

Cicero Samuel Santos Morais

SUBGRAFOS RAINBOW E COBERTURA DE ARESTAS EM GRAFOS

5 de fevereiro de 2026¹

¹Iniciado em 27 de novembro de 2025

Sumário

Sumário	1
1 Preliminares	3
2 Subgrafos <i>rainbow</i>	5
2.1 Árvores	5
3 Cobertura de arestas por diferentes estruturas e subgrafos	9
3.1 O problema do Carteiro Chinês	9
3.2 Florestas	11
3.3 Caminhos	11
3.4 Ciclos	13
3.5 Passeios em digrafos	13
4 Fronteira, Contorno e Periferia	17
4.1 Grafos cordais e vértices extremos	18
5 Tópicos em Matróides e Coberturas	19
5.1 Algumas famílias (potencialmente) úteis de matróides	19
5.2 Cobertura de matróides	20
5.3 Interseção de matróides	22
Bibliografia	25

Capítulo 1

Preliminares

Uma coloração de arestas de um grafo $G = (V, E)$ é uma atribuição de cores $c : E(G) \rightarrow \mathbb{N}$ a cada aresta de G . Dizemos que a coloração é *própria* se, para todo par de arestas e e f que compartilham uma extremidade, temos que $c(e) \neq c(f)$. O *número aresta-cromático* de G , denotado por $\chi'(G)$, é igual a menor quantidade de cores com as quais G admite uma coloração própria de arestas. Um limitante trivial para o número aresta-cromático é o grau máximo de G , isto é, $\Delta(G) \leq \chi'(G)$. König provou que a igualdade vale quando o grafo é bipartido.

Teorema 1.1 (König). *Se G é bipartido, então $\chi'(G) = \Delta(G)$.*

Shannon e Vizing provaram limitantes superiores para grafos quaisquer em função do grau máximo e da multiplicidade de arestas do grafo. A *multiplicidade de arestas* de um grafo G , denotada por $\mu(G)$, é a quantidade máxima de arestas múltiplas (com as mesmas extremidades) de G . Quando G é simples, $\mu(G) = 1$.

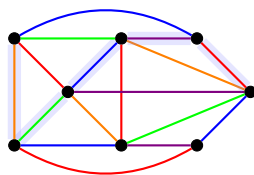
Teorema 1.2 (Shannon). $\chi'(G) \leq \frac{3}{2}\Delta(G)$.

Teorema 1.3 (Vizing). $\chi'(G) \leq \Delta(G) + \mu(G)$.

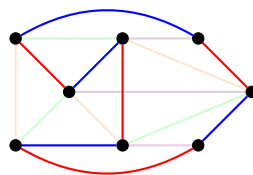
Podemos observar que cada classe de cor $E_i = \{e \in E(G) \mid c(e) = i\}$ induz um *emparelhamento* de G quando a coloração é própria, pois não é permitido haver duas arestas adjacentes com mesma cor.

Um caminho $P = (v_1, \dots, v_k)$ em um grafo aresta-colorido é uma *caminho rainbow* se $c(v_i) = c(v_j)$ para todo $i, j \in \{1, \dots, k\}$, isto é, se nenhuma cor aparece em mais de uma aresta do caminho. Na Figura 1.1a, exemplificamos uma coloração própria das arestas de um grafo e destacamos um caminho rainbow.

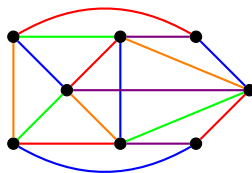
Dadas duas cores c_1 e c_2 , uma *cadeia de Kempe* com cores c_1 e c_2 em um grafo aresta colorido G é o subgrafo de G induzido pelas arestas com cores c_1 e c_2 , denotado por $G(c_1, c_2)$. Ilustramos uma cadeia de Kempe na Figura 1.1b. Note que, como $G(c_1, c_2)$ é um grafo induzido por dois emparelhamentos, então suas componentes conexas são ou ciclos ou caminhos. A utilidade nas cadeias



(a) Uma coloração própria do grafo e um caminho rainbow.



(b) Uma cadeia de Kempe com cores azul e vermelho.



(c) Operação de troca de cores da cadeia de Kempe com cores azul e vermelho.

Figura 1.1: Exemplos de coloração própria, caminho rainbow e cadeia de Kempe.

de Kempe está na observação de que podemos alternar as cores das arestas de $G(c_1, c_2)$, fazendo todas as arestas de cor c_1 se tornarem da cor c_2 e vice-versa, de forma que a coloração geral do grafo continuará sendo uma coloração própria. Exemplificamos essa operação na Figura 1.1c.

Capítulo 2

Subgrafos *rainbow*

Neste capítulo, iremos fazer uma revisão sobre os principais resultados combinatórios e algorítmicos sobre o problema de encontrar um subgrafo H em um grafo conexo aresta-colorido G tal que todas as arestas do subgrafo H tenham cores diferentes.

Denotaremos por $c : E(G) \rightarrow \mathbb{N}$ a coloração das arestas do grafo G . Dado um subgrafo $H \subseteq G$, denotaremos por $C(H) = \{c(e) \mid e \in E(H)\}$ o conjunto de cores utilizados em H . Vamos denotar a quantidade de cores diferentes utilizadas em H por $c(H) = |C(H)|$.

2.1 Árvores

Árvores geradoras muito coloridas

Antes de tratarmos sobre o problema de encontrar uma árvore geradora *rainbow*, vamos falar sobre o problema de otimização de encontrar uma árvore geradora com a *maior* quantidade de cores distintas possível. Isto é, queremos encontrar

$$\max\{c(T) \mid T \text{ é árvore geradora de } G\}.$$

Este problema foi estudado por (Broersma e Li 1997) que mostraram um algoritmo polinomial para encontrar uma árvore geradora de G com a quantidade máxima de cores. O objetivo desta subseção é apresentar este algoritmo.

Seja $F \subseteq G$ uma floresta geradora de G . Vamos dizer que F é *c-ótima* se F é *rainbow*, isto é, $c(F) = |E(F)|$, e não existe outra floresta geradora F' com $c(F') > c(F)$. Podemos pensar em F como uma *floresta geradora máxima rainbow* de G . Vamos chamar uma árvore geradora de G com a maior quantidade de cores de *árvore c-máxima*. O lema abaixo nos mostra que encontrar uma árvore *c-máxima* de G é equivalente a encontrar uma floresta *c-ótima* de G .

Lema 2.1. (Broersma e Li 1997) *T é uma árvore c -máxima de G se, e somente se, existe uma floresta c -ótima F de G tal que $C(T) = C(F)$ e T pode ser obtida adicionando arestas em F .*

Demonstração. Suponha primeiro que T é uma árvore c -máxima de G com $c(T) = k$. Observe que T possui como subgrafo uma floresta F com $C(F) = C(T)$ e $c(F) = k$. Suponha, por absurdo, que F não é c -ótima. Então, existe F' com $c(F') > k$. Porém, como G é conexo, existe T' com $c(T') \geq c(F') > c(T)$, contradizendo a c -maximalidade de T .

Agora, suponha que T é obtida adicionando arestas em uma floresta c -ótima F de G com $C(T) = C(F)$. Suponha, por absurdo, que T não é c -máxima e seja T' uma árvore geradora de G com $c(T') > c(T)$. Então T possui como subgrafo uma floresta F' tal que $c(F') = |E(F')| = c(T') > c(F)$, contradizendo a c -otimalidade de F . \square

Mostraremos a seguir que o problema de encontrar uma floresta geradora c -ótima é equivalente a um problema em matróides.

Um *matróide* é um par $M = (S, \mathcal{I})$, em que S é um conjunto finito de elementos e \mathcal{I} é uma família de subconjuntos de S tal que:

- (i) $\emptyset \in \mathcal{I}$;
- (ii) Todos os subconjunto próprios de um conjunto $I \in \mathcal{I}$ também estão em \mathcal{I} ;
- (iii) Se $U, V \in \mathcal{I}$ e $|U| = |V| + 1$, então existe um elemento $x \in U \setminus V$ tal que $V \cup \{x\} \in \mathcal{I}$.

Dados dois matróides definidos sobre o mesmo conjunto de elementos $M_1 = (S, \mathcal{I}_1)$ e $M_2 = (S, \mathcal{I}_2)$, o problema da interseção máxima busca encontrar $I \in \mathcal{I}_1 \cap \mathcal{I}_2$ com máxima cardinalidade. Existe um algoritmo polinomial para este problema (Teoria dos Matróides). Definiremos a seguir dois matróides sobre o conjunto de arestas $E(G)$ do nosso grafo de entrada que nos ajudarão a encontrar uma floresta geradora c -máxima de G .

Sejam

$$M_1 = (E(G), \mathcal{I}_1 = \{F \subseteq E(G) \mid (V(G), F) \text{ é uma floresta}\}) \text{ e}$$

$$M_2 = (E(G), \mathcal{I}_2 = \{E' \subseteq E(G) \mid c((V, E')) = |E'|\}).$$

Lema 2.2. (*ibid.*) M_1 e M_2 são matróides.

Demonstração. Verificamos que \emptyset está tanto em \mathcal{I}_1 quanto em \mathcal{I}_2 . Também podemos verificar facilmente que a propriedade de hereditariedade é válida para ambos \mathcal{I}_1 e \mathcal{I}_2 . Sejam $F_1, F_2 \subseteq E(G)$ florestas com $|F_1| = |F_2| + 1$. Então existe uma aresta $uv \in F_1$ tal que u e v estão em componentes diferentes de F_2 . Assim, $F_2 \cup \{e\}$ também é uma floresta e, portanto, pertence a \mathcal{I}_1 . Além disso, sejam $E_1, E_2 \subseteq \mathcal{I}_2$ com $|E_1| = |E_2| + 1$. Como $c((V, E_1)) = |E_1|$ e $c((V, E_2)) = |E_2|$, existe

uma aresta $e \in E_1$ cuja cor não aparece em nenhuma aresta de E_2 . Logo $E_2 \cup \{e\}$ também está em \mathcal{I}_2 . \square

Lema 2.3. (*ibid.*) F é uma floresta c -ótima de G se, e somente se, F é uma interseção máxima de M_1 e M_2 .

Demonstração. Por definição, qualquer conjunto $F \in \mathcal{I}_1 \cap \mathcal{I}_2$ é uma floresta de G com $c(F) = |E(F)|$. Logo, se F é c -ótima, então F é uma interseção de M_1 e M_2 . A inversa também é válida. \square

Com isto, basta utilizarmos o algoritmo de matróides para encontrar a interseção máxima, que é polinomial em $|E(G)|$, para encontrar uma floresta c -ótima e, em seguida, basta expandi-la adicionando arestas para encontrar uma árvore geradora c -máxima de G .

O que vamos fazer no restante desta subseção é tentar traduzir o algoritmo de matróides para uma linguagem mais da teoria dos grafos, obtendo uma caracterização para florestas c -ótimas em termos de operações em árvores.

Pergunta 1

Como podemos explorar Matróides de *Lattice Path* para obter novos resultados relacionados a caminhos *rainbow*?

Capítulo 3

Cobertura de arestas por diferentes estruturas e subgrafos

Dado um grafo G , dizemos que uma coleção de subconjuntos de arestas $\mathcal{F} = \{E_1, E_2, \dots, E_k \mid E_i \subseteq E(G)\}$ é uma *cobertura das arestas* de G se a união $\bigcup_{i=1}^k E_i$ é exatamente $E(G)$. Note que os subconjuntos não precisam ser disjuntos. No caso em que os conjuntos são dois-a-dois disjuntos, dizemos que \mathcal{F} é uma *decomposição das arestas* de G .

Quando estamos lidando com coberturas e decomposições, em geral queremos encontrar a menor família \mathcal{F} que cobre (decompõe) as arestas de um grafo de tal forma que todos os seus conjuntos obedeça a alguma restrição ou possua alguma propriedade pré-determinada. Por exemplo, podemos pedir que cada conjunto seja um caminho; ou ciclos; ou árvores; ou tenha no máximo x arestas; etc.. Neste capítulo, nós estudaremos coberturas de arestas por diferentes estruturas e subgrafos como esses que mencionamos acima.

3.1 O problema do Carteiro Chinês

No problema do Carteiro Chinês, o objetivo é encontrar o menor passeio fechado em um grafo de modo a cobrir todas as suas arestas, isto é, o passeio passa por cada aresta pelo menos uma vez. Este problema não se encaixa exatamente no conceito de cobertura de arestas que introduzimos anteriormente, porém ele é similar o suficiente e é útil para entender um pouco sobre problemas deste tipo. Esta seção foi baseada no documento (Chekuri 2010).

Primeiro, veja que quando o grafo é euleriano, então qualquer circuito euleriano no grafo é um passeio do Carteiro Chinês. Lembre-se que:

Teorema 3.1. (*Euler*) G é euleriano se, e somente se, o grau de todos os vértices de G é par.

O problema se torna interessante então quando G não é euleriano, ou seja, G possui pelo menos um vértice de grau ímpar. Ou, mais precisamente ainda, G pelo menos dois tais vértices, como afirma a proposição a seguir.

Proposição 3.1. (Folclore) *A quantidade de vértices de grau ímpar em um grafo é sempre par.*

Demonstração. Lembre-se que, pelo lema do aperto de mãos, a soma dos graus de um grafo G é igual a $2|E(G)|$, portanto par. Suponha, por absurdo, que a quantidade de vértices de grau ímpar em G é ímpar. Então, a soma dos graus de tais vértices também é ímpar. Juntando com a soma dos vértices de grau par (que também é par), temos que o somatório dos graus de G é ímpar. Absurdo. \square

Seja $x(e) \geq 1$ a quantidade de vezes que a aresta $e \in E(G)$ aparece em uma solução do Carteiro Chinês no grafo G . Note que o multigrafo M obtido a partir de G com o mesmo conjunto de vértices e com $x(e)$ cópias da aresta e é euleriano (o passeio do Carteiro Chinês induz um circuito euleriano). Além disso, veja que se $x(e) > 2$, o multigrafo M' obtido ao fazer $x'(e) = x(e) - 2$ também é euleriano e induz uma solução menor do Carteiro Chinês em G . Concluimos então que:

Proposição 3.2. *Em toda solução minimal do Carteiro Chinês em G , temos que $x(e) \in \{1, 2\}$ para toda aresta $e \in E(G)$.*

Logo, para resolver o Carteiro Chinês, nós queremos encontrar um valor $x(e) \in \{1, 2\}$ para cada aresta do grafo G de modo a minimizar $\sum_{e \in E(G)} c(e)x(e)$, onde $c(e)$ é o custo da aresta e no caso de G ser um grafo ponderado.

Seja $T \subseteq V(G)$ o conjunto de vértices de grau ímpar em G . Como vimos anteriormente, $|T|$ é par. Um T -join é um conjunto $J \subseteq E(G)$ de forma que, no grafo $(V(G), J)$, os vértices de T possuem grau ímpar e todos os demais vértices possuem grau par.

Em uma solução viável para o Carteiro Chinês, o conjunto J de arestas e tais que $x(e) > 1$ é um T -join. Para ver isso, suponha que J não é um T -join, então no multigrafo M induzido pela solução, algum vértice terá grau ímpar e, logo, o multigrafo não será euleriano, contradizendo o fato da solução ser viável. Consequentemente, se J é um T -join, conseguimos obter um multigrafo euleriano repetindo as arestas de J : como os vértices de T possuem grau ímpar, no multigrafo eles terão grau par; os demais vértices manterão a paridade. Concluimos então que, para encontrar uma solução ótima para o Carteiro Chinês, basta encontrarmos um T -join mínimo.

Para encontrar um T -join mínimo, utilizamos o seguinte algoritmo. Para cada $u, v \in T$, seja P_{uv} o caminho mínimo de u para v em G com peso total $w(P_{uv})$. Construa um grafo auxiliar H com $V(H) = T$ e, para cada $u, v \in T$, adicione a aresta e_{uv} com peso $w(P_{uv})$ em H . Encontre um emparelhamento máximo de custo mínimo M em H utilizando, por exemplo, o algoritmo Húngaro. Retornamos então o T -join $J = \{e \mid e \text{ está em um número ímpar de caminhos } P_{uv}, uv \in M\}$.



(a) Uma instância do Carteiro Chinês sem pesos e sua solução obtida (multigrafo) pelo algoritmo com custo $|E(G)| + 3$.

(b) Grafo auxiliar para a instância do Carteiro Chinês com emparelhamento máximo de custo mínimo destacado.

Figura 3.1: Aplicação do algoritmo de T -join do Carteiro Chinês.

3.2 Florestas

3.3 Caminhos

Nesta seção, vamos estudar a cobertura das arestas de grafos em que cada conjunto da cobertura forma uma caminho no grafo. O número de cobertura (não-disjunta/irrestrita) por caminhos de um grafo G é o menor número de caminhos necessários para cobrir todas as arestas de G , denotado por $\pi^*(G)$. Quando exigimos que os caminhos sejam disjuntos, o número de decomposição por caminhos de um grafo G é denotado por $\pi(G)$.

Estes parâmetros foram introduzidos e inicialmente estudados por (Harary 1970) e (Harary e Schwenk 1972). A seguinte conjectura de Gallai é uma importante conjectura ainda em aberto que tem ligação aos temas que estamos estudando.

Conjectura 3.1. (Gallai, add. ref.) Para todo grafo G , vale que $\pi(G) \leq \frac{n}{2}$.

Chung (adicionar ref.) fez uma conjectura análoga à conjectura de Gallai, porém para $\pi^*(G)$.

Conjectura 3.2. (Chung, add. ref.) Para todo grafo G , vale que $\pi^*(G) \leq \frac{n}{2}$.

Cobrindo árvores com caminhos

Para iniciar os nossos estudos, vamos verificar o número de cobertura por caminhos para a classe das árvores. Para uma árvore T , denote por $\ell(T)$ o número de folhas de T .

Teorema 3.2. (ibid.) Seja T uma árvore. Então $\pi^*(T) = \lceil \frac{\ell(T)}{2} \rceil$.

Demonstração. Primeiro, note que cada caminho em uma cobertura de T cobre no máximo 2 folhas de T , portanto temos que $\pi^*(T) \geq \lceil \frac{\ell(T)}{2} \rceil$. Vamos mostrar agora a igualdade.

Sejam $l_1, l_2, \dots, l_{\ell(T)}$ as folhas de T e denote por $P_{i,j}$ o caminho (único) entre l_i e l_j em T . Considere uma família \mathcal{P} composta pelos caminhos $P_{i, \ell(T)/2+i}$, para $i \in \{1, \ell(T)/2\}$. Suponha que existe uma aresta e em T que não foi coberta por \mathcal{P} . Observe que \mathcal{P} cobre todas as arestas incidentes a folhas de T , logo e não é incidente a uma folha. Assim, ao removermos e de T , obtemos duas componentes não-triviais T_1 e T_2 . Existem então folhas l_i, l_j, l_r, l_s tais que $P_{i,j} \subset T_1$ e $P_{r,s} \subset T_2$ e ambos estes caminhos estão em \mathcal{P} (por quê?). Remova $P_{i,j}$ e $P_{r,s}$ de \mathcal{P} e adicione $P_{i,r}$ e $P_{j,s}$. A nova família \mathcal{P}' cobre todas as mesmas arestas cobertas anteriormente, além de cobrir também a aresta e que não estava coberta. Repetimos então este procedimento no máximo $n-1$ vezes até que todas as arestas sejam cobertas por $\ell(T)/2$ caminhos. \square

Pergunta 2

Até o momento em que este documento foi escrito, pelo conhecimento do autor, sabe-se que os melhores limites obtidos são

- $\pi(G) \leq \frac{2n}{3}$.
- $\pi^*(G) \leq \frac{n}{2} + O(n^{3/4})$.

Podemos melhorar estes limites?

Comentário: verificar se realmente os resultados acima são os melhores limites obtidos até agora e adicionar referências para eles.

Na verdade, já foi provado que a conjectura de Chung (como é chamado o análogo da conjectura de Gallai para coberturas) é verdadeira. Ou seja, $\pi^*(G) \leq \lceil \frac{n}{2} \rceil$.

Comentário: adicionar resultados e perguntas ao parâmetro análogo para digrafos.

Particionando grafos em $\lceil \frac{n}{2} \rceil$ caminhos ou ciclos

Lovász (add. ref.) fez um grande avanço em direção à conjectura de Gallai. Ele provou que, se permitirmos que algumas das estruturas sejam ciclos e não apenas caminhos, conseguimos de fato não somente cobrir, mas até mesmo particionar as arestas do grafo, utilizando apenas $\lceil \frac{n}{2} \rceil$ estruturas. Algumas das consequências da demonstração de Lovász também são muito úteis para o assunto que estamos estudando.

Cobrimos os vértices de um grafo usando caminhos mínimos

Nesta seção, nós iremos estudar um pouco do trabalho de (Foucaud et al. 2023). Nele, os autores estudaram o problema de cobrir os vértices (não as arestas) de um grafo através de caminhos mínimos (induzidos).

3.4 Ciclos

Encontrando a base de ciclos de um grafo

Acreditem ou não, os ciclos de um grafo em certo sentido podem ser vistos como um espaço vetorial! Esse espaço vetorial tem bases: ciclos minimais que geram todos os outros através de uma certa operação. Nesta seção, nós vamos conhecer melhor esse espaço vetorial de ciclos e suas bases, além de um algoritmo para encontrar a base de ciclos de qualquer grafo. Esta seção vai ser baseada em (Horton 1984).

Cobertura dupla por ciclos

Uma cobertura é *dupla* se ela cobre cada elemento exatamente duas vezes. Existe uma conjectura chamada de “conjectura da cobertura dupla por ciclos”, que diz que todo grafo sem pontes admite uma cobertura por ciclos em que cada aresta aparece em exatamente dois ciclos da cobertura (add. ref.). Nesta seção, nós vamos ver alguns resultados sobre essa conjectura.

3.5 Passeios em digrafos

Mudar “trilhas” para “passeios” ao longo desta seção.

É fácil ver que, em um grafo não-direcionado e conexo, uma única trilha é suficiente para cobrir todas as arestas. Portanto, em qualquer grafo não-direcionado G , a quantidade mínima de trilha necessárias para cobrir suas arestas é igual a quantidade de componentes conexas de G . O problema de cobrir vértices e arestas por trilhas se torna mais interessante em grafos direcionados, pois as trilhas precisam respeitar a direção dos arcos.

Em um digrafo D , dizemos que um vértice v alcança (é alcançado por) um vértice u se existe um caminho de v (u) para u (v). Dado um conjunto $S \subseteq V(D)$, dizemos que S é um *conjunto de vértices incomparáveis* se, para quaisquer $v, u \in S$, temos que v não alcança u . O tamanho de um maior conjunto de vértices incomparáveis de D é denotado por $i_v(D)$. O famoso teorema de Dilworth relaciona o tamanho de uma menor cobertura de vértices por caminhos em um digrafo acíclico D com $i_v(D)$.

Teorema 3.3. (Dilworth) *Seja D um digrafo acíclico. O menor número de caminhos necessários para cobrir $V(D)$ é igual a $i_v(D)$.*

Como em um digrafo acíclico, todo caminho é uma trilha e toda trilha é um caminho, o teorema de Dilworth também nos dá o menor número de trilhas necessárias para cobrir o conjunto de vértices de um digrafo acíclico. Ntafos e Hakimi 1979 generalizaram o teorema de Dilworth para digrafos gerais.

Teorema 3.4. (*Ntafos e Hakimi 1979*) *Seja D um digrafo qualquer. O menor número de trilhas necessárias para cobrir $V(D)$ é igual a $i_v(D)$.*

Demonstração. Sejam $V_1, \dots, V_k \subseteq V(D)$ os conjunto de vértices que induzem as componentes fortemente conexas de D . Vamos provar o teorema por indução em k .

Quando $k = 1$, é fácil ver que uma única trilha consegue cobrir todos os vértices de D . Como todos os vértices da única componente fortemente conexa de D alcançam-se mutuamente, temos que $i_v(D) = 1$. Logo, para o caso base, o teorema segue.

Suponha então que o teorema vale para todo $k < n$ e seja $k = n$. Considere agora o digrafo $D' = (V', A')$ com $V' = \{v'_1, v'_2, \dots, v'_k\}$ e o arco (v'_i, v'_j) existe em A' se, e somente se, existe um vértice de V_i que alcança algum vértice de V_j em D . Note que D' é acíclico.

Seja S um conjunto de vértices incomparáveis de D' tal que $|S| = i_v(D')$. Sejam $D_u(S) = \{v'_i \mid v_i \text{ alcança } S \text{ ou } v_i \in S\}$ e $D_d(S) = \{v'_i \mid v'_i \text{ é alcançado por } S \text{ ou } v'_i \in S\}$. Veja que, em ambos os subgrafos $D[D_u(S)]$ e $D[D_d(S)]$, S também é um conjunto de vértices incomparáveis máximo. Temos dois casos.

Caso 1: $|D_u(S)| < k$ e $|D_d(S)| < k$. Neste caso, pela hipótese indutiva, temos que existe uma cobertura dos vértices por trilhas de $D'[D_u(S)]$ e de $D'[D_d(S)]$, ambas com tamanho $|S|$. Veja que, em $D'[D_u(S)]$ ($D'[D_d(S)]$), o conjunto S é um conjunto de fontes (sumidouros). Logo, podemos “colar” o fim das trilhas que cobrem $D'[D_u(S)]$ com o início das trilhas que cobrem $D'[D_d(S)]$, obtendo uma cobertura de tamanho $|S|$ para D' que induz uma cobertura de mesmo tamanho para D .

Caso 2: $|D_u(S)| = k = n$ ou $|D_d(S)| = k = n$. Neste caso, S é ou o conjunto de fontes ou o conjunto de sumidouros de D' . Em qualquer que seja o caso, sejam s uma fonte e t um sumidouro de D' . Pela hipótese indutiva, $D' - \{s, t\}$ possui uma cobertura de vértices por caminhos de tamanho $|S| - 1$. Adicionando o caminho $P_{s,t}$, obtemos uma cobertura de tamanho $|S|$ que induz uma cobertura de mesmo tamanho em D . \square

Correção: estes resultados do (ibid.) são para *passeios* em digrafos e não trilhas. Lembrar de consertar depois.

A demonstração acima não nos dá claramente um algoritmo. Resolvemos isto com um algoritmo de fluxo mínimo que descrevemos a seguir. Seja D' o DAG das componentes fortemente conexas de D . Adicione a D' dois vértices, s e t , tal que s é vizinho de entrada de todas as fontes de D' e t é vizinho de saída de todas as fontes. Para cada vértice original v de D' , subdivida-o em v' e v'' adicionando o arco (v', v'') de forma que todos os vizinhos de entrada de v serão vizinhos de entrada de v' e todos os vizinhos de saída de v serão vizinhos de saída

de v'' . Para cada arco (x, y) adicione uma restrição de fluxo $r_{x,y}$ tal que

$$r_{x,y} = \begin{cases} 1 & , \text{ se } x = v' \text{ e } y = v'' \\ 0 & \text{ caso contrário.} \end{cases}$$

Seja D_f este digrafo modificado e com as restrições de fluxo conforme descritas acima. Queremos encontrar um fluxo mínimo f tal que $f(x, y) \geq r_{x,y}$ para cada arco $(x, y) \in A(D_f)$ e que satisfaz as restrições de conservação de fluxo. O valor do fluxo f é dado por $w(f) = \sum_{(x,t) \in A(D_f)} f(x, t)$. O teorema a seguir relaciona o fluxo mínimo em D_f com $i_v(D)$.

Teorema 3.5. (*ibid.*) *O fluxo mínimo f em D_f tem valor $w(f) = i_v(D)$.*

Demonstração. É fácil ver que $w(f) \geq i_v(D)$, uma vez que cada vértice de um conjunto incomparável deve estar em um caminho diferente entre s e t . Por outro lado, seja $\{P_1, \dots, P_{i_v(D)}\}$ uma decomposição em caminhos mínima de D' . Observe que, passando uma unidade de fluxo por cada um desses caminhos, obtemos um fluxo que satisfaz as restrições. Logo, $w(f) \leq i_v(D)$, como queríamos. \square

Para resolver o problema de cobrir arestas, podemos definir um conjunto de arestas incomparáveis de maneira análoga, definindo também o parâmetro $i_e(D)$ que representa o tamanho de um maior tal conjunto em um digrafo D . A partir daí, todas as provas seguem similarmente e o algoritmo de fluxo pode ser aplicado diretamente em D adicionando, para cada arco e , uma restrição $r_e = 1$, obrigando que cada arco seja coberto pelo fluxo mínimo.

Capítulo 4

Fronteira, Contorno e Periferia

Neste capítulo, nós iremos estudar um pouco sobre tipos de vértices extremos, conceitos que surgiram nas áreas de Geometria Computacional e de Convexidade em Grafos. Mais especificamente iremos estudar sobre: vértices de fronteira, de contorno e periféricos, suas relações e propriedades em diferentes grafos. Ao longo do capítulo, vamos estabelecendo um pouco mais as ligações entre estes tipos de vértices e os demais assuntos que temos estudado.

A *distância* entre dois vértices u e v em um grafo G , denotada por $d_G(u, v)$ (ou simplesmente $d(u, v)$ quando o grafo está subentendido), é o comprimento de um menor caminho de u para v em G . A *eccentricidade* de um vértice v em G , denotada por $e_G(v)$ (ou $e(v)$), é a maior distância entre v e algum outro vértice de G . A menor eccentricidade dentre todos os vértices de G é o *raio* de G , denotado por $rad(G)$. A maior eccentricidade dentre todos os vértices de G é o *diâmetro* de G , denotado por $diam(G)$. Um vértice v é dito *central* em G se $e(v) = rad(G)$, e é dito *periférico* se $e(v) = diam(G)$. O conjunto de vértices centrais de um grafo G é chamado de *centro* de G e denotado por $Cen(G)$, enquanto que o conjunto de vértices periféricos é chamado de *periferia* de G e denotado por $Per(G)$. Um vértice v é chamado de *vértice de fronteira* de G se existe $u \in V(G)$ tal que $d(v, u) \geq d(w, u)$ para todo $w \in N_G(v)$. O conjunto de vértices de fronteira de G é denotado por $\partial(G)$.

Proposição 4.1. (Chartrand et al. 2003) *O seguinte vale para todo grafo G :*

1. $|Per(G)| \geq 2$.
2. $Per(G) \subseteq \partial(G) \subseteq V(G)$;

Demonstração. Seja P_1 um maior caminho mínimo em G . Sejam u e v as extremidades deste caminho. Então u e v são periféricos. Logo, segue que $|Per(G)| \geq 2$. Além disso, é claro que se u é um vértice periférico, u é um vértice de fronteira, pois sabemos que $e(u) = diam(G)$. \square

Pergunta 3

Em quais classes de grafos sempre existe um vértice $v \in \partial(G)$ ($\text{Ct}(G), \text{Per}(G)$) tal que $\partial(G-v) \leq \partial(G)$ ($\text{Ct}(G-v) \leq \text{Ct}(G), \text{Per}(G-v) \leq \text{Per}(G)$)?

Grafos com tal propriedade podem ser úteis para aplicar indução, então seria interessante caracterizar estas classes.

4.1 Grafos cordais e vértices extremos**Pergunta 4**

Como podemos caracterizar cada tipo de vértice $\partial(G)$, $\text{Per}(G)$, $\text{Ecc}(G)$ e $\text{Ct}(G)$ em termos da árvore de cliques de G , sendo G cordal?

Como essas caracterizações podem nos ajudar a encontrar coberturas para grafos cordais?

Capítulo 5

Tópicos em Matróides e Coberturas

Como vimos anteriormente, encontrar uma árvore geradora mais colorida de um grafo G é um problema polinomial. E uma das razões para isso é porque os subconjuntos $F \subseteq E(G)$ tais que F é acíclico em G definem uma matróide, assim como os conjuntos $R \subseteq E(G)$ tais que R é *rainbow* em G . E, como determinar o maior conjunto independente na interseção de duas matróides pode ser feito em tempo polinomial, também conseguimos encontrar a maior floresta *rainbow* de G em tempo polinomial, o que por sua vez nos permite facilmente encontrar uma árvore geradora maximalmente colorida em G . Este capítulo busca explorar ideias matroidais para resolver problemas de cobertura e problemas que envolvem estruturas *rainbow*, principalmente caminhos *rainbow*.

5.1 Algumas famílias (potencialmente) úteis de matróides

Nesta seção, busco listar algumas das famílias de matróides que, a meu ver (ou de acordo com minha intuição), tem potencial para serem utilizadas nos tipos de problemas que temos lidado.

Matróides de *Lattice Path*

Iremos utilizar a definição de (J. Bonin, Mier e Noy 2003). Uma outra boa e útil referência sobre este tipo de matróide é (Pol 2025). Algumas outras propriedades úteis sobre este tipo de matróide podem ser encontradas em (J. E. Bonin e de Mier 2006).

Sejam $P = p_1 p_2 \dots p_{m+r}$ e $Q = q_1 q_2 \dots q_{m+r}$ dois caminhos em uma grade tal que P nunca fica abaixo de Q . Seja $M[P, Q]$ a matróide cujo conjunto base é $[m+r]$ e seus conjuntos independentes são subconjuntos $X \subseteq [m+r]$. Cada

conjunto independente X representa um caminho $P(X)$ na região da grade entre Q e P tal que $P(X)$ é a string $s_1 s_2 \dots s_{m+r}$ em que:

$$s_i = \begin{cases} C \text{ (cima)}, & \text{se } i \in X \\ D \text{ (direita)}, & \text{se } i \notin X. \end{cases}$$

Por que pode ser útil? Podemos traduzir um grafo de intervalo para matróides de *lattice path* (*Vamos provar isso logo?*). Com isto, podemos utilizar resultados conhecidos em matróides e também tentar adaptar ou traduzir resultados matroidais específicos para a linguagem da teoria dos grafos. Um outro caminho que pode ser explorado é se existe algum tipo de correspondência entre grafos bipartidos completos e matróides de *lattice path*. É possível que exista matróides de *lattice path* ainda mais específicas para grafos completos, como as matróides Catalãs.

Matróides Gamóides

Seja $D = (V, A)$ um grafo direcionado e $X, Y \subseteq V(D)$. Dizemos que X é *conectado a* Y em D se existem $|Y|$ caminhos vértice-disjuntos de X para Y . (Os caminhos são inteiramente vértice-disjuntos, não apenas *internamente* vértice-disjuntos.)

Seja $D = (V, A)$ e $S, T \subseteq V(D)$. Uma matróide gamóide é uma matróide definida sobre o conjunto base T cujos conjuntos independentes são os subconjuntos $X \subseteq T$ tais que S é conectada a X em G .

Por que pode ser útil? Talvez possamos aplicar gamóides para encontrar sistemas de caminhos disjuntos que cobrem grafos ou digrafos. É interessante também procurar alguma relação com problemas de linkage e subdivisão de digrafos. Talvez uma boa referência para saber mais sobre isso seja (Albrecht 2018) e o livro (Oxley 2011).

Matróides Gráficas

Estas são as matróides mais simples que conhecemos relacionadas à grafos: a matróide definida sobre o conjunto de arestas de um grafo cujos conjuntos independentes são subconjuntos de arestas acíclicos.

5.2 Cobertura de matróides

Existe surpreendentemente poucos trabalhos sobre a quantidade mínima de caminhos para cobrir as arestas de um grafo ou, como definimos previamente, o número irrestrito de caminhos de um grafo. Se de fato for possível traduzir um grafo de intervalos (próprio...) para uma matróide de *lattice path*, podemos utilizar os resultados de coloração de matróides, incluindo o Algoritmo de Edmonds para coloração, para encontrar a menor quantidade de caminhos para cobrir as arestas de grafos de tal classe. É isto que tentaremos fazer nesta seção.

A correspondência entre caminhos crescentes em um grafo de intervalos próprios e uma matróide de *lattice path*

Para mostrar que podemos representar caminhos crescentes em um grafo de intervalos próprios como uma matróide de *lattice path*, vamos precisar da seguinte caracterização de matróides de *lattice path*.

Lema 5.1 (J. Bonin, Mier e Noy 2003; Oxley 2011). *Uma matróide de transversal é uma matróide de lattice path se, e somente se, existe uma ordenação do seu conjunto base e uma sequência de intervalos (N_1, N_2, \dots, N_r) desta ordenação com extremidades $N_i = [l_i, g_i]$ tal que $l_1 \leq l_2 \leq \dots \leq l_r$ e $g_1 \leq g_2 \leq \dots \leq g_r$.*

Demonstração. Adicionar prova aqui... □

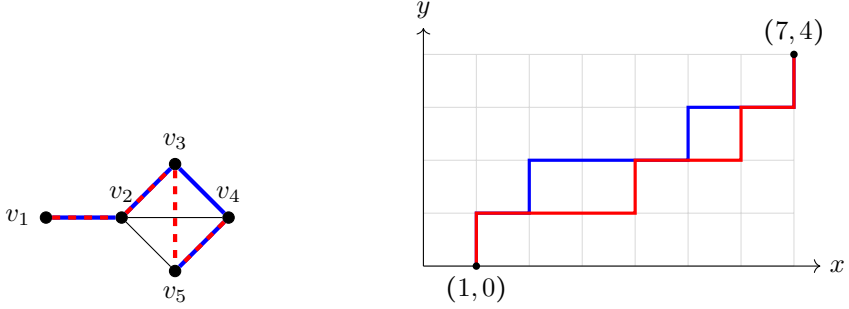
Proposição 5.1 (*). *Seja G um grafo conexo de intervalos próprios definido pela família de intervalos \mathcal{I} . Existe uma matróide de lattice path M_G e uma função sobrejetiva que mapeia cada conjunto independente de M_G em um caminho crescente em G .*

Demonstração. Como G é um grafo de intervalos próprios, podemos ordenar os intervalos de \mathcal{I} , obtendo uma sequência $(I_i = [s_i, t_i])_{i \in [n]}$ de intervalos tal que $s_1 < s_2 < \dots < s_n$ e $t_1 < t_2 < \dots < t_n$. Chame de v_i o vértice associado ao intervalo I_i nesta sequência. Além disso, seja $N_i = \{v_i v_j \mid v_j \in N_G(v_i), i < j\}$. Defina M_G como a matróide de transversal cujo conjunto base é $E(G)$ e apresentação $(N_i)_{i \in [n-1]}$ (podemos escrever $i \in [n-1]$ e não $i \in [n]$ porque N_n é vazio). Considere então uma ordenação lexicográfica de $E(G)$ em que $v_1 < v_2 < \dots < v_n$, cujo resultado é uma sequência $S = (e_1, e_2, \dots, e_m)$. Veja que, como G é um grafo de intervalos próprio, cada N_i é um intervalo de S . Além disso, pela forma como definimos cada N_i , suas extremidades também satisfazem o Lema 5.1. Portanto, concluímos que M_G é, de fato, uma matróide de *lattice path*.

Agora, seja $P = (v_{u_1}, v_{u_2}, v_{u_3}, \dots, v_{u_k})$ um caminho crescente em G , ou seja, $u_1 < u_2 < u_3 < \dots < u_k$. Então, o conjunto de arestas $\{u_1 u_2, u_2 u_3, \dots, u_{k-1} u_k\}$ é um transversal parcial de $(N_i)_{i \in [n-1]}$ e, portanto, é um conjunto independente de M_G . Conversamente, seja P um transversal de $(N_i)_{i \in [n-1]}$. Se P é um transversal perfeito, então P corresponde a uma sequência de $n-1$ arestas $(e_1, e_2, \dots, e_{n-1})$ tal que $e_i = v_i v_j$ com $i < j$ e $v_j \in N_G(v_i)$, portanto P corresponde a um caminho crescente Hamiltoniano em G . Caso P seja um transversal parcial, então P é um “esqueleto” de um caminho Hamiltoniano de G , ou seja, existe um caminho Hamiltoniano P' em G que usa exatamente as arestas de P . Concluímos então que o Teorema segue. □

A seguir, damos um exemplo da transformação de um grafo de intervalos próprio para uma matróide de *lattice path*. Seja a família de intervalos $\mathcal{I} = \{[1, 3], [2, 6], [4, 7], [5, 8], [6, 9]\}$ definindo o grafo de intervalos próprios G representado na Figura 5.1a. Além disso, seja a ordenação lexicográfica de $E(G)$

dada por $(e_1 = v_1v_2, e_2 = v_2v_3, e_3 = v_2v_4, e_4 = v_2v_5, e_5 = v_3v_4, e_6 = v_3v_5, e_7 = v_4v_5)$. A representação da matróide M_G são então os intervalos $N_1 = [1, 1]$, $N_2 = [2, 4]$, $N_3 = [5, 6]$, e $N_4 = [7, 7]$. Ilustramos M_G na Figura 5.1b.



(a) Grafo pipa de intervalos próprios G .

(b) Uma representação da matróide de *lattice paths* definida pelos caminhos crescentes de G .

Figura 5.1: Ilustração de um grafo de intervalos próprios G , a matróide de *lattice paths* como descrita no Teorema 5.1 e alguns caminhos crescentes representados nela.

O Algoritmo de Edmonds para coloração de matróides

Cobrimo as arestas de um grafo de intervalos próprio por caminhos

5.3 Interseção de matróides

Em livros sobre Teoria das Matróides, comumente existe um capítulo ou seção inteiramente dedicado à interseção de matróides. Dentre os resultados relevantes sobre este tema, está um outro algoritmo de Edmonds para encontrar o conjunto independente de maior cardinalidade na interseção de duas matróides.

Nesta seção, teremos dois objetivos principais. O primeiro é resolver o problema do k -CAMINHO RAINBOW ou, se desejar, CAMINHO RAINBOW MAIS LONGO. Este é um problema NP-completo, uma vez que generaliza o problema CAMINHO HAMILTONIANO. Nós tentaremos resolver o problema para os grafos de intervalo próprio traduzindo as ideias do algoritmo de Edmonds para encontrar o maior conjunto independente na interseção de duas matróides em um algoritmo completamente na linguagem da teoria dos grafos.

O segundo objetivo é provavelmente mais ambicioso. Um resultado de (Aharoni e Berger 2006) afirma que, para cobrir a interseção de duas matróides M_1 e M_2 , precisamos no máximo de duas vezes mais conjuntos independentes que a maior quantidade de conjuntos independentes necessários para cobrir M_1 ou M_2 .

Explicitamente:

$$\chi(M_1 \cap M_2) \leq 2 \cdot \max(\chi(M_1), \chi(M_2)).$$

Também tentaremos adaptar as ideias de (ibid.) para uma linguagem de teoria dos grafos durante o nosso percurso para chegar a este objetivo.

O problema do k -Caminho Rainbow

Cobrimo um grafo de intervalos próprio colorido em arestas por caminhos rainbow

Bibliografia

- Aharoni, Ron e Eli Berger (2006). “The Intersection of a Matroid and a Simplicial Complex”. Em: *Transactions of the American Mathematical Society* 358.11, pp. 4895–4917. ISSN: 00029947. URL: <http://www.jstor.org/stable/3845406> (acesso em 02/02/2026).
- Albrecht, Immanuel (2018). “Contributions to the Problems of Recognizing and Coloring Gammoids”. Tese de dout. FernUniversität in Hagen.
- Bonin, Joseph, Anna de Mier e Marc Noy (out. de 2003). “Lattice path matroids: enumerative aspects and Tutte polynomials”. Em: *J. Comb. Theory Ser. A* 104.1, pp. 63–94. ISSN: 0097-3165. DOI: [10.1016/S0097-3165\(03\)00122-5](https://doi.org/10.1016/S0097-3165(03)00122-5). URL: [https://doi.org/10.1016/S0097-3165\(03\)00122-5](https://doi.org/10.1016/S0097-3165(03)00122-5).
- Bonin, Joseph E. e Anna de Mier (2006). “Lattice path matroids: Structural properties”. Em: *European Journal of Combinatorics* 27.5, pp. 701–738. ISSN: 0195-6698. DOI: <https://doi.org/10.1016/j.ejc.2005.01.008>. URL: <https://www.sciencedirect.com/science/article/pii/S0195669805000296>.
- Broersma, Hajo e Xueliang Li (1997). “Spanning trees with many or few colors in edge-colored graphs”. eng. Em: *Discussiones Mathematicae Graph Theory* 17.2, pp. 259–269. URL: <http://eudml.org/doc/270702>.
- Chartrand, Gary et al. (2003). “Boundary vertices in graphs”. Em: *Discrete Mathematics* 263.1, pp. 25–34. ISSN: 0012-365X. DOI: [https://doi.org/10.1016/S0012-365X\(02\)00567-8](https://doi.org/10.1016/S0012-365X(02)00567-8). URL: <https://www.sciencedirect.com/science/article/pii/S0012365X02005678>.
- Chekuri, Chandra (mar. de 2010). *T-joins and applications*. <https://courses.grainger.illinois.edu/cs598csc/sp2010/Lectures/Lecture13.pdf>. Acesso em 16 de janeiro de 2026.
- Foucaud, Florent et al. (2023). *Covering a graph using shortest paths*. https://perso.limos.fr/ffoucaud/Talks/Slides/foucaud_IsometricPathCover_01.2023.pdf. Acesso em 03 de fevereiro de 2026.
- Harary, Frank (1970). “Covering and Packing in Graphs, I.” Em: *Annals of the New York Academy of Sciences* 175. URL: <https://api.semanticscholar.org/CorpusID:84767097>.
- Harary, Frank e Allen J. Schwenk (1972). “Evolution of the Path Number of a graph: Covering and Packing in Graphs”. Em: *Graph Theory and Computing*. Ed. por RONALD C. READ. Academic Press, pp. 39–45. ISBN: 978-

- 1-4832-3187-7. DOI: <https://doi.org/10.1016/B978-1-4832-3187-7.50009-X>. URL: <https://www.sciencedirect.com/science/article/pii/B978148323187750009X>.
- Horton, Joseph D. (ago. de 1984). *A Polynomial Time Algorithm to Find the Shortest Cycle Basis of a Graph*. <https://unbscholar.dspace.lib.unb.ca/server/api/core/bitstreams/ac1f8c27-83eb-409b-80b2-93057e9dee19/content>. Acesso em 03 de fevereiro de 2026.
- Ntafos, S.C. e S.L. Hakimi (1979). “On Path Cover Problems in Digraphs and Applications to Program Testing”. Em: *IEEE Transactions on Software Engineering* SE-5.5, pp. 520–529. DOI: [10.1109/TSE.1979.234213](https://doi.org/10.1109/TSE.1979.234213).
- Oxley, James (2011). *Matroid Theory, Second Edition*. Oxford Mathematics.
- Pol, Jon van der (out. de 2025). *Lattice path matroids*. <https://matroidunion.org/?p=5892>. Acesso em 02 de fevereiro de 2026.