Problemas Circulatórios em Grafos

Cristina Gomes Fernandes

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

Banca examinadora:

Prof. Dr. Paulo Feofiloff (orientador)	–	IME-USP
Profa. Dra. Yoshiko Wakabayashi	–	IME-USP
Prof. Dr. Cláudio L. Lucchesi	–	IMECC-UNICAMP
Prof. Dr. Arnaldo Mandel (suplente)	–	IME-USP
Prof. Dr. Jayme Szwarcfter (suplente)	–	NCE-UFRJ

–São Paulo, Setembro de 1992–

à
ON

Resumo

Neste trabalho estudamos três versões do problema de encontrar uma circulação inteira viável em um grafo. Na primeira versão, a orientação dos ciclos da circulação é consistente e coincide com a orientação do grafo. Na segunda, a orientação dos ciclos da circulação e a orientação do grafo são ignoradas. Na terceira, a orientação dos ciclos da circulação é consistente e a orientação do grafo é ignorada. Apresentamos os principais resultados conhecidos: o algoritmo de Hoffman que resolve a primeira versão, o algoritmo de Seymour que resolve a relaxação racional da segunda versão e uma redução que mostra que a terceira versão é um problema $\mathcal{NP}$ -completo.

Depois, estudamos alguns casos particulares das duas últimas versões relacionados com várias conjecturas famosas: a *conjectura da cobertura dupla por ciclos*, a *conjectura da imersão forte* e a *conjectura do 5-fluxo*, proposta por Tutte. São apresentados, entre outras coisas, os algoritmos de Jaeger e o algoritmo de Younger, este último baseado em uma demonstração de Seymour, que resolvem alguns dos casos particulares.

Abstract

In this work we study three forms of the problem of finding a feasible integer circulation in a graph. In the first, the orientation of the cycles of the circulation is consistent and coincides with the graph orientation. In the second, the orientation of the cycles of the circulation and the graph orientation are ignored. In the third, the orientation of the cycles of the circulation is consistent and the graph orientation is ignored. We present the main known results: an algorithm of Hoffman that solves the first form, an algorithm of Seymour and Arkin that solves the rational relaxation of the second form, and a reduction showing that the third form is $\mathcal{NP}$ -complete.

After this, we study particular cases of the two last forms related to several well-known conjectures: the *double cover conjecture*, the *strong embedding conjecture* and the *5-flow conjecture*, proposed by Tutte. We present, among other things, algorithms of Jaeger and of Younger, the latter based on a proof of Seymour, that solve some of these cases.

Agradecimentos

Ao Paulof por ter despertado o meu interesse pela teoria dos grafos, pela paciência, pela orientação e por todas as revoluções que tornaram este trabalho mais legível.

Ao Steve pela companhia desde o início do mestrado: nas disciplinas, no exame de qualificação, nos seminários, nos momentos de dúvida (escutando, aconselhando e achando contra-exemplos) e, como diz a Lulu, num fim de tese...

Ao Dodó pela idéia inicial e manutenção dos nossos seminários, pela paciência, pela leitura de algumas das várias versões e pelas sugestões.

Ao Carlinhos e à Dilma, por muuuuitas coisas mas principalmente pela energia que passaram nestes últimos dias, essencial para que este trabalho terminasse.

Ao Tito pela ajuda também essencial nestes últimos dias, lendo, corrigindo (provas e demonstrações) e cronometrando os minutos finais.

À Wânia pelo apoio, pela sala, pelas orações e principalmente pela confiança.

À Lulu e ao Daniel, pela energia contagiante e pela companhia em algumas madrugadas.

À Nami pelo apoio e pelo ombro.

À Yô, ao Yoshi, à Maria Angela e à Leila, por algumas conversas e referências.

Ao Arnaldo pela implantação da rede que facilitou e agilizou a digitação deste trabalho e pela paciência ao resolver vários dos meus problemas e dúvidas, diminuindo meu tempo gasto em RTFM.

Ao destino por não ter causado problemas na rede enquanto o Arnaldo viajava durante esta última semana ☺

Ao Mané, à Sônia e ao Edson, pelos vários empurrões.

À Joelma, à Adenilza e à Edna, por resolverem vários dos meus probleminhas.

Ao Claudio e ao Kunio pelo incentivo e pela torcida.

Finalmente, à minha mamãe e aos meus irmãos, pelo carinho e apoio.

Conteúdo

Introdução	1
1 Os problemas	3
1.1 Circulações viáveis	3
1.2 Colorações de circulações	7
1.3 Um breve histórico	8
2 O problema P1 e o algoritmo de Hoffman	10
2.1 As condições de Hoffman	10
2.2 A solução do problema P1	13
2.3 A relaxação fracionária do problema P1	15
2.4 Coloração de circulações consistentes	18
3 O problema P2 e o algoritmo de Seymour	21
3.1 As condições de Seymour	21
3.2 O algoritmo de Seymour: primeira etapa	23
3.3 O algoritmo de Seymour: segunda etapa	26
3.4 O subalgoritmo 6.1	31
3.5 O subalgoritmo 6.2	33
3.6 O subalgoritmo 6.3	35
3.7 O subalgoritmo 6.4	38
3.8 O problema P2	44
4 Os algoritmos de Jaeger e alguns casos particulares de $P2^+$ e P3	47
4.1 Os algoritmos de Jaeger	47
4.2 Construção de circulações fundamentais	49
4.3 Os problemas P2 e $P2^+$ com limites 1 e k	51

4.4	Os problemas P_2 e P_2^+ com limites k e k	52
4.5	O problema P_3 com limites 1 e k	54
4.6	Circulações e colorações orientáveis	57
4.7	A complexidade dos problemas	61
5	O algoritmo de Seymour/Younger	63
5.1	O algoritmo de Seymour/Younger	63
5.2	Algoritmo de Seymour/Younger: primeira etapa	64
5.3	O algoritmo de Seymour/Younger: segunda etapa	66
5.4	O subalgoritmo 11.1	68
6	Grafos em superfícies	71
6.1	Imersões no plano e imersões fortes em outras superfícies	71
6.2	Reduções a grafos 3-regulares e 5-regulares	74
6.3	Reduções a grafos com cintura grande	77
6.4	Grafos imersíveis no plano projetivo e no toro	80

Introdução

Existem diversas versões do problema de encontrar uma circulação inteira viável em um grafo. Por exemplo, a versão em que a orientação dos ciclos da circulação é consistente e coincide com a orientação do grafo (P1), a versão em que a orientação dos ciclos da circulação e a orientação do grafo são ignoradas (P2) e a versão em que a orientação dos ciclos da circulação é consistente e a orientação do grafo é ignorada (P3).

Algumas destas versões são generalizações combinatórias de problemas clássicos em teoria dos grafos que envolvem topologia: os problemas de imersões de grafos em superfícies e de coloração de faces.

Olhadas desta forma, as versões P2 e P3 têm extensões naturais ($P2^+$ e $P3^+$) envolvendo o conceito de coloração de circulações, que é uma generalização do conceito coloração de faces de imersões.

Neste trabalho, estudamos estas cinco versões (P1, P2, P3, $P2^+$ e $P3^+$).

No capítulo 1, depois da descrição precisa das cinco versões do problema, apresentamos um breve histórico.

No capítulo 2, descrevemos o algoritmo de Hoffman que resolve P1 e um algoritmo que mostra que P3 e $P3^+$ são equivalentes.

No capítulo 3, descrevemos o algoritmo de Seymour que resolve P2, sem a restrição a circulações inteiras. A versão restrita a circulações inteiras continua sem solução.

A partir do capítulo 4, nos concentramos em alguns casos particulares de P2, $P2^+$ e P3 cujo interesse vem da relação direta com os problemas de imersões de grafos em superfícies e de coloração de faces.

No capítulo 4, mostramos dois algoritmos de Jaeger e enunciamos diversas conjecturas existentes na literatura relacionadas a estes casos particulares, entre elas, a *conjectura da cobertura dupla por ciclos* e as *conjecturas de Tutte sobre*

circulações consistentes. Várias equivalências e implicações entre estes casos e estas conjecturas são discutidas. Mostramos também que $P3$ e $P2^+$ são $\mathcal{NP}$ -completos.

No capítulo 5, descrevemos o algoritmo de Younger, baseado em uma demonstração de Seymour, que melhora significativamente um dos resultados obtidos por Jaeger para $P3$.

No capítulo 6, nos restringimos a grafos imersos em superfícies. Analisamos primeiramente o caso dos grafos planares, relatando as implicações de alguns teoremas sobre coloração de faces para os casos particulares em questão. A generalização para grafos imersíveis em outras superfícies esbarra na famosa *conjectura da imersão forte*. São descritos algoritmos baseados nas idéias de Steinberg e Möller que demonstram a validade de uma das conjecturas de Tutte sobre circulações consistentes para grafos imersíveis no plano projetivo e no toro.

Capítulo 1

Os problemas

Existem diversas variações do problema de encontrar uma circulação viável em um grafo. Neste capítulo apresentaremos algumas variações do problema que podem ser vistas como generalizações de problemas clássicos de imersões de grafos em superfícies e de coloração de faces.

1.1 Circulações viáveis

Nesta seção, descreveremos três versões do problema de encontrar uma circulação viável num grafo: a versão em que a circulação é consistente, a versão sem orientação e a versão em que a circulação é consistente com a orientação do grafo.

Denotamos por VG o conjunto de vértices de um grafo G e por AG o conjunto de arestas de G .

Uma *trilha* em um grafo G é uma seqüência $\langle v_0, e_1, v_1, \dots, e_k, v_k \rangle$ onde $v_1, v_2, \dots, v_k$ são vértices, $e_1, e_2, \dots, e_k$ são arestas duas a duas distintas e v_{i-1} e v_i são as pontas da aresta e_i para todo i em $\{1, 2, \dots, k\}$. Note que um vértice pode ocorrer mais de uma vez numa trilha, mas uma aresta não. Uma trilha T usa uma aresta e de x para y se a única ocorrência de e em T é precedida de x e seguida de y .

Seja $\langle v_0, e_1, v_1, \dots, e_k, v_k \rangle$ uma trilha, digamos T . Denotamos por VT o conjunto $\{v_i : 0 \leq i \leq k\}$ e por AT o conjunto $\{e_i : 1 \leq i \leq k\}$. Note que $|AT| = k$ pois não há repetição de arestas na seqüência e que $|VT| \leq k + 1$ pois, como dissemos, pode haver repetição de vértices na seqüência. Se o grafo tiver uma orientação, denotamos por $\overrightarrow{AT}$ o conjunto $\{e_i : v_{i-1} \text{ é o início de } e_i \text{ e } v_i \text{ é o fim}\}$ e por $\overleftarrow{AT}$ o conjunto $\{e_i : v_{i-1} \text{ é o fim de } e_i \text{ e } v_i \text{ é o início}\}$.

Um *ciclo* é uma trilha $\langle v_0, e_1, v_1, \dots, e_k, v_k \rangle$ onde $v_0 = v_k$. O ciclo tem uma orientação natural independente da eventual orientação das arestas de G : a aresta e_i é usada de v_{i-1} para v_i .

Uma *circulação* em G é um par $(\mathcal{C}, m_{\mathcal{C}})$, onde $\mathcal{C}$ é um conjunto de ciclos de G e $m_{\mathcal{C}}$ é uma função de $\mathcal{C}$ em $\mathbb{Q}^+$ que representa a multiplicidade de cada ciclo na circulação. Chamaremos a função $m_{\mathcal{C}}$ de *função multiplicidade*. Frequentemente diremos *uma circulação $\mathcal{C}$* , ficando subentendida a função multiplicidade. Às vezes é conveniente supor que $m_{\mathcal{C}}$ está definida para todo ciclo de G , valendo 0 nos ciclos que não aparecem em $\mathcal{C}$. Se $m_{\mathcal{C}}$ é uma função inteira, dizemos que $\mathcal{C}$ é uma circulação *inteira*.

Seja $\mathcal{C}$ uma circulação. Dizemos que $\mathcal{C}$ *usa* uma aresta e se algum ciclo de $\mathcal{C}$ usa e . Denotamos por $p_{\mathcal{C}}$ e chamamos de *função peso* a função de AG em $\mathbb{Q}^+$ tal que

$$p_{\mathcal{C}}(e) = \sum \{m_{\mathcal{C}}(C) : C \in \mathcal{C}, AC \ni e\}.$$

No cálculo de $p_{\mathcal{C}}(e)$ são somadas as multiplicidades de todos os ciclos que usam e , independente do sentido em que e está sendo usada. Dada uma orientação D , denotamos por $p_{\mathcal{C}}^D$ a função de AG em $\mathbb{Q}$ definida por

$$p_{\mathcal{C}}^D(e) = \sum \{m_{\mathcal{C}}(C) : C \in \mathcal{C}, \vec{AC} \ni e\} - \sum \{m_{\mathcal{C}}(C) : C \in \mathcal{C}, \overleftarrow{AC} \ni e\}.$$

Em $p_{\mathcal{C}}^D(e)$ levamos em conta o sentido em que e é usada pelos ciclos de $\mathcal{C}$. Observe que, ao contrário de $p_{\mathcal{C}}$, a função $p_{\mathcal{C}}^D$ pode assumir valores negativos.

Dizemos que um ciclo C é *consistente com uma dada orientação D de G* se $\overleftarrow{AC} = \emptyset$. Dizemos que uma circulação $\mathcal{C}$ é *consistente com D* se todos os ciclos de $\mathcal{C}$ são consistentes com D . Dizemos que $\mathcal{C}$ é *consistente* se existe uma orientação com a qual $\mathcal{C}$ é consistente.

Nos problemas que estudaremos, para cada aresta do grafo consideraremos um par ordenado de números racionais positivos que chamaremos de *limite inferior* e *limite superior* da aresta. Os limites inferiores e superiores das arestas são dados através de um par ordenado de funções de AG em $\mathbb{Q}^+$ que chamaremos de *limites*.

Sejam ℓ, u limites. Dizemos que uma função q de AG em $\mathbb{Q}^+$ é ℓ, u -*viável* se $\ell \leq q \leq u$, ou seja, $\ell(e) \leq q(e) \leq u(e)$ para toda aresta e .

Problema P1(G, D, ℓ, u): Dado um grafo G , uma orientação D de G e limites ℓ, u , encontrar uma circulação inteira C consistente com D tal que p_c é ℓ, u -viável.

Problema P2(G, ℓ, u): Dado um grafo G e limites ℓ, u , encontrar uma circulação inteira C tal que p_c é ℓ, u -viável.

Problema P3(G, ℓ, u): Dado um grafo G e limites ℓ, u , encontrar uma circulação inteira consistente C tal que p_c é ℓ, u -viável.

Se C é uma circulação consistente com uma orientação D de G então $p_c = p_c^D$. Logo, no enunciado de P1, poderíamos ter utilizado p_c^D ao invés de p_c . O problema P3 poderia ter sido enunciado como: dado um grafo G e limites ℓ, u , encontrar uma orientação D de G tal que o problema P1(G, D, ℓ, u) tenha solução. Observe que, dado um grafo G e limites ℓ, u , toda solução de P3 é uma solução de P2.

Exemplo 1.1 Um grafo G com uma orientação e limites inferior e superior indicados nas arestas de G . Ao lado, uma solução de P1 que consiste de três ciclos.

Exemplo 1.2 Um grafo G e limites inferior e superior indicados nas arestas de G . Ao lado, uma solução de P2 que consiste de três ciclos.

Para cada um dos problemas existem instâncias sem solução (veja exemplo 1.4). Portanto a resolução de cada um deles envolve a determinação de condições necessárias e suficientes para a existência de solução.

Exemplo 1.3 Um grafo G e limites inferior e superior indicados nas arestas de G . Ao lado, uma solução de $P3$. A orientação com a qual a circulação é consistente está indicada nas arestas dos ciclos.

Exemplo 1.4 Não existe solução de P_i para o grafo G_i com os limites indicados abaixo, conforme mostramos a seguir.

Não existe solução de $P1$ para o grafo G_1 com os limites indicados pois todo ciclo que passa por v_1 usa a aresta e_1 . Teríamos no máximo 4 ciclos na circulação usando e_1 . Por outro lado, teríamos pelo menos 5 ciclos usando as outras duas arestas incidentes a v_1 . No grafo G_2 , todo ciclo que usa e_2 usa uma das outras duas arestas incidentes a v_2 . Teríamos pelo menos 4 ciclos usando e_2 . No entanto, no máximo 3 usando as duas outras arestas incidentes a v_2 . No grafo G_3 , olhando para os limites das arestas incidentes aos vértices v_3 e v_4 , vemos que as arestas do triângulo externo seriam usadas necessariamente uma vez pela circulação, enquanto que as arestas e_3 e e_4 seriam usadas exatamente duas vezes. Qualquer que seja a orientação destas arestas, não há circulação que use a outra aresta incidente a v entre 1 e 3 vezes.

Podemos definir relaxações dos problemas $P1$, $P2$ e $P3$ removendo a restrição de integridade das circulações, ou seja, admitindo circulações com multiplicidade fracionária. Chamaremos tais relaxações de $P1r$, $P2r$ e $P3r$, respectivamente.

Se $\mathcal{C}$ é uma circulação inteira então tanto a função p_c quanto a função p_c^p são inteiras. Assim podemos nos restringir a limites ℓ, u inteiros pois, para toda aresta e , vale que $\ell(e) \leq p_c(e) \leq u(e)$ se e somente se $\lceil \ell(e) \rceil \leq p_c(e) \leq \lfloor u(e) \rfloor$. Faremos algumas referências ao problema com limites fracionários, mas a maior parte do tempo consideraremos apenas limites inteiros.

1.2 Colorações de circulações

O conceito de coloração de uma circulação é uma generalização do conceito de coloração de faces de mapas de grafos planares. Esse conceito aplica-se apenas a circulações inteiras. Como veremos no capítulo 4, ele serve de ligação entre P2 e P3. Nesta seção, descreveremos duas variações do problema de encontrar uma circulação viável que envolvem esse conceito.

Dadas circulações $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_k$, denotamos por $\mathcal{C}_1 + \mathcal{C}_2 + \dots + \mathcal{C}_k$ a circulação definida pela coleção $\mathcal{C}_1 \cup \mathcal{C}_2 \cup \dots \cup \mathcal{C}_k$ e função multiplicidade $m_{c_1} + m_{c_2} + \dots + m_{c_k}$.

Uma k -coloração de uma circulação $\mathcal{C}$ é uma coleção $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_k$ de circulações inteiras tais que p_{c_i} é 0, 1-viável para todo i e a função peso da circulação $\mathcal{C}_1 + \mathcal{C}_2 + \dots + \mathcal{C}_k$ é p_c . Por exemplo, $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_k$ pode ser uma partição de $\mathcal{C}$ mas isto não é necessário.

Exemplo 1.5 Um grafo, uma circulação $\mathcal{C}$ que consiste de três ciclos, cada um com multiplicidade 1. A função peso de $\mathcal{C}$ está indicada nas arestas do grafo. A circulação $\mathcal{C}$ tem uma 2-coloração que consiste das circulações $\mathcal{C}_1$ e $\mathcal{C}_2$, cada uma, por sua vez, consistindo de um único ciclo.

Problema $P2^+(G, \ell, u, k)$: Dado um grafo G , limites ℓ, u e um natural k , encontrar uma circulação inteira C tal que p_C é ℓ, u -viável e uma k -coloração para C .

Problema $P3^+(G, \ell, u, k)$: Dado um grafo G , limites ℓ, u e um natural k , encontrar uma circulação inteira consistente C tal que p_C é ℓ, u -viável e uma k -coloração para C .

Seja C uma circulação inteira e t o valor máximo de p_C . Como existe uma aresta contida em t ciclos (considerando as multiplicidades), toda k -coloração de C é tal que $k \geq t$. Existem, no entanto, circulações que não têm uma t -coloração.

Exemplo 1.6 Um grafo G , uma circulação C que consiste de três ciclos. A função peso está indicada no grafo G . A circulação C não tem uma 2-coloração.

1.3 Um breve histórico

Em 1960, Hoffman [21] resolveu $P1$, apresentando condições necessárias e suficientes para que este tenha solução. Os problemas $P2$ e $P3$ não foram resolvidos ainda. A relaxação racional de $P2$ foi resolvida por Seymour [43] em 1979.

A resolução de $P2$ implicaria na demonstração da validade ou não da famosa *conjectura da cobertura dupla por ciclos*. Tal conjectura diz que *todo grafo 2-aresta-conexo tem uma cobertura dupla por ciclos*, ou seja, que o caso particular de $P2$ em que os limites inferiores e superiores são iguais a 2 em todas as arestas tem solução para todo grafo 2-aresta-conexo. Esta conjectura vale para grafos planares pois a coleção das fronteiras das faces de uma imersão de um grafo 2-aresta-conexo no plano é uma cobertura dupla por ciclos. A também famosa *conjectura da imersão forte* é a generalização deste fato para outras superfícies. Esta conjectura diz que *todo grafo 2-aresta-conexo tem uma imersão em alguma superfície tal que a fronteira de toda face é um ciclo*.

O caso particular de $P3$ em que o limite inferior vale 1 e o limite superior é constante em todas as arestas é bastante conhecido. Trata-se de uma generalização do *problema de coloração de mapas para grafos planares*. Em 1954, Tutte conjecturou [50] que *para todo grafo 2-aresta-conexo existe uma circulação inteira consistente $\mathcal{C}$ tal que p_c é 1,4-viável*. Em 1979, Jaeger demonstrou [25] que a conjectura vale com 7 no lugar de 4, ou seja, *para todo grafo 2-aresta-conexo existe uma circulação inteira consistente $\mathcal{C}$ tal que p_c é 1,7-viável*. Seymour melhorou esse resultado [44] em 1981, demonstrando a validade da conjectura com 5 no lugar de 4.

Em 1980, Little, Tutte e Younger mostraram [31] que toda circulação inteira consistente $\mathcal{C}$ possui uma t -coloração, onde t é o valor máximo de p_c , reduzindo assim o problema $P3^+$ ao problema $P3$.

Em 1984, Steinberg [45] e Möller [36] mostraram que a conjectura de Tutte sobre circulações consistentes vale para grafos imersíveis no plano projetivo e no toro. Em 1985, Goddyn [17] mostrou o mesmo para a conjectura da cobertura dupla por ciclos.

Como consequência dos trabalhos de Steinberg [45], Möller [36] e Goddyn [17], sabe-se que contra-exemplos minimais da conjectura de Tutte sobre circulações consistentes e da conjectura da cobertura dupla por ciclos são *snarks* com cintura maior que 6. Um snark é um grafo 2-aresta-conexo cúbico que não possui uma 3-coloração de arestas. O estudo de snarks começou em 1880 com uma demonstração falha do *teorema das quatro cores* [48]. Apenas quatro snarks eram conhecidos até 1975, quando Isaacs descreveu [23] duas classes infinitas deles. O termo *snark* foi criado por Gardner [15] em um artigo que despertou um grande interesse no assunto. Em 1980, Jaeger conjecturou [26] que *todo snark tem cintura no máximo 6*.

Capítulo 2

O problema P1 e o algoritmo de Hoffman

Hoffman apresentou [21] condições necessárias e suficientes para que o problema P1 tenha solução. Descreveremos, neste capítulo, o algoritmo de Hoffman que resolve o problema P1 e também a relaxação fracionária de P1. Descreveremos também um algoritmo que constrói uma coloração mínima para circulações inteiras consistentes, reduzindo o problema $P3^+$ ao problema P3.

2.1 As condições de Hoffman

Seja G um grafo. Um *corte* é um conjunto X de vértices de G tal que $X \neq \emptyset$ e $X \neq VG$. Dizemos que X é *trivial* se $|X| = 1$ ou $|VG - X| = 1$. Denotamos por $\delta(X)$ o conjunto das arestas com exatamente uma ponta em X . Se $|\delta(X)| = k$ então dizemos que X é um k -*corte*. Dada uma orientação de G , denotamos por $\delta^+(X)$ o conjunto das arestas cujo início está em X e o fim não está, e por $\delta^-(X)$ o conjunto das arestas cujo fim está em X e o início não está.

Dada uma função q de AG em $\mathbf{Q}^+$ e um conjunto E de arestas, denotamos por $q(E)$ o número $\sum\{q(e) : e \in E\}$.

Seja D uma orientação de G e X um corte. Observe que todo ciclo C consistente com D usa o mesmo número de arestas em $\delta^+(X)$ e em $\delta^-(X)$. Assim, dada uma circulação $\mathcal{C}$ consistente com D , temos que $p_c(\delta^-(X)) = p_c(\delta^+(X))$.

Dada uma circulação $\mathcal{C}$ consistente com D tal que $p_{\mathcal{C}}$ é ℓ, u -viável, temos que $\ell(\delta^-(X)) \leq p_{\mathcal{C}}(\delta^-(X)) = p_{\mathcal{C}}(\delta^+(X)) \leq u(\delta^+(X))$. Ou seja, uma condição necessária para que P1 tenha solução é que

$$\ell(\delta^-(X)) \leq u(\delta^+(X)) \text{ para todo corte } X.$$

Hoffman demonstrou que se $\ell \leq u$ então tal condição é suficiente para garantir a existência de uma função q de AG em $\mathbf{Q}^+$ tal que $q(\delta^-(X)) = q(\delta^+(X))$ para todo corte X . Descreveremos depois um algoritmo que transforma q em uma solução de P1.

Dizemos que os limites ℓ, u *satisfazem as condições de Hoffman num corte* X se $\ell(\delta^-(X)) \leq u(\delta^+(X))$. Dizemos simplesmente que ℓ, u *satisfazem as condições de Hoffman* se ℓ, u satisfazem as condições de Hoffman em todo corte X . Dizemos que uma função q *satisfaz as condições de Hoffman* se $q(\delta^-(X)) = q(\delta^+(X))$ para todo corte X . Observe que a validade da condição $\ell \leq u$ não decorre da validade da condição de Hoffman.

Exemplo 2.1 Os limites ℓ, u apresentados abaixo satisfazem as condições de Hoffman, porém existe uma aresta em que o limite inferior é maior que o limite superior.

Um *caminho* em G é uma trilha P tal que $|VP| = k + 1$, ou seja, é uma trilha que não repete vértices. Dizemos que um caminho P é *aumentador em relação a uma função* q *de* AG *em* $\mathbf{Q}^+$ se $q(e) < u(e)$ para toda aresta e em $\vec{AP}$ e $q(e) > \ell(e)$ para toda aresta e em $\overleftarrow{AP}$.

Algoritmo 1 (Hoffman) *Recebe um grafo* G , *uma orientação* D *para* G , *limites inteiros* ℓ, u *tais que* $\ell \leq u$ *e devolve:*

uma função ℓ, u -*viável com valores em* $\mathbf{Z}^+$ *que satisfaz as condições de Hoffman ou*

um corte X *tal que* $\ell(\delta^-(X)) > u(\delta^+(X))$.

Cada iteração do algoritmo começa com uma função $0, u$ -viável q com valores em $\mathbf{Z}^+$ que satisfaz as condições de Hoffman. A primeira iteração começa com a função nula no papel de q . Cada iteração consiste em:

Caso 1: $\ell \leq q$.

Pare e devolva q .

Caso 2: Existe aresta $\bar{e}$ tal que $q(\bar{e}) < \ell(\bar{e})$.

Sejam x e y as pontas final e inicial de $\bar{e}$. Seja X o conjunto $\{v \in VG : \text{existe caminho de } x \text{ a } v \text{ aumentador em relação a } q\}$.

Caso 2.1: $y \in X$.

Seja P um caminho de x a y aumentador em relação a q . Seja C o ciclo $P\langle y, \bar{e}, x \rangle$, t_1 o número $\min\{u(e) - q(e) : e \in \overrightarrow{AC}\}$ e t_2 o número $\min\{q(e) - \ell(e) : e \in \overleftarrow{AC}\}$. Seja t o mínimo entre t_1 e t_2 e q' a função que coincide com q exceto nas arestas de C , onde vale $q(e) + t$ se $e \in \overrightarrow{AC}$ e $q(e) - t$ se $e \in \overleftarrow{AC}$. Comece nova iteração com q' no papel de q .

Caso 2.2: $y \notin X$.

Pare e devolva X .

Para mostrar que o algoritmo funciona é preciso verificar que no início de cada iteração q é uma função $0, u$ -viável com valores em $\mathbf{Z}^+$ que satisfaz as condições de Hoffman. A hipótese vale para a função nula. Supondo que valha no início de uma iteração em que o caso 2.1 é executado, vamos mostrar que a hipótese vale para q' . Os números t_1 e t_2 são inteiros positivos pois ℓ, u e q são funções de AG em $\mathbf{Z}^+$, $\ell \leq u$ e P é um caminho aumentador. Então t é um inteiro positivo e portanto q' tem valores em $\mathbf{Z}^+$. Além disso, $q'(e) = q(e)$ para toda aresta e em $AG - AC$ e, pela escolha de t , $0 \leq q'(e) \leq u(e)$ para toda aresta e em AC , ou seja, q' é $0, u$ -viável, como queríamos demonstrar.

Se o algoritmo termina no caso 1, obviamente q é ℓ, u -viável. Se o algoritmo termina no caso 2.2 então é preciso mostrar que $\ell(\delta^-(X)) > u(\delta^+(X))$. No caso 2.2, $q(e) = u(e)$ para toda aresta e em $\delta^+(X)$, $q(e) \leq \ell(e)$ para toda aresta e em $\delta^-(X) - \{\bar{e}\}$, e além disso $q(\bar{e}) < \ell(\bar{e})$. Então, como q satisfaz as condições de Hoffman, temos que $\ell(\delta^-(X)) > q(\delta^-(X)) = q(\delta^+(X)) = u(\delta^+(X))$.

Para analisar o número de iterações do algoritmo é conveniente introduzir a noção de defeito.

Seja q uma função $0, u$ -viável que satisfaz as condições de Hoffman. O *defeito* de q , denotado por $\text{def}(q)$, é o número $\sum \{\ell(e) - q(e) : e \in AG, q(e) < \ell(e)\}$.

Pela escolha de $\bar{e}$ e porque P é um caminho aumentador, temos que $t > 0$. Logo, a cada iteração, $\text{def}(q)$ diminui de pelo menos t . Como, além disso, temos que t é inteiro, $\text{def}(q) \leq \ell(AG)$ e o número máximo de iterações é $\ell(AG)$, onde m é o número de arestas de G .

Podemos determinar se $\ell \leq q$ ou encontrar $\bar{e}$ tal que $q(\bar{e}) < \ell(\bar{e})$ em tempo $O(m)$. Se G está representado pelas listas de adjacências, o cálculo do conjunto X pode ser feito em tempo $O(m)$, assim como o cálculo de t_1, t_2 e de q' a partir de q . Logo o algoritmo tem complexidade $O(\ell(AG) \cdot m)$.

2.2 A solução do problema P1

Seja G um grafo e D uma orientação. Vamos descrever agora o algoritmo que, dada uma função q de AG em $\mathbb{Q}^+$ que satisfaz as condições de Hoffman, constrói uma circulação consistente com D cuja função peso é q .

Seja $\mathcal{C}$ uma circulação, C um ciclo e t um número racional positivo. Denotamos por $\mathcal{C} + tC$ a circulação $\mathcal{C}'$ tal que $\mathcal{C}' = \mathcal{C} \cup \{C\}$ e $m_{\mathcal{C}'}$ coincide com $m_{\mathcal{C}}$ exceto em C , sendo que $m_{\mathcal{C}'}(C) = m_{\mathcal{C}}(C) + t$. Observe que $\mathcal{C}'$ é realmente uma circulação. Se $\mathcal{C}$ é uma circulação inteira e t é inteiro então a circulação $\mathcal{C} + tC$ é inteira. Além disso, se $\mathcal{C}$ e C são consistentes com uma orientação dada, a circulação $\mathcal{C} + tC$ também é consistente com essa orientação.

Algoritmo 2 (calcula circulação consistente com peso pré-definido)

Recebe um grafo G , uma orientação D para G , uma função $\bar{q}$ de AG em $\mathbb{Q}^+$ que satisfaz as condições de Hoffman e devolve uma circulação consistente com D cuja função peso é $\bar{q}$. Se a função $\bar{q}$ for inteira então a circulação devolvida é inteira.

O algoritmo é iterativo. Cada iteração começa com uma circulação $\mathcal{C}$ consistente com D e com uma função q de AG em $\mathbb{Q}^+$ que satisfaz as condições de Hoffman e a igualdade $q + p_{\mathcal{C}} = \bar{q}$. A primeira iteração começa com a circulação vazia e $\bar{q}$ nos papéis de $\mathcal{C}$ e q , respectivamente. Cada iteração consiste em:

Caso 1: $q = 0$.

Pare, devolvendo a circulação $\mathcal{C}$.

Caso 2: $q \neq 0$.

Seja C_0 um ciclo consistente com D tal que $q(e) \neq 0$ para toda aresta e em AC_0 e t o valor mínimo de q nas arestas de C_0 . Seja $\mathcal{C}'$ a circulação $\mathcal{C} + tC_0$ e q' a função que coincide com q exceto nas arestas de C_0 , sendo que $q'(e) = q(e) - t$ para toda aresta e em AC_0 . Comece nova iteração com $\mathcal{C}'$ e q' nos papéis de $\mathcal{C}$ e q .

Para verificar que o algoritmo acima funciona, é preciso mostrar que no caso 2 realmente existe um ciclo C_0 consistente com D tal que $q(e) \neq 0$, para toda aresta e em AC_0 . Como $q \neq 0$, existe pelo menos uma aresta e_0 em AG tal que $q(e_0)$ não é nulo. Seja v_1 a ponta final de e_0 . Como q satisfaz as condições de Hoffman, temos que $q(\delta^-(v_1)) = q(\delta^+(v_1))$. Logo, existe uma aresta e_1 em $\delta^+(v_1)$ distinta de e_0 com q não nulo. A iteração dessa construção produz o ciclo C_0 desejado.

Se ocorre o caso 1, ou seja, $q = 0$, a circulação $\mathcal{C}$ é tal que $p_c = \bar{q}$.

Além disso, é preciso mostrar que, no início de cada iteração, $\mathcal{C}$ é uma circulação consistente com D , q é uma função de AG em $\mathbf{Q}^+$ que satisfaz as condições de Hoffman e a igualdade $q + p_c = \bar{q}$. No início da primeira iteração a hipótese vale pois $\mathcal{C}$ é vazia e $q = \bar{q}$. Supondo que valha no início de uma iteração onde ocorre o caso 2, vamos mostrar que vale para $\mathcal{C}'$ e q' . Como t é o valor mínimo de q nas arestas de C_0 e q é uma função de AG em $\mathbf{Q}^+$, temos que q' é uma função de AG em $\mathbf{Q}^+$. Além disso, C_0 é um ciclo e q satisfaz as condições de Hoffman, logo q' satisfaz as condições de Hoffman. A circulação $\mathcal{C}'$ é consistente com D pois $\mathcal{C}$ e C_0 são consistentes com D . Além disso,

$$\begin{aligned} p_{\mathcal{C}'}(e) &= \sum \{m_{\mathcal{C}'}(C) : C \in \mathcal{C}', AC \ni e\} \\ &= \sum \{m_{\mathcal{C}'}(C) : C \in \mathcal{C} \cup \{C_0\}, AC \ni e\} \\ &= \begin{cases} \sum \{m_{\mathcal{C}'}(C) : C \in \mathcal{C}, AC \ni e\} + m_{\mathcal{C}'}(C_0) & \text{se } e \in AC_0, \\ \sum \{m_{\mathcal{C}'}(C) : C \in \mathcal{C}, AC \ni e\} & \text{se } e \notin AC_0, \end{cases} \\ &= \begin{cases} p_c(e) + t & \text{se } e \in AC_0, \\ p_c(e) & \text{se } e \notin AC_0. \end{cases} \end{aligned}$$

Assim, temos que $q' + p_{\mathcal{C}'} = q + p_c = \bar{q}$, como queríamos demonstrar.

Se a função $\bar{q}$ é inteira então vamos mostrar que no início de cada iteração a circulação $\mathcal{C}$ e a função q são inteiras. Isso certamente vale no início da primeira

iteração. Supondo que valha no início de uma iteração onde ocorre o caso 2, temos que o valor de t é inteiro, logo a circulação C' e a função q' são inteiras. Ou seja, de fato a circulação C e a função q são inteiras no início de toda iteração, em particular no início da última iteração.

O número de iterações do algoritmo é no máximo m , onde m é o número de arestas de G , pois, a cada iteração, q' tem pelo menos uma componente nula a mais que q . Se G está representado pelas listas de adjacências, podemos determinar se $q \neq 0$ e construir C_0 em tempo $O(m)$. O acréscimo de C_0 à circulação pode ser feito em tempo $O(m)$. Portanto o algoritmo produz o resultado desejado em tempo $O(m^2)$.

2.3 A relaxação fracionária do problema P1

O algoritmo de Hoffman, juntamente com o algoritmo 2, resolve também o problema P1r. Para verificar isso, observe primeiramente que, dado um grafo G , uma orientação D e limites ℓ, u , existe um inteiro t tal que $t\ell, tu$ são limites inteiros. Submeta então G, D e $t\ell, tu$ ao algoritmo de Hoffman. Se o algoritmo devolver um corte X tal que $t\ell(\delta^-(X)) > tu(\delta^+(X))$, então $\ell(\delta^-(X)) > u(\delta^+(X))$. Se o algoritmo devolver uma função q , submeta q/t ao algoritmo 2 obtendo uma circulação ℓ, u -viável consistente com D .

A recíproca é verdadeira: é possível obter uma solução do problema P1 a partir de uma solução do problema P1r. Na verdade, o algoritmo de Hoffman devolve diretamente uma solução de P1, ou seja, demonstra que para limites inteiros o problema P1r é equivalente ao problema P1. Nesta seção, descrevemos um algoritmo que, dados limites inteiros, transforma uma solução de P1r em uma solução de P1. Tal algoritmo não foi encontrado na literatura.

Considerações análogas valem para o problema P3 já que ele é equivalente ao problema de encontrar uma orientação de forma que o problema P1 correspondente tenha solução.

Observe que se os limites ℓ, u não são inteiros, P1r e P3r podem ter apenas soluções fracionárias, ou seja, os problemas P1 e P3 correspondentes podem não ter solução (veja exemplo 2.2).

O que o algoritmo 3 faz é construir, a partir de uma função q de AG em $\mathbb{Q}^+$ que satisfaz as condições de Hoffman, uma função q' de AG em $\mathbb{Z}^+$ que

satisfaz as condições de Hoffman tal que $\lfloor q(e) \rfloor \leq q'(e) \leq \lceil q(e) \rceil$ para toda aresta e . Com isso, dados limites inteiros ℓ, u e uma solução do problema P1r, ou seja, uma circulação consistente $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é ℓ, u -viável, basta submeter $p_{\mathcal{C}}$ ao algoritmo 3, obtendo uma função q tal que $\lfloor p_{\mathcal{C}}(e) \rfloor \leq q(e) \leq \lceil p_{\mathcal{C}}(e) \rceil$ para toda aresta e , e agora submeter q ao algoritmo 2, obtendo uma circulação inteira consistente $\mathcal{C}'$ tal que $p_{\mathcal{C}'} = q$. Como ℓ, u são funções inteiras, temos $\ell(e) \leq \lfloor p_{\mathcal{C}}(e) \rfloor \leq q(e) \leq \lceil p_{\mathcal{C}}(e) \rceil \leq u(e)$, ou seja, $p_{\mathcal{C}'}$ é ℓ, u -viável.

Exemplo 2.2 Um grafo G com limites ℓ, u indicados nas arestas para o qual não existe circulação inteira consistente cuja função peso é ℓ, u -viável. Ao lado, uma circulação fracionária consistente cuja função peso é ℓ, u -viável.

Algoritmo 3 (transforma solução fracionária em inteira) Recebe um grafo G , uma orientação D , uma circulação $\mathcal{C}$ consistente com D e devolve uma circulação inteira $\mathcal{C}'$ consistente com D tal que $\lfloor p_{\mathcal{C}}(e) \rfloor \leq p_{\mathcal{C}'}(e) \leq \lceil p_{\mathcal{C}}(e) \rceil$ para toda aresta e .

O algoritmo é iterativo. Cada iteração começa com uma função q de AG em $\mathbb{Q}^+$ que satisfaz as condições de Hoffman tal que $\lfloor p_{\mathcal{C}}(e) \rfloor \leq q(e) \leq \lceil p_{\mathcal{C}}(e) \rceil$ para toda aresta e . A primeira iteração começa com $p_{\mathcal{C}}$ no papel de q . Cada iteração consiste em:

Caso 1: q é uma função inteira.

Submeta G, D e q ao algoritmo 2 que devolve uma circulação $\mathcal{C}'$.

Pare e devolva $\mathcal{C}'$.

Caso 2: q não é uma função inteira.

Seja C um ciclo tal que $q(e) \notin \mathbb{Z}^+$ para toda aresta e em AC . Seja t_1 o número $\min\{\lceil q(e) \rceil - q(e) : e \in \vec{AC}\}$ e t_2 o número $\min\{q(e) - \lfloor q(e) \rfloor : e \in \overleftarrow{AC}\}$. Seja t o mínimo entre t_1 e t_2 e q' a função que coincide com q exceto nas arestas de C , sendo que $q'(e) = q(e) + t$ para toda aresta e em $\vec{AC}$ e $q'(e) = q(e) - t$ para toda aresta e em $\overleftarrow{AC}$. Comece nova iteração com q' no papel de q .

Para mostrar que o algoritmo acima funciona, é preciso mostrar que no caso 2 realmente existe um ciclo C tal que $q(e) \notin \mathbb{Z}^+$ para toda aresta e em AC . Como q não é uma função inteira, existe pelo menos uma aresta e_0 em AG tal que $q(e_0) \notin \mathbb{Z}^+$. Seja v a ponta final de e_0 . Como q satisfaz as condições de Hoffman, temos que $q(\delta^-(v)) - q(\delta^+(v)) = 0$, ou seja, é inteiro. Logo existe uma aresta e_1 em $\delta(v)$ distinta de e_0 tal que $q(e_1) \notin \mathbb{Z}^+$. A iteração desta construção produz o ciclo C desejado. Note que o ciclo C resultante desta construção não é necessariamente consistente com D .

No caso 1, claramente G , D e q satisfazem as condições de aplicabilidade do algoritmo 2. A circulação C' é inteira pois q é inteira. Além disso, C' é consistente com D e $\lfloor p_c(e) \rfloor \leq p_{C'}(e) \leq \lceil p_c(e) \rceil$ para toda aresta e pois $p_c = q$.

É preciso verificar também que no início de cada iteração q é uma função de AG em $\mathbb{Q}^+$ que satisfaz as condições de Hoffman tal que $\lfloor p_c(e) \rfloor \leq q(e) \leq \lceil p_c(e) \rceil$ para toda aresta e . A hipótese vale no início da primeira iteração pois $q = p_c$. Supondo que valha no início de uma iteração em que ocorre o caso 2, vamos mostrar que vale também para q' . Primeiramente, como q é uma função de AG em $\mathbb{Q}^+$, temos que t_1 e t_2 são racionais positivos, ou seja, t é um racional positivo. Além disso, como $t \leq t_2 \leq q(e)$ para toda aresta e , temos que q' é uma função de AG em $\mathbb{Q}^+$. Como q satisfaz as condições de Hoffman, temos que q' também satisfaz as condições de Hoffman. Pela escolha de t , temos que $\lfloor p_c(e) \rfloor \leq \lfloor q(e) \rfloor \leq q'(e) \leq \lceil q(e) \rceil \leq \lceil p_c(e) \rceil$ para toda aresta e , como queríamos demonstrar.

Em cada iteração, q' tem pelo menos uma componente inteira a mais que q , logo o número de iterações do algoritmo é no máximo m , onde m é o número de arestas de G . Se G está representado pelas listas de adjacências, podemos determinar se q é uma função inteira ou não, construir C tal que $q(e) \notin \mathbb{Z}^+$ para

toda aresta e em AC , calcular t_1, t_2 e q' em tempo $O(m)$. No caso 1, executamos o algoritmo 2 cuja complexidade é $O(m^2)$. Portanto o algoritmo 3 funciona e tem complexidade $O(m^2)$.

2.4 Coloração de circulações consistentes

Little, Tutte e Younger mostraram [31] que toda circulação inteira consistente $\mathcal{C}$ possui uma t -coloração $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_t$, onde t é o valor máximo de p_c e $\mathcal{C} \cup \mathcal{C}_i$ é consistente para todo i . Nesta seção, descreveremos um algoritmo que demonstra esse fato de uma outra maneira.

Observe que se p_c assume valor máximo em uma aresta então esta aresta deve ser usada por todas as circulações $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_t$. O algoritmo primeiramente constrói t funções $q_1, q_2, \dots, q_t$ que serão as funções peso das circulações $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_t$. A partir dessas funções, através do algoritmo 2, obtém as circulações consistentes. Cada uma das funções é obtida em uma iteração do algoritmo através de uma execução do algoritmo de Hoffman. Os limites utilizados na chamada do algoritmo de Hoffman são escolhidos de forma que a função devolvida seja inteira, 0,1-viável e valha 1 em todas as arestas onde p_c é máxima.

Algoritmo 4 (coloração de circulações inteiras consistentes) *Recebe um grafo G , uma orientação D , uma circulação inteira $\mathcal{C}$ consistente com D e devolve uma t -coloração $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_t$ para $\mathcal{C}$ tal que t é o valor máximo de p_c e $\mathcal{C}_i$ é consistente com D para todo i .*

O algoritmo é iterativo. Cada iteração começa com um número i , uma função q de AG em $\mathbb{Z}^+$ e funções $q_1, q_2, \dots, q_i$ de AG em $\{0, 1\}$ tais que $q, q_1, q_2, \dots, q_i$ satisfazem as condições de Hoffman, $p_c = q + q_1 + q_2 + \dots + q_i$ e i é t menos o valor máximo de q . A primeira iteração começa com 0 e p_c nos papéis de i e q . Cada iteração consiste em:

Caso 1: $q = 0$.

Para todo j em $\{1, 2, \dots, i\}$, submeta G, D e q_j ao algoritmo 2, que devolve uma circulação inteira $\mathcal{C}_j$ consistente com D tal que $p_{\mathcal{C}_j} = q_j$. Pare e devolva $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_i$.

Caso 2: $q \neq 0$.

Seja E_1 o conjunto das arestas onde a função q é máxima e E_2 o conjunto das arestas onde q é nula. Sejam ℓ, u limites tais que ℓ e u valem 1 nas arestas de E_1 , ℓ e u valem 0 nas arestas de E_2 e, nas demais, ℓ vale 0 e u vale 1. Submeta G, D, ℓ, u ao algoritmo de Hoffman, que devolve uma função q_{i+1} ℓ, u -viável com valores em $\mathbf{Z}^+$ que satisfaz as condições de Hoffman. Seja q' a função $q - q_{i+1}$. Comece nova iteração com $i + 1$, q' , $q_1, q_2, \dots, q_{i+1}$ nos papéis de $i, q, q_1, q_2, \dots, q_i$.

No caso 1, como $q = 0$ e i é t menos o valor máximo de q , temos que $i = t$. Como $q_1, q_2, \dots, q_t$ são funções de AG em $\{0, 1\}$ que satisfazem as condições de Hoffman e a igualdade $p_c = q_1 + q_2 + \dots + q_i$, temos que $C_1, C_2, \dots, C_t$ é uma t -coloração para C . Podemos submeter G, D e q_j ao algoritmo 2 pois eles satisfazem as condições de aplicabilidade desse. Claramente, $C_1, C_2, \dots, C_t$ são circulações inteiras consistentes com D cujas funções peso são $q_1, q_2, \dots, q_t$, respectivamente. Portanto o algoritmo 4 em questão de fato devolve o que prometeu.

É preciso mostrar que, no início de cada iteração, q é uma função de AG em $\mathbf{Z}^+$ que satisfaz as condições de Hoffman e que $q_1, q_2, \dots, q_i$ são funções de AG em $\{0, 1\}$ que satisfazem as condições de Hoffman e a igualdade $p_c = q + q_1 + q_2 + \dots + q_i$, onde i é t menos o valor máximo de q . A primeira iteração começa com p_c no papel de q e $i = 0$. Como a função peso de toda circulação consistente satisfaz as condições de Hoffman, a hipótese vale no início da primeira iteração. Vamos supor que a hipótese valha no início de uma iteração onde ocorre o caso 2 e mostrar que vale para q' e $q_1, q_2, \dots, q_{i+1}$. Supondo que o algoritmo de Hoffman de fato devolva uma função q_{i+1} , temos que q_{i+1} é uma função de AG em $\mathbf{Z}^+$. Logo q' é uma função de AG em $\mathbf{Z}$. Vamos mostrar agora que q' é uma função positiva. Para isso observe que, $q_{i+1} \leq u$, pois q_{i+1} é ℓ, u -viável. Como, para toda aresta e , $u(e) = 0$ se $q(e) = 0$ e $u(e) = 1$ se $q(e) > 0$, e ainda q é uma função inteira, temos que $u \leq q$. Assim $q' = q - q_{i+1} \geq q - u \geq 0$, ou seja, q' é de fato uma função positiva. Como q e q_{i+1} satisfazem as condições de Hoffman, temos que $q'(\delta^+(X)) = (q - q_{i+1})(\delta^+(X)) = q(\delta^+(X)) - q_{i+1}(\delta^+(X)) = q(\delta^-(X)) - q_{i+1}(\delta^-(X)) = (q - q_{i+1})(\delta^-(X)) = q'(\delta^-(X))$ para todo corte X , ou seja, q' realmente satisfaz as condições de Hoffman. Como ocorreu o caso 2, $E_1 \neq \emptyset$. Logo, $\ell(e) > 0$ para alguma aresta e . Além disso, $u(e) \leq 1$ para toda

aresta e e como q_{i+1} é ℓ, u -viável, temos que $\ell \leq q_{i+1} \leq u$, ou seja, q_{i+1} é realmente uma função com valores em $\{0, 1\}$. Claramente, q_{i+1} satisfaz as condições de Hoffman. É preciso mostrar também que i é t menos o valor máximo de q no começo de cada iteração, ou seja, que o valor máximo de q' é um a menos que o de q . Primeiramente observe que, como $q_{i+1} \geq \ell$, temos que $q' \leq q - \ell$. Mas

$$(q - \ell)(e) = q(e) - \ell(e) = \begin{cases} q(e) - 1 & \text{se } e \in E_1, \\ q(e) & \text{se } e \notin E_1. \end{cases}$$

Pela definição de E_1 e como q é inteira, temos que $(q - \ell)(e) \leq t' - 1$ para toda aresta e , onde t' é o valor máximo de q' . Então, basta observar que $q_{i+1} \geq \ell \geq 0$ e, como $p_c = q + q_1 + q_2 + \dots + q_i$ e $q' = q - q_{i+1}$, temos que $p_c = q' + q_1 + q_2 + \dots + q_{i+1}$. Com isso, demonstramos que a hipótese vale para q' e $q_1, q_2, \dots, q_{i+1}$.

Finalmente, vamos mostrar que as condições para a aplicação do algoritmo de Hoffman no caso 2 estão satisfeitas. É óbvio que $\ell \leq u$. Resta mostrar que $\ell(\delta^-(X)) \leq u(\delta^+(X))$ para todo corte X . Seja X um corte e t' o valor máximo de q . Pela definição de ℓ e u , temos que $\ell(\delta^-(X)) = |E_1 \cap \delta^-(X)|$ e $u(\delta^+(X)) = |\delta^+(X) - E_2|$. Como q satisfaz as condições de Hoffman, temos que $t'|E_1 \cap \delta^-(X)| \leq q(\delta^-(X)) = q(\delta^+(X)) \leq t'|\delta^+(X) - E_2|$. Logo, $\ell(\delta^-(X)) \leq u(\delta^+(X))$, como queríamos demonstrar.

Para fazer a análise do tempo gasto pelo algoritmo, introduziremos a noção de tamanho de uma circulação. O *tamanho de uma circulação* $\mathcal{C}$, denotado por $\|\mathcal{C}\|$, é a soma dos pesos das arestas, isto é,

$$\|\mathcal{C}\| = \sum \{p_c(e) : e \in AG\}.$$

O algoritmo termina depois de t iterações, pois a cada iteração o valor máximo da função q diminui de 1. No caso 2, o cálculo de E_1 e E_2 , assim como a construção de ℓ, u , pode ser feito em tempo $O(m)$. Como já vimos, o algoritmo de Hoffman tem complexidade $O(\ell(AG) \cdot m)$. Neste caso, então, pode ser executado em tempo $O(m^2)$, pois $\ell \leq 1$. O cálculo de q' pode ser feito em tempo $O(m)$. No caso 1, executamos t vezes o algoritmo 2 cuja complexidade é $O(m^2)$. Logo, o algoritmo 4 tem complexidade $O(tm^2)$.

O algoritmo 4 reduz o problema $P3^+$ ao problema $P3$ pois, dado um grafo G , limites ℓ, u e um natural k , resolver $P3^+(G, \ell, u, k)$ é equivalente a resolver $P3(G, \ell, u')$, onde $u'(e)$ é o mínimo entre $u(e)$ e k para toda aresta e .

Capítulo 3

O problema P2 e o algoritmo de Seymour

Existem condições necessárias para que o problema P2 tenha solução que são semelhantes às condições de Hoffman para o problema P1. Seymour demonstrou que tais condições são suficientes para que a relaxação racional do problema P2 tenha solução. Neste capítulo descreveremos as condições de Seymour e um algoritmo que resolve a relaxação racional do problema P2, baseado na demonstração de Seymour. Compararemos este algoritmo com o algoritmo que resolve P1. No fim do capítulo discutiremos e comentaremos alguns resultados e conjecturas relacionadas ao problema P2.

3.1 As condições de Seymour

Seja G um grafo e ℓ, u limites em G . Existe uma condição necessária, semelhante à do problema P1, para que o problema P2 tenha solução:

$$\ell(e) \leq u(\delta X - e) \text{ para todo corte } X \text{ e toda aresta } e \text{ em } \delta(X),$$

onde abreviamos $\delta(X) - \{e\}$ por $\delta X - e$ para simplificar. Comparando com a condição de Hoffman, é como se e estivesse orientada em um sentido no corte X e as demais arestas do corte no outro sentido. Esta condição é necessária pois todo ciclo que usa e , usa mais alguma aresta do corte X . Ou seja, $p_c(e) \leq p_c(\delta X - e)$ e, se p_c é ℓ, u -viável, $\ell(e) \leq p_c(e) \leq p_c(\delta X - e) \leq u(\delta X - e)$.

Seymour demonstrou [43] que se $\ell \leq u$, tal condição é suficiente para que o problema P2r tenha solução. No entanto a condição não é suficiente para que

o problema P2 tenha solução (veja exemplo 3.1). A demonstração de Seymour consta de duas etapas que descreveremos logo depois de algumas definições.

Exemplo 3.1 Um grafo G e uma função inteira que é função peso apenas de circulações fracionárias. Ao lado uma circulação cuja função peso é a apresentada, constituída por quatro ciclos, cada um com multiplicidade $1/2$.

Seja X um corte e f uma aresta em $\delta(X)$. Os limites ℓ, u satisfazem a condição de Seymour para f e X se $\ell(f) \leq u(\delta X - f)$. Se ℓ, u satisfazem as condições de Seymour para f e para todo corte Y tal que $f \in \delta(Y)$ então dizemos que ℓ, u satisfazem as condições de Seymour para f . Se ℓ, u satisfazem as condições de Seymour para toda aresta de G então dizemos simplesmente que ℓ, u satisfazem as condições de Seymour.

Analogamente, dizemos que uma função q de AG em $\mathbb{Q}^+$ satisfaz as condições de Seymour para f e X se $q(f) \leq q(\delta X - f)$. Dizemos que q satisfaz as condições de Seymour para f se satisfaz as condições de Seymour para f e para todo corte Y tal que $f \in \delta(Y)$. Dizemos que q satisfaz as condições de Seymour se satisfaz as condições de Seymour para toda aresta de G .

A validade das condições de Seymour também não implica que $\ell \leq u$.

Exemplo 3.2 Os limites ℓ, u apresentados abaixo satisfazem as condições de Seymour, porém existe uma aresta em que o limite inferior é maior que o limite superior.

O algoritmo de Seymour para o problema P2r tem duas etapas: a primeira faz a verificação das condições de Seymour e constrói uma função ℓ, u -viável q que é função peso de uma circulação; a segunda constrói uma circulação cuja

função peso é a função q . Portanto a primeira etapa corresponde ao algoritmo de Hoffman e a segunda ao algoritmo 2. A idéia utilizada na primeira etapa do algoritmo de Seymour é essencialmente diferente da idéia do algoritmo de Hoffman. No entanto, é possível fazer uma analogia entre a segunda etapa e o algoritmo 2. Tal analogia será feita na seção 3.3, onde começaremos a descrição da segunda etapa do algoritmo de Seymour.

3.2 O algoritmo de Seymour: primeira etapa

O algoritmo abaixo é a primeira etapa do algoritmo de Seymour; ele encontra uma função q que satisfaz as condições de Seymour ou uma prova de que P2r não tem solução.

Dizemos que *os limites ℓ', u' são equivalentes aos limites ℓ, u* se, para toda aresta e tal que ℓ, u satisfaz as condições de Seymour para e , temos que os limites ℓ', u' satisfazem as condições de Seymour para e .

Algoritmo 5 (primeira etapa do algoritmo de Seymour) *Recebe um grafo G , limites ℓ_0, u_0 tais que $\ell_0 \leq u_0$ e devolve uma função q ℓ_0, u_0 -viável com valores em $\mathbb{Q}^+$ que satisfaz as condições de Seymour*

ou

um corte X e uma aresta $\bar{e}$ em $\delta(X)$ tais que $\ell_0(\bar{e}) > u_0(\delta X - \bar{e})$.

Se ℓ_0 e u_0 forem inteiras, a função q devolvida pelo algoritmo é inteira.

Cada iteração do algoritmo começa com um conjunto E de arestas e limites ℓ, u tais que $\ell_0 \leq \ell \leq u \leq u_0$, ℓ, u são equivalentes a ℓ_0, u_0 , ℓ coincide com u em E e ℓ, u satisfazem a condição de Seymour para toda aresta em E . A primeira iteração começa com $\emptyset, \ell_0, u_0$ nos papéis de E, ℓ, u , respectivamente. Cada iteração consiste em:

Caso 1: $E = AG$.

Pare e devolva ℓ .

Caso 2: $E \neq AG$.

Seja e_0 um elemento de $AG - E$. Escolha um corte X_0 de tal modo que $e_0 \in \delta(X_0)$ e $u(\delta X_0 - e_0) = \min\{u(\delta X - e_0) : \delta(X) \ni e_0\}$.

Seja t_0 o número $u(\delta X_0 - e_0)$.

Caso 2.1: $t_0 < \ell(e_0)$.

Seja X_1 um corte tal que $e_0 \in \delta(X_1)$ e $u_0(\delta X_1 - e_0) = \min\{u_0(\delta X - e_0) : \delta(X) \ni e_0\}$. Pare e devolva X_1 e e_0 .

Caso 2.2: $t_0 \geq \ell(e_0)$.

Seja t o mínimo entre t_0 e $u(e_0)$. Sejam ℓ', u' limites tais que $\ell'(e_0) = u'(e_0) = t$ e, nas demais arestas, ℓ' e u' coincidem com ℓ e u , respectivamente. Comece uma nova iteração com $E \cup \{e_0\}$, ℓ', u' nos papéis de E, ℓ, u .

Para mostrar que o algoritmo funciona é preciso verificar que, no início de cada iteração, $\ell_0 \leq \ell \leq u \leq u_0$, ℓ, u são equivalentes a ℓ_0, u_0 , ℓ coincide com u em E e ℓ, u satisfazem a condição de Seymour para toda aresta em E . Certamente a hipótese vale no início da primeira iteração. Supondo que valha no início de uma iteração em que o caso 2.2 é executado, vamos mostrar que a hipótese vale para $E \cup \{e_0\}$, ℓ', u' .

Primeiramente note que ℓ coincide com ℓ' em $AG - e_0$. Analogamente, u coincide com u' em $AG - e_0$. Portanto, para mostrar que $\ell_0 \leq \ell' \leq u' \leq u_0$, basta mostrar que $\ell(e_0) \leq \ell'(e_0) \leq u'(e_0) \leq u(e_0)$ pois $\ell_0 \leq \ell \leq u \leq u_0$. Como $\ell(e_0) \leq u(e_0)$, $\ell(e_0) \leq t_0$ e $\ell'(e_0) = u'(e_0) = \min\{t_0, u(e_0)\}$, temos que realmente $\ell(e_0) \leq \ell'(e_0) \leq u'(e_0) \leq u(e_0)$.

Agora vamos mostrar que ℓ', u' são equivalentes a ℓ_0, u_0 . Como ℓ, u são equivalentes a ℓ_0, u_0 , basta demonstrar que ℓ', u' são equivalentes a ℓ, u . Começaremos mostrando que ℓ', u' satisfazem as condições de Seymour para e_0 . Para isso, basta observar que $\ell'(e_0) = u'(e_0) \leq t_0 = \min\{u(\delta X - e_0) : \delta(X) \ni e_0\} = \min\{u'(\delta X - e_0) : \delta(X) \ni e_0\}$, pois u e u' coincidem em $AG - e_0$. Agora seja e uma aresta em $AG - e_0$. Supondo que ℓ, u satisfazem as condições de Seymour para e , ou seja, que $\ell(e) \leq \min\{u(\delta X - e) : \delta(X) \ni e\}$, vamos mostrar que $\ell'(e) \leq \min\{u'(\delta X - e) : \delta(X) \ni e\}$, seja X_1 o corte tal que $e \in \delta(X_1)$ e $u'(\delta X_1 - e) = \min\{u'(\delta X - e) : \delta(X) \ni e\}$. Basta mostrar que $\ell'(e) \leq u'(\delta X_1 - e)$. Temos duas possibilidades:

Caso 1: $e_0 \notin \delta(X_1)$ ou $u'(e_0) = u(e_0)$.

Neste caso, $\ell'(e) = \ell(e) \leq \min\{u(\delta X - e) : \delta(X) \ni e\} \leq u(\delta X_1 - e) = u'(\delta X_1 - e)$.

Caso 2: $e_0 \in \delta(X_1)$ e $u'(e_0) \neq u(e_0)$.

Caso 2.1: $e \in \delta(X_0)$.

Neste caso, $\ell'(e) = \ell(e) \leq u(e) \leq u(\delta X_0 - e_0) = t_0 = u'(e_0) \leq u'(\delta X_1 - e)$.

Caso 2.2: $e \notin \delta(X_0)$.

Seja X_2 aquele entre $X_0 \cap X_1$, $X_0 - X_1$, $X_1 - X_0$ e $\overline{X_0 \cup X_1}$ tal que $e \in \delta(X_2)$, $e_0 \notin \delta(X_2)$. Note que $\delta(X_2) \subseteq \delta(X_0) \cup \delta(X_1)$. Logo $\ell'(e) = \ell(e) \leq u(\delta X_2 - e) \leq u(\delta X_1 - \{e, e_0\}) + u(\delta X_0 - e_0) = u(\delta X_1 - e) - u(e_0) + u'(e_0) = u'(\delta X_1 - e)$.

Assim demonstramos que se ℓ, u satisfazem as condições de Seymour para e então ℓ', u' também satisfazem as condições de Seymour para e , completando assim a demonstração de que os limites ℓ', u' são equivalentes a ℓ, u .

Como claramente ℓ' coincide com u' em $E \cup \{e_0\}$, falta apenas mostrar que ℓ', u' satisfazem as condições de Seymour para toda aresta em $E \cup \{e_0\}$. Já vimos que os limites ℓ', u' satisfazem as condições de Seymour para e_0 . Vimos também que ℓ', u' são equivalentes a ℓ, u . Como ℓ, u satisfazem as condições de Seymour para toda aresta em E , temos que ℓ', u' realmente satisfazem as condições de Seymour para toda aresta em $E \cup \{e_0\}$.

Com isso, os invariantes valem para ℓ', u' , como queríamos demonstrar.

Caso os limites ℓ, u sejam funções inteiras, o valor de t_0 é inteiro e portanto as funções ℓ', u' são inteiras. Logo, a função q calculada pelo algoritmo é de fato inteira.

Se o algoritmo termina no caso 1 então ℓ satisfaz as condições de Seymour e $\ell_0 \leq \ell \leq u_0$. Se o algoritmo termina no caso 2.1 então, como ℓ, u são equivalentes a ℓ_0, u_0 e não satisfazem as condições de Seymour para e_0 , temos que ℓ_0, u_0 também não satisfazem as condições de Seymour para e_0 , ou seja, $\ell_0(e_0) > \min\{u_0(\delta X - e_0) : \delta(X) \ni e_0\}$. Portanto o par X_1, e_0 que o algoritmo devolve no caso 2.1 tem as propriedades previstas no enunciado do algoritmo.

Com isso encerramos a prova da corretude do algoritmo. A cada iteração, o conjunto E aumenta de um elemento, logo o número máximo de iterações do algoritmo é m , onde m é o número de arestas de G . Um algoritmo que calcula

fluxo máximo e corte mínimo entre dois vértices pode ser usado para obter o valor de t_0 : basta submeter o grafo $G - e_0$, os vértices x, y que são as pontas de e_0 , e u como função capacidade nas arestas ao tal algoritmo. O corte mínimo obtido corresponde a X_0 , e t_0 é o valor do fluxo máximo obtido. De forma análoga, podemos determinar X_1 no caso 2.1. Portanto usando um algoritmo de fluxo máximo [32] cuja complexidade seja $O(n^3)$, onde n é o número de vértices de G , o algoritmo 5 tem complexidade $O(mn^3)$.

3.3 O algoritmo de Seymour: segunda etapa

Nesta seção fazemos a descrição da segunda etapa do algoritmo de Seymour. Essa etapa constrói uma circulação C tal que $p_c = q$, onde q é a função construída na primeira etapa. Podemos fazer uma analogia entre essa etapa e o algoritmo 2. Ambos são um processo iterativo: a cada iteração um ciclo é escolhido e decrementado da função q . A função resultante do decréscimo não deve ter componentes negativas e deve continuar satisfazendo as condições de Hoffman ou de Seymour, conforme o caso. Em ambos os casos, portanto, o valor de decréscimo não deve ultrapassar o valor mínimo da função q nas arestas do ciclo. Além disso, para garantir a convergência do algoritmo, o valor de decréscimo deve ser máximo e estritamente positivo. Logo, nos dois casos, o valor de q nas arestas do ciclo deve ser não nulo. No algoritmo 2, o ciclo é orientado. Na segunda etapa do algoritmo de Seymour, o ciclo usa uma das aresta com q máximo incidente a cada um de seus vértices. No algoritmo 2, tomando como valor de decréscimo o mínimo da função q nas arestas do ciclo, as condições de Hoffman continuarão automaticamente satisfeitas. O mesmo não ocorre na segunda etapa do algoritmo de Seymour.

Seja q uma função de AG em $\mathbf{Q}^+$. Dado um racional t e um ciclo C , denotaremos por $q + tC$ a função que coincide com q exceto nas arestas de C , sendo que $q'(e) = q(e) + t$ para toda aresta e de C . Analogamente, definimos a função $q - tC$.

Dizemos que uma função q que satisfaz as condições de Seymour é *justa em um corte* X se existe uma aresta e em $\delta(X)$ tal que $q(e) = q(\delta X - e)$. Neste caso, dizemos que e é *especial em* X . Se $X = \{v\}$, dizemos simplesmente que q é justa em v .

Seja q uma função justa em X e f uma aresta especial em $\delta(X)$. Dada uma circulação $\mathcal{C}$ tal que $p_{\mathcal{C}} = q$, todo ciclo de $\mathcal{C}$ que utiliza alguma aresta em $\delta(X)$, utiliza a aresta f .

Dado um grafo G e uma função q de AG em $\mathbf{Q}$, denotamos por $Sup_G(q)$ e chamamos de *suporte de q* , o conjunto $\{e \in AG : q(e) \neq 0\}$.

Um *circuito* é uma trilha $\langle v_0, e_1, v_1, \dots, e_k, v_k \rangle$ tal que $v_0, v_1, \dots, v_{k-1}$ são dois a dois distintos e $v_0 = v_k$.

Dada uma função q de AG em $\mathbf{Q}^+$ e um vértice v , dizemos que uma aresta $\bar{e}$ em $\delta(X)$ é *máxima em v* se $q(\bar{e}) = \max\{q(e) : e \in \delta(v)\}$. Um circuito C é *adequado em relação a q* se $AC \subseteq Sup_G(q)$ e C usa uma aresta máxima em v para todo vértice v em VC .

Algoritmo 6 (segunda etapa do algoritmo de Seymour) *Recebe um grafo G , uma função q de AG em $\mathbf{Q}^+$ que satisfaz as condições de Seymour e devolve uma circulação cuja função peso é q .*

O algoritmo é recursivo.

Caso 1: $q = 0$.

Devolva a circulação vazia.

Caso 2: $q \neq 0$.

Seja φ uma função de VG em AG tal que $\varphi(v) \in \delta(v)$ e $q(\varphi(v)) = \max\{q(e) : e \in \delta(v)\}$. Submeta $G[Sup_G(q)]$ e φ ao subalgoritmo 6.1, descrito a seguir, que devolve um circuito C tal que $\varphi(v) \in AC$ para todo vértice v em VC e $AC \subseteq Sup_G(q)$. Submeta G , q e C ao subalgoritmo 6.3, que devolve o maior número t_0 tal que a função $q - t_0C$ é positiva e satisfaz as condições de Seymour. Além disso, se $t_0 < \min\{q(e) : e \in AC\}$ então o subalgoritmo 6.3 devolve também um corte X_0 e uma aresta $\bar{e}$ tais que $q - t_0C$ é justa em X_0 , $\bar{e}$ é especial em X_0 e se $t_0 = 0$ então X_0 não é trivial. Seja q' a função $q - t_0C$.

Caso 2.1: $t_0 = \min\{q(e) : e \in AC\}$ ou $(t_0 < \min\{q(e) : e \in AC\}$ e o corte X_0 é trivial).

Submeta G e q' ao algoritmo 6, que devolve uma circulação $\mathcal{C}'$ cuja função peso é q' . Seja $\mathcal{C}$ a circulação $\mathcal{C}' + t_0C$. Devolva $\mathcal{C}$.

Caso 2.2: $t_0 < \min\{q(e) : e \in AC\}$ e o corte X_0 não é trivial.

Sejam G_1 e G_2 os grafos obtidos de G pela contração de X_0 e $\overline{X_0}$, respectivamente. Para todo i em $\{1, 2\}$, seja q_i a restrição de q' a G_i . Submeta G_i e q_i ao algoritmo 6, que devolve uma circulação C_i cuja função peso é q_i . Submeta G , X_0 , $\bar{e}$, C_1 , C_2 ao subalgoritmo 6.2, que devolve uma circulação C' cuja função peso é q' . Seja C a circulação $C' + t_0 C$. Devolva C .

No caso 1, basta observar que a função peso da circulação vazia é nula, e portanto ao devolver a circulação nula o algoritmo está se comportando como prometeu.

No caso 2, desde que o subalgoritmo 6.3 funcione, o valor de t_0 é tal que $q - t_0 C$ satisfaz as condições de Seymour. Assim, como $q' = q - t_0 C$, podemos chamar o algoritmo com G e q' . No caso 2.2, como q' satisfaz as condições de Seymour e todo corte de G_1 e G_2 é um corte de G , temos que q_1 e q_2 também satisfazem as condições de Seymour. Logo podemos chamar o algoritmo com G e q_i para todo i em $\{1, 2\}$.

Se ocorre o caso 2.1 então C' é tal que $p_{C'} = q'$. Se ocorre o caso 2.2, desde que o subalgoritmo 6.2 funcione, C' também é tal que $p_{C'} = q'$. Tanto no caso 2.1 quanto no caso 2.2, temos que C é uma circulação cuja função peso é q .

Chamamos uma aresta e de *aresta de corte* se existe um 1-corte X tal que e é a única aresta em $\delta(X)$. No caso 2.2, $\varphi(v) \in \delta(v)$ e não é uma aresta de corte em $G[Sup_G(q)]$ pois q satisfaz as condições de Seymour e $q(\varphi(v)) \neq 0$ para todo vértice v . Com isso, como veremos, garantimos que as condições de aplicabilidade do subalgoritmo 6.1 estão satisfeitas. As funções p_{C_1} e p_{C_2} coincidem nas arestas de $\delta(X_0)$, pois $p_{C_1} = q_1$ e $p_{C_2} = q_2$, onde q_i é a restrição da função q ao grafo G_i para todo i . Além disso, como q é justa em X_0 , temos que p_{C_1} e p_{C_2} são justas em X_0 . Todo ciclo em C_1 e C_2 que usa arestas em $\delta(X_0)$, usa a aresta especial $\bar{e}$ de X_0 . Assim garantimos que as condições de aplicabilidade do subalgoritmo 6.2 também estão satisfeitas. Desde que o subalgoritmo 6.1 funcione, o circuito C é de fato um circuito adequado em relação a q . Logo as condições de aplicabilidade do subalgoritmo 6.3 também estão satisfeitas.

O número t_0 devolvido pelo subalgoritmo 6.3 é tal que a função $q - t_0C$ tem mais componentes nulas que a função q ou é justa em mais cortes que a função q . É basicamente este o fato que garante a convergência do algoritmo 6. Sejam m e n o número de arestas e o número de vértices de G , respectivamente. A fim de analisar a complexidade do algoritmo, seja $NJ_G(q)$ o conjunto $\{v \in VG : \text{a função } q \text{ não é justa em } v\}$, e sejam $c_{2.1}(G, q)$ e $c_{2.2}(G, q)$ as funções definidas por $c_{2.1}(G, q) = (n - 2)|Sup_G(q)| + |NJ_G(q)|$ e $c_{2.2}(G, q) = n - 3$. Observe que $c_{2.2}(G, q)$ depende apenas do número de vértices do grafo G .

Vamos provar por indução no número $n + |Sup_G(q)| + |NJ_G(q)|$ que o número de execuções do caso 2.1 e o número de execuções do caso 2.2 são limitados superiormente por $c_{2.1}(G, q)$ e $c_{2.2}(G, q)$, respectivamente, para grafos com mais que dois vértices. Observe que o caso 2.2 certamente não ocorre para grafos com no máximo dois vértices.

Suponha que o algoritmo 6 é chamado com G e q tais que $Sup_G(q) = \emptyset$. Então o algoritmo executa o caso 1 e portanto os casos 2.1 e 2.2 não são executados nenhuma vez. Segue que o número de execuções do caso 2.1 e o número de execuções do caso 2.2 são limitados superiormente por $c_{2.1}(G, q)$ e $c_{2.2}(G, q)$, respectivamente, como queríamos demonstrar.

Suponha agora que o algoritmo 6 é chamado com G e q tais que $Sup_G(q) \neq \emptyset$. Vamos supor que, para toda chamada do algoritmo 6 com parâmetros H e p tais que $|VH| + |Sup_H(p)| + |NJ_H(p)| < n + |Sup_G(q)| + |NJ_G(q)|$, o número de execuções dos casos 2.1 e 2.2 são limitados superiormente por $c_{2.1}(H, p)$ e $c_{2.2}(H, p)$, respectivamente. Queremos mostrar o mesmo para G e q .

Observe que certamente ocorre o caso 2, pois $Sup_G(q) \neq \emptyset$, e que $Sup_G(q') \subseteq Sup_G(q)$ e $NJ_G(q') \subseteq NJ_G(q)$. No caso 2.2, $Sup_{G_i}(q_i) \subseteq Sup_G(q')$ e $NJ_{G_i}(q_i) \subseteq NJ_G(q')$ para todo i .

Denotamos por $e_{2.1}(G, q)$ e $e_{2.2}(G, q)$ o número exato de execuções dos casos 2.1 e 2.2, respectivamente.

No caso 2.1, temos que se $t_0 = \min\{q(e) : e \in AC\}$ então $|Sup_G(q')| < |Sup_G(q)|$ senão X_0 é trivial e $|NJ_G(q')| < |NJ_G(q)|$ já que $t_0 > 0$ pelo subalgoritmo 6.3. Ou seja, $n + |Sup_G(q')| + |NJ_G(q')| < n + |Sup_G(q)| + |NJ_G(q)|$ e portanto, de acordo com a nossa hipótese, $e_{2.1}(G, q') \leq c_{2.1}(G, q')$ e $e_{2.2}(G, q') \leq c_{2.2}(G, q')$. Mas então temos que $e_{2.1}(G, q) = e_{2.1}(G, q') + 1 \leq c_{2.1}(G, q') + 1 \leq c_{2.1}(G, q)$ e $e_{2.2}(G, q) = e_{2.2}(G, q') \leq c_{2.2}(G, q') = c_{2.2}(G, q)$.

No caso 2.2, sejam n_1 e n_2 os números de vértices de G_1 e G_2 , respectivamente. Como X_0 não é trivial e G_i é a contração de X_0 ou $\overline{X_0}$, temos que $n > n_i \geq 3$. Logo, $n_i + |Sup_{G_i}(q_i)| + |NJ_{G_i}(q_i)| < n + |Sup_G(q)| + |NJ_G(q)|$ para todo i e portanto, de acordo com a nossa hipótese, $e_{2.1}(G_i, q_i) \leq c_{2.1}(G_i, q_i)$ e $e_{2.2}(G_i, q_i) \leq c_{2.2}(G_i, q_i)$ para todo i . Temos ainda que $n = n_1 + n_2 - 2$. Então

$$\begin{aligned}
 e_{2.1}(G, q) &= e_{2.1}(G_1, q_1) + e_{2.1}(G_2, q_2) \\
 &\leq c_{2.1}(G_1, q_1) + c_{2.1}(G_2, q_2) \\
 &= (n_1 - 2)|Sup_{G_1}(q_1)| + |NJ_{G_1}(q_1)| \\
 &\quad + (n_2 - 2)|Sup_{G_2}(q_2)| + |NJ_{G_2}(q_2)| \\
 &\leq (n_1 - 2)|Sup_G(q)| + |NJ_{G_1}(q_1)| \\
 &\quad + (n_2 - 2)|Sup_G(q)| + |NJ_{G_2}(q_2)| \\
 &\leq (n_1 - 2 + n_2 - 2)|Sup_G(q)| + |NJ_G(q)| \\
 &= (n - 2)|Sup_G(q)| + |NJ_G(q)| \\
 &= c_{2.1}(G, q),
 \end{aligned}$$

pois, como q' é justa em X_0 , $|NJ_{G_1}(q_1)| + |NJ_{G_2}(q_2)| = |NJ_G(q')|$. Além disso,

$$\begin{aligned}
 e_{2.2}(G, q) &= e_{2.2}(G_1, q_1) + e_{2.2}(G_2, q_2) + 1 \\
 &\leq c_{2.2}(G_1, q_1) + c_{2.2}(G_2, q_2) + 1 \\
 &= (n_1 - 3) + (n_2 - 3) + 1 \\
 &= (n_1 + n_2 - 2) - 3 \\
 &= n - 3 \\
 &= c_{2.2}(G, q).
 \end{aligned}$$

Com isso, concluímos que realmente $c_{2.1}(G, q)$ e $c_{2.2}(G, q)$ limitam superiormente o número de execuções do caso 2.1 e o número de execuções do caso 2.2, respectivamente.

Veremos mais tarde que a complexidade dos subalgoritmos 6.1, 6.2 e 6.3 é $O(m(n + m))$, $O(m \cdot q(AG))$ e $O(m^2 n^6)$, respectivamente. O número de execuções dos subalgoritmos 6.1 e 6.3 é $O(c_{2.1}(G, q) + c_{2.2}(G, q))$, ou seja, $O(nm)$. O número de execução do subalgoritmo 6.2 é $O(c_{2.2}(G, q))$, ou seja, $O(n)$. Com isso, o algoritmo 6 produz o resultado esperado em tempo $O(m^3 n^7 + nm \cdot q(AG))$.

3.4 O subalgoritmo 6.1

A cada iteração da segunda etapa do algoritmo de Seymour, como já dissemos, um ciclo conveniente é escolhido para ser acrescentado à circulação que está sendo construída. Nesta seção descrevemos um algoritmo, baseado no método de Seymour e Giles [43], para obter este ciclo.

Subalgoritmo 6.1 (calcula circuito com prescrição de arestas)

Recebe um grafo G , uma função φ de VG em AG tal que $\varphi(v) \in \delta(v)$ e não é uma aresta de corte para todo vértice v . Devolve um circuito C em G tal que $\varphi(v) \in AC$ para todo v em VC .

O algoritmo é recursivo e iterativo. Cada iteração começa com um caminho $\langle v_0, e_1, v_1, \dots, e_k, v_k \rangle$, digamos P , e uma aresta e , denotada também por e_{k+1} , tal que $e \in \delta(v_k) - \{e_k\}$ e $\varphi(v_i)$ é e_i ou e_{i+1} para todo i em $\{1, 2, \dots, k\}$. A primeira iteração começa com o caminho $\langle v_0 \rangle$ e com a aresta $\varphi(v_0)$, onde v_0 é um vértice qualquer de G . Cada iteração consiste em:

Seja w tal que v_k e w são as pontas de e .

Caso 1: $w \notin VP$.

Se $e \neq \varphi(w)$ então seja e' a aresta $\varphi(w)$ senão seja e' uma aresta qualquer incidente a w distinta de e . Comece nova iteração com o caminho $P \langle v_k, e, w \rangle$ e com a aresta e' nos papéis de P e e .

Caso 2: Existe i em $\{0, 1, \dots, k\}$ tal que $w = v_i$ e $e_{i+1} = \varphi(v_i)$.

Seja C o circuito $\langle v_i, e_{i+1}, v_{i+1}, \dots, e_k, v_k, e, w \rangle$. Pare e devolva C .

Caso 3: Existe i em $\{0, 1, \dots, k\}$ tal que $w = v_i$ e $e_{i+1} \neq \varphi(v_i)$.

Seja C o circuito $\langle v_i, e_{i+1}, v_{i+1}, \dots, e_k, v_k, e, w \rangle$ e G' o grafo obtido de G pela contração de VC . Seja u o vértice de G' que corresponde ao conjunto VC e φ' a função de VG' em AG' dada por:

$$\varphi'(v) = \begin{cases} \varphi(v) & \text{se } v \neq u, \\ \varphi(v_i) & \text{se } v = u. \end{cases}$$

Submeta G' e φ' ao algoritmo 6.1, obtendo um circuito C' de G' tal que $\varphi'(v) \in VC'$ para todo vértice v em VC' .

Caso 3.1: $u \notin VC'$.

Pare e devolva C' .

Caso 3.2: $u \in VC'$.

Sejam $u_0, u_1, \dots, u_s$ vértices de G' e $f_1, f_2, \dots, f_s$ arestas de G' tais que $C' = \langle u_0, f_1, u_1, \dots, f_s, u_s \rangle$, $u_1 = u$ e $\varphi(u_1) = f_2$. Seja j em $\{i, \dots, k\}$ tal que v_j é ponta de f_1 em G . Seja P' o ciclo $\langle v_j \rangle$ se $i = j$, $\langle v_j, e_{j+1}, v_{j+1}, \dots, v_k, e_{k+1}, v_i \rangle$ se $\varphi(v_j) = e_{j+1}$ e $i \neq j$, $\langle v_j, e_j, v_{j-1}, \dots, v_{i+1}, e_{i+1}, v_i \rangle$ se $\varphi(v_j) = e_j$ e $i \neq j$. Pare e devolva o ciclo $\langle u_0, f_1, v_j \rangle P' \langle v_i, f_2, u_2, \dots, f_s, u_s \rangle$.

Para mostrar que o algoritmo funciona, primeiramente note que no caso 1, se $e \neq \varphi(w)$, sempre existe uma aresta e' incidente a w distinta de e , pois $\varphi(w)$ não é uma aresta de corte.

Vamos mostrar que, no início de cada iteração, $e \in \delta(v_k) - \{e_k\}$ e $\varphi(v_i)$ é e_i ou e_{i+1} para todo i em $\{1, 2, \dots, k\}$. Na primeira iteração a hipótese vale pois $P = \langle v_0 \rangle$ e $e = \varphi(v_0)$. Supondo que valha no início de uma iteração em que ocorre o caso 1 então claramente a hipótese vale para $P \langle v_k, e, w \rangle$ e e' . Nos demais casos o algoritmo termina sem começar nova iteração.

É preciso verificar também que no caso 3 o algoritmo pode ser chamado com G' e q' . A ponta de $\varphi(v_i)$ distinta de v_i é o vértice v_{i-1} . Como $v_{i-1} \notin VC$, temos que $\varphi(v_i)$ é uma aresta de G' e uma de suas pontas em G' é o novo vértice u , ou seja, $\varphi'(u)$ é uma aresta incidente a u em G' . Além disso, $\varphi'(u)$ não é uma aresta de corte, pois $\varphi'(u) = \varphi(v_i)$. Nos demais vértices, φ' coincide com φ , logo G' e φ' satisfazem as condições de aplicabilidade do subalgoritmo 6.1, como queríamos demonstrar.

Para todo j , temos que $\varphi(v_j) = e_j$ ou $\varphi(v_j) = e_{j+1}$. No caso 2, como $e_{i+1} = \varphi(v_i)$, realmente C é tal que $\varphi(v) \in VC$ para todo vértice v em VC . No caso 3, C' é tal que $\varphi(v) \in VC'$ para todo vértice v em VC' . Se $u \notin VC'$ então basta observar que C' é um circuito em G . No caso 3.2, seja C'' o ciclo devolvido pelo algoritmo. Claramente $\varphi(v) \in AC''$ para todo vértice v em $VC' - \{u_1\}$. Como $\varphi(v_i) = \varphi'(u) = \varphi(u_1) = f_2$, temos que $\varphi(v_i) \in AC''$. Além disso, $\varphi(v) \in AP'$ para todo v em $VP' - \{v_i\}$, logo $\varphi(v) \in AC''$ para todo vértice v em AC'' . Ou seja, em todos os casos o ciclo devolvido está de acordo com a descrição do algoritmo.

O número de iterações do algoritmo, ou seja, o número de vezes que o caso 1 ocorre, é no máximo n , onde n é o número de vértices de G . No caso 3, o grafo G' tem pelo menos uma aresta a menos que G , o que garante que o número de chamadas recursivas dentro de uma chamada do algoritmo é no máximo m , onde m é o número de arestas de G . A contração de C , a construção de φ e a obtenção de C'' a partir de C' podem ser feitas em tempo $O(m)$. Assim o algoritmo tem complexidade $O(m(n + m))$.

3.5 O subalgoritmo 6.2

Seja G um grafo, X um corte e G_1 e G_2 os grafos obtidos de G pela contração de X e $\overline{X}$. Dadas circulações $\mathcal{C}_1$ e $\mathcal{C}_2$ em G_1 e G_2 , respectivamente, dizemos que $\mathcal{C}_1$ e $\mathcal{C}_2$ *casam em X* se $p_{\mathcal{C}_1}(e) = p_{\mathcal{C}_2}(e)$ para toda aresta e em $\delta(X)$ e existe uma aresta $\bar{e}$ em $\delta(X)$ tal que todo ciclo de $\mathcal{C}_1$ ou de $\mathcal{C}_2$ que usa alguma aresta em $\delta(X)$ usa $\bar{e}$ e apenas mais uma aresta em $\delta(X)$. Neste caso, dizemos que a aresta $\bar{e}$ é uma *aresta de ligação* entre $\mathcal{C}_1$ e $\mathcal{C}_2$.

Nesta seção descrevemos um algoritmo que, dadas circulações $\mathcal{C}_1$ e $\mathcal{C}_2$ em G_1 e G_2 , respectivamente, tais que $\mathcal{C}_1$ e $\mathcal{C}_2$ casam em X , constrói uma circulação $\mathcal{C}$ em G tal que a restrição de $p_{\mathcal{C}}$ a G_i coincide com $p_{\mathcal{C}_i}$ para todo i em $\{1, 2\}$.

Subalgoritmo 6.2 (junta duas circulações que casam em um corte)

Recebe um grafo G , um corte X , circulações $\mathcal{C}_1$ e $\mathcal{C}_2$ nos grafos G_1 e G_2 obtidos de G pela contração de X e $\overline{X}$, respectivamente, tais que $\mathcal{C}_1$ e $\mathcal{C}_2$ casam em X e devolve uma circulação $\mathcal{C}$ em G tal que a restrição de $p_{\mathcal{C}}$ a G_i coincide com $p_{\mathcal{C}_i}$ para todo i em $\{1, 2\}$.

Seja $\bar{e}$ uma aresta de ligação entre $\mathcal{C}_1$ e $\mathcal{C}_2$. Cada iteração do algoritmo começa com circulações $\mathcal{C}$, $\mathcal{C}'_1$ e $\mathcal{C}'_2$ em G , G_1 e G_2 , respectivamente, tais que $\mathcal{C}'_1$ e $\mathcal{C}'_2$ casam em X , $\bar{e}$ é uma aresta de ligação entre $\mathcal{C}'_1$ e $\mathcal{C}'_2$ e, para todo i , $(p_{\mathcal{C}})_i + p_{\mathcal{C}'_i} = p_{\mathcal{C}_i}$, onde $(p_{\mathcal{C}})_i$ é a restrição da função $p_{\mathcal{C}}$ ao grafo G_i . A primeira iteração começa com $\mathcal{C} = \emptyset$, $\mathcal{C}'_1 = \mathcal{C}_1$ e $\mathcal{C}'_2 = \mathcal{C}_2$. Cada iteração consiste em:

Caso 1: $p_{\mathcal{C}'_1} = 0$ e $p_{\mathcal{C}'_2} = 0$.

Pare e devolva $\mathcal{C}$.

Caso 2: $p_{c'_1} \neq 0$ ou $p_{c'_2} \neq 0$.

Ajuste a notação, permutando C'_1 e C'_2 se necessário, de modo que $p_{c'_1} \neq 0$. Seja C_1 um ciclo em C'_1 tal que $m_{c'_1}(C_1) > 0$.

Caso 2.1: $AC_1 \cap \delta(X) = \emptyset$.

Seja t o número $m_{c'_1}(C_1)$. Comece nova iteração com $\mathcal{C} + tC_1$, $C'_1 - tC_1$ e C'_2 nos papéis de $\mathcal{C}$, C'_1 e C'_2 , respectivamente.

Caso 2.2: $AC_1 \cap \delta(X) \neq \emptyset$.

Seja $\bar{e}'$ distinta de $\bar{e}$ em $AC_1 \cap \delta(X)$ e C_2 um ciclo em $\mathcal{C}_2$ que utiliza $\bar{e}'$ tal que $m_{c'_2}(C_2) > 0$. Seja C o ciclo em G cujas arestas são as arestas de $AC_1 \cup AC_2$. Seja t o mínimo entre $m_{c'_1}(C_1)$ e $m_{c'_2}(C_2)$. Comece nova iteração com $\mathcal{C} + tC$, $C'_1 - tC_1$ e $C'_2 - tC_2$ nos papéis de $\mathcal{C}$, C'_1 e C'_2 , respectivamente.

No caso 2.2 existe C_2 em $\mathcal{C}_2$ pois $p_{c'_2}(\bar{e}') = p_{c'_1}(\bar{e}') > 0$. Além disso, como $AC_1 \cap \delta(X) = \{\bar{e}, \bar{e}'\} = AC_2 \cap \delta(X)$, existe também o ciclo C .

Para mostrar que o algoritmo funciona é preciso mostrar que no início de cada iteração $\mathcal{C}$, C'_1 e C'_2 são circulações em G , G_1 e G_2 , respectivamente, tais que C'_1 e C'_2 casam em X , $\bar{e}$ é uma aresta de ligação entre C'_1 e C'_2 e, para todo i , $(p_c)_i + p_{c'_i} = p_{c_i}$. No início da primeira iteração, a hipótese vale certamente. No caso 2, vamos chamar de C' , C'_1 e C'_2 as circulações que serão usadas na nova iteração. Supondo que a hipótese valha para $\mathcal{C}$, C_1 e C_2 , vamos mostrar que vale para C' , C'_1 e C'_2 . Claramente C' , C'_1 e C'_2 são circulações em G , G_1 e G_2 , respectivamente. No caso 2.1, $AC_1 \subseteq AG_1 - \delta(X)$, logo $AC_1 \cap AG_2 = \emptyset$. Além disso, $p_{c'} = p_c + tC_1$, $p_{c'_1} = p_{c'_1} - tC_1$ e $p_{c'_2} = p_{c'_2}$, portanto $p_{c'_1}(e) = p_{c'_2}(e)$ para toda aresta e em $\delta(X)$ e $(p_c)_i + p_{c'_i} = p_{c_i}$, para todo i . No caso 2.2, $p_{c'} = p_c + tC$, $p_{c'_1} = p_{c'_1} - tC_1$ e $p_{c'_2} = p_{c'_2} - tC_2$, logo também $p_{c'_1}(e) = p_{c'_2}(e)$ para toda aresta e em $\delta(X)$ e $(p_c)_i + p_{c'_i} = p_{c_i}$, para todo i . Obviamente em ambos os casos, todo ciclo de C'_1 ou de C'_2 que usa alguma aresta em $\delta(X)$ usa $\bar{e}$ e apenas mais uma aresta em $\delta(X)$. Ou seja, C'_1 e C'_2 casam em X e $\bar{e}$ é uma aresta de ligação entre C'_1 e C'_2 . Com isso, a hipótese vale para C' , C'_1 e C'_2 , como queríamos demonstrar.

Se ocorre o caso 1 então $p_{c'_1} = p_{c'_2} = 0$ e portanto $(p_c)_i = p_{c_i}$, para todo i . Ou seja, o algoritmo de fato devolve o que promete.

Em cada iteração do subalgoritmo, o número $|Sup_{G_1}(p_{c'_1})| + |Sup_{G_2}(p_{c'_2})|$ diminui de pelo menos 1. Como $|Sup_{G_1}(p_{c_1})| + |Sup_{G_2}(p_{c_2})| \leq |AG_1| + |AG_2| \leq 2m$, onde m é o número de arestas de G , temos que o número máximo de iterações do subalgoritmo é $2m$.

O caso 1 pode ser executado em tempo $O(1)$. Vamos analisar o tempo suficiente para executar uma iteração onde ocorre o caso 2. A escolha de C_1 pode ser feita em tempo $O(1)$. Já a escolha de C_2 no caso 2.2 requer uma busca na circulação C'_2 que pode ser feita em tempo $O(\|C'_2\|)$. A remoção de tC_1 de C'_1 pode ser feita em tempo $O(1)$, assim como, no caso 2.2, a remoção de tC_2 de C'_2 . O acréscimo de tC_1 no caso 2.1, ou de tC no caso 2.2, à circulação C pode ser feita em tempo $O(\|C\|)$, pois o ciclo pode já fazer parte da circulação C e portanto precisamos procurá-lo primeiro. Assim, como $\|C'_2\| \leq \|C\|$, uma iteração onde ocorre o caso 2 pode ser executada em tempo $O(\|C\|)$, ou seja, $O(p_c(AG))$, pois $\|C\| = p_c(AG)$. Assim o algoritmo tem complexidade $O(m \cdot p_c(AG))$.

3.6 O subalgoritmo 6.3

A cada iteração da segunda etapa do algoritmo de Seymour um ciclo C é escolhido e decrementado de uma função positiva q que satisfaz as condições de Seymour. O valor de decréscimo, digamos t_0 , é o maior valor de t tal que a função resultante do decréscimo, ou seja, $q - tC$, é positiva e satisfaz as condições de Seymour. Arkin e Papadimitriou mostraram [5] como calcular t_0 . Utilizaram para isso uma técnica de Megiddo [34] para o cálculo do mínimo de funções racionais. Aqui apresentamos o subalgoritmo 6.3 e o subalgoritmo 6.4 que são uma versão simplificada do algoritmo sugerido por Arkin e Papadimitriou.

Seja q_{min} o valor mínimo de q nas arestas de C , ou seja,

$$q_{min} = \min\{q(e) : e \in AC\}.$$

Para toda aresta e , seja t_e o maior valor de t tal que a função $q - tC$ é positiva e satisfaz as condições de Seymour para e . O número t_0 é justamente o número $\min\{t_e : e \in E\}$. O subalgoritmo 6.3 determina t_0 calculando este mínimo. Para isso, quando e é uma aresta tal que $t_e < q_{min}$, calcula t_e através do subalgoritmo 6.4 que será descrito na seção 3.7.

Subalgoritmo 6.3 (calcula valor máximo de decréscimo do circuito)

Recebe um grafo G , uma função q de AG em $\mathbb{Q}^+$ que satisfaz as condições de Seymour, um circuito C adequado em relação a q e devolve t_0 . Se $t_0 < q_{\min}$ então devolve também um corte X_0 e uma aresta $\bar{e}$ tal que $q - t_0 C$ é justa em X_0 , $\bar{e}$ é especial em X_0 e se $t_0 = 0$ então X_0 não é trivial.

Cada iteração do algoritmo começa com uma aresta $\bar{e}$, um conjunto E de arestas, um número t_E e um corte $X_{\bar{e}}$ tais que se $E = \emptyset$ então $t_E = q_{\min}$ senão $t_E = \min\{t_e : e \in E\}$. Além disso, se $t_E < q_{\min}$ então $t_E = t_{\bar{e}}$, $(q - t_E C)$ é justa em $X_{\bar{e}}$ e $\bar{e}$ é especial em $X_{\bar{e}}$, e se $t_E = 0$ então o corte $X_{\bar{e}}$ não é trivial. A primeira iteração começa com uma aresta qualquer no papel de $\bar{e}$, com $E = \emptyset$, $t_E = q_{\min}$ e com um corte qualquer $X_{\bar{e}}$. Cada iteração consiste em:

Caso 1: $E = AG$.

Pare devolvendo t_E e se $t_E < q_{\min}$, devolva também $X_{\bar{e}}$ e $\bar{e}$.

Caso 2: $E \neq AG$.

Seja $\bar{e}$ uma aresta em $AG - E$.

Caso 2.1: $(q - q_{\min} C)(e) \leq \min\{(q - q_{\min} C)(\delta X - e) : \delta(X) \ni e\}$.

Comece nova iteração com $\bar{e}$, $E \cup \{e\}$, t_E e $X_{\bar{e}}$ nos papéis de $\bar{e}$, E , t_E e $X_{\bar{e}}$, respectivamente.

Caso 2.2: $(q - q_{\min} C)(e) > \min\{(q - q_{\min} C)(\delta X - e) : \delta(X) \ni e\}$.

Sejam x e y as pontas de e e D uma orientação para $G - e$. Submeta $G - e$, D , $x, y, 0, q_{\min}, 2q(e), 2q(e) - 2|\{e\} \cap AC|, q, q - q_{\min} C$ ao subalgoritmo 6.4 obtendo t_e e um corte X_e tal que $(q - t_e C)$ é justa e e é especial em X_e . Se $t_e \geq t_E$ então comece nova iteração com $\bar{e}$, $E \cup \{e\}$, t_E e $X_{\bar{e}}$. Senão comece nova iteração com e , $E \cup \{e\}$, t_e e X_e .

Para mostrar que o algoritmo funciona vamos primeiramente verificar que, no início de cada iteração, se $E = \emptyset$ então $t_E = q_{\min}$ senão $t_E = \min\{t_e : e \in E\}$ e se $t_E < q_{\min}$ então $t_E = t_{\bar{e}}$, $(q - t_E C)$ é justa em $X_{\bar{e}}$ e $\bar{e}$ é especial em $X_{\bar{e}}$, e

se $t_E = 0$ então o corte $X_{\bar{e}}$ não é trivial. A hipótese vale no início da primeira iteração vacuamente. Supondo que valha no início de uma iteração onde ocorre o caso 2, vamos mostrar que a hipótese vale no início da iteração seguinte. Se ocorre o caso 2.1 então $t_e = q_{\min}$ e portanto $t_E = \min\{t_{e'} : e' \in E \cup \{e\}\}$ e a hipótese vale para $\bar{e}$, $E \cup \{e\}$, t_E e $X_{\bar{e}}$. Se ocorre o caso 2.2 então basta mostrar que se $t_e = 0$ então X_e não é trivial, pois a descrição de X_e e a escolha dos novos valores de $\bar{e}$, E , t_E e $X_{\bar{e}}$ garantem que a hipótese vale no início da próxima iteração, desde que o subalgoritmo 6.4 funcione.

Vamos mostrar então, no caso 2.2, que se X_e é trivial então $t_e > 0$. Primeiramente observe que se X_e é trivial, temos que $|\delta(X_e) \cap AC| \in \{0, 2\}$ pois C é um circuito. Assim temos duas possibilidades:

Caso 1: $e \notin AC$.

Neste caso, $(q - t_e C)(\delta X_e - e) = (q - t_e C)(e) = q(e)$. Se $v \notin VC$ então $\delta(X_e) \cap AC$, logo $(q - t_e C)(\delta X_e - e) = q(\delta X_e - e)$, ou seja, $q(e) = q(\delta X_e - e)$ e então teria ocorrido o caso 2.1. Logo, $v \in VC$. Como $e \notin AC$ e C é adequado em relação a q , temos que $q(e) < q(\delta X_e - e)$, ou seja, $(q - t_e C)(\delta X_e - e) < q(\delta X_e - e)$. Logo, $t_e > 0$ e $\delta(X_e) \cap AC \neq \emptyset$.

Caso 2: $e \in AC$.

Neste caso, $|\delta(X_e) \cap AC| = 2$. Logo $(q - t_e C)(\delta X_e - e) = q(\delta X_e - e) - t_e$. Além disso, $(q - t_e C)(\delta X_e - e) = (q - t_e C)(e) = q(e) - t_e$. Assim, temos que $q(\delta X_e - e) = q(e)$ e portanto $(q - q_{\min})(e) = q(e) - q_{\min} = q(\delta X_e - e) - q_{\min} = (q - q_{\min} C)(\delta X_e - e)$. Ou seja, $t_e = q_{\min} > 0$.

Nos dois casos concluímos que $t_e > 0$, ou seja, se $t_e = 0$ então de fato X_e não é trivial, como queríamos demonstrar.

Como $q(e) \leq q(\delta X_e - e)$, temos que $q(\delta(X_e)) \geq 2q(e)$. No caso 2.2, temos ainda que $(q - q_{\min} C)(e) > (q - q_{\min} C)(\delta X_e - e)$, ou seja, $(q - q_{\min})(\delta(X_e)) < 2q(e) - 2|\{e\} \cap AC|$. Com isso, as condições iniciais do subalgoritmo 6.4 estão satisfeitas e, desde que ele funcione, o valor devolvido é de fato t_e .

Sejam m e n o número de arestas e o número de vértices de G , respectivamente. Para analisar a complexidade do algoritmo, notemos que a cada iteração uma nova aresta é acrescentada ao conjunto E . Inicialmente E começa com $\emptyset$ e o algoritmo termina quando $E = AG$, ou seja, são no máximo m iterações. Em

cada iteração onde ocorre o caso 2, um algoritmo para encontrar fluxo máximo e corte mínimo é executado uma vez para decidir se ocorre o caso 2.1 ou o caso 2.2. O caso 2.1 pode ser executado em tempo $O(1)$, enquanto que o caso 2.2 tem a complexidade do subalgoritmo 6.4 que, como veremos mais tarde, é $O(mn^6)$. Assim o subalgoritmo 6.3 funciona e tem complexidade $O(m^2n^6)$.

3.7 O subalgoritmo 6.4

O subalgoritmo 6.4 é uma variação do algoritmo de Ford e Fulkerson [12] (com refinamento de Edmonds e Karp [9]) para o cálculo do fluxo máximo e corte mínimo entre dois vértices.

Seja G um grafo, D uma orientação e x e y dois vértices. Dizemos que uma função f de AG em $\mathbf{Q}$ é um *fluxo de x a y* se $f(\delta^-(v)) = f(\delta^+(v))$ para todo vértice v de $VG - \{x, y\}$.

Seja c uma função de AG em $\mathbf{Q}^+$. Dizemos que um fluxo f de x a y é *c -viável* se $|f| \leq c$.

Seja f um fluxo c -viável de x a y . Dado um caminho P em G , chamamos de *função folga relativa a f e c* a função $\Delta_{c,f}$ de AP em $\mathbf{Q}^+$ definida por

$$\Delta_{c,f}(e) = \begin{cases} c(e) - f(e) & \text{se } e \in \vec{AP}, \\ c(e) + f(e) & \text{se } e \in \overleftarrow{AP}. \end{cases}$$

Observe que $\Delta_{c,f}$ é uma função positiva pois f é c -viável. Uma aresta e em AP está *saturada em relação a f e c* se $\Delta_{c,f}(e) = 0$. Dizemos que P é *aumentador em relação a f e c* se não existem arestas saturadas em AP .

Seja t um racional positivo e P um caminho de x a y aumentador em relação a f e c . O *fluxo de x a y obtido de f pela aumento de t em P* é o fluxo f' que coincide com f exceto nas arestas de P , onde vale $f(e) + t$ se $e \in \vec{AP}$ e $f(e) - t$ se $e \in \overleftarrow{AP}$. Se $t = \min\{\Delta_{c,f}(e) : e \in AP\}$, dizemos que f' foi obtido de f por uma *aumentação máxima em P* . Observe que nesse caso f' é ainda um fluxo c -viável e que uma aresta em AP ficou saturada em relação a f' e c .

A *intensidade de um fluxo f de x a y* , denotada por $\text{int}(f)$, é o número $f(\delta^+(x)) - f(\delta^-(x))$.

Um fluxo f é *máximo em relação a c* se f é c -viável e não existe outro fluxo c -viável de x a y com intensidade maior que f .

Ford e Fulkerson mostraram, através de seu algoritmo, que f é um fluxo máximo em relação a c se e somente se não existe caminho de x a y aumentador em relação a f e c . Como consequência disso, demonstraram que a intensidade de um fluxo máximo em relação a c é o número $\min\{c(\delta(X)) : x \in X, y \notin X\}$, que denotamos por $\text{cap}_{\min}(c)$.

O subalgoritmo 6.4 é uma generalização do algoritmo de Ford e Fulkerson que trabalha em um intervalo. Vamos então generalizar algumas das definições acima para um intervalo.

Para números racionais L e R , denotamos por $[L, R]$ o conjunto de todos os números racionais entre L e R , inclusive L e R . Seja g uma função de $AG \times [L, R]$ em $\mathbf{Q}$. Dado um racional t em $[L, R]$, denotamos por g_t a função $g(\cdot, t)$, ou seja, a função de AG em $\mathbf{Q}$ definida por $g_t(e) = g(e, t)$ para toda aresta e .

Dizemos que f é um $[L, R]$ -fluxo de x a y se f_t é um fluxo de x a y para todo t em $[L, R]$.

Seja c uma função de $AG \times [L, R]$ em $\mathbf{Q}^+$. Dizemos que um $[L, R]$ -fluxo f de x a y é c -viável se f_t é c_t -viável para todo t em $[L, R]$.

Seja f um $[L, R]$ -fluxo c -viável de x a y . Dado um caminho P em G , chamamos de *função folga relativa a f e c* a função Δ de $AP \times [L, R]$ em $\mathbf{Q}^+$ tal que $\Delta_t = \Delta_{c_t, f_t}$ para todo t em $[L, R]$. Observe que $\Delta \geq 0$ pois f é c -viável. Dada uma aresta e em AP , dizemos que e está $[L, R]$ -saturada em relação a f e c se $\Delta_t(e) = 0$ para todo t em $[L, R]$. Dizemos que P é $[L, R]$ -aumentador em relação a f e c se não existem arestas saturadas em AP .

Seja d uma função de $[L, R]$ em $\mathbf{Q}^+$ e P um caminho de x a y $[L, R]$ -aumentador em relação a f e c . O *fluxo obtido de f pela aumento de d em P* é o $[L, R]$ -fluxo f' de x a y em $[L, R]$ que coincide com f exceto nas arestas de P , onde vale $f(e, t) + d_t$ para todo t em $[L, R]$ se $e \in \vec{AP}$ e $f(e, t) - d_t$ para todo t em $[L, R]$ se $e \in \overleftarrow{AP}$. Seja $\overline{\Delta}(t) = \min\{\Delta_t(e) : e \in AP\}$ para todo t em $[L, R]$. Se $d = \overline{\Delta}$ então dizemos que f' foi obtido de f por uma $[L, R]$ -aumentação máxima em P . Observe que nesse caso f' é ainda um $[L, R]$ -fluxo c -viável e que uma aresta em AP ficou saturada em relação a f' e c .

Seja h uma função de $[L, R]$ em $\mathbf{Q}$. Para cada t em $[L, R]$, denotamos por h_t o número $h(t)$ para todo t em $[L, R]$.

Na verdade, o subalgoritmo 6.4 trabalha apenas com funções lineares. Tais funções serão representadas por pares de números. Assim, dado um par de

números h_L e h_R , a função h de $[L, R]$ em $\mathbb{Q}^+$ é definida por

$$h_t = \frac{R-t}{R-L}h_L + \frac{t-L}{R-L}h_R,$$

para todo t em $[L, R]$.

Uma função g de $AG \times [L, R]$ em $\mathbb{Q}$ é *linear* se a função $g(e, \cdot)$ de $[L, R]$ em $\mathbb{Q}^+$ é linear para toda aresta e . Observe que se g é linear então

$$g_t = \frac{R-t}{R-L}g_L + \frac{t-L}{R-L}g_R,$$

para todo t em $[L, R]$.

O objetivo do subalgoritmo 6.4 é, dadas funções lineares c de $AG \times [L, R]$ em $\mathbb{Q}^+$ e a de $[L, R]$ em $\mathbb{Q}^+$ tais que $\text{cap}_{\min}(c_L) \geq a_L$ e $\text{cap}_{\min}(c_R) < a_R$, determinar o maior valor em $[L, R]$, digamos t , tal que $\text{cap}_{\min}(c_t) = a_t$. O algoritmo, como já dissemos, trabalha apenas com funções lineares. Esta restrição traz alguns problemas pois a função $\overline{\Delta}$ que determina a $[L, R]$ -aumentação máxima nem sempre é linear. Ela é na verdade linear por trechos. O que o algoritmo faz então é tomar um intervalo contido em $[L, R]$, digamos $[L', R']$, com as mesmas características que $[L, R]$, onde $\overline{\Delta}$ é linear, para desse modo prosseguir trabalhando apenas com funções lineares.

Subalgoritmo 6.4 *Recebe um grafo G , uma orientação D para G , vértices x e y de G , racionais L_0 e R_0 , racionais a_{L_0} e a_{R_0} e funções c_{L_0} e c_{R_0} de AG em $\mathbb{Q}^+$ tais que $\text{cap}_{\min}(c_{L_0}) \geq a_{L_0}$ e $\text{cap}_{\min}(c_{R_0}) < a_{R_0}$. Devolve o maior valor em $[L_0, R_0]$, digamos t , tal que $\text{cap}_{\min}(c_t) \geq a_t$ e um corte X_0 tal que $x \in X_0$, $y \notin X_0$ e $c_t(\delta(X_0)) = \text{cap}_{\min}(c_t)$.*

Cada iteração do algoritmo começa com dois racionais L e R tais que $L_0 \leq L < R \leq R_0$, $\text{cap}_{\min}(c_L) \geq a_L$ e $\text{cap}_{\min}(c_R) < a_R$, e um $[L, R]$ -fluxo linear c -viável f de x a y . A primeira iteração começa com L_0, R_0 e o $[L_0, R_0]$ -fluxo nulo nos papéis de L, R e f , respectivamente. Cada iteração consiste em:

Seja X_0 o conjunto $\{v \in VG : \text{existe caminho de } x \text{ a } v \text{ } [L, R]\text{-aumentador em relação a } f \text{ e } c\}$.

Caso 1: $y \notin X_0$.

Calcule t_0 em $[L, R]$ tal que $c_{t_0}(\delta(X_0)) = a_{t_0}$. Pare devolvendo t_0 e X_0 .

Caso 2: $y \in X_0$.

Seja P um caminho de x a y $[L, R]$ -aumentador em relação a f e c de comprimento mínimo. Comece um novo processo iterativo com um conjunto T de pares de arestas de AP e racionais L' e R' tais que, para todo par de arestas (e, e') fora de T , $\Delta(e, \cdot) \leq \Delta(e', \cdot)$ ou $\Delta(e, \cdot) \geq \Delta(e', \cdot)$, $L \leq L' < R' \leq R$, $\text{cap}_{\min}(c_{L'}) \geq a_{L'}$ e $\text{cap}_{\min}(c_{R'}) < a_{R'}$. A primeira iteração deste processo novo começa com $AP \times AP$, L e R . Cada iteração consiste em:

Caso 2.1: $T = \emptyset$.

Seja f' o $[L', R']$ -fluxo obtido de f pela $[L', R']$ -aumentação máxima em P . Pare as iterações do deste processo iterativo e comece nova iteração do algoritmo com L', R' e f' nos papéis de L, R e f .

Caso 2.2: $T \neq \emptyset$.

Sejam (e, e') um par de arestas em T . Se $\Delta(e, \cdot) \leq \Delta(e', \cdot)$ ou $\Delta(e, \cdot) \geq \Delta(e', \cdot)$ então comece nova iteração deste processo iterativo com $T - \{(e, e')\}$, L' e R' nos papéis de T , L' e R' . Senão seja $\bar{t}$ em $[L', R']$ tal que $\Delta_{\bar{t}}(e) = \Delta_{\bar{t}}(e')$. Determine o valor de $\text{cap}_{\min}(c_{\bar{t}})$ através de um algoritmo para calcular fluxo máximo e corte mínimo. Se $\text{cap}_{\min}(c_{\bar{t}}) \geq a_{\bar{t}}$, comece nova iteração deste processo iterativo com $T - \{(e, e')\}$, $\bar{t}$ e R' nos papéis de T , L' e R' . Se $\text{cap}_{\min}(c_{\bar{t}}) < a_{\bar{t}}$, comece nova iteração deste processo iterativo com $T - \{(e, e')\}$, L' e $\bar{t}$ nos papéis de T , L' e R' .

Para mostrar que o algoritmo funciona, vamos mostrar que no início de cada iteração L, R e f são tais que $L < R$, $\text{cap}_{\min}(c_L) \geq a_L$ e $\text{cap}_{\min}(c_R) < a_R$, e f é um $[L, R]$ -fluxo linear c -viável de x a y . No início da primeira iteração, claramente a hipótese vale. Supondo que o processo iterativo do caso 2 funcione, temos que $L \leq L' < R' \leq R$, ou seja, $L_0 \leq L' < R' \leq R_0$. Além disso, $\text{cap}_{\min}(c_{L'}) \geq a_{L'}$ e $\text{cap}_{\min}(c_{R'}) < a_{R'}$. É preciso garantir também que f' é um fluxo linear c -viável de x a y . Certamente f' é um fluxo c -viável pois foi obtido de f pela $[L', R']$ -aumentação máxima em P . Para mostrar que f' é linear, basta mostrar

que no caso 2.1 $\bar{\Delta}$ é linear em $[L', R']$. Como, para todo par de arestas (e, e') , $\Delta(e, \cdot) \leq \Delta(e', \cdot)$ ou $\Delta(e, \cdot) \geq \Delta(e', \cdot)$, existe uma aresta $\bar{e}$ tal que $\bar{\Delta}_{L'} = \Delta_{L'}(\bar{e})$ e $\bar{\Delta}_{R'} = \Delta_{R'}(\bar{e})$. Observe que, pela definição de $\bar{\Delta}$ e por que $\Delta(e, \cdot)$ é linear para toda aresta e , temos que $\bar{\Delta} = \Delta(\bar{e}, \cdot)$ no intervalo $[L', R']$ e portanto $\bar{\Delta}$ é linear em $[L', R']$, como queríamos demonstrar.

Vamos mostrar agora que, no início de cada iteração do processo iterativo do caso 2, T é um conjunto de pares de arestas de AP e L' e R' são racionais tais que, para todo par de arestas (e, e') fora de T , $\Delta(e, \cdot) \leq \Delta(e', \cdot)$ ou $\Delta(e, \cdot) \geq \Delta(e', \cdot)$, $L \leq L' < R' \leq R$, $\text{cap}_{\min}(c_{L'}) \geq a_{L'}$ e $\text{cap}_{\min}(c_{R'}) < a_{R'}$. Claramente a hipótese vale no início da primeira iteração do processo. Supondo que a hipótese valha no início de uma iteração em que ocorre o caso 2.2, se $\Delta(e, \cdot) \leq \Delta(e', \cdot)$ ou $\Delta(e, \cdot) \geq \Delta(e', \cdot)$, claramente a hipótese vale para $T - \{(e, e')\}$, L' e R' . Se $\Delta(e, \cdot) \not\leq \Delta(e', \cdot)$ e $\Delta(e, \cdot) \not\geq \Delta(e', \cdot)$ então temos que $(\Delta(e, L') < \Delta(e', L') \text{ e } \Delta(e, R') > \Delta(e', R'))$ ou $(\Delta(e, L') > \Delta(e', L') \text{ e } \Delta(e, R') < \Delta(e', R'))$. Como $\Delta(e, \cdot)$ e $\Delta(e', \cdot)$ são funções lineares, realmente existe $\bar{t}$ em $[L', R']$ tal que $\Delta_{\bar{t}}(e) = \Delta_{\bar{t}}(e')$. Certamente $L' < \bar{t} < R'$ e portanto a escolha de L' e $\bar{t}$ ou $\bar{t}$ e R' , no caso 2.2, garantem que a hipótese vale no início de toda iteração.

Em cada iteração do processo iterativo do caso 2, o conjunto T diminui de um elemento, logo o número de iterações não passa de n^2 , onde n é o número de vértices do grafo. Como $\Delta(e, \cdot)$ e $\Delta(e', \cdot)$ são funções lineares, podemos determinar se $\Delta(e, \cdot) \leq \Delta(e', \cdot)$ ou $\Delta(e, \cdot) \geq \Delta(e', \cdot)$ ou determinar o valor de $\bar{t}$ em tempo $O(1)$. Utilizando um algoritmo para calcular fluxo máximo e corte mínimo com complexidade $O(n^3)$, uma execução do processo iterativo do caso 2 pode ser feita em tempo $O(n^5)$.

A função h de $[L_0, R_0]$ em $\mathbf{Q}^+$, definida por $h(t) = \text{cap}_{\min}(c_t)$ para todo t em $[L_0, R_0]$, é concava e a_t é linear, por isso, no caso 1 existe apenas um valor t_0 tal que $h(t_0) = a_{t_0}$ e $h(t) < a_t$ para todo t em $[t_0, R_0]$. Ou seja, t_0 é o maior valor em $[L_0, R_0]$ tal que $\text{cap}_{\min}(c_{t_0}) \geq a_{t_0}$.

O número de iterações do algoritmo é $O(mn)$, onde m é o número de arestas de G , assim como o refinamento do algoritmo de Ford e Fulkerson proposto por Edmonds e Karp. Abaixo apresentamos uma demonstração semelhante a demonstração de Lawler [29].

Vamos mostrar primeiramente que o comprimento do caminho aumentador escolhido no caso 2 nunca diminui. Para isso, sejam $\sigma_y^{(k)}$ o comprimento do menor

caminho aumentador em (L, R) de x a v depois de k aumentações máximas, e $\tau_v^{(k)}$ o comprimento do menor caminho aumentador em (L, R) de v a y depois de k aumentações máximas. Vamos mostrar primeiramente que $\sigma_v^{(k+1)} \geq \sigma_v^{(k)}$.

Supondo por absurdo que existam v e k tais que $\sigma_v^{(k+1)} < \sigma_v^{(k)}$ então seja v tal que $\sigma_v^{(k+1)} = \min\{\sigma_u^{(k+1)} : u \in VG \text{ e } \sigma_u^{(k+1)} < \sigma_u^{(k)}\}$. Seja P o menor caminho aumentador em (L, R) de x a v depois da $(k+1)$ -ésima aumentação, e e a última aresta de P , com pontas u, v . Temos que $\Delta_t(e) > 0$, portanto $\sigma_v^{(k+1)} = \sigma_u^{(k+1)} + 1$. A aresta e estava saturada de u a v depois da k -ésima aumentação, do contrário $\sigma_v^{(k)} \leq \sigma_u^{(k)} + 1 \leq \sigma_u^{(k+1)} + 1 = \sigma_v^{(k+1)}$. Mas se e estava saturada depois da k -ésima aumentação e $\Delta_t(e) > 0$ depois da $(k+1)$ -ésima aumentação então é porque e foi utilizada de v para u no caminho aumentador usado na $(k+1)$ -ésima aumentação. Como tal caminho era mínimo, $\sigma_u^{(k)} = \sigma_v^{(k)} + 1$ e portanto $\sigma_v^{(k)} = \sigma_u^{(k)} - 1 \leq \sigma_u^{(k+1)} - 1 \leq \sigma_v^{(k+1)} - 2$, contrariando a hipótese de que $\sigma_v^{(k)} > \sigma_v^{(k+1)}$. De forma análoga, podemos mostrar que $\tau_v^{(k+1)} \geq \tau_v^{(k)}$.

A cada aumentação máxima pelo menos uma aresta do caminho é saturada. Seja e , com pontas u, v , uma aresta que é saturada na $(k+1)$ -ésima aumentação máxima. O comprimento do caminho aumentador utilizado na $(k+1)$ -ésima aumentação máxima é $\sigma_u^{(k)} + \tau_u^{(k)} = \sigma_v^{(k)} + \tau_v^{(k)}$. Na próxima iteração em que e aparece em um caminho aumentador mínimo, digamos na $(l+1)$ -ésima aumentação, ela será utilizada no outro sentido (de v para u). Se e está orientada em G de u para v então $\sigma_v^{(k)} = \sigma_u^{(k)} + 1$ e $\sigma_u^{(l)} = \sigma_v^{(l)} + 1$.

Pelo que foi visto anteriormente e como $l > k$, segue que $\sigma_v^{(l)} \geq \sigma_v^{(k)}$ e $\tau_u^{(l)} \geq \tau_u^{(k)}$, assim $\sigma_u^{(l)} + \tau_u^{(l)} \geq \sigma_u^{(k)} + \tau_u^{(k)} + 2$. Ou seja, o comprimento do (próximo) caminho aumentador mínimo que utiliza e aumenta de pelo menos 2.

O comprimento máximo de um caminho aumentador é $n-1$. Assim nenhuma aresta pode ser saturada mais do que $n/2$ vezes. Como toda aumentação máxima satura pelo menos uma aresta, o número máximo de aumentações, ou seja, de iterações do subalgoritmo, não passa de $mn/2$. Com isso o algoritmo funciona e tem complexidade $O(mn^6)$.

3.8 O problema P2

O algoritmo de Seymour não resolve o problema P2. Dados limites inteiros ℓ, u tais que $\ell \leq u$ e ℓ, u satisfazem as condições de Seymour, a primeira etapa do

algoritmo transforma ℓ, u em uma função q que satisfaz as condições de Seymour. A segunda etapa, a partir de q , obtém uma circulação cuja função peso é q , resolvendo portanto o problema P2r. A circulação obtida pode não ser inteira. Infelizmente, pode nem existir uma circulação inteira cuja função peso é q .

Exemplo 3.3 Um grafo G e uma função inteira q que satisfaz as condições de Seymour indicada nas arestas de G . Não existe circulação inteira cuja função peso é q , como mostramos abaixo.

Para toda circulação inteira C , temos que $|p_c(\delta(X))|$ é par para todo corte X , pois todo ciclo de C usa um número par de arestas em $\delta(X)$ e tem multiplicidade inteira. No entanto, existe um corte X em G tal que $|q(\delta(X))|$ é ímpar.

Poderíamos então conjecturar que, dada uma função q satisfazendo as condições de Seymour, existe uma circulação inteira cuja função peso é q se $|q(\delta(X))|$ é par para todo corte X em G . No entanto, o exemplo 3.4 é um contra-exemplo para tal conjectura.

Seymour demonstrou [43] que se G é um grafo planar, q satisfaz as condições de Seymour e $|q(\delta(X))|$ é par para todo corte X então existe uma circulação inteira cuja função peso é q .

Além disso ele propôs a seguinte conjectura para o caso geral:

Conjectura 3.1 Para todo grafo G e toda função q tal que $q(e)$ é par para toda aresta e e q satisfaz as condições de Seymour, existe uma circulação inteira cuja função peso é q .

A validade desta conjectura implicaria que, dados limites inteiros ℓ, u tais que $\ell \leq u$ e ℓ, u satisfazem as condições de Seymour, existe uma circulação C tal que p_c é ℓ, u -viável e, para todo ciclo C em $\mathcal{C}$, $m_c(C) \in \mathbf{Z}^+/2$, onde $\mathbf{Z}^+/2 = \{t \in \mathbf{Q}^+ : 2t \in \mathbf{Z}^+\}$, como vamos mostrar a seguir. Pelo algoritmo 5, existe uma função inteira q tal que $\ell \leq q \leq u$ e q satisfaz as condições de Seymour. A

função q' de AG em $\mathbb{Q}^+$ definida por $q'(e) = 2q(e)$ para toda aresta e satisfaz as condições de Seymour. Além disso, $q'(e)$ é par para toda aresta e . A validade da conjectura 3.1 implicaria que existe uma circulação inteira cuja função peso é q' . A circulação C tal que $C = C'$ e $m_c(C) = m_{c'}(C)/2$ para todo C em $\mathcal{C}$ é uma circulação C cuja função peso é q e tal que $m_c(C) \in \mathbb{Z}^+/2$ para todo C em $\mathcal{C}$.

Exemplo 3.4 O grafo de Petersen e uma função inteira q que satisfaz as condições de Seymour indicada nas suas arestas. Nas arestas sem rótulo a função vale 1. Não existe circulação inteira cuja função peso é q , como mostramos abaixo.

Suponha por absurdo que exista uma circulação inteira cuja função peso é q . Como o grafo de Petersen é cúbico e existe uma aresta de peso 2 e duas de peso 1 em cada vértice, temos que todo ciclo de C utiliza alternadamente arestas de peso 1 e 2, ou seja, todo ciclo de C utiliza o mesmo número de arestas de peso 1 e 2. Como existe um corte X tal que $\delta(X)$ é justamente o conjunto das arestas de peso 2, temos que todo ciclo de C utiliza um número par de arestas de peso 2, ou seja, o comprimento de todo ciclo de C é um múltiplo de 4. Como o comprimento de um ciclo no grafo de Petersen é pelo menos 5, temos que o comprimento de todos os ciclos de C é 8 (não pode ser 12 pois existem apenas 5 arestas de peso 2). Como todo ciclo utiliza arestas de peso 1, a multiplicidade de todos os ciclos de C é 1. Assim a soma do comprimento de todos os ciclos de C é $q(AG)$, ou seja, 20. Mas isso é uma contradição pois tal soma deveria ser um múltiplo de 8.

Capítulo 4

Os algoritmos de Jaeger e alguns casos particulares de $P2^+$ e $P3$

Neste capítulo, examinaremos as conseqüências dos algoritmos de Jaeger sobre os casos particulares dos problemas $P2^+$ e $P3$ em que os limites inferiores valem 1 e os limites superiores valem k , onde k é um número natural. Diremos também alguma coisa sobre o caso particular do problemas $P2^+$ em que os limites inferiores e superiores valem k . Claramente todas as conseqüências sobre os casos particulares do problema $P2^+$, valem também para os casos particulares correspondentes do problema $P2$. A literatura registra várias conjecturas sobre estes casos particulares; todas continuam abertas.

4.1 Os algoritmos de Jaeger

Vamos descrever dois algoritmos de Jaeger: o primeiro constrói uma circulação inteira $\mathcal{C}$ tal que p_c é 1,2-viável em grafos 4-aresta-conexos; o segundo constrói uma circulação inteira $\mathcal{C}$ tal que p_c é 1,3-viável em grafos 3-aresta-conexos.

Os dois algoritmos de Jaeger baseiam-se fortemente no seguinte teorema de Nash-Williams [38]:

Teorema (Nash-Williams): *Para todo grafo $2k$ -aresta-conexo, existem k árvores geradoras duas a duas disjuntas nas arestas.*

Dado um grafo G e um conjunto E de arestas de G , dizemos que uma função q de AG em $\mathbf{Q}^+$ cobre E se $q(e) \neq 0$ para toda aresta e em E . Se $E = AG$, dizemos que q cobre G . Se $E = \{e\}$, dizemos que q cobre e .

Dado um grafo G e um subgrafo gerador conexo S de G , uma *circulação fundamental de S* é uma circulação inteira $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 0,1-viável e cobre $AG - AS$.

Algoritmo 7 (Jaeger, para grafos 4-aresta-conexos) *Recebe um grafo 4-aresta-conexo G e devolve uma circulação inteira $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 1,2-viável e uma 2-coloração para $\mathcal{C}$.*

Sejam T_1 e T_2 duas árvores geradoras de G disjuntas nas arestas. Para todo i em $\{1, 2\}$, submeta G e T_i ao algoritmo 9 que devolve uma circulação fundamental $\mathcal{C}_i$ de T_i . Seja $\mathcal{C}$ a circulação $\mathcal{C}_1 + \mathcal{C}_2$. Devolva as circulações $\mathcal{C}$, $\mathcal{C}_1$ e $\mathcal{C}_2$.

Existem duas árvores geradoras disjuntas pelo teorema de Nash-Williams, pois G é 4-aresta-conexo. Como poderemos conferir mais tarde, para i em $\{1, 2\}$, G e T_i satisfazem as condições iniciais do algoritmo 9. Assim, desde que ele funcione, $\mathcal{C}_i$ é uma circulação fundamental de T_i , ou seja, é uma circulação inteira tal que $p_{\mathcal{C}_i}$ é 0,1-viável e cobre as arestas de $AG - T_i$. Claramente $\mathcal{C}$ é uma circulação inteira e $\mathcal{C}_1$ e $\mathcal{C}_2$ é uma 2-coloração para $\mathcal{C}$. Como $AT_1 \cap AT_2 = \emptyset$, temos que $\mathcal{C}$ cobre G . Além disso, como $p_{\mathcal{C}_i}$ é 0,1-viável para todo i em $\{1, 2\}$, temos que $p_{\mathcal{C}} \leq 2$. Ou seja, de fato a circulação $\mathcal{C}$ é tal que $p_{\mathcal{C}}$ é 1,2-viável.

A determinação de duas árvores geradoras disjuntas nas arestas pode ser feita em tempo $O(m^2)$ [42], onde m é o número de arestas de G . Mostraremos mais tarde que o algoritmo 9 tem complexidade $O(m^2)$. Como $\mathcal{C}_1$ e $\mathcal{C}_2$ são circulações inteiras tais que $p_{\mathcal{C}_i}$ é 0,1-viável para todo i em $\{1, 2\}$, a construção de $\mathcal{C}$ a partir de $\mathcal{C}_1$ e $\mathcal{C}_2$ pode ser feita em tempo $O(m^2)$. Concluimos então que o algoritmo tem complexidade $O(m^2)$.

Agora vamos descrever o outro algoritmo de Jaeger.

Algoritmo 8 (Jaeger, para grafos 3-aresta-conexos) *Recebe um grafo 3-aresta-conexo G e devolve uma circulação inteira $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 1,3-viável e uma 3-coloração para $\mathcal{C}$.*

Seja G' obtido de G pela duplicação de todas as arestas. Sejam T'_1, T'_2 e T'_3 árvores geradoras em G' duas a duas disjuntas nas arestas e T_1, T_2 e T_3 as árvores correspondentes em G . Para todo i em

$\{1, 2, 3\}$, submeta G e T_i ao algoritmo 9 que devolve uma circulação fundamental $\mathcal{C}_i$ de T_i . Seja $\mathcal{C}$ a circulação $\mathcal{C}_1 + \mathcal{C}_2 + \mathcal{C}_3$. Devolva $\mathcal{C}$, $\mathcal{C}_1, \mathcal{C}_2$ e $\mathcal{C}_3$.

Novamente, existem três árvores geradoras duas a duas disjuntas nas arestas pelo teorema de Nash-Williams [38], pois, como G é 3-aresta-conexo, G' é 6-aresta-conexo. Como poderemos conferir mais tarde, para i em $\{1, 2, 3\}$, G e T_i satisfazem as condições iniciais do algoritmo 9. Assim, desde que ele funcione, $\mathcal{C}_i$ é uma circulação fundamental de T_i , ou seja, uma circulação inteira tal que $p_{\mathcal{C}_i}$ é 0,1-viável e cobre as arestas de $AG - T_i$. Claramente $\mathcal{C}$ é uma circulação inteira e $\mathcal{C}_1, \mathcal{C}_2$ e $\mathcal{C}_3$ é uma 3-coloração para $\mathcal{C}$. Como $AT'_1 \cap AT'_2 \cap AT'_3 = \emptyset$, segue que uma aresta aparece no máximo em duas das árvores T_1, T_2 e T_3 e portanto é coberta por uma das circulações $\mathcal{C}_1, \mathcal{C}_2$ e $\mathcal{C}_3$. Ou seja, $\mathcal{C}$ cobre G . Além disso, como $p_{\mathcal{C}_i}$ é 0,1-viável para todo i em $\{1, 2, 3\}$, temos que $p_{\mathcal{C}} \leq 3$. Logo $p_{\mathcal{C}}$ é 1,3-viável.

A análise da complexidade é praticamente idêntica a do primeiro algoritmo de Jaeger. A determinação das três árvores geradoras duas a duas disjuntas nas arestas pode ser feita em tempo $O(m^2)$ [42], onde m é o número de arestas do grafo G . A construção de T_1, T_2 e T_3 a partir de T'_1, T'_2 e T'_3 pode ser feita em tempo $O(n)$, onde n é o número de vértices do grafo G . O algoritmo 9 tem complexidade $O(m^2)$. Como $\mathcal{C}_1, \mathcal{C}_2$ e $\mathcal{C}_3$ são circulações inteiras tais que $p_{\mathcal{C}_i}$ é 0,1-viável para todo i em $\{1, 2, 3\}$, a construção de $\mathcal{C}$ a partir de $\mathcal{C}_1, \mathcal{C}_2$ e $\mathcal{C}_3$ pode ser feita em tempo $O(m^2)$. Concluimos então que o algoritmo tem complexidade $O(m^2)$.

4.2 Construção de circulações fundamentais

Vamos descrever um algoritmo que, dado um grafo G e um subgrafo gerador conexo S de G , encontra uma circulação fundamental de S .

Dada uma circulação $\mathcal{C}$, denotaremos por AC o conjunto $\bigcup \{AC : C \in \mathcal{C}\}$.

Dado um grafo G e um conjunto E de arestas de G , note que existe uma circulação inteira $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 0,1-viável e $AC = E$ se e somente se $|E \cap \delta(v)|$ é par para todo vértice v .

Dados conjuntos E_1 e E_2 , denotamos por $E_1 \Delta E_2$ o conjunto $(E_1 - E_2) \cup (E_2 - E_1)$. Dadas circulações inteiras C_1 e C_2 tais que p_{c_i} é 0,1-viável para todo i em $\{1, 2\}$, denotamos por $C_1 \Delta C_2$ a circulação inteira C tal que p_c é 0,1-viável e $AC = AC_1 \Delta AC_2$. Tal circulação existe pois, como $|AC_1 \cap \delta(v)|$ e $|AC_2 \cap \delta(v)|$ são números pares, $|(AC_1 \Delta AC_2) \cap \delta(v)|$ é par para todo vértice v .

Algoritmo 9 (constrói uma circulação fundamental) *Recebe um grafo G , um subgrafo gerador conexo S de G e devolve uma circulação fundamental de S .*

O algoritmo é iterativo. Cada iteração começa com uma circulação inteira C tal que p_c é 0,1-viável. A primeira iteração começa com a circulação vazia. Cada iteração consiste em:

Caso 1: p_c cobre $AG - AS$.

Pare e devolva C .

Caso 2: p_c não cobre $AG - AS$.

Seja $\bar{e}$ uma aresta que não é coberta por p_c . Sejam x e y as pontas de $\bar{e}$, P um caminho de x a y em S e C o ciclo $P \langle y, \bar{e}, x \rangle$.

Seja C' a circulação $C \Delta \{C\}$. Comece nova iteração com C' no papel de C .

No caso 2, existe um caminho de x a y pois S é gerador e conexo. É fácil ver que no início de cada iteração C é uma circulação inteira tal que p_c é 0,1-viável.

No caso 1, obviamente C é uma circulação fundamental de S , ou seja, o algoritmo devolve o que promete.

O conjunto $AC \cap (AG - AS)$ aumenta de pelo menos um elemento a cada iteração. Para verificar isso, vamos mostrar que, no caso 2, $|AC' \cap (AG - AS)| > |AC \cap (AG - AS)|$. Como $AC \subseteq AS \cup \{e\}$ e $C' = C \Delta \{C\}$, temos que $AC \cap (AG - AS) \subseteq AC' \cap (AG - AS)$. Além disso, temos que $\bar{e} \in AC'$, $\bar{e} \notin AC$ e $\bar{e} \notin AS$, logo $|AC' \cap (AG - AS)| \geq |(AC \cap (AG - AS)) \cup \{\bar{e}\}| = |AC \cap (AG - AS)| + 1$, como queríamos demonstrar. Portanto o número máximo de iterações é $O(m)$, onde m é o número de arestas de G . A determinação do ciclo C e o cálculo de $C \Delta \{C\}$ podem ser feitas em tempo $O(m)$. Logo, o algoritmo tem complexidade $O(m^2)$, como queríamos demonstrar.

4.3 Os problemas $P2$ e $P2^+$ com limites 1 e k

Nesta seção vamos nos concentrar nos casos particulares dos problemas 2 e 2^+ em que o limite inferior vale 1 e o limite superior vale k para toda aresta. Mostraremos as implicações dos algoritmos de Jaeger para estes casos [28, 18].

Nestes casos particulares, como o limite inferior de toda aresta é 1, uma condição necessária para que os problemas tenham solução é a ausência de 1-cortes no grafo, pois não existe ciclo que use a única aresta de um 1-corte. O mesmo vale para os demais casos particulares dos problemas 2^+ e 3 que serão analisados neste capítulo.

O fato abaixo mostra uma condição necessária e suficiente para que o problema $P2(G,1,1)$ tenha solução:

Fato 4.1 *Para todo grafo G , as seguintes afirmações são equivalentes:*

1. *Existe uma circulação inteira $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 1,1-viável.*
2. *G é euleriano.*

Condições necessárias e suficientes para que $P2(G,1,2)$ tenha solução não foram determinadas ainda. O fato abaixo mostra uma condição suficiente para que $P2^+(G,1,2,2)$ tenha solução. Claramente a mesma condição é suficiente também para que $P2(G,1,2)$ tenha solução.

Fato 4.2 *Para todo grafo sem 1-cortes e sem 3-cortes, existe uma circulação inteira $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 1,2-viável e uma 2-coloração para $\mathcal{C}$.*

Se G não é conexo então podemos considerar cada uma de suas componentes separadamente, ou seja, podemos considerar que G é conexo. Se G tem um 2-corte X , sejam f e f' as arestas em $\delta(X)$. Qualquer circulação $\mathcal{C}$ no grafo $G \setminus f$ corresponde a uma circulação $\mathcal{C}'$ em G tal que $p_{\mathcal{C}'}(e) = p_{\mathcal{C}}(e)$ para toda aresta e em $AG - \{f\}$ e $p_{\mathcal{C}'}(f) = p_{\mathcal{C}}(f')$. Com isso, podemos considerar que G não tem 2-cortes e, para provar a validade deste fato, podemos considerar apenas grafos 3-aresta-conexos. Dado um grafo G e uma aresta f , os cortes do grafo $G \setminus f$ são cortes de G . Logo, se G não tem 3-cortes então $G \setminus f$ também não tem 3-cortes. Ou seja, podemos considerar apenas grafos 4-aresta-conexos, concluindo a validade do fato 4.2 a partir do algoritmo 7.

As conjecturas abaixo [43, 27] sugerem que a ausência de 1-cortes em G é condição necessária e suficiente para que $P2(G,1,2)$ e $P2^+(G,1,2,4)$ tenham solução:

Conjectura 4.1 (circulação 1,2-viável) *Para todo grafo sem 1-cortes, existe uma circulação inteira $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 1,2-viável.*

Conjectura 4.2 (circulação 1,2-viável com 4-coloração) *Para todo grafo sem 1-cortes, existe uma circulação inteira $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 1,2-viável e uma 4-coloração para $\mathcal{C}$.*

O fato abaixo mostra que a conjectura 4.1 vale com 3 no lugar de 2:

Fato 4.3 *Para todo grafo sem 1-cortes, existe uma circulação inteira $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 1,3-viável e uma 3-coloração para $\mathcal{C}$.*

Como no fato 4.2, podemos considerar apenas grafos 3-aresta-conexos e o algoritmo 8 completa a demonstração deste fato.

4.4 Os problemas $P2$ e $P2^+$ com limites k e k

Nesta seção vamos analisar os casos particulares dos problema $P2$ e $P2^+$ em que os limites inferiores e superiores valem k em todas as arestas. Enunciaremos a famosa conjectura da cobertura dupla por ciclos e mostraremos as implicações dos resultados de Jaeger para estes casos [28, 18].

O fato abaixo mostra a equivalência entre $P2^+(G,1,2,k)$ e $P2^+(G,2,2,k+1)$.

Fato 4.4 *Para todo grafo, existe uma circulação inteira $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 1,2-viável e uma k -coloração para $\mathcal{C}$ se e somente se existe uma circulação inteira $\mathcal{C}'$ tal que $p_{\mathcal{C}'}$ é 2,2-viável e uma $(k+1)$ -coloração para $\mathcal{C}'$.*

Seja $\mathcal{C}$ uma circulação inteira tal que $p_{\mathcal{C}}$ é 1,2-viável e $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_k$ uma k -coloração para $\mathcal{C}$. Seja $\mathcal{C}_{k+1}$ a circulação

$$\mathcal{C}_1 \Delta \mathcal{C}_2 \Delta \dots \Delta \mathcal{C}_k$$

e $\mathcal{C}'$ a circulação $\mathcal{C}_1 + \mathcal{C}_2 + \dots + \mathcal{C}_{k+1}$. Primeiramente note que $\mathcal{C}_{k+1}$ é uma circulação inteira tal que $p_{\mathcal{C}_{k+1}}$ é 0,1-viável. Logo, $\mathcal{C}'$ é uma circulação inteira e

$\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_{k+1}$ é uma $(k+1)$ -coloração para $\mathcal{C}'$. Além disso, como $A\mathcal{C}_{k+1}$ é o conjunto das arestas cobertas por exatamente uma das funções $p_{c_1}, p_{c_2}, \dots, p_{c_k}$, temos que $p_{c'}$ é 2,2-viável.

Seja $\mathcal{C}'$ uma circulação inteira tal que $p_{c'}$ é 2,2-viável e $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_{k+1}$ uma $(k+1)$ -coloração para $\mathcal{C}'$. Seja $\mathcal{C}$ a circulação $\mathcal{C}_1 + \mathcal{C}_2 + \dots + \mathcal{C}_k$. Claramente $\mathcal{C}$ é uma circulação inteira. Como $p_{c'} = 2$, $p_{c_{k+1}}$ é 0,1-viável e $p_c = p_{c'} - p_{c_{k+1}}$, temos que p_c é 1,2-viável. Além disso, $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_k$ é uma k -coloração para $\mathcal{C}$.

Fato 4.5 *Para todo grafo sem 1-cortes e sem 3-cortes, existe uma circulação inteira $\mathcal{C}$ tal que p_c é 2,2-viável e uma 3-coloração para $\mathcal{C}$.*

Ou seja, a ausência de 1-cortes e 3-cortes em G é então uma condição suficiente para que $P2^+(G, 2, 2, 3)$ e $P2(G, 2, 2)$ tenham solução. A validade deste fato é consequência direta do fato 4.2 e do fato 4.4.

A conjectura da cobertura dupla por ciclos [43, 47] e uma de suas variantes bem conhecida [8, 18], enunciadas abaixo, dizem que a ausência de 1-cortes é uma condição necessária e suficiente para que $P2(G, 2, 2)$ e $P2^+(G, 2, 2, 5)$ tenham solução.

Conjectura 4.3 (cobertura dupla por ciclos) *Para todo grafo sem 1-cortes existe uma circulação inteira $\mathcal{C}$ tal que p_c é 2,2-viável.*

Conjectura 4.4 (cobertura dupla por ciclos com 5-coloração) *Para todo grafo sem 1-cortes existe uma circulação inteira $\mathcal{C}$ tal que p_c é 2,2-viável e uma 5-coloração para $\mathcal{C}$.*

Fato 4.6 *Para todo grafo sem 1-cortes, existe uma circulação inteira $\mathcal{C}$ tal que p_c é 4,4-viável e uma 7-coloração para $\mathcal{C}$.*

Pelo fato 4.3, existe uma circulação inteira $\mathcal{C}'$ tal que p_c é 1,3-viável e uma 3-coloração $\mathcal{C}'_1, \mathcal{C}'_2$ e $\mathcal{C}'_3$ para $\mathcal{C}'$. Para todo subconjunto não vazio S de $\{1, 2, 3\}$, seja $\mathcal{C}_S$ a circulação $\Delta\{\mathcal{C}'_i : i \in S\}$. Seja $\mathcal{C}$ a circulação

$$\sum \{\mathcal{C}_S : S \subseteq \{1, 2, 3\}, S \neq \emptyset\}.$$

Claramente, para todo S , $\mathcal{C}_S$ é uma circulação inteira tal que p_{c_S} é 0,1-viável. Logo $\mathcal{C}$ é uma circulação inteira e $\{\mathcal{C}_S : S \subseteq \{1, 2, 3\}, S \neq \emptyset\}$ é uma coloração

para $\mathcal{C}$. Na verdade, uma 7-coloração para $\mathcal{C}$. Para cada aresta e , seja S_e o conjunto $\{i \in \{1, 2, 3\} : e \in AC'_i\}$. Observe primeiramente que $S_e \neq \emptyset$. Temos que $e \in AC'_S$ se e somente se S é tal que $|S \cap S_e|$ é ímpar. A circulação $\mathcal{C}$ é tal que p_c é 4,4-viável pois, para cada aresta e , $|\{S \subseteq \{1, 2, 3\} : |S \cap S_e| \text{ é ímpar}\}| = 4$.

Fulkerson conjecturou [14] que para todo grafo cúbico sem 1-cortes existe uma coleção de seis emparelhamentos perfeitos que, juntos, cobrem cada aresta exatamente duas vezes. É possível construir, a partir de uma tal coleção de emparelhamentos, uma circulação inteira $\mathcal{C}$ tal que p_c é 4,4-viável e tem uma 6-coloração [18]. A seguinte variante da conjectura da cobertura dupla por ciclos [27] é então uma generalização da conjectura de Fulkerson para grafos não necessariamente cúbicos:

Conjectura 4.5 *Para todo grafo sem 1-cortes existe uma circulação inteira $\mathcal{C}$ tal que p_c é 4,4-viável e uma 6-coloração para $\mathcal{C}$.*

Pelo fato 4.4, a conjectura 4.1 é equivalente a conjectura 4.3, assim como a conjectura 4.2 é equivalente à conjectura 4.4.

4.5 O problema P3 com limites 1 e k

Nesta seção estudaremos o caso particular do problema P3 em que os limites inferiores valem 1 e os limites superiores valem k em todas as arestas do grafo. Mostraremos as implicações dos resultados de Jaeger [25] e enunciaremos três famosas conjecturas de Tutte relacionadas a esse caso.

Dada uma função q de AG em $\mathbf{Q}$, denotamos por $|q|$ a função definida por $|q|(e) = |q(e)|$ para toda aresta e .

Para todo grafo G , toda orientação D de G e toda circulação $\mathcal{C}$, existe uma circulação consistente $\mathcal{C}'$ tal que $p_{c'} = |p_c^D|$. Para mostrar isso, vamos primeiramente mostrar que p_c^D satisfaz as condições de Hoffman.

Denotaremos por $\vec{p}_c^D$ e $\overleftarrow{p}_c^D$ as funções definidas por:

$$\begin{aligned}\vec{p}_c^D(e) &= \sum \{m_c(C) : C \in \mathcal{C}, \vec{AC} \ni e\} \quad e \\ \overleftarrow{p}_c^D(e) &= \sum \{m_c(C) : C \in \mathcal{C}, \overleftarrow{AC} \ni e\}.\end{aligned}$$

Observe que $p_c^D = \vec{p}_c^D - \overleftarrow{p}_c^D$.

Denotando $\delta^+(X)$ e $\delta^-(X)$ por δ^+X e δ^-X , respectivamente, temos que, para todo corte X , $p_c^D(\delta^-X) = p_c^D(\delta^+X)$. Isso porque

$$\begin{aligned}
 p_c^D(\delta^-X) - p_c^D(\delta^+X) &= \vec{p}_c^D(\delta^-X) - \overleftarrow{p}_c^D(\delta^-X) - \vec{p}_c^D(\delta^+X) + \overleftarrow{p}_c^D(\delta^+X) \\
 &= \vec{p}_c^D(\delta^-X) + \overleftarrow{p}_c^D(\delta^+X) - (\overleftarrow{p}_c^D(\delta^-X) + \vec{p}_c^D(\delta^+X)) \\
 &= \sum\{|\vec{AC} \cap \delta^-X|_{m_c(C)} : C \in \mathcal{C}\} \\
 &\quad + \sum\{|\overleftarrow{AC} \cap \delta^+X|_{m_c(C)} : C \in \mathcal{C}\} \\
 &\quad - (\sum\{|\overleftarrow{AC} \cap \delta^-X|_{m_c(C)} : C \in \mathcal{C}\} \\
 &\quad + \sum\{|\vec{AC} \cap \delta^+X|_{m_c(C)} : C \in \mathcal{C}\}) \\
 &= \sum\{(|\vec{AC} \cap \delta^-X| + |\overleftarrow{AC} \cap \delta^+X|)_{m_c(C)} : C \in \mathcal{C}\} \\
 &\quad - \sum\{(|\overleftarrow{AC} \cap \delta^-X| + |\vec{AC} \cap \delta^+X|)_{m_c(C)} : C \in \mathcal{C}\} \\
 &= 0,
 \end{aligned}$$

pois $|\vec{AC} \cap \delta^-X| + |\overleftarrow{AC} \cap \delta^+X|$ é o número de vezes que o ciclo C usa uma aresta de um vértice de $VG - X$ para um vértice X , e $|\overleftarrow{AC} \cap \delta^-X| + |\vec{AC} \cap \delta^+X|$ é o número de vezes que o ciclo C usa uma aresta de um vértice de X para um vértice de $VG - X$. Da definição de ciclo, esses números são iguais e p_c^D realmente satisfaz as condições de Hoffman.

Para completar a demonstração, seja D' a orientação de G que coincide com D nas arestas onde p_c^D é positiva e, nas demais, é distinta de D . Observe que $|p_c^D|$ satisfaz as condições de Hoffman em relação à orientação D' . Submeta então G , D' e $|p_c^D|$ ao algoritmo 2 que devolverá uma circulação C' consistente com D' tal que $p_{C'} = |p_c^D|$.

A existência de uma circulação C tal que p_c é $1, k$ -viável não implica na existência de uma circulação consistente C' tal que $p_{C'}$ é $1, k$ -viável, ou seja, mesmo que o problema $P2(G, 1, k)$ tenha solução, o problema $P3(G, 1, k)$ pode não ter solução (veja exemplo 4.1).

No entanto, pelo fato abaixo, se o problema $P2^+(G, 1, 2, k)$ tem solução então o problema $P3(G, 1, 2k - 1)$ tem solução.

Fato 4.7 *Para todo grafo, se existe uma circulação inteira C tal que p_c é $1, 2$ -viável e uma k -coloração para C então existe uma circulação inteira consistente C' tal que $p_{C'}$ é $1, (2k - 1)$ -viável.*

Seja $\mathcal{C}$ uma circulação tal que $p_{\mathcal{C}}$ é 1,2-viável e $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_k$ uma k -coloração para $\mathcal{C}$. Seja $\mathcal{C}''$ a circulação

$$\mathcal{C}_1 + 2\mathcal{C}_2 + \dots + k\mathcal{C}_k$$

e D uma orientação qualquer de G . Observe que $|p_{\mathcal{C}''}^D|$ é $1, (2k-1)$ -viável. Seja D' a orientação de G que coincide com D nas arestas onde $p_{\mathcal{C}''}^D$ é positiva e, nas demais, é distinta de D . Submeta G , D' e $|p_{\mathcal{C}''}^D|$ ao algoritmo 2 que devolverá uma circulação consistente $\mathcal{C}'$ tal que $p_{\mathcal{C}'} = |p_{\mathcal{C}''}^D|$, ou seja, tal que $p_{\mathcal{C}'}$ é $1, (2k-1)$ -viável.

Exemplo 4.1 Um grafo G e uma circulação $\mathcal{C}$ que consiste dos três ciclos abaixo, todos com multiplicidade 1. A função $p_{\mathcal{C}}$ é 1,2-viável porém não existe uma circulação consistente $\mathcal{C}'$ tal que $p_{\mathcal{C}'}$ é 1,2-viável.

Fato 4.8 Para todo grafo sem 1-cortes e sem 3-cortes, existe uma circulação inteira consistente $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 1,3-viável.

Ou seja, a ausência de 1-cortes e 3-cortes em G é uma condição suficiente para que $P3(G, 1, 3)$ tenha solução. A validade deste fato é consequência direta dos fatos 4.4 e 4.7.

Tutte propôs [50] as seguintes conjecturas:

Conjectura 4.6 (circulação consistente 1,2-viável) Para todo grafo sem 1-cortes e sem 3-cortes, existe uma circulação inteira consistente $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 1,2-viável.

Conjectura 4.7 (circulação consistente 1,3-viável) Para todo grafo sem 1-cortes e sem uma subdivisão do grafo de Petersen, existe uma circulação inteira consistente $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 1,3-viável.

Conjectura 4.8 (circulação consistente 1,4-viável) *Para todo grafo sem 1-cortes, existe uma circulação consistente $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 1,4-viável.*

Fato 4.9 *Para todo grafo sem 1-cortes, existe uma circulação inteira consistente $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 1,7-viável.*

Pelo fato 4.3, existe uma circulação $\mathcal{C}'$ tal que $p_{\mathcal{C}'}$ é 1,3-viável e uma 3-coloração $\mathcal{C}_1, \mathcal{C}_2$ e $\mathcal{C}_3$ para $\mathcal{C}'$. Seja $\mathcal{C}''$ a circulação

$$\mathcal{C}_1 + 2\mathcal{C}_2 + 4\mathcal{C}_3$$

e D uma orientação qualquer de G . Observe que $|p_{\mathcal{C}''}^D|$ é 1,7-viável. Seja D' a orientação de G que coincide com D nas arestas onde $p_{\mathcal{C}''}^D$ é positiva e, nas demais, é distinta de D . Submeta G , D' e $|p_{\mathcal{C}''}^D|$ ao algoritmo 2 que devolverá uma circulação consistente $\mathcal{C}'$ tal que $p_{\mathcal{C}'} = |p_{\mathcal{C}''}^D|$, ou seja, tal que $p_{\mathcal{C}'}$ é 1,7-viável. Observe que se tivéssemos tomado $\mathcal{C}_1 + 2\mathcal{C}_2 + 3\mathcal{C}_3$ no lugar de $\mathcal{C}_1 + 2\mathcal{C}_2 + 4\mathcal{C}_3$ poderia existir uma aresta e usada por $\mathcal{C}_1$ e $\mathcal{C}_2$ num mesmo sentido e por $\mathcal{C}_3$ no sentido oposto, ou seja, $|p_{\mathcal{C}''}^D|(e)$ valeria 0.

Este fato mostra que a ausência de 1-cortes em G é condição necessária e suficiente para que P3($G, 1, 7$) tenha solução. Esta foi a primeira aproximação obtida para a conjectura 4.8:

No próximo capítulo mostraremos a melhor aproximação existente para esta conjectura: um trabalho de Seymour que garante a validade da conjectura com 5 no lugar de 4.

4.6 Circulações e colorações orientáveis

Definiremos nesta seção a noção de orientabilidade de um coloração de uma circulação $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 2,2-viável. Tal conceito pode ser usado para melhorar os resultados descritos nas seções anteriores.

Seja G um grafo e $\mathcal{C}$ uma circulação inteira tal que $p_{\mathcal{C}}$ é 2,2-viável. Dada uma k -coloração $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_k$ de $\mathcal{C}$, seja $\mathcal{C}'$ a circulação $\mathcal{C}_1 + \mathcal{C}_2 + \dots + \mathcal{C}_k$. Dizemos que a k -coloração $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_k$ é *orientável* se existe uma orientação D de G tal que $p_{\mathcal{C}'}^D = 0$.

Dadas duas circulações inteiras $\mathcal{C}_1$ e $\mathcal{C}_2$ tais que $p_{\mathcal{C}_1}$ e $p_{\mathcal{C}_2}$ são 0,1-viáveis e as arestas usadas ao mesmo tempo por $\mathcal{C}_1$ e $\mathcal{C}_2$ são usadas em sentido opostos,

podemos tomar $C_1 \Delta C_2$ de forma que as arestas usadas ao mesmo tempo por C_2 e $C_1 \Delta C_2$ são usadas em sentidos opostos. Observe que, neste caso, as arestas usadas ao mesmo tempo por C_1 e $C_1 \Delta C_2$ são usadas no mesmo sentido.

Fato 4.10 *Para todo grafo G , se existe uma circulação inteira consistente C tal que p_C é 1,2-viável então existe uma circulação inteira C' tal que $p_{C'}$ é 2,2-viável e uma 3-coloração orientável para C' .*

Seja C a circulação inteira consistente tal que p_C é 1,2-viável. Seja D a orientação de G com a qual C é consistente. Submeta G , D e p_C ao algoritmo 4 que devolverá uma 2-coloração C_1, C_2 para C , onde C_1 e C_2 são circulações consistentes com D . Seja C'_1 obtida de C_1 pela inversão de todos os seus ciclos. Tomemos $C'_1 \Delta C_2$ de tal forma que as arestas usadas ao mesmo tempo por C_2 e $C'_1 \Delta C_2$ são usadas em sentidos opostos. Seja então C' a circulação

$$C_1 + C_2 + C'_1 \Delta C_2.$$

Temos que C' é uma circulação inteira e $p_{C'}$ é 2,2-viável. Além disso, $p_{C'}^D = 0$, ou seja, $C_1, C_2, C'_1 \Delta C_2$ é uma 3-coloração orientável de C' .

Fato 4.11 *Para todo grafo G , se existe uma circulação inteira C tal que p_C é 2,2-viável e uma k -coloração orientável para C então existe uma circulação inteira consistente C' tal que $p_{C'}$ é 1,($k-1$)-viável.*

Seja D uma orientação qualquer de G e C uma circulação inteira tal que p_C é 2,2-viável e $C_1, C_2, \dots, C_k$ uma k -coloração para C . Seja C'' a circulação

$$0C_1 + 1C_2 + \dots + (k-1)C_k.$$

Observe que C'' é uma circulação inteira e que $p_{C''}^D$ é 1,($k-1$)-viável pois $C_1, C_2, \dots, C_k$ é uma coloração orientável de C . Seja D' a orientação de G que coincide com D nas arestas onde $p_{C''}^D$ é positiva e, nas demais, é distinta de D . Submeta G , D' e $|p_{C''}^D|$ ao algoritmo 2 que devolverá uma circulação inteira consistente C' tal que $p_{C'} = |p_{C''}^D|$, ou seja, tal que $p_{C'}$ é 1,($k-1$)-viável.

Pelo fato acima, a seguinte variação da conjectura da cobertura dupla por ciclos [27] implica a conjectura da circulação consistente 1,4-viável:

Conjectura 4.9 (cobertura dupla por ciclos com 5-coloração orientável)

Para todo grafo sem 1-cortes existe uma circulação inteira $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 2,2-viável e uma 5-coloração orientável para $\mathcal{C}$.

Pelo fato 4.10, vale a recíproca do fato 4.11 para $k = 3$. A recíproca do fato 4.11 vale também para $k = 4$, em decorrência dos dois próximos fatos e do fato 4.4.

Fato 4.12 *Para todo grafo G , se existe uma circulação inteira consistente $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 1,3-viável então existe uma circulação inteira $\mathcal{C}'$ tal que $p_{\mathcal{C}'}$ é 1,2-viável e uma 2-coloração para $\mathcal{C}'$.*

É possível generalizar a definição de circulações considerando que a função multiplicidade assume valores em outros grupos. Tutte estudou essa generalização e demonstrou que se existe uma circulação inteira consistente $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 1,3-viável então existe uma circulação $\mathcal{C}''$ cuja função multiplicidade assume valores em $\{0, 1\}^2$ tal que $p_{\mathcal{C}''}(e) \neq (0, 0)$ para toda aresta e . A função q_i de AG em $\{0, 1\}$ dada pela i -ésima componente de $p_{\mathcal{C}''}$, para i em $\{1, 2\}$, é uma função inteira e é 0,1-viável, obviamente. Além disso, q_i é função peso de uma circulação inteira $\mathcal{C}_i$. Seja $\mathcal{C}'$ a circulação $\mathcal{C}_1 + \mathcal{C}_2$. Do que foi dito acima $\mathcal{C}'$ é uma circulação inteira tal que $p_{\mathcal{C}'}$ é 1,2-viável e $\mathcal{C}_1, \mathcal{C}_2$ é uma 2-coloração para $\mathcal{C}'$.

Fato 4.13 *Para todo grafo G , se existe uma circulação inteira $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 2,2-viável e uma 3-coloração para $\mathcal{C}$ então existe uma circulação inteira $\mathcal{C}'$ tal que $p_{\mathcal{C}'}$ é 1,2-viável e uma 4-coloração orientável para $\mathcal{C}'$.*

Seja D uma orientação qualquer de G , $\mathcal{C}$ uma circulação inteira tal que $p_{\mathcal{C}}$ é 2,2-viável e uma 3-coloração $\mathcal{C}_1, \mathcal{C}_2$ e $\mathcal{C}_3$ para $\mathcal{C}$. Seja $\mathcal{C}''$ a circulação

$$\mathcal{C}_1 + \mathcal{C}_2 + \mathcal{C}_3.$$

Como $p_{\mathcal{C}}$ é 2,2-viável, temos que $|p_{\mathcal{C}''}^D|$ é 0,2-viável. Seja D' a orientação de G que coincide com D nas arestas onde $p_{\mathcal{C}''}^D$ é positiva e, nas demais, é distinta

de D . Submeta G , D' e $|p_{c''}^D|$ ao algoritmo 4 que devolverá uma 2-coloração $\mathcal{C}'_1, \mathcal{C}'_2$ para $\mathcal{C}''$, onde $\mathcal{C}_1$ e $\mathcal{C}_2$ são circulações consistentes com D' . Seja $\mathcal{C}'$ a circulação $\mathcal{C}'_1 + \mathcal{C}_1\Delta\mathcal{C}'_1 + \mathcal{C}_2\Delta\mathcal{C}'_1 + \mathcal{C}_3\Delta\mathcal{C}'_1$. A circulação $\mathcal{C}'$ é inteira e $p_{c'}$ é 2,2-viável pois $p_{c'_1}$ cobre as mesmas arestas que $p_{c''}$ já que $p_{c''}$ é 2,2-viável no grafo $G[AC'']$. Além disso, AC'' é o conjunto das arestas usadas no mesmo sentido por $\mathcal{C}_1, \mathcal{C}_2$ e $\mathcal{C}_3$ e ainda $\mathcal{C}''$ usa tais arestas exatamente nesse sentido. Para todo i , tomando $\mathcal{C}_i\Delta\mathcal{C}'_1$ de forma que as arestas usadas ao mesmo tempo por $\mathcal{C}'_1$ e $\mathcal{C}_i\Delta\mathcal{C}'_1$ são usadas em sentidos opostos, temos que $\mathcal{C}'_1, \mathcal{C}_1\Delta\mathcal{C}'_1, \mathcal{C}_2\Delta\mathcal{C}'_1, \mathcal{C}_3\Delta\mathcal{C}'_1$ é uma 4-coloração orientável para $\mathcal{C}'$.

Abaixo, um resumo das principais equivalências. Todas as circulações nele mencionadas são inteiras.

1. Existe uma circulação consistente $\mathcal{C}$ tal que p_c é 1,1-viável
 sse existe uma circulação $\mathcal{C}$ tal que p_c é 1,1-viável
 sse G é euleriano.
2. Existe uma circulação consistente $\mathcal{C}$ tal que p_c é 1,2-viável
 sse existe uma circulação $\mathcal{C}$ tal que p_c é 2,2-viável com uma 3-coloração orientável.
3. Existe uma circulação consistente $\mathcal{C}$ tal que p_c é 1,3-viável
 sse existe uma circulação $\mathcal{C}$ tal que p_c é 2,2-viável com uma 4-coloração orientável
 sse existe uma circulação $\mathcal{C}$ tal que p_c é 2,2-viável com uma 3-coloração
 sse existe uma circulação $\mathcal{C}$ tal que p_c é 1,2-viável com uma 2-coloração.
4. Existe uma circulação consistente $\mathcal{C}$ tal que p_c é 1,7-viável
 sse existe uma circulação $\mathcal{C}$ tal que p_c é 1,3-viável com uma 3-coloração
 sse existe uma circulação $\mathcal{C}$ tal que p_c é 4,4-viável e tem uma 7-coloração
 sse G não tem 1-cortes.

A seguir, uma tabela com as diversas implicações entre algumas das várias conjecturas mencionadas neste capítulo.

Legenda:

- 2C - Conjectura 4.1 (circulação 1,2-viável).
- 2C4 - Conjectura 4.2 (circulação 1,2-viável com 4-coloração).
- CDC - Conjectura 4.3 (cobertura dupla por ciclos).
- CDC5 - Conjectura 4.4 (cobertura dupla por ciclos com 5-coloração).
- $\vec{C}DC5$ - Conjectura 4.9 (cobertura dupla por ciclos com 5-coloração orientável).
- $\vec{C}4$ - Conjectura 4.8 (circulação consistente 1,4 viável).

4.7 A complexidade dos problemas

Nesta seção, mostramos que o problema $P3$ e o problema $P2^+$ são $\mathcal{NP}$ -completos.

Holyer mostrou [22] que o problema de encontrar uma 3-coloração das arestas de um grafo cúbico é $\mathcal{NP}$ -completo. Vamos mostrar que, dado um grafo G cúbico, existe uma 3-coloração das arestas de G se e somente se existe uma circulação inteira $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 1,2-viável e tem uma 2-coloração. Com isso, teremos demonstrado que o problema $P2^+(G, 1, 2, 2)$ é $\mathcal{NP}$ -completo.

Seja G um grafo cúbico. Suponha que exista uma 3-coloração das arestas de G , ou seja, que existam conjuntos E_1, E_2 e E_3 tais que $E_1 \cup E_2 \cup E_3 = AG$ e, para todo i em $\{1, 2, 3\}$, E_i é um emparelhamento em G . Como G é cúbico, E_i é perfeito. Sejam $\mathcal{C}_1$ e $\mathcal{C}_2$ as circulações tais que $p_{\mathcal{C}_i}$ é 0,1-viável e $A\mathcal{C}_i = E_i \cup E_3$ para todo i em $\{1, 2\}$. Seja $\mathcal{C}$ a circulação $\mathcal{C}_1 + \mathcal{C}_2$. Observe que $\mathcal{C}$ é uma circulação tal que $p_{\mathcal{C}}$ é 1,2-viável e claramente $\mathcal{C}_1, \mathcal{C}_2$ é uma 2-coloração para $\mathcal{C}$.

Suponha agora que existe uma circulação inteira $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 1,2-viável e uma 2-coloração $\mathcal{C}_1, \mathcal{C}_2$ para $\mathcal{C}$. Para todo i em $\{1, 2\}$, seja E_i o conjunto das

arestas cobertas apenas por p_{c_i} , e seja E_3 o conjunto das arestas cobertas por p_{c_1} e p_{c_2} . As arestas de E_2 são as arestas de peso 2. Como G é cúbico, temos que p_{c_i} cobre exatamente duas arestas incidentes em cada vértice para todo i em $\{1, 2\}$. Com isso, temos que E_1, E_2 e E_3 são emparelhamentos perfeitos, ou seja, formam uma 3-coloração das arestas de G .

Com isso terminamos a demonstração e concluímos que o problema $P2^+(G, 1, 2, 2)$ é $\mathcal{NP}$ -completo. Pela equivalência 3 do resumo da seção 4.6, $P2^+(G, 1, 2, 2)$ é equivalente a $P3(G, 1, 3)$, ou seja, tanto $P2^+$ quanto $P3$ são $\mathcal{NP}$ -completos.

Capítulo 5

O algoritmo de Seymour/Younger

Este capítulo trata do caso particular do problema P3 em que os limites inferiores valem 1 e os limites superiores valem k , para um natural k . Como já vimos no capítulo 4, Jaeger mostrou que, quando $k = 7$, o problema tem solução para todo grafo sem 1-cortes. Seymour melhorou este resultado [44], mostrando que, para todo grafo sem 1-cortes, o problema P3 tem solução para $k = 5$, ou seja, todo grafo sem 1-cortes tem uma circulação inteira consistente $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 1,5-viável. Neste capítulo descreveremos um algoritmo desenvolvido por Younger [52] e inspirado na demonstração de Seymour.

5.1 O algoritmo de Seymour/Younger

Dado um grafo 2-aresta-conexo G , o algoritmo de Seymour/Younger constrói uma circulação inteira $\mathcal{C}$ tal que $|p_{\mathcal{C}}^D|$ é 1,5-viável, onde D é uma orientação arbitrária de G . Para obter uma circulação inteira consistente $\mathcal{C}'$ tal que $p_{\mathcal{C}'}$ é 1,5-viável, seja D' a orientação de G que coincide com D nas arestas onde $p_{\mathcal{C}}^D$ é positiva e, nas demais, é distinta de D . Submeta G , D' e $|p_{\mathcal{C}}^D|$ ao algoritmo 2, que devolve uma circulação inteira consistente $\mathcal{C}'$ tal que $p_{\mathcal{C}'}$ é 1,5-viável. Como veremos, o algoritmo de Seymour/Younger tem complexidade $O(n^5)$, enquanto o algoritmo 2 tem complexidade $O(m^2)$. Logo a complexidade total do processo é $O(n^6 + m^2)$.

Uma *seqüência encadeada de ciclos* é uma seqüência $\langle C_0, e_1, C_1, \dots, e_k, C_k \rangle$ tal que $C_0, C_1, \dots, C_k$ são ciclos de G dois a dois disjuntos nos vértices e, para todo

j em $\{1, 2, \dots, k\}$, e_j é uma aresta com uma ponta em $VC_0 \cup VC_1 \cup \dots \cup VC_{j-1}$ e outra em VC_j e
 ou existe uma aresta f_j distinta de e_j com uma ponta em $VC_0 \cup VC_1 \cup \dots \cup VC_{j-1}$ e outra em VC_j ,
 ou existe um 2-corte X_j tal que $VC_j \subseteq X_j$ e $X_j \cap (VC_0 \cup VC_1 \cup \dots \cup VC_{j-1}) = \emptyset$.

Dada uma seqüência encadeada de ciclos $\langle C_0, e_1, C_1, \dots, e_k, C_k \rangle$, digamos $\mathcal{S}$, denotamos por $\mathcal{C}_{\mathcal{S}}$ a circulação que consiste dos ciclos $C_1, C_2, \dots, C_k$. Observe que $\mathcal{C}_{\mathcal{S}}$ é uma circulação inteira e $p_{\mathcal{C}_{\mathcal{S}}}$ é 0,1-viável. Dizemos que $\mathcal{S}$ gera G se $VC_0 \cup VC_1 \cup \dots \cup VC_k = VG$.

Dois fatos, demonstrados por Seymour, são essenciais para o funcionamento do algoritmo de Seymour/Younger:

1. Para todo grafo 2-aresta-conexo G existe uma seqüência encadeada de ciclos que gera G .
2. Para toda seqüência encadeada de ciclos que gera G existe uma circulação inteira $\mathcal{C}$ tal que $p_{\mathcal{C}}^D$ é 0,2-viável e cobre $AG - AC_{\mathcal{S}}$, onde D é uma orientação arbitrária de G .

O algoritmo então consiste de duas etapas: a primeira é a construção de uma seqüência encadeada de ciclos que gera G ; a segunda é a construção de uma circulação inteira $\mathcal{C}$ tal que $|p_{\mathcal{C}}^D|$ é 1,5-viável, a partir da seqüência encadeada de ciclos construída na primeira etapa.

5.2 Algoritmo de Seymour/Younger: primeira etapa

Nesta seção descrevemos, através do algoritmo abaixo, a primeira etapa do algoritmo de Seymour/Younger.

Algoritmo 10 (primeira etapa do algoritmo de Seymour/Younger)

Recebe um grafo 2-aresta-conexo G e devolve uma seqüência encadeada de ciclos que gera G .

Cada iteração do algoritmo começa com uma seqüência encadeada de ciclos $\langle C_0, e_1, C_1, \dots, e_i, C_i \rangle$. A primeira iteração começa com a seqüência $\langle C_0 \rangle$, onde C_0 é um ciclo qualquer de G . Cada iteração consiste em:

Seja S_i o grafo $G[AC_0 \cup AC_1 \cup \dots \cup AC_i \cup \{e_1, \dots, e_i\}]$.

Caso 1: $VS_i = VG$.

Pare e devolva $\langle C_0, e_1, C_1, \dots, e_i, C_i \rangle$.

Caso 2: $VS_i \neq VG$.

Seja $\bar{e}$ uma aresta em $\delta(VS_i)$ e v a ponta de $\bar{e}$ em $VG - VS_i$. Comece um novo processo iterativo com um conjunto de vértices Y tal que $v \in Y \subseteq VG - VS_i$ e $|\delta(Y) - \delta(VS_i)| \leq 1$. A primeira iteração deste processo começa com $VG - VS_i$. Cada iteração consiste em:

Caso 2.1: $|\delta(VS_i) \cap \delta(Y)| > 1$.

Seja f uma aresta em $\delta(VS_i) \cap \delta(Y)$ distinta de $\bar{e}$. Seja v' a ponta de f em Y . Seja X um corte mínimo em $G[Y]$ tal que $v \in X$ e $v' \notin X$. Se $\delta(X) \geq 1$ em $G[Y]$ então comece nova iteração do processo iterativo com X no papel de Y . Se $\delta(X) \geq 2$ em $G[Y]$ então seja C um ciclo em $G[Y]$ que passa por v e v' . Comece nova iteração do algoritmo com $\langle C_0, e_1, C_1, \dots, e_i, C_i, \bar{e}, C \rangle$ no papel de $\langle C_0, e_1, C_1, \dots, e_i, C_i \rangle$.

Caso 2.2: $|\delta(VS_i) \cap \delta(Y)| = 1$.

Seja C o ciclo $\langle v \rangle$. Comece nova iteração do algoritmo com $\langle C_0, e_1, C_1, \dots, e_i, C_i, \bar{e}, C \rangle$ no papel de $\langle C_0, e_1, C_1, \dots, e_i, C_i \rangle$.

Primeiramente note que se ocorre o caso 2.1 e $\delta(X) \geq 2$ então existem dois caminhos de v a v' em $G[Y]$ disjuntos nas arestas. Esses caminhos, juntos, determinam um ciclo em $G[Y]$ que passa por v e v' .

Agora vamos mostrar que no início de cada iteração do algoritmo $\langle C_0, e_1, C_1, \dots, e_i, C_i \rangle$ é uma seqüência encadeada de ciclos. Obviamente $\langle C_0 \rangle$ é uma seqüência encadeada de ciclos. No início de cada iteração onde ocorre o caso 2, supondo que $\langle C_0, e_1, C_1, \dots, e_i, C_i \rangle$ é uma seqüência encadeada de ciclos, vamos mostrar que $\langle C_0, e_1, C_1, \dots, e_i, C_i, \bar{e}, C \rangle$ é uma seqüência encadeada de ciclos. No caso 2.1, como $VC \subseteq Y \subseteq VG - VS_i$, temos que C e C_j são disjuntos

nos vértices para todo j em $\{0, 1, \dots, i\}$. Além disso, $\bar{e}$ e f tem uma ponta em $\delta(VS_i)$ e outra em VC . Logo, $\langle C_0, e_1, C_1, \dots, e_i, C_i, \bar{e}, C \rangle$ é de fato uma sequência encadeada de ciclos. No caso 2.2, também temos que C e C_j são disjuntos nos vértices para todo j em $\{0, 1, \dots, i\}$, pois $VC = \{v\} \subseteq Y$. Além disso, $\delta(Y) = 2$ e $Y \subseteq VG - VS_i$.

Vamos mostrar agora que, no início de cada iteração do processo iterativo do caso 2, $v \in Y \subseteq VG - VS_i$ e $|\delta(Y) - \delta(VS_i)| \leq 1$. Obviamente no início da primeira iteração a hipótese vale. Supondo que valha no início de uma iteração onde ocorre o caso 2.1 e $\delta(X) \leq 1$, então $v \in X \subseteq Y \subseteq VG - VS_i$ e $|\delta(X) - \delta(VS_i)| \leq 1$, pois $\delta(VS_i) \cap AG[Y] = \emptyset$, ou seja, $\delta(X) - \delta(VS_i) = \delta(X)$ em $G[Y]$.

O número de iterações do processo iterativo do caso 2 é $O(n)$, onde n é o número de vértices de G , pois o conjunto Y diminui de pelo menos um elemento a cada iteração. A determinação de f pode ser feita em tempo $O(m)$, onde m é o número de arestas de G , enquanto que a determinação de X pode ser feita através de um algoritmo para o cálculo de fluxo máximo e corte mínimo, ou seja, em tempo $O(n^3)$. Se $\delta(X) \geq 2$ então, a partir do fluxo máximo obtido, é possível construir C em tempo $O(n)$.

No total, portanto, o caso 2 pode ser executado em tempo $O(n^4)$. Como os ciclos $C_1, C_2, \dots, C_k$ são dois a dois disjuntos nos vértices, temos que $k \leq n$, logo o número de iterações do algoritmo é $O(n)$. A complexidade do algoritmo 10 é então $O(n^5)$.

5.3 O algoritmo de Seymour/Younger: segunda etapa

Dado um grafo G , uma orientação D , um natural k não nulo e um conjunto E de arestas de G , dizemos que uma função q de AG em $\mathbb{Z}^+$ *cobre_k* E se $q(e) \bmod k \neq 0$ para toda aresta e em E . Se $E = AG$, dizemos que q *cobre_k* G . Se $E = \{e\}$, dizemos que q *cobre_k* e .

Algoritmo 11 (Seymour/Younger) *Recebe um grafo 2-aresta-conexo G , uma orientação D e devolve uma circulação inteira C tal que $|p_C^D|$ é 1,5-viável.*

Primeiramente, submeta o grafo G ao algoritmo 10 que devolve uma seqüência encadeada de ciclos $\langle C_0, e_1, C_1, \dots, e_k, C_k \rangle$, digamos $\mathcal{S}$, que gera G . Para todo i em $\{0, 1, \dots, k\}$, seja S_i o grafo $G[AC_0 \cup AC_1 \cup \dots \cup AC_i \cup \{e_1, \dots, e_i\}]$ e, E_i o conjunto $AG - (AC_0 \cup AC_1 \cup \dots \cup AC_k \cup \{e_{i+1}, \dots, e_k\})$. Cada iteração do algoritmo começa com um número i em $\{0, 1, \dots, k\}$ e uma circulação inteira $\mathcal{C}$ tal que $p_{\mathcal{C}}^D$ cobre₃ E_i . Submeta G e S_k ao algoritmo 9, obtemos uma circulação fundamental $\mathcal{C}_0$ de S_k . A primeira iteração começa com k e $\mathcal{C}_0$ nos papéis de i e $\mathcal{C}$. Cada iteração consiste em:

Caso 1: $i = 0$.

Submeta G , D , 2 e $\mathcal{C}$ ao subalgoritmo 11.1 que devolve uma circulação $\mathcal{C}'$ tal que $|p_{\mathcal{C}'}^D|$ é 0,2-viável e, para toda aresta e , $p_{\mathcal{C}'}^D$ cobre e e se e somente se $p_{\mathcal{C}}^D$ cobre₃ e . Pare e devolva a circulação $\mathcal{C}' + 3\mathcal{C}_S$.

Caso 2: $i > 0$ e $p_{\mathcal{C}}^D$ cobre₃ e_i .

Comece nova iteração com $i - 1$ e $\mathcal{C}$ nos papéis de i e $\mathcal{C}$.

Caso 3: $i > 0$ e $p_{\mathcal{C}}^D$ não cobre₃ e_i .

Seja $\mathcal{C}$ um ciclo que usa e_i e exatamente uma aresta em E_i , digamos f_i . Seja $\mathcal{C}'$ aquela entre $\mathcal{C} + \mathcal{C}$ e $\mathcal{C} + 2\mathcal{C}$, tal que $p_{\mathcal{C}'}^D$ cobre₃ e_i e f_i . Comece nova iteração com $i - 1$ e $\mathcal{C}'$ nos papéis de i e $\mathcal{C}$.

Como S_0 é um grafo conexo e, para todo i em $\{0, 1, \dots, k\}$, a aresta e_i tem uma ponta em VS_{i-1} e a outra em VC_i , temos que o grafo S_i é conexo. Além disso, $\mathcal{S}$ gera G , logo S_k é um subgrafo gerador conexo de G e as condições de aplicabilidade do algoritmo 9 estão satisfeitas.

Vamos mostrar que se ocorre o caso 3 então existe um ciclo que passa por e_i e usa exatamente uma aresta em E_i . Se existe um 2-corte X_i tal que $VC_i \subseteq X_i$ e $X_i \cap (VC_0 \cup VC_1 \cup \dots \cup VC_{i-1}) = \emptyset$ então vamos mostrar que ocorre o caso 2. Seja f_i a aresta em $\delta(X_i)$ distinta de e_i . Como X_i é um 2-corte e $p_{\mathcal{C}}^D$ satisfaz as condições de Hoffman, temos que $p_{\mathcal{C}}^D(f_i) = p_{\mathcal{C}}^D(e_i)$. Mas $f_i \in E_i$, logo $p_{\mathcal{C}}^D$ cobre₃ f_i e portanto $p_{\mathcal{C}}^D$ cobre₃ e_i , ou seja, ocorre o caso 2 como queríamos demonstrar. Assim se ocorre o caso 3, existe uma aresta f_i distinta de e_i com uma ponta em VS_{i-1} e outra em VC_i . Sejam x_1 e y_1 as pontas de e_i em VS_{i-1} e VC_i , respectivamente,

e x_2 e y_2 as pontas de f_i em VS_{i-1} e VC_i , respectivamente. Como o subgrafo S_{i-1} é conexo, existe um caminho P de x_1 a x_2 em S_{i-1} . Obviamente existe um caminho P' de y_2 a y_1 em C_i . Além disso, $P\langle x_2, f_i, y_2 \rangle P'\langle y_1, e_i, x_1 \rangle$ é um ciclo, digamos C , pois C_i é disjunto de S_{i-1} . Claramente C usa e_i e f_i é a única aresta de C em E_i .

Denotemos por $\mathcal{C}_1$ e $\mathcal{C}_2$ as circulações $\mathcal{C} + C$ e $\mathcal{C} + 2C$ do caso 3. Vamos mostrar que pelo menos uma entre $p_{\mathcal{C}_1}^D$ e $p_{\mathcal{C}_2}^D$ cobre₃ e_i e f_i . Como $p_{\mathcal{C}}^D$ não cobre₃ e_i , temos que $p_{\mathcal{C}}^D(e_i) \bmod 3 = 0$, e então $p_{\mathcal{C}_1}^D(e_i) \bmod 3 \neq 0$ e $p_{\mathcal{C}_2}^D(e_i) \bmod 3 \neq 0$. Os números $p_{\mathcal{C}}^D(f_i) \bmod 3$, $p_{\mathcal{C}_1}^D(f_i) \bmod 3$ e $p_{\mathcal{C}_2}^D(f_i) \bmod 3$ são distintos, logo apenas um deles pode ser nulo, ou seja, pelo menos uma entre $p_{\mathcal{C}_1}^D$ e $p_{\mathcal{C}_2}^D$ cobre₃ e_i e f_i , como queríamos demonstrar.

Vamos agora mostrar que, no começo de cada iteração, $\mathcal{C}$ é uma circulação tal que $p_{\mathcal{C}}^D$ cobre₃ E_i . Na primeira iteração é óbvio pois $i = k$ e $\mathcal{C}$ é uma circulação fundamental de S_k . Supondo que $p_{\mathcal{C}}^D$ cobre₃ E_i no início de uma iteração onde ocorre o caso 2, claramente a hipótese vale na iteração seguinte. Supondo que a hipótese valha no início de uma iteração onde ocorre o caso 3, como $AC \cap E_{i-1} = \{e_i, f_i\}$, $p_{\mathcal{C}}^D$ e $p_{\mathcal{C}'}^D$ coincidem, exceto em AC , $E_{i-1} - E_i = \{e_i\}$ e $p_{\mathcal{C}'}^D$ cobre₃ e_i e f_i , temos que $p_{\mathcal{C}'}^D$ cobre₃ E_{i-1} .

No caso 1, $i = 0$ logo $p_{\mathcal{C}}^D$ cobre₃ E_0 . Desde que o subalgoritmo 11.1 funcione, a circulação $\mathcal{C}'$ é tal que $p_{\mathcal{C}'}$ cobre E_0 . Observe que $E_0 = AG - AC_S$. Seja $\mathcal{C}''$ a circulação $\mathcal{C}' + 3\mathcal{C}_S$. Para toda aresta e em $AG - AC_S$, temos que $p_{\mathcal{C}'}(e) = p_{\mathcal{C}''}(e)$. Logo $|p_{\mathcal{C}''}^D| \in \{1, 2\}$ para toda aresta e em $AG - AC_S$ pois $|p_{\mathcal{C}'}^D|$ é 0,2-viável e cobre $AG - AC_S$.

5.4 O subalgoritmo 11.1

Younger descreveu um algoritmo [52] que, dado um grafo G , uma orientação D de G e um natural não nulo k , transforma uma circulação inteira $\mathcal{C}$ em uma circulação inteira $\mathcal{C}'$ tal que $|p_{\mathcal{C}'}^D|$ é 0, k -viável e $p_{\mathcal{C}'}^D$ é nula exatamente nas arestas onde $p_{\mathcal{C}}^D$ é nula (mod $k+1$). Este algoritmo reduz o problema de encontrar uma circulação inteira consistente $\mathcal{C}'$ tal que $p_{\mathcal{C}'}$ é 1, k -viável ao problema de encontrar uma circulação inteira $\mathcal{C}$ tal que $p_{\mathcal{C}}^D$ cobre _{$k+1$} G .

Dado um grafo G , uma orientação D e uma circulação $\mathcal{C}$, dizemos que uma trilha T é orientada em relação a $p_{\mathcal{C}}^D$ se $p_{\mathcal{C}}^D(e) > 0$ para toda aresta e em $\vec{AT}$ e

$p_c^D(e) < 0$ para toda aresta e em $\overleftarrow{AT}$.

Subalgoritmo 11.1 *Recebe um grafo G , uma orientação D , um natural não nulo k , uma circulação inteira $\mathcal{C}$ e devolve uma circulação inteira $\mathcal{C}'$ tal que $|p_{\mathcal{C}'}^D|$ é $0, k$ -viável e, para toda aresta e , $p_{\mathcal{C}'}^D$ cobre e se e somente se $p_{\mathcal{C}}^D$ cobre e .*

Cada iteração do algoritmo começa com uma circulação inteira $\mathcal{C}'$ tal que $p_{\mathcal{C}'}^D(e) \bmod (k+1) = p_{\mathcal{C}}^D(e) \bmod (k+1)$ para toda aresta e . A primeira iteração começa com $\mathcal{C}$ no papel de $\mathcal{C}'$. Cada iteração consiste em:

Caso 1: $|p_{\mathcal{C}'}^D|$ é $0, k$ -viável.

Pare e devolva $\mathcal{C}'$.

Caso 2: Existe uma aresta $\bar{e}$ tal que $|p_{\mathcal{C}'}^D|(\bar{e}) > k$.

Seja x o início de $\bar{e}$ se $p_{\mathcal{C}'}^D(\bar{e}) < 0$, o fim de $\bar{e}$ se $p_{\mathcal{C}'}^D(\bar{e}) > 0$. Seja y a outra ponta de $\bar{e}$. Seja P um caminho de y a x orientado em relação a $p_{\mathcal{C}'}^D$ e C o ciclo $P \langle x, \bar{e}, y \rangle$. Comece nova iteração com $\mathcal{C}' - kC$ no papel de $\mathcal{C}'$.

Primeiramente vamos mostrar que no caso 2 existe um caminho de y a x orientado em relação a $p_{\mathcal{C}'}^D$. Seja Y o conjunto $\{v \in VG : \text{existe caminho de } y \text{ a } v \text{ orientado em relação a } p_{\mathcal{C}'}^D\}$. Vamos demonstrar que $x \in Y$. Para toda aresta e em $\delta^+(Y)$, $p_{\mathcal{C}'}^D(e) \leq 0$, e, para toda aresta e em $\delta^-(Y)$, $p_{\mathcal{C}'}^D(e) \geq 0$. Portanto, $p_{\mathcal{C}'}^D(\delta^+(Y)) \leq 0$ e $p_{\mathcal{C}'}^D(\delta^-(Y)) \geq 0$. Como $p_{\mathcal{C}'}^D$ satisfaz as condições de Hoffman, $p_{\mathcal{C}'}^D(\delta^+(Y)) = p_{\mathcal{C}'}^D(\delta^-(Y))$. Ou seja, $p_{\mathcal{C}'}^D(\delta^+(Y)) = p_{\mathcal{C}'}^D(\delta^-(Y)) = 0$. Logo $p_{\mathcal{C}'}^D(e) = 0$ para toda aresta e em $\delta(Y)$. Mas $p_{\mathcal{C}'}^D(\bar{e}) \neq 0$, então $\bar{e} \notin \delta(Y)$ e portanto $x \in Y$, como queríamos demonstrar.

É fácil ver que no início de cada iteração a circulação $\mathcal{C}'$ é inteira e tal que $p_{\mathcal{C}'}^D(e) \bmod (k+1) = p_{\mathcal{C}}^D(e) \bmod (k+1)$ para toda aresta e .

Para analisar o número de iterações do algoritmo introduziremos o conceito de excesso de uma função q de AG em $\mathbb{Z}^+$. O *excesso de uma função q em relação a k* , denotado por $exc(q)$, é o número $\sum\{q(e) - k : e \in AG, q(e) > k\}$.

Vamos mostrar que a cada iteração o excesso de $|p_{\mathcal{C}'}^D|$ diminui de pelo menos um. Seja $\mathcal{C}''$ a circulação $\mathcal{C}' - kC$. Para toda aresta e fora de AC , $p_{\mathcal{C}'}^D(e) = p_{\mathcal{C}''}^D(e)$. O ciclo C é orientado em relação a $p_{\mathcal{C}'}^D$, logo, para toda aresta e em AC , temos que, se $|p_{\mathcal{C}'}^D|(e) < k$ então $|p_{\mathcal{C}''}^D|(e) = k - |p_{\mathcal{C}'}^D|(e)$ senão $|p_{\mathcal{C}''}^D|(e) = |p_{\mathcal{C}'}^D|(e) - k$.

Portanto, para toda aresta e em AC tal que $|p_{c'}^D|(e) < k$, temos que $|p_{c''}^D|(e) < k$. Além disso, como $|p_{c'}^D|(\bar{e}) > k$, temos que $|p_{c''}^D|(\bar{e}) = |p_{c'}^D|(\bar{e}) - k$. Portanto $\text{exc}(|p_{c''}^D|) \leq \text{exc}(|p_{c'}^D|) - k$, ou seja, $\text{exc}(|p_{c'}^D|)$ diminui de pelo menos um a cada iteração e o número de iterações é no máximo $|p_c^D|(AG)$, ou seja, é $O(\|C\|)$.

O cálculo de C pode ser feito em tempo $O(m)$, onde m é o número de arestas do grafo G . A remoção de kC da circulação C' pode ser feita em tempo $O(m)$ também. Assim uma iteração do algoritmo tem complexidade $O(m)$ e o algoritmo portanto tem complexidade $O(m\|C\|)$.

Como consequência deste algoritmo, temos a seguinte versão equivalente da conjectura da circulação consistente 1,4-viável:

Conjectura 5.1 (circulação consistente 1,4-viável (mod 5)) *Para todo grafo sem 1-cortes, existe uma circulação C tal que p_c^D cobre G , onde D é uma orientação qualquer de G .*

De forma análoga, podemos definir versões equivalentes da conjectura da circulação consistente 1,3-viável e da conjectura da circulação consistente 1,2-viável.

Capítulo 6

Grafos em superfícies

Como já dissemos na introdução, os casos particulares dos problemas P2, P2⁺ e P3 estudados no capítulo 4 e 5 são uma generalização dos problemas de imersão de grafos em superfícies e coloração de faces. É por isso que as restrições dos problemas P2, P2⁺ e P3 a grafos imersíveis em uma dada superfície são interessantes e muito estudadas. Todas as conjecturas mencionadas no capítulo 4 são verdadeiras quando nos restringimos a grafos planares. A conjectura da cobertura dupla por ciclos e a conjectura da circulação consistente 1,4-viável valem também para grafos imersíveis no plano projetivo e no toro.

6.1 Imersões no plano e imersões fortes em outras superfícies

Nesta seção mostraremos que, quando nos restringimos a grafos planares, todas as conjecturas mencionadas no capítulo 4 são verdadeiras.

Ao tratar de grafos imersos em uma superfície, é natural restringir a atenção a grafos 2-aresta-conexos; ou, caso contrário, as fronteiras de algumas faces deixam de ser ciclos. Mas a condição de conexidade não é suficiente para garantir que todas as faces são delimitadas por ciclos, conforme discussão abaixo.

Seja G um grafo planar 2-aresta-conexo. Em qualquer mapa de G , a fronteira de cada face é um ciclo. Seja então $\mathcal{C}$ a circulação que consiste das fronteiras de todas as faces de um mapa de G . Como cada aresta aparece na fronteira de exatamente duas faces, a circulação $\mathcal{C}$ é tal que $p_{\mathcal{C}}$ é 2,2-viável. Ou seja, a conjectura da cobertura dupla por ciclos vale para grafos planares.

Fato 6.1 *Se as faces de um mapa M de um grafo planar 2-aresta-conexo G possuem uma k -coloração então existe uma circulação inteira C tal que p_C é 2,2-viável e uma k -coloração orientável para C .*

Seja M um mapa de G com uma k -coloração de faces. Seja C a circulação que consiste das fronteiras de todas as faces desse mapa e, para todo i em $\{1, 2, \dots, k\}$, seja C_i a circulação que consiste das fronteiras das faces de cor i . Temos que p_{C_i} é 0,1-viável pois faces de uma mesma cor não são adjacentes, ou seja, as fronteiras são disjuntas nas arestas.

Os três teoremas fundamentais sobre coloração de faces de grafos planares (teorema das cinco cores, teorema das quatro cores e teorema de Grötzsch), enunciados abaixo, quando traduzidos para a linguagem de circulações, têm consequências interessantes sobre as diversas conjecturas enunciadas no capítulo 4, conforme veremos.

Fato 6.2 *Para todo grafo planar sem 1-cortes, existe uma circulação inteira C tal que p_C é 2,2-viável e uma 5-coloração orientável para C .*

A validade desse fato é consequência direta do fato 6.1 e do conhecido teorema abaixo:

Teorema das cinco cores [7]: *Para todo mapa de um grafo planar 2-aresta-conexo existe uma 5-coloração das faces.*

Das implicações, vistas no capítulo 4, entre as variantes da conjectura da cobertura dupla por ciclos, temos que todas elas valem para grafos planares. Do fato 4.4, valem também as conjecturas sobre circulações 1,2-viáveis para grafos planares.

Fato 6.3 *Para todo grafo planar sem 1-cortes, existe uma circulação inteira C tal que p_C é 2,2-viável e uma 4-coloração orientável para C .*

Este fato é uma versão mais forte do fato anterior. Sua validade decorre do fato 6.1 e do também conhecido teorema abaixo:

Teorema das quatro cores (Appel e Haken) [3, 4]: *Para todo mapa de um grafo planar 2-aresta-conexo existe uma 4-coloração das faces.*

Como consequência do fato acima e do fato 6.1, concluímos a validade do seguinte:

Fato 6.4 *Para todo grafo planar sem 1-cortes, existe uma circulação inteira consistente $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 1,3-viável.*

Com isso, as conjecturas da circulação consistente 1,4-viável e da circulação consistente 1,3-viável valem para grafos planares.

Fato 6.5 *Para todo grafo planar G sem 1-cortes e sem 3-cortes, existe uma circulação inteira consistente $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 1,2-viável.*

Existe um mapa de G com uma 3-coloração das faces pelo teorema abaixo:

Teorema de Grötzsch [19] *Para todo mapa de um grafo planar 2-aresta-conexo sem 3-cortes existe uma 3-coloração das faces.*

Do fato 6.1 então existe uma circulação inteira $\mathcal{C}'$ tal que $p_{\mathcal{C}'}$ é 2,2-viável e uma 3-coloração orientável para $\mathcal{C}'$. Pelo fato 4.11, existe uma circulação inteira consistente $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 1,2-viável.

Fato 6.6 *Para todo grafo planar G sem 1-cortes, existe uma circulação inteira $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 4,4-viável e uma 6-coloração para $\mathcal{C}$.*

Pelo teorema das 4 cores, existe uma circulação inteira $\mathcal{C}'$ tal que $p_{\mathcal{C}'}$ é 2,2-viável e uma 4-coloração $\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3, \mathcal{C}_4$. Seja $\mathcal{C}$ a circulação $\mathcal{C}_1\Delta\mathcal{C}_2 + \mathcal{C}_1\Delta\mathcal{C}_3 + \mathcal{C}_1\Delta\mathcal{C}_4 + \mathcal{C}_2\Delta\mathcal{C}_3 + \mathcal{C}_2\Delta\mathcal{C}_4 + \mathcal{C}_3\Delta\mathcal{C}_4$. Observe que $p_{\mathcal{C}}$ é 4,4-viável e $\mathcal{C}_1\Delta\mathcal{C}_2, \mathcal{C}_1\Delta\mathcal{C}_3, \mathcal{C}_1\Delta\mathcal{C}_4, \mathcal{C}_2\Delta\mathcal{C}_3, \mathcal{C}_2\Delta\mathcal{C}_4, \mathcal{C}_3\Delta\mathcal{C}_4$ é uma 6-coloração para $\mathcal{C}$. Ou seja, a conjectura 4.5 vale para grafos planares.

Assim demonstramos que realmente todas as conjecturas do capítulo 4 valem para grafos planares.

Para superfícies distintas do plano, nem toda fronteira de face é um ciclo, mesmo que o grafo seja 2-aresta-conexo (veja exemplo 6.1).

Uma *imersão forte de um grafo G em uma superfície S* é uma imersão em que a fronteira de cada face é um ciclo. Dizemos que G tem uma *imersão forte em S* , se existe uma imersão de G em S tal que a fronteira de toda face é um ciclo.

A seguinte conjectura é bastante conhecida:

Conjectura da imersão forte: *Para todo grafo G , se existe uma imersão forte de G em alguma superfície S então existe uma circulação $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 2,2-viável.*

Exemplo 6.1 *Uma imersão do grafo completo de 5 vértices no plano projetivo (os pontos diagonalmente opostos são identificados). A fronteira da face f não é um ciclo.*

A validade desta conjectura implicaria na validade da conjectura da cobertura dupla por ciclos, pelo fato abaixo.

Fato 6.7 *Para todo grafo G , se existe uma imersão forte de G em alguma superfície S então existe uma circulação inteira C tal que p_c é 2,2-viável.*

Seja G um grafo 2-aresta-conexo para o qual existe uma imersão forte em alguma superfície S . A circulação C que consiste das fronteiras de todas as faces da imersão forte em S é tal que p_c é 2,2-viável. Observe também que se existe uma k -coloração para as faces da imersão então a circulação tem uma k -coloração. Além disso, se S é uma superfície orientada e existe uma k -coloração para as faces da imersão então a circulação tem uma k -coloração orientável. Temos assim as seguintes versões da conjectura da imersão forte [28]:

Conjectura da imersão forte com 5-coloração: *Todo grafo 2-aresta-conexo tem uma imersão forte com uma 5-coloração das faces em alguma superfície.*

Conjectura da imersão forte com 5-coloração orientável: *Para todo grafo G , se existe uma imersão forte de G em alguma superfície S então existe uma circulação C tal que p_c é 2,2-viável.*

6.2 Reduções a grafos 3-regulares e 5-regulares

Para resolver alguns casos particulares dos problemas P2, P2⁺ e P3, mais especificamente, para demonstrar a validade da conjectura da cobertura dupla por ciclos,

de suas variantes e da conjectura da circulação consistente 1,4-viável, podemos restringir a atenção a grafos cúbicos.

Descreveremos aqui um algoritmo para a conjectura da cobertura dupla por ciclos: o algoritmo demonstra que podemos nos restringir a grafos cúbicos para demonstrar a conjectura da cobertura dupla por ciclos. Ou seja, que é suficiente provar a conjectura para grafos cúbicos para mostrar que ela vale em geral. Vamos descrever a redução da seguinte maneira: suporemos a existência de um algoritmo que, dado um grafo cúbico 2-aresta-conexo, devolve uma circulação inteira $\mathcal{C}'$ tal que $p_{\mathcal{C}'}$ é 2,2-viável. Utilizando esse algoritmo, vamos mostrar como obter uma circulação $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 2,2-viável em um grafo 2-aresta-conexo arbitrário.

Seja G um grafo e f e f' arestas com exatamente uma ponta em comum, digamos v . O grafo obtido de G pelo desligamento de f e f' em v é o grafo obtido de G pela remoção das arestas f e f' e acréscimo de uma nova aresta ligando as pontas de f e f' distintas de v .

Uma das etapas da redução baseia-se numa construção de Fleischner [11], descrita pelo fato abaixo:

Construção de Fleischner: *Seja G um grafo simples e 3-aresta-conexo com um vértice v tal que $|\delta(v)| \geq 4$. Sejam e_1, e_2 e e_3 arestas distintas de $\delta(v)$ tais que $\{e_1, e_2, e_3\} \neq \delta(X)$ para todo 3-corte X . Sejam G_1 e G_3 os grafos obtido de G pelo desligamento de e_2 e e_3 , e e_1 e e_2 , respectivamente, em v . Então pelo menos um entre G_1 e G_3 é 2-aresta-conexo.*

Primeiramente observe que a hipótese de que $\{e_1, e_2, e_3\} \neq \delta(X)$ para todo 3-corte X é claramente necessária para que a construção seja válida.

Sejam X_1 e X_3 cortes tais que $v_1 \in X_1 - X_3$, $v_2 \in X_1 \cap X_3$, $v_3 \in X_3 - X_1$ e $v \notin X_1 \cup X_3$. Os conjuntos $X_1 - X_3$, $X_3 - X_1$, $X_1 \cap X_3$ e $\overline{X_1 \cup X_3}$ formam

uma partição de VG . Cada um deles é um corte pois não é vazio e, como G é 3-aresta-conexo, é na verdade um k -corte para $k \geq 3$. Existem então pelo menos 6 arestas com pontas em elementos distintos dessa partição de VG , três delas são e_1, e_2 e e_3 . Disso, é possível concluir que pelo menos um entre X_1 e X_3 é um k -corte para $k \geq 4$. Ou seja, não é possível que tanto X_1 quanto X_3 sejam 3-cortes. Portanto pelos menos um entre G_1 e G_3 é 2-aresta-conexo.

Para simplificar a descrição do algoritmo, vamos adotar a seguinte convenção: na definição de cada caso, fica subentendido que nenhum dos casos anteriores se aplica.

Algoritmo 12 (Redução a grafos 3-regulares) *Recebe um grafo 2-aresta-conexo G e devolve uma circulação $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 2,2-viável.*

O algoritmo é recursivo.

Caso 1: G tem um laço e .

Submeta $G - e$ ao algoritmo, obtendo uma circulação $\mathcal{C}'$ tal que $p_{\mathcal{C}'}$ é 2,2-viável. Seja C o ciclo formado apenas pela aresta e . Devolva $\mathcal{C}' + 2C$.

Caso 2: G tem um 2-corte X .

Sejam f e f' as duas arestas em $\delta(X)$. Submeta $G \setminus f$ ao algoritmo, obtendo uma circulação $\mathcal{C}'$ tal que $p_{\mathcal{C}'}$ é 2,2-viável. Seja C a circulação correspondente a $\mathcal{C}'$ em G . Devolva a circulação $\mathcal{C}$.

Caso 3: G tem um par de arestas paralelas.

Seja f e f' duas arestas paralelas. Submeta $G - f$ ao algoritmo, obtendo uma circulação $\mathcal{C}'$ tal que $p_{\mathcal{C}'}$ é 2,2-viável. Seja C o ciclo formado apenas pelas arestas f e f' e C' um ciclo de $\mathcal{C}'$ que usa f' . Devolva a circulação $\mathcal{C}' + (C \Delta C') + C$.

Caso 4: G tem um vértice v tal que $|\delta(v)| > 3$.

Sejam e_1, e_2 e e_3 arestas distintas de $\delta(v)$ tais que $\{e_1, e_2, e_3\} \neq \delta(X)$ para todo 3-corte X . Sejam G_1 e G_3 os grafos obtidos de G pelo desligamento de e_2 e e_3 , e e_1 e e_2 , respectivamente, em v . Seja r em $\{1, 3\}$ tal que G_r é 2-aresta-conexo. Submeta

G_r ao algoritmo, obtendo uma circulação $\mathcal{C}'$ tal que $p_{\mathcal{C}'}$ é 2,2-viável. Seja $\mathcal{C}$ a circulação correspondente a $\mathcal{C}'$ em G . Devolva a circulação $\mathcal{C}$.

Caso 5: G é simples e 3-regular.

Submeta G a um algoritmo hipotético, cuja existência estamos supondo, que devolve uma circulação inteira $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 2,2-viável. Devolva a circulação $\mathcal{C}$.

Observe que nos casos 1 a 4, o grafo que é submetido recursivamente ao algoritmo tem menor número de arestas que o grafo G . Nos casos 1 a 3 claramente o grafo que é submetido ao algoritmo recursivamente é 2-aresta-conexo. No caso 4, G é 3-aresta-conexo por que os casos anteriores não se aplicam, logo existe r em $\{1, 3\}$ tal que G_r é 2-aresta-conexo pela construção de Fleishner.

Assim, garantimos que para demonstrar a validade da conjectura da cobertura dupla por ciclos podemos nos restringir a grafos cúbicos. Ou seja, o contra-exemplo mínimo para esta conjectura é um grafo cúbico. Pelo que foi visto no capítulo 4, para grafos cúbicos com uma 3-coloração de arestas existe uma circulação $\mathcal{C}$ tal que $p_{\mathcal{C}}$ é 2,2-viável, ou seja, a conjectura vale. Em consequência, um contra-exemplo mínimo para esta conjectura é um snark, onde um *snark* é um grafo cúbico 2-aresta-conexo para o qual não existe uma 3-coloração das arestas.

É possível modificar este algoritmo de forma a concluir também que para demonstrar a validade das variantes da conjectura da cobertura dupla por ciclos e da circulação consistente 1,4-viável, podemos nos restringir a grafos cúbicos. Ou seja, um contra-exemplo mínimo para estas conjecturas é também um snark.

Para demonstrar a validade da conjectura da circulação consistente 1,2-viável podemos nos restringir a grafos 5-regulares sem 1-cortes e sem 3-cortes. Para isso, observe primeiramente que se G é 2-aresta-conexo sem 3-cortes então os grafos $G - e$, $G \setminus f$ e $G - f$ nos casos 1, 2 e 3 do algoritmo 12, respectivamente, são 2-aresta-conexos e sem 3-cortes. Já no caso 4, nem sempre o grafo G_r é 2-aresta-conexo e sem 3-cortes, porém com a seguinte alteração isso passa a ser verdade:

Caso 4: G tem um vértice v tal que $|\delta(v)| > 3$ e $|\delta(v)| \neq 5$.

A prova de que existe r em $\{1, 3\}$ tal que G_r é 2-aresta-conexo e sem 3-cortes é uma variação da demonstração apresentada para a construção de Fleishner.

6.3 Reduções a grafos com cintura grande

A *cintura* de um grafo é o comprimento do menor circuito não trivial. Goddyn [17] mostrou que para a demonstrar a validade da conjectura da cobertura dupla por ciclos podemos nos restringir a grafos com cintura pelo menos 7. O mesmo foi mostrado Möller [36] para a conjectura da circulação consistente 1,4-viável.

Como consequência do que será mostrado nesta seção, é possível demonstrar que a conjectura da cobertura dupla por ciclos e a conjectura da circulação consistente 1,4-viável valem para grafos imersíveis no plano projetivo e no toro.

Aqui descreveremos um algoritmo, baseado nas idéias de Möller, que serve para demonstrar que para provar a validade da conjectura da circulação consistente 1,4-viável (mod 5) podemos nos restringir a grafos com cintura pelo menos 7. Já vimos no capítulo 5 que essa conjectura é equivalente à conjectura da circulação consistente 1,4-viável. O algoritmo é uma generalização do algoritmo 12. Neste último aplicamos a construção de Fleishner a vértices, enquanto que no algoritmo que descreveremos a seguir aplicamos a construção de Fleishner a circuitos pequenos.

Como na seção anterior, este algoritmo supõe a existência de um algoritmo que, dado um grafo 2-aresta-conexo G' com cintura grande, devolve uma circulação C' tal que $p_{C'}$ cobre G' (mod 5) e, utilizando esse algoritmo, encontra em um grafo 2-aresta-conexo G uma circulação C tal que p_C cobre G (mod 5).

Para simplificar a descrição do algoritmo, como no algoritmo 12, vamos adotar a seguinte convenção: na definição de cada caso, fica subentendido que nenhum dos casos anteriores se aplica.

Algoritmo 13 (Redução para grafos com cintura maior que 6) *Recebe um grafo 2-aresta-conexo G , uma orientação D de G e devolve uma circulação C tal que p_C cobre G (mod 5).*

O algoritmo é recursivo. Seja k a cintura de G e C um circuito de comprimento mínimo. Digamos que $C = \langle v_1, e_1, v_2, \dots, v_k, e_k, v_1 \rangle$.

Caso 1: $AC = AG$.

Devolva $\{C\}$.

Caso 2: G tem um 2-corte X .

Sejam f e f' arestas tais que $\delta(X) = \{f, f'\}$. Submeta $G \setminus f$ ao algoritmo, obtendo uma circulação C' que cobre $G \setminus f \pmod{5}$. Seja C a circulação correspondente a C' em G . Devolva a circulação C .

Caso 3: $k = 6$ e $G - \{e_1, e_3, e_5\}$ não é 2-aresta-conexo.

Seja G' o grafo $G - \{e_2, e_3, e_5, e_6\} \cup \{f\}$, onde f é uma nova aresta com pontas v_3 e v_6 (veja figura abaixo). Submeta G' ao algoritmo, obtendo uma circulação C' que cobre $G' \pmod{5}$. Seja $\mathcal{C}_f$ a circulação que consiste de todos os ciclos de C' que contém a aresta f e tal que $m_{\mathcal{C}_f}(C) = m_{C'}(C)$ para todo C em $\mathcal{C}_f$. Seja C'' a circulação $C' + \mathcal{C}_f + (\mathcal{C}_f \Delta C')$ e $\mathcal{C}_j$ a circulação $C'' + jC$ para todo j em $\{1, 2, 3, 4\}$ e r em $\{1, 2, 3, 4\}$ tal que $p_{C_r}^D$ cobre AC . Devolva C_r .

Caso 4: $k \leq 5$ ou ($k = 6$ e $G - \{e_1, e_3, e_5\}$ é 2-aresta-conexo).

Seja E um subconjunto não vazio de AC tal que $|AC - E| \leq 3$ e $G - E$ é 2-aresta-conexo. Seja G' o grafo $G - E$. Submeta G' ao algoritmo, obtendo uma circulação C' que cobre $G' \pmod{5}$. Seja $\mathcal{C}_j$ a circulação $C'' + jC$ para todo j em $\{1, 2, 3, 4\}$ e r em $\{1, 2, 3, 4\}$ tal que $p_{C_r}^D$ cobre $AC \pmod{5}$. Devolva C_r .

Caso 5: $k \geq 7$.

Submeta G a um algoritmo hipotético que devolve uma C circulação tal que p_C^D cobre $G \pmod{5}$. Devolva a circulação C .

Observe que nos casos 2 a 4, o grafo que é submetido recursivamente ao algoritmo tem menor número de arestas que o grafo G . É preciso verificar também se o grafo que é submetido recursivamente ao algoritmo é 2-aresta-conexo. No caso 2 claramente isto é verdade. Nos casos 3 e 4, o grafo G' é 3-aresta-conexo, logo, no caso 4, se $k \leq 4$ então o conjunto E pode consistir de uma única aresta de C . Claramente neste caso o grafo $G - E$ é certamente 2-aresta-conexo. No caso 4, se $k = 5$ então seja G'' o grafo obtido de G pela contração das arestas com duas pontas em VC . Seja u o vértice que corresponde a VC em G'' . Aplicando a construção de Fleischner ao grafo G'' no vértice u , podemos concluir que pelo menos um entre $G - \{e_1, e_3\}$ e $G - \{e_2, e_4\}$ é 2-aresta-conexo.

Se $k = 6$ então seja G'' o grafo obtido de G pela contração das arestas com duas pontas em VC . Seja u o vértice que corresponde a VC em G'' . Aplicando a construção de Fleischner ao grafo G'' duas vezes, podemos concluir que um entre $G - \{e_1, e_3, e_5\}$ e $G - \{e_2, e_3, e_5, e_6\} \cup \{f\}$ é 2-aresta-conexo. Ou seja, ou se aplica o caso 3 ou o caso 4.

Tanto no caso 3 quanto no caso 4, é preciso mostrar que existe r . Para isso, no caso 3, temos que notar que $\mathcal{C}''$ é uma circulação em G tal que $p_{\mathcal{C}''}^D(e) \bmod 5 \neq 0$

para toda aresta em $AG - \{e_1, e_3, e_5\}$. Além disso, $p_{c''}^D(e_2) = p_{c''}^D(e_6)$, pois $e_2, e_6 \notin AG'$, e $p_{c''}^D(e_3) = p_{c''}^D(e_5) = 0$. Temos quatro possibilidades para a multiplicidade de C em $\mathcal{C}$ e no máximo três valores que não podem ser escolhidos: o valor que faz com que e_2 e e_6 tenham peso $0 \pmod{5}$, o valor que faz com que e_1 tenha peso $0 \pmod{5}$ e o valor que faz com que e_4 tenha peso $0 \pmod{5}$. No caso 4, temos novamente quatro possibilidades para a multiplicidade de C em $\mathcal{C}$ e no máximo três valores que não podem ser escolhidos pois $|AC - E| \leq 3$. Ou seja, sempre existe r em $\{1, 2, 3, 4\}$ tal que $p_{c_r}^D$ cobre $AC \pmod{5}$.

Com isso, garantimos que para demonstrar a validade da conjectura da circulação consistente 1,4-viável $\pmod{5}$, e portanto a conjectura da circulação consistente 1,4-viável, podemos nos restringir a grafos com cintura pelo menos 7.

Com bases nas idéias de Goddyn [17], é possível modificar este algoritmo 13 e garantir que, para a conjectura da cobertura dupla por ciclos, também podemos nos restringir a grafos com cintura pelo menos 7.

Para a conjectura da cobertura dupla por ciclos e para a conjectura da circulação consistente 1,4-viável, podemos concluir então que um contra-exemplo mínimo é um snark com cintura maior que 6. Jaeger propôs [26] a seguinte conjectura:

Conjectura dos snarks: Não existem snarks com cintura maior que 6.

A validade dessa conjectura implicaria então na validade das conjectura da cobertura dupla por ciclos e da conjectura da circulação consistente 1,4-viável.

6.4 Grafos imersíveis no plano projetivo e no toro

Fazendo uma pequena alteração no algoritmo 13 obtemos um algoritmo que demonstra a validade da conjectura da circulação consistente 1,4-viável para grafos imersíveis no plano, no plano projetivo ou no toro. Ou seja, a conjectura da circulação consistente 1,4-viável vale também para grafos imersíveis no plano projetivo ou no toro. Analogamente, é possível modificar o algoritmo 13 de forma a concluir que a conjectura da cobertura dupla por ciclos vale também para grafos imersíveis no plano projetivo ou no toro.

A alteração no algoritmo é bastante simples: para grafos com cintura menor que 3 o algoritmo funciona igualmente. Se a cintura for pelo menos 3, ou seja, o grafo não tem laços nem arestas paralelas, então neste caso ao invés de tomarmos circuitos, tomamos apenas fronteiras de faces. Ou seja, neste caso o circuito C será uma fronteira de face. Dessa forma, garantimos que os grafos que são submetidos recursivamente ao algoritmo continuam sendo imersíveis no plano, no plano projetivo ou no toro. Além disso, é preciso mostrar que nunca ocorre o caso 5, ou seja, que todo grafo 3-aresta-conexo imersível no plano, no plano projetivo ou no toro tem uma face de comprimento no máximo 6.

Dada uma superfície S , se G é um grafo que possui uma imersão em S então o número $|VG| - |AG| + |F|$, onde F é o conjunto das faces de uma imersão de G em S , é uma constante que independe da imersão considerada. Esse número é a *característica de Euler da superfície S* e será denotado por $\theta(S)$. As características de Euler do plano, do plano projetivo e do toro são respectivamente 2, 1 e 0. A *característica de Euler de um grafo G* , denotada por $\theta(G)$, é o número $\max\{\theta(S) : S \text{ é uma superfície e existe uma imersão de } G \text{ em } S\}$.

Vamos mostrar que um grafo 3-aresta-conexo G que não tem nenhuma face de comprimento menor que 7 é tal que $\theta(G) < 0$. Seja S uma superfície onde G é imersível. O número de arestas na fronteira de cada face é pelo menos 7. Cada aresta aparece na fronteira de duas faces (eventualmente aparece duas vezes na fronteira de uma mesma face pois a imersão considerada pode não ser forte), logo $7f \leq 2m$. Cada vértice tem grau pelo menos 3 pois G é 3-aresta-conexo, logo $2m \geq 3n$. Assim $|VG| - |AG| + |F| = n - m + f \leq \frac{2m}{3} - m + \frac{2m}{7} = \frac{14m - 21m + 6m}{21} = -\frac{m}{21} < 0$. Ou seja, de fato G não é imersível nem no plano, nem no plano projetivo, nem no toro.

Com isso, concluímos que a conjectura da circulação consistente 1,4-viável, assim como a conjectura da cobertura dupla por ciclos, vale para grafos imersíveis no plano, no plano projetivo e no toro.

Usando uma idéia semelhante, Steinberg e Younger [46] demonstraram que a conjectura da circulação consistente 1,2-viável vale para grafos sem 1-cortes e sem 3-cortes imersíveis no plano projetivo.

Bibliografia

- [1] L. M. Almeida e Silva,
Fluxos inteiros em grafos.
Dissertação de Mestrado, UNICAMP, Campinas (1991).
- [2] K. Appel, W. Haken,
The four color proof suffices.
The Mathematical Intelligencer 8 1 (1976) 10-20.
- [3] K. Appel, W. Haken,
Every planar map is four colorable part I: discharging.
Illinois J. Math. 21 (1977) 429-490.
- [4] K. Appel, W. Haken,
Every planar map is four colorable part II: reducibility.
Illinois J. Math. 21 (1977) 491-567.
- [5] E. M. Arkin, C. H. Papadimitriou,
On the complexity of circulations.
J. Algorithms 7 (1986) 134-145.
- [6] J. C. Bermond,
Shortest coverings of graphs with cycles.
J. Combinatorial Theory (B) 35 (1983) 297-308.
- [7] J. A. Bondy, U. S. R. Murty,
Graph theory with applications.
North Holland, New York, Oxford (1976).
- [8] U. A. Celmins,
On cubic graphs that do not have an edge 3-coloring.

- Thesis, University of Waterloo, Ontario, Canada (1984).
- [9] J. Edmonds, R. M. Karp,
Theoretical improvements in algorithmics efficiency for network flow problems.
J. ACM 19 (1972) 248-264.
- [10] P. Feofiloff, C.L. Lucchesi,
Algoritmos para igualdades minimax em grafos.
VI Escola de Computação, UNICAMP (1988).
- [11] H. Fleischner,
Eine Gemeinsame Basis für die Theorie der Eulerschen Graphen und den Satz von Petersen.
Monatshefte für Mathematik 81 (1976) 267-278.
- [12] L. R. Ford, D. R. Fulkerson,
Maximal flow through a network.
Canadian J. of Math. 8 (1956) 399-404.
- [13] P. Fraisse,
Cycle covering in bridgeless graphs.
J. Combinatorial Theory (B) 39 (1985) 146-152.
- [14] D. R. Fulkerson,
Blocking and antiblocking pairs of polyhedra.
Math. Programming 1 (1971) 168-197.
- [15] M. Gardner,
Mathematical games.
Sci. Amer. 234 (1976) 126-130.
- [16] M. R. Garey, D. S. Jonhson, L. Stockmeyer.
Some simplified $\mathcal{NP}$ -complete graph problems.
Theor. Computer Sci. 1 (1976) 237-267.
- [17] L. Goddyn,
A girth requirement for cycle double cover conjecture.

- Annals Disc. Math. 27, ed. B. R. Alspach and C. D. Godsil,
North-Holland, Amsterdam, New York, Oxford (1985) 13-26.
- [18] L. Goddyn,
Cycle cover of graphs.
Thesis, University of Waterloo, Ontario, Canada (1988).
- [19] H. Grötzsch,
Zur Theorie der Diskreten Gebilde.
VII Ein Dreifarbensatz für dreikreisfreie Netze auf der Kugel,
Wiss. Z. Martin-Luther-Univ. Halle-Wittenberg Math. Natur.
Reihe. 8 (1958/1959) 109-120.
- [20] B. Grünbaum,
Grötzsch's theorem on 3-colorings
Michigan Math. J. 10 (1963) 303-310.
- [21] A. J. Hoffman,
Some recent applications of the theory of linear inequalities to extremal analysis.
Proc. Symposium on Appl. Math. 10 (1960) 113-126.
- [22] I. Holyer,
The NP-completeness of edge colouring.
SIAM J. Comput. 10 (1981) 718-720.
- [23] R. Isaacs,
Infinite families of nontrivial trivalent graphs which are not Tait colorable.
Amer. Math. Monthly 82 no 3 (1975) 221-239.
- [24] F. Jaeger,
Balanced valuations and flows in multigraphs.
Proc. Amer. Math. Soc. 55 (1976) 237-242.
- [25] F. Jaeger,
Flows and generalized coloring theorems in graphs.
J. Combinatorial Theory (B) 26 (1979) 205-216.

- [26] F. Jaeger, T. Swart,
Conjecture 1.
Combinatorics 79, Annals Disc. Math. 9 ed. M. Deza, I. G. Rosenberg,
North-Holland, Amsterdam, New York, Oxford (1980) 305.
- [27] F. Jaeger,
On circular flows in graphs.
Proc. Hung. Coll. on Combinatorics, Eger, Hungary (1981) 391-402.
- [28] F. Jaeger,
A survey of the cycle double cover conjecture.
Annals of Discrete Mathematics 27 (1985) 1-12.
- [29] E. L. Lawler,
Combinatorial optimization: network and matroids.
Holt, Rinehart and Winston, New York (1976).
- [30] C. H. C. Little, R. D. Ringeisen,
On the strong graph embedding conjecture.
Proc. of the Ninth South. Conf. on Combin. Graph Theory and Comput.
Utilitas Mathematica, Winnipeg (1978) 479-487.
- [31] C. H. C. Little, W. T. Tutte, D. H. Younger,
A theorem on integer flows.
ARS Combinatoria 26 (A) (1988) 109-112.
- [32] V. M. Malhotra, M. P. Kumar, S. N. Maheshwari,
An $O(n^3)$ algorithm for finding maximum flows in networks.
Inf. Process. Lett. (1978) 277-278.
- [33] MacLane,
A structural characterization of planar combinatorial graphs.
Math. J. 3 (1937) 340-472.
- [34] N. Megiddo,
Combinatorial optimization with rational objective functions.
Math. Op. Research 4 (1979) 414-424.

- [35] G. J. Minty,
A theorem on three-coloring the edges of a trivalent graph.
J. Combinatorial Theory 2 (1967) 164-167.
- [36] M. Möller,
5-Flüsse auf Graphen vom Geschlecht 1.
Diplomarbeit, Bielefeld (1984)
- [37] M. Möller, H. G. Carstens, G. Brinkmann,
Nowhere-zero flows in low genus graphs.
J. Graph Theory 12 (1988) 183-190.
- [38] C. St.-J. A. Nash-Williams,
Edge-disjoint spanning trees of finite graphs.
J. London Math. Soc. 36 (1961) 445-450.
- [39] C. St.-J. A. Nash-Williams,
Decomposition of finite graphs into forests.
J. London Math. Soc. 39 (1964) 12.
- [40] O. Ore,
The four color problem.
Academic Press, New York, London (1967).
- [41] M. Preissmann,
Sur les colorations des arêtes des graphes cubiques.
Thesis, Grenoble, France (1981).
- [42] J. Roskind, R. Tarjan,
A note on finding minimum-cost edge-disjoint spanning trees.
Math. Op. Resear. 10 4 (1985) 701-708.
- [43] P. D. Seymour,
Sums of circuits.
Graph Theory and Related Topics, ed. J. A. Bondy, U. S. R. Murty,
Academic Press, New York (1979).

- [44] P. D. Seymour,
Nowhere-zero flows.
J. Combinatorial Theory (B) 30 (1981) 130-135.
- [45] R. Steinberg,
Tutte's 5-flow conjecture for the projective plane.
J. Graph Theory 8 (1984) 277-285.
- [46] R. Steinberg, D. H. Younger,
Grötzsch's theorem for the projective plane.
ARS Combinatoria 28 (1989) 15-31.
- [47] G. Szekeres,
Polyhedral decomposition of cubic graphs.
Bull. Austral. Math. Soc. 8 (1973) 367-387.
- [48] P. G. Tait,
Remarks on Coloring of Maps.
Proc. Roy. Soc. Edinb. (A) 10 (1880) 729.
- [49] W. T. Tutte,
On the imbedding of linear graphs in surfaces.
Proc. London Math. Soc. Ser. 2 51 (1949) 474-489.
- [50] W. T. Tutte,
A contribution to the theory of chromatic polynomials.
Canad. J. Math. 6 (1954) 80-91.
- [51] W. T. Tutte,
On the algebraic theory of graph colorings.
J. Combinatorial Theory 2 (1966) 15-50.
- [52] D. H. Younger,
Integer flows.
J. Graph Theory 7 (1983) 349-357.
- [53] J. W. T. Youngs,
Minimal imbeddings and the genus of a graph.
J. Math. and Mec. 12 (1963) 303-315.

Índice

- $\mathcal{C} + tC$, 13
- $\mathbb{Z}^+/2$, 45
- h_t , 39
- $q + tC$, 26
- $[L, R]$ -fluxo c -viável, 39
- $[L, R]$ -fluxo entre dois vértices, 39
- $NJ_G(q)$, 29
- VT , 3
- $\overleftarrow{AT}$, 3
- AT , 3
- AC , 49
- $\overrightarrow{AT}$, 3
- cobre _{k} , 66
- $\delta^+(X)$, 10
- $\delta^-(X)$, 10
- $\Delta_{c,f}$, 38
- $\delta(X)$, 10
- Δ , 50
- k -coloração, 7
- k -coloração orientável, 57
- k -corte, 10
- m_c , 4
- p_c , 4
- p_c^D , 4
- $\overleftarrow{p}_c^D$, 54
- $\overrightarrow{p}_c^D$, 54
- $\sigma_v^{(k)}$, 43
- $Sup_G(q)$, 27
- $\|\mathcal{C}\|$, 20
- $\tau_v^{(k)}$, 43
- algoritmo 10: primeira etapa do algoritmo de Seymour/Younger, 64
- algoritmo 11: Seymour/Younger, 66
- algoritmo 2: calcula circulação consistente com peso pré-definido, 13
- algoritmo 3: transforma solução fracionária em inteira, 16
- algoritmo 4: coloração de circulações inteiras consistentes, 18
- algoritmo 5: primeira etapa do algoritmo de Seymour, 23
- algoritmo 6: segunda etapa do algoritmo de Seymour, 27
- algoritmo 7: Jaeger, para grafos 4-aresta-conexos, 48
- algoritmo 8: Jaeger, para grafos 3-aresta-conexos, 48
- algoritmo 9: construção de circulações fundamentais, 50
- aresta $[L, R]$ -saturada, 39
- aresta de corte, 28
- aresta de ligação, 33

- aresta especial, 26
- aresta máxima, 27
- aresta saturada, 38
- aumentação, 38, 39
- aumentação máxima, 38, 39
- caminho $[L, R]$ -aumentador, 39
- caminho aumentador, 38
- característica de Euler de uma superfície, 81
- ciclo, 4
- ciclo consistente, 4
- cintura, 77
- circuito, 27
- circuito adequado em relação a uma função, 27
- circulação, 4
- circulação consistente, 4
- circulação fundamental, 48
- circulação inteira, 4
- circulações que casam em um corte, 33
- cobre, 47
- coloração de uma circulação, 7
- condições de Hoffman, 11
- condições de Seymour, 22
- conjectura da circulação 1,2-viável com 4-coloração, 52
- conjectura da circulação consistente 1,4-viável (mod 5), 70
- conjectura da cobertura dupla por ciclos, 53
- conjectura da cobertura dupla por ciclos com 5-coloração, 53
- conjectura da imersão forte, 73
- conjectura da imersão forte com 5-coloração, 74
- conjectura da imersão forte com 5-coloração orientável, 74
- conjectura de circulação 1,2-viável, 52
- conjectura dos snarks, 80
- conjecturas de Tutte, 56
- corte, 10
- corte trivial, 10
- fluxo c -viável, 38
- fluxo entre dois vértices, 38
- fluxo máximo, 38
- função ℓ, u -viável, 4
- função folga, 38, 39
- função justa, 26
- função multiplicidade, 4
- função peso, 4
- funções lineares, 39
- imersão forte, 73
- intensidade de um fluxo, 38
- limite inferior, 4
- limite superior, 4
- limites, 4
- limites equivalentes, 23
- limites inteiros, 7
- problema $P2^+$, 8
- problema $P3^+$, 8
- problema $P1$, 5, 11
- problema $P1r$, 15
- problema $P2$, 5, 21
- problema $P2r$, 21

problema P3, 5
problemas P1r, P2r e P3r, 6
redução de $P3^+$ para P3, 20
relaxações fracionárias, 6
seqüência encadeada de ciclos, 63
snark, 77
subalgoritmo 11.1, 69
subalgoritmo 6.1: calcula circuito
 com prescrição de arestas, 31
subalgoritmo 6.2: junta duas circulações
 que casam em um corte, 33
subalgoritmo 6.3: calcula valor
 máximo de decréscimo do circuito, 35
subalgoritmo 6.4, 40
suporte de uma função, 27
tamanho do uma circulação, 20
teorema das cinco cores, 72
teorema das quatro cores, 72
teorema de Grötzsch, 73
trilha, 3
trilha orientada em relação a uma
 função de AG em $\mathbb{Q}^+$, 68