



O POSTO DE UMA CONVEXIDADE DE GRAFOS

Igor da Fonseca Ramos

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia de Sistemas e Computação, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia de Sistemas e Computação.

Orientadores: Jayme Luiz Szwarcfiter
Vinícius Fernandes dos Santos

Rio de Janeiro
Abril de 2014

O POSTO DE UMA CONVEXIDADE DE GRAFOS

Igor da Fonseca Ramos

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Examinada por:

Prof. Jayme Luiz Szwarcfiter, Ph.D.

Prof. Vinícius Fernandes dos Santos, D.Sc.

Prof. Valmir Carneiro Barbosa, Ph.D.

Prof. Danilo Artigas da Rocha, D.Sc.

Prof. Luérbio Faria, D.Sc.

RIO DE JANEIRO, RJ – BRASIL

ABRIL DE 2014

Ramos, Igor da Fonseca

O posto de uma convexidade de grafos/Igor da Fonseca
Ramos. – Rio de Janeiro: UFRJ/COPPE, 2014.

XI, 59 p.: il.; 29, 7cm.

Orientadores: Jayme Luiz Szwarcfiter

Vinícius Fernandes dos Santos

Dissertação (mestrado) – UFRJ/COPPE/Programa de
Engenharia de Sistemas e Computação, 2014.

Referências Bibliográficas: p. 56 – 59.

1. Posto. 2. Convexidade de Grafos. 3.
Teoria de Grafos. I. Szwarcfiter, Jayme Luiz *et al.*
II. Universidade Federal do Rio de Janeiro, COPPE,
Programa de Engenharia de Sistemas e Computação. III.
Título.

*À Carolina, cuja contribuição foi
além do que ela mesma imagina.*

Agradecimentos

“I don’t know half of you half as well as I should like; and I like less than half of you half as well as you deserve.”

J.R.R. Tolkien, *The Fellowship of the Ring*

Muitas pessoas contribuíram de formas diferentes para que esse trabalho pudesse ser feito e sou grato a todas elas. Apesar disso, algumas delas se destacaram de alguma maneira e merecem um agradecimento especial.

Em primeiro lugar, agradeço aos meus orientadores, Jayme e Vinícius, pela paciência que tiveram comigo ao longo do meu mestrado e pela ajuda sempre muito eficiente.

Também gostaria de demonstrar a minha gratidão aos amigos do Laboratório de Algoritmos e Combinatória por tornarem as horas que se passaram lá muito mais agradáveis e por aturarem (ou participarem) das nossas longas conversas sobre política, religião, o sexo dos anjos e, por vezes, até mesmo sobre grafos.

Agradeço, ainda, aos professores que contribuíram para a minha educação desde o colégio até o mestrado. Quase tudo que aprendi teve um papel importante na minha formação como pessoa, na minha vida acadêmica ou em ambas e sou muito grato por isso.

Os meus pais também merecem um agradecimento pelo papel que tiveram e pelo investimento que fizeram em minha formação. Ao meu irmão, Ruan, companheiro das noites que passei acordado, também agradeço pelos momentos em que eu precisava de uma pausa e ia conversar sobre a vida, o universo e tudo mais em seu quarto, atrapalhando seus sagrados jogos.

Finalmente, o mais especial de todos os agradecimentos à Carolina, pela compreensão, pelo apoio, pelo incentivo, pelas broncas e por ler partes de textos sobre assuntos distantes da sua realidade só porque pedi.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

O POSTO DE UMA CONVEXIDADE DE GRAFOS

Igor da Fonseca Ramos

Abril/2014

Orientadores: Jayme Luiz Szwarcfiter

Vinícius Fernandes dos Santos

Programa: Engenharia de Sistemas e Computação

Considerando-se as convexidades de grafos baseadas em caminhos mais estudadas — geodética, monofônica e P_3 — a envoltória convexa de um conjunto de vértices S é o menor conjunto convexo do grafo que contém S , representada por $H(S)$. Um conjunto de vértices S é dito convexamente independente se $v \notin H(S \setminus \{v\})$ para todo $v \in S$. O posto $rk(G)$ de um grafo é a cardinalidade de um conjunto convexamente independente máximo.

Esse trabalho apresenta um estudo sobre a complexidade de tempo do cálculo de $rk(G)$. Prova-se a \mathcal{NP} -completude do problema para grafos *split* e bipartidos na convexidade P_3 , bem como para grafos sem clique separadora na convexidade monofônica. Já no caso de árvores nas convexidades geodética, monofônica e P_3 , bem como de grafos *threshold* e de intervalo biconexos na convexidade P_3 , são apresentados algoritmos polinomiais. Finalmente, prova-se, ainda, um limite superior justo para $rk(G)$ na convexidade P_3 que, por sua vez, torna possível provar de forma mais simples o limite justo para o número de Radon provado por Henning, Rautenbach e Schäfer.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

THE RANK OF A GRAPH CONVEXITY

Igor da Fonseca Ramos

April/2014

Advisors: Jayme Luiz Szwarcfiter

Vinícius Fernandes dos Santos

Department: Systems Engineering and Computer Science

We consider the most common graph convexities based on paths of vertices — geodetic, monophonic and P_3 . The convex hull of a set of vertices S is the smallest convex set that contains S and is denoted by $H(S)$. A set of vertices S is said to be convexly independent if $v \notin H(S \setminus \{v\})$ for all $v \in S$. The rank $rk(G)$ of a graph is the cardinality of a maximum convexly independent set.

This text presents a study about the time complexity of determining $rk(G)$. We prove the \mathcal{NP} -completeness of the problem for split and bipartite graphs under the P_3 -convexity, as well as for graphs with no clique separator in the context of the monophonic convexity. On the other hand, for trees the problem can be solved in polynomial time under the three convexities considered and we also show that the rank of a threshold or biconnected interval graph can be determined in polynomial time under the P_3 -convexity. Finally, we prove a tight upper bound for $rk(G)$ in the P_3 -convexity that, in turn, provides a simpler proof to the already known tight upper bound for the Radon number given by Henning, Rautenbach and Schäfer.

Sumário

Lista de Figuras	x
1 Introdução	1
1.1 Teoria de Conjuntos	2
1.2 Teoria de Grafos	4
1.2.1 Percorrendo um grafo	7
1.2.2 Classes de grafos	8
1.3 Complexidade de algoritmos	14
1.3.1 Complexidade de tempo	15
1.3.2 Classes de complexidade	16
2 Apresentação do problema	18
2.1 Convexidade	18
2.1.1 Convexidades de caminhos relevantes	20
2.1.2 Envoltória convexa	22
2.1.3 Parâmetros de uma convexidade de grafos	24
2.1.4 Posto	27
2.2 Formalização do problema	29
2.3 Problemas relacionados	30
2.3.1 Empacotamento de conjuntos	30
2.3.2 Clique	31
2.3.3 Número de empacotamento aberto	31
3 Resultados	33
3.1 \mathcal{NP} -completude	33
3.1.1 Grafos <i>split</i> na convexidade P_3	33
3.1.2 Grafos bipartidos na convexidade P_3	37
3.1.3 Átomos na convexidade monofônica	39
3.2 Algoritmos polinomiais	41
3.2.1 Grafos <i>threshold</i> na convexidade P_3	41
3.2.2 Árvores na convexidade P_3	42

3.2.3	Árvores nas convexidades geodética e monofônica	48
3.2.4	Grafos de intervalo biconexos na convexidade P_3	49
3.3	Limite superior	52
4	Conclusões	54
4.1	Trabalhos futuros	55
	Referências Bibliográficas	56

Lista de Figuras

1.1	Representação gráfica de G .	5
1.2	O grafo \overline{G} .	6
1.3	Subgrafos de G .	6
1.4	Alguns grafos completos.	9
1.5	Exemplo de grafo <i>split</i> e duas possíveis partições <i>split</i> .	10
1.6	Exemplo de grafo <i>threshold</i> .	10
1.7	Um grafo bipartido com três bipartições distintas de seus vértices.	11
1.8	Um grafo bipartido completo, chamado de $K_{3,5}$.	11
1.9	Um exemplo de árvore.	12
1.10	Alguns exemplos de caminhos.	12
1.11	Alguns exemplos de ciclos.	13
1.12	Uma corda em destaque em um ciclo de cinco vértices.	13
1.13	Um modelo de intervalos.	14
1.14	Um exemplo de grafo de intervalo.	14
2.1	Um polígono pode ser côncavo ou convexo.	19
2.2	Exemplo de conjuntos convexo e não convexo.	20
2.3	Exemplo na convexidade monofônica.	21
2.4	Exemplo na convexidade P_3 .	21
2.5	Envoltória convexa no contexto da geometria euclideana.	22
2.6	Exemplo de envoltória convexa.	22
2.7	Algoritmo iterativo para a envoltória convexa.	23
2.8	Conjunto de envoltória mínimo na convexidade P_3 .	24
2.9	Exemplo do número de Radon.	25
2.10	Exemplo do número de Carathéodory.	26
2.11	Exemplo de conjunto convexamente independente.	28
2.12	Exemplo de conjunto convexamente dependente.	29
2.13	Relação dos vértices de grau um com o posto.	30
2.14	Duas das cliques máximas do grafo.	31
2.15	Um empacotamento aberto do grafo.	32

3.1	Exemplo da redução do Teorema 3.4. Dada a família de conjuntos $\mathcal{S} = \{S_1 = \{1, 6\}, S_2 = \{1, 2\}, S_3 = \{2, 3\}, S_4 = \{3, 4\}, S_5 = \{4, 5\}, S_6 = \{5, 6\}\}$, obtém-se o grafo da figura.	37
3.2	Exemplo da redução do Teorema 3.6. A partir do grafo <i>split</i> da figura 3.1, encontra-se o grafo acima.	39
3.3	Considere que as linhas pontilhadas representam caminhos com um número arbitrário de vértices e que são representados apenas os vértices necessários para que a figura não perca generalidade.	49

Capítulo 1

Introdução

“It’s a dangerous business, Frodo, going out your door. You step onto the road, and if you don’t keep your feet, there’s no knowing where you might be swept off to.”

J.R.R. Tolkien, *The Lord of the Rings*

Em diversos campos da ciência se estuda a relação entre pares de objetos ou indivíduos. Os grafos são estruturas matemáticas utilizadas com frequência para modelar esses vínculos. Ao estudar propriedades e parâmetros dos grafos, podemos extrair mais informações sobre as relações representadas por eles.

Desde 1736, quando Leonhard Euler resolveu o problema das Sete Pontes de Königsberg — que foi a pedra fundamental da teoria de grafos [20] —, essas estruturas foram aplicadas a problemas de muitas áreas do conhecimento. Há exemplos de seu uso na química, na física, na sociologia, na biologia, na linguística, no marketing entre outras [1, 10, 27, 36]. Atualmente, na era da internet e das redes sociais, os grafos são muito importantes para descrever a topologia das redes e representar as relações entre pessoas. Nesse contexto, os grafos são muito úteis na solução de problemas como o roteamento de pacotes [33], sendo aplicados também à computação distribuída [5, 30, 32, 34]. Podem, ainda, ajudar um usuário de uma rede social a identificar conhecidos com amigos em comum ou a modelar a propagação de ideias e opiniões nessas redes [17, 21, 29, 35].

Muitos campos surgiram dentro da teoria de grafos. Problemas relacionados a distância, coloração de vértices [31] e arestas [25], fluxo em redes [18, 26], e a vários outros assuntos já foram muito estudados e aplicados em diferentes contextos. Um conceito de aplicação relativamente recente aos grafos é a noção de convexidade, originada na geometria euclidiana. Por meio dos parâmetros de uma convexidade

de grafos, podemos estudar problemas como a disseminação de uma doença, de informação ou de uma campanha publicitária. Dentre os parâmetros mais conhecidos, podemos destacar o número de envoltória, o número de Radon e o número de Carathéodory. Todos esses conceitos serão explorados com mais detalhes posteriormente nesse trabalho.

Um parâmetro das convexidades de grafos que ainda não foi muito estudado sob o aspecto da complexidade de algoritmos é o posto. Seu nome vem da álgebra linear, pois trata-se de uma espécie de “coeficiente de independência” dos vértices de um grafo em uma determinada convexidade, algo semelhante ao posto de uma matriz. O objetivo deste trabalho é, portanto, estudar aspectos de complexidade do cálculo do posto para diversas classes de grafos. Provamos que o problema de determinar o posto de um grafo na convexidade P_3 é NP-difícil, mesmo para grafos *split* e grafos bipartidos. Apresentamos, ainda, algoritmos polinomiais para o cálculo desse parâmetro em grafos biconexos de intervalo, grafos *threshold* e árvores.

Esta dissertação está organizada em 4 capítulos. A seguir são apresentados conceitos mais básicos da teoria de conjuntos, da teoria de grafos e da complexidade de algoritmos. Muitos deles são bastante conhecidos para aqueles que já possuem alguma familiaridade com essas áreas, porém são reapresentados com a finalidade de facilitar a leitura por um público mais amplo e garantir uma maior coerência de notação e dos conceitos. A seguir, no Capítulo 2, é feita uma introdução à convexidade de grafos a fim de se apresentar o problema a ser explorado posteriormente, no Capítulo 3, em que são apresentados os resultados. Por fim, o Capítulo 4 contém as conclusões sobre o conteúdo apresentado.

1.1 Teoria de Conjuntos

“If you can’t explain it simply, you don’t understand it well enough.”

Albert Einstein

Nesta seção são apresentados conceitos da teoria de conjuntos com a finalidade de estabelecer a notação a ser usada ao longo do texto e facilitar a leitura por parte daqueles que não estão completamente familiarizados com o assunto. A teoria de conjuntos é fundamental neste trabalho, visto que é a base de grande parte da teoria de grafos e dos conceitos de convexidade [19].

Inicialmente, é preciso definir a relação binária entre um **elemento** e um **conjunto**. Dizemos que um objeto e **pertence** ou **não pertence** a um conjunto A . Representamos essas relações por $e \in A$ e $e \notin A$ e dizemos, no primeiro caso, que e é um elemento de A . Podemos relacionar também dois conjuntos. Se $A = \{1, 2\}$

e $B = \{1, 2, 3\}$ são tais que todo elemento de A pertence a B , dizemos que A está **contido** em B . Naturalmente, podemos afirmar que B **contém** A . Denotamos essas relações por $A \subset B$ e $B \supset A$, respectivamente.

Agora considere os conjuntos $A = \{1, 2\}$, $B = \{1, 2\}$ e $C = \{1, 2, 3\}$. Nesse caso, A contém B e B contém A e portanto, podemos afirmar que A é **igual** a B , o que denotamos por $A = B$. Já no caso de A e C , A está contido em C , mas C não está contido em A , ou seja, $A \subset C$ e $C \not\subset A$. Diz-se, então, que A é diferente de C , o que é representado por $A \neq C$. Uma vez que existe ao menos um elemento em C que não pertence a A e $A \subset C$, dizemos que C **contém propriamente** o conjunto A . Como, em alguns casos, é importante diferenciar se a inclusão de um conjunto em outro pode ou não ser própria, ao longo dessa dissertação usaremos uma notação que torna essas relações sempre explícitas. O conjunto A está contido ou é igual ao conjunto B e ao conjunto C , o que será representado por $A \subseteq B$ e $A \subseteq C$. Quando a inclusão de um conjunto em outro for obrigatoriamente própria, isso estará explícito como em $A \subsetneq C$.

Ainda considerando as relações de pertinência e inclusão, é importante ressaltar a diferença entre um conjunto estar contido em outro e pertencer a outro. A relação \in é exclusiva para um elemento e um conjunto. É possível relacionar dois conjuntos dessa forma, mas isso significa que um é elemento do outro. Um exemplo pode ser dado por $A = \{1, 2\}$, $B = \{1, 2, 3\}$ e $C = \{\{1, 2\}, 3\}$. O conjunto A está contido no conjunto B , mas é um elemento de C e, portanto, deve-se representar essas relações como $A \subseteq B$ e $A \in C$.

Caso um conjunto não possua nenhum elemento, ele é chamado de **conjunto vazio** e representado pelo símbolo \emptyset . Uma vez que não tem nenhum elemento, o conjunto vazio está contido em qualquer outro, ou seja, $\emptyset \subseteq A$, sendo A um conjunto qualquer, até mesmo o próprio conjunto vazio. É importante ressaltar que isso não é o mesmo que dizer que todo conjunto possui o conjunto vazio como elemento. Outro tipo de conjunto relevante é aquele que possui exatamente um elemento. Esses conjuntos são chamados de **unitários**. Ademais, o número de elementos de um conjunto é a sua **cardinalidade**, ou seja, \emptyset é um conjunto de cardinalidade zero e o conjunto unitário tem cardinalidade um. Representamos a cardinalidade do conjunto S por $|S|$.

Podemos também relacionar dois conjuntos para descrever outros. As operações de **união**, **interseção** e **diferença** são as três principais formas de fazer isso. Considere os conjuntos $A = \{1, 2\}$ e $B = \{2, 3\}$. Se dizemos que o conjunto C é o resultado da união entre os conjuntos A e B , o que denotamos por $C = A \cup B$, estamos afirmando que C contém todos os elementos que pertencem a ao menos um dos conjuntos A e B sem que nenhum elemento que não esteja em um destes faça parte de C . Dessa forma, $C = A \cup B = \{1, 2, 3\}$. Se, por outro lado, D for

o resultado da interseção entre A e B , o que representamos como $D = A \cap B$, isso significa que D contém todos os elementos que estão presentes *simultaneamente* em A e em B , isto é, $D = A \cap B = \{2\}$. Tanto a união quanto a interseção são operações comutativas e isso as distingue da diferença. Esta descreve um conjunto que contém todos os elementos que estão no primeiro conjunto e, simultaneamente, não estão no segundo. Considere os conjuntos E e F , em que o primeiro é a diferença entre A e B e o segundo a diferença entre B e A , o que denotamos, respectivamente, por $E = A \setminus B$ e $F = B \setminus A$. Isso significa que $E = A \setminus B = \{1\}$ e que $F = B \setminus A = \{3\}$. Por conveniência, se quisermos excluir exatamente um elemento e de um conjunto G , escreveremos $G - e$ em vez de $G \setminus \{e\}$.

Se dois conjuntos A e B são tais que $A \cap B = \emptyset$, dizemos que A e B são **disjuntos**. Outra forma de descrever tal relação é escrever $A \parallel B$. Ambas as representações — a interseção resultando no conjunto vazio e as linhas paralelas — são equivalentes e são usadas de forma indistinta ao longo do texto. Se $C = A \cup B$, sendo $A \parallel B$, dizemos que A e B formam uma **partição** de C . Um conjunto pode ser **particionado** em qualquer número de subconjuntos, mas, ao caso em que são formados exatamente dois subconjuntos, dá-se o nome de **bipartição**.

É preciso, ainda, descrever o que é uma **família de conjuntos**. Dado um **conjunto base** S , podemos definir uma família de subconjuntos de S , a qual chamaremos de $\mathcal{S} = \{S_1, S_2, \dots, S_n\}$, em que $S_i \subseteq S$ para $1 \leq i \leq n$. Tomemos como exemplo o conjunto $A = \{1, 2, 3, 4, 5\}$ e uma família de subconjuntos de A , $\mathcal{A} = \{\emptyset, \{1\}, \{1, 3\}, \{3, 5\}, \{2, 4, 5\}\}$. Temos, portanto, o conjunto base A , de modo que, para todo $A_i \in \mathcal{A}$, este é um subconjunto de A .

Por fim, é preciso explorar a diferença entre máximo e maximal e entre mínimo e minimal em relação a uma propriedade. Um conjunto S é **máximo** com relação a uma propriedade se não há nenhum outro conjunto de cardinalidade maior que a satisfaça e **mínimo** se não há outro de cardinalidade menor. Por outro lado, S é dito **maximal** se não há nenhum outro $S' \supsetneq S$ que também apresente tal propriedade. De maneira similar, um conjunto S é **minimal** com relação a uma propriedade se não há $S' \subsetneq S$ que também a satisfaça.

1.2 Teoria de Grafos

“Que progresso estamos fazendo.
Na Idade Média me teriam
queimado. Agora se contentam
com queimar meus livros.”

Sigmund Freud

Assim como a Seção 1.1 introduziu conceitos básicos da teoria de conjuntos, o objetivo do texto a seguir é apresentar o conteúdo de teoria de grafos que serve de alicerce para todo o restante do trabalho [6, 22, 37]. Novamente, o objetivo é convencionar uma notação e facilitar a leitura desta dissertação por pessoas que não estejam tão familiarizadas com a área.

A primeira definição a ser apresentada é a base de todo o conteúdo desta seção. Um **grafo** $G = (V, E)$ é um par composto por um **conjunto de vértices** ou **nós** V e um **conjunto de arestas** ou **arcos** E . Este contém pares não ordenados de elementos de V . Um exemplo, que será usado ao longo de toda essa seção, seria um grafo G composto pelos conjuntos $V = \{a, b, c, d\}$ e $E = \{\{a, b\}, \{a, c\}, \{a, d\}, \{b, c\}, \{b, d\}\}$. Isso significa que G é formado por quatro **vértices**, rotulados de a, b, c e d , e por cinco **arestas**, que representam ligações entre pares de vértices. Para simplificar a notação, é frequente omitirmos as chaves e a vírgula ao nos referirmos a uma aresta, representando-a apenas pelos vértices relacionados por ela. O conjunto E poderia, portanto, ser reescrito como $\{ab, ac, ad, bc, bd\}$. Por convenção, normalmente dizemos que um grafo tem n vértices e m arestas, ou seja, $|V| = n$ e $|E| = m$. A cardinalidade de V também é chamada de **ordem** do grafo.

Em alguns casos faz-se necessário especificar a qual grafo pertencem os conjuntos V e E e, para isso, acrescenta-se o nome do grafo entre parênteses após o nome do conjunto. Dessa forma, $V(G)$ indica que nos referimos aos **vértices de G** e $E(G)$ significa que falamos das **arestas de G** .

Outra forma de descrever um grafo é por meio de um desenho em que os vértices são representados por círculos e, ligando um par de vértices por uma linha, formamos as arestas. A Figura 1.1 mostra essa representação para o grafo G .

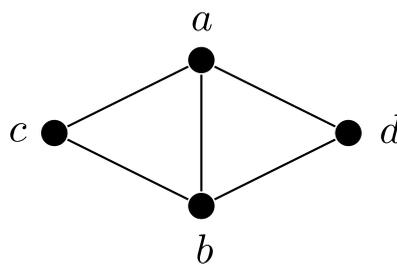


Figura 1.1: Representação gráfica de G .

Se existe uma aresta $uv \in E(G)$, dizemos que os vértices u e v são **adjacentes** ou **vizinhos**. Além disso, a aresta uv é **incidente** aos vértices u e v . Estes, por sua vez, são os **extremos** de uv . Essas relações dão origem a dois conjuntos, a **vizinhança aberta** de um vértice $v \in V(G)$, representada por $N(v)$, e a **vizinhança fechada** de v , que denotamos por $N[v]$. Ambos os conjuntos incluem todos os vértices adjacentes a v , mas, enquanto v é um elemento de $N[v]$, ele não pertence a $N(v)$. No grafo G acima, a vizinhança aberta do vértice c é $N(c) = \{a, b\}$, enquanto sua vizinhança

fechada é $N[c] = \{a, b, c\}$. Os vértices a e b , que são vizinhos de todos os outros, são chamados de vértices **universais**. Nesse caso, note que $N[a] = N[b] = V(G)$.

O número de vértices adjacentes a $v \in V(G)$ — a cardinalidade de sua vizinhança aberta — é chamado de **grau** de v . Representamos essa grandeza por $d(v)$. O maior e o menor graus dentre todos os vértices de G são denotados, respectivamente, por $\Delta(G)$ e por $\delta(G)$. Quando estiver claro a qual grafo esses parâmetros se referem, os parênteses podem ser omitidos para tornar a notação mais simples.

Dado um grafo $G = (V, E)$, o seu **complemento** é $\overline{G} = (V, \overline{E})$, em que \overline{E} é a diferença entre o conjunto de todas as arestas possíveis para os vértices em V e o conjunto E . No exemplo acima, os pares de vértices são $\{ab, ac, ad, bc, bd, cd\}$ e o conjunto E contém as arestas $\{ab, ac, ad, bc, bd\}$. Dessa forma, $\overline{E} = \{cd\}$. O grafo \overline{G} pode ser visto na Figura 1.2.

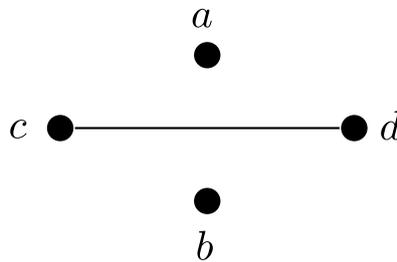


Figura 1.2: O grafo \overline{G} .

Ainda considerando um grafo $G = (V, E)$, dizemos que $H = (V', E')$ é um **subgrafo** de G se $V' \subseteq V$ e $E' \subseteq E$. Denota-se essa relação por $H \subset G$. Se E' contém todas as arestas de E que são incidentes a algum vértice de V' , então H é um subgrafo **induzido** de G . Seja $S \subseteq V$. Se H é subgrafo de G induzido pelos vértices de S , isso pode ser representado por $H = G[S]$. Se, por outro lado, H for induzido por $V \setminus S$, escrevemos $H = G - S$. A Figura 1.3 mostra dois subgrafos de G . Um **subgrafo gerador** de G é um grafo $H \subset G$ tal que $V(H) = V(G)$.

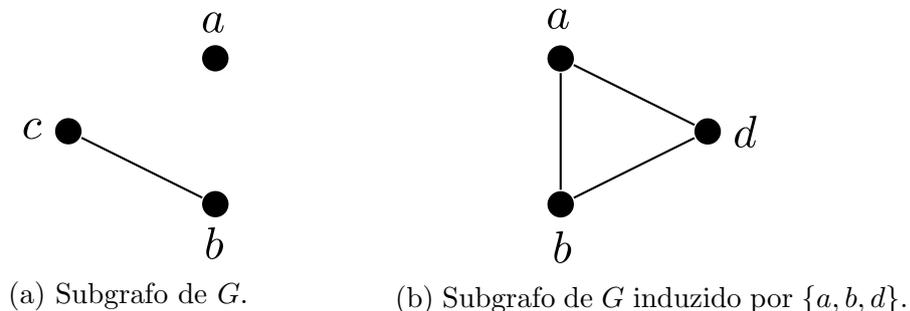


Figura 1.3: Subgrafos de G .

Por fim, se $S \subseteq V(G)$ é tal que $G[S]$ não tem nenhuma aresta, dizemos que S é um **conjunto independente** ou, ainda, que S é **estável**. Por outro lado, se

$G[S]$ tem arestas entre todos os pares de vértices distintos de S , dizemos que este conjunto é uma **clique**.

1.2.1 Percorrendo um grafo

A convexidade de grafos frequentemente relaciona-se com as formas de se percorrer um grafo, além de conceitos relacionados como conectividade e distância. Os principais percursos em um grafo são os **caminhos** e os **ciclos**.

Um **uv -caminho** em G é uma sequência de vértices distintos v_1, v_2, \dots, v_k tal que $v_1 = u$, $v_k = v$, $v_i \in V(G)$ para $1 \leq i \leq k$ e $v_i v_{i+1} \in E(G)$ para $1 \leq i < k$. Outra forma de descrever um uv -caminho é dizer que, partindo do vértice u , seguimos pelas arestas do grafo e anotamos os vértices por que passamos, terminando obrigatoriamente no vértice v sem nunca visitar o mesmo vértice mais de uma vez. Em G há três ab -caminhos, a saber a, b ; a, c, b ; e a, d, b .

Se abrimos uma exceção e terminamos um caminho no mesmo vértice em que começamos, temos o que chamamos de **ciclo**. No grafo G , podemos encontrar alguns ciclos como a, c, b, d, a e a, b, c, a . Observe que, na verdade, nenhum percurso fechado tem realmente começo ou fim, afinal sua natureza circular faz com que possamos considerar qualquer um de seus vértices como começo sem que efetivamente tenhamos um ciclo distinto.

Finalmente, outra definição indispensável diz respeito ao **comprimento** de um caminho ou ciclo, o qual corresponde ao número de arestas percorridas. Observe que, em um caminho, o comprimento corresponde ao número de vértices visitados menos uma unidade, enquanto, em um ciclo, há o mesmo número de vértices e arestas.

Conectividade de grafos

As formas de se percorrer um grafo são importantes para a questão de conectividade. Dizemos que um grafo G é **conexo** se existe um uv -caminho para quaisquer $u, v \in V(G)$ tais que $u \neq v$. Isso significa que o grafo G , que vem sendo utilizado como exemplo até o momento, é conexo, enquanto \overline{G} não é, sendo chamado de **desconexo**, pois não há caminho que ligue os vértices a e b a qualquer outro.

Um grafo pode ser dividido em **componentes conexos**. Um componente conexo de G é um conjunto maximal $S \subseteq V(G)$ tal que $G[S]$ é conexo. Considerando o grafo G que vem sendo utilizado como exemplo, há apenas um componente conexo, enquanto em \overline{G} há três, $\{a\}$, $\{b\}$ e $\{c, d\}$.

Se um vértice, ao ser removido, faz com que deixe de existir caminho entre algum par de vértices de um grafo, ele é chamado de uma **articulação**. Um subconjunto de vértices de um grafo que é capaz de torná-lo desconexo é um **separador**. Se não

há nenhuma clique separadora em um grafo conexo, este recebe o nome de **átomo**.

Também classificam-se os grafos com relação ao número de vértices que devem ser removidos para torná-lo desconexo, isto é, em função da cardinalidade de um separador mínimo. Diz-se que um grafo é **k -conexo** se for preciso remover ao menos k vértices para desconectá-lo, o que é equivalente a dizer que há ao menos k caminhos independentes — que não compartilham nenhum vértice entre si à exceção dos extremos — entre quaisquer dois vértices do grafo. No grafo G , é preciso remover, no mínimo, os vértices a e b para que G torne-se desconexo e, portanto, trata-se de um grafo 2-conexo. Cabe observar que qualquer grafo k -conexo, em que $k \geq 2$, não pode possuir nenhuma articulação. Aos grafos 2-conexos dá-se o nome especial de **biconexos**. Finalmente, note que um grafo que é k -conexo também é $(k-1)$ -conexo, por definição.

Distâncias em grafos

As distâncias entre vértices de um grafo são muito importantes para diversos problemas. Um dos principais, na era da internet, é o roteamento de pacotes pelas redes de computadores. A **distância** entre dois vértices $u, v \in V(G)$ é o comprimento do menor uv -caminho.

Um dos conceitos relacionados às distâncias em grafos é a **excentricidade** de um vértice $v \in V(G)$. Esse nome é dado à maior distância entre v e qualquer outro vértice de G . O conjunto de vértices de menor excentricidade em um grafo é chamado de **centro**. A maior dentre todas as excentricidades em um grafo é o seu **diâmetro**.

1.2.2 Classes de grafos

Vários problemas são muito difíceis quando consideramos um grafo qualquer, sem que tenhamos nenhuma noção quanto à sua estrutura, mas tornam-se mais simples se consideramos apenas alguns grafos específicos. Há, ainda, propriedades de alguns grafos que podem ser estudadas separadamente. Uma forma de permitir a análise de apenas alguns grafos que têm características em comum é dividi-los em **classes de grafos**. Uma classe agrupa todos os grafos que compartilham algumas propriedades e características específicas [8].

Dadas duas classes de grafos, dizemos que uma é **subclasse** da outra se os grafos que pertencem à primeira formam um subconjunto daqueles que pertencem à segunda. Isso significa que esses grafos, além de terem todas as propriedades da superclasse, apresentam outras características comuns que permitem a definição de uma nova classe.

A seguir, serão apresentadas algumas classes de grafos que terão relevância ao longo deste trabalho.

Grafos completos

Um **grafo completo** de n vértices, que recebe o nome especial de K_n , é tal que, para todo par de vértices $u, v \in V(K_n)$, vale que $uv \in E(K_n)$. Em outras palavras, isso significa que um grafo completo é aquele que tem todas as arestas possíveis dado o seu conjunto de vértices. Dessa forma, o número de arestas em K_n é o número de pares distintos de vértices, dado por $\binom{n}{2} = \frac{n(n-1)}{2}$. Na Figura 1.4 estão representados alguns exemplos.

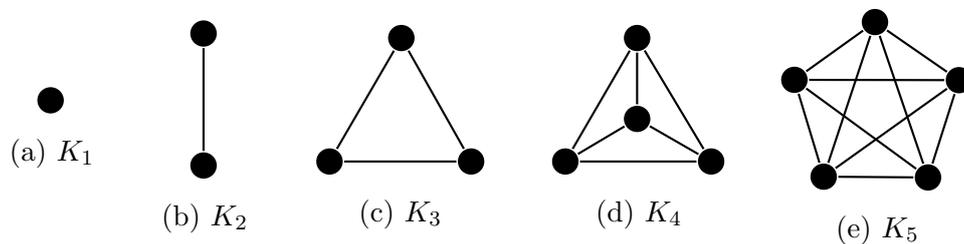


Figura 1.4: Alguns grafos completos.

Todo grafo completo é, portanto, conexo e a distância entre quaisquer vértices em K_n é um. Note que os grafos completos não se encaixam muito bem na definição de k -conectividade, visto que não é possível torná-los desconexos. Diz-se, então, que um K_n é $(n - 1)$ -conexo, por definição.

Grafos *split*

Os **grafos *split*** são aqueles cujo conjunto de vértices pode ser particionado em dois subconjuntos não vazios, C e I , sendo o primeiro uma clique e o segundo estável. Isso significa que, à exceção do K_1 , todo grafo completo é também *split*.

Chamamos a partição $V(G) = C \cup I$ em que C é uma clique e I é um conjunto independente, ambos com ao menos um elemento, de uma **partição *split***. Conforme pode ser visto na Figura 1.5, uma partição *split* não é necessariamente única. Nela, os vértices preenchidos pertencem ao conjunto C , enquanto os demais fazem parte do conjunto independente I .

Observando a Figura 1.5, podemos perceber que não há outra partição *split* do grafo representado. De fato, dado um grafo *split* G , qualquer partição *split* $V(G) = C \cup I$ em que C é uma clique e I é estável tem que satisfazer a uma das três possibilidades a seguir.

- (a) C é uma clique máxima e I é um conjunto independente máximo e a partição *split* é única;

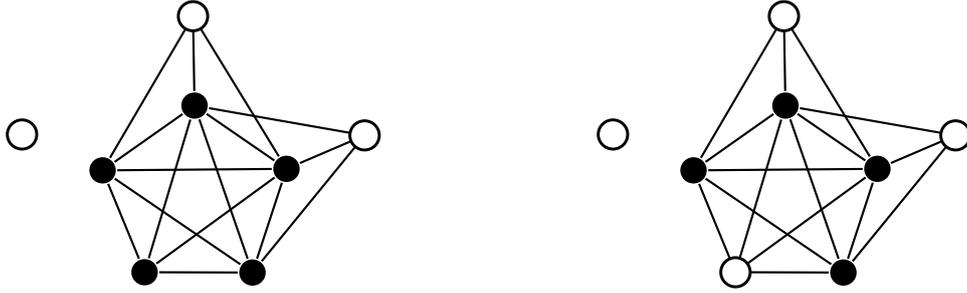


Figura 1.5: Exemplo de grafo *split* e duas possíveis partições *split*.

- (b) C é uma clique máxima e existe um vértice $v \in C$ tal que $I \cup \{v\}$ é um conjunto independente máximo;
- (c) I é um conjunto independente máximo e existe um vértice $v \in I$ tal que $C \cup \{v\}$ é uma clique máxima.

Disso, pode-se concluir que um grafo *split* qualquer tem exatamente uma ou duas partições *split* não isomorfas.

Grafos *threshold*

Os **grafos *threshold*** são uma subclasse dos grafos *split* em que as vizinhanças de todos os vértices são aninhadas. Dizemos que as vizinhanças de dois vértices $u, v \in V(G)$ são **aninhadas** se $N(u) \subseteq N(v)$. Isso significa que, para quaisquer dois vértices u e v de um grafo *threshold*, $N(u) \subseteq N(v)$ ou $N(v) \subseteq N(u)$. Na Figura 1.6 vê-se um exemplo de grafo *threshold*.

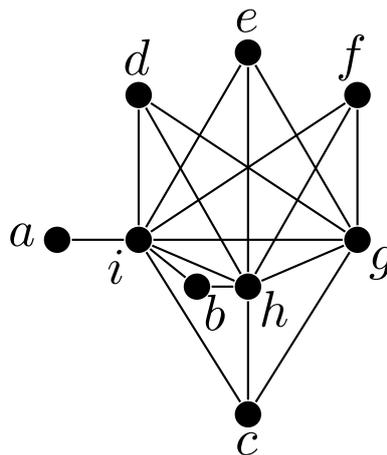


Figura 1.6: Exemplo de grafo *threshold* em que $N(a) \subseteq N(b) \subseteq N(c) \subseteq N(d) \subseteq N(e) \subseteq N(f) \subseteq N(g) \subseteq N(h) \subseteq N(i)$.

Essa ordem dos vértices, devido às vizinhanças aninhadas, pode ser dada simplesmente pelos graus dos nós. Se os colocamos em ordem não decrescente de grau v_1, v_2, \dots, v_n com $d(v_1) \leq d(v_2) \leq \dots \leq d(v_n)$, temos naturalmente $N(v_1) \subseteq N(v_2) \subseteq \dots \subseteq N(v_n)$.

Grafos bipartidos

Um grafo $G = (V, E)$ é **bipartido** se podemos particionar $V = X \cup Y$ de modo que, se existe aresta $uv \in E$, então $u \in X$ e $v \in Y$. Outra forma de caracterizar grafos bipartidos é dizer que G é bipartido se não tem nenhum ciclo ímpar. A Figura 1.7 mostra um exemplo de grafo bipartido e algumas possíveis bipartições de seus vértices.

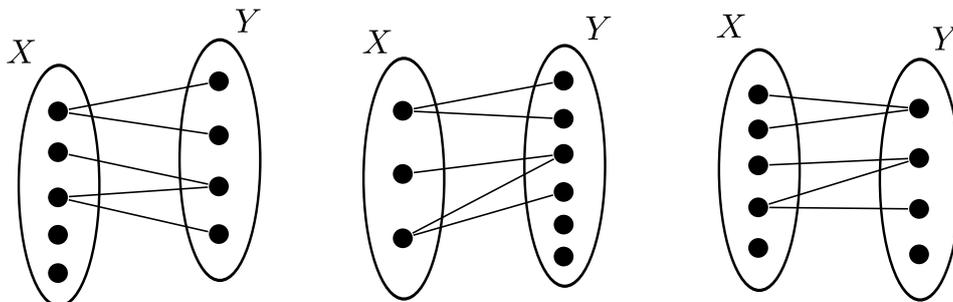


Figura 1.7: Um grafo bipartido com três bipartições distintas de seus vértices.

Dizemos que um grafo é **bipartido completo**, denotado por $K_{s,t}$, se tem $s + t$ vértices bipartidos em $V(K_{s,t}) = X \cup Y$ de modo que $|X| = s$, $|Y| = t$ e todo vértice de X é adjacente a todo nó de Y . A Figura 1.8 mostra um exemplo.

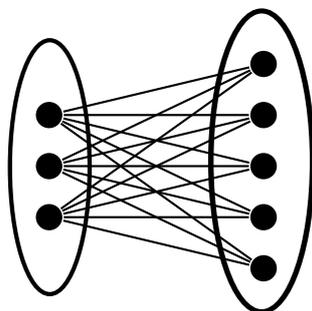


Figura 1.8: Um grafo bipartido completo, chamado de $K_{3,5}$.

Árvores

Uma **árvore** é um grafo conexo e acíclico. Isso também significa que há exatamente um caminho entre cada par de vértices. Como não há ciclos, não pode haver ciclos ímpares e, portanto, as árvores são uma subclasse dos grafos bipartidos. A Figura 1.9 mostra um exemplo.

Se adicionamos uma aresta a uma árvore T , ela obrigatoriamente forma um ciclo. Ao mesmo tempo, se um arco for removido, o grafo torna-se desconexo. Dado o número n de vértices de uma árvore, o número de arestas m é sempre $n - 1$.

Aos vértices de grau um de uma árvore dá-se o nome de **folhas**. Os demais são chamados de **nós internos**. Podemos escolher um vértice para ser a raiz da árvore,

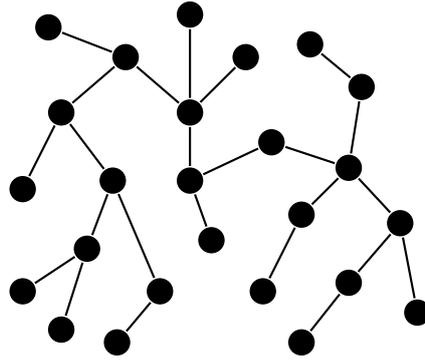


Figura 1.9: Um exemplo de árvore.

dizendo que a mesma foi **enraizada**. Com isso, dado um vértice qualquer v , um de seus vizinhos que se encontra após v no caminho da raiz até uma folha é chamado de **filho** de v , enquanto o vizinho que aparece logo antes de v nesse mesmo caminho é o seu **pai**. Quando a árvore está enraizada, a **subárvore** com raiz em um vértice v é o subgrafo induzido por v e por todos os seus descendentes.

Por fim, um grafo em que todo componente conexo é acíclico é chamado de uma **floresta**. Conforme o nome indica, uma floresta é a união de uma ou mais árvores.

Caminhos

Um **caminho** é uma árvore que não tem nenhuma ramificação. Esse tipo de grafo é denotado por P_n , sendo n o número de vértices. Alguns exemplos podem ser vistos na Figura 1.10.

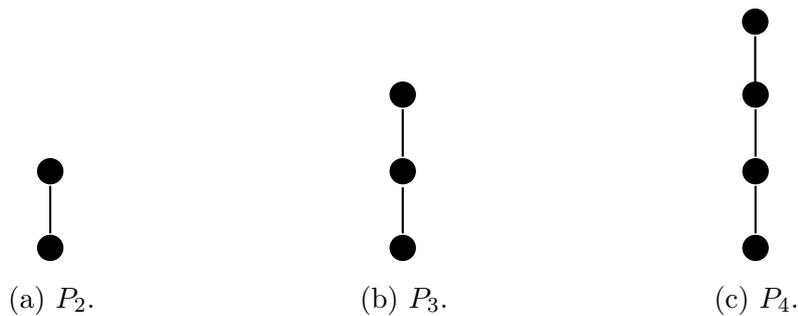


Figura 1.10: Alguns exemplos de caminhos.

Ciclos

Os **ciclos** são grafos formados por um único ciclo, ou seja, vértices de grau dois conectados em uma cadeia fechada. Representa-se um ciclo de n vértices como C_n . A Figura 1.11 mostra alguns exemplos.

Um ciclo pode ser classificado como par ou ímpar em função de seu número de vértices e arestas. Cabe observar que C_{2n} é bipartido para qualquer valor de n ,

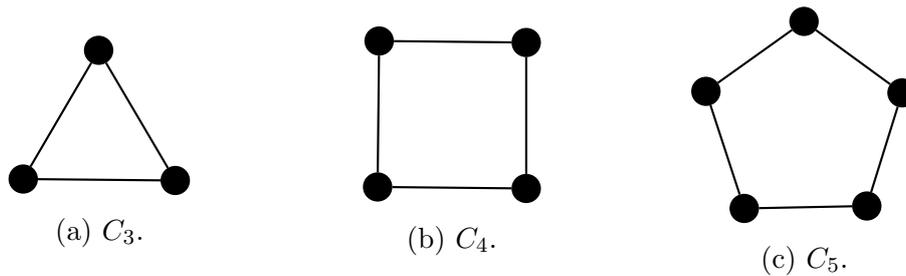


Figura 1.11: Alguns exemplos de ciclos.

enquanto C_{2n+1} , por ser um ciclo ímpar, não.

Grafos cordais

Um **grafo cordal** não contém nenhum C_n , $n \geq 4$, como subgrafo induzido, ou seja, todo ciclo de quatro ou mais vértices tem ao menos uma **corda**. Uma corda é uma aresta que divide o ciclo, conforme mostra a Figura 1.12.

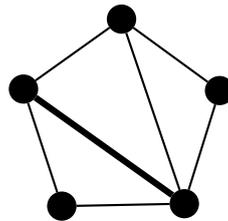


Figura 1.12: Uma corda em destaque em um ciclo de cinco vértices.

Os grafos cordais são muito estudados por terem algumas propriedades relevantes como a existência de ao menos dois **vértices simpliciais** e de um **esquema de eliminação perfeito**. Um vértice simplicial $v \in V(G)$ é tal que $N(v)$ é uma clique, e um esquema de eliminação perfeito é uma ordem dos vértices do grafo v_1, v_2, \dots, v_n tal que, dado v_i , todo par de vértices v_j, v_k com $i < j < k$ e tal que $v_i v_j \in E(G)$ e $v_i v_k \in E(G)$ também satisfaz $v_j v_k \in E(G)$.

Grafos de intervalo

A classe dos **grafos de intervalo** é uma subclasse dos grafos cordais e compreende todo grafo G que pode ser construído a partir de um **modelo de intervalos**. A Figura 1.13 mostra um desses modelos, a partir do qual será construído um grafo a ser usado como exemplo da forma que será descrita a seguir.

Para construir o grafo $G = (V, E)$ a partir de um modelo, constrói-se V associando cada intervalo a exatamente um vértice de modo que cada nó corresponda, por sua vez, a exatamente um intervalo. No exemplo da Figura 1.13, criaremos os vértices v_i correspondentes a I_i . Uma aresta $v_i v_j$ existe se, e somente se, os intervala-

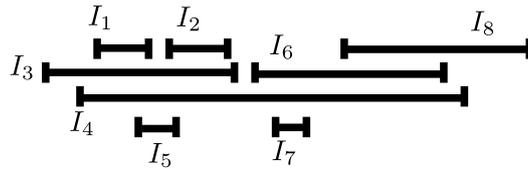


Figura 1.13: Um modelo de intervalos.

los I_i e I_j têm alguma interseção, ou seja, I_i e I_j compartilham alguma coordenada x . A partir do modelo acima, portanto, constrói-se o grafo da Figura 1.14.

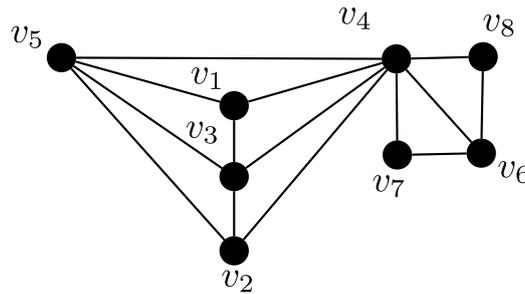


Figura 1.14: Grafo de intervalo criado a partir do modelo da Figura 1.13.

1.3 Complexidade de algoritmos

“Lord, what fools these mortals
bel!”

William Shakespeare, A
Midsummer Night’s Dream

A complexidade de um algoritmo é uma medida dos recursos exigidos para que a sequência de passos seja executada de modo a se obter uma solução para o problema. Existem diversos parâmetros a serem medidos, sendo os mais comuns tempo de execução e espaço de armazenamento necessário. Chamamos as medidas desses dois parâmetros, respectivamente, de **complexidade de tempo** e **complexidade de espaço** [37].

Uma vez que o objetivo dessa dissertação é fazer uma análise da complexidade de tempo do cálculo do posto de um grafo, esta seção tem como foco principal essa medida. Para isso, será feita uma breve introdução à notação assintótica e ao cálculo da complexidade de tempo, seguida das classes de complexidade de maior interesse para este trabalho.

1.3.1 Complexidade de tempo

Uma das formas mais comuns — se não a mais comum — de se encontrar um algoritmo nos dias de hoje é na forma de *software*. Desde os primeiros computadores, o tempo de execução de um programa depende de fatores como velocidade de processamento da máquina, alocação de memória, entre outros fatores. Nos dias de hoje, em que temos diversos aparelhos com características muito distintas executando algoritmos, como computadores, celulares, *tablets*, TVs, entre outros, a variação do poder de processamento é ainda mais significativa. É, portanto, impossível medir a eficiência de um algoritmo com uma métrica como o tempo, pois dizer que uma determinada sequência de passos leva um segundo em um *desktop* dos mais robustos é completamente diferente de dizer que um algoritmo pode ser executado no mesmo tempo por um celular de custo não muito alto. É preciso, dessa forma, estabelecer uma métrica para o tempo de execução de um algoritmo que seja independente de outros fatores que não a sua eficiência.

Para resolver esse tipo de problema, mede-se a complexidade de tempo de um algoritmo em função do tamanho de sua entrada, determinando-se não o tempo de execução, mas o quanto ele cresce à medida em que há mais informação a ser processada. Tomando como exemplo um grafo, que precisa ser dado por meio de seus conjuntos de vértices e arestas, mede-se a complexidade de tempo de um algoritmo que o utilize como entrada em função das cardinalidades de cada um desses conjuntos.

Outro ponto importante na medição da eficiência de um algoritmo é que, se não é possível medir o tempo, faz-se necessária outra forma de determinar o quão rápido ele será executado. Isso é feito contando-se o número de operações de tempo constante necessárias para que a solução seja encontrada. Uma operação de tempo constante é qualquer uma que seja independente do tamanho da entrada, como operações aritméticas, desvios condicionais, entre outras. A representação do número de operações é dada por meio de notação assintótica.

Notação assintótica

A notação assintótica é usada com o objetivo de determinar o comportamento de uma função utilizando outra como parâmetro. Isso é representado por $f(x) = O(g(x))$, $f(x) = \Omega(g(x))$ e $f(x) = \Theta(g(x))$, cujos significados serão descritos a seguir.

Dizemos que $f(x) = O(g(x))$ se a função f é limitada superiormente pela função g multiplicada por alguma constante k a partir de algum valor x_0 . Isso significa que, se um algoritmo tem $O(1)$ operações constantes, então o seu tempo de execução independe do tamanho da entrada, enquanto outro que tenha $O(n)$ executa um

número de operações constantes que cresce linearmente com relação ao parâmetro n . Observe que podemos descartar qualquer constante, visto que a função dentro dos parênteses pode ser multiplicada por qualquer fator k , o que significa que $O(2n)$ é o mesmo que $O(n)$.

Já uma função $f(x) = \Omega(g(x))$ se f é limitada inferiormente por g multiplicada por alguma constante positiva k . Naturalmente, todo algoritmo tem $\Omega(1)$ operações, visto que não é possível fazer qualquer operação em um tempo que seja menor do que uma constante. Essa notação é especialmente relevante quando desejamos provar que um algoritmo mais eficiente é impossível. Um exemplo é a ordenação utilizando comparações, a qual sabemos que tem custo $\Omega(n \log n)$.

Por fim, $f(x) = \Theta(g(x))$ significa que, simultaneamente, $f(x) = O(g(x))$ e $f(x) = \Omega(g(x))$, ou seja, a complexidade de um algoritmo representada com Θ é justa.

1.3.2 Classes de complexidade

Com base no modelo teórico da máquina de Turing, algumas classes de complexidade de algoritmos foram definidas. Tal modelo é uma abstração matemática que representa uma máquina capaz de executar um algoritmo movimentando uma cabeça de leitura e escrita sobre uma fita limitada à esquerda e infinita à direita, dividida em células que podem conter símbolos ou estar em branco. Uma máquina de Turing é equivalente aos computadores que utilizamos atualmente, sendo uma ferramenta muito importante para se explorar os limites do modelo de computação atual. Um dos resultados mais clássicos da computação foi obtido pelo próprio Alan Turing, que provou que há problemas que não podem ser resolvidos dentro deste modelo como o Problema da Parada.

Uma máquina de Turing pode ser determinística ou não determinística. No primeiro caso, ela sempre resolve corretamente o problema, enquanto o segundo permite que sejam usadas algumas “adivinhações” para se tentar chegar a um resultado, não havendo problemas se, em alguns casos, chegue-se ao resultado errado, desde que haja um resultado correto que possa ser verificado. O tempo de execução de um algoritmo em uma máquina de Turing é dado pelo número de operações executadas pela cabeça de leitura e escrita, entre movimentações e alterações na fita.

Apesar de existir uma máquina de Turing de propósito geral, a construção de um desses modelos se dá com uma finalidade específica, isso é, para um determinado problema, existe uma máquina de Turing que o resolve. Uma máquina de propósito geral é chamada de máquina de Turing universal e é capaz de receber como entrada outra máquina e executar o algoritmo correspondente.

Uma das classes de complexidade mais importantes é chamada de \mathcal{NP} e com-

preende todos os problemas que podem ser resolvidos em tempo polinomial por uma **máquina de Turing não determinística**. Uma vez que, nesse modelo, podemos fazer uso de “adivinhações”, uma forma simples de se construir tal máquina é escolher uma resposta qualquer dentre todas as que forem possíveis e, em seguida, verificar se tal resposta está correta. Isso significa que a classe \mathcal{NP} compreende todos os problemas cuja solução pode ser verificada em tempo polinomial.

Outra classe relevante é conhecida como \mathcal{P} , a qual engloba todos os problemas que podem ser resolvidos em tempo polinomial por uma **máquina de Turing determinística**. Note que há uma restrição maior para \mathcal{P} do que para \mathcal{NP} , visto que não se pode mais tentar acertar a solução correta de forma aleatória, sendo preciso seguir sempre uma mesma sequência previsível de operações para se chegar ao resultado. Naturalmente, sabemos que $\mathcal{P} \subseteq \mathcal{NP}$, visto que, se uma máquina de Turing determinística pode resolver o problema em tempo polinomial, tudo que a máquina de Turing não determinística precisa fazer é seguir os mesmos passos para chegar à solução.

No ano 2000, o *Clay Mathematics Institute* selecionou sete problemas considerados alguns dos mais desafiadores da matemática contemporânea e ofereceu um prêmio de um milhão de dólares para quem resolvesse qualquer um deles corretamente. Até o momento, seis desses problemas permanecem em aberto, entre eles a pergunta se \mathcal{P} é igual a \mathcal{NP} .

Uma terceira classe de complexidade é a dos problemas \mathcal{NP} -completos, que também é uma subclasse de \mathcal{NP} , que compreende problemas cuja solução pode ser verificada em tempo polinomial, mas para os quais não se conhece nenhuma solução eficiente. Até o momento, não se conhece nenhum algoritmo polinomial para resolver esse tipo de problema, sendo preciso usar algoritmos exponenciais para esse fim. Há, contudo, reduções com custo polinomial entre todos os problemas dessa classe e, portanto, resolver um deles rapidamente leva toda a classe para dentro de \mathcal{P} . Alguns problemas clássicos são \mathcal{NP} -completos como o problema do caixeiro viajante, o 3-SAT, o problema do ciclo hamiltoniano, o problema da mochila e a coloração de grafos.

Capítulo 2

Apresentação do problema

“Do or do not. There is no try.”

Mestre Yoda, Star Wars

Este trabalho tem como objetivo estudar o **posto**, um dos parâmetros relacionados à convexidade de grafos [38]. Para isso, é preciso começar descrevendo uma convexidade de grafos e, posteriormente, apresentar o parâmetro e, finalmente, formalizar o problema em questão.

A seguir, na Seção 2.1, será apresentada a convexidade de grafos, bem como alguns dos principais parâmetros relacionados ao assunto. Na Seção 2.2, define-se formalmente o problema a ser apresentado e que aspectos serão o foco do capítulo seguinte, que apresenta os resultados obtidos. Finalmente, são descritos, na Seção 2.3, alguns problemas relacionados que são utilizados nas provas de \mathcal{NP} -completude ou que possuem resultados derivados dos teoremas sobre o problema do CONJUNTO CONVEXAMENTE INDEPENDENTE.

2.1 Convexidade

“It’s the job that’s never started
that takes longest to finish.”

J.R.R. Tolkien, The Lord of the
Rings

O conceito de convexidade vem, originalmente, da geometria euclidiana. Nesse contexto, é dito que um polígono é convexo se, ao ligarmos quaisquer dois de seus pontos por um segmento de reta, todo este segmento está contido dentro do polígono. A Figura 2.1a mostra um polígono não convexo — também chamado de côncavo — e a Figura 2.1b contém um exemplo convexo.



(a) Um polígono côncavo de dez lados. (b) Um polígono convexo de dez lados.

Figura 2.1: Um polígono pode ser côncavo ou convexo.

De modo geral, se V é um conjunto, uma família \mathcal{C} de subconjuntos de V é uma **convexidade** se as seguintes condições são satisfeitas:

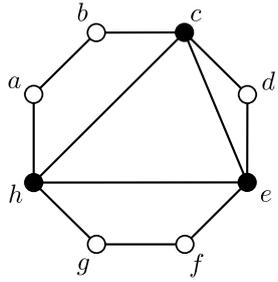
- (i) os conjuntos \emptyset e V pertencem a \mathcal{C} ;
- (ii) \mathcal{C} é fechada por interseção; e
- (iii) a união de uma cadeia de elementos ordenada por inclusão de \mathcal{C} está em \mathcal{C} .

Se a convexidade é finita, a condição (iii) é sempre satisfeita. Se um conjunto $S \in \mathcal{C}$, dizemos que S é **convexo** [38].

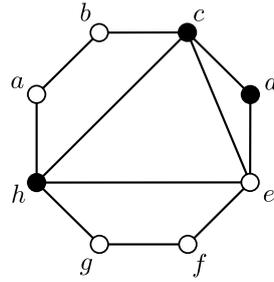
Exemplo: Se $V = \{1, 2, 3, 4\}$ e $\mathcal{C} = \{\emptyset, \{1\}, \{1, 2\}, \{1, 3\}, \{1, 2, 3, 4\}\}$, então \mathcal{C} é uma convexidade, uma vez que satisfaz às condições (i), (ii) e (iii).

Quando aplicamos o conceito de convexidade a um grafo G , o conjunto universo é $V(G)$ e formamos subconjuntos de vértices de G que podem ser ou não convexas. Frequentemente são estudadas convexidades em que os conjuntos formados apresentem algum tipo de estrutura ou propriedade relevante. No caso deste trabalho, temos como foco a chamada **convexidade de caminhos**, a qual aproxima-se bastante da definição original da geometria euclidiana. Para formar uma dessas convexidades, deve-se considerar uma família de caminhos \mathcal{F} de um grafo $G = (V, E)$. Dado um par de vértices $u, v \in V$, definimos o **intervalo fechado** $I_{\mathcal{F}}(u, v)$, o qual consiste de todos os vértices que fazem parte de algum caminho de \mathcal{F} que ligue u a v , inclusas as extremidades. A família de caminhos será omitida na notação de intervalo quando estiver clara pelo contexto.

A Figura 2.2 mostra dois exemplos de conjunto. Considerando-se a família \mathcal{F} com todos os caminhos mais curtos do grafo, os vértices pretos da Figura 2.2a formam um conjunto convexo, enquanto, na Figura 2.2b não. Note que o conjunto da segunda figura não é convexo porque há dois caminhos mínimos entre h e d e, enquanto h, c, d está totalmente incluído no conjunto, h, e, d não. É preciso, portanto, incluir o vértice e no conjunto para que ele torne-se convexo.



(a) Os vértices pretos formam um conjunto convexo.



(b) Os vértices pretos formam um conjunto não convexo.

Figura 2.2: Considerando uma família \mathcal{F} com todos os caminhos mais curtos do grafo, a Figura 2.2a mostra um conjunto convexo, enquanto o conjunto da Figura 2.2b não é convexo.

2.1.1 Convexidades de caminhos relevantes

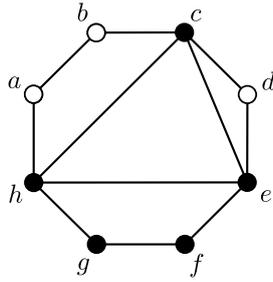
Apesar de existirem diversas possíveis convexidades em um grafo, algumas delas são mais conhecidas e estudadas. Dentre elas, podemos citar como principais exemplos a convexidade **geodética**, a **monofônica** e a \mathbf{P}_3 [2, 9, 11, 12]. As três são convexidades de caminhos, sendo, portanto, diferenciadas pela família de caminhos escolhida.

Convexidade geodética

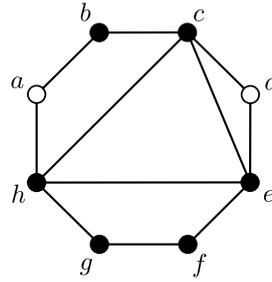
A convexidade mais estudada é a convexidade geodética, cujos conjuntos convexos são construídos com base em caminhos mínimos [12]. Isso significa que, se $S \subseteq V(G)$ é um conjunto convexo, então, para todo par de vértices $u, v \in S$, todo nó que faça parte de um caminho de comprimento mínimo entre u e v também deve pertencer a S . Note que, se houver mais de um uv -caminho mínimo, todos os vértices que façam parte de qualquer um desses caminhos devem estar no conjunto S . Um exemplo pode ser visto na Figura 2.2, pois a família dos caminhos mais curtos do grafo é justamente aquela que caracteriza a convexidade geodética.

Convexidade monofônica

A convexidade monofônica, na qual os conjuntos convexos são formados a partir de caminhos induzidos, também é uma das mais relevantes e já foi estudada por diversos autores [14]. Se $S \subseteq V(G)$ é convexo nesse contexto, então, para todo par de vértices $u, v \in S$, qualquer nó que faça parte de algum caminho induzido de u até v , ainda que tal caminho não tenha comprimento mínimo, deve pertencer a S . Um exemplo com um conjunto convexo e outro não, pode ser visto na Figura 2.3.



(a) Os vértices pretos formam um conjunto convexo do grafo na convexidade monofônica.

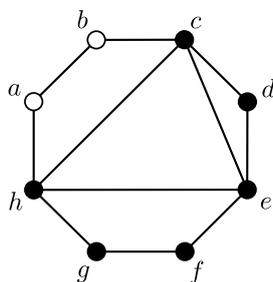


(b) Os vértices pretos não formam um conjunto convexo na convexidade monofônica.

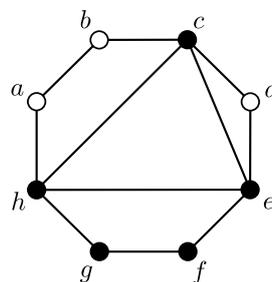
Figura 2.3: Enquanto o conjunto formado pelos vértices pretos na Figura 2.3a é convexo, os vértice a e b fazem parte de um ch -caminho induzido e, portanto, fazem com que o conjunto representado na Figura 2.3b não seja convexo.

Convexidade P_3

Outra convexidade relevante é a chamada convexidade P_3 [4, 9, 15]. Nela, a família de caminhos é composta por todos os caminhos de comprimento dois do grafo, ou seja, todos os caminhos com três vértices. Daí o nome P_3 . Um ponto interessante a ser observado é que, nessa convexidade, há diversas propriedades locais advindas do fato de que os caminhos são curtos. Por um lado, a característica principal de um conjunto convexo $S \subseteq V(G)$ é que, para quaisquer $u, v \in S$, qualquer outro nó que esteja no meio de um caminho de comprimento dois ligando u a v também deve fazer parte do conjunto. Por outro, também podemos verificar facilmente se um conjunto S é convexo considerando apenas a vizinhança de cada vértice de $V(G) \setminus S$. S é convexo se, e somente se, não há nenhum vértice $v \in V(G) \setminus S$ que tenha dois ou mais vizinhos pertencentes ao conjunto S . A Figura 2.4 mostra um exemplo com um conjunto convexo e outro não convexo.



(a) Os vértices pretos formam um conjunto convexo do grafo na convexidade P_3 .



(b) Os vértices pretos não formam um conjunto convexo na convexidade P_3 .

Figura 2.4: O vértice d , incluído no conjunto da Figura 2.4a, mas que não faz parte do conjunto da Figura 2.4b tem dois vizinhos, c e e , que fazem parte de S em ambos os casos. Dessa forma, o conjunto deixa de ser convexo com a remoção de d .

A convexidade P_3 é o foco principal desta dissertação e, dessa forma, a não ser que o contrário seja dito de forma explícita, deve-se considerar que todo o texto e todos os exemplos a seguir são construídos nesse contexto.

2.1.2 Envoltória convexa

Outro conceito que pode ser mais facilmente compreendido se mencionada a sua origem geométrica é a noção de **envoltória convexa**. A envoltória convexa de um conjunto de pontos é o menor polígono convexo que contenha todos os pontos desse conjunto, conforme mostra a figura 2.5.

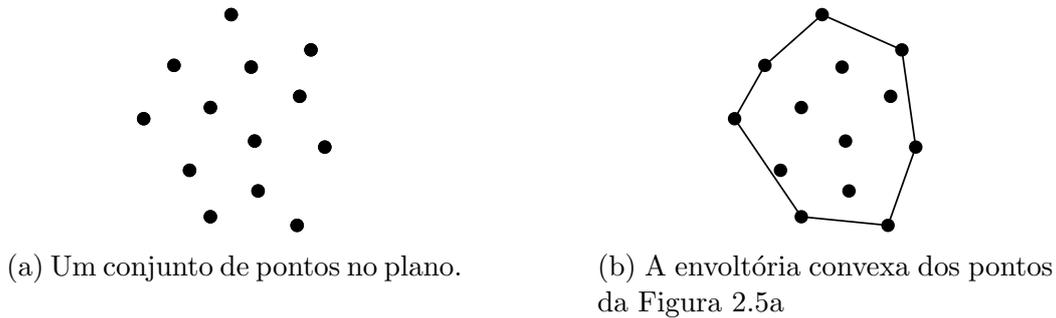


Figura 2.5: Envoltória convexa no contexto da geometria euclidiana. O menor polígono convexo que contém todos os pontos de um conjunto.

Quando tratamos de uma convexidade em grafos, a envoltória convexa de um conjunto S não é muito diferente da geometria. Por vezes chamado de **fecho convexo**, trata-se do menor conjunto convexo que contém S , que denotamos por $H_{\mathcal{C}}(S)$ [13]. Quando isso estiver claro pelo contexto, pode-se omitir \mathcal{C} . Um exemplo pode ser visto na Figura 2.6.

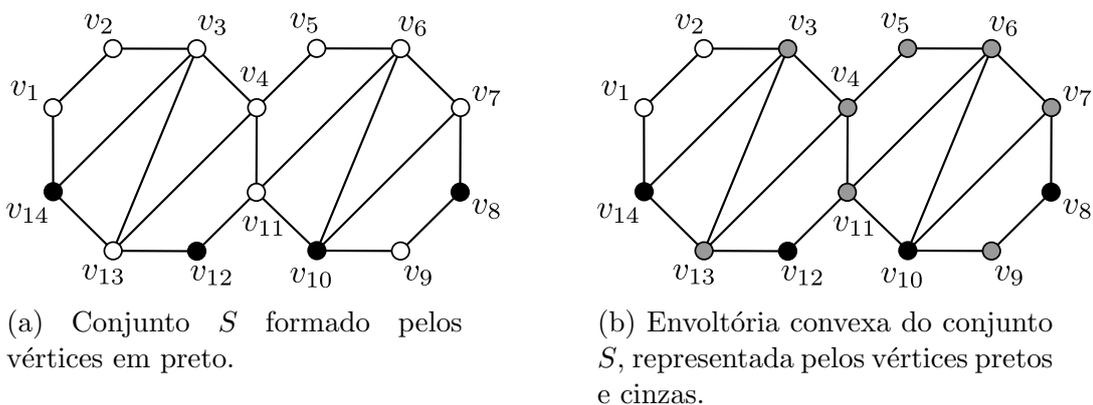


Figura 2.6: Exemplo de envoltória convexa em um grafo na convexidade P_3 . A Figura 2.6a contém um conjunto S e a Figura 2.6b a sua envoltória convexa $H_{P_3}(S)$.

Para obtermos a envoltória convexa de um conjunto $S \subseteq V(G)$, podemos utilizar um algoritmo iterativo, adicionando todos os vértices necessários para que o conjunto

torne-se convexo a cada iteração. Dizemos que esses vértices adicionados a cada iteração são **gerados** pelos demais. A Figura 2.7 mostra a execução desse algoritmo com a finalidade de produzir a envoltória convexa da Figura 2.6.

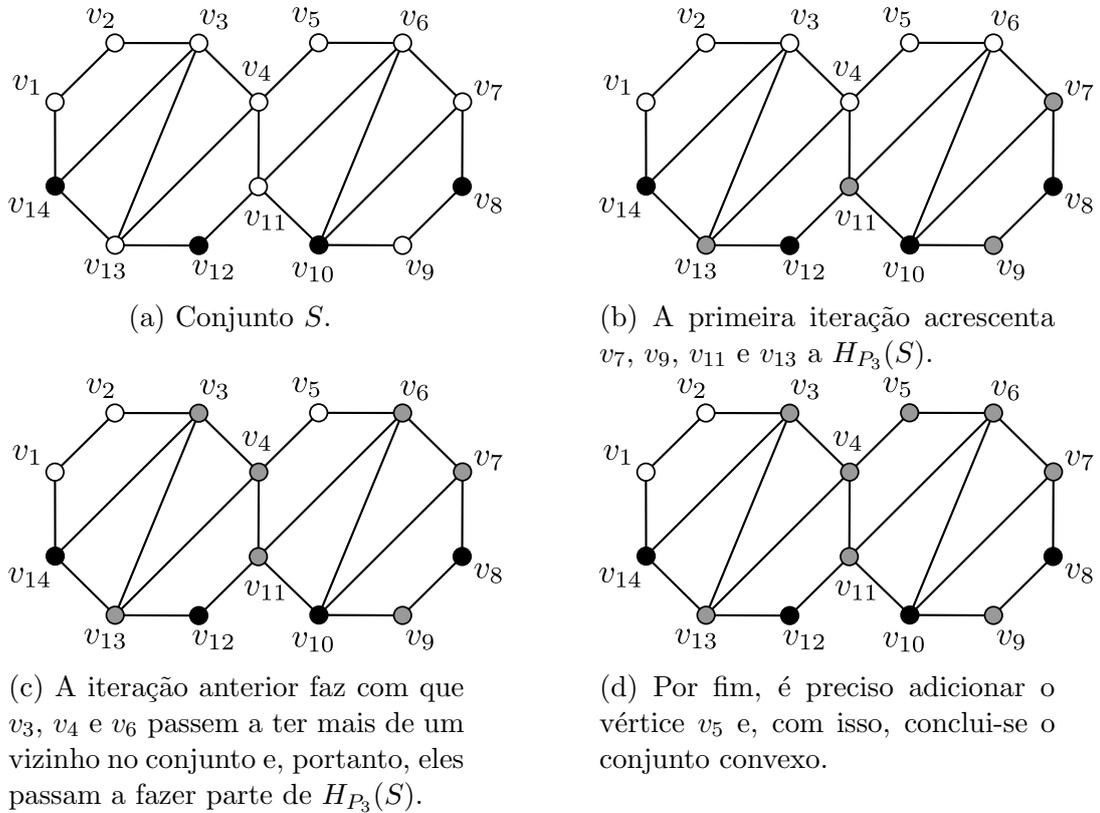


Figura 2.7: Exemplo passo a passo do algoritmo iterativo que forma a envoltória convexa de um conjunto. Na convexidade P_3 , a cada etapa, os vértices que têm ao menos dois vizinhos no conjunto são adicionados à envoltória.

Na convexidade P_3 , este conceito pode ser aplicado a diversos contextos diferentes em problemas reais que envolvem a “contaminação” de um vértice por seus vizinhos. Em contextos como difusão de informação ou de um anúncio em uma rede social, contágio de uma doença ou contaminação de algum recurso a convexidade P_3 pode ser utilizada na produção de um modelo que represente esses processos.

Considere, por exemplo, a propagação de uma campanha publicitária em uma rede social. Muitas pessoas têm preguiça de assistir a vídeos, principalmente se eles contêm apenas o posicionamento de um produto. Essa situação muda de figura, porém, se alguns de seus amigos começam a compartilhá-lo e a chance do usuário se dar o trabalho de verificar o conteúdo é maior. Caso a campanha seja boa, há também uma grande chance dessa pessoa também vir a compartilhar o vídeo em questão. Uma forma de modelar esse comportamento com uma pequena simplificação é supor que uma pessoa vê um vídeo e o compartilha sempre que dois de seus amigos já o fizeram. Dessa forma, a envoltória convexa, de acordo com a convexidade P_3 , do conjunto de usuários que recebe o vídeo inicialmente determina o

alcance total que a campanha terá.

2.1.3 Parâmetros de uma convexidade de grafos

Há diversos parâmetros relacionados às convexidades de grafos, os quais podem, em muitos casos, ser aplicados diretamente a problemas reais. A seguir, serão apresentados alguns dos mais importantes e mais estudados até o momento, bem como o principal objeto desta pesquisa, o posto, que é um parâmetro ainda pouco explorado das convexidades de grafos.

Número de envoltória

Tomando novamente o exemplo da campanha publicitária, muitas vezes nos será oferecida a opção de pagar para que um determinado número de usuários receba o vídeo inicialmente e possa dar início à difusão da propaganda. Isso significa que é interessante determinar o mínimo que será preciso pagar para que exista a possibilidade de que o vídeo seja assistido por todos os usuários da rede.

Um conjunto S tal que $H(S) = V(G)$ é chamado de **conjunto de envoltória**. No contexto acima, qualquer conjunto de envoltória é capaz de atingir toda a rede, mas é preferível pagar por um número menor de usuários e, portanto, é interessante descobrir qual é o **número de envoltória** do grafo, $h(G)$, que corresponde à cardinalidade do menor conjunto de envoltória de G .

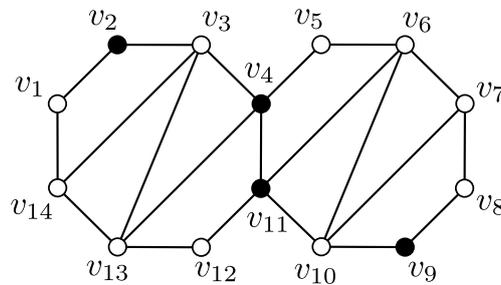


Figura 2.8: O conjunto formado pelos vértices pretos é um conjunto de envoltória mínimo do grafo. Isso significa que o número de envoltória desse grafo é quatro.

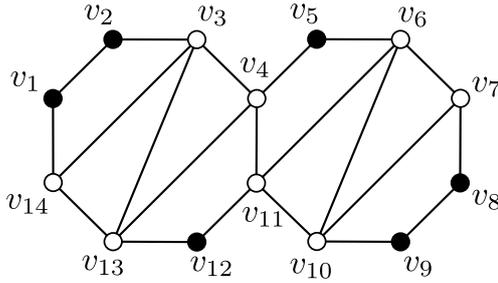
O grafo da Figura 2.8 tem diversos conjuntos de envoltória de tamanho maior ou igual a quatro. Não há, contudo, nenhum conjunto $S \subseteq V(G)$ que gere todos os demais vértices do grafo e tenha cardinalidade menor ou igual a três. Dessa forma, conclui-se que o número de envoltória do grafo é quatro.

Número de Radon

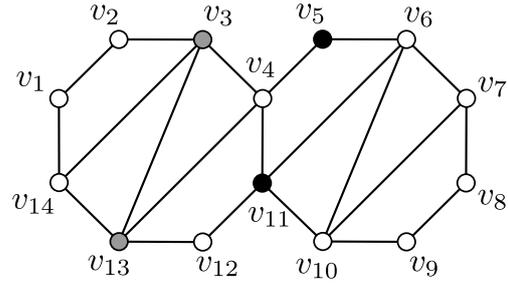
Outro conceito importado da geometria é o de **partição de Radon** de um conjunto R , que é uma divisão de R em dois subconjuntos $R_1 \cup R_2 = R$ de modo

que $H(R_1) \cap H(R_2) \neq \emptyset$. Caso não seja possível construir tal partição, dizemos que R é um conjunto **antirradon**.

O chamado **número de Radon**, denotado por $r(G)$, é a cardinalidade do maior conjunto antirradon $R \subseteq V(G)$ acrescida de uma unidade, ou seja, $r(G) = \max\{|R| \mid R \text{ é antirradon}\} + 1$ [15]. A Figura 2.9 mostra um grafo em que estão representados um conjunto antirradon máximo e uma partição de Radon de um conjunto de vértices, que podem ser vistos, respectivamente, nas Figuras 2.9a e 2.9b.



(a) Um conjunto antirradon máximo do grafo. O número de Radon é, portanto, $r(G) = 7$.



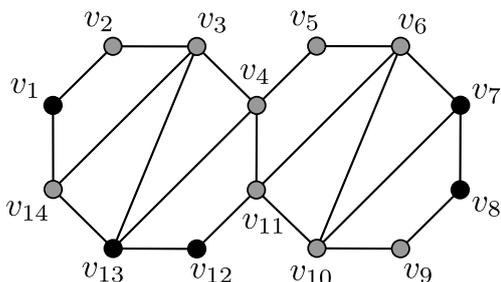
(b) Como $v_4 \in H(\{v_3, v_{13}\}) \cap H(\{v_5, v_{11}\})$, a figura representa uma partição de Radon do conjunto $S = \{v_3, v_5, v_{11}, v_{13}\}$.

Figura 2.9: A Figura 2.9a mostra um conjunto antirradon máximo do grafo acima. Note que, apesar de ter cardinalidade menor do que o número de Radon, o conjunto $S = \{v_3, v_5, v_{11}, v_{13}\}$ apresenta uma partição de Radon, como mostra a Figura 2.9b.

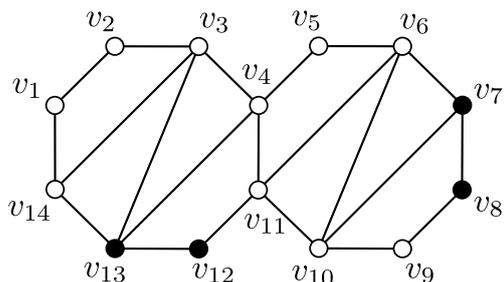
Número de Carathéodory

Seja $G = (V, E)$ um grafo qualquer. Se, para todo conjunto $S \subseteq V$, e todo elemento $v \in H(S)$, existe $F \subseteq S$ tal que $v \in H(F)$ e $|F| \leq k$, dizemos que o menor valor possível para k é o **número de Carathéodory** de G . Esse conceito é inspirado no teorema de Carathéodory [4], que diz que todo ponto da envoltória convexa de um conjunto de pontos S no \mathbb{R}^d está também na envoltória de um subconjunto de S de ordem não superior a $d + 1$.

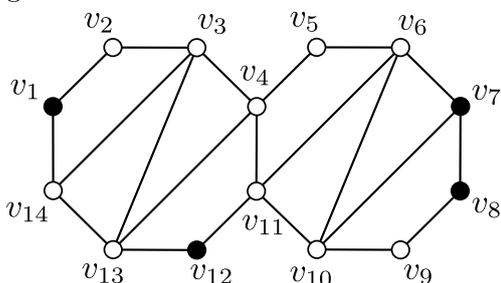
A Figura 2.10 mostra um exemplo em que o número de Carathéodory do grafo na convexidade P_3 é ao menos cinco, pois o conjunto $S = \{v_1, v_7, v_8, v_{12}, v_{13}\}$ é um conjunto de envoltória, mas nenhum de um de seus subconjuntos de cardinalidade quatro é capaz de gerar o vértice v_9 . Uma vez que, $A \subseteq B$ implica $H(A) \subseteq H(B)$ nesta convexidade, nenhum subconjunto próprio de S é capaz de gerar v_9 e, portanto, o número de Carathéodory do grafo é ao menos cinco.



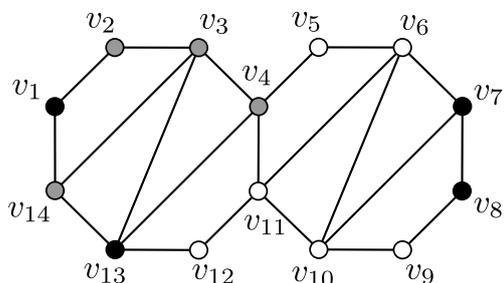
(a) Escolhe-se um conjunto $S = \{v_1, v_7, v_8, v_{12}, v_{13}\}$, e forma-se sua envoltória convexa $H(S)$, que nesse caso, inclui todos os vértices do grafo.



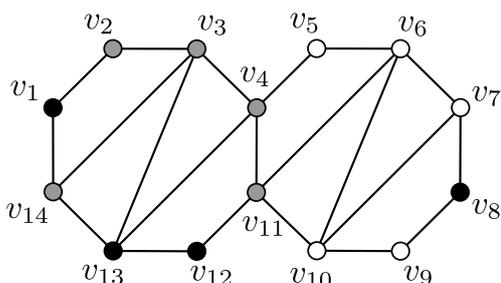
(b) Removendo-se v_1 do conjunto S , a envoltória convexa dos vértices restantes não gera nenhum outro nó do grafo.



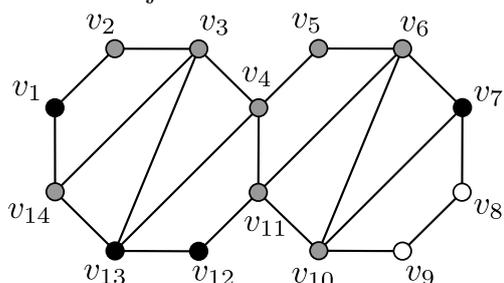
(c) Ao remover o vértice v_{13} de S , a situação é semelhante à da Figura 2.10b.



(d) Se o vértice removido de S é v_{12} , a envoltória convexa passa a incluir alguns vértices de G , mas v_5, v_6, v_9, v_{10} e v_{11} , que fazem parte de $H(S)$, ainda não foram gerados por um subconjunto de S .



(e) Se v_7 é removido de S , a envoltória convexa alcança v_{11} , mas v_5, v_6, v_9 e v_{10} continuam sem serem gerados.



(f) Finalmente, removendo-se v_8 de S , todos os vértices do grafo à exceção de v_9 foram gerados por algum subconjunto de S .

Figura 2.10: A Figura 2.10a mostra um conjunto $S = \{v_1, v_7, v_8, v_{12}, v_{13}\}$, cuja envoltória convexa é igual a $V(G)$. Nas Figuras 2.10b a 2.10f, formam-se todos os subconjuntos de S com quatro vértices e são mostradas as respectivas envoltórias convexas. Como v_9 não faz parte da envoltória convexa de nenhum subconjunto de S , mas está em $H(S)$, o número de Carathéodory de G é, no mínimo, cinco.

2.1.4 Posto

Considere novamente o problema em que desejamos divulgar um anúncio em uma rede social. Muitas vezes nós não temos o direito de escolher exatamente quais usuários farão parte do grupo que receberá o vídeo inicialmente, sendo possível apenas determinar a quantas pessoas a campanha será divulgada nesse primeiro momento. Isso significa que pode não ser uma opção tão eficiente utilizar o número de envoltória do grafo de usuários, visto que tal número pode ser pequeno e o conjunto de envoltória muito específico, tornando a divulgação pouco eficiente.

Uma possível solução para isso é escolher uma quantidade de pessoas maior ou igual à cardinalidade do maior conjunto de envoltória minimal do grafo. Dessa forma, aumenta-se as chances de que o alcance da campanha seja máximo. Para isso, podemos usar o posto do grafo.

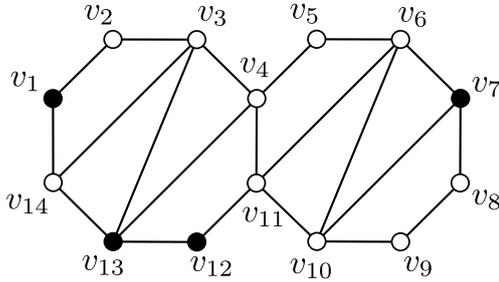
Conjuntos convexamente independentes e o posto de um grafo

No exemplo acima, estamos tentando selecionar um grupo de pessoas tais que, se uma delas for retirada do conjunto, não será alcançada pela propaganda que desejamos divulgar. Esta ideia está intimamente relacionada com o conceito de um conjunto convexamente independente. Dado um grafo $G = (V, E)$, diz-se que $S \subseteq V$ é **convexamente independente** se, para todo $v \in S$, $v \notin H(S - v)$. Caso essa condição não seja satisfeita para algum vértice de S , dizemos que trata-se de um **conjunto convexamente dependente**. A cardinalidade do maior conjunto convexamente independente de G é o **posto** desse grafo, que denotamos por $rk(G)$.

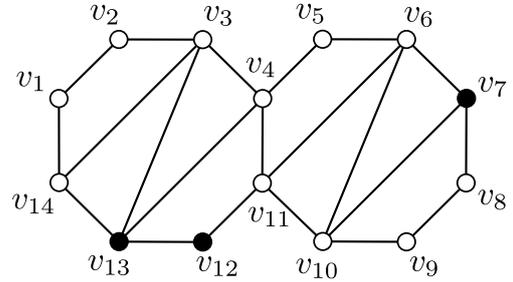
A Figura 2.11 mostra um exemplo de conjunto convexamente independente S de um grafo. Para provar isso, remove-se cada um dos vértices do conjunto com a finalidade de verificar se o elemento retirado de S é gerado pelos demais. No caso, pode-se observar que isso não é verdade para nenhum vértice e, portanto, S é convexamente independente.

Já na Figura 2.12, há um exemplo de conjunto convexamente dependente. Confirmar que um conjunto é convexamente dependente é mais simples do que o caso anterior, pois basta mostrar que um vértice v é gerado por $S - v$, não sendo necessário exibir todos os subconjuntos obtidos pela retirada de um elemento de S . Na figura, pode-se notar que o vértice v_4 é gerado por v_1 e v_{13} , o que permite concluir, portanto, que o conjunto representado é convexamente dependente.

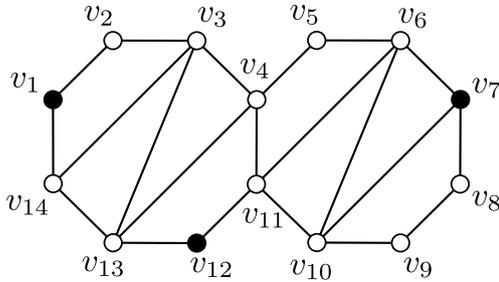
Note que todo conjunto de envoltória minimal é também convexamente independente, o que pode ser provado de forma simples por contradição. Suponha que $S \subseteq V(G)$ é um conjunto de envoltória minimal e convexamente dependente. Isso significa que existe $v \in S$ tal que $v \in H(S - v)$. $S - v$, porém, também é um conjunto de envoltória, visto que $S \subseteq H(S - v)$ e, portanto, S não poderia ser con-



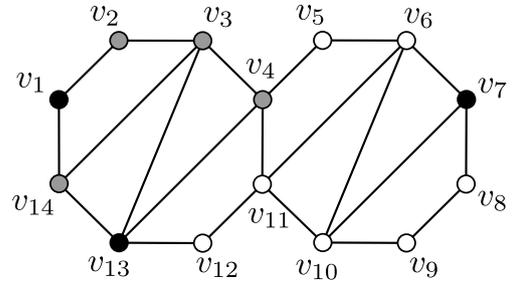
(a) Os vértices pretos formam o conjunto S .



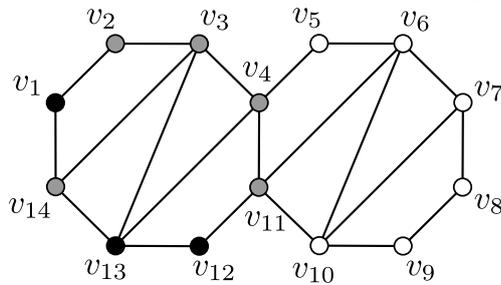
(b) Se o vértice v_1 é removido de S , nenhum outro é gerado pelos restantes. Dessa forma, $v_1 \notin H(S - v_1)$.



(c) Assim como no caso de v_1 , $H(S - v_{13}) = S - v_{13}$ e, portanto, v_{13} não é gerado pelos demais.

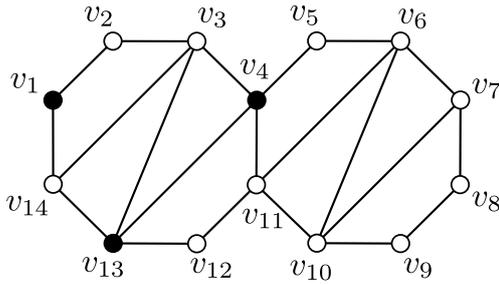


(d) Se v_{12} é removido de S , apesar de alguns vértices de G serem gerados, $v_{12} \notin H(S - v_{12})$.

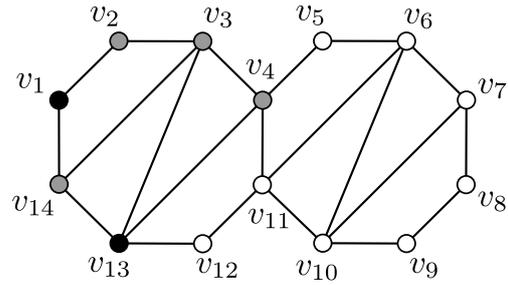


(e) A remoção de v_7 do conjunto S também gera alguns vértices, porém v_7 não é um deles.

Figura 2.11: O conjunto $S = \{v_1, v_7, v_{12}, v_{13}\}$ é convexamente independente, conforme mostram as Figuras 2.11b a 2.11e. Em cada uma delas, um dos vértices de S é removido do conjunto e computa-se a envoltória convexa do restante. Em nenhum dos casos o elemento removido é gerado pelos demais, o que prova que S é convexamente independente.



(a) Os vértices pretos formam o conjunto S .



(b) Se v_4 é removido de S , a envoltória convexa de $S - v_4$ ainda contém esse vértice.

Figura 2.12: Considerando-se o conjunto $S = \{v_1, v_4, v_{13}\}$, pode-se notar na Figura 2.12b que $v_4 \in H(S - v_4)$, o que significa que S é convexamente dependente.

junto de envoltória minimal. Isso significa que o posto é um limite superior para a cardinalidade de um conjunto de envoltória minimal de um grafo e, portanto, pode ser útil para a solução do problema proposto.

Assim como o número de envoltória, o número de Radon e o número de Carathéodory, o posto é um dos parâmetros clássicos da convexidade de grafos [38]. Apesar disso, até o momento, a literatura a respeito do posto de um grafo é bastante escassa.

2.2 Formalização do problema

“Out, damned spot; out, I say.”

William Shakespeare, Macbeth

Uma vez definidos os conceitos de conjunto convexamente independente e de posto de um grafo, é possível, afinal, formular o problema de decisão que é o principal objeto de estudo desta dissertação.

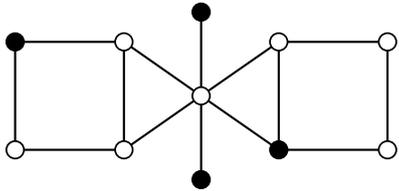
CONJUNTO CONVEXAMENTE INDEPENDENTE

Entrada: Um grafo G e um inteiro k .

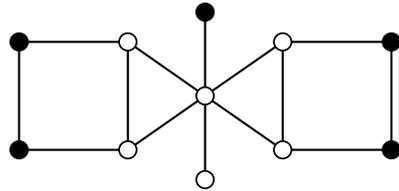
Pergunta: Existe conjunto convexamente independente S com $|S| \geq k$?

Note que todo conjunto com até dois vértices é convexamente independente em qualquer convexidade de caminhos, visto que não é possível que apenas um vértice gere quaisquer outros. Dessa forma, se G tem ao menos dois vértices e $k \leq 2$, a resposta para o problema acima é trivialmente sim. Todos os resultados do próximo capítulo, portanto, restringem-se ao caso em que $k \geq 3$ e G tem ao menos três vértices.

A princípio, todo vértice de grau um é forte candidato a fazer parte de um conjunto convexamente independente máximo. Essa suposição é fundamentada pelo fato de que esses vértices, por terem apenas um vizinho, não podem ser gerados por outros. Na Figura 2.13, porém, há um exemplo, na convexidade P_3 , que mostra que nem sempre o conjunto convexamente independente máximo conterá todos esses vértices.



(a) Conjunto convexamente independente maximal incluindo ambos os vértices de grau um.



(b) Um conjunto convexamente independente máximo do grafo.

Figura 2.13: Na Figura 2.13a, a inclusão de ambos os vértices de grau um no conjunto convexamente independente faz com que não se possa ter mais do que quatro nós no conjunto. Já na Figura 2.13b, ao deixar um desses vértices de grau um de fora, o nó central não é gerado e torna-se possível formar um conjunto convexamente independente de cardinalidade cinco, que é o posto do grafo.

O exemplo da Figura 2.13 mostra, portanto, que o problema do CONJUNTO CONVEXAMENTE INDEPENDENTE pode ser relativamente difícil de ser resolvido.

2.3 Problemas relacionados

“A horse! A horse! My kingdom
for a horse!”

William Shakespeare, Richard III

Alguns problemas relacionam-se com o CONJUNTO CONVEXAMENTE INDEPENDENTE, seja por auxiliarem nas provas de \mathcal{NP} -completude do Capítulo 3 ou por ser possível obter algum tipo de resultado derivado do posto. A seguir, é feita uma breve descrição de cada um deles.

2.3.1 Empacotamento de conjuntos

Dada uma família de subconjuntos $\mathcal{F} = \{S_1, \dots, S_n\}$, um **empacotamento** é uma subfamília $\mathcal{E} \subseteq \mathcal{F}$ tal que qualquer par de conjuntos $S_i, S_j \in \mathcal{E}$ satisfaz $S_i \parallel S_j$. Um dos 21 problemas \mathcal{NP} -completos de Karp [28], o problema do EMPACOTAMENTO DE CONJUNTOS consiste em determinar o tamanho de um empacotamento máximo do grafo. A descrição formal do problema, em sua versão de decisão, é dada a seguir.

aberto, denotado por $\rho_o(G)$. O problema do EMPACOTAMENTO ABERTO é difícil, tendo sido provada a sua \mathcal{NP} -completude para grafos cordais.

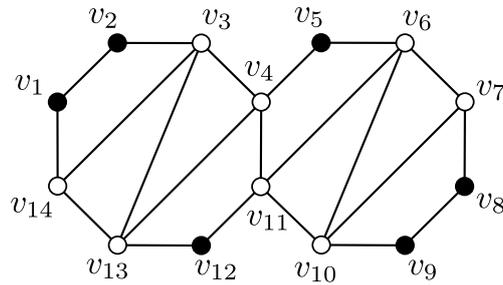


Figura 2.15: Um empacotamento aberto do grafo.

A Figura 2.15 mostra um exemplo de empacotamento aberto do grafo. Isso significa que o número de empacotamento aberto satisfaz a inequação $\rho_o(G) \geq 6$.

Todo empacotamento aberto é um conjunto convexamente independente na convexidade P_3 e, portanto, $\rho_o(G) \leq rk(G)$. Em alguns casos, conforme será mostrado no próximo capítulo, a recíproca é verdadeira e, para esses grafos, $rk(G) = \rho_o(G)$.

Note que, no grafo da Figura 2.15, o posto é seis se consideramos a convexidade P_3 . Dessa forma, pode-se ter certeza de que o número de empacotamento aberto do grafo é, também, seis.

A descrição formal do problema do EMPACOTAMENTO ABERTO em sua versão de decisão é dada abaixo.

EMPACOTAMENTO ABERTO

Entrada: Um grafo G e um inteiro k .

Pergunta: $\rho_o(G) \geq k$?

Capítulo 3

Resultados

“There is nothing like looking, if you want to find something. You certainly usually find something, if you look, but it is not always quite the something you were after.”

J.R.R. Tolkien, *The Hobbit*

Este capítulo apresenta os resultados obtidos a respeito do problema proposto no Capítulo 2. Primeiro, mostra-se que o problema do CONJUNTO CONVEXAMENTE INDEPENDENTE MÁXIMO é \mathcal{NP} -completo nas convexidades P_3 e monofônica. Na Seção 3.2 mostra-se alguns casos em que o problema pode ser resolvido em tempo polinomial. O capítulo é encerrado, afinal, com um limite superior geral para o posto de um grafo.

3.1 \mathcal{NP} -completude

“Get your facts first, then you can distort them as you please.”

Mark Twain

O problema do CONJUNTO CONVEXAMENTE INDEPENDENTE mostra-se computacionalmente difícil nas convexidades P_3 e monofônica, mesmo para classes de grafos bastante simples. Esta seção apresenta alguns teoremas de \mathcal{NP} -completude para o problema.

3.1.1 Grafos *split* na convexidade P_3

O problema do CONJUNTO CONVEXAMENTE INDEPENDENTE mostra-se difícil para grafos *split* na convexidade P_3 , apesar da relativa simplicidade dessa classe.

Para provar este resultado, é conveniente apresentar alguns lemas sobre características de conjuntos e envoltórias convexas nesses grafos.

Lema 3.1 *Se G é um grafo split, $C \subseteq V(G)$ é uma clique e v_1 e v_2 são vértices distintos de C , então $C \subseteq H(\{v_1, v_2\})$.*

Em outras palavras, o Lema 3.1 significa que quaisquer dois vértices distintos v_1 e v_2 de uma clique C geram todos os demais nós de C . Este resultado pode ser verificado de forma simples. É preciso apenas observar que todos os demais vértices da clique são vizinhos de v_1 e v_2 e, dessa forma, qualquer conjunto convexamente independente que contenha v_1 e v_2 precisa também incluir os demais vértices de C . Com isso, é possível enunciar mais um lema.

Lema 3.2 *Seja G um grafo split com $\delta(G) \geq 2$ e partição split $V(G) = C \cup I$ em que C é uma clique e I é um conjunto independente. Se $S \subseteq V(G)$ é convexamente independente e tem ao menos três vértices, então $S \subseteq I$.*

Prova: A prova deste lema se dá por contradição. Suponha que $S \subseteq V(G)$ é um conjunto convexamente independente de G , mas que $S \not\subseteq I$. Uma vez que $\delta(G) \geq 2$, todo vértice é adjacente a, no mínimo, dois elementos de C . Isso significa que, pelo Lema 3.1, se há dois vértices de C em S , então todo o grafo é gerado, inclusive os demais nós do conjunto S , que não poderia, portanto, ser convexamente independente. Resta, portanto, a possibilidade de haver exatamente um ou nenhum vértice da clique C no conjunto S .

Considere agora o caso em que há exatamente um vértice $v \in C \cap S$. Seja $u \in I$ um segundo vértice qualquer de S . Como $\delta(G) \geq 2$, mesmo que $uv \in E(G)$, ainda é preciso que exista outra aresta entre u e outro vértice $w \in C$. Dessa forma, $w \in H(S)$ e, com isso, toda a clique C é gerada por v e w . Como consequência, $H(\{u, v\}) = V(G)$ e, portanto, qualquer outro vértice do conjunto S o tornaria convexamente dependente. Conclui-se, então, que, para termos S convexamente independente com $|S| \geq 3$, é preciso que $S \subseteq I$. ■

Lema 3.3 *Seja G um grafo split com $\delta(G) \geq 2$ e partição split $V(G) = C \cup I$ em que C é uma clique e I é um conjunto independente. $S \subsetneq V(G)$ tal que $|S| \geq 3$ é convexamente independente se, e somente se, $H(S) = S$. Além disso, $V(G)$ não é convexamente independente.*

Prova: É fácil ver que $V(G)$ não pode ser convexamente independente, visto que quaisquer dois vértices da clique C geram todos os demais de acordo com o Lema 3.1, o que torna $V(G)$ convexamente dependente. Dito isso, resta provar que S é convexamente independente se, e somente se, $H(S) = S$, o que também será feito por contradição.

Suponha que $S \subsetneq V(G)$ é um conjunto convexamente independente de G e que $H(S) \neq S$. Sabe-se, de acordo com o Lema 3.2, que $S \subseteq I$. Para que algum vértice seja gerado, é preciso que haja $u, v \in S$ tais que u e v têm um vizinho em comum. Como ambos fazem parte do conjunto independente I , qualquer vértice adjacente a eles precisa ser parte de C . Seja $w \in C$ esse vizinho de u e v . Como $\delta(G) \geq 2$, sabemos que u tem mais algum vizinho em C além de w . Como C é uma clique, tal vértice também é adjacente a w e, portanto, também faz parte de $H(S)$, que contém agora dois vértices de C e, pelo Lema 3.1, $C \in H(\{u, v\})$, o que resulta em $H(\{u, v\}) = V(G)$. Como S tem ao menos três vértices, trata-se de um conjunto convexamente dependente. Dessa forma, para que S seja convexamente independente é preciso que $H(S) = S$.

Para a provar que $H(S) = S$ implica S ser convexamente independente, o argumento é semelhante. Suponha que $S \subsetneq V(G)$ é convexamente dependente e que $H(S) = S$. Para que isso seja verdade, é preciso que algum vértice de S tenha dois vizinhos que também pertençam ao conjunto. Isso só pode acontecer se houver algum vértice da clique em S , o que, conforme mostrado acima, faz com que $H(S) = V(G)$. Uma vez que $S \neq V(G)$, a envoltória convexa de S não pode ser igual ao próprio conjunto. Fica provado, portanto, que, se $H(S) = S$, então S é convexamente independente. ■

Finalmente, é possível enunciar e provar a \mathcal{NP} -completude do problema CONJUNTO CONVEXAMENTE INDEPENDENTE para grafos *split* na convexidade P_3 .

Teorema 3.4 *O problema do CONJUNTO CONVEXAMENTE INDEPENDENTE é \mathcal{NP} -completo na convexidade P_3 , mesmo para grafos *split* com $\delta(G) \geq 2$.*

Prova: Uma vez que a envoltória convexa de um subconjunto de vértices pode ser computada em tempo polinomial, pode-se concluir que o problema está em \mathcal{NP} .

A redução a ser apresentada parte do problema do EMPACOTAMENTO DE CONJUNTOS, o qual é sabidamente \mathcal{NP} -completo [28]. Este tem como entrada uma família \mathcal{S}' de subconjuntos $S'_i \in \mathcal{S}$ de um conjunto base $\bigcup S'_i$ e um inteiro k' . A pergunta é se existem, em \mathcal{S}' , k' ou mais subconjuntos disjuntos. A partir dessa entrada, pode-se construir uma instância do problema do CONJUNTO CONVEXAMENTE INDEPENDENTE. É preciso, portanto, construir um grafo G e determinar um inteiro k de modo que a resposta para a pergunta $rk(G) \geq k$ permita responder à questão original do EMPACOTAMENTO DE CONJUNTOS.

Os elementos do conjunto base $\bigcup S'_i$ são vértices de G e há também mais dois vértices distintos, w_i e z_i para cada um dos subconjuntos S'_i . Com isso, $V(G)$ está completo e podemos construir o conjunto das arestas do grafo. Os elementos do conjunto base, junto com todos os vértices z_i formam uma clique C . Cada w_i , por

sua vez, é adjacente a z_i e aos vértices correspondentes aos elementos de S'_i . Por fim, $k = k'$ completa a instância do problema do CONJUNTO CONVEXAMENTE INDEPENDENTE. Note que os vértices w_i formam um conjunto independente I , de modo que G é um grafo *split* com partição $V(G) = C \cup I$. Finalmente, sem perda de generalidade, considere apenas valores de $k' \geq 3$. Um exemplo pode ser visto na figura 3.1.

Para provar a equivalência entre os problemas, é preciso mostrar que G tem um conjunto convexamente independente de tamanho k ou maior se, e somente se, \mathcal{S}' contém ao menos $k' = k$ subconjuntos disjuntos.

Primeiro, suponha que \mathcal{S}' contém k' conjuntos disjuntos $S_1, S_2, \dots, S_{k'}$, ou seja, k conjuntos sem nenhum elemento em comum. Note que, pela construção de G , isso significa que $S = \{w_1, w_2, \dots, w_k\}$, um subconjunto da parte estável do grafo, não contém dois vértices w_i e w_j tais que $N(w_i) \cap N(w_j) \neq \emptyset$. Conclui-se, portanto, que $H(S) = S$ e, pelo Lema 3.3, S é um conjunto convexamente independente, logo G contém um conjunto convexamente independente de cardinalidade maior ou igual k .

Agora suponha que há, em G , um subconjunto de vértices $S \subseteq V(G)$ convexamente independente e tal que $|S| \geq k$. Do Lema 3.2, conclui-se que $S \subseteq I$ e, portanto, podemos dizer, sem perda de generalidade, que $S = \{w_1, w_2, \dots, w_k\}$, visto que remover um vértice de um conjunto convexamente independente não pode torná-lo convexamente dependente. Novamente é possível fazer uso do Lema 3.3 para concluirmos que nenhum vértice é gerado por S e, portanto, isso significa que não há vértice v tal que $vw_i \in E(G)$ e $vw_j \in E(G)$ para $1 \leq i < j \leq k$. Pela construção de G , se w_i e w_j não têm vizinho em comum, então S_i e S_j são disjuntos. Dessa forma, se há ao menos k vértices cujas adjacências não se intersectam, há o mesmo número — maior ou igual a k' — de subconjuntos disjuntos em \mathcal{S}' , o que conclui a prova. ■

Um exemplo da redução apresentada na prova acima é dado na Figura 3.1. O grafo é montado a partir da família de conjuntos $\mathcal{S} = \{S_1 = \{1, 6\}, S_2 = \{1, 2\}, S_3 = \{2, 3\}, S_4 = \{3, 4\}, S_5 = \{4, 5\}, S_6 = \{5, 6\}\}$ e o conjunto convexamente independente máximo com três vértices representa um empacotamento de conjuntos máximo da família \mathcal{S} , que tem tamanho três e é composto por S_1, S_3 e S_5 . Cabe observar que, caso todos os conjuntos da família tenham ao menos três elementos, fica caracterizada também a \mathcal{NP} -completude para grafos 4-*split*.

Do Teorema 3.4, aliado ao Lema 3.3, decorre naturalmente um corolário.

Corolário 3.5 *O problema do EMPACOTAMENTO ABERTO é \mathcal{NP} -completo para grafos *split* com $\delta(G) \geq 2$.*

Prova: O Lema 3.3 implica que um conjunto de ao menos três vértices é convexamente independente em um grafo *split* se, e somente se, também é um empa-

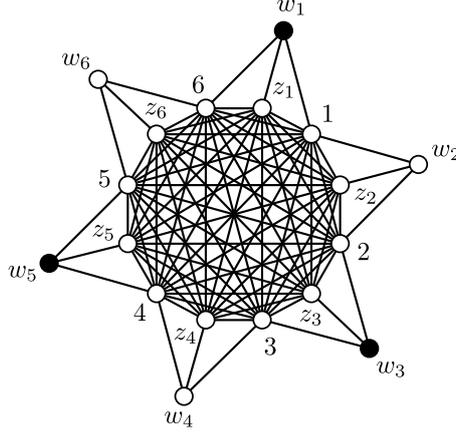


Figura 3.1: Exemplo da redução do Teorema 3.4. Dada a família de conjuntos $\mathcal{S} = \{S_1 = \{1, 6\}, S_2 = \{1, 2\}, S_3 = \{2, 3\}, S_4 = \{3, 4\}, S_5 = \{4, 5\}, S_6 = \{5, 6\}\}$, obtém-se o grafo da figura.

cotamento aberto. Isso significa que, para esta classe de grafos, os problemas do CONJUNTO CONVEXAMENTE INDEPENDENTE e do EMPACOTAMENTO ABERTO são equivalentes e, dessa forma, se o primeiro é \mathcal{NP} -completo, o último também o é. ■

3.1.2 Grafos bipartidos na convexidade P_3

Assim como no caso dos grafos *split*, o problema do CONJUNTO CONVEXAMENTE INDEPENDENTE também é computacionalmente difícil para grafos bipartidos na convexidade P_3 , como será mostrado a seguir.

Teorema 3.6 *O problema do CONJUNTO CONVEXAMENTE INDEPENDENTE é \mathcal{NP} -completo na convexidade P_3 , mesmo para grafos bipartidos com diâmetro máximo três.*

Prova: Conforme dito na prova do Teorema 3.4, o problema está em \mathcal{NP} . A redução que será usada para mostrar a \mathcal{NP} -completude neste caso parte do próprio problema do CONJUNTO CONVEXAMENTE INDEPENDENTE para grafos *split*, que é \mathcal{NP} -completo conforme provado no Teorema 3.4.

Sejam G' um grafo *split* com $\delta(G') \geq 2$ e k' um inteiro que compõem uma instância do CONJUNTO CONVEXAMENTE INDEPENDENTE. Para provar a \mathcal{NP} -completude, é preciso encontrar uma instância do problema que tenha como entrada um grafo bipartido G , com diâmetro máximo três, e um inteiro k de modo que a resposta desta última leve à solução para a primeira.

Para construir o grafo G , considere a partição *split* de G' em $V(G') = C' \cup I'$ em que C' é uma clique e I' é estável. Todos os vértices de G' são também vértices de G . Além desses, há também três vértices distintos $x_1, x_2, y \in V(G)$. As arestas do grafo

são construídas de modo a formar-se uma bipartição dos vértices $V(G) = C \cup I$ em que $C = C' \cup \{y\}$ e $I = I' \cup \{x_1, x_2\}$. Sejam $u, v \in V(G')$. $uv \in E(G)$ se, e somente se, $uv \in E(G')$ e $u \in C'$ e $v \in I'$. Em outras palavras, na construção de G , as adjacências de G' são mantidas entre vértices da clique e do conjunto independente, enquanto as arestas entre vértices de C' são descartadas. Há também arestas de x_1 e x_2 para todos os vértices de C , assim como de y para todos os vértices de I . Por conveniência, x_1 , x_2 e y serão chamados de vértices universais, pois são vizinhos de todos os vértices possíveis no grafo. Para completar a instância do problema, faça $k = k'$. Note que o diâmetro de G é, no máximo, três, visto que é sempre possível usar os vértices universais para ligar quaisquer outros. Finalmente, sem perda de generalidade, considere apenas valores de $k' \geq 4$. Um exemplo pode ser visto na figura 3.2.

É preciso mostrar que existe $S' \subseteq V(G')$, convexamente independente e com cardinalidade maior ou igual a k' , se, e somente se, há $S \subseteq V(G)$, convexamente independente e com de cardinalidade maior ou igual a k .

Primeiro, considere um conjunto convexamente independente $S' \subseteq V(G)$ tal que $|S'| \geq k'$. Pelo Lema 3.2, pode-se concluir que $S' \subseteq I'$ e o Lema 3.3 implica que $H(S') = S'$, ou seja, S' não gera nenhum vértice de G' . Seja $S = S'$. Isso significa que $S \subseteq I$ e $|S| \geq k$. Uma vez que a única diferença entre adjacências de I' e de I é a existência do vértice universal y , é fácil ver que apenas y será gerado por S . O conjunto S é, portanto, convexamente independente, visto que, para que algum vértice de S pudesse ser gerado pelos demais, seria preciso que ao menos dois de seus vizinhos — necessariamente do conjunto C — fossem adjacentes a pelo menos dois vértices de S , o que sabemos não ser verdade. Uma vez que S é um conjunto convexamente independente de G e tem cardinalidade no mínimo k , está concluída essa parte da prova.

Agora suponha que $S \subseteq V(G)$ é um conjunto convexamente independente tal que $|S| \geq k$. Para mostrar que G' tem um conjunto convexamente independente de cardinalidade no mínimo k' , basta provar que $S \subseteq I$ e que $H(S) = S \cup \{y\}$, pois as adjacências de I' são preservadas na construção de I e não haver outro vértice de C que não y em $H(S)$ implica que $S' = S$ também não gera nenhum vértice de C' e $|S'| \geq k'$. Isso é suficiente de acordo com os Lemas 3.2 e 3.3.

Note que, se $u, v \in C \cap S$, então x_1 e x_2 são gerados. Como estes são adjacentes a todos os vértices de C , isso resulta em $C \subseteq H(\{u, v\})$. Uma vez que as adjacências de I são preservadas em relação a I' , $H(\{u, v\}) = V(G)$ e S é convexamente dependente. Não é possível, portanto, que haja dois ou mais vértices de C no conjunto S . Suponha, então, que há exatamente um vértice $v \in C \cap S$. Se $v = y$, S é convexamente dependente, pois quaisquer dois vértices de I geram y . Por outro lado, $v \neq y$ também não é possível, visto que, ao gerar y , haverá dois vértices de C em $H(S)$, o

que, por sua vez, faz com que $H(S) = V(G)$. Como apenas três vértices de G geram todo o grafo e $k \geq 4$, isso torna S convexamente dependente. Não resta, portanto, outra possibilidade que não $S \subseteq I$. Além disso, se algum vértice $v \in C$, $v \neq y$, tiver mais de um vizinho em S , isso tornaria o conjunto convexamente dependente, pois v e y gerariam todos os demais vértices de G . Dessa forma, conclui-se que $S \subseteq I$ e $H(S) = S \cup \{y\}$ e, portanto, o problema do CONJUNTO CONVEXAMENTE INDEPENDENTE é \mathcal{NP} -completo para grafos bipartidos com diâmetro máximo três.

■

Tomando como referência o grafo da Figura 3.1 e aplicando-se a redução acima, o resultado obtido pode ser visto na figura 3.2.

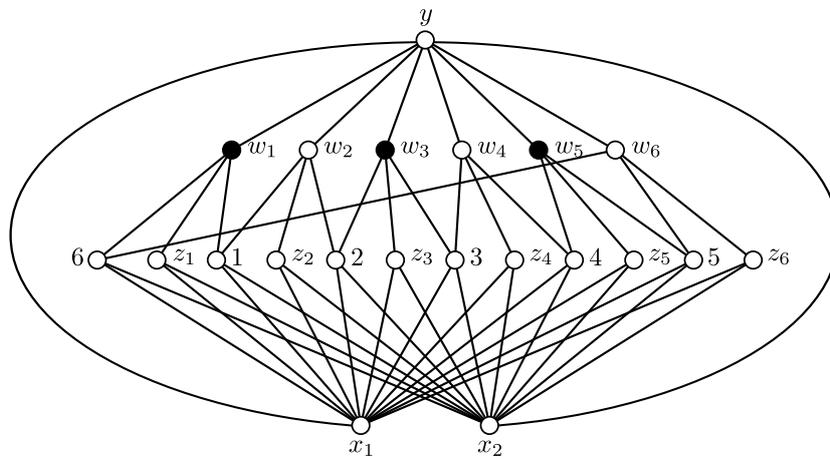


Figura 3.2: Exemplo da redução do Teorema 3.6. A partir do grafo *split* da figura 3.1, encontra-se o grafo acima.

3.1.3 Átomos na convexidade monofônica

A seguir, será mostrado que, assim como na convexidade P_3 , o problema do CONJUNTO CONVEXAMENTE INDEPENDENTE também é \mathcal{NP} -completo na convexidade monofônica. Para essa prova, é empregado o resultado a seguir, apresentado por Dourado et al [14].

Teorema 3.7 *Se G é um átomo, mas não um grafo completo, então todo par de vértices não adjacentes é um conjunto de envoltória de G na convexidade monofônica.*

Além disso, é preciso também mostrar que o problema da CLIQUE MÁXIMA, sabidamente \mathcal{NP} -completo para grafos gerais, também o é para grafos sem clique separadora.

Lema 3.8 *O problema da CLIQUE MÁXIMA é \mathcal{NP} -completo, mesmo para grafos sem clique separadora.*

Prova: Uma vez que o problema é \mathcal{NP} -completo para grafos gerais, é certo que ele está em \mathcal{NP} . A redução a ser apresentada parte do próprio problema da CLIQUE MÁXIMA para grafos gerais.

Considere um grafo conexo G' a ser usado como instância para o problema da CLIQUE MÁXIMA para grafos gerais. Se G' é um átomo, não há nada a provar, então devemos assumir que o grafo tem uma clique separadora. Isso implica $|V(G')| \geq 3$. Constrói-se um átomo G fazendo a união disjunta entre G' e $C_{|V(G')|}$ seguida da adição de arestas de modo que cada vértice de G' seja adjacente a exatamente um outro de $C_{|V(G')|}$ e cada vértice de $C_{|V(G')|}$ tenha exatamente um vizinho de G' .

Note que nenhuma clique separadora de G' pode desconectar G , pois é sempre possível formar um caminho pelos vértices de $C_{|V(G')|}$. Por outro lado, nenhum vértice de $C_{|V(G')|}$ pode fazer parte de uma clique separadora, uma vez que todos têm apenas um vizinho dentre os vértices de G' . Ademais, nenhuma clique de G é maior do que a clique máxima de G' , pois são adicionadas apenas cliques de tamanho dois e nenhuma das que já existiam pode aumentar pelo simples acréscimo de vértices com apenas um vizinho de G' . Isso significa que a maior clique de G tem o mesmo tamanho da maior clique de G' e, dessa forma, sendo G um átomo, a prova está concluída. ■

É possível, agora, afirmar e provar a \mathcal{NP} -completude do problema.

Teorema 3.9 *O problema do CONJUNTO CONVEXAMENTE INDEPENDENTE MÁXIMO é \mathcal{NP} -completo na convexidade monofônica, mesmo para átomos.*

Prova: Como o cálculo da envoltória convexa de um subconjunto de vértices de um grafo pode ser feito em tempo polinomial na convexidade monofônica, o problema do CONJUNTO CONVEXAMENTE INDEPENDENTE está em \mathcal{NP} . Para mostrar a \mathcal{NP} -completude, é feita uma redução a partir do problema da CLIQUE MÁXIMA para átomos, que é \mathcal{NP} -completo de acordo com o Lema 3.8.

Seja G um grafo sem clique separadora. De acordo com o Teorema 3.7, quaisquer dois vértices não adjacentes geram todo o grafo e, com isso, nenhum conjunto convexamente independente com cardinalidade maior do que dois pode conter um par de vértices não adjacentes. Isso significa que um conjunto convexamente independente de tamanho ao menos três deve ser necessariamente uma clique. Por outro lado, toda clique é um conjunto convexamente independente na convexidade monofônica, pois o único caminho induzido entre dois de seus vértices é a própria aresta que os liga e nenhum outro vértice é gerado. Fica claro, dessa forma, que o tamanho do maior conjunto convexamente independente de um átomo na convexidade monofônica coincide com o da maior de suas cliques e, portanto, descobrir o primeiro permite conhecer de imediato o segundo. Conclui-se, então, que o problema

do CONJUNTO CONVEXAMENTE INDEPENDENTE é \mathcal{NP} -completo para átomos na convexidade monofônica. ■

3.2 Algoritmos polinomiais

“If the facts don’t fit the theory,
change the facts.”

Albert Einstein

A Seção 3.1 mostra que o problema do CONJUNTO CONVEXAMENTE INDEPENDENTE é computacionalmente difícil nas convexidades P_3 e monofônica, mesmo quando restrito a algumas classes de grafos. A seguir, são apresentadas algumas outras classes para as quais é possível resolver o problema em tempo polinomial.

3.2.1 Grafos *threshold* na convexidade P_3

Uma vez que o problema do CONJUNTO CONVEXAMENTE INDEPENDENTE é \mathcal{NP} -completo para grafos *split*, pode-se questionar se ele pode ser resolvido eficientemente para alguma das subclasses desses grafos. No caso dos grafos *threshold*, há um algoritmo polinomial para o cálculo do posto, derivado diretamente do teorema abaixo.

Teorema 3.10 *Se G é um grafo threshold conexo com ao menos três vértices e $D \subseteq V(G)$ é o conjunto com todos os vértices $v \in V(G)$ tal que $d(v) = \delta(G)$, então:*

- (i) *se G é uma estrela, então $rk(G) = |V(G)| - 1$;*
- (ii) *senão, se $\delta(G) = 1$, então $rk(G) = |D| + 1$;*
- (iii) *senão, $rk(G) = 2$.*

Prova: Em (i) fica claro que o vértice central é gerado pelos demais, mas, por terem apenas um vizinho, as folhas não podem ser geradas. O conjunto convexamente independente máximo é composto, portanto, por todos os vértices de grau um.

No caso (iii), $\delta(G) \geq 2$ e, uma vez que as vizinhanças são aninhadas e todo grafo *threshold* também é *split*, quaisquer dois vértices geram todo o grafo, ao adicionarem ao menos dois vértices da clique à envoltória convexa. Dessa forma, a cardinalidade máxima de um conjunto convexamente independente é dois.

Finalmente, no caso (ii), o conjunto D é convexamente independente, pois, por terem grau um, não podem ser gerados. Como as vizinhanças são aninhadas, todos os vértices de D têm que ter um mesmo vizinho $v \in V(G) \setminus D$ e, portanto, $H(D) =$

$D \cup \{v\}$. É possível, portanto, escolher mais um vértice $u \in V(G) \setminus D$, $u \neq v$, de modo que $D \cup \{u\}$ é convexamente independente. Ainda, $D \cup \{u\}$ é máximo, visto que quaisquer dois vértices $u, v \in V(G) \setminus D$ fazem com que $H(\{u, v\}) = V(G) \setminus D$, não sendo possível acrescentar mais nenhum vértice ao conjunto $\{u, v\}$ sem torná-lo convexamente dependente, conforme o argumento do caso (iii) acima. ■

Do Teorema 3.10 decorre diretamente um algoritmo polinomial para o cálculo do posto de um grafo *threshold* G . Uma vez que se pode determinar os graus de todos os vértices de G em tempo $O(m)$ e que isso torna possível computar $|D|$ em tempo $O(n)$, é possível resolver o problema do CONJUNTO CONVEXAMENTE INDEPENDENTE em tempo $O(n + m)$ para grafos *threshold*.

3.2.2 Árvores na convexidade P_3

Assim como a \mathcal{NP} -completude do problema do CONJUNTO CONVEXAMENTE INDEPENDENTE para grafos *split* na convexidade P_3 , a dificuldade desse problema para grafos bipartidos também levanta a questão da complexidade de tempo para suas subclasses. Tomando a ideia do cálculo do número de Radon para árvores por Dourado et al [16], pode-se chegar a um algoritmo polinomial para o cálculo do posto para essa classe de grafos.

Seja T uma árvore enraizada em um de seus vértices $r \in V(T)$. A subárvore com raiz em um vértice $v \in V(T)$ chama-se T_v , sendo $T_r = T$. Um conjunto convexamente independente máximo $S^* \subseteq V(T)$ pode ser encontrado em tempo polinomial por meio de um algoritmo de programação dinâmica.

É preciso definir primeiro o conceito de **carga**, que será usado como base para a relação de recorrência. Um vértice $u \in V(T)$ envia uma unidade de carga para $v \in V(T)$ se, e somente se, $u \in H_{T-v}(S-v)$ e $v \in N(u)$. Em outras palavras, apesar de u e v serem vizinhos, u não depende de v para estar na envoltória de S . O total de cargas que v recebe com relação ao conjunto S é representado por $ch(v)$ e é igual a $|N(v) \cap H_{T-v}(S-v)|$.

Lema 3.11 *Seja $S \subseteq V(T)$ um conjunto convexamente independente de T . $v \in V(T)$ é gerado por $S - v$ se, e somente se, $ch(v) \geq 2$.*

Prova: Se $ch(v) \geq 2$, então v tem ao menos dois vizinhos u_1 e u_2 tais que $u_1, u_2 \in H_{T-v}(S-v)$. Como adicionar um vértice ao grafo não pode remover nenhum outro da envoltória convexa de um conjunto na convexidade P_3 , é possível reinserir v na árvore e será verdade que $u_1, u_2 \in H_T(S-v)$. Como tem dois vizinhos na envoltória convexa, o vértice v é gerado por $S - v$.

Da mesma forma, se v é gerado por $S - v$, então há $u_1, u_2 \in N(v)$ de modo que $u_1, u_2 \in H_T(S-v)$ e isso não depende de v já estar em $H_T(S-v)$. Se, ao remover

v da árvore, u_i não estiver em $H_{T-v}(S - v)$, então u_i tem exatamente dois vizinhos em T que também estão em $H_T(S - v)$ e v é obrigatoriamente um deles, o que, por sua vez, é uma contradição. Dessa forma, sabe-se que $u_1, u_2 \in H_{T-v}(S - v)$ e, uma vez que $u_1, u_2 \in N(v)$, $ch(v) \geq 2$, o que completa a prova. ■

O corolário a seguir é uma consequência imediata do Lema 3.11.

Corolário 3.12 $S \subseteq V(T)$ é convexamente independente se, e somente se, não existe $v \in S$ tal que $ch(v) \geq 2$.

Para descrever o algoritmo, é preciso antes introduzir uma relação de recorrência. Será chamado de **contribuição** de v o valor de $P_v(i, j, k)$, correspondente ao maior conjunto convexamente independente que contém apenas vértices da subárvore enraizada em v nas condições determinadas por i, j e k , conforme a descrição abaixo. Caso $P_v(i, j, k)$ não esteja bem-definido, atribui-se o valor $-\infty$ à sua contribuição para que não seja possível que este atrapalhe outros cálculos.

- $i = 1$ significa que v recebe uma unidade de carga de seu pai, enquanto $i = 0$ indica o contrário.
- $j = 1$ significa que v é parte do conjunto convexamente independente sendo considerado no momento, enquanto $j = 0$ implica o oposto.
- k é a quantidade de carga que v recebe de seus filhos.

Note que, ao calcularmos $P_v(i, j, k)$, $ch(v) = i + k$.

Se p_v representa o pai do vértice v e definindo $N'(v) = N(v) - p_v$, considere as funções a seguir:

$$f(v, i) = \max\{P_v(i, 0, 0), P_v(i, 0, 1)\} \quad (3.1)$$

$$h(v, i) = \max\left\{\max_{2 \leq k < d(v)} \{P_v(i, 0, k)\}, \max_{0 \leq k \leq d(v)} \{P_v(i, 1, k)\}\right\} \quad (3.2)$$

$$g(v, i_1, i_2) = h(v, i_1) - f(v, i_2) \quad (3.3)$$

Uma vez que, na Equação 3.1, $ch(v) < 2$ ou, do contrário, $ch(v) = 2$ e v depende de p_v para estar em $H(S)$, é possível ter certeza de que v não envia carga para seu pai. Se $j = 1$ ou $k \geq 2$, v sempre estará em $H(S)$, independentemente da carga que possa vir de p_v . A função $f(v, i)$, então, determina a maior contribuição possível de v sem que este envie carga a seu pai.

A função $h(v, i)$, na Equação 3.2, leva em consideração todas as possibilidades de j e k deixadas de lado por f , representando a contribuição máxima de v quando ele deve enviar uma unidade de carga para p_v .

Finalmente, a função $g(v, i_1, i_2)$, na Equação 3.3, é o ganho — em termos de contribuição — obtido por um vértice v , que não está enviando carga para seu pai, ao passar a fazê-lo.

É possível, agora, determinar o valor de $P_v(i, j, k)$ para qualquer vértice $v \in V(T)$.

$$P_v(0, 0, 0) = \sum_{u \in N'(v)} f(u, 0); \quad (3.4)$$

$$P_v(0, 0, 1) = \begin{cases} -\infty, & \text{se } v \text{ não tem filhos,} \\ \sum_{u \in N'(v)} f(u, 0) + \max_{u \in N'(v)} g(u, 0, 0), & \text{caso contrário;} \end{cases} \quad (3.5)$$

$$P_v(0, 0, 2) = \begin{cases} -\infty, & \text{se } v \text{ tem menos de 2 filhos,} \\ \sum_{u \in N'(v)} f(u, 1) + \max_{\substack{\forall X \subseteq N'(v) \\ |X|=2}} \sum_{u \in X} g(u, 0, 1), & \text{caso contrário;} \end{cases} \quad (3.6)$$

$$P_v(0, 0, k) = \begin{cases} -\infty, & \text{se } v \text{ tem menos de } k \text{ filhos,} \\ \sum_{u \in N'(v)} f(u, 1) + \max_{\substack{\forall X \subseteq N'(v) \\ |X|=k}} \sum_{u \in X} g(u, 1, 1), & \text{caso contrário;} \end{cases} \quad (3.7)$$

$$P_v(0, 1, 0) = \sum_{u \in N'(v)} f(u, 1) + 1; \quad (3.8)$$

$$P_v(0, 1, 1) = \begin{cases} -\infty, & \text{se } v \text{ não tem filhos,} \\ \sum_{u \in N'(v)} f(u, 1) + \max_{u \in N'(v)} g(u, 1, 1) + 1, & \text{caso contrário;} \end{cases} \quad (3.9)$$

$$P_v(0, 1, k) = -\infty; \quad (3.10)$$

$k \geq 2$

$$P_v(1, 0, 0) = \begin{cases} -\infty, & \text{se } v = r, \\ \sum_{u \in N'(v)} f(u, 0), & \text{caso contrário;} \end{cases} \quad (3.11)$$

$$P_v(1, 0, 1) = \begin{cases} -\infty, & \text{se } v \text{ não tem filhos ou } v = r, \\ \sum_{u \in N'(v)} f(u, 1) + \max_{u \in N'(v)} g(u, 0, 1), & \text{caso contrário;} \end{cases} \quad (3.12)$$

$$P_v(1, 0, k) = \begin{cases} -\infty, & \text{se } v \text{ tem menos de } k \text{ filhos ou } v = r, \\ \sum_{u \in N'(v)} f(u, 1) + \max_{\substack{\forall S \subseteq N'(v) \\ |S|=k}} \sum_{u \in S} g(u, 1, 1), & \text{caso contrário;} \end{cases} \quad (3.13)$$

$$P_v(1, 1, 0) = \begin{cases} -\infty, & \text{se } v = r, \\ \sum_{u \in N'(v)} f(u, 1) + 1, & \text{caso contrário;} \end{cases} \quad (3.14)$$

$$P_v(1, 1, k) = -\infty, \quad k \geq 1. \quad (3.15)$$

Teorema 3.13 *A relação de recorrência acima calcula $P_v(i, j, k)$ corretamente e esse cálculo pode ser feito em tempo polinomial.*

Prova: A corretude da relação de recorrência pode ser provada por indução. Considere um vértice $v \in V(T)$ e todas as possibilidades para i, j e k .

Caso v seja uma folha de T , k deve ser 0 e, do contrário, diz-se que a contribuição de v é $-\infty$. Há quatro casos em que isso ocorre: $P_v(0, 0, 0)$, $P_v(0, 1, 0)$, $P_v(1, 0, 0)$ e $P_v(1, 1, 0)$. Uma vez que v é uma folha, seu único vizinho é p_v e isso significa que $P_v(i, j, 0)$ é sempre uma contribuição válida. Nesse caso, pode ser visto nas Equações 3.4, 3.8, 3.11 e 3.14 que $P_v(i, 0, 0) = \sum_{u \in N'(v)} f(u, 0)$ e $P_v(i, 1, 0) = \sum_{u \in N'(v)} f(u, 1) + 1$. Como ambas as somas resultam em zero, pois $N'(v) = \emptyset$, ficamos com $P_v(i, 0, 0) = 0$ e $P_v(i, 1, 0) = 1$, que está correto, visto que a contribuição máxima de T_v só pode ser zero se $v \notin S$ ou um caso contrário.

Para os demais vértices, uma análise mais detalhada faz-se necessária.

Se $i = 0$, $j = 0$ e $k = 0$, queremos o número máximo de nós em um conjunto convexamente independente que estejam também em T_v , dado que v não recebe carga alguma tanto de p_v quanto de seus filhos. Para isso, cada $u \in N'(v)$ deve dar a contribuição máxima sem enviar carga a v , isto é,

$$P_v(0, 0, 0) = \sum_{u \in N'(v)} f(u, 0).$$

Caso i seja trocado para 1 enquanto os valores j e k permanecem inalterados, a ideia é exatamente a mesma, embora seja necessário considerar o caso em que v é a raiz e, portanto, não tem pai que possa lhe enviar qualquer carga.

Supondo $i = 0$, $j = 0$ e $k = 1$, é preciso levar em conta que algum $u \in N'(v)$ deve enviar uma carga para v e, dessa forma, sua contribuição máxima é dada por $h(u, 0)$ em vez de $f(u, 0)$. Como v não está em $H(S)$, pois $ch(v) = 1$, e uma vez que é preciso escolher u de modo a maximizar a contribuição total de v , pode-se chegar a

$$P_v(0, 0, 1) = \max_{u \in N'(v)} \left(\sum_{w \in N'(v) \setminus \{u\}} f(w, 0) + h(u, 0) \right).$$

É possível, no entanto, adicionar $f(u, 0) - f(u, 0)$ a $P_v(0, 0, 1)$ sem alterar o resultado, o que resulta, afinal, em

$$P_v(0, 0, 1) = \sum_{u \in N'(v)} f(u, 0) + \max_{u \in N'(v)} g(u, 0, 0).$$

Considerando qualquer outro caso em que $j = 0$ e $k > 0$, a situação é semelhante. É preciso apenas atentar para o fato de que $ch(v)$ influencia a quantidade de carga que v envia para seus filhos e, portanto, o parâmetro i dado às funções f e g . Se $ch(v) < 2$, então v não envia carga para seus filhos. Contudo, se $ch(v)$ for exatamente dois, cada filho de v recebe uma unidade de carga, à exceção daqueles que enviam para v . Isso implica o uso de f com $i = 1$ e de g com $i_1 = 0$ e $i_2 = 1$. Além disso, se $ch(v) > 2$, então v não depende especificamente de nenhum de seus vizinhos para entrar em $H(S)$ e, portanto, f deve ser usada com $i = 1$ e g com $i_1 = i_2 = 1$. Para escolher quais filhos de v lhe enviarão cargas, é possível usar o mesmo argumento anterior e simplesmente adicionar os k maiores valores de g a $P_v(i, 0, 0)$.

Por fim, quando $j = 1$ e $k > 0$, há duas possibilidades. Se $ch(v) \geq 2$, não é possível formar um conjunto convexamente independente e, portanto, $P_v(i, 1, k) = -\infty$ sempre que $i + k \geq 2$. Se, por outro lado, $ch(v) < 2$, é possível adicionar v ao conjunto S de modo que ele jamais dependerá de qualquer vizinho para estar em

$H(S)$ e deve, portanto, enviar carga a todos eles. É preciso, ainda, adicionar v à contagem de vértices de S que pertencem a T_v . Isso significa que ainda é possível usar as mesmas fórmulas acima, mas sempre que f ou g forem utilizadas, i ou i_2 devem valer um e é preciso somar uma unidade a cada $P_v(i, 1, k)$.

Considerando a complexidade de tempo $P_v(i, j, k)$, no pior caso será necessário calcular f e g para todos os vértices da árvore. É possível assumir que será preciso calcular $f(v, 0)$, $f(v, 1)$, $h(v, 0)$ e $h(v, 1)$ para todo $v \in V(T)$. A função f pode ser computada em $O(1)$ se tivermos os valores de $P_v(i, 0, 0)$ e $P_v(i, 0, 1)$ correspondentes. Já a função h necessita de $O(\Delta(G))$ para cada vértice, mas como o grafo é uma árvore, a soma dos graus de todos os vértices é $O(n)$, então o tempo gasto para o cálculo de $h(v, 0)$ e $h(v, 1)$ para todo $v \in V(T)$ se todos os valores de $P_v(i, j, k)$ estiverem disponíveis é $O(n)$. Para computar cada $P_v(i, j, k)$, pode ser necessário, ainda, escolher os k maiores valores de $g(u, i_1, i_2)$ para todo $u \in N'(v)$. Para cada $v \in V(T)$, há $O(d(v))$ valores diferentes para k e, ordenando esses valores de g em ordem não decrescente e usando somas acumuladas, isso pode ser feito em $O(d(v) \log d(v)) \leq O(d(v) \log \Delta(T))$. A complexidade de tempo total para que se possa fazer o cálculo para todos os vértices é, então, $\sum_{v \in V(T)} O(d(v) \log \Delta(T))$, que, por sua vez, pode ser reescrita como $O(\log \Delta(T)) \sum_{v \in V(T)} O(d(v))$. Substituindo a soma dos graus de todos os vértices de T por $O(n)$, obtém-se o resultado final $O(n \log \Delta(T))$, que é claramente polinomial no tamanho da entrada.

Por fim, todos os cálculos que levam à complexidade de tempo do algoritmo consideram que, ao calcular o valor de $P_v(i, j, k)$, todos os demais $P_u(i, j, k)$, $u \neq v$, dos quais ele depende já foram previamente computados. É possível observar, entretanto, que $P_v(i, j, k)$ não depende de nenhum outro resultado além daqueles dados por seus filhos. Dessa forma, percorrendo a árvore do maior nível até a raiz será sempre possível calcular $P_v(i, j, k)$, o que conclui a prova. ■

Corolário 3.14 $rk(T) = |S^*|$ é dado por

$$\max_{j \in \{0,1\}, 0 \leq k \leq d(r)} P_r(0, j, k).$$

Ademais, esse valor pode ser obtido em tempo $O(n \log \Delta(T))$.

Prova: Uma vez que $P_v(i, j, k)$ é o maior número de vértices na subárvore com raiz em v que podem estar simultaneamente em um conjunto convexamente independente sob as condições impostas por i , j e k , quando $r = v$ temos a cardinalidade de um conjunto convexamente independente máximo para toda a árvore, isto é, o posto de T . Como é possível calcular todos os valores de $P_v(i, j, k)$ em tempo $O(n \log \Delta(T))$, é possível obter o posto da árvore T com esta mesma complexidade. ■

3.2.3 Árvores nas convexidades geodética e monofônica

No caso da convexidade P_3 foi preciso usar um algoritmo de programação dinâmica relativamente sofisticado a fim de se descobrir o posto de um grafo. Para as convexidades geodética e monofônica, porém, o problema torna-se mais simples.

Primeiramente, note que as convexidades geodética e monofônica são equivalentes para árvores, tendo em vista que há apenas um único caminho entre quaisquer dois vértices e, portanto, este é o caminho mais curto e, naturalmente, trata-se de um caminho induzido. Isso significa que todos os resultados apresentados abaixo não precisam de nenhuma adaptação para que valham em ambos os casos. Dessa forma, é possível enunciar o teorema a seguir.

Teorema 3.15 *Seja T uma árvore $S \subseteq V(T)$ o conjunto das folhas de T . S é um conjunto convexamente independente máximo de T nas convexidades geodética e monofônica.*

Prova: A prova é composta por duas partes. Na primeira mostra-se que S é um conjunto convexamente independente. A seguir, é demonstrado o fato de que S é máximo. Argumentar que S é convexamente independente é simples, tendo em vista que nenhuma folha pode ser gerada por uma convexidade de caminhos. Dessa forma, nenhum vértice $v \in S$ pode estar em $H(S - v)$. A segunda parte da prova é feita por contradição. Suponha que S' é um conjunto convexamente independente máximo de T com o maior número possível de folhas e que $v \in V(T) \setminus S'$ é uma folha da árvore que ficou de fora de S' . Há duas possibilidades para que v não possa fazer parte de S' :

- (a) $v \in H(S')$;
- (b) há algum vértice $u \in S'$ tal que $u \in H(S' - u + v)$.

É naturalmente impossível que (a) seja verdade, visto que uma folha não pode ser gerada. Resta, portanto, o caso (b).

Na Figura 3.3, que auxiliará no restante da prova, considere que as arestas pontilhadas representam, na verdade, caminhos entre os vértices com um número arbitrário de vértices. Dessa forma, pode-se representar apenas os vértices necessários sem perda de generalidade. Considere, nessa figura, que $v = v_3$ e que $u = v_2$. Suponha, ainda, que as folhas v_5 e v_6 fazem parte de S' .

Considere os vw -caminhos P_w para todo $w \in S'$. Note que nenhum desses caminhos pode conter mais do que dois vértices de S' , sendo um deles o próprio w . Do contrário, haveria um vértice de S' que poderia ser gerado por dois outros, o que tornaria o conjunto convexamente dependente. Na figura, se qualquer vértice

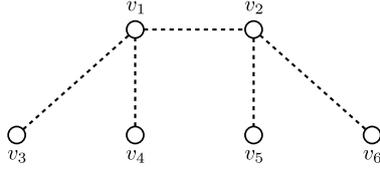


Figura 3.3: Considere que as linhas pontilhadas representam caminhos com um número arbitrário de vértices e que são representados apenas os vértices necessários para que a figura não perca generalidade.

no caminho de v_3 até v_5 além de v_2 também estiver em S' , este seria gerado por v_2 e v_5 ou, junto com v_5 , geraria v_2 .

Além disso, o mesmo argumento pode ser usado para afirmar que existe um vértice $u \in S'$ que está em todos os vw -caminhos e não é uma folha de T . Caso isso não fosse verdade, u seria gerado por algum outro par de vértices de S' . Suponha que $v_3 = u$ não está em algum caminho. Isso implicaria algum vértice à sua esquerda na figura — como v_1 ou v_4 — estar em S' . Isso significaria, porém, que o caminho desse vértice até v_5 ou v_6 geraria v_3 .

Dessa forma, o único vértice do conjunto que impede a entrada de v é exatamente u e, portanto, o conjunto $S = S' - u \cup \{v\}$ é convexamente independente e $|S| = |S'|$. Isso, porém, significa que S é um conjunto convexamente independente máximo com mais folhas do que S' , uma contradição. Conclui-se, portanto, que todas as folhas da árvore podem estar em qualquer conjunto independente máximo.

Por fim, note que o conjunto S das folhas de T é um conjunto de envoltória, pois todo nó interno está no caminho entre algum par de folhas. Conclui-se, portanto, que S é um conjunto convexamente independente máximo da árvore. ■

O Teorema 3.15 permite que se chegue facilmente à conclusão apresentada no Corolário 3.16.

Corolário 3.16 *Pode-se calcular o posto de uma árvore nas convexidades geodética e monofônica em tempo $O(n)$.*

Prova: De acordo com o Teorema 3.15, o conjunto S das folhas de uma árvore é um conjunto convexamente independente máximo nas convexidades geodética e monofônica. Isso significa que basta contar o número de vértices de grau um de T para determinar $rk(T)$. O número de folhas pode ser contado em $O(n + m)$, o que, em uma árvore, equivale a $O(n)$. Dessa forma, o tempo para se calcular $rk(T)$ é $O(n)$. ■

3.2.4 Grafos de intervalo biconexos na convexidade P_3

O fato de que todo grafo de intervalo G é também cordal torna possível obter um algoritmo guloso simples, baseado em uma de suas representações por uma família

de intervalos, que permite calcular o posto de G na convexidade P_3 em tempo polinomial. Para isso, emprega-se o resultado a seguir.

Lema 3.17 *Sejam G um grafo cordal biconexo e $S \subseteq V(G)$ tal que $|S| \geq 2$. Se $u, v \in S$ são tais que $d(u, v) \leq 2$, então $H(S) = V(G)$.*

Seja \mathcal{I} uma família de intervalos, $|\mathcal{I}| \geq 2$, tal que o grafo de intervalo correspondente G é biconexo. Se o diâmetro de G é menor ou igual a dois, então quaisquer dois de seus vértices geram todo o grafo e, portanto, $rk(G) = 2$. Caso contrário, o algoritmo consiste em repetidamente adicionar ao conjunto S o vértice $v \in V(G)$ cujo intervalo correspondente termina mais cedo e remover do grafo todo $u \in V(G)$ tal que $d(u, v) \leq 2$ — o próprio v incluso, visto que $d(v, v) = 0$.

Teorema 3.18 *S é um conjunto convexamente independente máximo.*

Prova: Uma vez que G é um grafo de intervalo e todos os pares de vértices de S têm distância ao menos três, nenhum nó é gerado e, portanto, S é convexamente independente.

Para provar que S é um conjunto convexamente independente máximo, escolha a menor solução lexicograficamente em termos dos finais dos intervalos, $v_1, v_2, \dots, v_{rk(G)} \in S'$. Considere, ainda, que a resposta dada pelo algoritmo u_1, u_2, \dots, u_k não é ótima. Escolha a diferença mais à esquerda entre as sequências, isto é, selecione i de modo que $u_i \neq v_i$ e $u_j = v_j$ para $1 \leq j < i$.

Se $v_i < u_i$, então $i > 1$, pois u_i é o menor vértice, lexicograficamente, de G . Além disso, $d(v_{i-1}, v_i) \leq 2$, o que significa que $H(\{v_{i-1}, v_i\}) = V(G)$, por sua vez, implicando que nenhum outro vértice pode ser adicionado a S' sem torná-lo convexamente dependente. Isso é uma contradição, pois assumimos que a sequência contida em S' é mais longa do que aquela contida em S e, nesse caso, $|S| \geq |S'|$.

É preciso, portanto, assumir que $v_i > u_i$. No entanto, $d(v_{i-1}, u_i) \geq 3$, pois ambos estão presentes em S' que, por sua vez, não poderia ter mais nenhum outro vértice em caso contrário. Além disso, como o intervalo correspondente a u_i termina, no mínimo, tão cedo quanto v_i , a $d(u_i, v_{i+1})$ não pode ser menor do que $d(v_i, v_{i+1})$ e, dessa forma, deve ser ao menos três. Isso significa que nenhum vértice é também gerado por u_i e v_{i+1} . Isso significa que $v_1, v_2, \dots, v_{i-1}, u_i, v_{i+1}, \dots, v_{rk(G)}$ é uma solução lexicograficamente menor do que a contida por S' , uma contradição. ■

O Teorema 3.18 afirma que o algoritmo guloso funciona. É preciso apenas determinar o tempo necessário para sua execução. Para isso, considere cada uma das etapas a serem executadas para se chegar ao conjunto convexamente independente máximo S em um grafo de intervalo biconexo G .

- (i) Obter a representação de G por uma família de intervalos \mathcal{I} .

- (ii) Ordenar os intervalos pelo seu fim, não sendo necessário qualquer desempate.
- (iii) Escolher o vértice correspondente ao intervalo que termina mais cedo. e adicioná-lo a S .
- (iv) Remover o vértice escolhido e todos os demais a uma distância menor do que três dele.
- (v) Se ainda houver algum vértice em G , retorne a (iii).

Teorema 3.19 *O conjunto convexamente independente máximo de um grafo de intervalo biconexo pode ser encontrado em tempo linear.*

Prova: Considere as etapas do algoritmo descritas anteriormente. Sabe-se que (i) pode ser feito em tempo $O(n + m)$, pois equivale ao reconhecimento de grafos de intervalo por meio de uma ordenação de suas cliques maximais [7]. Já a etapa (ii) pode ser feita em tempo $O(n)$ aproveitando-se o resultado da fase anterior.

Escolher o vértice v conforme (iii) e adicioná-lo ao conjunto S é feito com custo $O(1)$. Considerando a remoção do vértice escolhido e de todo $u \in V(G)$ tal que $d(u, v) \leq 2$, descrita em (iv), cabe observar que tal etapa deve ser repetida até que todos os vértices tenham sido removidos. Cada um dos nós de G será removido exatamente uma vez e, para encontrar os vértices a serem apagados, basta seguir as arestas do grafo uma única vez. Mesmo se consideramos que cada aresta pode ser visitada duas vezes, não resta dúvida de que (iii), (iv) e (v) podem ser completadas em $O(n + m)$.

Dessa forma, o tempo total de execução do algoritmo é $O(n + m) + O(n) + O(n + m)$, o que resulta em $O(n + m)$. ■

Observe que, a partir dos Teoremas 3.18 e 3.19, pode-se resolver o problema do EMPACOTAMENTO ABERTO para grafos de intervalo biconexos.

Corolário 3.20 *O problema do EMPACOTAMENTO ABERTO pode ser resolvido em tempo linear para grafos de intervalo biconexos.*

Pode-se também concluir que o algoritmo para se chegar ao empacotamento aberto de maior cardinalidade é o mesmo que nos dá o conjunto convexamente independente máximo.

Conjectura para outros grafos de intervalo

O algoritmo guloso apresentado para grafos de intervalo biconexos não funciona e nem pôde ser adaptado facilmente para grafos de intervalo que contenham alguma articulação. A seguir, porém, é apresentada uma conjectura.

Conjectura 3.21 *O problema do CONJUNTO CONVEXAMENTE INDEPENDENTE pode ser resolvido em tempo polinomial para grafos de intervalo na convexidade P_3 .*

A Conjectura 3.21 baseia-se na suposição de que é possível usar um algoritmo de programação dinâmica semelhante àquele usado para árvores com a finalidade de se resolver o problema para grafos de intervalo. Cabe ressaltar, porém, que o problema do CONJUNTO CONVEXAMENTE INDEPENDENTE é \mathcal{NP} -completo para a classe dos grafos *split*, ele também o é para a superclasse dos grafos *split* e dos grafos de intervalo, a classe dos grafos cordais.

3.3 Limite superior

“Little by little, one travels far.”

J.R.R. Tolkien

Pode-se relacionar o posto a outros parâmetros da convexidade de grafos. Conforme dito anteriormente, por exemplo, todo conjunto de envoltória minimal é convexamente independente, o que permite concluir que $h(G) \leq rk(G)$. Igualmente, qualquer conjunto antirradon também é convexamente independente, o que nos leva à conclusão de que $r(G) - 1 \leq rk(G)$. Tomando o limite para o número de Radon [24] dado por $r(G) - 1 \leq \frac{2n}{\delta(G)+1}$, pode-se mostrar que este também é um limite superior para o posto de um grafo, chegando-se também a uma prova mais simples para a desigualdade do artigo original por consequência. Isso também significa que o posto é um limite mais justo para o número de Radon, visto que $r(G) - 1 \leq rk(G) \leq \frac{2n}{\delta(G)+1}$.

Teorema 3.22 *Seja G um grafo com grau mínimo $\delta(G)$. Então $rk(G) \leq \frac{2n}{\delta(G)+1}$. Ademais, esse limite é justo.*

Prova: Primeiramente é possível observar que a igualdade é atingida nos grafos completos, o que mostra que o limite é justo.

Sejam $S \subseteq V(G)$ um conjunto convexamente independente máximo e $R = V(G) \setminus S$ seu complemento. Considere S_i o subconjunto que contém os vértices de S que têm exatamente i vizinhos em S e R_j o subconjunto dos vértices de R que são adjacentes a exatamente j elementos de S . Para simplificar a notação, as cardinalidades desses conjuntos serão representadas por $r_i = |R_i|$ e $s_i = |S_i|$.

Note que, como S é convexamente independente, $S_i = \emptyset$ para $i \geq 2$ e, portanto, $S = S_0 \cup S_1$. Além disso, observe que nenhum vértice de S_1 pode ter um vizinho em $\bigcup_{i \geq 3} R_i$ e que, se um vértice $v \in S_0$ tem mais de um vizinho em $\bigcup_{i \geq 3} R_i$, então $v \in H(S - v)$ e, dessa forma, isso não é possível, pois S é convexamente independente. Note, ainda, que os conjuntos S_i e R_j definem uma partição de $V(G)$

e que $\sum_{v \in S} |N(v) \cap R| = \sum_{w \in R} |N(w) \cap S|$. Dessas observações pode-se obter as equações a seguir.

$$rk(G) = s_0 + s_1 \quad (3.16)$$

$$s_0 + s_1 + \sum_{i=0}^{\Delta(G)} r_i = n \quad (3.17)$$

$$\sum_{i=3}^{\Delta(G)} ir_i \leq s_0 \quad (3.18)$$

$$\sum_{v \in S_0} d(v) + \sum_{v \in S_1} (d(v) - 1) = \sum_{i=0}^{\Delta(G)} ir_i \quad (3.19)$$

Das Equações 3.18 e 3.19 segue que.

$$\begin{aligned} \sum_{v \in S_0} d(v) + \sum_{v \in S_1} (d(v) - 1) &\leq r_1 + 2r_2 + s_0 \\ \sum_{v \in S_0} (d(v) - 1) + \sum_{v \in S_1} (d(v) - 1) &\leq r_1 + 2r_2 \leq 2(r_1 + r_2) \\ \sum_{v \in S_0} (\delta(G) - 1) + \sum_{v \in S_1} (\delta(G) - 1) &\leq 2(r_1 + r_2) \\ \frac{\delta(G) - 1}{2}(s_0 + s_1) &\leq r_1 + r_2. \end{aligned}$$

Combinando a desigualdade acima com a Equação 3.17, obtém-se:

$$\begin{aligned} s_0 + s_1 + r_0 + r_1 + r_2 + \sum_{i=3}^{\Delta(G)} r_i &= n \\ s_0 + s_1 + r_1 + r_2 &\leq n \\ s_0 + s_1 + \frac{\delta(G) - 1}{2}(s_0 + s_1) &\leq n \\ s_0 + s_1 &\leq \frac{2n}{\delta(G) + 1}, \end{aligned}$$

o que completa a prova. ■

Capítulo 4

Conclusões

“If we shadows have offended,
Think but this, and all is mended:
That you have but slumbered here,
While these visions did appear;
And this weak and idle theme,
No more yielding but a dream,
Gentles, do not reprehend.
If you pardon, we will mend.”

William Shakespeare, A
Midsummer Night’s Dream

Nesta dissertação foram apresentados os resultados obtidos ao longo do mestrado em ordem semelhante à cronológica. Algumas inversões foram feitas com o objetivo de facilitar a compreensão e tornar o texto mais coeso.

Inicialmente foram apresentados os resultados relacionados à \mathcal{NP} -completude do problema do CONJUNTO CONVEXAMENTE INDEPENDENTE nas convexidades P_3 e monofônica. Os teoremas da Seção 3.1 mostram que resolver esse problema para grafos *split* e bipartidos na convexidade P_3 e para átomos na convexidade monofônica em tempo polinomial permite encontrar uma solução, também em tempo polinomial, para todos os demais problemas \mathcal{NP} -completos como a CLIQUE MÁXIMA e o EMPACOTAMENTO DE CONJUNTOS.

A seguir, a Seção 3.2 contém algoritmos polinomiais para se encontrar o posto de alguns grafos. Nos casos das árvores nas convexidades monofônica e geodética e dos grafos de intervalo biconexos na convexidade P_3 , são apresentados algoritmos lineares, enquanto um fator logarítmico precisa ser acrescentado no caso das árvores na convexidade P_3 .

Finalmente, na Seção 3.3, é apresentado um limite superior justo para o posto de um grafo. Esse resultado mostra que o posto é um limite superior mais justo

para o número de Radon.

Em alguns casos, o resultado obtido para o CONJUNTO CONVEXAMENTE INDEPENDENTE na convexidade P_3 é também válido para o problema do EMPACOTAMENTO ABERTO, como a \mathcal{NP} -completude para grafos *split* — uma subclasse dos grafos cordais, para os quais já se conhecia a \mathcal{NP} -completude desse problema [23] — e o algoritmo polinomial para grafos de intervalo biconexos.

A maior parte dos resultados apresentados foi submetida em um artigo que encontra-se ainda sob avaliação.

4.1 Trabalhos futuros

Este trabalho tem foco na convexidade P_3 , havendo poucos resultados para outras convexidades importantes como a geodética e a monofônica. Uma questão natural decorrente disso é se o problema do CONJUNTO CONVEXAMENTE INDEPENDENTE também é \mathcal{NP} -completo na convexidade geodética para grafos gerais. Além disso, também pode-se perguntar para que outras classes de grafos o problema é \mathcal{NP} -completo nas convexidades P_3 e monofônica.

Outro caminho a ser seguido é buscar outros casos particulares para os quais o posto pode ser encontrado em tempo polinomial. Nas convexidades geodética e monofônica, as classes que podem ser estudadas incluem os grafos *split*, bipartidos, cordais e de intervalo, para os quais já há solução na convexidade P_3 . A conjectura 3.21 também é um ponto de partida relevante.

É possível, ainda, buscar limites mais justos para o posto e estabelecer mais relações com outros parâmetros de uma convexidade. As estruturas específicas das classes de grafos podem ser úteis para que se estabeleçam limites melhores para casos específicos.

Finalmente, há a possibilidade de se buscar algoritmos aproximativos nos casos em que a \mathcal{NP} -completude estiver provada. No caso dos grafos regulares, por exemplo, há uma relação entre os conjuntos de envoltória na convexidade P_3 e os *feedback vertex sets*, que são conjuntos de vértices cuja remoção elimina todos os ciclos do grafo [3]. Isso permite a obtenção de um algoritmo polinomial para o cálculo da envoltória convexa na convexidade, o que pode despertar a suspeita de que o problema do posto pode ser também polinomial para grafos cúbicos. Já nos casos de grafos regulares com grau maior do que três, não há nenhuma indicação de que o problema possa ser polinomial e a abordagem aproximativa pode vir a ser efetiva.

Referências Bibliográficas

- [1] Z. AGUR. Fixed points of majority rule cellular automata with application to plasticity and precision of the immune system. *Complex Systems*, v. 5, n. 3, pp. 351–357, 1991.
- [2] D. ARTIGAS, S. DANTAS, M. C. DOURADO e J. L. SZWARCFITER. Partitioning a graph into convex sets. *Discrete Mathematics*, v. 311, n. 17, pp. 1968–1977, 2011.
- [3] R. BARBOSA, D. RAUTENBACH, V. F. DOS SANTOS e J. L. SZWARCFITER. On minimal and minimum hull sets. *Electronic Notes in Discrete Mathematics*, v. 44, n. 0, pp. 207 – 212, 2013.
- [4] R. M. BARBOSA, E. M. COELHO, M. C. DOURADO, D. RAUTENBACH e J. L. SZWARCFITER. On the carathéodory number for the convexity of paths of order three. *SIAM Journal on Discrete Mathematics*, v. 26, n. 3, pp. 929–939, 2012.
- [5] J.-C. BERMOND e D. PELEG. The power of small coalitions in graphs., In: *SIROCCO*, pp. 173–184, 1995.
- [6] J. A. BONDY e U. S. R. MURTY. *Graph theory with applications*, v. 6. Macmillan London, 1976.
- [7] K. S. BOOTH e G. S. LUEKER. Testing for the consecutive ones property, interval graphs, and graph planarity using pq-tree algorithms. *Journal of Computer and System Sciences*, v. 13, n. 3, pp. 335 – 379, 1976.
- [8] A. BRANDSTÄDT, J. P. SPINRAD e OTHERS. *Graph classes: a survey*, v. 3. Siam, 1999.
- [9] C. C. CENTENO, S. DANTAS, M. C. DOURADO, D. RAUTENBACH e J. L. SZWARCFITER. Convex partitions of graphs induced by paths of order three. *Discrete Mathematics & Theoretical Computer Science*, v. 12, n. 5, pp. 175–184, 2010.

- [10] P. DOMINGOS e M. RICHARDSON. Mining the network value of customers, In: *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 57–66. ACM, 2001.
- [11] M. C. DOURADO, F. PROTTI e J. L. SZWARCFITER. Algorithmic aspects of monophonic convexity. *Electronic Notes in Discrete Mathematics*, v. 30, pp. 177–182, 2008.
- [12] M. C. DOURADO, F. PROTTI, D. RAUTENBACH e J. L. SZWARCFITER. Some remarks on the geodetic number of a graph. *Discrete Mathematics*, v. 310, n. 4, pp. 832–837, 2010.
- [13] M. C. DOURADO, F. PROTTI, D. RAUTENBACH e J. L. SZWARCFITER. On the hull number of triangle-free graphs. *SIAM Journal on Discrete Mathematics*, v. 23, n. 4, pp. 2163–2172, 2010.
- [14] M. C. DOURADO, F. PROTTI e J. L. SZWARCFITER. Complexity results related to monophonic convexity. *Discrete Applied Mathematics*, v. 158, n. 12, pp. 1268–1274, 2010.
- [15] M. C. DOURADO, D. RAUTENBACH, V. FERNANDES DOS SANTOS, P. M. SCHÄFER, J. L. SZWARCFITER e A. TOMAN. An upper bound on the p_3 -radon number. *Discrete Mathematics*, v. 312, n. 16, pp. 2433–2437, 2012.
- [16] M. C. DOURADO, D. RAUTENBACH, V. F. DOS SANTOS, P. M. SCHÄFER, J. L. SZWARCFITER e A. TOMAN. Algorithmic and structural aspects of the p_3 -radon number. *Annals of Operations Research*, v. 206, n. 1, pp. 75–91, 2013.
- [17] P. A. DREYER JR e F. S. ROBERTS. Irreversible k -threshold processes: Graph-theoretical threshold models of the spread of disease and of opinion. *Discrete Applied Mathematics*, v. 157, n. 7, pp. 1615–1627, 2009.
- [18] J. EDMONDS e R. M. KARP. Theoretical improvements in algorithmic efficiency for network flow problems. *Journal of the ACM (JACM)*, v. 19, n. 2, pp. 248–264, 1972.
- [19] H. B. ENDERTON. *Elements of set theory*. Academic Press, 1977.
- [20] L. EULER. Solutio problematis ad geometriam situs pertinentis. *Commentarii academiae scientiarum Petropolitanae*, v. 8, pp. 128–140, 1741.
- [21] J. R. FRENCH JR. A formal theory of social power. *Psychological review*, v. 63, n. 3, pp. 181, 1956.

- [22] M. C. GOLUMBIC. *Algorithmic graph theory and perfect graphs*, v. 57. Elsevier, 2004.
- [23] M. HENNING e P. SLATER. Open packing in graphs. *JOURNAL OF COMBINATORIAL MATHEMATICS AND COMBINATORIAL COMPUTING*, v. 29, pp. 3–16, 1999.
- [24] M. A. HENNING, D. RAUTENBACH e P. M. SCHÄFER. Open packing, total domination, and the P_3 -Radon number. *Discrete Mathematics*, v. 313, n. 9, pp. 992 – 998, 2013.
- [25] I. HOLYER. The np-completeness of edge-coloring. *SIAM Journal on Computing*, v. 10, n. 4, pp. 718–720, 1981.
- [26] J. E. HOPCROFT e R. M. KARP. An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs. *SIAM Journal on computing*, v. 2, n. 4, pp. 225–231, 1973.
- [27] S. HUANG. Gene expression profiling, genetic networks, and cellular states: an integrating concept for tumorigenesis and drug discovery. *Journal of Molecular Medicine*, v. 77, n. 6, pp. 469–480, 1999.
- [28] R. KARP. Reducibility among combinatorial problemsIn: R. Miller, J. Thatcher e J. Bohlinger (Eds.), *Complexity of Computer Computations*, The IBM Research Symposia Series, Springer US, pp. 85–103, 1972.
- [29] D. KEMPE, J. KLEINBERG e É. TARDOS. Maximizing the spread of influence through a social network, In: *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 137–146. ACM, 2003.
- [30] S. KUTTEN e D. PELEG. Fault-local distributed mending, In: *Proceedings of the fourteenth annual ACM symposium on Principles of distributed computing*, pp. 20–27. ACM, 1995.
- [31] E. MALAGUTI e P. TOTH. A survey on vertex coloring problems. *International Transactions in Operational Research*, v. 17, n. 1, pp. 1–34, 2010.
- [32] N. H. MUSTAFA e A. PEKEC. Listen to your neighbors: How (not) to reach a consensus. *SIAM Journal on Discrete Mathematics*, v. 17, n. 4, pp. 634–660, 2004.
- [33] B. PEIS, M. SKUTELLA e A. WIESE. Packet routing: Complexity and algorithmsIn: *Approximation and Online Algorithms*, Springer, pp. 217–228, 2010.

- [34] D. PELEG. Local majorities, coalitions and monopolies in graphs: a review. *Theoretical Computer Science*, v. 282, n. 2, pp. 231–257, 2002.
- [35] S. POLJAK e M. SÚRA. On periodical behaviour in societies with symmetric influences. *Combinatorica*, v. 3, n. 1, pp. 119–121, 1983.
- [36] F. S. ROBERTS. *Graph Theory and Its Applications to Problems of Society*. SIAM, 1978.
- [37] J. L. SZWARCFITER. *Grafos e algoritmos computacionais*, v. 2. Campus, 1988.
- [38] M. VAN DE VEL. *Theory of convex structures*, v. 50. Elsevier, 1993.