u_{i-1})^{-1} \cdot b_i \cdot g(u_i) = 1$, para $i = 2, \dots, t$. Deste modo obtemos:

$$\begin{aligned} g(u_0)^{-1} \cdot \phi(e) \cdot g(u_t) &= g(u_0)^{-1} \cdot b_1 b_2 \dots b_t \cdot g(u_t) = g(u_0)^{-1} \cdot b_1 \cdot g(u_1) \cdot g(u_1)^{-1} \cdot b_2 \cdot g(u_2) \cdot \\ g(u_2)^{-1} \cdot b_3 \cdot g(u_3) \cdot \dots \cdot g(u_{t-1})^{-1} \cdot b_t \cdot g(u_t) &= g(u_0)^{-1} \cdot b_1 \cdot g(u_1) \cdot 1 \cdot 1 \dots 1 = g(u_0)^{-1} \cdot b_1 \cdot g(u_1) \in \\ \Gamma'(e) &= \Gamma(e). \end{aligned}$$

Além disso, claramente $g(u)^{-1} \cdot \phi(a) \cdot g(w) \in \Gamma(a)$ para todo arco $a = uw \in AD - \{e\}$. Logo, g restrito a VD é uma função viável para o $\text{PVC}(D, r, \phi, \Gamma)$ e, portanto, podemos supor que $|\phi(a)| \leq 1$, para cada arco $a \in AD$.

□

É fácil ver que para resolver o problema da viabilidade cohomológica podemos supor que para cada arco $a = uw \in AD$, $a^{-1} = wu$ também é um arco de D , com $\Gamma(a^{-1}) = \Gamma(a)^{-1} = \{x^{-1} : x \in \Gamma(a)\}$ e $\phi(a^{-1}) = \phi(a)^{-1}$, se $a^{-1} \notin AD$. Doravante sempre que nos referirmos ao PVC vamos supor que isto ocorra e que $|\phi(a)| \leq 1$ para todo $a \in AD$.

Para encontrarmos uma função viável, faremos uso de funções pré-viáveis.

Dizemos que uma função $f : |VD| \rightarrow G_k$ é **pré-viável** se:

- (a) $f(r) = 1$;
- (b) para cada arco $a = uw \in AD$ vale que se $f(u)^{-1} \cdot \phi(a) \cdot f(w) \notin \Gamma(a)$ então $f(u) = f(w) = 1$.

Para quaisquer funções $f, g : VD \rightarrow G_k$, dizemos que $f \preceq g$ se, e somente se, $f(v) \preceq g(v)$, para todo $v \in VD$. Note que, se acrescentarmos o elemento ∞ ao conjunto das funções de VD a G_k , temos um reticulado (definido pela ordem parcial). Aqui também vamos indicar a operação $\inf$ por $\wedge$ e $\sup$ por $\vee$.

Proposição 4.1.10 *Se $f_1, f_2 : VD \rightarrow G_k$ são pré-viáveis então $f = f_1 \wedge f_2$ é pré-viável.*

Dem: Claramente $f(r) = 1$. Seja $a = uw \in AD$ tal que $y = f(u)^{-1} \cdot \phi(a) \cdot f(w) \notin \Gamma(a)$. Se não existir um tal arco em AD então f é pré-viável. Precisamos mostrar que $f(u) =$

$f(w) = 1$. Como $y \notin \Gamma(a)$ então $y \neq 1$. Supondo que não ocorre $f(u) = f(w) = 1$, devido ao lema 4.1.8 e usando a^{-1} para concluir que $y^{-1} = f(w)^{-1} \cdot \phi(a)^{-1} \cdot f(u) \notin \Gamma(a)^{-1}$ podemos supor, sem perda de generalidade, que $f(u) \not\preceq \phi(a) \cdot f(w)$. Como $f(u) \not\preceq \phi(a) \cdot f(w)$ e $f = f_1 \wedge f_2$ então existe $i \in \{1, 2\}$ tal que $y \preceq f(u)^{-1} \cdot \phi(a) \cdot f_i(w)$. Por outro lado, sabemos que o primeiro símbolo de $f(u)^{-1}$ e y são iguais, pois $f(u) \not\preceq \phi(a) \cdot f(w)$. Assim, $y \preceq f(u)^{-1} \cdot \phi(a) \cdot f_i(w)$ e $f(u)^{-1} \cdot \phi(a) \cdot f_i(w)$ é um segmento de $f_i(u)^{-1} \cdot \phi(a) \cdot f_i(w)$ e como $y \notin \Gamma(a)$ então $f_i(u)^{-1} \cdot \phi(a) \cdot f_i(w) \notin \Gamma(a)$. Usando o fato de que f_i é pré-viável obtemos que $f_i(u) = f_i(w) = 1$ e, portanto, $f(u) = f(w) = 1$, pois $f \preceq f_i$. Contradizendo a hipótese que assumimos.

□

Corolário 4.1.11 *Dada uma função $f : VD \rightarrow G_k$, se existe alguma função pré-viável $g : VD \rightarrow G_k$ tal que $g \succeq f$, então existe uma função pré-viável mínima $\bar{f} : VD \rightarrow G_k$ tal que $\bar{f} \succeq f$.*

□

Dada uma função $f : VD \rightarrow G_k$, denotaremos por $\bar{f}$ a função pré-viável mínima g tal que $g \succeq f$. Se não existir uma tal função diremos que $\bar{f} = \infty$. Indicaremos por σ o comprimento máximo das palavras $\phi(a)$ e das palavras que estão em $\Gamma(a)$, onde $a \in AD$, isto é:

$$\sigma = \max_{a \in AD} \{ \{|\phi(a)|\} \cup \{|x| : x \in \Gamma(a)\} \}.$$

Proposição 4.1.12 *Considere uma função $f : VD \rightarrow G_k$. Se $\bar{f}$ é finito então*

- (a) $f(r) = 1$ e $|f(v)| < 2\sigma|VD|$, para cada $v \in VD$;
- (b) para cada arco $a = uw \in AD$ temos que se $f(u)^{-1} \cdot \phi(a) \cdot f(w) \notin \Gamma(a)$ então $f(u) \preceq \phi(a) \cdot f(w)$ ou $f(w) \preceq \phi(a)^{-1} \cdot f(u)$.

Dem: (a) Claramente $f(r) = 1$, pois $f(r) \preceq \bar{f}(r) = 1$.

Usaremos indução no comprimento l_u de um caminho não-orientado mínimo de r a u para mostrar que $|\bar{f}(u)| \leq 2\sigma l_u$. Tome um vértice $v \in VD$. Se $l_v = 1$ então existe um

arco em D cujos extremos são r e v . Podemos assumir, sem perda de generalidade, que $a = rv \in AD$. Como $\bar{f}$ é pré-viável, $x = \bar{f}(r)^{-1} \cdot \phi(a) \cdot \bar{f}(v) \in \Gamma(a)$ ou $\bar{f}(r) = \bar{f}(v) = 1$. Se $\bar{f}(v) = 1$ então $f(v) = 1$ e, trivialmente, $|f(v)| \leq 2\sigma$. Se $x \in \Gamma(a)$ então $\bar{f}(v) = \phi(a)^{-1} \cdot \bar{f}(r) \cdot x = \phi(a)^{-1} \cdot x$. Assim, $|\bar{f}(v)| \leq \sigma + \sigma = 2\sigma$.

Suponha que para todo $u \in VD$ tal que um caminho não-orientado mínimo de r a u tem comprimento l_u , onde l_u é menor do que um certo l , então $|\bar{f}(u)| \leq 2\sigma l_u$. Seja $v \in VD$ e suponha que um caminho não-orientado mínimo de r a v tenha comprimento l . Seja u o último vértice em um caminho não-orientado mínimo de r a v . Podemos supor, sem perda da generalidade, que $a = uv$ é o último arco neste caminho. Temos que $\bar{f}(u)^{-1} \cdot \phi(a) \cdot \bar{f}(v) \in \Gamma(a)$ ou $\bar{f}(r) = \bar{f}(v) = 1$. Se $\bar{f}(v) \neq 1$ então $\bar{f}(u)^{-1} \cdot \phi(a) \cdot \bar{f}(v) = y$, para algum $y \in \Gamma(a)$. Assim, $\bar{f}(v) = \phi(a)^{-1} \cdot \bar{f}(u) \cdot y$. Logo, $|\bar{f}(v)| \leq |\phi(a)| + |\bar{f}(u)| + |y| \leq \sigma + 2\sigma(l - 1) + \sigma = 2\sigma l$.

Portanto, para cada $v \in VD$ temos que $|\bar{f}(v)| \leq 2\sigma l$, onde l é o comprimento de um caminho não-orientado mínimo de r a v . Assim, $|f(v)| \leq |\bar{f}(v)| \leq 2\sigma l < 2\sigma|VD|$ (pois $l \leq |VD| - 1$), para cada vértice $v \in VD$.

(b) Suponha que $f(u) \not\preceq \phi(a) \cdot f(w)$ e $f(w) \not\preceq \phi(a)^{-1} \cdot f(u)$. Como $f(u) \not\preceq \phi(a) \cdot f(w)$ então o primeiro símbolo de $f(u)^{-1} \cdot \phi(a) \cdot f(w)$ é igual ao primeiro símbolo de $f(u)^{-1}$. Similarmente, $f(w) \not\preceq \phi(a)^{-1} \cdot f(u)$ implica que o último símbolo de $f(u)^{-1} \cdot \phi(a) \cdot f(w)$ é igual ao último símbolo de $f(w)$. Segue que $f(u)^{-1} \cdot \phi(a) \cdot f(w)$ é um segmento de $\bar{f}(u)^{-1} \cdot \phi(a) \cdot \bar{f}(w)$, pois $f \preceq \bar{f}$. Temos, então, que $\bar{f}(u)^{-1} \cdot \phi(a) \cdot \bar{f}(w) \notin \Gamma(a)$, pois, por hipótese, $f(u)^{-1} \cdot \phi(a) \cdot f(w) \notin \Gamma(a)$ e, além disso, $\Gamma(a)$ é hereditário. Pela pré-viabilidade de $\bar{f}$, segue que $\bar{f}(u) = \bar{f}(w) = 1$ e, portanto, $f(u) = f(w) = 1$. Isto contradiz a hipótese que assumimos, pois, trivialmente, 1 é segmento inicial de qualquer palavra.

□

Algoritmo para encontrar $\bar{f}$

Entrada: Um grafo orientado conexo D , uma função $\phi : AD \rightarrow G_k$, uma função $f : VD \rightarrow G_k$ e para cada arco $a \in AD$ um subconjunto hereditário $\Gamma(a)$ de G_k .

Saída: $\bar{f}$.

1. Enquanto f não é pré-viável e f não viola as condições (a) ou (b) da Proposição 4.1.12 faça o seguinte:
 - (a) escolha um arco $a = uw \in AD$ tal que $f(u)^{-1} \cdot \phi(a) \cdot f(w) \notin \Gamma(a)$ e $f(w) \not\leq \phi(a)^{-1} \cdot f(u)$.
 - (b) escreva $\phi(a) \cdot f(w) = xy$, com $y \in \Gamma(a)$ e $|y|$ máximo e faça $f(u) := x$.
2. Se f é pré-viável então faça $\bar{f} := f$ senão faça $\bar{f} := \infty$.
3. Devolva $\bar{f}$.

Proposição 4.1.13 *Em cada iteração do algoritmo para encontrar $\bar{f}$, a atualização de f aumenta $|f(u)|$, para algum vértice u , e não modifica $\bar{f}$.*

Dem: Considere uma iteração do algoritmo. Seja f' a função f após a atualização. Segue pela validade do item (b) da Proposição 4.1.12 e da escolha do arco a feita pelo algoritmo que $f(u) \preceq \phi(a) \cdot f(w)$ (o Lema 4.1.8 e o fato de f não ser pré-viável implicam que esta escolha sempre é possível). Como $f(u)^{-1} \cdot \phi(a) \cdot f(w) \notin \Gamma(a)$ e $x^{-1} \cdot \phi(a) \cdot f(w)$ pertence a $\Gamma(a)$ (pois $x^{-1} \cdot \phi(a) \cdot f(w) = y$ e $y \in \Gamma(a)$), então $f(u)$ deve ser um segmento próprio de x . Portanto, $|f'(u)| > |f(u)|$.

Mostraremos, agora, que $\bar{f}' = \bar{f}$. Precisamos mostrar que $f' \preceq \bar{f}$, isto é, $f'(u) \preceq \bar{f}(u)$. Existem as seguintes possibilidades:

caso 1: $\bar{f}(u) \preceq \phi(a) \cdot f(w)$.

Neste caso, $\bar{f}(u)^{-1} \cdot \phi(a) \cdot f(w) \preceq \bar{f}(u)^{-1} \cdot \phi(a) \cdot \bar{f}(w)$. Como $\bar{f}(u)^{-1} \cdot \phi(a) \cdot \bar{f}(w) \in \Gamma(a)$ (pois $\bar{f}(w) \neq 1$, já que $\bar{f}(w) \not\leq \phi(a)^{-1} \cdot f(u)$) então $\bar{f}(u)^{-1} \cdot \phi(a) \cdot f(w) \in \Gamma(a)$. Segue, pela escolha de y feita pelo algoritmo que $x \preceq \bar{f}(u)$.

caso 2: $\bar{f}(u) \not\leq \phi(a) \cdot f(w)$.

caso 2.1 $\phi(a) \cdot f(w) \preceq \bar{f}(u)$.

Neste caso, existe $z \preceq \bar{f}(u)$ tal que $z^{-1} \cdot \phi(a) \cdot f(w) = 1 \in \Gamma(a)$. Assim, $x \preceq z \preceq \bar{f}(u)$.

caso 2.2: $\phi(a) \cdot f(w) \not\leq \bar{f}(u)$.

Temos então $\bar{f}(u)^{-1} \cdot \phi(a) \cdot f(w) \preceq \bar{f}(u)^{-1} \cdot \phi(a) \cdot \bar{f}(w) \in \Gamma(a)$. Assim, $x \preceq \bar{f}(u)$.

Portanto, $\bar{f}' = \bar{f}$.

□

Cada iteração do algoritmo aumenta $|f(u)|$ para algum vértice u . Assim, em no máximo $2\sigma|VD|^2$ iterações o item (a) ou (b) da Proposição 4.1.12 será violada ou $\bar{f}$ será encontrado. Portanto, o algoritmo para encontrar $\bar{f}$ é polinomial.

Para cada arco $a = uw \in AD$, seja $f_a : VD \rightarrow G_k$ a função assim definida:

- $f_a(u) = \phi(a)$;
- $f_a(v) = 1$, se $v \neq u$.

Considere agora os seguintes conjuntos:

$$E = \{ \{a, a'\} : a, a' \in AD \text{ e } \bar{f}_a \vee \bar{f}_{a'}, \text{ é finito e pré-viável } \};$$

$$E' = \{ \{a, a^{-1}\} : a \in AD \text{ e } \phi(a) \notin \Gamma(a) \}.$$

Proposição 4.1.14 *Sejam E , E' e f_a os objetos acima definidos. Se existe um subconjunto X de AD tal que cada par de arcos em X pertence a E e, além disso, X intercepta cada par em E' , então a função $f = \bigvee_{a \in X} \bar{f}_a$ é viável. Se não existe um tal subconjunto então não existe função viável para o $PVC(D, r, \phi, \Gamma)$.*

Dem: Suponha que exista um tal subconjunto X . Como $\bar{f}_a \vee \bar{f}_{a'}$ é finito e pré-viável para cada par de arcos $a, a' \in X$, então f é finito e $f(r) = 1$. Suponha que f não é viável, ou seja, que exista um arco $a = uw \in AD$ para o qual $f(u)^{-1} \cdot \phi(a) \cdot f(w) \notin \Gamma(a)$.

Seja $f(u) = \bar{f}_{a'}(u)$ e $f(w) = \bar{f}_{a''}(w)$. Como $\{a', a''\} \in E$ então $f = \bigvee_{a \in X} \bar{f}_a$ é pré-viável. Logo $f(u) = f(w) = 1$, pois assumimos a hipótese de que $f(u)^{-1} \cdot \phi(a) \cdot f(w) \notin \Gamma(a)$. Deste modo, temos que $\phi(a) \notin \Gamma(a)$, pois $f(u)^{-1} \cdot \phi(a) \cdot f(w) = 1 \cdot \phi(a) \cdot 1 = \phi(a)$. Segue que $\{a, a^{-1}\} \in E'$. Como X intercepta cada par em E' , então $a \in X$ ou $a^{-1} \in X$. Podemos assumir, por simetria, que $a \in X$. Como $f = \bigvee_{e \in X} \bar{f}_e$ e $a \in X$ então $\bar{f}_a \preceq f$. Logo,

$\phi(a) = f_a(u) \preceq \bar{f}_a(u) \preceq f(u) = 1$. Mas isto contradiz o fato de que $\phi(a) \notin \Gamma(a)$, pois, trivialmente, $1 \in \Gamma(a)$.

Suponha, agora, que exista uma função viável f para o $\text{PVC}(D, r, \phi, \Gamma)$. Seja $X = \{a = uw \in AD : \phi(a) \preceq f(u)\}$. Mostraremos que X intercepta cada par em E' e que cada par de arcos em X pertence a E . Inicialmente, mostraremos que X intercepta cada par em E' . Suponha que exista algum arco $a = uw \in AD$ com $\phi(a) \notin \Gamma(a)$ tal que $a \notin X$ e $a^{-1} \notin X$. Pela construção de X , temos que $\phi(a) \not\preceq f(u)$ e $\phi(a)^{-1} \not\preceq f(w)$, ou seja, $\phi(a)$ não é o primeiro símbolo de $f(u)$ e $\phi(a)^{-1}$ não é o primeiro símbolo de $f(w)$ (lembre que $|\phi(a)| = 1$). Logo, $\phi(a)$ é um símbolo presente na palavra $f(u)^{-1} \cdot \phi(a) \cdot f(w)$. Como $\Gamma(a)$ é hereditário e $\phi(a) \notin \Gamma(a)$ então $f(u)^{-1} \cdot \phi(a) \cdot f(w) \notin \Gamma(a)$. Mas isto é um absurdo pois f é viável.

Mostraremos, agora, que cada par de arcos em X pertence a E . Sejam a' e a'' dois arcos em X . Mostraremos que $\{a', a''\} \in E$, ou seja, que $f' = \bar{f}_{a'} \vee \bar{f}_{a''}$ é pré-viável e finito. Como $a', a'' \in X$ e f é viável então $\bar{f}_{a'} \preceq f$ e $\bar{f}_{a''} \preceq f$. Logo, f' é finito e $f'(r) = 1$. Seja $a = uw \in AD$ tal que $y = f'(u)^{-1} \cdot \phi(a) \cdot f'(w) \notin \Gamma(a)$. Precisamos mostrar que $f'(u) = f'(w) = 1$. Como $\bar{f}_{a'}$ e $\bar{f}_{a''}$ são pré-viáveis então podemos assumir, por simetria, que $f'(u) = \bar{f}_{a'}(u)$ e $f'(w) = \bar{f}_{a''}(w)$, pois se tivéssemos $f'(u) = \bar{f}_{a'}(u)$ e $f'(w) = \bar{f}_{a'}(w)$ então $f'(u) = \bar{f}_{a'}(u) = 1$ e $f'(w) = \bar{f}_{a'}(w) = 1$. Podemos assumir, pelo Lema 4.1.8, que $f'(w) \not\preceq \phi(a)^{-1} \cdot f'(u)$. Suponha, inicialmente, $f'(u) \not\preceq \phi(a) \cdot f'(w)$. Neste caso, o primeiro símbolo de $f'(u)^{-1}$ e o último símbolos de $f'(w)$ são iguais ao primeiro e último símbolos de y , respectivamente. Como $f' \preceq f$ então y é um segmento de $f(u)^{-1} \cdot \phi(a) \cdot f(w)$. Como $f(u)^{-1} \cdot \phi(a) \cdot f(w) \in \Gamma(a)$, pois f é viável, e y é um segmento de $f(u)^{-1} \cdot \phi(a) \cdot f(w)$ então $y \in \Gamma(a)$. Isto contradiz a escolha que fizemos para o arco a e, portanto, $f'(u) \preceq \phi(a) \cdot f'(w)$.

Como $\bar{f}_{a''}(u) \preceq f'(u)$, $y \notin \Gamma(a)$ e $f'(u) \preceq \phi(a) \cdot f'(w)$ então $\bar{f}_{a''}(u)^{-1} \cdot \phi(a) \cdot f'(w) \notin \Gamma(a)$. Mas $f'(w) = \bar{f}_{a''}(w)$, e assim, obtemos que $\bar{f}_{a''}(u)^{-1} \cdot \phi(a) \cdot \bar{f}_{a''}(w) \notin \Gamma(a)$. Segue que $\bar{f}_{a''}(u) = \bar{f}_{a''}(w) = 1$. Como $f'(w) = \bar{f}_{a''}(w)$ então $f'(w) = 1$. Deste modo, $f'(u) \preceq \phi(a) \cdot f'(w) = \phi(a)$. Como $|\phi(a)| = 1$ e $f'(w) = 1$ então:

(i) $f'(u) = \phi(a)$ e, então, $y = f'(u)^{-1} \cdot \phi(a) \cdot f'(w) = \phi(a)^{-1} \cdot \phi(a) \cdot 1 = 1$ ou

(ii) $f'(u) = 1$ e, então, $y = 1 \cdot \phi(a) \cdot 1 = \phi(a)$.

Como $y \notin \Gamma(a)$ então $f'(u) = 1$.

□

Estamos agora em condições de apresentar o algoritmo que resolve o problema da viabilidade cohomológica. Como veremos, este problema será reduzido ao problema da 2-satisfatibilidade.

Algoritmo para o problema da viabilidade cohomológica:

Entrada: Um grafo orientado conexo D , um vértice $r \in VD$, uma função $\phi : AD \rightarrow G_k$ e para cada arco $a \in AD$ um subconjunto hereditário $\Gamma(a)$ de G_k .

Saída: Devolve uma função viável f , caso exista. Caso contrário, devolve.
'não'

1. Construa os conjuntos E e E' .
2. Faça $\overline{E} = (AD \times AD) - E$.
3. $\mathcal{F} = \prod_{\{a, a^{-1}\} \in E'} (x_a + x_{a^{-1}}) \cdot \prod_{\{b, b'\} \in \overline{E}} (\overline{x_b} + \overline{x_{b'}})$.
4. Se $\mathcal{F}$ for satisfatível então fixe uma atribuição de valores que satisfaz $\mathcal{F}$, devolva $f = \bigvee \overline{f_a}$, onde x_a é verdadeiro, e pare.
5. Se $\mathcal{F}$ não for satisfatível então responda que não existe função viável e pare.

Analisando cada um dos passos do algoritmo proposto, não é muito difícil obter o seguinte resultado.

Teorema 4.1.15 *O problema da viabilidade cohomológica pode ser resolvido por um algoritmo polinomial em $|AD| + \sigma + k$.*

□

4.2 PHF com padrão fixo específico

Nesta seção estudaremos o PHF com padrão P , onde P pertence à classe $\mathcal{IP}$ que consiste dos grafos formados por dois vértices e k arestas múltiplas ($k \geq 1$).

Em 1989, Suzuki, Akama e Nishizeki [21] apresentaram um algoritmo de complexidade $O(n \log n)$ que encontra o número máximo de caminhos disjuntos k_M entre dois vértices s e t de um dado grafo planar G . Estes autores obtiveram, ainda, um algoritmo de mesma complexidade que encontra k caminhos disjuntos entre dois vértices s e t de G , desde que $k \leq k_M$.

Recentemente, Ripphausen-Lipa, Wagner e Weihe apresentaram um algoritmo linear para o problema de encontrar uma coleção máxima de caminhos disjuntos de s a t em um grafo planar.

Estes algoritmos utilizam técnicas distintas e bastante interessantes. Ambos podem ser eficientemente implementados. Por este motivo, resolvemos apresentá-los neste trabalho.

4.2.1 Algoritmos baseado em divisão-e-conquista

Nesta subseção apresentamos dois algoritmos: o primeiro deles resolve o PHF com padrão P , onde $P \in \mathcal{IP}$, e o segundo encontra um vértice-homeomorfismo de P em um grafo de entrada G .

4.2.1.1 Problema do número máximo de caminhos disjuntos de s a t

Veremos aqui um algoritmo de complexidade $O(n \log n)$ devido a Suzuki, Akama e Nishizeki [21] para o seguinte problema:

Problema do número máximo de caminhos disjuntos de s a t

Entrada: Um grafo não-orientado planar G , imerso no plano, e dois vértices

$s, t \in VG$, $s \neq t$.

Saída: O número máximo, k_M , de caminhos disjuntos de s a t .

Figura 4.11: O grafo G está desenhado com linhas sólidas e o grafo G^+ com linhas tracejadas. O circuito C , indicado com linhas sólidas em negrito, separa s e t .

Assim, este algoritmo resolve o PHF com padrão P , para $P \in \mathcal{IP}$.

Determinar se $k_M = 1$ pode, facilmente, ser feito em $O(n)$. Deste modo, assumiremos que $k_M \geq 2$. Podemos também assumir que G é simples, s e t não são adjacentes em G e que o vértice t está na face externa de G .

Para resolver o problema acima, o algoritmo que veremos constrói vários grafos. Vejamos a primeira construção.

- **Construção de G^+** (a partir de G).

Seja G^+ o grafo obtido, a partir de G , colocando-se um novo vértice F^* em cada face F de G e ligando o vértice F^* a cada um dos vértices em VF (vértices na fronteira de F).

Veja Figura 4.11. Mais precisamente, o grafo G^+ é o grafo simples com

$$VG^+ = VG \cup (FG)^* \text{ e}$$

$$AG^+ = \{\{F^*, v\} : F^* \in (FG)^* \text{ e } v \in VF\},$$

onde $(FG)^* = \{F^* : F \text{ é face de } G\}$.

Diremos que um circuito C em G^+ **separa** u e v , sendo $u, v \in VG^+$, quando os vértices u e v ficam em regiões distintas de $\mathbb{R}^2 - \text{Im}(C)$, onde $\text{Im}(X)$ para $X \subseteq G$ é a região de $\mathbb{R}^2$ definida pelo desenho de X .

Veremos a seguir que o problema original pode ser reduzido a um outro problema sobre G^+ . Para isso, o seguinte teorema bastante conhecido em Teoria dos Grafos (veja [5]) será útil:

Teorema 4.2.1 *Sejam s e t dois vértices de um grafo não-orientado G que não são adjacentes. Existem k caminhos disjuntos de s a t em G se, e somente se, para qualquer corte X que separa s e t em G , temos $|X| \geq k$.*

□

Usando este teorema, é fácil provar o seguinte resultado.

Teorema 4.2.2 *Existem k caminhos disjuntos de s a t em G se, e somente se, todo circuito, em G^+ , que separa s e t tem comprimento no mínimo $2k$.*

Dem: Segue pelo Teorema 4.2.1 que é suficiente mostrarmos que existe em G um corte X que separa s e t com $|X| \leq k$ se, e somente se, existe, em G^+ , um circuito C que separa s e t tal que $|C| \leq 2k$.

Suponha, inicialmente, que existe, em G^+ um circuito que separa s e t com $|C| \leq 2k$. Note que

$$|VC \cap VG| = |C|/2 \leq k \quad \text{e} \quad VC \cap VG \text{ é um corte em } G \text{ que separa } s \text{ e } t.$$

Suponha, agora, que existe em G um corte X que separa s e t com $|X| \leq k$. Suponha, ainda, que X é minimal. Seja G' a componente de $G - X$ que contém s . Seja $G'' = G[VG' \cup$

$X]$. Os vértices de X estão na face externa de G'' . Assuma que quando percorremos a face externa de G'' no sentido horário encontramos os vértices $v_0, v_1, \dots, v_{|X|-1}, v_{|X|} = v_0$ nesta ordem. Claramente, v_i e v_{i+1} estão na fronteira de uma mesma face de G'' . Seja F_i a face de G'' separada da face externa pela aresta $v_i v_{i+1}$. Seja J uma curva simples no plano que satisfaz:

1. $J \cap Im(G) = Im(X)$.
2. O intervalo de J entre os vértices v_i e v_{i+1} está no interior da face F_i de G ($0 \leq i \leq |X| - 1$).

Observe que s e t estão em regiões distintas de $\mathbb{R}^2 - J$. Claramente, o circuito $C = (v_0, F_0^*, v_1, F_1^*, \dots, v_0)$ em G^+ separa s e t e $|C| = 2|X| \leq 2k$.

□

Corolário 4.2.3 *Em G o número máximo de caminhos disjuntos de s a t é a metade do comprimento de um circuito mínimo em G^+ que separa s e t .*

□

O Corolário 4.2.3 nos garante que para encontrarmos k_M é suficiente encontrarmos em G^+ o comprimento de um circuito mínimo que separa s e t .

Seja $P = (s, v_1, v_2, \dots, v_{r-1}, t)$ um caminho mínimo de s a t em G^+ . Claramente, qualquer circuito que separa s e t em G^+ passa através de algum vértice em $VP - \{s, t\}$. Deste modo, para encontrarmos o comprimento de um circuito mínimo que separa s e t em G^+ é suficiente encontrarmos o comprimento de um menor circuito que separa s e t e que passa por v_j , para cada v_j em $VP - \{s, t\}$.

Com este objetivo construiremos a partir de G^+ um novo grafo G^s .

- **Construção de G^s** (a partir de G^+ e P).

Inicialmente, trocamos cada vértice $v_j \in VP - \{s, t\}$ por dois novos vértices u_j e w_j . Em seguida, cada aresta $v_j v_{j+1}$ que está no caminho P é trocada por duas arestas

Figura 4.12: Grafo G^s ; caminhos P' e P'' na fronteira da face externa.

paralelas, no sentido geométrico, $u_j u_{j+1}$ e $w_j w_{j+1}$. Quando percorremos o caminho P de s para t , as arestas que incidiam em v_j pelo lado esquerdo, no grafo G^+ , passam a incidir em u_j no grafo G^s , e as arestas que incidiam em v_j pelo lado direito, no grafo G^+ , passam a incidir em w_j no grafo G^s . Finalmente, nós retiramos deste novo grafo os vértices s e t . Veja na Figura 4.12 um exemplo de um grafo G^s .

Claramente, um circuito mínimo em G^+ que separa s e t e passa por v_j corresponde a um caminho mínimo de u_j a w_j em G^s . Desta maneira, para encontrarmos k_M é suficiente encontrarmos um caminho mínimo de u_j a w_j , para cada $1 \leq j \leq r-1$, e obter o mínimo dos comprimentos destes caminhos.

Sejam P' e P'' os caminhos em G^s que correspondem naturalmente ao caminho P em G^+ . Para cada $1 \leq j \leq r-1$, seja C_j um caminho mínimo de u_j a w_j em G^s . Seja W um passeio de w_1 a u_1 na fronteira da face externa de G^s , no sentido anti-horário. Seja C o passeio fechado em G^s obtido concatenando o caminho C_j , a seção de $(P'')^{-1}$ que vai de w_j a w_1 , o passeio W e a seção de P' que vai de u_1 a u_j . A imagem do passeio C no plano juntamente com seu interior será chamado $\text{in}(C_j)$. Seja W' um passeio de w_{r-1} a u_{r-1} , na face externa de G^s , no sentido horário. Seja C' o passeio fechado obtido concatenando o caminho C_j , a seção de P'' que vai de w_j a w_{r-1} , o passeio W' e a seção de $(P')^{-1}$ que vai de u_{r-1} a u_j . A imagem de C' no plano juntamente com seu interior será chamada

$\text{out}(C_j)$.

Um subgrafo $\mathcal{F}$ de G^s é uma floresta de ligação mínima para (G^s, l, h) , se:

- $\mathcal{F}$ é uma floresta,
- $u_i, w_i, i = l, \dots, h$, são folhas de $\mathcal{F}$,
- Para cada $i = l, \dots, h$, existe um caminho de u_i a w_i em $\mathcal{F}$ que tem comprimento igual ao de um caminho mínimo de u_i a w_i em G^s .

O algoritmo que apresentaremos a seguir encontra uma floresta de ligação mínima para G^s , caso seja executado com entrada $(G^s, 1, r - 1)$. Nós usaremos esta floresta de ligação para encontrar o comprimento de um menor caminho entre os caminhos mínimos de u_j a w_j , para $j = 1, \dots, r - 1$, que como já vimos é igual a $2k_M$.

Algoritmo para encontrar uma floresta de ligação mínima

Entrada: Um grafo H e dois inteiros l e h .

Saída: Uma floresta de ligação mínima para (H, l, h) .

0. Faça $\mathcal{F}$ ser o grafo vazio.
1. Faça $j = \lfloor (l + h)/2 \rfloor$.
2. Encontre C_j em H (veja explicação abaixo).
Seja H_{in} o subgrafo de H que está na região $\text{in}(C_j)$ e seja H_{out} o subgrafo de H que está na região $\text{out}(C_j)$.
3. Se $V\mathcal{F} \cap VC_j = \emptyset$ então faça $\mathcal{F} = \mathcal{F} + C_j$, caso contrário encontre um subgrafo C_j' de C_j tal que $\mathcal{F} + C_j'$ é uma floresta que possui um caminho de u_j a w_j e o comprimento deste caminho é igual ao de um caminho mínimo de u_j a w_j em H . Faça $\mathcal{F} = \mathcal{F} + C_j'$. Veja abaixo como encontrar C_j' .
4. Se $l \leq j - 1$ então execute este algoritmo com entrada $(H_{in}, l, j - 1)$.
5. Se $j + 1 \leq h$ então execute este algoritmo com entrada $(H_{out}, j + 1, h)$.

6. Devolva $\mathcal{F}$ e pare.

Como encontrar o caminho C_j no passo 2 do algoritmo.

Na primeira chamada o algoritmo encontra dois caminhos C_1 e C_{r-1} . O caminho C_1 é o subgrafo de W que é um caminho de u_1 a w_1 e o caminho C_{r-1} é o subgrafo de W' que é um caminho de u_{r-1} a w_{r-1} . Considere uma chamada qualquer do algoritmo. Se $VC_h \cap VC_l = \emptyset$ então encontramos um caminho de u_j a w_j no subgrafo de G^s cuja imagem está na região $in(C_h) \cap out(C_l)$.

Se $VC_h \cap VC_l \neq \emptyset$ então seja u_j' o primeiro vértice em VC_h que encontramos ao percorrer o caminho C_l a partir de u_l e seja w_j' o primeiro vértice em VC_h que encontramos ao percorrer C_l^{-1} a partir de w_l . Como C_l e C_h são subgrafos de $\mathcal{F}$, que é uma floresta, então as seções destes caminhos entre u_j' e w_j' são idênticas. Deste modo, a seção de C_j entre estes dois vértices deve ser idêntica à seção de C_l (e C_h) entre estes dois vértices. Assim, para encontrar C_j é suficiente encontrar, no subgrafo de G^s cuja imagem está na região escura da Figura 4.13, um caminho mínimo Q_j' de u_j a u_j' , um caminho mínimo Q_j'' de w_j' a w_j e, finalmente, fazer $C_j = Q_j' \cdot C_l[u_j', w_j'] \cdot Q_j''$.

É fácil ver que os caminhos C_j , para cada $j = 1, \dots, r-1$, podem ser encontrados em tempo proporcional ao número de arestas de G^s .

Como obter o subgrafo C_j' no passo 3 do algoritmo.

Suponha que $V\mathcal{F} \cap VC_j \neq \emptyset$. Seja u' o primeiro vértice em $V\mathcal{F}$ que encontramos ao percorrer o caminho C_j a partir de u_j e seja w' o primeiro vértice em $V\mathcal{F}$ que encontramos ao percorrer o caminho C_j^{-1} a partir de w_j . Seja P_j' a seção de C_j entre os vértices u_j e u' e seja P_j'' a seção de C_j entre os vértices w' e w_j . Observe que u' e w' podem estar em VC_h ou VC_l , onde C_h e C_l são os caminhos de u_h a w_h e de u_l a w_l que estão em $\mathcal{F}$, respectivamente. Desta forma, precisamos analisar os seguintes casos:

1. $u', w' \in VC_h$;
2. $u', w' \in VC_l$;
3. $u' \in VC_h$ e $w' \in VC_l$;

Figura 4.13: Caso em que $VC_h \cap VC_l \neq \emptyset$. Neste caso apenas as arestas que estão na região escura são usadas para encontrar C_j .

Figura 4.14: O caminho C_j é mostrado em tracejado normal e o subgrafo C_j' em tracejado em negrito.

4. $u' \in VC_l$ e $w' \in VC_h$.

Se ocorre 1 ou 2 então faça $C_j' = P_j' + P_j''$ (veja Figura 4.14).

Suponha que ocorre o caso 3. Se $VC_h \cap VC_l \neq \emptyset$ então faça $C_j' = P_j' + P_j''$ (veja Figura 4.15). Assuma, então, que $VC_h \cap VC_l = \emptyset$. Seja x o primeiro vértice que encontramos em VC_l ao percorrer C_j a partir de u_j . Note que existe um tal vértice, pois $w' \in VC_l \cap VC_j$. Seja y o primeiro vértice de VC_h que encontramos ao percorrer C_j^{-1} a partir de x . Observe que u' é um vértice que precede x em C_j e está em $VC_j \cap VC_h$. Seja Q a seção de C_j que vai de x a y . Faça $C_j' = P_j' + P_j'' + Q$ (veja Figura 4.16). É fácil ver que o caminho de u_j a w_j em $F + C_j'$ tem comprimento igual ao de um caminho mínimo de u_j a w_j em G^s . Finalmente, se ocorre o caso 4, C_j' pode ser encontrado de modo similar.

Figura 4.15: Caso 3 quando $VC_h \cap VC_l \neq \emptyset$ (similar aos casos 1 e 2). Note que o subgrafo C_j' , mostrado em tracejado em negrito, é o mesmo.

Figura 4.16: Caso 3 quando $VC_h \cap VC_l = \emptyset$. Neste caso, $C_j' = P_j' + P_j'' + Q$.

É fácil ver que o passo 3 do algoritmo pode ser feito em $O(n)$. Assim uma árvore de ligação mínima pode ser encontrada em $O(n \log r)$, ou seja, em $O(n \log n)$.

A partir de uma floresta de ligação mínima $\mathcal{F}$, encontraremos o comprimento de um caminho mínimo entre u_i e w_i em G^s , para $i = 1, \dots, r - 1$.

O problema seguinte é bastante conhecido:

Problema do ancestral comum mais próximo

Entrada: Uma árvore T , um vértice $r \in VT$, que é a raiz da árvore T ,
e $r - 1$ pares de vértices $(u_1, w_1), \dots, (u_{r-1}, w_{r-1})$.

Saída: O ancestral comum mais próximo de u_i e w_i , para $i = 1, \dots, r - 1$.

Como $u_i \neq w_j$, para $i, j = 1, \dots, r - 1$, então o problema do ancestral comum mais próximo pode ser resolvido em $O(n \log n)$. Como $\mathcal{F}$ é uma floresta que possui um caminho entre u_i e w_i , então u_i e w_i estão em uma mesma árvore em $\mathcal{F}$. Fixe uma raiz em cada árvore em $\mathcal{F}$ e execute um algoritmo para resolver o problema do ancestral comum mais próximo para cada árvore em $\mathcal{F}$.

Seja a_i o ancestral comum mais próximo de u_i e w_i . O comprimento do caminho em $\mathcal{F}$ que vai de u_i a w_i é dado por

$$\text{dist}(u_i) + \text{dist}(w_i) - 2 \cdot \text{dist}(a_i),$$

onde $\text{dist}(u)$, $u \in V\mathcal{F}$, indica o comprimento do caminho de u à raiz da árvore em que está o vértice u . Claramente, $\text{dist}()$ pode ser encontrado em $O(n)$, para todo vértice em $\mathcal{F}$ simultaneamente. Como o comprimento do caminho de u_i a w_i em $\mathcal{F}$ é igual ao comprimento de um circuito mínimo que passa por v_i em G^+ então segue pelos comentários feitos após o Corolário 4.2.3 o seguinte teorema.

Teorema 4.2.4 *Dado um grafo planar G e dois vértices $s, t \in VG$, o número máximo k_M de caminhos disjuntos de s a t pode ser encontrado em $O(n \log n)$.*

□

Seja G o grafo de entrada para o PHF com padrão P , $P \in \mathcal{IP}$. Seja m o vértice-mapeamento definido por:

$$m(s') = s, \quad m(t') = t,$$

onde s', t' são os vértices de P .

Claramente, P é m -vértice-homeomorfo a G se, e somente se, $k \leq k_M$.

4.2.1.2 Problema dos k caminhos disjuntos de s a t

Suzuki, Akama e Nishizeki [21] apresentaram um algoritmo de complexidade $O(n \log n)$ para o seguinte problema:

Problema dos k caminhos disjuntos de s a t

Entrada: Um grafo não-orientado planar G , imerso no plano, dois vértices $s, t \in VG$, $s \neq t$, e um inteiro k , $k \leq k_M$.

Saída: k caminhos disjuntos de s a t em G .

Assim, se $P \in \mathcal{IP}$ e G é um grafo planar este algoritmo encontra um vértice homeomorfismo de P em G em tempo $O(n \log n)$. Este algoritmo utiliza um algoritmo auxiliar que passaremos a apresentar.

Inicialmente, iremos definir alguns termos que usaremos para definir o problema com que trabalharemos. Seja F uma face de um grafo G . Sejam $M_1, \dots, M_k$ subconjuntos de VF dois a dois disjuntos. Diremos que o conjunto $\mathcal{M} = \{M_1, \dots, M_k\}$ é uma **malha**. Os vértices que estão em $\bigcup_{i=1}^k M_i$ serão ditos **terminais**. Uma **rede** será um par ordenado $\mathcal{R} = (G, \mathcal{M})$ em que o primeiro elemento é um grafo não-orientado e o segundo elemento, uma malha.

Seja $\mathcal{R} = (G, \{M_1, \dots, M_k\})$ uma rede. Seja T_i um subgrafo de G que é uma árvore. Diremos que T_i **conecta** M_i se os terminais de M_i são as folhas de T_i . Uma **floresta de conexão** $\mathcal{F}$ para $\mathcal{R}$ é um subgrafo de G que é uma floresta e cujos componentes $T_1, \dots, T_k$ são tais que T_i conecta M_i , $i = 1, \dots, k$, e $VT_i \cap VT_j = \emptyset$, para quaisquer i, j , com $i \neq j$. Escreveremos, $\mathcal{F} = T_1 + \dots + T_k$.

Figura 4.17: Os vértices numerados são os terminais. O número indica o conjunto da malha em que está o respectivo terminal. O subgrafo em negrito é uma floresta de conexão.

Podemos, agora, enunciar o seguinte problema:

Problema da Floresta de Conexão PFC

Entrada: Uma rede $\mathcal{R} = (G, \mathcal{M})$.

Pergunta: Existe uma floresta de conexão para $\mathcal{R}$?

Em caso afirmativo, encontrar uma tal floresta de conexão.

Para resolver o PFC podemos assumir, sem perda de generalidade, que os terminais são vértices da fronteira da face externa de G . A face externa de G será denotada por E . Além disso, podemos assumir que G é simples.

Seja J uma curva simples no plano, que passa por todos os vértices que estão na fronteira de E e contém em seu interior a imagem de G no plano. Seja v_0 um vértice qualquer em VE . O primeiro e o último vértice de M_i que são encontrados quando percorremos J a partir de v_0 no sentido horário serão chamados **vértice inicial** e **vértice final** de M_i , respectivamente. Denotaremos estes vértices por $s(M_i)$ e $t(M_i)$, respectivamente.

Na Figura 4.17 os vértices $s(M_i)$ e $t(M_i)$, $i = 1, \dots, 7$, estão marcados com um círculo.

Figura 4.18: A figura mostra a construção de P_2 a partir de W_2 . As arestas em tracejado em negrito são as arestas que foram retiradas de W_2 .

Assumiremos que as arestas na lista de adjacência de cada vértice dos grafos com que trabalharemos estão ordenadas com base na curva J . Mais precisamente, a primeira aresta na lista de adjacência para um vértice v será a primeira aresta adjacente a v que encontramos quando giramos, no sentido horário, um segmento qualquer vv' da curva J .

Denotaremos por $W(M_i)$ o passeio de $s(M_i)$ a $t(M_i)$ na fronteira de E , que não cruza com J e cuja primeira aresta é a primeira aresta na lista de adjacência para $s(M_i)$ e cuja última aresta é a última aresta na lista de adjacência para $t(M_i)$.

Em seguida, apresentamos um algoritmo para resolver o PFC.

Algoritmo para resolver o PFC

Entrada: Uma rede $\mathcal{R} = (G, \mathcal{M})$.

Saída: Devolve, se existir, uma floresta de conexão para $\mathcal{R}$.

Caso contrário, responde que não existe uma tal floresta.

1. Modifique a numeração das malhas de $\mathcal{M}$ de modo que ao percorrermos a curva J a partir de v_0 no sentido horário encontremos os vértices finais desta malha na seguinte ordem: $t(M_1), \dots, t(M_k)$.
2. Para cada $i = 1, \dots, k$ faça:
 - 2.1. Se os terminais de M_i estão em componentes distintos de G ou existe algum terminal de M_j , $j \neq i$, em $W(M_i)$ então responda que **não** existe floresta de conexão para $\mathcal{R}$ e pare.
 - 2.2. Construa, a partir de $W(M_i)$, uma árvore T_i que conecta os terminais de M_i (basta retirar vértices e arestas de $W(M_i)$ de modo que T_i não possua nenhum circuito ou vértice com grau um que não seja terminal).
 - 2.3. Faça $G = G - VW(M_i)$.
3. Devolva $\mathcal{F} = T_1 + \dots + T_k$ e pare.

Observe que $W(M_i)$ pode, inicialmente, conter circuitos e vértices com grau um que não são terminais.

Teorema 4.2.5 *Se uma rede $\mathcal{R} = (G, \mathcal{M})$ contém uma floresta de conexão, então o algoritmo para o PFC encontra uma tal floresta.*

Dem: Seja $\mathcal{M} = \{M_1, \dots, M_k\}$. Nós denotaremos por $\mathcal{R}_i = (G_i, \mathcal{M}_i)$ a rede no início da i -ésima iteração do algoritmo, onde $\mathcal{M}_i = (M_i, \dots, M_k)$. Evidentemente, $\mathcal{M}_1 = \mathcal{M}$ e $G_1 = G$.

Suponha que exista uma floresta de conexão $\mathcal{F}_i = T_i + T_{i+1} + \dots + T_k$ em $\mathcal{R}_i$ ($1 \leq i \leq k-1$). Vamos mostrar que existe uma floresta de conexão em $\mathcal{R}_{i+1}$.

Suponha que não exista uma floresta de conexão em $\mathcal{R}_{i+1}$. Assim, algum T_j ($i+1 \leq j \leq k$) na floresta $\mathcal{F}_i$ de $\mathcal{R}_i$ intercepta $W(M_i)$, pois $G_{i+1} = G_i - VW(M_i)$ e $\mathcal{M}_{i+1} = \mathcal{M} - \{M_i\}$.

Seja $v \in VW(M_i) \cap VT_j$. Existe em T_j um caminho P_j de v a $t(M_j)$. Na árvore T_i ,

Figura 4.19: No desenho a linha tracejada representa a curva J . O desenho ilustra a posição relativa dos vértices v , $t(M_i)$, $s(M_i)$, $t(M_j)$.

existe um caminho P_i que vai de $s(M_i)$ a $t(M_i)$. Como $j > i$, então o vértice $t(M_j)$ é encontrado quando percorremos a curva J no sentido horário entre $t(M_i)$ e $s(M_i)$. Como $v \in W(M_i)$ então encontramos v entre $s(M_i)$ e $t(M_i)$ quando percorremos a fronteira da face externa de G_i . Logo, quando percorremos a fronteira da face externa de G_i , a partir de $s(M_i)$, os vértices $s(M_i)$, v , $t(M_i)$, $t(M_j)$ aparecem nesta ordem (veja Figura 4.19). Logo os caminhos P_i e P_j não são disjuntos nos vértices contradizendo, o fato de que $\mathcal{F}_i = T_i + \dots + T_j + \dots + T_k$ é uma floresta de conexão em $\mathcal{R}_i$.

□

Agora estamos em condições de apresentar o algoritmo de Suzuki, Akama e Nishizeki [21] para o problema dos k caminhos disjuntos.

Inicialmente, daremos uma idéia geral do algoritmo. Seja $X \subset VG$ um corte mínimo de vértices em G que separa s e t . Seja J uma curva fechada simples tal que $J \cap Im(G) = Im(X)$.

Seja G_{in} o subgrafo de G cuja imagem no plano está no interior de J e seja G_{out} o subgrafo de G cuja imagem no plano está no exterior de J . Seja G_i o grafo que obtemos ao adicionar a G_{in} um vértice t' e uma aresta entre cada vértice de X e t' e seja G_o o grafo que obtemos ao adicionar a G_{out} um vértice s' e uma aresta entre cada vértice

Figura 4.20: Concatenando os caminhos representados por linhas pontilhadas nos grafos G_i e G_o obtemos k caminhos disjuntos de s a t em G .

de X e s' , como mostrado na Figura 4.20. Encontre k caminhos disjuntos entre s e t' em G_i e k caminhos disjuntos entre s' e t em G_o . Concatenando seções apropriadas dos k primeiros caminhos e dos k últimos caminhos obtemos k caminhos disjuntos de s a t em G , como mostramos na Figura 4.20. Para que o método descrito acima funcione é necessário que em todo grafo G haja um corte de vértices X que separa s e t tal que $|X| = k$. Claramente, isto nem sempre é verdade. O algoritmo que apresentaremos segue a seguinte estratégia: se não existir um tal corte de vértices no grafo G então o algoritmo irá retirar de G algumas arestas e vértices redundantes de maneira que no novo grafo haja um tal corte de vértices.

Lema 4.2.6 *Seja $v \in VG^+$ um vértice que não é adjacente a s ou t em G^+ . Seja G_v^+ o grafo obtido a partir de G^+ contraindo-se toda aresta incidente em v . Se C é um circuito mínimo em G^+ que separa s e t e passa por v então, o circuito C_v , que é o correspondente a C em G_v^+ , é um circuito mínimo em G_v^+ que separa s e t e passa por v . Além disso, $|C_v| = |C| - 2$.*

□

Lema 4.2.7 *Suponha que exista em G^+ um vértice $v \in VG^+$ que não é adjacente a s ou t . Seja G' o grafo obtido a partir de G , aplicando-se uma das seguintes operações:*

1. *Se $v \in VG$ então faça $G' = G - v$.*
2. *Se $v \in (FG)^*$, ou seja, v é um vértice que corresponde a uma face F de G , então contraia todos os vértices que estão na fronteira de F a um único vértice.*

Se o comprimento de um circuito mínimo em G^+ que passa por v e separa s e t é no mínimo $2k + 2$, então em G existem k caminhos disjuntos de s a t se, e somente se, em G' existem k caminhos disjuntos de s a t .

Dem: Observe que $(G')^+$ é o grafo obtido a partir de G^+ contraindo-se todas as arestas incidentes em v . Suponha, inicialmente, que no grafo G existem k caminhos disjuntos de s a t . Seja C' um circuito mínimo que separa s e t em $(G')^+$. Se C' passa por v então o circuito C em G^+ , que corresponde a C' , também passa por v e separa s e t . Por hipótese, $|C| \geq 2k + 2$. Segue pelo Lema 4.2.6 que $|C'| \geq 2k$. Se C' não passa por v então C' é um circuito que separa s e t em G^+ . Como em G existem k caminhos disjuntos de s a t , então o Teorema 4.2.2 nos garante que $|C'| \geq 2k$. Em qualquer um dos casos C' é um circuito em $(G')^+$ que separa s e t e $|C'| \geq 2k$. Segue, pelo Teorema 4.2.2, que em G' existem k caminhos disjuntos de s a t .

Suponha, agora, que em G' existem k caminhos disjuntos de s a t .

Considere k caminhos disjuntos de s a t em G' . Se nenhum destes caminhos passa por v então estes k caminhos também são caminhos em G . Suponha que um destes caminhos passe por v . Se $v \in VG$ então adicionando ao caminho que passa por v as

arestas incidentes em v obtemos k caminhos de disjuntos s a t em G . Se $v = F^* \in (FG)^*$, então adicionando ao caminho que passa por v arestas da fronteira da face F , obtemos k caminhos disjuntos de s a t em G .

Algoritmo dos k caminhos disjuntos de s a t

Entrada: Um grafo não-orientado G e dois vértices $s, t \in VG, s \neq t$.

Saída: k caminhos disjuntos de s a t .

1. Faça $\overline{G} = G$.
2. Construa G^+ .
3. Encontre um caminho mínimo de s a t no grafo G^+ . Seja P este caminho.
4. Se $|P| > 4$ faça o seguinte:
 - 4.1. Encontre o vértice v que está no meio de P .
 - 4.2. Encontre um circuito mínimo em G^+ que passa por v e separa s e t . Seja C este circuito.
 - 4.3. Enquanto $|C| > 2k$ e $|P| > 4$ faça o seguinte:
 - 4.3.1. Contraia todas as arestas que incidem em v no grafo G^+ .
 - 4.3.2. Atualize G, P e C .
5. Se $|P| \leq 4$ então encontre k caminhos disjuntos de s a t usando o procedimento que descreveremos adiante.
6. Se $|P| > 4$ e $|C| = 2k$ então faça o seguinte:
 - 6.1. Construa a partir de $X = VG \cap VC$ os grafos G_i e G_o .
 - 6.2. Execute este algoritmo com entrada (G_i, s, t') .
 - 6.3. Execute este algoritmo com entrada (G_o, s', t) .

- 6.4. Obtenha a partir dos $2k$ caminhos disjuntos construídos em G_i e G_o , k caminhos disjuntos de s a t no grafo G , conforme discutimos no início desta parte.
7. Se algum dos k caminhos construídos em G (pelo passo 6.4) passa por algum vértice u que foi reduzido ou que teve em algum passo suas arestas incidentes contraídas, então use as arestas da fronteira da face em que está o vértice u no grafo $\overline{G}$ ou as arestas que incidem em u em $\overline{G}$, respectivamente, para construir k caminhos de s a t no grafo original $\overline{G}$.

Figura 4.21: Caso em que $g_s = g_t = k$ e $|P| = 2$. Note que todos os terminais estão na face externa de G' .

A seguir explicaremos como obter os k caminhos de uma maneira direta no caso em que $|P| \leq 4$. O método que apresentamos é baseado no algoritmo para encontrar uma floresta de conexão que apresentamos na seção anterior. Seja g_s o grau do vértice s em G e seja g_t o grau do vértice t em G . Sejam $s_1, \dots, s_{g_s}$ os vértices adjacentes a s , ordenados no sentido horário a partir do caminho P . Sejam $t_1, \dots, t_{g_t}$ os vértices adjacentes a t , ordenados no sentido horário a partir do caminho P , como mostram as Figuras 4.21 e 4.22. Seja $G' = G - \{s, t\}$. A maneira como obteremos os k caminhos como desejamos difere em função dos graus de g_s e g_t e do comprimento de P . Temos as seguintes possibilidades:

Figura 4.22: Caso em que $g_s = g_t = k$ e $|P| = 4$. Em G' , os terminais estão em faces que estão separadas pelo vértice v .

caso 1: $g_s = g_t = k$.

caso 1.1 $|P| = 2$.

Neste caso, os vértices s e t estão em uma mesma face no grafo G . Claramente, no grafo G existem k caminhos disjuntos de s a t se, e somente se, existe uma floresta de conexão para a rede $\mathcal{R} = (G', \{\{s_i, t_i\} : 1 \leq i \leq k\})$. Veja Figura 4.21. Encontre uma floresta de conexão para $\mathcal{R}$, usando o algoritmo apresentado na seção anterior. A partir desta floresta podemos facilmente obter k caminhos disjuntos de s a t em G .

caso 1.2: $|P| = 4$.

Neste caso existe um único vértice de VG em $VP - \{s, t\}$. Seja v este vértice. Veja Figura 4.22. Seja A_e (A_d) o conjunto das arestas no grafo G' que incidem em v pela esquerda (direita) quando percorremos o caminho P de s para t . Seja G'' o grafo obtido a partir de G' substituindo o vértice v por dois vértices v_e e v_d e fazendo as arestas que estão em A_e incidir em v_e e as arestas que estão em A_d incidir em v_d . Veja Figura 4.23. No grafo G , os k caminhos disjuntos de s a t podem não passar por v ou então os vértices:

1. s_1, v e t_1 ou

Figura 4.23: Os terminais estão na face externa de G'' .

2. s_k, v e t_1 ou
3. s_k, v e t_k ou
4. s_1, v e t_k

aparecem em um dos caminhos nesta ordem. Considere as seguintes redes:

- $\mathcal{R}_1 = (G'' - A_e, \{\{s_i, t_i\} : 1 \leq i \leq k\})$;
- $\mathcal{R}_2 = (G'' - A_d, \{\{s_i, t_i\} : 1 \leq i \leq k\})$;
- $\mathcal{R}_3 = (G'', \{\{s_{i-1}, t_i\} : 2 \leq i \leq k\} \cup \{\{s_k, v_e\}, \{v_d, t_1\}\})$;
- $\mathcal{R}_4 = (G'', \{\{s_i, t_{i-1}\} : 2 \leq i \leq k\} \cup \{\{v_e, t_k\}, \{s_1, v_d\}\})$.

Claramente, existem k caminhos disjuntos de s a t em G se, e somente se, existe uma floresta de conexão para no mínimo uma das redes $\mathcal{R}_1, \mathcal{R}_2, \mathcal{R}_3$ ou $\mathcal{R}_4$. Usando o algoritmo descrito no início desta parte encontramos uma tal floresta de conexão. A partir desta floresta, encontramos k caminhos disjuntos de s a t no grafo G .

caso 2: $g_s \neq k$ ou $g_t \neq k$.

Seja $S = \{s_i : 1 \leq i \leq g_s\}$ e seja $T = \{t_i : 1 \leq i \leq g_t\}$.

caso 2.1: $|P| = 2$

Existem k caminhos disjuntos que vão de algum vértice de S até algum vértice de T . Podemos assumir, sem perda de generalidade, que um deles vai de s_1 a t_1 . Seja W_1 um passeio de s_1 a t_1 que passa apenas por arestas que estão na face externa de G' (W_1 é obtido da mesma forma que P_i no algoritmo para encontrar uma floresta de conexão). Podemos obter a partir de W_1 um caminho P_1 de s a t . Faça $G_1 = G - VW_1$, $S_1 = S - VW_1$ e $T_1 = T - VW_1$. Seja $i_2 = \min\{i : s_i \in S_1\}$ e seja $j_2 = \min\{j : t_j \in T_1\}$. Então existem $k - 1$ caminhos disjuntos que vão de algum vértice de S_1 a algum vértice de T_1 . Seja W_2 um passeio de s_{i_2} a t_{j_2} em G_1 que passa apenas por arestas que estão na sua face externa. Podemos obter a partir de W_2 um caminho P_2 de s a t no grafo G . Claramente, P_1 e P_2 são disjuntos. Repetindo este procedimento nós podemos encontrar k caminhos disjuntos de s a t no grafo G .

caso 2.2: $|P| = 4$.

Procedendo de modo análogo ao caso 2.1 para cada uma das redes $\mathcal{R}_1, \mathcal{R}_2, \mathcal{R}_3$ ou $\mathcal{R}_4$ encontraremos k caminhos disjuntos de s a t em G .

Analisaremos, agora, a complexidade do algoritmo acima para encontrar k caminhos disjuntos de s a t em um grafo planar G . O grafo G^+ pode facilmente ser obtido em tempo $O(n)$. O circuito C pode ser obtido a partir de um caminho mínimo de u a w no grafo G^s , como foi feito no algoritmo para encontrar uma floresta de ligação mínima. A atualização de G , P e C é proporcional ao número de arestas removidas. Desta forma o algoritmo, a menos da parte recursiva, gasta tempo $O(n)$. Indicaremos por $T(n, r)$ a complexidade do algoritmo para encontrar k caminhos disjuntos de s a t , sendo r o comprimento de um caminho mínimo de s a t em G^+ . Temos que

$$T(n, r) \leq T(n_i, r_i) + T(n_o, r_o) + O(n), \text{ para } r > 4.$$

$$T(n, 4) = O(n),$$

$$T(n, 2) = O(n).$$

Aqui n_i e n_o indicam o número de vértices de G_i e G_o , respectivamente; r_i indica o comprimento de um caminho mínimo de s a t' em G_i ; e r_o , o comprimento de um caminho mínimo de s' a t em G_o .

Assim, a complexidade do algoritmo é $O(n \log n)$.

Teorema 4.2.8 *O problema dos k caminhos disjuntos de s a t para grafos planares pode ser resolvido em tempo $O(n \log n)$.*

□

4.2.2 Algoritmo baseado em busca em profundidade

Recentemente Ripphausen-Lipa, Wagner e Weihe [17] apresentaram um algoritmo linear para o seguinte problema:

Problema de Menger - $PM(G, s, t)$

Entrada: Um grafo não-orientado planar G imerso no plano e dois vértices distintos $s, t \in VG$.

Saída: Uma coleção máxima de caminhos disjuntos de s a t .

Claramente, este algoritmo resolve o PHF com padrão P , para $P \in \mathbb{P}$.

Note que o problema de Menger pode ser resolvido em tempo $O(n \log n)$ utilizando os algoritmos apresentados na seção anterior. Nesta seção apresentaremos o algoritmo de Ripphausen-Lipa, Wagner e Weihe.

Assumiremos que as arestas serão representadas através de uma lista de incidência ordenada, no sentido anti-horário, para cada vértice. Cada item da lista para um vértice v será um arco que sai de v . Assumiremos que a lista para o vértice s é $e_1, \dots, e_k$. Assumiremos, sem perda de generalidade, que t é um vértice que está na face externa de G . Nesta seção diremos, simplesmente, grafo para nos referirmos a grafo orientado, planar, imerso no plano.

Antes de apresentarmos o algoritmo para o $PM(G, s, t)$, daremos uma idéia de como o mesmo funciona. Seja (G, s, t) a entrada para o problema de Menger. O algoritmo constrói, a partir de G , um grafo orientado D , trocando cada aresta $\{v, w\}$, onde v e w são distintos de s e t , pelos arcos (v, w) e (w, v) , cada aresta $\{s, v\}$ pelo arco (s, v) e cada aresta $\{v, t\}$ pelo arco (v, t) . O algoritmo constrói em D uma coleção máxima de caminhos disjuntos de s a t e a partir desta coleção obtém em G uma coleção máxima de caminhos disjuntos de s a t . O algoritmo é iterativo e na i -ésima iteração tenta construir um caminho de s a t em D cujo primeiro arco é e_i fazendo uma busca em profundidade mais à direita, isto é, uma busca em profundidade em que para cada vértice v os arcos que saem de v são utilizados no sentido anti-horário, com exceção do reverso do arco que entra em v na árvore de busca. Além disso, sempre que for feito um *backtracking*, removeremos o arco utilizado para fazê-lo. Deste modo, em cada momento, a árvore de busca consistirá, simplesmente, de um **caminho de busca**, a menos do instante em que ocorre um conflito próprio (conforme explicação abaixo). Seja $(s, v_1, v_2, \dots, v_{i-1} = u, v_i = v)$ o caminho de busca em um certo instante da execução do algoritmo. O vértice v é dito **vértice-líder** e o arco uv , **arco líder**. Na lista de incidência para v o primeiro arco depois do reverso do arco líder é o **arco-de-continuação**. Se o vértice v já está no caminho de busca ou em algum dos caminhos de s a t construídos nas iterações anteriores então dizemos que ocorreu um **conflito**. O algoritmo permite três tipos de conflito:

1. **Conflito Próprio à Esquerda - CPE**: o vértice v está no caminho de busca e o arco-líder atinge o caminho de busca pelo lado esquerdo do caminho.
2. **Conflito Não-próprio à Esquerda - CNE**: v está em algum caminho de s a t construído pelo algoritmo e o arco-líder atinge este caminho pelo lado esquerdo deste caminho.
3. **Conflito Não-próprio à Direita - CND**: v está em algum caminho de s a t construído pelo algoritmo e o arco-líder atinge este caminho pelo lado direito deste caminho.

Assim que ocorre um destes conflitos o algoritmo atua no sentido de eliminá-los. Vejamos os tratamentos feitos pelo algoritmo em cada caso:

- **tratamento para CPE e CNE**: o algoritmo executa um *backtracking*. Lembre-se que ao ser feito um *backtracking* o arco-líder é retirado do grafo.

- **tratamento para CND:** Sejam P e Q as seções do caminho atingido que vão de s a v e de v a t , respectivamente. O algoritmo concatena o caminho de busca com Q formando um novo caminho de s a t . Em seguida, P passa a ser o novo caminho de caminho de busca. Note que, agora, ocorre um CNE. O algoritmo trata, então, este conflito como vimos acima.

O algoritmo evita que ocorra **Conflito Próprio à Direita - CPD**, ou seja, que o vértice-líder v já esteja no caminho de busca e o arco líder atinja o caminho de busca pelo lado direito. Com este propósito, o algoritmo mantém um *contador global* e um *contador local* para cada vértice. O contador global é incrementado de uma unidade sempre que se tem um novo caminho de busca, ou seja, no começo de cada iteração e sempre que ocorre um CND. O contador local de um vértice w assume o valor do contador global sempre que w é o vértice-líder. Quando o algoritmo toma um arco vw para prosseguir a busca, é feita a comparação do contador local de v com o de w . Se forem iguais, vw é removido e é escolhido um novo arco-de-continuação. Não é óbvio que desta maneira não ocorrerão conflitos próprio à direita. Mais tarde, provaremos que, de fato, estes conflitos são evitados pelo algoritmo.

Na Figura 4.24 são mostrados os caminhos construídos ao se executar o algoritmo.

A Figura 4.25 ilustra a situação em que ocorre um conflito à esquerda. Note que os arcos do circuito envolvido neste conflito são retirados do grafo por *backtracking* e o arco vw é retirado porque os contadores locais de v e w são iguais. Caso contrário, ocorreria um conflito à direita. Observe que o tratamento deste conflito por *backtracking* resultaria em um grafo que não contém dois caminhos de s a t disjuntos nos vértices, como o grafo inicial.

Algoritmo para o problema de Menger

Entrada: Um grafo G e dois vértices distintos $s, t \in VG$.

Saída: Uma coleção máxima de caminhos disjuntos de s a t .

0. Construa o grafo D , conforme explicamos no início desta seção.

1. $cont_global = 0$.

Figura 4.24: Caminhos construídos pelo algoritmo.

Figura 4.25: Conflito à esquerda.

2. Para $i = 1$ até k faça:

2.1. Comece o caminho de busca pelo arco e_i .

2.2. $cont_global = cont_global + 1$.

2.3. Enquanto o vértice líder não é s ou t faça:

2.3.1. $cont_local(\text{vértice-líder}) = cont_global$.

2.3.2. Se ocorrer algum conflito execute o respectivo tratamento. No caso de um CND faça $cont_global = cont_global + 1$. Vá para 2.3.

2.3.3. Se o arco-de-continuação é o reverso do arco-líder ou se a lista de adjacência do vértice v está vazia então execute um *backtracking* e vá para 2.3.

2.3.4. Seja a o arco-de-continuação. Se o $cont_local$ nos extremos de a são iguais remova a , senão continue a busca por a (a passa a ser o novo arco-líder). Vá para 2.3.

3. Encontre os caminhos de s a t em G que correspondem aos caminhos encontrados em D . Devolva estes caminhos e pare.

Quando o algoritmo executa um *backtracking* dizemos que o vértice-líder foi **liberado**.

Seja $a_1, \dots, a_m$ a seqüência dos arcos que foram retirados ao terminar a execução do algoritmo. Seja $D_0 = D$ e $D_i = D_{i-1} - \{a_i\}$, para $i = 1, \dots, m$. Chamaremos o grafo D_0 de **grafo original**. Chamaremos D_i de **grafo atual** quando estivermos considerando a i -ésima iteração do algoritmo.

Note que:

1. No grafo atual não existe arco saindo pelo lado direito de um caminho construído pelo algoritmo (exceto, possivelmente, o reverso de algum arco do caminho), pois o arco-de-continuação é o primeiro arco à direita do arco-líder. No grafo original pode haver arcos que saem pelo lado direito de algum destes caminhos.
2. Seja uv o arco-líder num determinado instante da execução do algoritmo. Se v é liberado então todos os arcos que saíam de v no grafo original foram retirados, com exceção, eventualmente, do reverso de uv .

Diremos que um arco vw é **especial** se, em algum momento durante a execução do algoritmo, vw é retirado do grafo porque os contadores locais de v e w são iguais e vw não seria retirado na próxima iteração do algoritmo por causa de um conflito.

Lema 4.2.9 *Seja $v \in VD$ e a_1 e a_2 dois arcos distintos que entram em v . Assuma que a_2 esteve no caminho de busca em algum momento depois que a_1 esteve. Então quando a_2 esteve no caminho de busca o reverso de a_2 já tinha sido retirado de D .*

Dem: Se v foi liberado entre os momentos em que a_1 e a_2 estiveram no caminho de busca então no momento em que a_2 esteve no caminho de busca não havia nenhum arco saindo de v , exceto possivelmente o reverso de a_1 . Em particular, o reverso de a_2 já tinha sido retirado do grafo original. Caso v não tenha sido liberado entre estes dois momentos, então ocorreu um CND em v e o caminho foi atingido pelo arco a_2 . O reverso de a_2 é um arco que sai pelo lado direito deste caminho e, portanto, já tinha sido retirado do grafo original neste momento.

□

Lema 4.2.10 *Existe uma solução do $PM(D, s, t)$ que usa apenas arcos de D_m .*

Dem: A demonstração será feita por indução sobre i , $i = 0, \dots, m$. Para $i = 0$ a afirmação é, trivialmente, verdadeira. Assuma que a afirmação é válida para $i - 1$, com $i > 0$. Existem duas possibilidades:

caso 1: $a_i = vw$ não é especial.

Podemos assumir que a_i é retirado de D_{i-1} por causa de um conflito à esquerda, próprio ou não; caso contrário, nenhum caminho de s a t em D_{i-1} pode conter a_i . Seja $\mathcal{P}$ uma solução para o $\text{PM}(D_{i-1}, s, t)$. Podemos assumir que algum caminho P em $\mathcal{P}$ passa por a_i , caso contrário $\mathcal{P}$ é uma solução para o $\text{PM}(D_i, s, t)$. Seja P_1 a seção de P entre s e w e P_2 a seção de P entre w e t . Considere o instante anterior à retirada de a_i . Seja Q o caminho que passava por w neste instante e Q_1 a seção de Q entre s e w e Q_2 a seção de Q entre w e o último vértice de Q (que será t se Q não for o caminho de busca). Temos as seguintes possibilidades:

caso 1.1: w é o único vértice comum entre os caminhos P_1 e Q .

Como não existe arco saindo pelo lado direito de Q , então se algum caminho P' em $\mathcal{P}$, distinto de P , passa por um dos vértices internos de Q_1 , ele deve sair pelo lado esquerdo de Q_1 . Como P_1 separa o lado esquerdo de Q_1 de t e P' e P são disjuntos, então P' não pode alcançar t . Mas isto é um absurdo. Segue que nenhum caminho em $\mathcal{P}$ intercepta Q_1 . Seja $P'' = Q_1 \cdot P_2$. Substituindo P por P'' na coleção $\mathcal{P}$ obtemos uma solução para $\text{PM}(D_i, s, t)$.

caso 1.2: w não é o único vértice comum entre os caminhos P_1 e Q .

Seja u o penúltimo vértice de P_1 que é comum a P_1 e Q (lembre-se que o último é w).

Suponha que u está em Q_1 . Seja Q_1' a seção de Q_1 entre u e w . De modo similar ao caso 1.1 podemos mostrar que nenhum caminho em $\mathcal{P}$ intercepta Q_1' . Substituindo em $\mathcal{P}$ o caminho P por $P_1[s, u] \cdot Q_1' \cdot P_2$ obtemos uma solução para $\text{PM}(D_i, s, t)$.

Suponha, agora, que u está em Q_2 . Seja Q_2' a seção de Q_2 entre w e u . De maneira similar ao caso 1.1 podemos mostrar que nenhum caminho em $\mathcal{P}$ intercepta Q_2' . Substituindo em $\mathcal{P}$ o caminho P por $P_1[s, u] \cdot (Q_2')^{-1} \cdot P_2$ obtemos uma solução para $\text{PM}(D_i, s, t)$. Precisamos mostrar que ainda estão no grafo os reversos dos arcos que estão em Q_2' . Considere o circuito $C = (Q_2')^{-1} \cdot P_1^{-1}[w, u] = (u = u_0, u_1, \dots, u_l = u)$. Suponha que o arco

$u_i u_{i+1}$, $0 \leq i \leq l$, tenha sido retirado do grafo. Considere o caminho de busca P_B no momento em que $u_i u_{i+1}$ foi retirado. Seja u_j o último vértice de P_B no qual P_B entra em C pelo lado esquerdo (isto ocorre pelo menos uma vez pois s está do lado esquerdo de C). Depois de u_j , o caminho P_B deve passar por $u_j u_{j-1}$, caso contrário, $u_j u_{j-1}$ seria um arco saindo pelo lado direito de P_B . Assim, P_B começa a percorrer o reverso de C . O caminho P_B deve sair de C em algum vértice $u_{j'}$ para voltar e começar a percorrer este circuito no sentido correto. Mais ainda, P_B deve sair pelo lado direito de C pois u_j é o último vértice de P_B no qual este caminho entra em C pelo lado esquerdo. Como não pode existir arco saindo pelo lado direito de P_B , então $u_{j'} u_{j'-1}$ já foi retirado. Mas isto é um absurdo pois este arco está em P_1 ou Q_2 . Logo, ainda estão no grafo os reversos de todos os arcos que estão em Q_2' . Portanto a coleção de caminhos que mencionamos é uma coleção para o $\text{PM}(D_i, s, t)$.

caso 2: $a_i = vw$ é especial.

Mostraremos que, imediatamente antes de $a_i = vw$ ser retirado, não existia, no grafo atual, nenhum arco saindo de w . Como vw é um arco especial, então o vértice w já tinha sido liberado pelo caminho de busca atual quando vw é retirado. Considere o caminho de busca na última vez em que w foi o vértice-líder e no momento em que vw é removido. Seja w_0 o último vértice comum entre estes dois caminhos. Seja $P_1 \cdot P_2$ o primeiro caminho e seja $P_1 \cdot P_2'$ o segundo caminho. Seja $P_2 = (w_0, w_1, \dots, w_l = w)$.

Quando $w_{l-1}w$ foi retirado do grafo não existia nenhum arco saindo de w , com exceção, eventualmente, de ww_{l-1} . Mostraremos que este arco também foi retirado. Note que o contador global mudou entre as retiradas de wv e de vw pois, caso contrário, o arco-líder do caminho de busca quando vw é removido não poderia ter v como sua ponta, pois o contador local teria o mesmo valor nos extremos do arco-líder. Deste modo, quando wv foi retirado o arco do caminho de busca que entrava em w era um arco distinto de $w_{l-1}w$. Segue pelo Lema 4.2.9 que, no momento em que $w_{l-1}w$ é o arco-líder, o arco ww_{l-1} já foi retirado do grafo original. Assim, no momento imediatamente anterior à retirada de vw não existia no grafo nenhum arco saindo de w e, portanto, nenhum caminho de s a t passava pelo arco vw . Segue que $\mathcal{P}$ é uma solução para $\text{PM}(D_i, s, t)$.

□

Lema 4.2.11 *Se o arco vw é removido por causa de um CPE envolvendo um circuito C e*

o arco wv ainda não foi retirado, então mais tarde wv será retirado porque os contadores locais de v e w são iguais.

Dem: Após a retirada de wv todos os arcos que o algoritmo toma como arco-de-continuação antes de wv são orientados para o interior de C . Logo, o algoritmo não alcança t (pois t está na face externa) ou qualquer caminho já construído, diferente do caminho de busca, antes de wv ser o arco-de-continuação. Quando wv é considerado pelo algoritmo os contadores locais de v e w são iguais e ele é retirado.

□

Lema 4.2.12 *Durante a execução do algoritmo não ocorre nenhum CPD.*

Dem: Suponha que o primeiro CPD ocorre no vértice w_0 e envolve o circuito $C = (w_0, w_1, \dots, w_l, w_{l+1} = w_0)$. Considere o momento em que ocorre este conflito.

Inicialmente, mostraremos por indução que os arcos $w_i w_{i-1}$, $i = 1, \dots, l+1$, já foram retirados e que $w_{i-1} w_{i-2}$ foi retirado antes de $w_i w_{i-1}$, para $1 < i \leq l+1$.

O arco $w_{l+1} w_l$ já deve ter sido retirado do grafo pois o caminho de busca atual passa por w_0 e deixa este arco à sua direita. Suponha que $w_i w_{i-1}$, $1 < i \leq l+1$, já tenha sido retirado. Vamos mostrar que $w_{i-1} w_{i-2}$ foi retirado antes de $w_i w_{i-1}$.

Se $w_i w_{i-1}$ fosse um arco especial então w_{i-1} teria sido liberado pelo menos uma vez antes de $w_i w_{i-1}$ ser retirado. E, então, $w_{i-1} w_i$ já teria sido retirado. Mas isto é impossível pois o conflito envolve o circuito C que passa por este arco. Suponha que $w_i w_{i-1}$ foi retirado por causa de um conflito. No momento em que ocorre o CPD envolvendo o circuito $C = (w_0, w_1, \dots, w_l, w_{l+1} = w_0)$, o arco $w_{i-1} w_i$ está no caminho de busca. Quando $w_i w_{i-1}$ é retirado, w_i é o vértice líder antes de $w_{i-1} w_i$ fazer parte do caminho de busca. Antes de w_i ser liberado, todos os arcos que saem de w_i , com exceção do reverso do arco-líder são retirados. O arco $w_i w_{i+1}$ está em C , assim ele não foi retirado quando w_i foi liberado. Assim, $w_{i+1} w_i$ era o arco-líder no momento em que w_i foi liberado. Um raciocínio análogo nos leva a concluir que w_{i+2} era o vértice que antecedia w_{i+1} no caminho de busca no momento em que $w_i w_{i-1}$ é retirado. Prosseguindo, obtemos que $(w_{i-1}, w_{i-2}, \dots, w_0 = w_{l+1}, w_l, \dots, w_i)$ é uma seção do caminho de busca no momento

em que $w_i w_{i-1}$ foi retirado. Logo, $w_i w_{i-1}$ foi retirado por causa de um CPE. Segue pelo Lema 4.2.11, que $w_{i-1} w_i$ seria retirado porque os contadores locais de w_{i-1} e w_i são iguais. Mas isto é impossível pois $w_{i-1} w_i$ está no caminho de busca no momento em que ocorre o CPD envolvendo o circuito C .

Desta forma, quando $w_i w_{i-1}$ é retirado do grafo, w_{i-1} é liberado e, portanto, o arco $w_{i-1} w_{i-2}$ já foi retirado. Logo, os arcos $w_i w_{i-1}$, $i = 1, \dots, l+1$, são retirados antes de ocorrer o CPD envolvendo C .

Mostraremos, agora, que $w_0 w_1$ é retirado porque os contadores locais de w_0 e de w_1 são iguais. Segue pelo Lema 4.2.11, que é suficiente mostrar que $w_1 w_0$ é retirado por causa de um CPE em w_0 . Suponha que isto não seja verdade. Neste caso, em algum momento o caminho de busca atinge C e prossegue por $w_i w_{i-1}$. Neste estágio, $w_{i+1} w_i$ ainda não foi retirado. Mais tarde, $w_{i+1} w_i$ será um arco do caminho de busca (pois provamos acima que este arco será retirado quando w_i é liberado). Segue pelo Lema 4.2.9 que $w_i w_{i+1}$ será retirado antes de $w_{i+1} w_i$. Isto contradiz o que demonstramos anteriormente. Portanto, durante a execução do algoritmo não ocorre nenhum CPD.

□

Teorema 4.2.13 *O algoritmo para o problema de Menger é correto.*

Dem: Segue pelo Lema 4.2.12 que o algoritmo considera todos os casos. Todo arco em AD_m que sai de s no grafo D_m está em algum caminho da coleção encontrada. Segue pelo Lema 4.2.10 que o algoritmo encontra uma coleção máxima de caminhos disjuntos de s a t no grafo não-orientado G .

□

Claramente, o algoritmo apresentado acima para o problema de Menger é linear.

Capítulo 5

Caminhos e circuitos $p \bmod q$ em grafos orientados

O PHF-O para certos padrões em que o problema é $\mathcal{NP}$ -completo foi bastante usado para provar que certos problemas envolvendo modularidade de caminhos e circuitos em grafos orientados são $\mathcal{NP}$ -completos. Neste capítulo, estudaremos alguns destes problemas, em especial os dois seguintes:

Problema do caminho de comprimento $p \bmod q$

Entrada : Um grafo orientado D , e dois vértices s e t .

Pergunta : Existe em D um caminho de s a t de comprimento $p \bmod q$?

Problema do circuito de comprimento $p \bmod q$

Entrada : Um grafo orientado D .

Pergunta : Existe em D um circuito de comprimento $p \bmod q$?

Os números p e q são inteiros não-negativos e $p < q$.

Estudaremos na primeira seção deste capítulo alguns problemas envolvendo modularidade de caminhos e circuitos, sem fazer restrição na entrada do problema. Os resultados desta seção sobre a $\mathcal{NP}$ -completude de vários problemas são devidos a Thomassen [22] bem como a Arkin, Papadimitriou e Yannakakis [2]. Na segunda seção, estudaremos os problemas mencionados acima restringindo a entrada dos problemas a grafos bipartidos.

Os resultados que apresentaremos nesta seção não foram encontrados na literatura.

Neste capítulo todos os grafos considerados são conexos. Mesmo que não seja mencionado explicitamente, todas as reduções a serem apresentadas são polinomiais. Chamaremos de caminho/circuito par (ímpar) um caminho/circuito de comprimento par (ímpar).

5.1 Caso geral

O PHF-O em que o padrão consiste de um caminho de comprimento dois é conhecido na literatura como o **problema do caminho que passa por um vértice**. Este problema pode ser enunciado da seguinte maneira:

Dado um grafo orientado D e vértices $s, t, m \in VD$, distintos, decidir se existe um caminho de s a t que passa por m .

Conforme vimos no Capítulo 3, este problema é $\mathcal{NP}$ -completo. Arkin, Papadimitriou e Yannakakis [2] usaram este fato para provar que, para $q \geq 2$, o problema do caminho de comprimento $p \bmod q$ é $\mathcal{NP}$ -completo.

Veremos em seguida a redução apresentada por estes autores.

Teorema 5.1.1 *Sejam p e q inteiros tais que $0 \leq p < q$ e $q \geq 2$. O problema do caminho de comprimento $p \bmod q$ é $\mathcal{NP}$ -completo.*

Dem: Para p e q fixos é imediato que o problema está em $\mathcal{NP}$. Vamos mostrar que o problema do caminho passando por um vértice pode ser reduzido a este problema.

Dado um grafo orientado D e três vértices $s, t, m \in VD$, construímos um grafo D' que possui um caminho de s a t de comprimento $p \bmod q$ se, e somente se, existe em D um caminho de s a t que passa por m .

Sejam d_1, d_2 inteiros tais que $d_1 + d_2 = p \bmod q$, $0 < d_1, d_2 < q$. Seja D' o grafo obtido de D inserindo $d_1 - 1$ vértices em cada arco que sai de s , $d_2 - 1$ vértices em cada arco que sai de m e $q - 1$ vértices em cada um dos demais arcos.

Claramente, existe em D' um caminho de s a t de comprimento $p \bmod q$ se, e somente se, existe em D um caminho de s a t que passa por m .

□

Note que quando $p = 0$ e $q = 1$ o problema do caminho de comprimento $p \bmod q$ se reduz ao problema da existência de um caminho de s a t . É bastante conhecido o fato de que este problema é fácil (veja por exemplo [1]). Temos assim, o seguinte resultado.

Teorema 5.1.2 *Existe um algoritmo linear em $|AD|$ para o problema do caminho de comprimento $0 \bmod 1$.*

□

É interessante observar que enquanto o problema de decidir se um grafo orientado D possui um caminho par (ímpar) de s a t ($s, t \in VD$) é $\mathcal{NP}$ -completo, o problema de decidir se um grafo não-orientado G possui um caminho par (ímpar) de s a t ($s, t \in VG$) pode ser resolvido em tempo linear. LaPaugh e Papadimitriou [11] apresentaram um algoritmo linear para este problema.

O problema do PHF-O para grafos orientados em que o padrão é um grafo desconexo com dois arcos (Figura 3.11(a)) é conhecido na literatura como **problema dos dois caminhos**. Este problema pode ser enunciado da seguinte maneira:

Dado um grafo orientado D e vértices $s_1, t_1, s_2, t_2 \in VD$, distintos, decidir se existe um caminho de s_1 a t_1 e um caminho de s_2 a t_2 disjuntos nos vértices.

Vimos no Capítulo 3 que este problema é $\mathcal{NP}$ -completo (quer os vértices s_i e t_j sejam distintos ou não). Thomassen [22] usou este resultado para provar o seguinte teorema.

Teorema 5.1.3 *Dado um grafo orientado D e um arco $e \in AD$, o problema de decidir se existe em D um circuito de comprimento par que passa por e é $\mathcal{NP}$ -completo.*

Dem: É suficiente mostrar que o problema dos dois caminhos para grafos orientados onde os vértices dados são todos distintos pode ser reduzido ao problema de decidir se um grafo orientado possui um circuito par passando por um arco.

Dado um grafo orientado D e quatro vértices distintos $s_1, s_2, t_1, t_2 \in VD$, construímos um grafo D' que possui um circuito par passando por (t_2, s_1) se, e somente se, existe em D um caminho de s_1 a t_1 e um caminho de s_2 a t_2 disjuntos nos vértices.

Seja D' o grafo obtido de D inserindo um vértice em cada arco de $AD - \{(t_2, s_1), (t_1, s_2)\}$. Se (t_2, s_1) e (t_1, s_2) não estão em D' , acrescente-os.

Claramente, existe em D um caminho de s_1 a t_1 e um caminho de s_2 a t_2 disjuntos nos vértices se, e somente se, existe um circuito par passando por (t_2, s_1) em D' .

□

Teorema 5.1.4 *Dado um grafo orientado D e um arco $e \in AD$, o problema de decidir se existe em D um circuito ímpar que passa por e é $\mathcal{NP}$ -completo.*

□

Claramente, este último resultado também pode ser obtido fazendo-se uma redução similar àquela na demonstração do Teorema 5.1.3.

Usando os dois últimos teoremas (ou apenas um deles) é fácil provar o seguinte resultado.

Teorema 5.1.5 *Dado um grafo orientado D e um vértice $s \in VD$, o problema de decidir se D contém um circuito par (ímpar) que passa por s é $\mathcal{NP}$ -completo.*

□

O problema do PHF-O para grafos orientados em que o padrão é um circuito de comprimento dois é conhecido como o **problema do circuito que passa por dois vértices**. Este problema pode ser enunciado da seguinte forma:

Dado um grafo orientado D e dois vértices distintos $u, v \in VD$, decidir se existe um circuito que passa por u e v .

Arkin, Papadimitriou e Yannakakis [2] usaram o fato de que este problema é $\mathcal{NP}$ -completo para provar que o problema do circuito de comprimento $p \bmod q$, para $p \neq 0$ e $q > 2$, também é $\mathcal{NP}$ -completo.

Apresentaremos em seguida a prova deste resultado.

Cabe aqui observar que a $\mathcal{NP}$ -completude do problema do circuito par (ímpar) que passa por um determinado vértice também pode ser provada via problema do circuito que passa por dois vértices.

Teorema 5.1.6 *Sejam p e q inteiros tais que $0 < p < q$ e $q > 2$. O problema do circuito de comprimento $p \bmod q$ é $\mathcal{NP}$ -completo.*

Dem: Fixe p e q como acima. É suficiente mostrar que o problema do circuito que passa por dois vértices pode ser reduzido ao problema de decidir se um grafo orientado possui um circuito de comprimento $p \bmod q$.

Sejam d_1 e d_2 inteiros tais que $d_1 + d_2 = p \bmod q$, $0 < d_1, d_2 < q$. Dado um grafo orientado D e dois vértices $s, t \in VD$, seja D' o grafo obtido de D inserindo-se $d_1 - 1$ vértices em cada arco que sai de s , $d_2 - 1$ vértices em cada arco que sai de t e $q - 1$ vértices em cada um dos demais arcos.

Claramente, existe em D' um circuito de comprimento $p \bmod q$ se, e somente se, existe em D um circuito passando pelos vértices s e t .

□

Vamos agora analisar o caso em que $q \leq 2$. Quando $q = 1$ (e portanto $p = 0$) o problema é o de decidir se um grafo orientado possui um circuito. Isto pode claramente ser feito em tempo linear. Quando $q = 2$ e $p = 1$ é fácil ver que o problema pode ser resolvido em tempo polinomial. Para isso, basta observar que, se um grafo possui um passeio fechado ímpar então ele possui um circuito ímpar. E claramente, a existência ou não de um passeio fechado ímpar pode ser comprovada em tempo polinomial. Vale aqui ressaltar que o problema do circuito ímpar pode ser resolvido em tempo linear. A descrição de um algoritmo linear obtido por Ronaldo Hashimoto pode ser encontrada em sua dissertação [10]. Em resumo, temos os seguintes resultados:

Teorema 5.1.7 *Existe algoritmo linear para o problema do circuito de comprimento 0 mod 1.*

□

Teorema 5.1.8 *Existe algoritmo linear para o problema do circuito de comprimento 1 mod 2.*

□

É desconhecida a complexidade computacional deste problema quando $q = 2$ e $p = 0$, ou seja, do problema do circuito par. Trata-se de um problema que está em aberto a mais de 20 anos, tendo sido levantado por Younger (*cf.* Thomassen [22]). Para grafos planares, foi provada a existência de um algoritmo polinomial para o problema do circuito par. Tal resultado é devido a Thomassen [23]. Recentemente, Galluccio e Loeb1 [8] apresentaram um algoritmo polinomial para o seguinte problema (que é uma generalização do problema do circuito par): decidir se todos os circuitos em um grafo orientado planar são de comprimento $p \bmod q$, onde $0 \leq p < q$.

Finalmente, observamos que também é desconhecida a complexidade computacional do problema do circuito de comprimento $p \bmod q$, no caso em que $p = 0$ e $q > 2$.

Nas duas tabelas a seguir resumimos os resultados desta seção.

p	q	complexidade	resultados usados
$p \geq 0$	$q \geq 2$	$\mathcal{NP}$ -completo	Teorema 5.1.1
0	1	linear	Teorema 5.1.2

Tabela 5.1: Complexidade do problema do caminho de comprimento $p \bmod q$.

p	q	complexidade	resultados usados
$p > 0$	$q > 2$	$\mathcal{NP}$ -completo	Teorema 5.1.6
0	1	linear	Teorema 5.1.7
1	2	linear	Teorema 5.1.8
0	$q \geq 2$	aberto	comentários (pág. 122)

Tabela 5.2: Complexidade do problema do circuito de comprimento $p \bmod q$.

5.2 Grafos bipartidos

Na seção anterior vimos que o problema do circuito (caminho) de comprimento $p \bmod q$ é $\mathcal{NP}$ -completo para a maioria dos valores de p e q . Uma pergunta natural que surge é se tais problemas continuam tendo a mesma complexidade quando restringimos a entrada a grafos bipartidos. Nesta seção apresentaremos os resultados que obtivemos sobre esta questão.

Teorema 5.2.1 *O problema do circuito de comprimento $p \bmod q$, q ímpar, pode ser reduzido ao mesmo problema com entrada restrita a grafos bipartidos.*

Dem: Seja D o grafo fornecido como entrada para o problema sem restrição na entrada. Seja D_q o grafo obtido a partir de D , inserindo q vértices em cada arco. Claramente, D_q é bipartido e D_q possui um circuito de comprimento $p \bmod q$ se, e somente se, D possui um circuito de comprimento $p \bmod q$.

□

Teorema 5.2.2 *O problema do circuito de comprimento $p \bmod q$ pode ser polinomialmente reduzido ao problema do circuito de comprimento $2p \bmod 2q$ com entrada restrita a grafos bipartidos.*

Dem: Seja D o grafo fornecido como entrada para o problema do circuito de comprimento $p \bmod q$. Seja D_1 o grafo obtido a partir de D , inserindo um vértice em cada arco. Claramente, D_1 é bipartido e D_1 possui um circuito de comprimento $2p \bmod 2q$ se, e somente se, D possui um circuito de comprimento $p \bmod q$.

□

Teorema 5.2.3 *O problema do circuito de comprimento $2p \bmod 2q$ com entrada restrita a grafos bipartidos pode ser reduzido ao problema do circuito de comprimento $p \bmod q$.*

Dem: Seja $D(X, Y)$ o grafo bipartido fornecido como entrada para o primeiro problema. Seja D' o grafo obtido a partir de D , inserindo $q - 1$ vértices em cada arco que sai de X .

Suponha que D' possua um circuito de comprimento $p \bmod q$. Seja C' um tal circuito. Então o circuito C em D correspondente a C' tem $p \bmod q$ arcos que vão de Y para X . Portanto, C tem comprimento $2p \bmod 2q$.

Suponha, agora, que D possua um circuito de comprimento $2p \bmod 2q$. Seja C um tal circuito. O comprimento de C é $k(2q) + 2p$, para algum $k \geq 0$. O número de vértices de C em X é $(1/2)[k(2q) + 2p] = kq + p$. Assim, o comprimento do circuito correspondente a C em D' é $q(kq + p) + (kq + p) = (kq + k + p)q + p$, ou seja, $p \bmod q$.

Portanto, D tem um circuito de comprimento $2p \bmod 2q$ se, e somente se, D' possui um circuito de comprimento $p \bmod q$.

□

Segue diretamente dos dois teoremas anteriores o seguinte corolário.

Corolário 5.2.4 *O problema do circuito de comprimento $2p \bmod 2q$ com entrada restrita a grafos bipartidos é equivalente ao problema do circuito de comprimento $p \bmod q$.*

□

O próximo teorema segue diretamente da observação de que grafos bipartidos não possuem circuitos ímpares e do corolário anterior.

Teorema 5.2.5 *Quando q é ímpar o problema do circuito de comprimento $0 \bmod q$ com entrada restrita a grafos bipartidos é equivalente ao problema do circuito de comprimento $0 \bmod q$.*

□

Na tabela da Figura 5.3 reunimos os resultados mencionados acima sobre a complexidade do problema do circuito de comprimento $p \bmod q$ em grafos bipartidos, para os diversos valores de p e q .

p	q	complexidade	resultados usados
0	2	linear	Teorema 5.2.3 + Teorema 5.1.7
0	q par, $q \geq 4$	em aberto	Corolário 5.2.4 + comentários (pág. 122)
0	q ímpar, $q > 1$	em aberto	Teorema 5.2.5 + comentários (pág. 122)
$p \neq 0$	q ímpar	$\mathcal{NP}$ -completo	Teorema 5.2.1 + Teorema 5.1.6
p ímpar	q par	constante	$\nexists$ circuito ímpar em grafos bipartidos
2	4	linear	Teorema 5.2.3 + Teorema 5.1.8
p par, $p \neq 0$	q par, $q > 4$	$\mathcal{NP}$ -completo	Teorema 5.2.2 + Teorema 5.1.6

Tabela 5.3: Complexidade do problema do caminho de comprimento $p \bmod q$ em grafos bipartidos.

Passaremos agora a estudar a complexidade do problema do caminho de comprimento $p \bmod q$ com entrada restrita a grafos bipartidos.

Teorema 5.2.6 *O problema do caminho de comprimento $p \bmod q$, com q ímpar, $q \geq 3$, e entrada restrita a grafos bipartidos é $\mathcal{NP}$ -completo.*

Dem: Pelo Teorema 5.1.1, é suficiente mostrar que para $q \geq 3$, q ímpar, o problema do caminho de comprimento $p \bmod q$ pode ser reduzido ao problema do caminho de comprimento $p \bmod q$ com entrada restrita a grafos bipartidos.

Seja D o grafo fornecido na entrada do primeiro problema. Seja D_q o grafo obtido a partir de D , inserindo-se q vértices em cada arco. Claramente, D_q é bipartido e D_q possui um caminho de s a t de comprimento $p \bmod q$ se, e somente se, existe em D um caminho de s a t de mesma modularidade.

□

Teorema 5.2.7 *O problema do caminho de comprimento $p \bmod q$, q par, $q > 2$, e entrada restrita a grafos bipartidos é $\mathcal{NP}$ -completo.*

Dem: Novamente pelo Teorema 5.1.1, é suficiente mostrar que o problema do caminho de comprimento $0 \bmod q$, $q \geq 2$, pode ser reduzido ao problema do caminho de comprimento $p \bmod q$, $q > 2$, e entrada restrita a grafos bipartidos.

Seja D o grafo fornecido na entrada do primeiro problema. Seja D' o grafo obtido a partir de D , inserindo-se um vértice em cada arco e acrescentando-se o caminho $(t = t_0, e_1, t_1, \dots, e_p, t_p)$ disjunto de D a menos de t (caso $p = 0$, então apenas modificar o nome de t para t_0). Claramente, D' é bipartido e existe em D' um caminho de s a t_p de comprimento $p \bmod 2q$ se, e somente se, existe em D um caminho de s a t de comprimento $0 \bmod q$.

□

É bastante conhecido o fato de que existe algoritmo polinomial para o problema do caminho de comprimento $0 \bmod 1$ (veja [1]). O problema do caminho de comprimento $p \bmod q$, para $p = 0$ ($p = 1$) e $q = 2$, e entrada restrita a grafos bipartidos pode ser facilmente solucionado. Sejam D , s e t os dados fornecidos na entrada deste problema. Basta verificar se s e t estão no mesmo conjunto da partição (em conjuntos distintos da partição) e resolver o problema do caminho de comprimento $0 \bmod 1$, com entrada D , s e t .

Cabe notar que a variante do problema do caminho de comprimento $p \bmod q$, q par, $q > 2$, em que o grafo de entrada D é bipartido e os vértices s e t estão no mesmo conjunto da partição (conjuntos distintos da partição) é trivial nos casos em que p é ímpar (p é par).

Na tabela seguinte reunimos os resultados mencionados acima sobre a complexidade do problema do caminho de comprimento $p \bmod q$ em grafos bipartidos, para os diversos valores de p e q .

p	q	complexidade	resultados usados
$p \geq 0$	q ímpar, ≥ 3	$\mathcal{NP}$ -completo	Teorema 5.2.6
$p \geq 0$	q par, $q > 2$	$\mathcal{NP}$ -completo	Teorema 5.2.7
1	2	linear	comentários (pág. 126)
0	2	linear	comentários (pág. 126)

Tabela 5.4: Complexidade do problema do caminho de comprimento $p \bmod q$ em grafos bipartidos.

Conclusão

Apesar de o conceito de *homeomorfismo* ser bastante conhecido na Teoria dos Grafos, não encontramos na literatura um trabalho que abrangesse o estudo dos vários tipos de homeomorfismo e toda a gama de problemas como os que mencionamos nesta dissertação. Muitos dos problemas aqui tratados se apresentam na literatura formulados de uma outra maneira, sem mencionar explicitamente que se trata de um problema de homeomorfismo. O nosso trabalho consistiu em levantar várias questões sobre este tema, procurando respondê-las de uma maneira didática para facilitar o estudo dos tópicos aqui abordados, e ao mesmo tempo, fornecer uma visão ampla dos resultados existentes. Acreditamos, assim, que este trabalho possa ser útil às pessoas que desejam conhecer o estado-da-arte com relação a homeomorfismo em grafos ou queiram aprofundar estudos sobre este assunto. Esperamos, também, que este trabalho tenha permitido ao leitor identificar questões que ainda não foram satisfatoriamente respondidas e outras que poderiam ser melhor exploradas.

Dois problemas que merecem destaque especial são o problema do vértice-homeomorfismo com grafo padrão fixo e com vértice-mapeamento fixo, tanto para grafos não-orientados como para grafos orientados com entrada restrita a grafos planares. Seria interessante encontrar algoritmos polinomiais para esses problemas que também sejam eficientes do ponto de vista prático.

Bibliografia

- [1] A. V. Aho, J. E. Hopcroft and J. D. Ullman. *The design and analysis of computer algorithms*. Addison-Wesley, Reading, Mass., 1974.
- [2] E. M. Arkin, C. H. Papadimitriou and M. Yannakakis. Modularity of cycles and paths in graphs. *Journal of the Association for Computing Machinery*, 38(2):255–274, 1991.
- [3] T. Asano. An approach to the subgraph homeomorphism problem. *Theoretical Computer Science*, 38:249–267, 1985.
- [4] B. Bollobás. *Graph theory, an introductory course*. Springer-Verlag, New York, 1979.
- [5] J. A. Bondy and U. S. R. Murty. *Graph theory with applications*. Macmillan, London, 1976.
- [6] S. Even, A. Itai and A. Shamir. On the complexity of the timetable and multicommodity flow problems. *SIAM J. Comput.*, 5(4):691–703, 1976.
- [7] S. Fortune, J. Hopcroft and J. Wyllie. The directed subgraph homeomorphism problem. *Theoretical Computer Science*, 10:111–121, 1980.
- [8] A. Galluccio and M. Loeb. Cycles of prescribed modularity in planar digraphs. Preprint, 1992.
- [9] M. R. Garey and D. S. Johnson. *Computers and intractability: a guide to the theory of NP-completeness*. Freeman, New York, 1979.
- [10] R. Hashimoto. Circuitos e caminhos pares/ímpares em grafos. Dissertação de Mestrado, IME-USP. Em preparação, 1993.

- [11] A. S. LaPaugh and C. H. Papadimitriou. The even-path problem for graphs and digraphs. *Networks*, 14:507–513, 1984.
- [12] A. S. LaPaugh and R. L. Rivest. The subgraph homeomorphism problem. *Journal of Computer and System Sciences*, 20:133–149, 1980.
- [13] C. L. Lucchesi. *Introdução à teoria dos grafos*. IMPA, Rio de Janeiro, 1979.
- [14] J. F. Lynch. The equivalence of theorem proving and the interconnection problem. *(ACM) SIGDA Newsletter*, 5(3):31–36, 1975.
- [15] C. H. Papadimitriou and K. Steiglitz. *Combinatorial optimization: algorithms and complexity*. Prentice-Hall, New Jersey, 1982.
- [16] Y. Perl and Y. Shiloach. Finding two disjoint paths between two pairs of vertices in a graph. *Journal of the Association for Computing Machinery*, 25(1):1–9, 1978.
- [17] H. Ripphausen-Lipa, D. Wagner and K. Weihe. The vertex-disjoint Menger problem in planar graphs. Technical Report 324, T. U. Berlin, 1992.
- [18] N. Robertson and P. D. Seymour. Graph minors XIII: The disjoint paths problem. Preprint, 1992.
- [19] A. Schrijver. Finding k disjoint paths in a directed planar graph. Technical Report 9206, CWI, Amsterdam, 1992.
- [20] Y. Shiloach. A polynomial solution to the undirected two paths problem. *Journal of the Association for Computing Machinery*, 27(3):445–456, 1980.
- [21] H. Suzuki, T. Akama and T. Nishizeki. Algorithms for finding internally disjoint paths in a planar graph. *Electronics and Communication in Japan*, 72(10):55–67, 1989.
- [22] C. Thomassen. Even cycles in directed graphs. *Europ. J. Combinatorics*, 6:85–89, 1985.
- [23] C. Thomassen. The even cycle problem for planar digraphs. *Journal of Algorithms*, 15(1):61–75, 1989.