

**Volume 82, 2016**

**Editores**

**Alexandre Loureiro Madureira (Editor Chefe)**

Laboratório Nacional de Computação Científica - LNCC

Petrópolis, RJ, Brasil

**Amanda Liz Pacífico Manfrim Perticarrari**

Universidade Estadual Paulista Júlio de Mesquita Filho - UNESP

Jaboticabal, SP, Brasil

**Edson Luiz Cataldo Ferreira**

Universidade Federal Fluminense - UFF

Niterói, RJ, Brasil

**Eduardo V. O. Teixeira (Editor Executivo)**

Universidade Federal do Ceará - UFC

Fortaleza, CE, Brasil

**Jorge Manuel Vieira Capela**

Universidade Estadual Paulista - UNESP

Araraquara, SP, Brasil

**Sandra Augusta Santos**

Universidade Estadual de Campinas - UNICAMP

Campinas, SP, Brasil

A Sociedade Brasileira de Matemática Aplicada e Computacional - SBMAC publica, desde as primeiras edições do evento, monografias dos cursos que são ministrados nos CNMAC.

Para a comemoração dos 25 anos da SBMAC, que ocorreu durante o XXVI CNMAC em 2003, foi criada a série **Notas em Matemática Aplicada** para publicar as monografias dos minicursos ministrados nos CNMAC, o que permaneceu até o XXXIII CNMAC em 2010.

A partir de 2011, a série passa a publicar, também, livros nas áreas de interesse da SBMAC. Os autores que submeterem textos à série Notas em Matemática Aplicada devem estar cientes de que poderão ser convidados a ministrarem minicursos nos eventos patrocinados pela SBMAC, em especial nos CNMAC, sobre assunto a que se refere o texto.

O livro deve ser preparado em **Latex (compatível com o Miktex versão 2.9)**, as **figuras em eps** e deve ter entre **80 e 150 páginas**. O texto deve ser redigido de forma clara, acompanhado de uma excelente revisão bibliográfica e de **exercícios de verificação de aprendizagem** ao final de cada capítulo.

Veja todos os títulos publicados nesta série na página  
[http://www.sbmac.org.br/p\\_notas.php](http://www.sbmac.org.br/p_notas.php)

# FERRAMENTAS ESTRUTURAIS EM GRAFOS CORDAIS

Lilian Markenzon  
markenzon@nce.ufrj.br

Núcleo de Computação Eletrônica  
Programa de Pós-Graduação em Informática  
Universidade Federal do Rio de Janeiro

Christina Fraga Esteves Maciel Waga  
waga@ime.uerj.br

Departamento de Estruturas Matemáticas  
Instituto de Matemática e Estatística  
Universidade do Estado do Rio de Janeiro

**Sociedade Brasileira de Matemática Aplicada e Computacional**

São Carlos - SP, Brasil  
2016

Coordenação Editorial: Maria do Socorro Nogueira Rangel

Coordenação Editorial da Série: Fernando Rodrigo Rafaeli

Editora: SBMAC

Capa: Matheus Botossi Trindade

Patrocínio: SBMAC

Copyright ©2016 by Lilian Markenzon e Christina Fraga Esteves Maciel Waga. Direitos reservados, 2016 pela SBMAC. A publicação nesta série não impede o autor de publicar parte ou a totalidade da obra por outra editora, em qualquer meio, desde que faça citação à edição original.

**Catálogo elaborado pela Biblioteca do IBILCE/UNESP**  
**Bibliotecária: Maria Luiza Fernandes Jardim Froner**

Markenzon, Lilian

Ferramentas Estruturais em Grafos Cordais - São Carlos, SP :  
SBMAC, 2016, 96 p., 21.5 cm - (Notas em Matemática  
Aplicada; v. 82)

e-ISBN 978-85-8215-073-3

1. Grafos Cordais 2. Propriedades Estruturais 3. Ferramentas  
I. Markenzon, L. II. Waga, Christina F. E. M. III. Título. IV. Série

CDD - 51

Ao Newton.  
(L.M.)

Aos meus pais Kleber (*in memoriam*) e Cléa.  
Aos meus filhos Augusto e Isabela.  
(C.F.E.M.W.)



# Agradecimentos

Agradecemos à SBMAC e aos editores da série Notas em Matemática Aplicada, pela oportunidade; aos alunos, que nos instigaram com dúvidas e questionamentos durante tantos anos de ensino; aos colegas, que nos estimularam; ao CNPq, pela bolsa concedida.



# Conteúdo

<b>Prefácio</b>	<b>xi</b>
<b>1 Conceitos Preliminares</b>	<b>1</b>
1.1 Grafos em geral . . . . .	1
1.1.1 Representação de grafos . . . . .	3
1.2 Propriedades, suficiência e necessidade . . . . .	4
1.3 Classes de grafos . . . . .	6
1.4 Grafos cordais . . . . .	7
1.5 Exercícios . . . . .	9
1.6 Notas bibliográficas . . . . .	10
<b>2 Ferramentas Estruturais</b>	<b>11</b>
2.1 Sequências de vértices . . . . .	11
2.1.1 Esquema de eliminação perfeita . . . . .	11
2.1.2 Esquema de eliminação bicompatível . . . . .	15
2.1.3 Outros esquemas de eliminação . . . . .	16
2.2 Cliques maximais e grafo de cliques . . . . .	16
2.3 Separadores minimais e árvore de cliques . . . . .	19
2.3.1 Construção da árvore de cliques e determinação de $\mathcal{S}$ . . . . .	21
2.3.2 Propriedades de árvores de cliques . . . . .	23
2.4 Grafo de cliques reduzido . . . . .	24
2.5 Grafo de cliques críticas . . . . .	26
2.6 Subgrafos proibidos . . . . .	29
2.7 Exercícios . . . . .	29
2.8 Notas bibliográficas . . . . .	30
<b>3 Subclasses de Grafos Cordais</b>	<b>31</b>
3.1 Grafos de intervalo . . . . .	31
3.1.1 Subclasses de grafos de intervalo . . . . .	34
3.2 Grafos cordais unicamente representáveis . . . . .	35
3.3 Grafos ptolemaicos . . . . .	35
3.3.1 Subclasses de grafos ptolemaicos . . . . .	39
3.4 A hierarquia . . . . .	41
3.5 Exercícios . . . . .	42
3.6 Notas bibliográficas . . . . .	42

<b>4</b>	<b>Grafos de Intervalo Próprio</b>	<b>45</b>
4.1	A classe . . . . .	45
4.2	Reconhecimento . . . . .	48
4.3	Um problema: ciclo hamiltoniano . . . . .	50
4.4	Uma subclasse: grafos split-indiferença . . . . .	53
4.4.1	A contagem de grafos split-indiferença . . . . .	56
4.5	Exercícios . . . . .	60
4.6	Notas bibliográficas . . . . .	61
<b>5</b>	<b>Grafos Estritamente Cordais</b>	<b>63</b>
5.1	A classe . . . . .	63
5.2	Reconhecimento . . . . .	65
5.3	$C_c(G)$ de grafos estritamente cordais . . . . .	66
5.3.1	Propriedades . . . . .	66
5.3.2	Algoritmo de construção . . . . .	69
5.4	Um problema: dominação . . . . .	70
5.4.1	Dominação conexa . . . . .	74
5.4.2	Grafos bem dominados . . . . .	74
5.4.3	Dominação e novas subclasses . . . . .	76
5.5	Grafos $AC$ . . . . .	76
5.5.1	Reconhecimento . . . . .	77
5.6	Grafos potência de 3-folha . . . . .	78
5.6.1	Reconhecimento . . . . .	80
5.7	Grafos estritamente de intervalo . . . . .	80
5.7.1	Reconhecimento . . . . .	81
5.8	Comparando classes . . . . .	82
5.8.1	Grafos estritamente de intervalo $\times$ grafos de intervalo próprio . . . . .	82
5.8.2	Quatro subclasses . . . . .	83
5.9	A hierarquia . . . . .	84
5.10	Exercícios . . . . .	85
5.11	Notas bibliográficas . . . . .	86

# Prefácio

Um dos assuntos mais atuais em grafos é a área que trata de classes e soluções computacionais para problemas nas mesmas. Neste contexto, a classe dos grafos cordais e suas subclasses é, sem dúvida, uma das mais estudadas. A intenção deste texto é mostrar os aspectos importantes na abordagem de uma classe de grafos quando se prioriza um enfoque algorítmico e, para tal, utilizaremos uma hierarquia de subclasses de grafos cordais. Alguns tópicos são especialmente relevantes para cada classe estudada: a definição, caracterizações, o desenvolvimento de algoritmos eficientes de reconhecimento, o estabelecimento de propriedades estruturais específicas para a solução de problemas e o relacionamento com outras classes.

Fator decisivo, mas pouco valorizado, no tratamento dos tópicos mencionados é o emprego de estruturas específicas, concebidas para lidar com cada classe de maneira adequada e que são essenciais para o desenvolvimento de algoritmos eficientes. Propomos, então, o levantamento de um conjunto de *ferramentas estruturais* – estruturas diversas com as quais é possível estabelecer, de uma forma mais sistemática, as propriedades das classes estudadas – descrevendo como são constituídas, suas propriedades e peculiaridades.

É importante salientar que existem poucos textos didáticos sobre o assunto. Podemos destacar os livros de Golumbic [29] e Brandstädt *et al.* [9]; somente o primeiro, entretanto, trata de problemas algorítmicos. Em português, o livro de Szwarcfiter [71] é referência introdutória sobre algoritmos em grafos.

Nosso texto é destinado a alunos de graduação e pós-graduação com conhecimentos básicos de Teoria de Grafos e Algoritmos em Grafos. Pretendemos oferecer uma visão mais focada em alguns aspectos não explorados habitualmente em cursos regulares e, assim, despertar um interesse maior pela pesquisa na área.

O Capítulo 1 contém a revisão dos conceitos básicos necessários à compreensão do texto, noções gerais sobre classes de grafos e fundamentos de grafos cordais.

No Capítulo 2 introduzimos as ferramentas estruturais utilizadas, definindo e apresentando as principais propriedades de cada uma delas: sequências de vértices; cliques maximais e grafo de cliques; separadores minimais e árvore de cliques; grafo de cliques reduzido; grafo de cliques críticas e subgrafos proibidos. A eficiência dos algoritmos de geração de cada uma é considerada e, em alguns casos, aplicações são descritas.

No Capítulo 3 revemos três ramos na hierarquia de grafos cordais: os grafos de intervalo, os grafos unicamente representáveis e os grafos ptolemaicos, arrolando, para cada um deles, algumas subclasses. Poderemos observar como, a partir das diversas caracterizações baseadas nas ferramentas descritas, essas subclasses se interrelacionam.

No Capítulo 4, uma classe bastante conhecida, os grafos de intervalo próprio, ilustra diversos itens abordados anteriormente. A caracterização por sequência canônica é estudada, dando origem a um eficiente algoritmo de reconhecimento.

A sequência canônica é também utilizada na solução do problema de determinação do caminho hamiltoniano. A subclasse dos grafos split-indiferença é revista, propiciando um exemplo interessante do problema da contagem de elementos de uma classe.

Finalmente, no Capítulo 5, os grafos estritamente cordais são estudados. Para essa classe, o grafo de cliques críticas tem um papel preponderante, sendo utilizado na solução eficiente de problemas de dominação. Tratamos de diversas subclasses: os grafos *AC*, os grafos potência de 3-folha e os grafos estritamente de intervalo, recentemente definidos. São analisados com detalhes: a definição de cada classe, a caracterização por subgrafos proibidos e por propriedades estruturais, o algoritmo de reconhecimento com complexidade linear. O relacionamento de diversas classes tratadas no texto encerra o capítulo.

A evolução da cadeia hierárquica das classes grafos é ilustrada ao final dos capítulos pertinentes. Ao término de cada capítulo, incluímos também uma lista de exercícios bem como notas bibliográficas referentes aos assuntos tratados.

Rio de Janeiro, 1 de setembro de 2016.

Lilian Markenzon  
Christina Fraga Esteves Maciel Waga

# Capítulo 1

## Conceitos Preliminares

Neste capítulo apresentamos conceitos fundamentais e terminologia de teoria de grafos, necessários à compreensão do texto. Além disso, apresentamos as noções de classe de grafos e, em especial, de grafos cordais.

Como pré-requisitos, esperamos do leitor certa familiaridade com a teoria de conjuntos e funções, bem como noções de combinatória elementar. São igualmente requeridos conhecimentos acerca de estruturas de dados básicas (listas em suas diversas modalidades) e complexidades de tempo e espaço de algoritmos. Todos os algoritmos são apresentados em pseudo-código, de forma que a implementação em linguagens de programação de alto nível se dê com relativa facilidade.

### 1.1 Grafos em geral

Um grafo *não orientado*, ou simplesmente *grafo*  $G$  é um par  $(V, E)$  ou  $(V_G, E_G)$ , onde  $V$  é o conjunto de *vértices* e  $E$  é o conjunto de *arestas*; cada aresta é um par não ordenado de vértices. Dada uma aresta  $\{v, w\} \in E$ , os vértices  $v$  e  $w$  são denominados *extremidades* da aresta; dizemos também que a aresta *incide* sobre os vértices  $v$  e  $w$ . Uma aresta com as extremidades iguais é denominada um *laço*.

Os grafos que consideramos neste texto são finitos (possuem número finito de vértices) e não têm laços. Usaremos as convenções  $n = |V|$  e  $m = |E|$  denominados, respectivamente, a *ordem* e o *tamanho* do grafo. Assumimos que os vértices são rotulados por letras ou números, a menos de indicação em contrário.

Dois vértices são *adjacentes* ou *vizinhos* quando são extremidades de uma aresta. Analogamente, duas arestas distintas incidentes ao mesmo vértice são chamadas *adjacentes*. O *conjunto de vizinhos* ou *vizinhança aberta* de um vértice  $v \in V$  é o conjunto  $N(v) = \{w \in V : \{v, w\} \in E\}$  e sua *vizinhança fechada* é  $N[v] = N(v) \cup \{v\}$ . Estes conceitos podem ser estendidos para um subconjunto de vértices  $Y \subseteq V$ ,  $N(Y) = \bigcup_{v \in Y} N(v)$  e  $N[Y] = \bigcup_{v \in Y} N[v]$ .

A cardinalidade  $N(v)$  é denominada o *grau* do vértice  $v$ , notado  $d(v)$ . O *grau máximo* de um grafo é  $\Delta(G) = \max_{v \in V} \{d(v)\}$ . Um vértice  $v$  é *universal* quando  $N[v] = V$  e é *pendente* se  $d(v) = 1$ . Um grafo é *k-regular* se todos os seus vértices têm grau igual a  $k$ .

Dois vértices  $u, v \in V$  são *gêmeos verdadeiros* em  $G$  se  $N[u] = N[v]$  e são *gêmeos falsos* se  $N(u) = N(v)$  e  $\{u, v\} \notin E$ .

O grafo  $G$  da Figura 1.1 possui 9 vértices e 13 arestas,  $\Delta(G) = 4$ , os vértices  $a$  e  $f$  são gêmeos falsos e os vértices  $e$  e  $i$  são gêmeos verdadeiros.

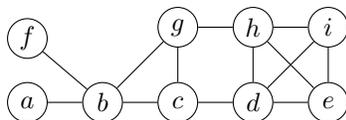


Figura 1.1: Grafo  $G$

Um grafo  $G' = (V', E')$  é *subgrafo* de  $G = (V, E)$  quando  $V' \subseteq V$  e  $E' \subseteq E$ . Dado um subconjunto de vértices  $Y \subseteq V$ , denominamos  $G[Y] = (Y, \{\{x, y\} \in E : x, y \in Y\})$  o *subgrafo de  $G$  induzido por  $Y$* . Em outras palavras,  $G[Y]$  é o subgrafo de  $G$  que tem  $Y$  como conjunto de vértices e possui todas as arestas de  $E$  com extremidades pertencentes a  $Y$ .

O conjunto  $C \subseteq V$  é uma *clique* quando  $G[C]$  for um *grafo completo*, isto é, o grafo que possui todas as arestas possíveis. Uma  $k$ -clique,  $k \geq 1$ , é uma clique de cardinalidade  $k$ . A *cardinalidade de uma clique máxima*, isto é, a de maior cardinalidade dentre todas as cliques de  $G$ , é denotada por  $\omega(G)$ . Um vértice  $v \in V$  é *simplicial* em  $G$  se  $N(v)$  é uma clique.

Um *caminho* em  $G = (V, E)$  é uma sequência  $\langle v_1, \dots, v_k \rangle$  de  $k \geq 1$  vértices distintos tal que para  $k > 1$  tem-se  $\{v_i, v_{i+1}\} \in E$ , para  $i = 1, \dots, k - 1$ . O *comprimento* do caminho é  $(k - 1)$ . Um *ciclo* é um caminho de comprimento maior ou igual a três em que somente o primeiro e o último vértices coincidem. Um grafo que não possui ciclos é dito *acíclico*. Uma *corda* é qualquer aresta incidente a dois vértices não consecutivos de um ciclo. Um grafo  $G$  é *conexo* se para todo par de vértices distintos  $u$  e  $v$  existe um caminho de  $u$  até  $v$ . Caso contrário, ele é chamado de *desconexo*. A *distância* entre os vértices  $u$  e  $v$ ,  $dist(u, v)$  ou  $dist_G(u, v)$ , é o menor caminho entre eles. A maior distância de um grafo conexo é denominada o *diâmetro* do grafo é denotada por  $diam(G)$ . Um caminho (ciclo) em  $G$  que passa por todas as arestas exatamente uma vez e admite repetição de vértices é chamado *euleriano*; um caminho (ciclo) em  $G$  que passa por todos os vértices exatamente uma vez é chamado *hamiltoniano*.

Um grafo  $G' = (V', E')$  é uma *componente* ou *componente conexa* de  $G = (V, E)$  se  $G'$  é *subgrafo conexo maximal* de  $G$ , isto é, não existe um grafo conexo  $G'' = (V'', E'')$  com  $V' \subset V'' \subset V$  e  $E' \subset E'' \subset E$ .

Diz-se que um grafo conexo  $G$  é um grafo  $k$ -*conexo* quando é necessário remover pelo menos  $k$  vértices para desconectá-lo mas a remoção de qualquer subconjunto com  $(k - 1)$  de vértices não o desconecta. Se  $k = 2$ ,  $G$  é denominado *biconexo*.

Uma *árvore*  $T$  é um grafo conexo e acíclico. Os vértices penderes de uma árvore são chamados *folhas*. Uma *subárvore* de um grafo conexo  $G$  é qualquer subgrafo de  $G$  que seja uma árvore. Uma *árvore geradora* de  $G$  é uma subárvore de  $G$  que possui todos os vértices de  $G$ .

Considere um subconjunto  $Y \subset V$  de vértices de um grafo  $G = (V, E)$ .  $Y$  é um *conjunto dominante* de  $G$  se  $N[Y] = V$ .  $Y$  é um *conjunto independente* de  $G$  se quaisquer dois vértices de  $Y$  não são adjacentes. Um conjunto independente é dito *maximal* se não houver um outro conjunto independente em  $G$  de maior cardinalidade que o contenha. O *número de independência* de  $G$ , denotado por  $\beta_0(G)$ , é a cardinalidade do maior conjunto independente maximal.

Na Figura 1.2 temos quatro subgrafos do grafo  $G$  da Figura 1.1. O grafo  $G_1 = G[\{a, b, d, g, i\}]$  é um subgrafo induzido,  $G_2$  é um subgrafo induzido 2-regular,  $G_3$  é uma clique, sendo que  $\omega(G) = 4$ , e  $G_4$  é uma árvore geradora de  $G$ . Por exemplo, o caminho  $\langle a, b, c, d, e \rangle$  nos dá que  $diam(G) = 4$  e  $\langle b, c, d, e, i, h, g, b \rangle$  é um ciclo do grafo. O grafo não possui caminhos hamiltonianos. Os conjuntos  $\{b, h\}$  e  $\{a, f, g, d\}$  são independentes maximais e  $\beta_0(G) = 4$ .

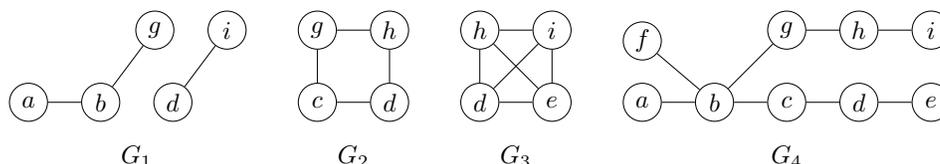


Figura 1.2: Subgrafos de  $G$

Dois grafos  $G = (V, E)$  e  $G' = (V', E')$  são *isomorfos* quando existe uma função bijetora  $\iota : V \rightarrow V'$  que preserva adjacência, isto é, para quaisquer  $u, v \in V$ ,  $\{u, v\} \in E$  se e somente se  $\{\iota(u), \iota(v)\} \in E'$ . A função  $\iota$  é denominada um *isomorfismo de grafos*. Os grafos da Figura 1.3 são isomorfos pois a função

$$\iota = \{(a, 1), (b, 2), (g, 3), (h, 4), (c, 5), (d, 6), (e, 7), (f, 8)\}$$

é um isomorfismo.

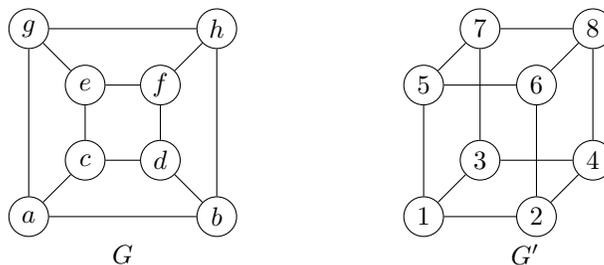


Figura 1.3: Grafos isomorfos

### 1.1.1 Representação de grafos

Um grafo  $G$  é definido através de um par de conjuntos,  $V$  de  $n$  vértices e  $E$  de  $m$  arestas. Uma representação de um grafo é uma forma de descrever sua estrutura e seus componentes. Naturalmente, a representação geométrica de um grafo  $G$  por meio de círculos (os vértices) ligados por segmentos de reta (as arestas), como o da Figura 1.1, é inadequada quando queremos resolver computacionalmente algum problema.

Dois estruturas muito utilizadas são as matrizes e os conjuntos de adjacência. Em ambas, os vértices do grafo são rotulados.

A *matriz de adjacência* de  $G$  é a matriz quadrada  $A$  de ordem  $n$  na qual a cada linha e a cada coluna se associa um vértice, e tal que:

$$A(i, j) = \begin{cases} 1 & \text{se } \{i, j\} \in E \\ 0 & \text{caso contrário.} \end{cases}$$

A representação de um grafo  $G = (V, E)$  por *conjuntos de adjacência* consiste em mencionar, para todo vértice  $v \in V$ , o conjunto  $N(v)$ , construindo um conjunto de pares da forma  $(v, N(v))$ .

Essas duas representações são clássicas e existem outras mais adequadas a problemas específicos em grafos.

Considere a seguinte correspondência dos vértices do grafo  $G$  da Figura 1.1:

$$\{(a, 1), (b, 2), (c, 3), (d, 4), (e, 5), (f, 6), (g, 7), (h, 8), (i, 9)\}$$

A matriz de adjacência é:

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Para o mesmo grafo teríamos a seguinte representação por conjuntos de adjacência:

$$\left\{ \begin{array}{lll} (a, \{b\}), & (b, \{a, c, f, g\}), & (c, \{b, d, g\}) \\ (d, \{c, e, h, i\}), & (e, \{d, h, i\}), & (f, \{b\}), \\ (g, \{b, c, h\}), & (h, \{d, e, g, i\}), & (i, \{d, e, h\}) \end{array} \right\}$$

Na Figura 1.4 vemos a implementação por listas encadeadas dessa representação.

## 1.2 Propriedades, suficiência e necessidade

Vejamos agora alguns conceitos mais gerais. Seja  $A$  um conjunto não vazio e  $\mathcal{P}$  uma propriedade unária sobre elementos. Se  $A$  possui a propriedade  $\mathcal{P}$ , diz-se que  $A$  é um  $\mathcal{P}$ -conjunto.

Considere um  $\mathcal{P}$ -conjunto  $A$ . Uma propriedade  $\mathcal{P}$  é *hereditária* se todo subconjunto  $A' \subseteq A$  é também um  $\mathcal{P}$ -conjunto e é *super-hereditária* quando todo superconjunto  $A''$ ,  $A \subseteq A''$ , é um  $\mathcal{P}$ -conjunto.

Um  $\mathcal{P}$ -conjunto  $A$  é *minimal* em relação à propriedade  $\mathcal{P}$  se todo subconjunto próprio  $A' \subseteq A$  não é um  $\mathcal{P}$ -conjunto e é *maximal* em relação à propriedade  $\mathcal{P}$  quando todo  $A''$ ,  $A \subseteq A''$ , não é um  $\mathcal{P}$ -conjunto.

Por exemplo, a propriedade  $\mathcal{P}$  para subconjuntos de vértices de um grafo  $G = (V, E)$ ,

$$\mathcal{P}: Y \text{ é um conjunto independente}$$

é hereditária para os subconjuntos de  $Y \subset V$ . Já a propriedade

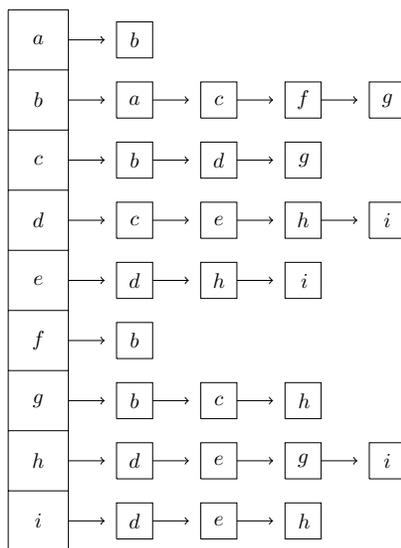


Figura 1.4: Implementação dos conjuntos de adjacência

$\mathcal{P}'$ :  $Y$  é um conjunto dominante

é super-hereditária para os superconjuntos de  $Y \subset V$ .

Propriedades são proposições básicas e podem ser conectadas formando novas proposições. Sejam  $\mathcal{P}$  e  $\mathcal{Q}$  propriedades. Classicamente temos cinco conectivos lógicos (um unário e quatro binários), que nos fornecem as proposições:  $(\neg\mathcal{P})$  (negação: não),  $(\mathcal{P} \wedge \mathcal{Q})$  (conjunção: e),  $(\mathcal{P} \vee \mathcal{Q})$  (disjunção: ou),  $(\mathcal{P} \rightarrow \mathcal{Q})$  (condicional: se então) e  $(\mathcal{P} \leftrightarrow \mathcal{Q})$  (bicondicional: se e somente se). A proposição condicional  $(\mathcal{P} \rightarrow \mathcal{Q})$  estabelece que a propriedade  $\mathcal{P}$  é a *condição suficiente* para que tenhamos a  $\mathcal{Q}$  e que  $\mathcal{Q}$  é a *condição necessária* dada a propriedade  $\mathcal{P}$ . Existe uma proposição diretamente relacionada à proposição condicional de grande importância, inclusive como uma técnica de demonstração, a proposição *contrapositiva*  $((\neg\mathcal{Q}) \rightarrow (\neg\mathcal{P}))$  do condicional. Sua importância decorre do fato dessas proposições, o condicional e a contrapositiva, serem (logicamente) equivalentes.

Podemos associar o condicional à inclusão de conjuntos. Consideremos  $\mathcal{P}$  e  $\mathcal{Q}$  propriedades sobre elementos de um conjunto  $A$  e a proposição

$$\alpha : \text{Para todo } x \in A, \text{ se } \mathcal{P}(x) \text{ então } \mathcal{Q}(x).$$

Podemos observar na Figura 1.5 o diagrama de Venn para esta situação. Pela proposição, todo elemento que possui a propriedade  $\mathcal{P}$  necessariamente satisfaz  $\mathcal{Q}$ . É possível a existência de um elemento  $y$  que tenha a propriedade  $\mathcal{Q}$  mas não a  $\mathcal{P}$ , e também um elemento  $x$  que não possua nem a propriedade  $\mathcal{P}$  nem a  $\mathcal{Q}$ .

Vamos ver um exemplo para grafos.

**Teorema 1.1.** *Se os grafos  $G$  e  $G'$  são isomorfos então eles têm o mesmo número de vértices e de arestas.*

Mas a recíproca não vale pois existem grafos com o mesmo o número de vértices e de arestas que não são isomorfos.

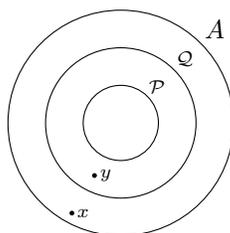


Figura 1.5: Inclusão no conjunto  $A$

Considere a proposição bicondicional ( $\mathcal{P} \leftrightarrow \mathcal{Q}$ ). A propriedade  $\mathcal{P}$  é uma *condição necessária e suficiente* para que a propriedade  $\mathcal{Q}$  seja satisfeita, dizemos que as propriedades são (logicamente) equivalentes. E mais, essa proposição nos fornece o esquema de enunciado para um *teorema de caracterização*. No diagrama de Venn os conjuntos  $\mathcal{P}$  e  $\mathcal{Q}$  seriam iguais. Um exemplo clássico de teorema de caracterização em grafos, apresentado por Euler em 1736, é visto a seguir.

**Teorema 1.2.** *Seja  $G$  um grafo conexo.  $G$  possui um ciclo euleriano se e somente se todos os vértices têm grau par.*

### 1.3 Classes de grafos

O conjunto de todos os grafos que satisfazem uma dada propriedade constitui uma *classe de grafos*. A importância de se definir novas classes reside no fato de que, enquanto muitos problemas para grafos em geral são difíceis de serem resolvidos computacionalmente, a solução para certas classes pode ser encontrada de maneira mais eficiente. Vejamos algumas classes bem conhecidas.

O grafo  $G$  com  $n$  vértices é dito *completo*, denotado por  $K_n$ , se é  $(n-1)$ -regular, isto é, se todo vértice é adjacente a cada um dos  $(n-1)$  outros vértices.

Um *grafo ciclo* é um grafo conexo em que todos os vértices têm grau igual a dois. Denotamos o grafo ciclo com  $n$  vértices por  $C_n$ .

Se todos os vértices de  $C_{n-1}$  forem adjacentes a um vértice  $v \notin C_{n-1}$ , temos o *grafo roda*  $W_n$  com  $n$  vértices.

Um *grafo bloco* é um grafo conexo onde cada componente biconexa é uma clique.

O *grafo caminho*  $P_n$  é uma árvore com exatamente duas folhas.

Um *caterpillar* (que conserva seu nome em inglês) é uma árvore na qual a remoção de todas as folhas resulta em um grafo caminho com  $n \geq 1$  vértices.

Um *grafo bipartido* é o grafo cujo conjunto de vértices pode ser particionado em dois subconjuntos  $V_1$  e  $V_2$  de forma que nenhuma aresta tenha ambas as extremidades no mesmo subconjunto. Ele é denotado por  $G = (V_1 \cup V_2, E)$ . Se cada vértice de  $V_1$  é adjacente a cada vértice de  $V_2$ ,  $G$  é dito um *grafo bipartido completo*  $K_{k,\ell}$  sendo  $k = |V_1|$  e  $\ell = |V_2|$ .

O *grafo estrela*  $S_n$  com  $n$  vértices é um grafo bipartido cujo conjunto  $V_1$  possui um único vértice e este vértice é adjacente a todos os vértices em  $V_2$ , isto é,  $S_n = K_{1,n-1}$ .

Existem, na literatura, diversas outras classes sendo estudadas, tais como os *grafos planares*, aqueles que podem ser desenhados sobre uma superfície plana sem

que as arestas se cruzem, e os *grafos hamiltonianos*, aqueles que possuem um ciclo simples contendo todos os vértices do grafo.

A *definição* de uma classe consiste em mencionar a propriedade satisfeita pelos grafos que a constituem, como fizemos nesses exemplos. Uma *caracterização* (na verdade um teorema de caracterização) de uma classe de grafos é uma propriedade alternativa porém equivalente a propriedade apresentada na definição. Tradicionalmente, as caracterizações das classes são feitas proibindo-se a presença de certos subgrafos. Dado um conjunto  $\mathcal{H}$  de grafos, um grafo  $G$  é  $\mathcal{H}$ -livre ou *livre de  $\mathcal{H}$*  quando  $G$  não contém nenhum elemento de  $\mathcal{H}$  como subgrafo. Dizemos, assim, que os grafos de  $\mathcal{H}$  são os *subgrafos proibidos* para a classe dos grafos  $\mathcal{H}$ -livres.

Outros aspectos são especialmente relevantes quando se prioriza um enfoque algorítmico: o estabelecimento de propriedades estruturais específicas para a solução de problemas, a contagem e enumeração de seus elementos e o relacionamento com outras classes.

Quando uma nova classe de grafos é definida, o *problema do reconhecimento* da pertinência de um elemento é fundamental. Em termos computacionais, trata-se da obtenção de um algoritmo para verificar se um grafo dado pertence a essa classe. Tal reconhecimento nem sempre é possível em tempo polinomial. Se a classe é sabidamente a interseção de outras que possuam algoritmos de reconhecimento conhecidos, duas abordagens podem ser consideradas:

**Abordagem 1:** Verificar se o grafo pertence às classes conhecidas.

**Abordagem 2:** Desenvolver um algoritmo específico para a nova classe.

A Abordagem 2 nos parece mais desafiadora pois a ideia é apresentar uma caracterização que permita desenvolver um algoritmo no mínimo mais simples, comparado aos encontrados na literatura.

Outra questão interessante é como certas classes de grafos se relacionam, isto é, dadas duas classes distintas, devemos decidir se uma contém a outra, se são disjuntas ou quais são os grafos pertencentes à interseção. Podemos, desta forma, estabelecer hierarquias por inclusão de classes.

Deve-se considerar também os *problemas da enumeração e contagem* dos grafos de um determinada classe. Exemplificando, se quisermos todos os grafos com  $n$  vértices que possuam a propriedade  $\mathcal{P}$  poderíamos desenvolver um algoritmo “força bruta”: gerar todos os grafos com  $n$  vértices, verificar quais possuem a propriedade  $\mathcal{P}$  e contá-los. Esse processo tem, em geral, tempo exponencial. É, então, desejável estabelecer um método direto para essas tarefas. Como curiosidade, o número de caterpillars (não rotulados) com  $n$  vértices é dado pela fórmula  $2^{n-4} + 2^{\lfloor \frac{n-4}{2} \rfloor}$ .

## 1.4 Grafos cordais

Apresentamos uma definição clássica de grafos cordais. Seja  $\mathcal{F}$  uma família finita de conjuntos não vazios. O *grafo de interseção* de  $\mathcal{F}$  é obtido representando cada conjunto da família por um vértice e determinando a existência de uma aresta quando os conjuntos correspondentes aos vértices das extremidades se interceptam.

Agora, vamos considerar uma árvore e  $\mathcal{F}$  um conjunto de subárvores desta. Vamos estabelecer que a cada subárvore fica associado um vértice, e dois vértices são adjacentes quando as duas subárvores correspondentes têm alguma interseção.

Naturalmente não são consideradas as interseções da subárvore consigo mesma pois caso contrário teríamos laços no grafo. O grafo obtido é um grafo de interseção de uma família de subárvores e é denominado *grafo de subárvores*.

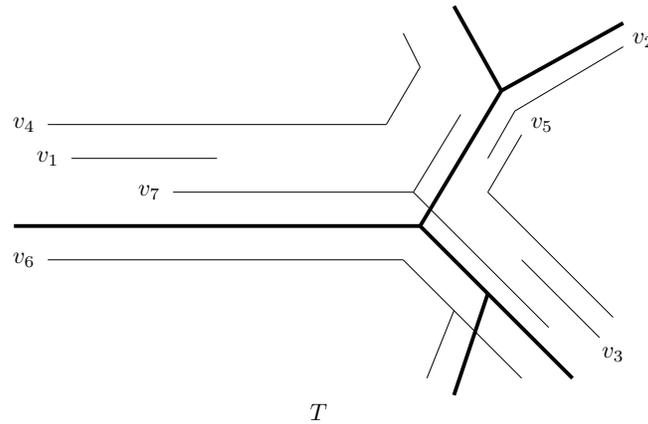


Figura 1.6: Árvore e as subárvores

Vejamos um exemplo na Figura 1.6, uma árvore  $T$  com 7 subárvores, já rotuladas por  $v_1, \dots, v_7$ . Na Figura 1.7 temos a tabela com as interseções entre as subárvores que corresponde à matriz de adjacência do grafo de subárvores ao lado.

$\cap$	$v_1$	$v_2$	$v_3$	$v_4$	$v_5$	$v_6$	$v_7$
$v_1$	0	0	0	1	0	1	1
$v_2$	0	0	0	1	1	0	1
$v_3$	0	0	0	0	1	1	1
$v_4$	1	1	0	0	1	1	1
$v_5$	0	1	1	1	0	1	1
$v_6$	1	0	1	1	1	0	1
$v_7$	1	1	1	1	1	1	0

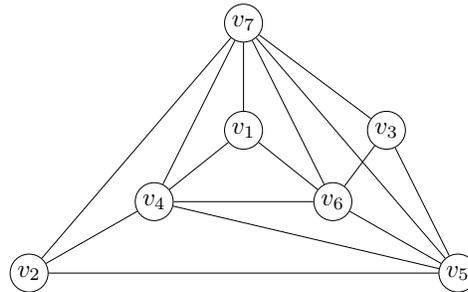


Figura 1.7: Interseções e grafo de subárvores

Um grafo  $G = (V, E)$  é um *grafo cordal* quando todo ciclo de tamanho maior do que três possui uma corda. Grafos cordais foram também denominados na literatura *triangularizados* ou de *circuito rígido*.

O conceito de grafos de interseção nos fornece uma primeira caracterização para a classe dos grafos cordais.

**Teorema 1.3.** *O grafo  $G$  é cordal se e somente se é um grafo de subárvores.*

Nos próximos capítulos estudaremos a classe dos grafos cordais, suas propriedades e algumas subclasses.

## 1.5 Exercícios

1. Considere o grafo  $G$  da Figura 1.8.
  - (a)  $G$  possui gêmeos falsos? E verdadeiros?
  - (b) Quais são os vértices simpliciais?
  - (c) Determine os invariantes  $\Delta(G)$ ,  $\omega(G)$ ,  $diam(G)$  e  $\beta_0(G)$ .
  - (d) Existe algum ciclo ou caminho euleriano em  $G$ ? E hamiltoniano?
  - (e) Indique uma árvore geradora.
  - (f)  $G$  possui o grafo  $K_{1,3}$  como subgrafo induzido? E uma estrela  $S_4$ ?
  - (g) Indique conjuntos distintos de vértices tais que a remoção desconecte o grafo.
  - (h) Indique dois conjuntos dominantes minimais de cardinalidades diferentes.

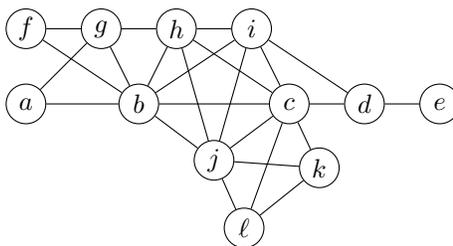


Figura 1.8: Grafo  $G$

2. Indique verdadeiro ou falso, justificando ou apresentando um contraexemplo.
  - (a) Seja  $G = (V, E)$  um grafo. Dado  $n$  é sempre possível determinar  $m$ .
  - (b) Se  $G$  é um grafo conexo então todos os subgrafos induzidos são conexos.
  - (c) A classe dos grafos eulerianos e a dos hamiltonianos é disjunta.
  - (d) A matriz de adjacência de  $G$  é única a menos de permutação de linhas.
  - (e) A relação de isomorfismo entre grafos é um relação de equivalência.
  - (f) Um grafo  $G$  é bipartido se e somente se  $G$  não contém ciclos de comprimento ímpar.
  - (g) Em qualquer componente conexa de um grafo  $G$ , seus vértices pertencem a um ciclo.
  - (h) Em uma árvore, o caminho entre dois vértices é único.
  - (i) Seja  $G$  um grafo conexo. Então  $\beta_0(G) \leq \omega(G)$ .
3. Apresente todas os grafos caterpillar não rotulados com 8 vértices.
4. Apresente vantagens e desvantagens das representações de grafos por matriz de adjacência e por conjuntos de adjacência.
5. Complete a tabela de inclusão entre classes usando  $=$ ,  $\subset$  e  $\not\subset$  (a quarta linha já foi preenchida, como exemplo).

	árvore	bipartido	bloco	caminho	caterpillar	ciclo	completo	cordal
árvore								
bipartido								
bloco								
caminho	$\subset$	$\subset$	$\subset$	$=$	$\subset$	$\not\subset$	$\not\subset$	$\subset$
caterpillar								
ciclo								
completo								
cordal								

## 1.6 Notas bibliográficas

Os livros de Diestel [19], Gross e Yellen [31] e West [75] abordam, de forma ampla, a teoria de grafos; em português, podemos citar Boaventura [60]. Já os livros de McHugh [59] e Szwarcfiter [71] são mais específicos para aspectos referentes a algoritmos em grafos. O livro do Spinrad [70] é um clássico para representações de grafos.

O livro de Szwarcfiter e Markenzon [72] abrange o tópico estruturas de dados de forma clara e detalhada. Em complexidade de algoritmos, dois livros são fundamentais: Garey e Johnson [26] e Brassard e Bratley [12].

Brandstädt *et al.* [9] é um excelente texto sobre classes de grafos e suas propriedades. O site *Information System on Graph Classes and their Inclusions* é uma enciclopédia de classes de grafos com uma aplicação de consulta para pesquisa em classes de grafos particulares ([www.graphclasses.org](http://www.graphclasses.org)).

A classe dos grafos cordais é uma das mais extensamente estudadas em teoria de grafos, havendo vasta literatura disponível a seu respeito. Referências essenciais são o artigo de Blair e Peyton [3] e o livro de Golumbic [29]. Historicamente devemos citar a caracterização de grafos cordais por grafos de subárvores de Gavril [27].

## Capítulo 2

# Ferramentas Estruturais

Neste capítulo apresentamos as *ferramentas estruturais* que serão utilizadas nos próximos capítulos. Reunimos sob esse título estruturas diversas com as quais será possível estabelecer, de uma forma mais sistemática, as propriedades das classes estudadas em nosso texto.

As ferramentas são definidas expressamente para os grafos cordais, tendo em vista sua eficiência quando aplicadas a esta classe. Dois conceitos são fundamentais para o desenvolvimento das ferramentas estruturais – clique maximal e separador minimal de vértices – e estes permeiam todo o capítulo. Dentre as ferramentas escolhidas, temos: grafo de interseção de cliques, árvore de cliques, grafo de cliques reduzido, grafo de cliques críticas e esquemas de eliminação de vértices. Além destas, fazemos uma revisão no conceito de subgrafo proibido.

### 2.1 Sequências de vértices

Uma *sequência* é uma lista de números ou objetos considerados em uma ordem especial. A cada termo de uma sequência é atribuído um número de 1 a  $n$  definindo a posição (índice) do termo na sequência.

Vamos ver ao longo desse texto a enorme importância de sequências de vértices no desenvolvimento de propriedades teóricas e algorítmicas de grafos cordais. Nos concentraremos neste capítulo em sequências que levam o nome de *esquemas de eliminação*. Uma sequência  $\langle v_1, \dots, v_n \rangle$  de vértices de um grafo  $G$  é um *esquema de eliminação* satisfazendo a propriedade  $\mathcal{P}$  se para todo  $i \in \{1, \dots, n\}$  o vértice  $v_i$  tem a propriedade  $\mathcal{P}$  no grafo restante  $G_i = G[\{v_i, \dots, v_n\}]$ .

Nessa seção, alguns esquemas de eliminação são apresentados: o esquema de eliminação perfeita, o esquema de eliminação bicompatível, o esquema de eliminação simples e o esquema de eliminação forte. Nos capítulos que se seguem, outras sequências de vértices serão apresentadas e suas propriedades estudadas.

#### 2.1.1 Esquema de eliminação perfeita

Seja  $G = (V, E)$  um grafo e  $\sigma = \langle v_1, \dots, v_n \rangle$  uma sequência de vértices. Dizemos que  $\sigma$  é um *esquema de eliminação perfeita* (*ep*) se cada  $v_i$  é um vértice simplicial no subgrafo induzido  $G[\{v_i, \dots, v_n\}]$ . Por abuso de notação, podemos dizer que

$\sigma(i) = v_i$  e, da mesma forma, a posição de um vértice  $v$  na sequência é dada por  $\sigma^{-1}(v)$ .

O esquema de eliminação perfeita deve sua importância à caracterização de grafos cordais.

**Teorema 2.1.** *Um grafo  $G$  é cordal se e somente se  $G$  tem um esquema de eliminação perfeita.*

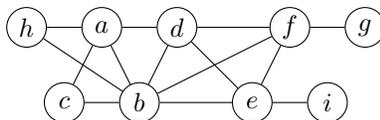


Figura 2.1: Grafo cordal  $G_1$

Pela definição, é fácil determinar um *eep* para um dado grafo. Inicia-se com uma sequência de vértices vazia. A cada passo, um vértice simplicial é escolhido, acrescentado ao fim da sequência e excluído do grafo. Ao final do processo, obtemos um *eep*. Esquemas de eliminação perfeita não são únicos para um certo grafo  $G$ . Tomemos como exemplo o grafo  $G_1$  da Figura 2.1. Dois possíveis esquemas de eliminação perfeita são:

$$\langle h, c, i, g, a, b, d, f, e \rangle \text{ e } \langle c, h, a, d, b, g, i, e, f \rangle.$$

Para determinar um *eep* de um grafo cordal de maneira eficiente, utilizamos percursos especiais, como o percurso de cardinalidade máxima e o percurso em largura lexicográfica. Um *percurso* sobre um grafo é um procedimento sistemático de varredura, que consiste em explorar exatamente uma vez cada um dos elementos presentes na estrutura. O critério de escolha do elemento a ser visitado é estabelecido de maneira diversa para cada tipo de percurso.

No *percurso em largura lexicográfica*, o critério de escolha depende de um rótulo atribuído aos vértices, que armazena os índices dos vizinhos de  $v$  já explorados até o momento. A cada passo, é escolhido como vértice a visitar um dentre os que possuam o maior rótulo. Esses são ordenados lexicograficamente (ordenação também conhecida como ordem do dicionário), por exemplo,  $7641 < 77$  e  $542 < 5421$ . A construção do rótulo de cada vértice utiliza outra sequência,  $label[v]$ . Note que o *eep* é construído do fim para o início, isto é, quando o vértice  $v$  é escolhido, ele é acrescentado ao início da sequência já obtida. Ao fim da execução, a variável  $index(v)$  indica a posição de vértice na sequência.

O símbolo  $||$  representa a concatenação entre sequências.



Evidentemente, dado um grafo, existem diferentes percursos em largura lexicográfica, pois, em alguns passos, pode ocorrer que mais de um vértice tenha o mesmo rótulo.

Para maior eficiência do Algoritmo 2.1, sua implementação emprega uma estrutura de dados específica. O conjunto  $V'$  deve ser mantido em uma estrutura ortogonal, constituída por uma lista “vertical” duplamente encadeada, cujos elementos representam valores distintos de *label*, ordenados decrescentemente. De cada nó da lista vertical, parte uma lista “horizontal”, também duplamente encadeada, de vértices que possuem o mesmo valor de *label*. Apenas os nós correspondentes a valores de *label* sendo utilizados estão presentes na lista vertical. Utilizando essa estrutura de armazenamento o algoritmo tem complexidade de tempo de  $O(n + m)$ .

É fácil observar que a apresentação do *eep* de um grafo cordal não é suficiente para representá-lo. Outros conceitos serão necessários. Dado um grafo cordal  $G$ , um *eep*  $\sigma$  e  $v \in V$ , o conjunto

$$X_\sigma(v) = \{w \in N(v) : \sigma^{-1}(w) > \sigma^{-1}(v)\}$$

é denominado *conjunto de adjacência monótona* de  $v$  relativo ao *eep*  $\sigma$ . Observe que, para todo vértice  $v \in V$ , os elementos do conjunto  $X_\sigma(v)$  são exatamente os elementos da clique à qual o vértice  $v$  é ligado no decorrer da construção do *eep*  $\sigma$  (a sequência *label* no algoritmo).

Agora sim, podemos definir uma representação para um grafo cordal. Seja  $G$  um grafo cordal e  $\sigma = \langle v_1, \dots, v_n \rangle$  um *eep* de  $G$ . A *representação por conjuntos de adjacência monótona* de  $G$  consiste em uma sequência de pares

$$\mathcal{R}_\sigma(G) = \langle (v_1, X_\sigma(v_1)), \dots, (v_n, X_\sigma(v_n)) \rangle.$$

Por exemplo, utilizando o *eep*  $\langle g, i, f, e, h, d, c, b, a \rangle$ , a representação do grafo  $G_1$ , visto na Figura 2.1, é:

$$\mathcal{R}_\sigma(G_1) = \langle (g, \{f\}), (i, \{e\}), (f, \{b, d, e\}), (e, \{b, d\}), (h, \{a, b\}), (d, \{a, b\}), (c, \{a, b\}), (b, \{a\}), (a, \emptyset) \rangle.$$

Esta estrutura sugere um processo de construção para grafos cordais: iniciando com um grafo trivial (composto apenas pelo vértice  $v_n$ ), acrescenta-se, no  $i$ -ésimo passo, um novo vértice  $v_{n+1-i}$  ligado à clique  $X_\sigma(v_{n+1-i})$  do grafo já existente.

Na realidade, o percurso em largura lexicográfica pode ser aplicado a um grafo qualquer, obtendo uma sequência de vértices. Pelo Teorema 2.1 sabemos que essa sequência será um esquema de eliminação perfeita se e somente se o grafo for cordal. Então é possível reconhecer se um grafo é cordal da seguinte forma:

**Passo 1:** Aplicar o percurso em largura lexicográfica determinando uma sequência de vértices.

**Passo 2:** Verificar se esta sequência é um *eep*.

Para implementar o Passo 2 bastaria percorrer a sequência de vértices e testar, para cada um, se seu conjunto de adjacência monótona é uma clique. Se executada desta forma, a implementação do Passo 2 teria complexidade de tempo de  $O(nm)$ , o que pode ser evitado.

O algoritmo que se segue testa se uma sequência de vértices é um esquema de eliminação perfeita em tempo linear, utilizando um artifício interessante. Seja a sequência  $\sigma = \langle v_1, \dots, v_n \rangle$ . Quando o vértice  $v_i$  é analisado, precisaríamos analisar todos os vértices de seu conjunto de adjacência monótona para o teste de clique. Em vez disso, o algoritmo determina o vértice de  $X_\sigma(v_i)$  mais próximo de  $v_i$  na sequência, por exemplo  $v_j$ , e “atrasa” o teste de adjacência dos outros elementos para quando  $v_j$  for analisado. Esse “atraso” permite que cada adjacência seja testada somente uma vez.

Para implementar essa ideia, o algoritmo utiliza um conjunto de vértices auxiliar,  $A(v)$ , para cada vértice do grafo. Na realidade, esse conjunto é um multiconjunto, uma vez que o mesmo vértice pode ser copiado de diversos conjuntos de adjacência.

---

**Algoritmo 2.2.** *Testar- $eep$* 


---

**Entrada:** grafo cordal  $G = (V, E)$  e sequência  $\sigma = \langle v_1, \dots, v_n \rangle$ ;  
**Saída:**  $\sigma$  é  $eep$  ou não;  
**Início**  
 $eep \leftarrow true$ ;  
**Para**  $v \in V$  **faça**  $A(v) \leftarrow \emptyset$ ;  
**Para**  $i = 1, \dots, n - 1$  **faça**  
 $v \leftarrow \sigma(i)$ ;  
 $X \leftarrow \{x \in N(v) : \sigma^{-1}(v) < \sigma^{-1}(x)\}$ ;     % adjacência monótona de  $v$   
**Se**  $X \neq \emptyset$  **então**  
 $u \leftarrow \sigma(\min\{\sigma^{-1}(x) : x \in X\})$ ;     % determinar vértice próximo  
 $A(u) \leftarrow A(u) \cup \{v\}$ ;  
**Se**  $A(v) \setminus N(v) \neq \emptyset$  **então**  
 $eep \leftarrow false$ ;  
**Fim.**

---

A complexidade de tempo do Algoritmo 2.2 é

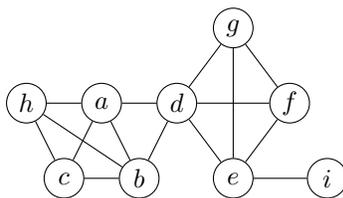
$$|V| + \sum_{v \in V} N(v) + \sum_{u \in V} A(u).$$

Como cada conjunto de adjacência  $N(v)$  é percorrido uma vez e como eles aparecem como parte dos conjuntos  $A(u)$ , ambos os somatórios podem ser substituídos por  $m$ . O reconhecimento de um grafo cordal tem então complexidade de  $O(n + m)$ .

### 2.1.2 Esquema de eliminação bicompatível

Um  $eep$   $\sigma = \langle v_1, \dots, v_n \rangle$  de um grafo cordal  $G = (V, E)$  é um *esquema de eliminação bicompatível* ( $eeb$ ) quando a sequência na ordem inversa  $\sigma^{-1}$  é também um  $eep$ .

O esquema de eliminação bicompatível existe para grafos de intervalo próprio e suas subclasses, como veremos no Capítulo 4. O grafo  $G_2$ , visto na Figura 2.2, tem o esquema bicompatível  $\sigma = \langle c, h, a, b, d, g, f, e, i \rangle$  pois  $\sigma^{-1} = \langle i, e, f, g, d, b, a, h, c \rangle$  é também um  $eep$ . Observe que, mesmo para classes em que esta propriedade acontece, nem todo  $eep$  é um  $eeb$ . Por exemplo,  $\sigma' = \langle h, i, c, a, f, e, b, g, d \rangle$  é um  $eep$  mas sua inversa não, e conseqüentemente,  $\sigma'$  não é um  $eeb$ .

Figura 2.2: Grafo cordal  $G_2$ 

### 2.1.3 Outros esquemas de eliminação

Os vértices  $u$  e  $v$  são *compatíveis* no grafo  $G$  se  $N[u] \subseteq N[v]$  ou  $N[v] \subseteq N[u]$ . Um vértice  $v$  é *simples* se os vértices de  $N(v)$  são compatíveis ou, alternativamente, o conjunto  $\{N[u] : u \in N(v)\}$  pode ser linearmente ordenado por inclusão de conjuntos. Note que um vértice simples é simplicial.

Uma sequência de vértices  $\langle v_1, \dots, v_n \rangle$  é um *esquema de eliminação simples* se para cada  $1 \leq i \leq n$ , o vértice  $v_i$  é simples em  $G[v_i, \dots, v_n]$ .

O esquema de eliminação simples é utilizado na caracterização de *grafos fortemente cordais*. Está provado que um grafo é *fortemente cordal* se e somente se tem um esquema de eliminação simples. Entretanto, esse não é o único esquema de eliminação que caracteriza essa classe.

Uma sequência de vértices  $\langle v_1, \dots, v_n \rangle$  é chamada um *esquema de eliminação forte* se é um esquema de eliminação simples e para quaisquer  $i < j < k$ , sendo  $v_j, v_k \in N(v_i)$ , tem-se  $N[v_j] \subseteq N[v_k]$  em  $G[v_i, \dots, v_n]$ .

Pode-se também afirmar que um grafo é fortemente cordal se e somente se tem um esquema de eliminação forte.

Das definições conclui-se que um esquema de eliminação forte é simples, mas a recíproca nem sempre é verdadeira.

Por exemplo, o grafo  $G_2$ , da Figura 2.2, tem um esquema de eliminação perfeita que é um esquema de eliminação simples e também forte:  $\sigma = \langle h, c, i, a, b, f, g, e, d \rangle$ . Entretanto, o *esp*  $\sigma' = \langle g, f, i, e, d, a, b, h, c \rangle$  não é um esquema de eliminação simples porque o primeiro vértice da sequência  $g$  não é simples. Seu conjunto de adjacência é  $N(g) = \{f, d, e, g\}$ ; os vértices  $d$  e  $e$  não são compatíveis pois  $N[d] = \{a, b, d, e, f, g\} \not\subseteq N[e] = \{d, e, f, g, i\}$  e  $N[e] \not\subseteq N[d]$ .

## 2.2 Cliques maximais e grafo de cliques

Vamos rever o conceito de cliques em grafos cordais e mostrar como essa estrutura é poderosa para a classe. Seja  $G = (V, E)$  um grafo cordal. O conjunto  $Q \subseteq V$  é uma *clique* do grafo  $G$  quando  $G[Q]$  é um subgrafo completo de  $G$ . A clique  $Q$  é *maximal* quando não existe uma clique  $Q'$  em  $G$  tal que  $Q \subseteq Q'$  e é *máxima* em  $G$  quando para toda clique  $Q'$  de  $G$ ,  $|Q'| \leq |Q|$ . Denotamos por  $\mathbb{Q}$  o conjunto de todas as cliques maximais de um grafo cordal  $G$ . No grafo  $G_1$ , da Figura 2.1, temos 6 cliques maximais e

$$\mathbb{Q} = \{ \{a, b, c\}, \{a, b, h\}, \{a, b, d\}, \{b, d, e, f\}, \{f, g\}, \{e, i\} \}.$$

Vale ressaltar alguns tipos de cliques maximais. Uma *clique simplicial* é uma clique maximal que contém pelo menos um vértice simplicial. Uma clique simplicial  $Q$  é dita uma *clique limítrofe* quando existe uma outra clique maximal  $Q'$  tal que  $Q \cap Q'$  é o conjunto de vértices não simpliciais de  $Q$ . É possível classificar os vértices segundo sua pertinência a cliques maximais.

**Teorema 2.2.** *Um vértice é simplicial em um grafo cordal  $G$  se e somente se pertence a exatamente uma clique maximal.*

No grafo  $G_2$  da Figura 2.2, as cliques  $\{a, b, c, h\}$  e  $\{e, i\}$  são cliques limítrofes; a clique  $\{d, e, f, g\}$  é simplicial porém não é limítrofe.

O Algoritmo 2.3, que determina as cliques maximais de um grafo cordal  $G$ , percorre um esquema de eliminação perfeita de  $G$  do fim para o início, efetuando uma construção indutiva das cliques. Ao se considerar o vértice  $v_i$ , sua adjacência monótona é analisada: se ela já existe no grafo construído até o momento, a clique maximal é aumentada; caso contrário, uma nova clique é criada.

---

**Algoritmo 2.3.** *Determinar-cliques-maximais*

---

**Entrada:** grafo cordal  $G = (V, E)$  e eep  $\sigma = \langle v_1, \dots, v_n \rangle$ ;  
**Saída:**  $\mathbb{Q}$ ;  
**Início**  
 $q \leftarrow 1$ ;  $p \leftarrow 0$ ;  $\mathbb{Q} \leftarrow \emptyset$ ;  
 $Q_1 \leftarrow \{v_n\}$ ;  $\eta(v_n) \leftarrow 1$ ;  
**Para**  $i = n - 1, \dots, 1$  **faça**  
 $X_\sigma(v_i) \leftarrow \{u \in N(v_i) : \sigma^{-1}(u) > \sigma^{-1}(v)\}$ ;  
 $w \leftarrow z \in X_\sigma(v_i)$  tal que  $\sigma^{-1}(z)$  é mínimo;  
 $k \leftarrow \eta(w)$ ;  
**Se**  $|X_\sigma(v_i)| < |Q_k|$  **então**  
 $q \leftarrow q + 1$ ;  
 $Q_q \leftarrow \{v_i\} \cup X_\sigma(v_i)$ ;      % nova clique maximal  $Q_q$   
 $\mathbb{Q} \leftarrow \mathbb{Q} \cup Q_q$ ;  
 $\eta(v_i) \leftarrow q$ ;  
**senão**  
 $Q_k \leftarrow \{v_i\} \cup Q_k$ ;      % clique  $Q_k$  é aumentada  
 $\eta(v_i) \leftarrow k$ ;  
**Fim.**

---

A complexidade de tempo do Algoritmo 2.3 é  $O(n + m)$ , tendo em vista que o algoritmo percorre um esquema de eliminação perfeita e analisa o conjunto de adjacência de cada vértice. Como a nova clique tem somente a sua cardinalidade comparada com a de uma clique já existente, a complexidade não se altera.

Vamos observar o acompanhamento do Algoritmo 2.3, aplicado ao grafo  $G_1$  da Figura 2.1, sendo  $\sigma = \langle g, i, f, e, h, d, c, b, a \rangle$ , na tabela que se segue. Após a inicialização, temos, quando  $i = 7$ , a formação da clique maximal  $Q_1 = \{a, b, c\}$ . Quando  $i = 6$ ,  $|X_\sigma(d)|$  é menor do que  $|Q_1|$ , o que implica na criação de uma nova clique. Outras cliques maximais surgem nas iterações  $i = 5$ ,  $i = 4$ ,  $i = 2$  e  $i = 1$ .

$i$	$v_i$	$X_\sigma(v_i)$	$w$	$k$	$q$	$clique$	$\eta(v_i)$
					1	$Q_1 = \{a\}$	1
8	$b$	$\{a\}$	$a$	1		$Q_1 = \{a, b\}$	1
7	$c$	$\{a, b\}$	$b$	1		$Q_1 = \{a, b, c\}$	1
6	$d$	$\{a, b\}$	$b$	1	2	$Q_2 = \{a, b, d\}$	2
5	$h$	$\{a, b\}$	$b$	1	3	$Q_3 = \{a, b, h\}$	3
4	$e$	$\{b, d\}$	$d$	2	4	$Q_4 = \{b, d, e\}$	4
3	$f$	$\{b, d, e\}$	$e$	4		$Q_4 = \{b, d, e, f\}$	4
2	$i$	$\{e\}$	$e$	4	5	$Q_5 = \{e, i\}$	5
1	$g$	$\{f\}$	$f$	4	6	$Q_6 = \{f, g\}$	6

Uma observação importante, o conjunto  $\mathbb{Q}$  também é uma representação para um grafo cordal  $G$ . O Teorema 2.3 mostra que o armazenamento de  $\mathbb{Q}$  é bastante eficiente.

**Teorema 2.3.** *Seja  $G = (V, E)$  um grafo cordal e  $\mathbb{Q} = \{Q_1, \dots, Q_q\}$  seu conjunto de cliques maximais. Então a complexidade de espaço de  $\sum_{i=1}^q |Q_i|$  é  $O(n + m)$ .*

Demonstração:

Observamos que, a cada iteração do Algoritmo 2.3, é determinado o conjunto de adjacência monótona de um vértice. Então,  $\sum_{j=1}^n (v_j, X_\sigma(v_j)) = n + m$ . Ora, as cliques maximais constituem alguns pares  $(v_j, X_\sigma(v_j))$ . Então  $\sum_{i=1}^q |Q_i| \leq n + m$ . ■

A sequência de cliques  $\langle Q_1, \dots, Q_q \rangle$ , construída pelo Algoritmo 2.3, pode ser utilizada na construção indutiva do grafo. Essa sequência terá diversas aplicações no decorrer de nosso texto.

O *grafo de interseção de cliques* ou *grafo de cliques* de um grafo  $G$ ,  $C(G)$ , é o grafo cujos vértices são as cliques maximais de  $G$  e cujas arestas ligam vértices que correspondem a cliques não disjuntas. Assim,  $V_{C(G)} = \mathbb{Q}$  e  $\{Q, Q'\} \in E_{C(G)}$  quando  $Q \neq Q'$  e  $Q \cap Q' \neq \emptyset$ . A Figura 2.3 mostra o grafo de cliques do grafo  $G_1$ , apresentado na Figura 2.1.

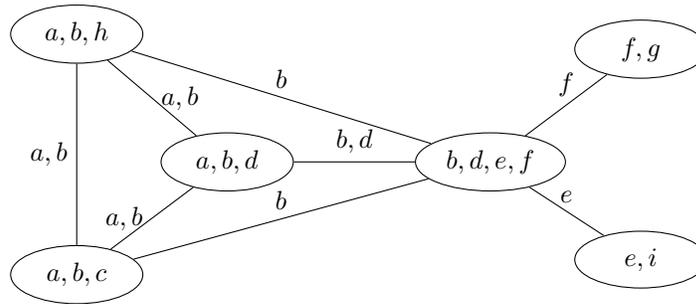


Figura 2.3: Grafo de cliques  $C(G_1)$

A construção do grafo de cliques  $C(G)$  não é muito eficiente, uma vez que é necessário estabelecer todas as interseções entre as cliques maximais. Existem  $n^2$  pares de cliques maximais; estabelecer a interseção em cada par tem complexidade de tempo de  $O(n)$ , uma vez que cada clique maximal tem  $O(n)$  vértices. O algoritmo tem então complexidade de tempo de  $O(n^3)$ . Felizmente, como veremos

nas próximas seções, temos soluções mais simples para trabalhar com as cliques maximais.

É importante ressaltar que existem grafos de cliques de grafo cordais que não são eles mesmos grafos cordais. A Figura 2.4 mostra o grafo  $G_3$  e  $C(G_3)$ . Observe que  $C(G_3)$  não cordal.

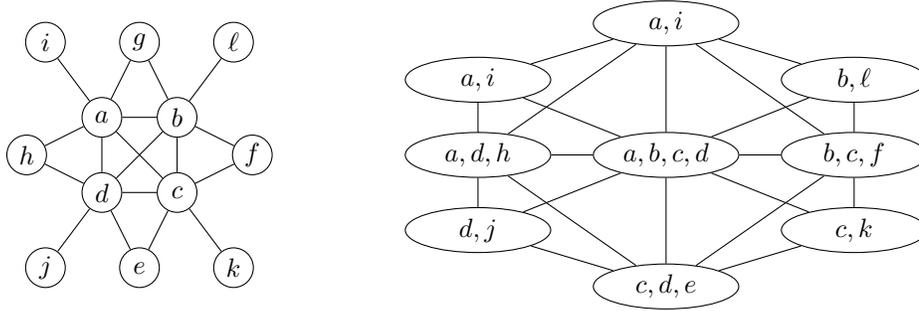


Figura 2.4: Grafo  $G_3$  e  $C(G_3)$

### 2.3 Separadores minimais e árvore de cliques

Em grafos cordais, trabalhamos com dois tipos de conjuntos de vértices separadores: o conceito clássico, de separador do grafo, e também o conceito de separador de vértices, mais utilizado para a classe.

Um subconjunto  $S \subset V$  é um *separador* de  $G$  se ao menos dois vértices da mesma componente conexa de  $G$  estão em componentes distintas de  $G[V \setminus S]$ . O conjunto  $S$  é um *separador minimal* de  $G$  se  $S$  é um separador e nenhum subconjunto próprio de  $S$  separa o grafo.

O conjunto  $S$  é um *separador de vértices* para vértices não adjacentes  $u$  e  $v$  (um *uv-separador*) se a remoção de  $S$  do grafo separa  $u$  e  $v$  em componentes conexas distintas. Se nenhum subconjunto próprio de  $S$  é um *uv-separador* então  $S$  é um *uv-separador minimal*. Se  $S$  é um *uv-separador minimal* para algum par de vértices  $u$  e  $v$ , ele é chamado *separador minimal de vértices (smv)*. Um separador minimal do grafo é sempre um separador minimal de vértices mas a recíproca não é verdadeira. Denotamos por  $\mathbb{S}$  o conjunto de separadores minimais de vértices de  $G$ .

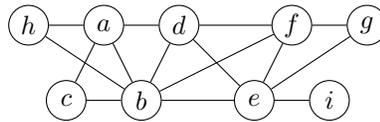


Figura 2.5: Grafo  $G_4$

Por exemplo, no grafo  $G_4$  da Figura 2.5,  $\{a, b\}$ ,  $\{b, d\}$  e  $\{e\}$  são os separadores minimais do grafo e o conjunto de separadores minimais de vértices é  $\mathbb{S} = \{\{a, b\}, \{b, d\}, \{e, f\}, \{e\}\}$ .

Os separadores minimais de vértices são úteis na caracterização de grafos cordais.

**Teorema 2.4.** *Um grafo  $G$  é cordal se e somente se todo separador minimal de vértices de  $G$  é uma clique.*

Uma *árvore de cliques*  $T$  de um grafo cordal  $G$  é uma árvore cujos vértices são as cliques maximais de  $G$  e, para quaisquer cliques maximais  $Q$  e  $Q'$ , cada clique no caminho entre  $Q$  e  $Q'$  em  $T$  contém  $Q \cap Q'$ .

A Figura 2.6 apresenta duas árvores de cliques de  $G_4$ .

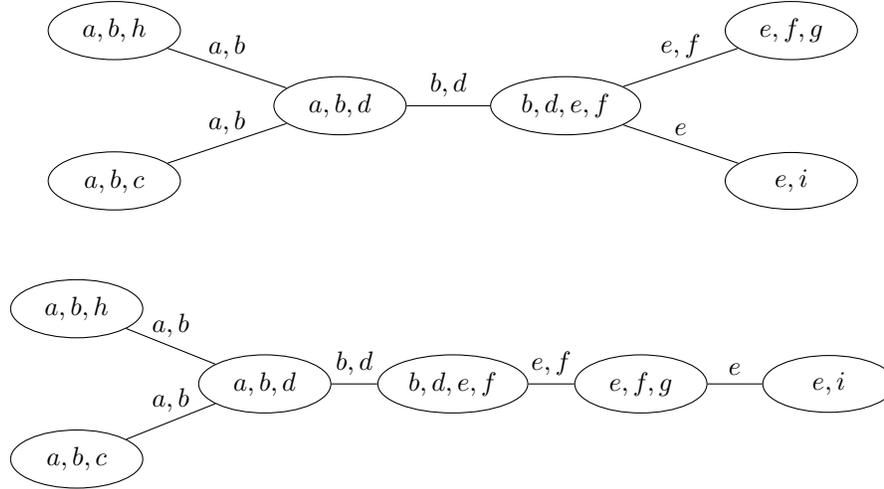


Figura 2.6: Árvores de cliques de  $G_4$

Em grafos cordais, existe uma ligação importante entre separadores minimais de vértices e árvores de cliques, como pode ser visto no próximo teorema.

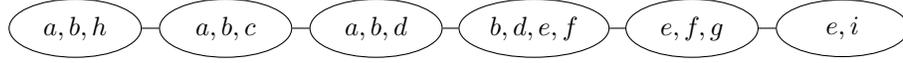
**Teorema 2.5.** *Seja  $T = (\mathbb{Q}, E_T)$  uma árvore de cliques de um grafo cordal  $G$ . Um conjunto  $S \subset V$  é um separador minimal de vértices de  $G$  se e somente se  $S = Q \cap Q'$  para alguma aresta  $\{Q, Q'\} \in E_T$ .*

Observe que um mesmo *smv* pode aparecer mais de uma vez na árvore de cliques. Denotamos por  $\mathbb{M}$  o multiconjunto de separadores minimais de vértices de  $G$  que correspondem a arestas da árvore de cliques. Devemos ressaltar que o multiconjunto  $\mathbb{M}$  é o mesmo para toda árvore de cliques de  $G$ . Chamamos de *multiplicidade* de um separador minimal de vértices, denotado por  $\mu(S)$ , o número de vezes que ele aparece em  $\mathbb{M}$ .

No grafo  $G_4$ , o conjunto  $\mathbb{M} = \{\{a, b\}, \{a, b\}, \{b, d\}, \{e, f\}, \{e\}\}$ . Seja o *smv*  $S = \{a, b\}$ ;  $\mu(S) = 2$ .

A *folhagem mínima* (*leafage*) de um grafo cordal  $G$ ,  $\ell(G)$ , é o menor número de folhas possíveis dentre todas as árvores de cliques de  $G$ . Na Figura 2.6 podemos observar que a primeira árvore de cliques do grafo  $G_4$  tem quatro folhas, e a segunda, três folhas. Entretanto, a folhagem mínima de  $G_4$  é 2; a Figura 2.7 mostra uma árvore de cliques de  $G_4$  com apenas duas folhas.

**Teorema 2.6.** *Uma clique maximal é uma folha em alguma árvore de cliques se e somente se ela é uma clique limítrofe.*

Figura 2.7: Folhagem mínima de  $G_4$ 

### 2.3.1 Construção da árvore de cliques e determinação de $\mathbb{S}$

O algoritmo para determinação das cliques maximais, Algoritmo 2.3, pode ser modificado para construir uma árvore de cliques e mesmo determinar o conjunto de separadores maximais de vértices. O Teorema 2.7 fornece o embasamento teórico, mostrando que a construção de uma árvore de cliques pode ser feita durante o percurso para determinação das cliques maximais. À medida que as cliques maximais são determinadas, a árvore de cliques vai sendo construída.

**Teorema 2.7.** *Seja  $G = (V, E)$  um grafo cordal e  $\sigma = \langle v_1, \dots, v_n \rangle$  um eep de  $G$ . Seja  $T_i$  uma árvore de cliques do subgrafo induzido  $G[\{v_i, \dots, v_n\}]$ . Se a clique  $S$  corresponde a uma aresta de  $T_i$  então ela corresponde também a uma aresta da árvore de cliques  $T$  de  $G$ .*

O algoritmo para construção da árvore de cliques é simples e será apresentado primeiro. Durante a execução do Algoritmo 2.3, apresentado na Seção 2.2, devemos, para cada vértice  $v_i$ , determinar se  $X_\sigma(v_i)$  é ou não uma clique maximal no grafo corrente. No novo algoritmo, faremos o mesmo. Se sim, a clique é simplesmente aumentada, como já ocorre no Algoritmo 2.3. Em caso contrário, quando uma nova clique é criada, a ligação na árvore de cliques deve ser estabelecida. Note que as modificações são mínimas. A complexidade de tempo do Algoritmo 2.4 é também  $O(n + m)$ .

---

#### Algoritmo 2.4. Determinar-árvore-de-cliques

---

**Entrada:** grafo cordal  $G = (V, E)$  e eep  $\sigma = \langle v_1, \dots, v_n \rangle$ ;  
**Saída:** árvore de cliques  $T = (V_T, E_T)$ ;  
**Início**  
 $q \leftarrow 1$ ;  
 $Q_1 \leftarrow \{v_n\}$ ;  $\eta(v_n) \leftarrow 1$ ;  
 $V_T \leftarrow \{Q_1\}$ ;  $E_T \leftarrow \emptyset$ ;  
**Para**  $i = n - 1, \dots, 1$  **faça**  
 $X_\sigma(v_i) \leftarrow \{u \in N(v_i) : \sigma^{-1}(u) > \sigma^{-1}(v)\}$ ;  
 $w \leftarrow z \in X_\sigma(v_i)$  tal que  $\sigma^{-1}(z)$  é mínimo;  
 $k \leftarrow \eta(w)$ ;  
**Se**  $|X_\sigma(v_i)| < |Q_k|$  **então**  
 $q \leftarrow q + 1$ ;  
 $Q_q \leftarrow \{v_i\} \cup X_\sigma(v_i)$ ;                   % nova clique maximal  $Q_q$   
 $V_T \leftarrow V_T \cup \{Q_q\}$ ;                   % incluir  $Q_q$  em  $T$   
 $E_T \leftarrow E_T \cup \{Q_q, Q_k\}$ ;  
 $\eta(v_i) \leftarrow q$ ;  
**senão**  
 $Q_k \leftarrow \{v_i\} \cup Q_k$ ;                   % clique  $Q_k$  é aumentada  
 $\eta(v_i) \leftarrow k$ ;  
**Fim.**

---

Vamos rever o acompanhamento do algoritmo para o grafo  $G_1$ . Somente as iterações  $i = 6, 5, 4, 2, 1$  são alteradas, com a adição de vértices e arestas aos conjuntos  $V_T$  e  $E_T$ .

$i$	$v_i$	$X_\sigma(v_i)$	$w$	$k$	$q$	$V_T$	$E_T$
					1	$V_T = \{Q_1\}$	$\emptyset$
6	$d$	$\{a, b\}$	$b$	1	2	$V_T \cup \{Q_2\}$	$E_T \cup \{Q_2, Q_1\}$
5	$h$	$\{a, b\}$	$b$	1	3	$V_T \cup \{Q_3\}$	$E_T \cup \{Q_3, Q_1\}$
4	$e$	$\{b, d\}$	$d$	2	4	$V_T \cup \{Q_4\}$	$E_T \cup \{Q_4, Q_2\}$
2	$i$	$\{e\}$	$e$	4	5	$V_T \cup \{Q_5\}$	$E_T \cup \{Q_5, Q_4\}$
1	$g$	$\{f\}$	$f$	4	6	$V_T \cup \{Q_6\}$	$E_T \cup \{Q_6, Q_4\}$

Vamos agora tratar da determinação do conjunto de separadores minimais de vértices de um grafo  $G$ . Já sabemos que o conjunto  $\mathbb{M}$  de separadores minimais de vértices é um multiconjunto. A remoção das repetições é usualmente executada separadamente, se constituindo num segundo passo de muitos algoritmos conhecidos. Para evitar esse processamento, vamos lançar mão de propriedades do percurso em largura lexicográfica e estabelecer durante o percurso a multiplicidade de cada  $smv$ .

Não é difícil ver que, quando da execução do Algoritmo 2.1, como os rótulos dos vértices precisam estar ordenados, de alguma forma os conjuntos de adjacência monótona  $X_\sigma$  também o estão. O próximo teorema mostra exatamente como.

**Teorema 2.8.** *Seja  $G = (V, E)$  um grafo cordal e  $\sigma = \langle v_1, \dots, v_n \rangle$  um  $eep$  de  $G$  obtido por um percurso em largura lexicográfica. Seja  $\mathbb{L}_j = \langle label(v_i) : |label(v_i)| = j, i = 1, \dots, n \rangle$  a sequência de rótulos de comprimento  $j$  fornecidos pelo percurso em largura lexicográfica. Então  $\mathbb{L}_j$  está em ordem lexicográfica.*

Demonstração:

Quando um vértice  $v$  é escolhido durante a execução de um percurso em largura lexicográfica,  $label(v)$  é, nesse momento, lexicograficamente maior ou igual aos rótulos dos vértices remanescentes. Assuma que  $label(v) \in \mathbb{L}_j$  e existe um vértice  $w$  a ser visitado após  $v$  no percurso tal que  $label(v) = label(w)$  and  $w \in N(v)$ . O rótulo  $label(w)$  será modificado e será maior que  $label(v)$ . Entretanto a cardinalidade de  $label(w)$  também será maior do que a cardinalidade de  $label(v)$  e a ordenação de  $\mathbb{L}_j$  não é modificada. ■

O Teorema 2.8 nos mostra que rótulos iguais aparecem juntos em  $\mathbb{L}_j$ . Logo, separadores minimais de vértices também.

O Algoritmo 2.5 é uma extensão do Algoritmo 2.3, para determinação das cliques maximais (utilizamos agora um  $eep$  produzido por um percurso em largura lexicográfica). Novamente, durante a execução do algoritmo, devemos, para cada vértice  $v_i$ , determinar se  $X_\sigma(v_i)$  é ou não uma clique maximal no grafo corrente. Se sim, a clique é simplesmente aumentada, como já ocorre no Algoritmo 2.3. Se uma nova clique é criada, temos um novo  $smv$ , que é então comparado com o anterior de igual cardinalidade, incrementando, se necessário, a multiplicidade do separador.

**Algoritmo 2.5.** *Determinar-separadores-minimais-vértices*


---

**Entrada:** grafo cordal  $G = (V, E)$  e eep  $\sigma$  fornecida por lex;  
**Saída:**  $\mathbb{S}$  e multiplicidade dos separadores minimais de vértices;  
**Início**  
 $q \leftarrow 1; j \leftarrow 0;$   
 $Q_1 \leftarrow \{v_n\}; \eta(v_n) \leftarrow 1;$   
**Para**  $i = 1, \dots, n$  **faça**  $last[i] \leftarrow 0;$   
 $\mathbb{S} \leftarrow \emptyset; S_0 \leftarrow \emptyset;$   
**Para**  $i \leftarrow n - 1, \dots, 1$  **faça**  
 $X_\sigma(v_i) \leftarrow \{u \in N(v_i) : \sigma^{-1}(u) > \sigma^{-1}(v_i)\};$   
 $w \leftarrow z \in X_\sigma(v_i)$  tal que  $\sigma^{-1}(z)$  é mínimo;  
 $k \leftarrow \eta(w);$   
**Se**  $|X_\sigma(v_i)| < |Q_k|$  **então**  
 $q \leftarrow q + 1;$   
 $Q_q \leftarrow \{v_i\} \cup X_\sigma(v_i);$  % nova clique maximal  $Q_q$   
 $\eta(v_i) \leftarrow q;$   
 $tam \leftarrow |X_\sigma(v_i)|; c \leftarrow last[tam];$   
**Se**  $X_\sigma(v_i) \neq S_c$  **então**  
 $j \leftarrow j + 1; S_j \leftarrow X_\sigma(v_i);$  % novo  $smv$   
 $last[tam] \leftarrow j; \mu(j) \leftarrow 1;$   
 $\mathbb{S} \leftarrow \mathbb{S} \cup S_j;$   
**senão**  
 $\mu(c) \leftarrow \mu(c) + 1;$  % incrementar multiplicidade  
**senão**  
 $Q_k \leftarrow \{v_i\} \cup Q_k;$  % clique  $Q_k$  é aumentada  
 $\eta(v_i) \leftarrow k;$   
**Fim.**

---

### 2.3.2 Propriedades de árvores de cliques

Podemos enunciar de maneira mais teórica algumas propriedades de árvores de cliques de grafos cordais que já foram intuitivamente apresentadas até agora.

Considere o grafo  $G = (V, E)$ , seu conjunto de cliques maximais  $\mathbb{Q}$  e o conjunto  $\mathcal{T}$  de todas as árvores cujo conjunto de vértices é  $\mathbb{Q}$ .

**Propriedade 2.1.** *Propriedade da interseção de cliques (PIC)*

Para quaisquer duas cliques maximais distintas  $Q$  e  $Q'$ , o conjunto  $Q \cap Q'$  está contido em todas as cliques no caminho entre  $Q$  e  $Q'$  em  $T \in \mathcal{T}$ .

Denotamos por  $\mathcal{T}_{PIC}$  o subconjunto de  $\mathcal{T}$  das árvores que satisfazem a PIC.

**Teorema 2.9.** *Um grafo conexo  $G$  é cordal se e somente se existe uma árvore  $T \in \mathcal{T}_{PIC}$ .*

**Propriedade 2.2.** *Propriedade da subárvore induzida (PSI)*

Para todo vértice  $v \in V$ ,  $\mathbb{Q}(v)$ , o subconjunto de  $\mathbb{Q}$  de todas as cliques maximais que contêm o vértice  $v \in V$ , induz uma subárvore em  $T \in \mathcal{T}$ .

Denotamos por  $\mathcal{T}_{PSI}$  o subconjunto de  $\mathcal{T}$  das árvores que satisfazem a PSI.

**Teorema 2.10.** *Um grafo conexo  $G$  é cordal se e somente se existe uma árvore  $T \in \mathcal{T}_{PSI}$ .*

**Propriedade 2.3.** *Propriedade da sequência de cliques (running intersection property) (RIP)*

Considere a sequência  $\langle Q_1, \dots, Q_q \rangle$ ,  $q \geq 1$ , de cliques maximais. Para toda clique  $Q_j$ ,  $j = 2, \dots, q$ , existe uma clique  $Q_i$ ,  $1 \leq i \leq j - 1$ , com

$$I = Q_j \cap (Q_1 \cup \dots \cup Q_{j-1}) \quad (2.3.1)$$

tal que  $I \neq \emptyset$  é uma clique e  $I \subset Q_i$ .

Observe que, por construção, a sequência  $\langle Q_1, \dots, Q_q \rangle$ , obtida pela Algoritmo 2.3, satisfaz a *RIP*. Seja uma destas sequências. É possível construir uma árvore  $T \in \mathcal{T}$  tal que cada clique maximal  $Q_j$  é adjacente à clique  $Q_i$  (não necessariamente única) que satisfaz equação 2.3.1.

Denotamos por  $\mathcal{T}_{RIP}$  o subconjunto de  $\mathcal{T}$  das árvores obtidas desta forma.

**Teorema 2.11.** *Um grafo conexo  $G$  é cordal se e somente se existe uma árvore  $T \in \mathcal{T}_{RIP}$ .*

A *árvore de cliques valorada* de  $G$ ,  $C_\nu(G) = (\mathbb{Q}, E')$ , é uma árvore de cliques de  $G$  com a atribuição de valor as arestas  $\nu : E \rightarrow \mathbb{N}$  tal que  $\nu(\{Q, Q'\}) = |Q \cap Q'|$ .

Denotamos por  $\mathcal{T}_{PAM}$  o subconjunto de  $\mathcal{T}$  de todas as árvores geradoras de peso máximo de  $C_\nu(G)$ .

**Teorema 2.12.** *Um grafo conexo  $G$  é cordal se e somente se existe uma árvore  $T \in \mathcal{T}_{PAM}$ .*

A equivalência de todas as propriedades pode ser considerada, evidenciando o fato que o conjunto de árvores de cliques é único.

**Teorema 2.13.**  $\mathcal{T}_{PIC} = \mathcal{T}_{PSI} = \mathcal{T}_{RIP} = \mathcal{T}_{PAM}$ .

## 2.4 Grafo de cliques reduzido

Seja  $G$  um grafo cordal. O *grafo de cliques reduzido*,  $C_r(G)$ , é uma estrutura que deriva do grafo de cliques, e, de alguma forma, exhibe de maneira mais clara suas propriedades. É utilizado na caracterização de classes e resolução de problemas.

Duas cliques maximais  $Q$  e  $Q'$  de  $G$  formam um *par de separação* se  $Q \cap Q'$  é não vazio e todo caminho em  $G$  de um vértice de  $Q \setminus Q'$  para um vértice de  $Q' \setminus Q$  contém um vértice de  $Q \cap Q'$ . A possibilidade de bem caracterizar essas cliques permite relacioná-las com separadores minimais de vértices.

**Teorema 2.14.** *Um conjunto  $S$  é um separador minimal de vértices de um grafo cordal  $G$  se e somente se existem cliques maximais  $Q$  e  $Q'$  de  $G$  que formam um par de separação tal que  $S = Q \cap Q'$ .*

Agora podemos definir a estrutura. O *grafo de cliques reduzido*  $C_r(G)$  de  $G$  é o grafo cujos vértices são as cliques maximais de  $G$  e cujas arestas  $\{Q, Q'\}$  ligam cliques  $Q$  e  $Q'$  que formam pares de separação.

O grafo  $C_r(G)$  tem uma propriedade efetiva em relação às árvores de cliques.

**Teorema 2.15.** *Seja  $G$  um grafo cordal conexo. O grafo de cliques reduzido  $C_r(G)$  é a união de todas as árvores de cliques de  $G$ .*

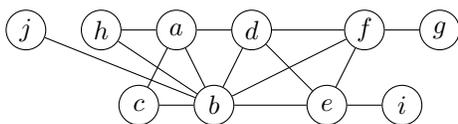


Figura 2.8: Grafo  $G_5$

Observe o grafo  $G_5$  da Figura 2.8. A Figura 2.9 mostra o grafo de cliques e o grafo de cliques reduzido de  $G_5$ . Observe que, apesar do conjunto  $\{b\}$  ser um *smv*, nem todas as arestas correspondentes a esse conjunto aparecem em  $C_r(G)$ : enquanto as cliques maximais  $\{b, j\}$  e  $\{a, b, h\}$  são um par de separação, as cliques maximais  $\{b, d, e, f\}$  e  $\{a, b, h\}$  não o são. Isso é fácil de compreender observando que existe um caminho entre o vértice  $a$  e o vértice  $e$  que não passa por  $b$ .

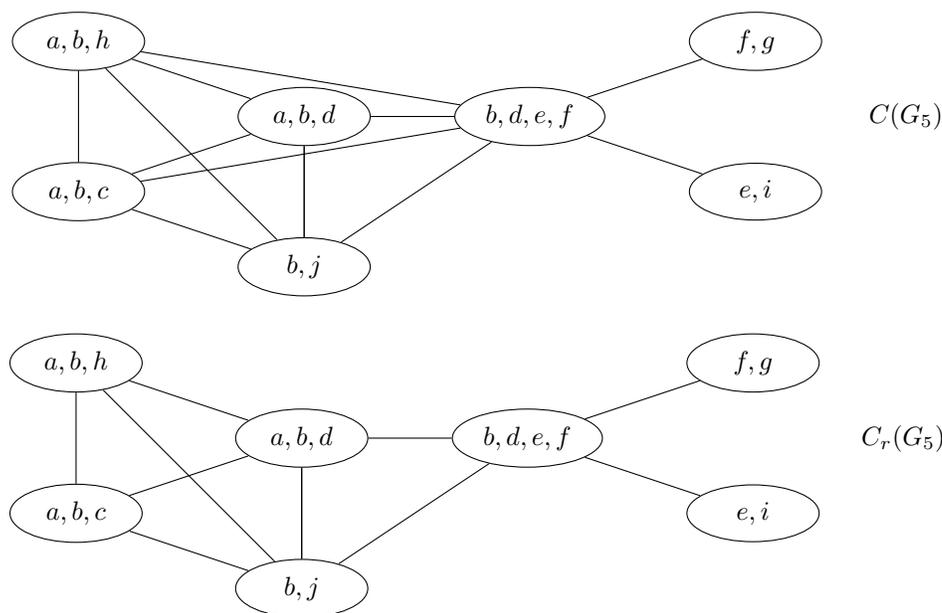


Figura 2.9:  $C(G_5)$  e  $C_r(G_5)$

O algoritmo para determinar o grafo cliques reduzido tem a mesma complexidade de tempo da determinação do grafo de cliques,  $O(n^3)$ . O raciocínio é semelhante. Precisamos estudar todos os pares de cliques maximais, tendo ainda que verificar se cada par forma um par de separação. Para isso, devemos primeiramente estabelecer, em  $C_r(G)$ ,  $q$  componentes conexas, contendo, cada uma, uma clique maximal. Em seguida, ordenar os separadores minimais de vértices por tamanho; utilizando uma ordenação por distribuição (*bucket sort*) a complexidade não se altera. Finalmente, para cada *smv*, verificar se ambas as cliques de cada par o contém e, caso estejam em componentes conexas distintas, incluir a aresta em  $C_r(G)$ .

Da mesma forma que podemos encontrar grafos de clique não cordais para grafos cordais, o mesmo pode ser agora observado. A Figura 2.10 mostra novamente o grafo  $G_3$ , já visto na Figura 2.4, e  $C_r(G_3)$ . Também nesse caso,  $C_r(G_3)$  não é cordal. As

arestas com traçado mais forte indicam arestas que pertencem a todas as árvores de cliques.

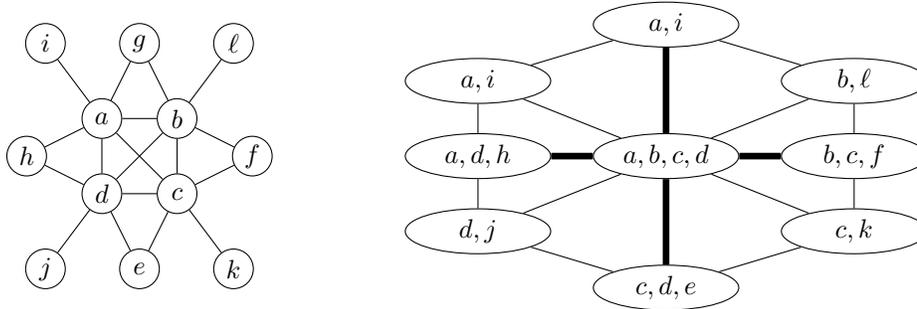


Figura 2.10: Grafo  $G_3$  e  $C_7(G_3)$

## 2.5 Grafo de cliques críticas

A estrutura apresentada nessa seção é particularmente interessante para grafos de grandes dimensões que apresentem vizinhanças idênticas, uma vez que a dimensão do grafo pode ser significativamente reduzida. Em grafos cordais, auxiliam na caracterização de algumas subclasses.

Dizemos que uma clique  $Q$  de  $G$  é uma *clique crítica* quando todos os seus vértices são gêmeos verdadeiros e é maximal no sentido de que não existe nenhum vértice  $v \in V$  de tal que  $Q \cup \{v\}$  é uma clique crítica. O *grafo de cliques críticas* de  $G$ , denotado por  $C_c(G)$ , é o grafo cujo conjunto de vértices são as cliques críticas de  $G$  e tal que duas cliques críticas  $Q$  e  $Q'$  são adjacentes sempre que algum vértice de  $Q$  for adjacente em  $G$  a vértices em  $Q'$ .

A Figura 2.11, mostra o grafo de cliques críticas do grafo  $G_2$ , da Figura 2.2. A redução da dimensão é sensível.

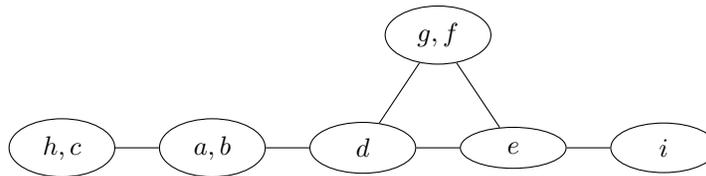


Figura 2.11: Grafo  $C_c(G_2)$

Sendo  $G$  um grafo cordal, algumas propriedades do grafo de cliques críticas relacionam seus vértices e as cliques maximais:

**Propriedade 2.4.** *Seja  $G$  um grafo cordal. Todo vértice de  $G$  pertence a exatamente uma clique crítica.*

**Propriedade 2.5.** *Seja  $G$  um grafo cordal. Então toda clique maximal é a união de cliques críticas.*

**Propriedade 2.6.** *Seja  $G$  um grafo cordal. Então qualquer clique maximal de  $C_c(G)$  tem no máximo um vértice simplicial.*

**Propriedade 2.7.** *Seja  $G$  um grafo cordal. Então  $C_c(G)$  é um grafo cordal.*

A respeito dessa última propriedade, caso  $C_c(G)$  tivesse um ciclo,  $G$  também teria, tomando, simplesmente, um vértice de cada clique crítica do ciclo em  $C_c(G)$ .

O Teorema 2.16 diz respeito ao isomorfismo de  $C(G)$  e  $C(C_c(G))$ . Observe que, apesar das arestas serem valoradas pela cardinalidade da interseção das cliques maximais, ambos os grafos podem ser considerados sem essa informação, levando em conta apenas a adjacência entre os vértices que compõem a clique.

**Teorema 2.16.** *Seja  $G = (V, E)$  um grafo cordal. Então os grafos  $C(G)$  e  $C(C_c(G))$  são isomorfos.*

Demonstração:

Uma clique maximal  $Q$  de  $G$  é composta por vértices simpliciais e vértices pertencentes a separadores minimais de vértices. Os vértices simpliciais de qualquer clique, se existirem, formam uma clique crítica, uma vez que são gêmeos verdadeiros. Os vértices restantes pertencem a:

1. um único separador minimal de vértices:

O  $smv$  forma uma clique crítica.

2. ao menos dois separadores minimais de vértices:

Considere  $S$  e  $S'$  contidos em  $Q$ . Se  $S \cap S' = \emptyset$  então cada um forma uma clique crítica. Caso contrário, eles formam ao menos duas cliques críticas, dependendo de outras interseções. No pior caso, cada vértice poderá determinar uma clique crítica.

Cada clique maximal  $Q$  de  $G$  corresponde a exatamente uma clique maximal  $Q_c$  de  $C_c(G)$  com a mesma ou menor cardinalidade. Pela Proposição 2.5, a clique  $Q$ , de vértices de  $G$ , corresponde à união de cliques críticas  $C_1, \dots, C_k$ ,  $1 \leq k \leq |Q|$ , que compõem  $Q_c$  em  $C_c(G)$ .

Considerando a relação de adjacência, as cliques críticas  $C_1, \dots, C_k$ , formadas por vértices adjacentes em  $G$ , são adjacentes em  $C_c(G)$ . Se os vértices de  $C_i$  pertencem a um  $smv$  de  $G$ ,  $C_i$  pertence a pelo menos duas cliques maximais em  $C(G)$  e a mesma adjacência ocorrerá em  $C(C_c(G))$ .

Então, existe um isomorfismo de vértices e arestas de  $C(G)$  e  $C(C_c(G))$ . ■

O mesmo raciocínio pode ser estendido a árvores de cliques.

**Teorema 2.17.**  *$T$  é uma árvore de cliques de um grafo cordal  $G$  se e somente se  $T$  é isomorfa a uma árvore de cliques  $T_c$  de  $C_c(G)$ .*

Demonstração:

Sejam  $Q$  e  $Q'$  dois vértices (cliques maximais de  $G$ ) de uma árvore de cliques  $T$  de  $G$ . Por definição de árvore de cliques, cada vértice no caminho entre  $Q$  e  $Q'$  em  $T$  contém  $Q \cap Q'$ . As cliques maximais  $Q$  e  $Q'$  têm cliques maximais correspondentes  $Q_c$  e  $Q'_c$ . Observando que  $Q_c$  é uma partição de  $Q$ , bem como  $Q'_c$  é uma partição de  $Q'$ , temos que  $Q \cap Q'$  é uma clique crítica composta por vértices oriundos de separadores minimais de vértices de  $G$ . Assim,  $Q \cap Q' = \bigcup_{X \in Q_c \cap Q'_c} X$ . Isso significa que existe um caminho correspondente entre  $Q_c$  e  $Q'_c$  e uma árvore de cliques  $T_c$

pode ser construída a partir de  $C(C_c(G))$ , isomorfa à árvore  $T$ . ■

Dado um grafo  $G = (V, E)$ , podemos esboçar um algoritmo para a obtenção de  $C_c(G)$ . Uma vez que cada vértice de  $V$  pertence a exatamente uma clique crítica, o algoritmo para obtenção do grafo de cliques críticos consiste em ordenar lexicograficamente as vizinhanças fechadas dos vértices e agrupar em cliques os vértices com mesma vizinhança. A complexidade de tempo é  $O(n^2)$ . Nesse caso, o algoritmo é o mesmo para grafos cordais ou não cordais. Entretanto, veremos no Capítulo 5 como ele pode ser simplificado para algumas subclasses.

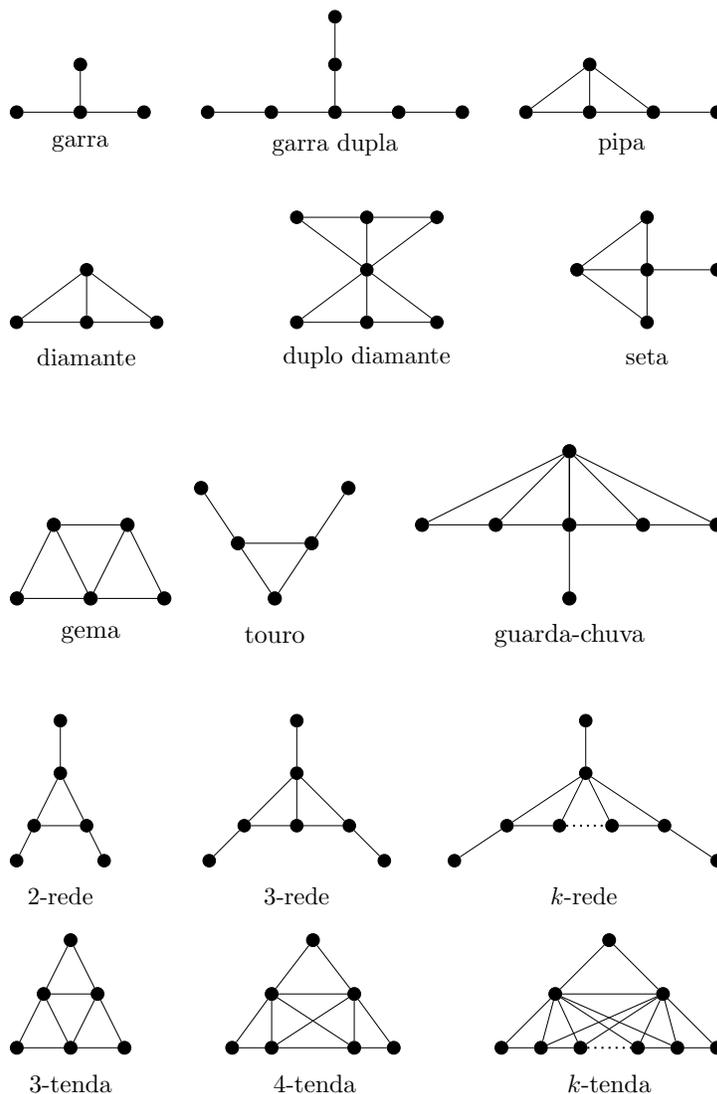


Figura 2.12: Subgrafos proibidos clássicos

## 2.6 Subgrafos proibidos

Já vimos no Capítulo 1, Seção 1.3, que, na caracterização de classes de grafos, usualmente são utilizados *subgrafos proibidos*. Reunimos na Figura 2.12 uma coleção de grafos que aparecem em caracterizações de subclasses de grafos cordais. Por serem de natureza mais teórica, tais caracterizações não serão, em geral, demonstradas nesse texto.

## 2.7 Exercícios

1. A sequência  $\sigma = \langle h, c, b, a, e, d, g, f, i \rangle$  é um esquema de eliminação perfeita para o grafo  $G_2$  da Figura 2.2? Sim ou não, explique porquê.
2. Encontre um *eep* produzido pelo percurso em largura lexicográfica para o grafo  $G_2$  da Figura 2.2.
3. Utilizando o percurso em largura lexicográfica, encontre um esquema de eliminação perfeita para o grafo  $G_4$  da Figura 2.4. Determine também sua sequência de cliques maximais.
4. Construa duas árvores de cliques para o grafo do item anterior. Qual a folhagem mínima desse grafo?
5. Construa todas as árvores de cliques distintas do grafo  $G_2$  da Figura 2.2. Quantas são?
6. Elabore um algoritmo que gera os conjuntos de adjacência de um grafo  $G$ , sendo conhecidos seu conjunto de cliques maximais  $\mathbb{Q}$  e o número de vértices  $n$  de  $G$ .
7. Apresente um grafo  $G$  tal que  $C_r(G)$  seja igual a  $C(G)$ .
8. Apresente um grafo  $G$  que seja igual a  $C_c(G)$ .  $G$  deve ter, no mínimo, 10 vértices.
9. Construa o grafo de cliques críticas do grafo  $G_2$  da Figura 2.2.
10. Justifique a propriedade:  
*Seja  $G$  um grafo cordal. Então qualquer clique maximal de  $C_c(G)$  tem no máximo um vértice simplicial.*
11. Apresente um algoritmo linear para determinar os vértices simpliciais de um grafo cordal.
12. Grafos bloco são grafos cordais livres de diamante. Mostre que o grafo  $G_2$  não é um grafo bloco, assinalando um subgrafo induzido que seja um diamante.

## 2.8 Notas bibliográficas

As ferramentas estruturais foram definidas em ocasiões distintas para aplicações relacionadas com as mais diversas classes de grafos. O percurso em largura lexicográfica, utilizado no reconhecimento de grafos cordais, foi desenvolvido por Rose *et al.* [66]. A representação por conjuntos de adjacência monótona foi apresentada em Markenzon e Vernet [53]. A geração de todos os esquemas de eliminação perfeita de um grafo cordal pode ser encontrada em Chandran *et al.* [15]. O esquema de eliminação bicompatível é mencionado no trabalho de Panda e Das [63] e utilizado na determinação do ciclo hamiltoniano de um grafo de intervalo próprio. O conceito de vértice simples, o esquema de eliminação simples e o esquema de eliminação forte foram apresentados por Farber [21, 22], onde é provada a caracterização de grafos fortemente cordais. Sawada e Spinrad [68] apresentam um algoritmo linear para a conversão de um esquema de eliminação simples em forte.

O trabalho de Blair e Peyton [3] é fundamental para o estudo de separadores minimais de vértices e árvores de cliques. As árvores de cliques foram introduzidas independentemente, em 1974, por Gavril [27] e Buneman [13]. O algoritmo básico para sua determinação é encontrado em Spinrad [70]. O algoritmo de determinação dos separadores minimais de vértices e suas multiplicidades é encontrado em Markenzon e Pereira [51]. Outros textos úteis relacionados a cliques maximais, grafo de cliques e árvores de cliques são os de Shibata [69] e Hare e Takemura [34].

O grafo de cliques reduzido foi apresentado em Galinier *et al.* [25] e retomado por Habib e Stacho [33]. O conceito de cliques críticas foi introduzido por Lin *et al.* [47] e utilizado na definição de grafo de cliques críticas por Kennedy [40] e Kennedy *et al.* [41]. Sua principal aplicação é a caracterização de algumas subclasses de grafos cordais, como, por exemplo, os grafos estritamente cordais [11] e os grafos potência de 3-folha [8], que serão vistos no Capítulo 5.

Um levantamento exaustivo de subgrafos proibidos e as classes relacionadas aos mesmos pode ser encontrado em Brandstädt *et al.* [9].

## Capítulo 3

# Subclasses de Grafos Cordais

Nesse capítulo vamos rever três ramos bem conhecidos na hierarquia de grafos cordais: os grafos de intervalo, os grafos unicamente representáveis e os grafos ptolemaicos. Esses três ramos são as raízes das diversas subclasses tratadas nos próximos capítulos. Para cada um deles, vamos apresentar a definição da classe e as caracterizações mais conhecidas. Procuraremos acrescentar também algumas caracterizações que possam ser melhor utilizadas sob o ponto de vista algorítmico, isto é, que nos forneçam, na medida do possível, subsídios para a manipulação eficiente dessas classes. Para tal, a utilização das ferramentas apresentadas no Capítulo 2 é imprescindível.

### 3.1 Grafos de intervalo

Vimos, no Capítulo 1, que grafos cordais são os grafos de interseção de subárvores de uma árvore. O conceito de grafo de interseção pode ser aplicado a uma bem conhecida subclasse de cordais: os grafos de intervalo. Um grafo  $G = (V, E)$  é dito *grafo de intervalo* se ele é o grafo de interseção de uma família finita de intervalos  $I$  em uma reta real.

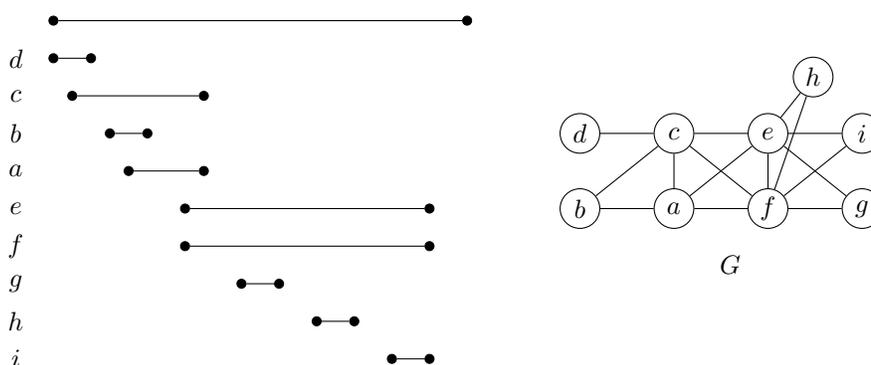


Figura 3.1: Intervalos e o grafo de interseção

A Figura 3.1 apresenta uma família de intervalos e seu grafo de interseção. Existem muitas caracterizações para essa classe, as primeiras surgindo em 1962. O

conceito que se segue é fundamental para a caracterização apresentada no Teorema 3.1.

Um *asteroide tripla* (AT) num grafo  $G$  é uma tripla de vértices tal que cada par de vértices é ligado por um caminho que evita a vizinhança do terceiro vértice. Um conjunto  $A \subseteq V$  é um *conjunto asteroideal* se cada tripla de vértices em  $A$  é um asteroide tripla. O tamanho máximo de um conjunto asteroideal é denotado  $a(G)$ .

**Teorema 3.1.**  $G$  é um grafo de intervalo se e somente se  $G$  é livre de AT, isto é,  $G$  não contém um asteroide tripla.

No grafo da Figura 2.1, apresentado no Capítulo 2, os vértices  $c$ ,  $g$  e  $i$  formam um asteroide tripla. Uma propriedade importante dos grafos de intervalo é exibida na próxima caracterização.

**Teorema 3.2.**  $G$  é um grafo de intervalo se e somente se  $G$  tem uma árvore de cliques que é um caminho.

Um resultado imediato do teorema apresenta uma caracterização por folhagem mínima (Capítulo 2, Seção 2.3) para a classe.

**Corolário 3.1.** Seja  $G$  um grafo cordal não completo.  $G$  é um grafo de intervalo se e somente se  $\ell(G) = 2$ .

Se a árvore de cliques de um grafo cordal  $G$  é um caminho, chamamos essa árvore um *caminho de cliques*. Na Figura 3.2, são apresentadas duas árvores de cliques para o grafo  $G$  da Figura 3.1. Note que somente a segunda é um caminho de cliques. Um grafo de intervalo pode ter diversas árvores de cliques.

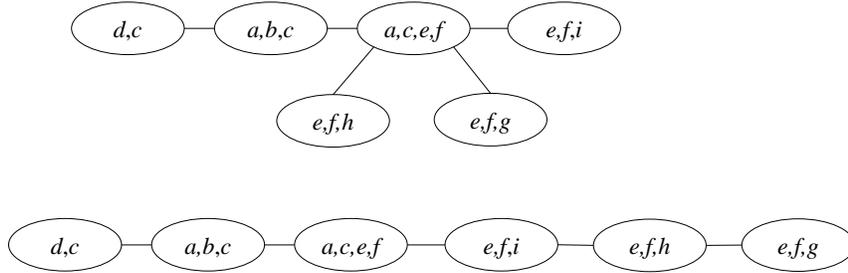


Figura 3.2: Árvores de cliques do grafo  $G$

No Teorema 3.3, grafos de intervalo são caracterizados em função de seus subgrafos proibidos. Os mesmos podem ser encontrados no Capítulo 2, Seção 2.6.

**Teorema 3.3.**  $G$  é um grafo de intervalo se e somente se  $G$  não contém uma garra dupla, um guarda-chuva, uma  $k$ -rede,  $k \geq 2$ , uma  $k$ -tenda,  $k \geq 3$  ou um  $C_k$ ,  $k \geq 4$ , como subgrafos induzidos.

Vamos relembrar a seguir uma propriedade dos vértices de um grafo de intervalo, que nos será bastante útil no Capítulo 4. Para tal, uma nova sequência precisa ser definida.

Considere uma sequência de vértices  $\delta = \langle v_1, \dots, v_n \rangle$  de  $V$ . Esta sequência é denominada *gulosa* quando para quaisquer vértices  $v_i, v_j, v_k, i < j < k$ , tem-se

$$\text{se } \{v_i, v_k\} \in E \text{ então } \{v_i, v_j\} \in E. \quad (3.1.1)$$

**Teorema 3.4.** *Um grafo  $G = (V, E)$  é um grafo de intervalo se e somente se existe uma sequência gulosa de  $V$ .*

Para o grafo  $G$  da Figura 3.1, uma sequência gulosa dos vértices é  $\delta = \langle d, c, b, a, e, f, i, h, g \rangle$ .

Um algoritmo para determinar uma sequência gulosa utiliza o caminho de cliques  $\langle Q_1, \dots, Q_q \rangle$  do grafo de intervalo. A cada iteração, os vértices de uma clique maximal são analisados. Os vértices que são encontrados pela primeira vez no percurso são armazenados na lista  $bucket[i], 1 \leq i \leq q$ . Para cada vértice, o rótulo  $H(v)$  indica a última clique da sequência à qual o vértice pertence. Note que, dessa forma, para cada vértice é conhecido o seu percurso no caminho de cliques. Ao fim do processo, as listas armazenadas em  $bucket$  são concatenadas.

**Algoritmo 3.1.** *Determinar-Sequência-Gulosa*

---

**Entrada:** grafo de intervalo  $G = (V, E)$  e caminho de cliques  $\langle Q_1, \dots, Q_q \rangle$ ;  
**Saída:** sequência gulosa  $\delta$ ;  
**Início**  
 $\delta \leftarrow \langle \rangle$ ;  
**Para**  $v \in V$  **faça**  $H(v) \leftarrow 0$ ;  
**Para**  $i = 1, \dots, q$  **faça**  
    **Para todo**  $v \in Q_i$  **faça**           % percorrer clique  $Q_i$   
        **Se**  $H(v) = 0$  **então**  
            inclua  $v$  no  $bucket[i]$ ;  
             $H(v) \leftarrow i$ ;  
            mover  $v$  para o final da lista  $bucket[j], 1 \leq j \leq i$ , à qual pertence;  
**Para**  $i = 1, \dots, q$  **faça**  
     $\delta \leftarrow \delta || bucket[i]$ ;           % concatenar as listas  
**Fim.**

---

Vamos acompanhar o Algoritmo 3.1 no grafo  $G$ , da Figura 3.1. Consideramos o caminho de cliques  $\langle \{c, d\}, \{a, b, c\}, \{a, c, e, f\}, \{e, f, i\}, \{e, f, h\}, \{e, f, g\} \rangle$ . Após a segunda iteração temos a seguinte situação:

$H$	a	b	c	d	e	f	g	h	i
	2	2	2	1	0	0	0	0	0

$i$	$Q_i$	$bucket$
1	$\{c, d\}$	$bucket[1] = c, d$
2	$\{a, b, c\}$	$bucket[1] = d, c$ $bucket[2] = a, b$

Os vértices  $c$  e  $d$  pertencem à clique  $Q_1$ : são incluídos em  $bucket[1]$ ;  $H(c)$  e  $H(d)$  recebem 1. Note que  $bucket[1]$  não sofrerá mais inclusões, mas a ordem dos vértices na lista poderá ser alterada, o que acontece na segunda iteração. Ao se considerar a clique  $Q_2 = \{a, b, c\}$ , os vértices incluídos em  $bucket[2]$  são  $a$  e  $b$  ( $c$  já

foi considerado na iteração anterior);  $H(c)$  também é alterado, passando a ter valor 2. Além disso, a ordem dos vértices na lista  $bucket[1]$  é modificada, porque o vértice  $c$  é transferido para o fim da lista. Ao fim da 6ª iteração temos:

		a		b		c		d		e		f		g		h		i
$H$		3		2		3		1		6		6		6		5		4

$i$		$Q_i$		$bucket$
1		$\{c, d\}$		$bucket[1] = c, d$
2		$\{a, b, c\}$		$bucket[1] = d, c$ $bucket[2] = a, b$
...		...		...
6		$\{e, f, g\}$		$bucket[1] = d, c$ $bucket[2] = b, a$ $bucket[3] = e, f$ $bucket[4] = i$ $bucket[5] = h$ $bucket[6] = g$

Considere um grafo de intervalo  $G = (V, E)$ , um caminho de cliques  $\langle Q_1, \dots, Q_q \rangle$  e  $\delta = \langle v_1, \dots, v_n \rangle$  uma seqüência gulosa de  $G$  computada pelo Algoritmo 3.1. Tem-se então:

**Propriedade 3.1.** *A lista  $bucket[i]$ ,  $1 \leq i \leq q$ , tem pelo menos um elemento.*

O Algoritmo 3.1 tem complexidade de tempo  $O(n+m)$ . O algoritmo inicializa  $H$  para os vértices; em seguida, percorre as cliques maximais do grafo, o que sabemos ser  $O(n+m)$  (Teorema 2.3). Se  $bucket[i]$ ,  $1 \leq i \leq q$ , for implementado por uma lista duplamente encadeada, e cada vértice apontar para a sua posição nessa lista, a transferência do vértice para o fim da lista leva tempo constante. Assim, temos a complexidade linear do algoritmo.

Observe que a seqüência gulosa  $\langle d, c, b, a, e, f, i, h, g \rangle$  não é um *eep* de  $G$ , pois  $c$  não é simplicial no subgrafo  $G[\{c, b, a, e, f, i, h, g\}]$ . Já a seqüência na ordem inversa  $\langle g, h, i, e, a, b, c, d \rangle$  é um *eep*. Na verdade, esta é uma propriedade de qualquer ordenação gulosa para os grafos de intervalo.

**Teorema 3.5.** *Considere uma seqüência gulosa  $\delta = \langle v_1, \dots, v_n \rangle$  de um grafo de intervalo  $G = (V, E)$ . A seqüência  $\delta^{-1} = \langle v_n, \dots, v_1 \rangle$  é um esquema de eliminação perfeita de  $G$ .*

Demonstração:

Considere a seqüência  $\langle v_n, \dots, v_1 \rangle$ . O vértice  $v_n$  da seqüência gulosa é sempre um vértice simplicial do grafo  $G$  por pertencer a uma única clique maximal. Se  $v_i \in V$ ,  $1 \leq i \leq n-1$ , não é simplicial em  $G[\{v_i, \dots, v_1\}]$ ,  $N(v_i)$  não é uma clique, isto é, existem vértices  $v_j, v_k \in N(v_i)$ ,  $1 \leq j < k \leq i-1$ , tais que  $\{v_j, v_k\} \notin E$ . Por definição, Propriedade (3.1.1), se  $\{v_j, v_i\} \in E$  então  $\{v_j, v_k\} \in E$ . Contradição. ■

### 3.1.1 Subclasses de grafos de intervalo

Uma importante subclasse de grafos de intervalo surge naturalmente: os *grafos de intervalo próprio*, classe que, por si só, tem sido bastante estudada. Um *grafo de*

*intervalo próprio* é o grafo de interseção de uma família de intervalos em uma reta real na qual nenhum intervalo está propriamente contido em outro. Essa classe será estudada com detalhes no Capítulo 4.

Outra subclasse de grafos de intervalo são os grafos caterpillar, já mencionados no Capítulo 1, Seção 1.3. Um *caterpillar*  $G = (V, E)$ ,  $|V| \geq 3$ , é o grafo em que a remoção de todas as folhas resulta em um caminho. Uma caracterização da classe decorre do fato desses grafos serem exatamente a interseção entre a classe dos grafos de intervalo e a das árvores. Os grafos caterpillar aparecem em diversos ramos da hierarquia que estudamos em nosso texto. Ainda nesse capítulo, eles aparecem como subclasse dos grafos ptolemaicos.

### 3.2 Grafos cordais unicamente representáveis

Uma abordagem interessante para a definição de subclasses é estabelecer restrições nas próprias ferramentas estruturais quando aplicadas ao grafo. A primeira delas é a árvore de cliques.

Um grafo cordal é um *grafo cordal unicamente representável* (resumidamente *grafo ur-cordal*) se ele tem exatamente uma árvore de cliques. A caracterização da classe decorre diretamente de seus separadores minimais de vértices.

**Teorema 3.6.** *Seja  $G = (V, E)$  um grafo cordal.  $G$  é unicamente representável se e somente se não existe continência própria entre pares de separadores minimais de vértices distintos e todos têm multiplicidade 1.*

Um grafo de intervalo que tem uma única árvore de cliques é chamado um *grafo de intervalo unicamente representável* ou, simplesmente, *ur-intervalo*.

Na Figura 3.3 vemos três grafos. O grafo  $G$  não é ur-cordal porque o *smv*  $\{e, f\}$  tem multiplicidade 2. O grafo  $G'$  é um grafo de intervalo, mas não é ur-intervalo, porque o *smv*  $\{a\}$  está contido no *smv*  $\{a, c\}$ ; já o grafo  $G''$  é ur-intervalo.

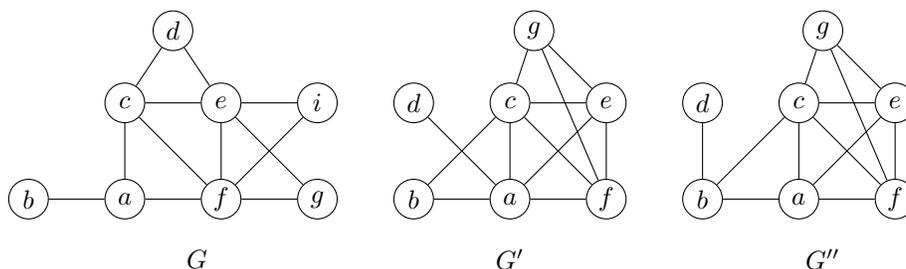


Figura 3.3: Exemplos de grafos

### 3.3 Grafos ptolemaicos

Um ramo bem rico e não muito estudado na hierarquia de cordais é formado pelos grafos ptolemaicos e suas subclasses. Um grafo cordal conexo  $G$  é *ptolemaico* se quaisquer vértices  $u, v, w, x$  de  $G$  satisfazem a desigualdade ptolemaica

$$d(u, v)d(w, x) \leq d(u, w)d(v, x) + d(u, x)d(v, w).$$

O teorema que se segue apresenta uma caracterização clássica para os grafos ptolemaicos, encontrada na literatura, utilizando subgrafos proibidos e cliques maximais.

**Teorema 3.7.** *As seguintes condições são equivalentes:*

1.  $G$  é ptolemaico;
2.  $G$  é cordal e livre de gema;
3. Para quaisquer cliques maximais não disjuntas  $Q$  e  $Q'$  de  $G$ ,  $Q \cap Q'$  separa vértices de  $Q \setminus Q'$  e  $Q' \setminus Q$ .

O grafo de cliques reduzido, apresentado no Capítulo 2, Seção 2.4, suporta uma caracterização alternativa para essa classe.

**Teorema 3.8.** *Um grafo cordal  $G$  é ptolemaico se e somente se o grafo de cliques reduzido e o grafo de interseção de cliques são iguais.*

Demonstração:

( $\rightarrow$ ) De forma equivalente temos que provar que quaisquer duas cliques maximais de  $G$  distintas não disjuntas formam um par de separação se e somente se o grafo de interseção de cliques e o grafo de cliques reduzido de  $G$  são iguais.

Sejam  $C(G) = (V_c, E_c)$  o grafo de interseção de cliques e  $C_r(G) = (V_r, E_r)$  o grafo de cliques reduzido de um grafo ptolemaico  $G$ . Pelas definições,  $V_c = V_r = \mathbb{Q}$  (conjunto de cliques maximais) e, pelo Teorema 2.15,  $E_r \subseteq E_c$ . Considere uma aresta  $\{Q, Q'\} \in E_c$ . Pelo Teorema 3.7, quaisquer duas cliques maximais distintas não disjuntas  $Q$  e  $Q'$  de  $G$  formam um par de separação. Pela definição de grafo de cliques reduzido, a aresta  $\{Q, Q'\} \in E_r$ . Logo,  $E_c = E_r$ .

( $\leftarrow$ ) Considere  $C(G) = C_r(G)$ . Pelo Teorema 2.14, todas as interseções de cliques maximais são separadores minimais de vértices de  $G$ , isto é, formam pares de separação, ou ainda, são separadas por sua interseção. Pelo Teorema 3.7,  $G$  é ptolemaico. ■

No Capítulo 2, Seção 2.1, vimos que é imediato deduzir um procedimento indutivo para construir um grafo cordal  $G$ , percorrendo um esquema de eliminação perfeita em ordem reversa: iniciando com o grafo trivial  $G_n = (\{v_n\}, \emptyset)$ , adicione, a cada passo, um novo vértice  $v_i$ ,  $i = n - 1, \dots, 1$ , adjacente aos vértices do conjunto de adjacência monótono de  $v_i$ , isto é, a clique  $X_\sigma(v_i)$  de  $G_{i+1}$ . É possível particularizar esta construção para algumas classes de grafos cordais restringindo a clique já existente, embora isto não seja conhecido para grafos ptolemaicos. Entretanto, é conhecida uma construção indutiva para grafos ptolemaicos, que mostramos a seguir.

Seja  $G = (V, E)$  um grafo,  $v \in V$  e  $u \notin V$ . Uma extensão de  $G$  para um grafo  $G' = (V', E')$  é uma *extensão de vértice único* se ela se enquadra em um dos seguintes casos:

- ( $\alpha$ )  $V' = V \cup \{u\}$  e  $E' = E \cup \{v, u\}$ , isto é, um vértice pendente é adicionado.
- ( $\beta$ )  $V' = V \cup \{u\}$  e  $E' = E \cup \{\{x, u\} : x \in N[v]\}$ , isto é,  $u$  é um gêmeo verdadeiro de  $v$ .

( $\gamma$ )  $V' = V \cup \{u\}$  e  $E' = E \cup \{\{x, u\} : x \in N(v)\}$ , isto é,  $u$  é um gêmeo falso de  $v$ .

Vamos mostrar o resultado dessas operações sobre um grafo bem simples, um grafo completo com 4 vértices (Figura 3.4).  $G'$  mostra o resultado da operação  $\alpha$  aplicada ao vértice  $d$ ;  $G''$  o resultado da operação  $\beta$  aplicada ao vértice  $b$  (obteríamos o mesmo resultado se a operação fosse aplicada ao vértice  $a$  ou ao vértice  $c$ ) e  $G'''$  mostra o resultado da operação  $\gamma$  aplicada ao vértice  $u$ .

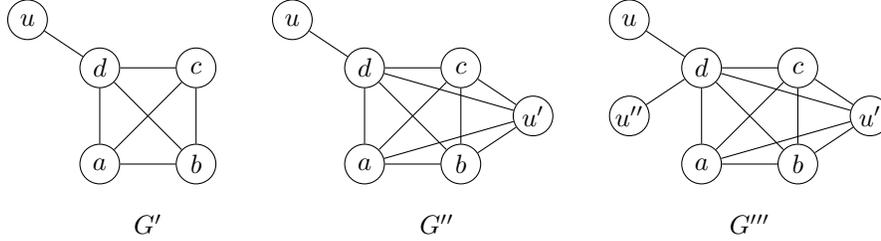


Figura 3.4: Operações  $\alpha$ ,  $\beta$  e  $\gamma$

Essas operações dão origem a outra caracterização para grafos ptolemaicos.

**Teorema 3.9.** *Um grafo  $G$  é ptolemaico se e somente se  $G$  é obtido a partir de um único vértice por uma seqüência de extensões de vértice único sendo a aplicação da operação  $\gamma$  restrita aos vértices simpliciais.*

O caso especial de  $\gamma$ , definido no Teorema 3.9, é denotado por  $\gamma^*$  e incluído no conjunto de regras para construir a extensão de vértice único de um grafo.

Com base neste conceito, definimos uma seqüência de construção para um grafo ptolemaico  $G = (V, E)$ . Uma *seqüência de extensões de vértice único (sext)* de  $G$  é uma seqüência de triplas

$$\pi(G) = \langle \pi(1), \dots, \pi(n) \rangle$$

sendo  $\pi(i) = (x_i, v_i, u_i)$ ,  $i = 2, \dots, n$ , tal que

1.  $x_i \in \{\alpha, \beta, \gamma^*\}$ ;
2.  $v_i = u_j$ ,  $j < i$ ;
3.  $u_i \neq u_k$ ,  $1 \leq k \leq i - 1$ ;

e  $\pi(1)$  é uma tripla inicial  $(\emptyset, \emptyset, u_1)$  onde o símbolo  $\emptyset$  denota a ausência de parâmetro. O grafo  $G[\{u_1, \dots, u_i\}]$  é denotado  $G_{(i)}$ ,  $1 \leq i \leq n$ ;  $G_{(0)} = (\emptyset, \emptyset)$  e  $G_{(n)} = G$ .

Vamos voltar aos grafos da Figura 3.4. Uma *sext* do grafo é

$$\pi(G''') = \langle (\emptyset, \emptyset, a), (\beta, a, b), (\beta, a, c), (\beta, a, d), (\alpha, d, u), (\beta, b, u'), (\gamma^*, u, u'') \rangle.$$

As operações utilizadas na construção indutiva de um grafo ptolemaico nos ajudam a provar outra caracterização, dessa vez baseada nos separadores minimais de vértices de um grafo cordal.

Primeiramente, alguns conceitos precisam ser revistos. Seja  $\mathcal{F} = \{F_1, \dots, F_k\}$ ,  $F_i \subset V$ ,  $1 \leq i \leq k$ , uma família de conjuntos.  $\mathcal{F}$  é uma *família laminar* se  $F_i \cap F_j \neq \emptyset$  implica que  $F_i \subseteq F_j$  ou  $F_j \subseteq F_i$ ,  $1 \leq i, j \leq k$ . Em outras palavras, uma família de

conjuntos é chamada *laminar* se quaisquer dois conjuntos ou são disjuntos ou um é subconjunto do outro. Uma família vazia é laminar. Observe que a laminaridade de conjuntos é uma propriedade hereditária.

**Teorema 3.10.** *Seja  $G = (V, E)$  um grafo cordal.  $G$  é ptolemaico se e somente se a família de separadores minimais de vértices contida em cada clique maximal do grafo é laminar.*

Demonstração:

( $\rightarrow$ ) A prova é por indução. O grafo ptolemaico  $G = (\{u_1\}, \emptyset)$  tem uma clique maximal e a laminaridade da família de separadores minimais de vértices é trivialmente satisfeita. Seja  $G = (V, E)$  um grafo ptolemaico com  $|V| = n - 1$  tal que a família de separadores minimais de vértices contida em cada clique maximal do grafo é laminar e seja  $G' = (V', E')$  uma extensão de vértice único de  $G$  pela aplicação de uma das operações  $\alpha$ ,  $\beta$  ou  $\gamma^*$ ,  $v \in V$ ,  $u \notin V$  e  $V' = V \cup \{u\}$ .

Três casos podem ocorrer:

- $(\alpha, v, u)$

Nesse caso,  $E' = E \cup \{\{v, u\}\}$  e  $G'$  tem uma clique maximal  $Q$  a mais que  $G$ , separada pelo  $smv \{v\}$ . Como o novo  $smv$  é um conjunto unitário, ele mantém a laminaridade de todas as famílias às quais ele pertence em  $G'$ .

- $(\beta, v, u)$

O vértice  $u$  é um gêmeo verdadeiro de  $v$  em  $G'$ . Se  $v$  é simplicial em  $G$ , a clique maximal à qual ele pertence é aumentada pela inclusão do vértice  $u$ , os separadores minimais de vértices do grafo não são modificados e a laminaridade é preservada; caso contrário, todas as cliques maximais e todos os separadores minimais de vértices aos quais  $v$  pertence são aumentados com a inclusão do vértice  $u$ . Então, a propriedade de inclusão desses conjuntos é preservada e, conseqüentemente, também a laminaridade.

- $(\gamma^*, v, u)$

Por definição,  $v$  é um vértice simplicial em  $G$ . Se  $v$  é um vértice pendente,  $(\gamma^*, v, u)$  é equivalente a  $(\alpha, w, u)$  onde  $N(v) = \{w\}$ .

Considere  $v$  um vértice não pendente e  $Q$  a clique maximal tal que  $v \in Q$ . Seja  $\mathcal{F}_G(Q)$  a família de separadores minimais de vértices que pertencem a  $Q$  no grafo  $G$ . Os vértices  $u$  e  $v$  são gêmeos falsos em  $G'$ . O grafo  $G'$  tem uma clique maximal  $Q' = N(v) \cup \{u\}$  a mais do que  $G$ . Se  $N(v)$  já constitui um  $smv$  de  $G$ , para qualquer clique maximal  $X$  de  $G'$ ,  $\mathcal{F}_{G'}(X) = \mathcal{F}_G(X)$  e a laminaridade é mantida. Caso contrário,  $N(v)$  é um novo  $smv$ . Como  $v$  é também simplicial em  $G'$ , ele pertence somente a  $Q$  e  $\mathcal{F}_{G'}(Q) = \mathcal{F}_G(Q) \cup \{N(v)\}$ . Por hipótese,  $\mathcal{F}_G(Q)$  é laminar. Além disso,  $N(v)$  contém todos os membros de  $\mathcal{F}_G(Q)$ . Então,  $\mathcal{F}_{G'}(Q) = \mathcal{F}_{G'}(Q')$  é uma família laminar. Todas as famílias pertencentes a outras cliques maximais não são modificadas.

Conseqüentemente, se  $G$  é um grafo ptolemaico então a família de separadores minimais de vértices contida em cada clique maximal do grafo é laminar.

( $\leftarrow$ ) Considere um grafo  $G$  não ptolemaico (porém cordal) tal que todas as famílias de separadores minimais de vértices em cada clique maximal são laminares (propriedade  $\mathcal{P}$ ). Pelo Teorema 3.7,  $G$  contém um grafo gema como subgrafo

induzido. É possível remover todos os vértices de  $G$  exceto os cinco vértices que pertencem ao grafo gema. Como a propriedade  $\mathcal{P}$  é hereditária, o subgrafo gema satisfaz  $\mathcal{P}$ , o que é uma contradição. Então,  $G$  é ptolemaico. ■

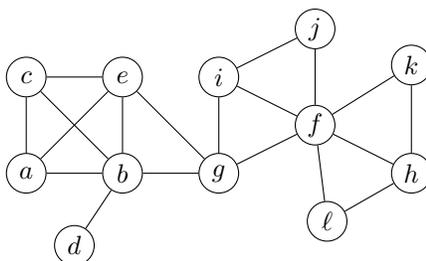


Figura 3.5: Exemplo de grafo ptolemaico

Observe o grafo ptolemaico da Figura 3.5. Por exemplo, a clique maximal  $\{a, b, c, e\}$  contém dois separadores minimais,  $\{b, e\}$  e  $\{b\}$ , que formam uma família laminar. Da mesma forma, a clique  $\{i, f, g\}$ ; os separadores contidos nessa clique são  $\{f, i\}$ ,  $\{f\}$  e  $\{g\}$ , formando uma família laminar.

### 3.3.1 Subclasses de grafos ptolemaicos

Algumas subclasses de grafos ptolemaicos conhecidas podem ser identificadas. Os grafos cordais laminares foram definidos recentemente, tornando mais estrita a condição de laminaridade dos separadores minimais de vértices do grafo. Os grafos bloco, também subclasse dos ptolemaicos, são conhecidos desde a década de 60. A terceira subclasse que será ainda mencionada, os grafos estritamente cordais, serão bem explorados no Capítulo 5.

Um grafo cordal é um *grafo cordal laminar* quando o conjunto  $\mathbb{S}$  de todos os separadores minimais de vértices é laminar. Essa classe pode ser caracterizada por subgrafos proibidos.

**Teorema 3.11.** *Seja  $G = (V, E)$  um grafo cordal.  $G$  é um grafo cordal laminar se e somente se  $G$  é livre de gema e de duplo diamante.*

Demonstração:

( $\rightarrow$ ) Seja  $G$  um grafo cordal laminar. Vamos supor que o grafo gema é um subgrafo induzido de  $G$ . Como a laminaridade no conjunto de separadores minimais de vértices é uma propriedade hereditária, o grafo gema é laminar. Contradição. A prova é análoga para o duplo diamante.

( $\leftarrow$ ) Considere  $G$  um grafo cordal não laminar. Existem  $S, S' \in \mathbb{S}$ ,  $S \neq S'$  tais que  $S \cap S' \neq \emptyset$ ,  $S \not\subseteq S'$  e  $S' \not\subseteq S$ . Vamos supor que  $v \in S \cap S'$ ,  $u \in S, u \notin S'$  e  $w \in S', w \notin S$ . Seja  $\mathcal{F}_Q = \{S \in \mathbb{S} : S \subset Q\} \subset \mathbb{S}$ . Dois casos devem ser estudados.

Existe uma clique maximal  $Q$  tal que  $S \cup S' \subseteq Q$ . Consequentemente,  $\mathcal{F}_Q$  não é laminar; pelo Teorema 3.10,  $G$  não é ptolemaico e, pelo Teorema 3.7,  $G$  contém uma gema.

No segundo caso, para toda clique maximal  $Q$  de  $G$ ,  $S \cup S' \not\subseteq Q$ . Assim, existem quatro cliques distintas, sem perda de generalidade, vamos nomeá-las por  $Q_1, Q_2, Q_3$  e  $Q_4$  tais que  $Q_1 \cap Q_2 = S$  e  $Q_3 \cap Q_4 = S'$ . Se  $Q_i \cap Q_j = S'' \neq S \cap S'$ ,

$i = 1, 2, j = 3, 4$ , então  $\mathcal{F}_{Q_i}$  e  $\mathcal{F}_{Q_j}$  não são laminares e obtemos uma gema. Caso contrário,  $Q_1 \cap Q_3 = Q_1 \cap Q_4 = Q_2 \cap Q_3 = Q_2 \cap Q_4 = S \cap S'$ . Considerando os vértices  $x_1 \in Q_1 \setminus Q_2, x_2 \in Q_2 \setminus Q_1, x_3 \in Q_3 \setminus Q_4, x_4 \in Q_4 \setminus Q_3$ , o conjunto  $\{x_1, x_2, x_3, x_4, u, v, w\}$  induz um duplo diamante em  $G$ . ■

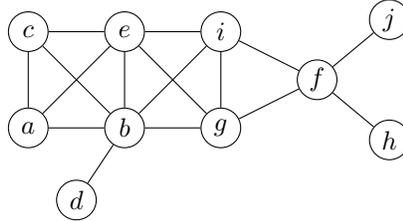


Figura 3.6: Exemplo de grafo cordal laminar

O grafo da Figura 3.6 é um grafo cordal laminar. Seu conjunto de separadores minimais de vértices é  $\mathbb{S} = \{\{e, b\}, \{b\}, \{i, g\}, \{f\}\}$ , que é um conjunto laminar.

Os grafos cordais laminares são ptolemaicos. O grafo da Figura 3.5, que é ptolemaico, não é laminar: os separadores minimais de vértices  $\{i, f\}$  e  $\{f, h\}$  não formam uma família laminar.

Outra subclasse importante é a constituída pelos grafos bloco, que já foi mencionada no Capítulo 1, Seção 1.3. Lá, um *grafo bloco* foi definido como um grafo conexo onde cada componente biconexa é um grafo completo. A importância da classe nesse texto nos leva a mencionar outras caracterizações para a mesma. Vamos ver que cada uma delas pode ser útil em diferentes ocasiões.

**Teorema 3.12.** *São equivalentes:*

1.  $G$  é um grafo bloco.
2.  $G$  é um grafo conexo onde cada componente biconexa é um grafo completo.
3.  $G$  é um grafo cordal livre de diamante.
4.  $G$  é cordal e qualquer separador minimal de vértices é um conjunto unitário.

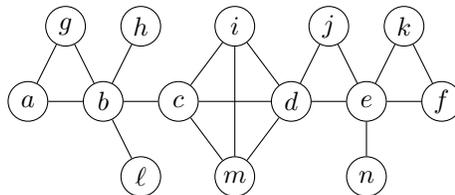


Figura 3.7: Exemplo de grafo bloco

É fácil ver que nesse caso os separadores minimais de vértices contidos em uma clique maximal obedecem trivialmente ao Teorema 3.10, e um grafo bloco é um grafo ptolemaico. E mais, obedecem à definição de grafos cordais laminares. Conclui-se que árvores constituem uma subclasse de grafos bloco, e grafos caterpillar constituem uma subclasse de árvores. Uma hierarquia imediata se forma assim. Na

Figura 3.7 temos um grafo bloco e podemos observar que seus separadores minimais de vértices constituem uma família laminar.

Os grafos bloco estão no cerne da definição de uma classe que vem sendo bastante estudada nos últimos dez anos: os *grafos estritamente cordais*, que é o grafo obtido pela adição de zero ou mais gêmeos verdadeiros a cada vértice de um grafo bloco  $G$ .

Grafos estritamente cordais possuem também propriedades baseadas em separadores minimais de vértices, como apresentado no seguinte teorema, que será provado mais tarde.

*Seja  $G$  um grafo cordal e  $\mathbb{S}$  o conjunto de separadores minimais de vértices.  $G$  é um grafo estritamente cordal se e somente se quaisquer dois separadores minimais de vértices distintos de  $\mathbb{S}$  são disjuntos.*

Não é difícil concluir que, também nesse caso, os separadores minimais de vértices estabelecem famílias laminares nas cliques maximais; eles simplesmente não se interceptam. Então, grafos estritamente cordais são ptolemaicos.

### 3.4 A hierarquia

A Figura 3.8 apresenta a hierarquia de subclasses de cordais mencionada até agora.

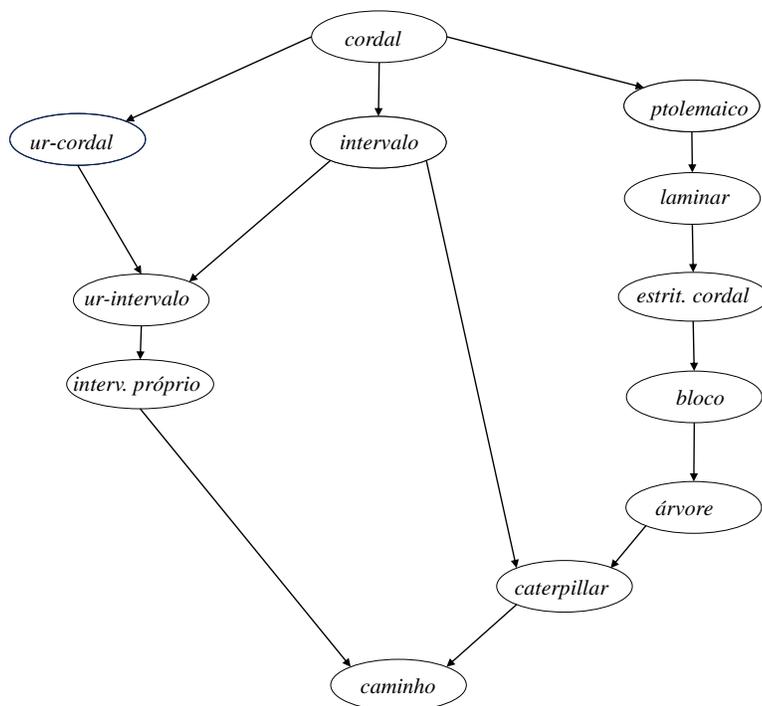


Figura 3.8: Hierarquia parcial

### 3.5 Exercícios

1. Dado o grafo de intervalo  $G$ , a família de intervalos é única?
2. O grafo  $G''$  da Figura 3.3 é um grafo de intervalo. Mostre um caminho de cliques de  $G''$  e apresente uma sequência gulosa para o grafo.
3. Demonstre ou forneça um contraexemplo: A sequência gulosa é única.
4. O *grafo estrela* com  $n + 1$  vértices,  $n > 0$ , denotado  $K_{1,n}$ , tem a seguinte definição:

$$K_{1,n} = (\{1, \dots, n + 1\}, \{\{1, i\} : i = 2, \dots, n + 1\}).$$

- (a) Prove que o grafo estrela é um grafo de intervalo.
  - (b) Apresente um caminho de cliques para  $K_{1,6}$ .
  - (c) Apresente uma sequência gulosa para  $K_{1,6}$  baseada no caminho de cliques do item (b).
5. Afirmamos, no texto, que o grafo  $G$  da Figura 3.3 não é ur-cordal. Mostre, então, duas árvores de cliques distintas para  $G$ .
  6. Apresente duas árvores de cliques para o grafo  $G'$  da Figura 3.3, sendo pelo menos uma delas um caminho de cliques.
  7. Considere o grafo  $G'$  da Figura 3.3. A remoção do vértice  $d$  o transformaria num grafo ur-cordal? Explique porquê.
  8. Apresente uma *sext* para os grafos exibidos nas figuras:
    - (a) Figura 3.5.
    - (b) Figura 3.6.
    - (c) Figura 3.7.
  9. Se  $G = (V, E)$  é um grafo bloco então existe um vértice  $v \in V$  tal que  $G \setminus \{v\}$  também é um grafo bloco conexo? É possível caracterizá-lo?
  10. Apresente um exemplo de grafo estritamente cordal que não seja um grafo bloco.

### 3.6 Notas bibliográficas

Subclasses de grafos cordais têm sido exaustivamente estudadas. Um levantamento sistemático de referência sobre o assunto se encontra em Brandstädt *et al.* [9]. O sistema computacional *Information System on Graph Classes* disponibiliza no site [www.graphclasses.org](http://www.graphclasses.org) um auxílio precioso aos interessados.

Grafos de intervalo têm sido muito pesquisados, desde 1962, quando caracterizados no trabalho de Lekkerkerker e Boland [45], que exibiram os subgrafos proibidos. A caracterização apresentada por Gilmore e Hoffman [28], (Teorema 3.2), é utilizada pela maioria dos algoritmos de reconhecimento. Booth e Lueker [7] empregaram a estrutura de dados árvores PQ para encontrar a ordem linear das cliques maximais,

se esta existe; já Korte e Mohring [42], uma variação da árvore PQ. Outros algoritmos, utilizando o percurso em largura lexicográfica, foram apresentados por Hsu e Ma [38], Habib *et al.* [32] e Corneil *et al.* [18]. Mais recentemente, Kratsch *et al.* [43] modificaram o algoritmo de Korte e Mohring para apresentar um algoritmo certificador de reconhecimento.

Os grafos cordais unicamente representáveis e os grafo *ur*-intervalo foram definidos por Kumar e Madhavan [44]. A subclasse das *k*-árvores unicamente representáveis foi estudada por Markenzon *et al.* [50] e por Markenzon *et al.* [52].

Grafos ptolemaicos foram introduzidos por Howarka [37]. Diferentes caracterizações foram apresentadas por Bandelt e Mulder [1] (sequência de operações), Uehara e Uno [74] (separadores minimais de vértices) e Markenzon e Waga [55] (grafos de cliques reduzido). Das subclasses mencionadas, a mais conhecida é a dos grafos bloco, definidos por Harary [35] em 1963 e a caracterização por subgrafo proibido pode ser encontrada em [1]. A classe dos grafos cordais laminares foi definida por Markenzon e Waga [55].



## Capítulo 4

# Grafos de Intervalo Próprio

Neste capítulo vamos rever grafos de intervalo próprio. Conhecidos desde a década de 60, possuem fortes propriedades estruturais. Ao longo dos anos, diversos algoritmos de reconhecimento foram propostos e muitos problemas difíceis para grafos de intervalo foram resolvidos para essa subclasse.

Primeiramente, estudamos as abordagens utilizadas no reconhecimento, introduzindo uma atualização de um dos primeiros algoritmos apresentados. Para tal, a definição da sequência de vértices canônica é revista. Para essa classe, vamos ver a solução de uma aplicação importante: determinamos um ciclo hamiltoniano do grafo.

O capítulo traz também o estudo de uma subclasse dos grafos de intervalo próprio, os grafos split-indiferença. Outro problema básico é resolvido: a contagem dos grafos pertencentes à classe.

### 4.1 A classe

Como vimos no capítulo anterior, Seção 3.1, um *grafo de intervalo próprio (GIP)* é o grafo de interseção de uma família intervalos em uma reta real no qual nenhum intervalo está propriamente contido em outro.

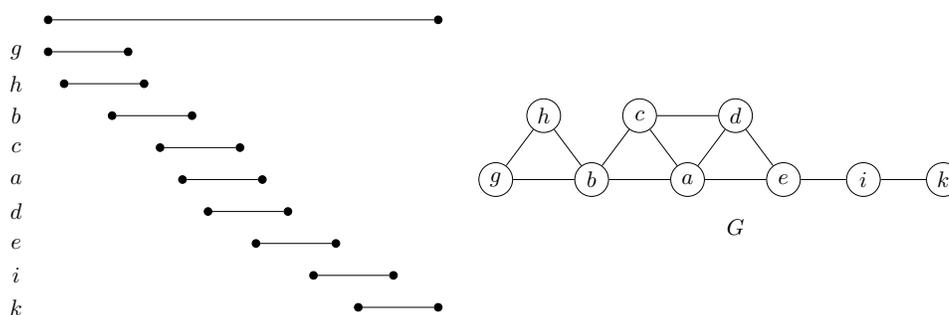


Figura 4.1: Intervalos e o grafo de intervalo próprio

Na Figura 4.1, vemos uma família de intervalos e seu grafo de interseção  $G$ . O grafo  $G$  é um grafo de intervalo próprio. Note que todos os intervalos do modelo têm o mesmo tamanho.

Grafos de intervalo próprio foram definidos ao longo dos anos com outros nomes. A seguir, apresentamos algumas definições.

Um *grafo de intervalo unitário* é um grafo de intervalo onde todos os intervalos são unitários ou têm o mesmo tamanho.

Um grafo  $G = (V, E)$  é um *grafo indiferença* se existe uma função  $f : V \rightarrow \mathbf{R}$  e um número real  $k$  tal que para quaisquer  $u, v \in V$ ,  $\{u, v\} \in E$  se  $|f(u) - f(v)| \leq k$ .

Diversas caracterizações são conhecidas na literatura para grafos de intervalo próprio, resumidas no próximo teorema.

**Teorema 4.1.** *São equivalentes:*

1.  $G$  é um grafo de intervalo unitário.
2.  $G$  é um grafo de intervalo próprio.
3.  $G$  é um grafo indiferença.
4.  $G$  é um grafo de intervalo livre de garra.
5. Existe uma ordenação de  $V$  tal que para todo  $v \in V$ , os vértices de  $N[v]$  são consecutivos na ordenação (condição de vizinhança).
6. Existe uma ordenação de  $V$  tal que os vértices pertencentes à mesma clique maximal são consecutivos na ordenação (condição de clique).

Uma propriedade de grafos de intervalo próprio, pouco mencionada, é provada no Teorema 4.2.

**Teorema 4.2.** *Se  $G$  é um grafo de intervalo próprio então  $G$  é um grafo ur-intervalo.*

Demonstração:

Seja  $\mathbb{S}$  o conjunto de separadores minimais de vértices de  $G$ . Se  $G$  é um grafo unicamente representável então, por definição,  $\mu(S) = 1$ ,  $S \not\subset S'$  e  $S' \not\subset S$ , para quaisquer  $S, S' \in \mathbb{S}$ . Vamos provar então que, se  $\mu(S) \neq 1$  ou  $S \subset S'$ , para certos  $S, S' \in \mathbb{S}$ ,  $G$  não é grafo de intervalo próprio.

Se  $\mu(S) > 1$  então existem ao menos três cliques maximais  $Q, Q'$  e  $Q''$  tais que  $Q \cap Q' = Q \cap Q'' = Q' \cap Q'' = S$ . Para  $x \in Q \setminus (Q' \cup Q'')$ ,  $y \in Q' \setminus (Q \cup Q'')$  e  $z \in Q'' \setminus (Q \cup Q')$  existe um caminho de  $x$  para  $y$ ,  $x$  para  $z$  e  $y$  para  $z$  que passa por ao menos um vértice de  $S$ , digamos  $w$ . Então esses quatro vértices induzem uma garra.

Se  $S \subset S'$  existem ao menos três cliques maximais tais que  $S \subset Q$ ,  $S' \not\subset Q$ ,  $S' \subset Q'$  e  $S' \subset Q''$ . Obviamente  $S$  é subconjunto de  $Q'$  e  $Q''$ . Então, o raciocínio da prova é similar ao apresentado para o caso anterior e  $G$  possui um subgrafo induzido que é uma garra. Em ambos os casos,  $G$  não é um grafo de intervalo próprio. ■

Na Figura 4.2 são apresentados exemplos de grafos pertencentes à hierarquia de grafos de intervalo mencionada até agora. O grafo  $G$  é de intervalo, mas não é ur-intervalo; o *smv*  $\{e, f\}$  tem multiplicidade 3. O grafo  $G'$  é ur-intervalo, mas não é próprio, uma vez que os vértices  $a, b, d$  e  $f$  formam um subgrafo garra; o grafo  $G''$  é de intervalo próprio.

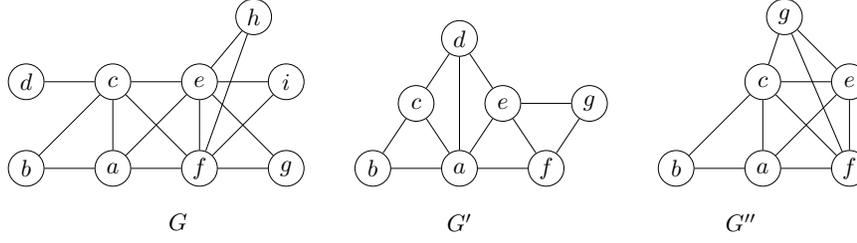


Figura 4.2: Grafos de intervalo, ur-intervalo e de intervalo próprio

A determinação da sequência gulosa, apresentada no Capítulo 3, Seção 3.1, nos permite estabelecer uma caracterização alternativa para grafos de intervalo próprio. Primeiramente apresentamos uma nova ordenação linear dos vértices.

Considere uma sequência dos vértices  $\delta = \langle v_1, \dots, v_n \rangle$  de  $V$ . Esta sequência é denominada *canônica* quando para quaisquer vértices  $v_i, v_j, v_k$ ,  $i < j < k$ , tem-se

$$\text{se } \{v_i, v_k\} \in E \text{ então } \{v_i, v_j\} \in E \text{ e } \{v_j, v_k\} \in E. \quad (4.1.1)$$

A seguinte conclusão é imediata.

**Lema 4.1.** *Toda sequência canônica é uma sequência gulosa.*

**Teorema 4.3.** *Um grafo cordal  $G = (V, E)$  é um grafo de intervalo próprio se e somente se existe uma sequência canônica dos vértices de  $V$ .*

Demonstração:

( $\rightarrow$ ) Como  $G$  é um grafo de intervalo, considere sua representação por intervalos  $\{I_v = [a_v, b_v], v \in V\}$ . Seja a sequência  $\delta = \langle v_1, \dots, v_n \rangle$  obtida da relação de ordem linear em  $V$  tal que sempre que  $a_{v_i} < a_{v_j}$  ou que os intervalos coincidam ( $I_{v_i} = I_{v_j}$ ),  $v_i$  precede  $v_j$  ( $i < j$ ) na sequência.

Temos que provar que esta sequência é canônica para o grafo  $G$ , isto é, se  $v_i, v_j, v_k \in V$ , com  $i < j < k$  e  $\{v_i, v_k\} \in E$ , então  $\{v_i, v_j\} \in E$  e  $\{v_j, v_k\} \in E$ .

Se  $I_{v_i} = I_{v_k}$  então  $I_{v_i} = I_{v_j} = I_{v_k}$ ,  $I_{v_i} \cap I_{v_j} \neq \emptyset$  e  $I_{v_j} \cap I_{v_k} \neq \emptyset$ . Conclui-se que  $\{v_i, v_j\} \in E$  e  $\{v_j, v_k\} \in E$ . Por outro lado, seja  $a_{v_i} < a_{v_k}$ . Por hipótese,  $\{v_i, v_k\} \in E$ , ou seja,  $I_{v_i} \cap I_{v_k} \neq \emptyset$ . Podemos concluir que  $a_{v_k} \leq b_{v_i}$ . Dois casos devem ser analisados:

- $I_{v_j}$  coincide com  $I_{v_i}$  ou com  $I_{v_k}$ . Temos que  $\{v_i, v_j\} \in E$  e  $\{v_j, v_k\} \in E$ .
- $a_{v_i} < a_{v_j} < a_{v_k}$ . Mas  $a_{v_k} \leq b_{v_i}$ . Assim  $a_{v_i} < a_{v_j} < a_{v_k} \leq b_{v_i}$ . Então  $I_{v_i} \cap I_{v_j} \neq \emptyset$  implica em  $\{v_i, v_j\} \in E$  e  $I_{v_j} \cap I_{v_k} \neq \emptyset$  implica em  $\{v_j, v_k\} \in E$ .

Logo,  $\delta$  é uma sequência canônica de  $V$ .

( $\leftarrow$ ) Seja  $\delta$  uma sequência canônica de  $V$ . O Lema 4.1 nos diz que  $G$  é um grafo de intervalo. Pelo Teorema 4.1, temos que provar que  $G$  é livre de garra para que seja um grafo de intervalo próprio. Vamos supor que  $G$  contém uma garra com vértices  $a, b, c$  e  $d$  e arestas  $\{a, b\}$ ,  $\{a, c\}$  e  $\{a, d\}$ . O vértice  $a$  não precede  $b, c$  e  $d$  na sequência canônica  $\delta$  pois senão, pela Propriedade (4.1.1),  $b, c$  e  $d$  seriam adjacentes (contradição). Analogamente,  $b, c$  e  $d$  não podem preceder  $a$ .

Considere que o vértice  $a$  se encontre intercalado entre os vértices  $b, c$  e  $d$ . Sem perda de generalidade, vamos analisar duas subsequências,  $\langle b, a, c, d \rangle$  e  $\langle b, c, a, d \rangle$ .

Na primeira  $a$  e  $d$  são adjacentes, então  $c$  e  $d$  são. Na segunda, como  $b$  e  $a$  são adjacentes,  $b$  e  $c$  também o são. Ambos os casos implicam em contradição. ■

Temos duas consequências imediatas.

**Corolário 4.1.** *Um grafo de intervalo  $G = (V, E)$  é um grafo de intervalo próprio se e somente se uma seqüência gulosa de  $V$  satisfaz a Propriedade (4.1.1).*

**Corolário 4.2.** *Considere um grafo de intervalo próprio  $G = (V, E)$ . Existe uma seqüência de vértices que é uma seqüência canônica, satisfaz a condição de vizinhança e a condição de clique.*

Por exemplo, uma seqüência canônica dos vértices do grafo da Figura 4.1 é:  $\delta = \langle g, h, b, c, a, d, e, i, k \rangle$ . Observe que as condições de clique e vizinhança são satisfeitas. Os vértices pertencentes às cliques  $\{b, g, h\}$ ,  $\{a, b, c\}$ ,  $\{a, c, d\}$ ,  $\{a, d, e\}$ ,  $\{e, i\}$  e  $\{i, k\}$  aparecem consecutivamente. Para as vizinhanças fechadas, o resultado também pode ser observado. Por exemplo,  $N[d] = \{a, c, d, e\}$ ; esses vértices ocupam as posições 4, 5, 6 e 7 em  $\delta$ .

É possível também relacionar a seqüência canônica com o esquema de eliminação bicompatível, apresentado no Capítulo 2, Seção 2.1.2.

**Teorema 4.4.** *Considere um grafo de intervalo próprio  $G = (V, E)$ . Toda seqüência canônica dos vértices de  $V$  é um esquema de eliminação bicompatível para  $G$ .*

Demonstração:

A seqüência reversa de uma seqüência gulosa é um esquema de eliminação perfeita (Teorema 3.5). Resta provar que, para um grafo de intervalo próprio, a própria seqüência também o é. Sabemos que, numa seqüência canônica, os vértices pertencentes à mesma clique maximal são consecutivos (condição de clique). Para grafos de intervalo próprio, podemos classificar os vértices de uma clique maximal em três tipos: vértices simpliciais (tipo 1), vértices que aparecem em um único separador minimal de vértices (tipo 2) e vértices que aparecem em dois ou mais separadores de vértices (tipo 3). Na construção da seqüência gulosa executada pelo Algoritmo 3.1, os vértices são tratados diferentemente em cada um desses tipos: para cada clique, vértices simpliciais aparecem em primeiro lugar, depois vértices tipo 2 e, finalmente, vértices tipo 3. Seja  $\delta = \langle v_1, \dots, v_n \rangle$  uma seqüência canônica. Os  $i \geq 1$  primeiros vértices de  $\delta$  são obrigatoriamente simpliciais e assim são os primeiros vértices de um esquema de eliminação perfeita. Se forem removidos, a primeira clique do caminho de cliques desaparece, e os vértices tipo 2 se tornam simpliciais. O raciocínio pode ser repetido, e temos um esquema de eliminação perfeita. ■

Vamos retomar ao exemplo da Figura 4.1. Pelo Teorema 4.4, a seqüência canônica  $\delta$  é um esquema de eliminação bicompatível. Então,  $\delta = \langle g, h, b, c, a, d, e, i, k \rangle$  e  $\delta^{-1} = \langle k, i, e, d, a, c, b, h, g \rangle$  são esquemas de eliminação perfeita.

## 4.2 Reconhecimento

Existem duas abordagens usuais para o reconhecimento de um grafo de intervalo próprio  $G$ . Dado um grafo cordal, pode-se inicialmente verificar se o grafo é de intervalo e então verificar se é próprio. Na segunda abordagem verifica-se diretamente se o grafo cordal  $G$  é próprio.

A primeira abordagem foi desprezada por muitos anos, porque o algoritmo de reconhecimento de grafos de intervalo, apesar de linear, não tem uma implementação trivial. Entretanto, o fato de um grafo de intervalo próprio ser também um grafo ur-intervalo pode simplificar muito a solução. Vamos então utilizar essa estratégia.

As propriedades estabelecidas nos Teoremas 4.2 e 4.3 nos permitem apresentar um reconhecimento bem simples para a classe. Primeiramente construímos uma árvore de cliques do grafo  $G$ . Como o grafo de intervalo próprio é unicamente representável, essa árvore precisa ser um caminho, caso contrário já podemos afirmar que  $G$  não pertence à classe. Em seguida, construímos uma sequência gulosa, testando se essa obedece à condição de clique (Corolário 4.2).

O algoritmo a seguir determina se um grafo cordal  $G$  é um grafo de intervalo próprio ou não.

**Passo 1:** Construir uma árvore de cliques  $T$  de  $G$ ;

Se  $T$  é um caminho de cliques  $\langle Q_1, \dots, Q_q \rangle$  então ir para o Passo 2  
senão  $G$  não é um grafo de intervalo próprio;

**Passo 2:** Construir uma sequência gulosa  $\delta = \langle v_1, \dots, v_n \rangle$ ;

**Passo 3:** Se  $\delta$  satisfaz a condição de clique então  $G$  é grafo de intervalo próprio.

---

**Algoritmo 4.1.** *Reconhecer-Intervalo-Próprio*

---

**Entrada:** grafo  $G = (V, E)$ , caminho de cliques  $\langle Q_1, \dots, Q_q \rangle$  e  
sequência gulosa  $\delta = \langle v_1, \dots, v_n \rangle$ ;  
**Saída:**  $G$  é de intervalo próprio ou *não*;  
**Início**  
   $proprio \leftarrow true$ ;  $j \leftarrow 1$ ;  
  **Enquanto**  $j \leq q$  **faça**  
     $prim(j) \leftarrow \min\{i : v_i \in Q_j\}$ ;                   % extremidades de  $Q_j$  em  $\delta$   
     $ult(j) \leftarrow \max\{i : v_i \in Q_j\}$ ;  
    **Se**  $ult(j) - prim(j) \neq |Q_j| - 1$  **então**           % testar cardinalidade de  $Q_j$   
       $proprio \leftarrow false$ ;  
       $p \leftarrow j$ ;  
       $j \leftarrow q$ ;  
       $j \leftarrow j + 1$ ;  
  **Se**  $proprio$  **então** SIM  
  **senão** NÃO;  
**Fim.**

---

O Passo 1 foi visto no Capítulo 2 e sabemos que é possível resolvê-lo com um algoritmo de complexidade linear. Caso a árvore de cliques seja um caminho, prosseguimos para o Passo 2. O Algoritmo 3.1 determina uma sequência gulosa; ele foi apresentado no Capítulo 3, Seção 3.1, quando revimos grafos de intervalo e, como mostramos, sua complexidade é linear. Nos resta desenvolver o Passo 3, para o qual utilizaremos a condição de clique. Observe que, pelo Corolário 4.2, qualquer uma das condições mencionada poderia ser aqui utilizada. O algoritmo percorre o caminho de cliques, determinando para cada uma delas o menor e o maior índice de seus elementos na ordenação gulosa (rótulos *prim* e *ult*). É imediato que, para que a condição seja obedecida, a diferença desses índices precisa ser a cardinalidade

da clique. Quanto à complexidade do passo, como percorremos os vértices de cada clique maximal, sabemos, pelo Teorema 2.3, que esse cálculo tem complexidade de tempo de  $O(n + m)$ .

Considere o grafo  $G$  da Figura 4.1. Seja um caminho de cliques de  $G$ :

$$\langle \{b, g, h\}, \{a, b, c\}, \{a, c, d\}, \{a, d, e\}, \{e, i\}, \{i, k\} \rangle.$$

Para esse caminho, obtemos a sequência gulosa  $\langle g, h, b, c, a, d, e, i, k \rangle$ .

Mostramos o acompanhamento do algoritmo na tabela a seguir.

$j$	$Q_j$	$prim$	$ult$	$ult - prim =  Q_j  - 1$
1	$\{b, g, h\}$	1	3	$V$
2	$\{a, b, c\}$	3	5	$V$
3	$\{a, c, d\}$	4	6	$V$
4	$\{a, d, e\}$	5	7	$V$
5	$\{e, i\}$	7	8	$V$
6	$\{i, k\}$	8	9	$V$

Assim, a sequência gulosa satisfaz a condição de clique e, conseqüentemente, o grafo é de intervalo próprio.

Considere agora o grafo  $G$  da Figura 4.2. Seja um caminho de cliques de  $G$

$$\langle \{c, d\}, \{a, b, c\}, \{a, c, e, f\}, \{e, f, g\}, \{e, f, h\}, \{e, f, i\} \rangle$$

e a sequência gulosa  $\langle d, c, b, a, e, f, g, h, i \rangle$ . O acompanhamento da execução do algoritmo de reconhecimento é:

$j$	$Q_j$	$prim$	$ult$	$ult - prim =  Q_j  - 1$
1	$\{c, d\}$	1	2	$V$
2	$\{a, b, c\}$	2	4	$V$
3	$\{a, c, e, f\}$	2	6	$F$
7	—	—	—	—

A sequência gulosa não satisfaz a condição de clique: o algoritmo para após processar a clique  $Q_p$  com  $p = 3$ . Concluimos que o grafo é de intervalo mas não próprio.

### 4.3 Um problema: ciclo hamiltoniano

A sequência canônica é fundamental para a solução de problemas em grafos de intervalo próprio. Por exemplo, a solução da determinação de um ciclo hamiltoniano.

Primeiramente vamos mostrar o algoritmo que obtém o ciclo hamiltoniano; depois, apresentaremos a prova de que o algoritmo está correto.

Observando o grafo  $G$  da Figura 4.3, podemos pensar em um algoritmo bastante imediato. Como o grafo é biconexo, precisamos construir dois caminhos que percorram as cliques maximais: o primeiro, de  $Q_1$  a  $Q_q$  e o segundo, de  $Q_q$  a  $Q_1$ . Como todos os separadores minimais de vértices possuem pelo menos dois vértices, podemos ligar, em cada um dos caminhos, um vértice de cada separador ao próximo.

Parece uma solução bem simples, não? Infelizmente, teremos problemas. Para o grafo  $G$  da Figura 4.3, a idéia funciona bastante bem. Podemos considerar o primeiro caminho composto pelos vértices  $c, e, f$ , vértices pertencentes respectivamente

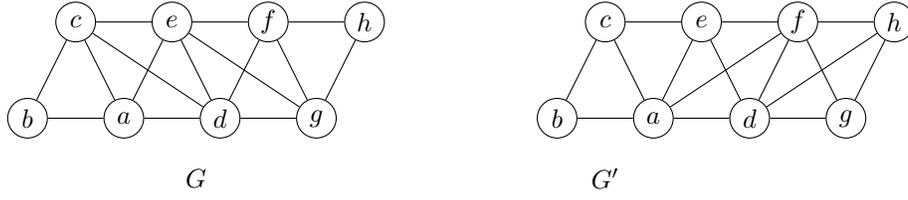


Figura 4.3: Grafos de intervalo próprio hamiltonianos

aos separadores minimais de vértices  $S_1 = \{a, c\}$ ,  $S_2 = \{d, e\}$  e  $S_3 = \{f, g\}$  de  $G$ . O segundo caminho seria  $b, a, d, g, h$  e a composição dos dois caminhos nos dá o ciclo hamiltoniano. Entretanto, para o grafo  $G'$  da mesma figura, a estratégia fracassa. Se escolhermos os vértices  $a, e, f$ , pertencentes respectivamente aos separadores minimais de vértices  $S'_1 = \{a, c\}$ ,  $S'_2 = \{a, e\}$  e  $S'_3 = \{d, f\}$  de  $G'$ , não teremos como construir o segundo caminho, e a solução correta não é encontrada. Vamos então ver como solucionar esse problema. Já podemos adiantar que a solução se apoia mesmo na elaboração de dois caminhos, porém com maior cuidado na escolha dos vértices.

---

**Algoritmo 4.2.** *Determinar-Ciclo-Hamiltoniano*


---

**Entrada:** grafo de intervalo próprio biconexo  $G = (V, E)$ ,  
caminho de cliques  $\langle Q_1, \dots, Q_q \rangle$ ;  $\delta = \langle v_1, \dots, v_n \rangle$ ;  
rótulos  $prim[i], ult[i], i = 1, \dots, q$ ,  
 $S_1 = Q_1 \cap Q_2, \dots, S_{q-1} = Q_{q-1} \cap Q_q$ ,  
**Saída:** ciclo hamiltoniano de  $G$ ;  
**Início**  
 $ciclo \leftarrow \langle \rangle$ ;  
**Para**  $i = 1, \dots, n$  **faça**  $marca(v_i) \leftarrow false$ ;  
**Para**  $i = 1, \dots, q - 1$  **faça** (\*)  
**Se**  $i > 1$  e  $r_{i-1} \in S_i$  **então**  $r_i \leftarrow r_{i-1}$  % determinar vértices restritos  
**senão**  
 $r_i \leftarrow v \in S_i$  tal que  $marca(v) = false$ ;  
 $marca(r_i) \leftarrow true$ ;  
**Para**  $j = prim(i), \dots, ult(i)$  **faça** % incluir restantes em ciclo  
**Se**  $marca(v_j) = false$  **então**  
 $ciclo \leftarrow ciclo || v_j$ ;  
 $marca(v_j) \leftarrow true$ ;  
**Para**  $j = prim(q), \dots, ult(q)$  **faça** (\*\*)  
**Se**  $marca(v_j) = false$  **então**  $ciclo \leftarrow ciclo || v_j$ ;  
 $ciclo \leftarrow ciclo || r_{q-1}$ ;  
**Para**  $i = q - 2, \dots, 1$  **faça** (\*\*\*)  
**Se**  $r_i \neq r_{i+1}$  **então**  $ciclo \leftarrow ciclo || r_i$ ;  
**Fim.**

---

A entrada do algoritmo é o grafo de intervalo próprio  $G$ , seu caminho de cliques  $\langle Q_1, \dots, Q_q \rangle$ , os separadores minimais de vértices  $S_1 = Q_1 \cap Q_2, \dots, S_{q-1} = Q_{q-1} \cap Q_q$ , e uma sequência canônica  $\delta = \langle v_1, \dots, v_n \rangle$ . O algoritmo executa dois percursos no grafo  $G$ . O primeiro percurso é pela sequência das cliques maximais, porém levando também em consideração a sequência canônica. No momento em que a clique  $Q_i$  é analisada, um vértice de  $S_i$  é escolhido e separado como *restrito*, isto

é, será utilizado no segundo percurso. Se  $S_i$  contém o vértice que foi considerado restrito na iteração anterior, este vértice é repetido na iteração  $i$ . Esses vértices especiais são armazenados na variável  $r_i$ . Os outros são incluídos em uma sequência *ciclo*, que durante o primeiro percurso armazena um caminho de vértices da clique  $Q_1$  até a clique  $Q_q$ . O segundo percurso é simples: deve-se concatenar os vértices restritos  $r_i$  ao fim de *ciclo*, na ordem inversa,  $i$  variando de  $q - 1$  a 1. Relembramos que a notação  $\|$ , utilizada no algoritmo, significa concatenação.

Vamos acompanhar o algoritmo para o grafo  $G'$  da Figura 4.3. São dados de entrada:

- o caminho de cliques  $\{\{a, b, c\}, \{a, c, e\}, \{a, d, e, f\}, \{d, f, g, h\}\}$ ;
- os separadores minimais de vértices:  $S_1 = \{a, c\}$ ,  $S_2 = \{a, e\}$  e  $S_3 = \{d, f\}$ ;
- a sequência gulosa  $\delta = \langle b, c, a, e, d, f, g, h \rangle$ ;
- os rótulos  $prim[1] = 1, ult[1] = 3; prim[2] = 2, ult[2] = 4;$   
 $prim[3] = 3, ult[3] = 6; prim[4] = 5, ult[4] = 8.$

Primeiramente, o rótulo *marca* é inicializado com *false* para todos os vértices. No percurso de cliques maximais, as cliques  $Q_1, Q_2$  e  $Q_3$  são computadas, como visto na tabela que se segue.

$i$	$j$	<i>marca</i>								$r_i$	<i>ciclo</i>
		$b$	$c$	$a$	$e$	$d$	$f$	$g$	$h$		
1		F	F	T	F	F	F	F	F	$a$	
	1, 2, 3	T	T	T	F	F	F	F	F		$\langle b, c \rangle$
2		T	T	T	F	F	F	F	F	$a$	
	2, 3, 4	T	T	T	T	F	F	F	F		$\langle b, c, e \rangle$
3		T	T	T	T	T	F	F	F	$d$	
	3, 4, 5, 6	T	T	T	T	T	T	F	F		$\langle b, c, e, f \rangle$

A computação da última clique  $Q_4$  acrescenta ao caminho dois vértices:  $g$  e  $h$ ; a sequência *ciclo* é:  $\langle b, c, e, f, g, h \rangle$ . Finalmente, o caminho formado pelos vértices restritos é acrescentado: *ciclo* =  $\langle b, c, e, f, g, h, d, a \rangle$ .

Assim, o ciclo hamiltoniano do grafo é  $\langle b, c, e, f, g, h, d, a, b \rangle$ .

**Teorema 4.5.** *Seja  $G$  um grafo de intervalo próprio biconexo. O Algoritmo 4.2 computa um ciclo hamiltoniano de  $G$ .*

Demonstração:

O Algoritmo 4.2 estabelece dois caminhos entre  $Q_1$  e  $Q_q$ , criando a sequência *ciclo*. Considere o comando assinalado por (\*). Primeiramente são separados e marcados os vértices restritos e todos os vértices não restritos de  $V$  são incluídos na sequência, e também marcados. Como a sequência canônica está sendo percorrida, a inclusão dos vértices não marcados pertencentes a  $Q_q$  é executada no comando assinalado por (\*\*). No comando assinalado por (\*\*\*), os vértices restritos são incluídos na sequência. Então, cada vértice aparece em *ciclo* uma vez.

É preciso provar que temos um ciclo, isto é, que cada vértice é adjacente ao próximo na sequência. Como o grafo  $G$  é ur-intervalo,  $S_i \setminus S_{i+1} \neq \emptyset$  e  $S_{i+1} \setminus S_i \neq \emptyset$ , para  $1 \leq i < q - 1$ . Isso nos assegura que cada  $r_i$  é definido e assim os vértices restritos formam um caminho.

Resta analisar o caminho formado pelos outros vértices. O algoritmo percorre as cliques maximais de  $G$ . Então, para cada clique, é necessário garantir a entrada e

a saída da clique no caminho. Na iteração  $i = 1$ , os vértices simpliciais e os vértices pertencentes a  $S_1 \setminus \{r_1\}$  são incluídos. Como  $|S_1| \geq 2$ , um caminho é construído. Sejam  $i > 1$  e a clique maximal  $Q_i$ . A passagem de  $Q_{i-1}$  para  $Q_i$  é sempre um vértice não restrito e marcado de  $S_{i-1}$ . Como num *smv* nunca existem dois vértices restritos, a passagem está garantida. Na saída da clique temos que considerar dois casos:

- a clique não tem vértices simpliciais.

Caso 1: se só existem vértices marcados, isso significa que o separador  $S_i$  só tinha um vértice disponível e a entrada na clique foi feita por  $v \in S_{i-1} \cap S_i$ . A saída é feita pelo mesmo vértice da entrada.

Caso 2: se existem vértices de  $S_i$  não marcados, eles são incluídos e a saída é por um deles.

- a clique tem vértices simpliciais.

Neste caso o caminho precisa passar pelos simpliciais antes de decidir o vértice de saída e recair no caso 2 anterior. A saída nunca é pelo mesmo vértice da entrada, pois, como existem simpliciais na clique, não existe  $v \in S_{i-1} \cap S_i$ .

O caminho é sempre construído. O ciclo hamiltoniano é fechado com a inclusão dos vértices restritos  $r_i$ . ■

Podemos ver que o Algoritmo 4.2 tem complexidade de tempo de  $O(n + m)$  quando aplicado a um grafo de intervalo biconexo. A inicialização de *marca* tem complexidade de  $O(n)$ . No primeiro percurso, o algoritmo explora a sequência de cliques maximais e, em cada uma delas, os seus vértices, obedecendo à sequência canônica. Esse percurso, que separa os vértices restritos, tem complexidade  $O(n + m)$ . No segundo, os vértices restritos são processados. Como temos, no máximo,  $(q - 1)$  vértices restritos, o segundo percurso tem complexidade no pior caso de  $O(n)$ . Logo, o algoritmo tem complexidade de tempo de  $O(n + m)$ .

## 4.4 Uma subclasse: grafos split-indiferença

Já observamos que, ao introduzir uma subclasse, existem alguns problemas básicos a serem tratados. Assim, sempre mencionamos a caracterização e o reconhecimento eficiente. Nessa seção, vamos tratar de outro problema, pouco mencionado e bastante importante: a contagem dos elementos da classe a menos de isomorfismo. Para isso, usaremos como exemplo a classe dos grafos split-indiferença.

Primeiramente, vamos nos concentrar na caracterização da classe. Um grafo é um *grafo split* se seu conjunto de vértices puder ser particionado em uma clique e um conjunto independente (não há pares de vértices adjacentes). A clique maximal que contém essa clique é chamada *clique base*. Note que vértices simpliciais podem pertencer à clique base, e um deles poderia pertencer ao conjunto independente. Os *grafos split-indiferença* são, simultaneamente, grafos split e grafos de intervalo próprio. A Figura 4.4 mostra três exemplos de grafos split. No grafo  $G$ , a clique base é  $\{a, c, d, e, f\}$ ;  $G$  é split, é de intervalo, porém não é split-indiferença. Nos grafos  $G'$  e  $G''$  a clique base é  $\{a, c, e, f\}$  e ambos são split-indiferença.

Uma caracterização para essa classe é a seguinte.

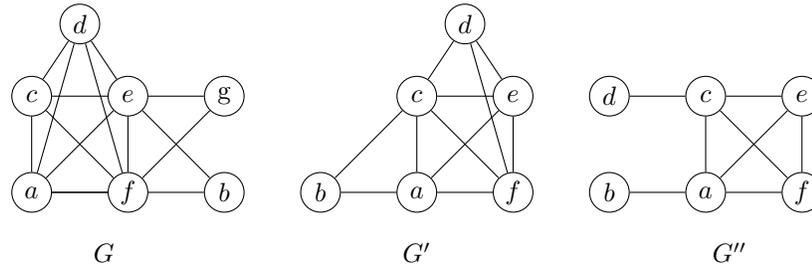


Figura 4.4: Exemplos de grafos split

**Teorema 4.6.** *Seja  $G$  um grafo conexo.  $G$  é um grafo split-indiferença se e somente se*

1.  $G$  é um grafo completo, ou
2.  $G$  possui duas cliques maximais  $Q_1$  e  $Q_2$ , tais que  $|Q_1 \setminus Q_2| = 1$  ou
3.  $G$  possui três cliques maximais  $Q_1$ ,  $Q_2$  e  $Q_3$ , sendo  $|Q_1 \setminus Q_2| = |Q_3 \setminus Q_2| = 1$ , tais que
  - (a)  $Q_1 \cap Q_3 = \emptyset$  ou
  - (b)  $Q_1 \cup Q_3 = V$ .

Demonstração:

( $\rightarrow$ ) Vamos provar por construção, considerando o número de cliques maximais. Observe que, num grafo split, por definição, todas as cliques maximais exceto a clique base possuem apenas um vértice simplicial.

- $G$  possui uma clique maximal:  
 $G$  é completo e, por definição, é split e de intervalo próprio (caso 1).
- $G$  possui duas cliques maximais:  
Qualquer grafo com duas cliques maximais é de intervalo próprio. Como é também grafo split, é necessário que pelo menos uma das duas cliques maximais tenha somente um vértice simplicial. Então, sem perda de generalidade,  $|Q_1 \setminus Q_2| = 1$  (caso 2).
- $G$  possui três cliques maximais:  
Suponha a árvore de cliques de  $G$ , a sequência  $\langle Q_1, Q_2, Q_3 \rangle$ , sendo  $Q_2$  a clique base. Como  $G$  é split, possui duas cliques maximais com um único simplicial. Então  $|Q_1 \setminus Q_2| = |Q_3 \setminus Q_2| = 1$ . Como também é de intervalo próprio,  $G$  é livre de garra (Teorema 4.1). Se  $Q_1 \cap Q_3 = \emptyset$ , não existe vértice pertencente a  $Q_1 \cap Q_2 \cap Q_3$  e o teorema é satisfeito. Se  $Q_1 \cap Q_3 \neq \emptyset$ , não podemos ter vértice simplicial em  $Q_2$ , o que formaria uma garra. Então  $Q_1 \cup Q_3$  precisa incluir todos os vértices de  $Q_2$ , sendo  $Q_1 \cup Q_3 = V$  (caso 3).
- $G$  possui quatro ou mais cliques maximais:  
Suponha a árvore de cliques de  $G$ , um caminho com as quatro cliques maximais,  $\langle Q_1, Q_2, Q_3, Q_4 \rangle$ , sendo  $Q_2$  a clique base. Pela definição de split,

$Q_2 \cap Q_4 \neq \emptyset$  e considere  $Q_2 \cap Q_4 = S$ . Pela propriedade da interseção de cliques da árvore de cliques, sabemos que  $S \subseteq Q_2 \cap Q_3$ , o que contraria o fato de  $G$  ser unicamente representável e, por conseguinte, de intervalo próprio. Sendo assim, o grafo  $G$  tem, no máximo, três cliques maximais.

( $\leftarrow$ ) O caso 1 é trivial. No caso 2, qualquer grafo com duas cliques é de intervalo próprio. Como  $|Q_1 \setminus Q_2| = 1$ , o vértice  $v \in (Q_1 \setminus Q_2)$  e algum vértice simplicial de  $Q_2$  formam um conjunto independente; os vértices restantes formam uma clique (que está contida na clique base). Logo, o grafo é split-indiferença.

No caso 3, se  $Q_1 \cap Q_3 = \emptyset$ , o raciocínio é análogo ao do caso anterior. Seja  $Q_1 \cup Q_3 = V$ . Como  $|Q_1 \setminus Q_2| = |Q_3 \setminus Q_2| = 1$ , os vértices de  $Q_1$  e  $Q_3$  que são simpliciais formam em  $G$  um conjunto independente. A clique  $Q_2$  está contida em  $Q_1$  e em  $Q_3$ ; o grafo é split. O grafo  $G$  não seria grafo de intervalo próprio se contivesse uma garra. Como a clique  $Q_2$  não tem vértices simpliciais, o grafo é de intervalo próprio. ■

A Figura 4.5 mostra uma representação esquemática acompanhada de exemplo para os casos apresentados no Teorema 4.6.

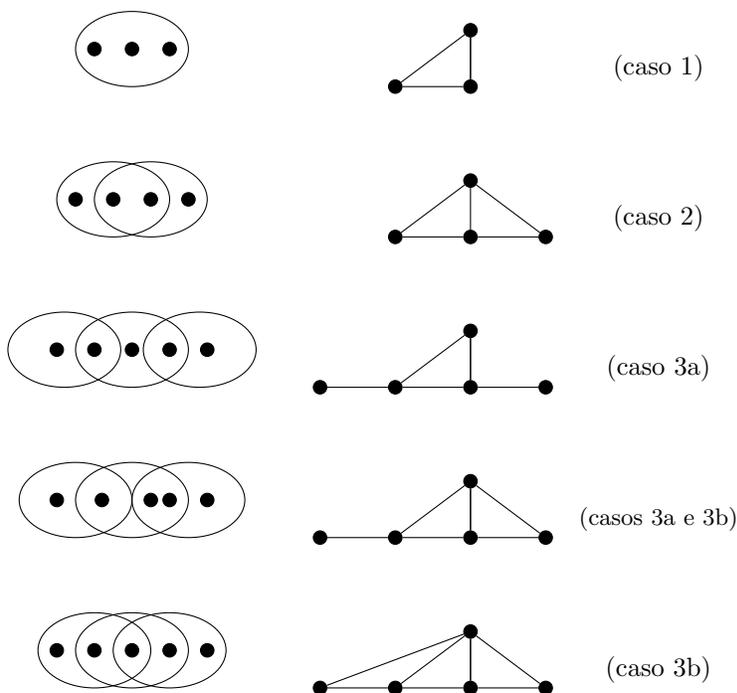


Figura 4.5: Casos do Teorema 4.6

O Teorema 4.6 nos fornece um algoritmo de reconhecimento imediato. A abordagem, nesse caso, é verificar diretamente se um grafo cordal é ou não split-indiferença. Para tal, é suficiente analisar suas cliques maximais.

O algoritmo a seguir determina se um grafo cordal  $G$  é um grafo split-indiferença ou não.

**Passo 1:** Construir uma árvore de cliques  $T$  de  $G$ ;

Se  $T$  é um caminho de cliques  $\langle Q_1, \dots, Q_q \rangle$ ,  $q \leq 3$ , então ir para o Passo 2 senão  $G$  não é um grafo split-indiferença;

**Passo 2:** Determinar o conjunto  $Simp_i$ ,  $1 \leq i \leq q$ , de simpliciais;

**Passo 3:** Testar os casos do Teorema 4.6.

O algoritmo para o reconhecimento de um grafo split-indiferença tem complexidade de tempo de  $O(n+m)$ . O Passo 1 utiliza o Algoritmo 2.3 e tem complexidade linear. O Passo 2 percorre as cliques maximais e computa, para cada vértice  $v \in V$ , o número de vezes que  $v$  aparece em uma clique maximal. Os vértices que estão em apenas uma clique são simpliciais (Teorema 2.2). O Passo 3 analisa a estrutura das cliques maximais, e também tem complexidade de tempo de  $O(n+m)$ .

#### 4.4.1 A contagem de grafos split-indiferença

Um problema importante quando da definição de uma nova classe de grafos, o *problema da contagem* consiste em determinar a quantidade total de grafos distintos, pertencentes à classe em questão, que podem ser construídos utilizando um conjunto de vértices com  $n$  elementos.

O problema de contagem do número de grafos rotulados é bastante estudado na literatura. Entretanto, o problema tem solução bem mais complexa ao tratarmos os grafos não rotulados, uma vez que nesse caso o isomorfismo dos elementos precisa ser considerado. Veja o exemplo que se segue. A Figura 4.6 ilustra a contagem de árvores que possuem  $V = \{a, b, c, d\}$  como conjunto de vértices. Se o isomorfismo não é levado em conta, existem 16 árvores distintas que se podem construir com os vértices dados; entretanto, há apenas duas árvores distintas não isomorfas com quatro vértices.

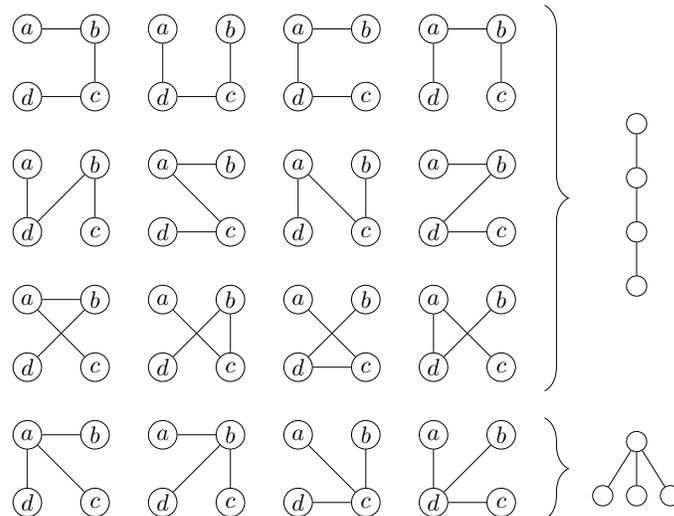


Figura 4.6: Árvores distintas, rotuladas e não rotuladas

A contagem segue os casos mencionados no Teorema 4.6. Seja um grafo split-indiferença com  $n$  vértices. O número de grafos com  $i$  cliques,  $1 \leq i \leq 3$ , é denotado  $\mathcal{N}_i$ .

1.  $G$  é um grafo completo:  $\mathcal{N}_1 = 1$ .
2.  $G$  possui duas cliques maximais  $Q_1$  e  $Q_2$  tais que  $|Q_1 \setminus Q_2| = 1$  e  $S = Q_1 \cap Q_2$ . Nesse caso, o *smv*  $S$  pode ter cardinalidade variando de 1 a  $n - 2$ , uma vez que  $Q_1$  possui um vértice simplicial e  $Q_2$  precisa ter, no mínimo, um vértice simplicial. Então  $\mathcal{N}_2 = n - 2$ .
3.  $G$  possui três cliques maximais  $Q_1, Q_2$  e  $Q_3$  tais que  $|Q_1 \setminus Q_2| = |Q_3 \setminus Q_2| = 1$ ,  $S_1 = Q_1 \cap Q_2$  e  $S_2 = Q_2 \cap Q_3$ . Vamos considerar que a clique base  $Q_2$  tem  $c = n - 2$  elementos; os separadores minimais de vértices têm cardinalidade  $|S_1| = k$  e  $|S_2| = \ell$ ,  $k \geq \ell$ .

**Caso  $n$  par.**

- (a)  $Q_1 \cap Q_3 = \emptyset$ .

Precisamos considerar todos os casos para as cardinalidades de  $S_1$  e  $S_2$ , lembrando que  $k \geq \ell$ . Então, para  $k = 1$  temos apenas um grafo, com  $\ell = 1$ , para  $k = 2$  temos duas possibilidades, com  $\ell = 1, 2$ , e assim sucessivamente até  $k = c/2$ , com  $\ell = 1, 2, \dots, c/2$ , e o número de grafos distintos aumentando. Quando  $k = c/2 + 1$ , o número de grafos distintos sofre uma reversão uma vez que só podemos ter  $\ell = 1, 2, \dots, c/2 - 1$ , pois  $S_1 \cap S_2 = \emptyset$ . Nesse momento, o número de grafos é  $c/2 - 1$ , decrescendo sucessivamente até  $k = c - 1$  e  $\ell = 1$ . A Tabela 4.1 apresenta a distribuição.

$k$	$\ell$	número de grafos
1	1	1
2	1	2
	2	
3	1	3
	2	
	3	
...	...	...
$c/2$	1	$c/2$
	...	
$c/2 + 1$	...	$c/2 - 1$
	$c/2 - 1$	
...	...	...
$c - 1$	1	1
		total: $c^2/4$

Tabela 4.1: Caso  $n$  par (a)

O número de grafos até aqui é:

$$\sum_{i=1}^{c/2} i + \sum_{i=1}^{c/2-1} i = \frac{c^2}{4}.$$

(b)  $Q_1 \cup Q_3 = V$ .

$k$	$\ell$	número de grafos
$c/2 + 1$	$c/2$ $c/2 + 1$	2
$c/2 + 2$	$c/2 - 1$ $\dots$ $c/2 + 2$	4
$\dots$	$\dots$	$\dots$
$c - 1$	2 3 $\dots$ $c - 1$	$c - 2$
		total: $c^2/4 - c/2$

Tabela 4.2: Caso  $n$  par (b)

Como  $k \geq \ell$ , precisamos apenas considerar  $k \geq c/2 + 1$ , uma vez que a clique maximal  $Q_2$  não tem vértices simpliciais. Quando  $k = c/2 + 1$ , temos somente  $\ell = c/2$  ou  $\ell = c/2 + 1$  (dois grafos distintos), para que a interseção aconteça. Quando  $k = c/2 + 2$ , temos  $\ell = c/2 - 1, c/2, c/2 + 1, c/2 + 2$  (quatro grafos distintos), e assim sucessivamente até  $k = c - 1$ , quando temos  $\ell = 2, \dots, c - 1$  ( $c - 2$  grafos). A Tabela 4.2 mostra a distribuição.

Para esse caso, o número de grafos é

$$\sum_{i=1}^{c/2-1} 2i = \frac{c^2 - 2c}{4}.$$

Então o número total de grafos para o caso par é

$$\mathcal{N}_3 = \frac{c^2}{4} + \frac{c^2 - 2c}{4} = \frac{c^2 - c}{2}.$$

### Caso $n$ ímpar.

Para  $n$  ímpar, temos pequenas modificações no caso de existirem três cliques no grafo. Precisamos entretanto considerar, como no caso  $n$  par,  $Q_1 \cap Q_3 = \emptyset$  e  $Q_1 \cup Q_3 = V$ . As Tabelas 4.3 e 4.4 ilustram a situação.

(a)  $Q_1 \cap Q_3 = \emptyset$  (Tabela 4.3).

O número de grafos até aqui é:

$$2 \left( \sum_{i=1}^{(c-1)/2} i \right) = \frac{c^2 - 1}{4}.$$

(b)  $Q_1 \cap Q_3 = V$  (Tabela 4.4).

Para esse caso, o número de grafos é:

$$\sum_{i=1}^{(c-1)/2} (2i - 1) = \sum_{i=1}^{(c-1)/2} 2i + \frac{c-1}{2} = \frac{c^2 - 2c + 1}{4}.$$

$k$	$\ell$	número de grafos
1	1	1
2	1 2	2
3	1 2 3	3
...	...	...
$(c-1)/2$	1 ... $(c-1)/2$	$(c-1)/2$
$(c+1)/2$	1 ... $(c-1)/2$	$(c-1)/2$
...	...	...
$c-1$	1	1
		total: $(c^2-1)/4$

Tabela 4.3: Caso  $n$  ímpar (a)

$k$	$\ell$	número de grafos
$(c+1)/2$	$(c+1)/2$	1
$(c+3)/2$	$(c-1)/2$ $(c+1)/2$ $(c+3)/2$	3
...	...	...
$c-1$	2 3 ... $c-1$	$c-2$
		total: $(c^2-2c+1)/4$

Tabela 4.4: Caso  $n$  ímpar (b)

Então o número total de grafos para o caso ímpar é

$$\mathcal{N}'_3 = \frac{c^2-1}{4} + \frac{c^2-2c+1}{4} = \frac{c^2-c}{2}.$$

O resultado é o mesmo, independente do fato de  $n$  ser par ou ímpar. Então, lembrando que  $c = n - 2$ , o número total de grafos split-indiferença com  $n$  vértices é dado pela fórmula:

$$\mathcal{N}_1 + \mathcal{N}_2 + \mathcal{N}_3 = 1 + n - 2 + \frac{c^2-c}{2} = \frac{n^2-3n+4}{2}.$$

Como exemplo, a Figura 4.7 exhibe todos os grafos split-indiferença com  $n = 7$ .

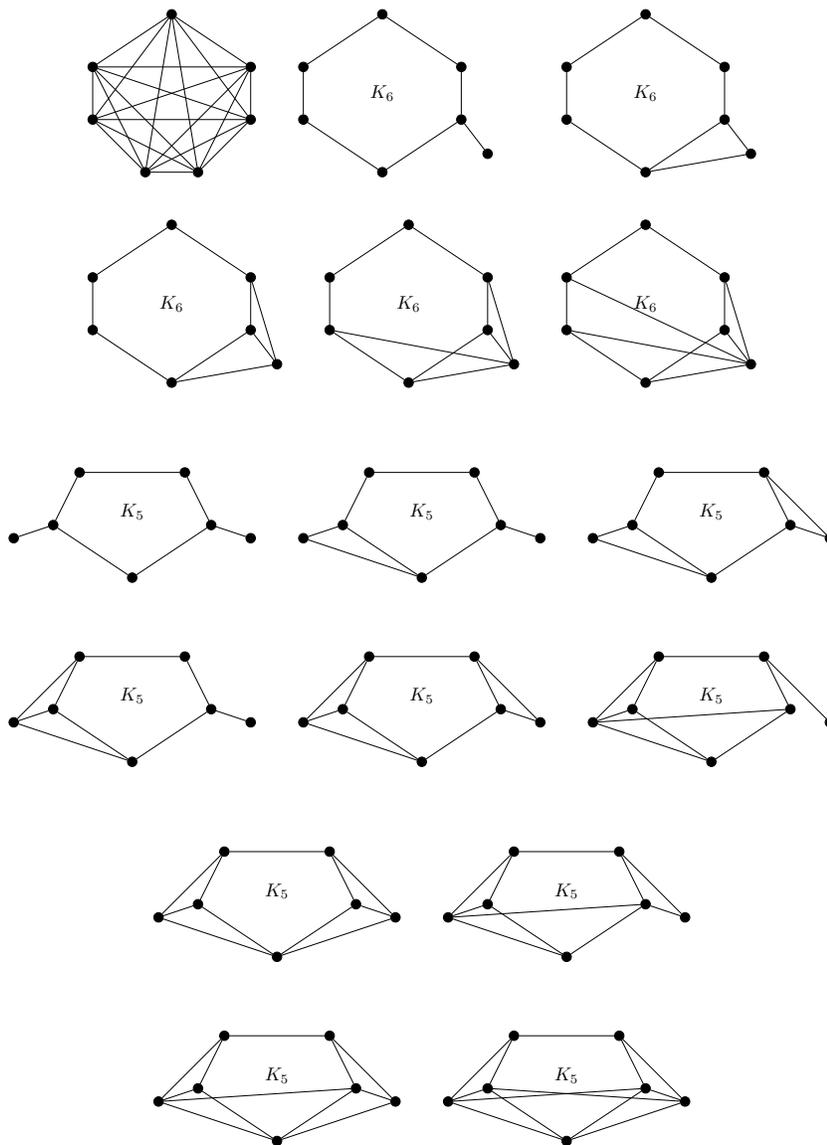


Figura 4.7: Grafos split-indiferença,  $n = 7$

## 4.5 Exercícios

1. Apresente um caminho de cliques para o grafo  $G$  da Figura 4.2.
2. Demonstre que a classe dos *caterpillar* é subclasse própria dos grafos de intervalo. Em outras palavras: todo caterpillar é um grafo de intervalo, mas a recíproca não é verdadeira.
3. Acompanhe o Algoritmo 4.1 quando este é aplicado ao grafo  $G'$  da Figura 4.2.
4. Dado um grafo  $G = (V, E)$  e um inteiro positivo  $k$ , a  $k$ -ésima potência de  $G$  é o grafo  $G^k = (V, E')$  no qual dois vértices são adjacentes quando a distância entre eles em  $G$  é no máximo  $k$ .

A  $k$ -ésima potência de um grafo caminho é denotada  $P_n^k$ ,  $1 \leq k < n$ .

- (a) Construa o grafo  $P_9^2$ .
- (b) Apresente um caminho de cliques para o grafo  $P_9^2$ .
- (c) Demonstre que  $P_n^k$  é um grafo de intervalo próprio.
- (d) Apresente uma construção indutiva para  $P_n^k$ , baseada na cardinalidade de suas cliques maximais e de seus separadores minimais de vértices.

5. Apresente todos os grafos split-indiferença com  $n = 8$  vértices.

6. Um *grafo colar* pode ser indutivamente definido da seguinte maneira:

- Um grafo caminho com  $n_1$  vértices,  $n_1 \geq 2$ , é um grafo colar.
- Se  $G = (V, E)$  é um grafo colar,  $Q \subseteq V$  é uma clique maximal de  $G$  e  $v \notin V$ , então o grafo  $G' = (V \cup \{v\}, E \cup \{\{v, w\} : w \in Q\})$  é um grafo colar.
- Nada mais é um grafo colar.

Pede-se:

- (a) Dê um exemplo de grafo colar que não seja um caminho.
- (b) Seja um grafo colar  $G = (V, E)$  com  $n$  vértices e  $q$  cliques maximais. Quantos vértices simpliciais e quantos separadores minimais de vértices distintos tem  $G$ ?
- (c) Prove que um grafo colar é um grafo de intervalo próprio.
- (d) Mostre que a recíproca do item anterior não é verdadeira.
- (e) Apresente um esquema de eliminação bicompatível para o seu exemplo do item (a).
- (f) Apresente um esquema de eliminação perfeita que não seja bicompatível.
- (g) Mostre que um grafo colar biconexo tem um ciclo hamiltoniano.

7. Caracterize os grafos split-indiferença que possuem um ciclo hamiltoniano.

## 4.6 Notas bibliográficas

Por suas características estruturais, a classe dos grafos de intervalo próprio vem sendo muito estudada na literatura. São encontradas diversas caracterizações e denominações, provadas equivalentes, por Roberts [64, 65], Bogart e West [6] e Panda e Das [63]. Podem ser citados alguns algoritmos de reconhecimento: Figueiredo *et al.* [23], Panda e Das [63], Corneil [17] e Li e Wu [46]. A caracterização e o reconhecimento utilizados nesse capítulo foram propostos por Looges e Olariu [49]; a revisão do algoritmo de reconhecimento foi apresentada por Markenzon e Waga [57].

A solução para o problema da determinação do ciclo hamiltoniano se baseia em [39].

Os grafos split-indiferença foram definidos e caracterizados por Ortiz *et al.* [62]; problemas de coloração e a determinação do corte máximo (subgrafo bipartido máximo) foram estudados para a classe [14, 5]. A Figura 4.6 foi exibida em [54].



## Capítulo 5

# Grafos Estritamente Cordais

Os grafos estritamente cordais constituem uma classe com significativas propriedades estruturais. Nesse capítulo apresentamos caracterizações, também baseadas nas ferramentas estruturais do Capítulo 2. Como resultado imediato, um algoritmo de reconhecimento bastante simples pode ser exibido. Em particular, o grafo de cliques críticas tem papel relevante para essa classe, e são estabelecidas propriedades específicas e um algoritmo linear para a construção da estrutura.

Como exemplo de solução de problemas para a classe, abordamos um tópico clássico em teoria de grafos, o da dominação de vértices. Tratamos também de duas variantes desse problema. Resolvemos a dominação conexa, e a determinação dos grafos estritamente cordais bem dominados.

Ainda com base nas ferramentas, grafos *AC*, grafos potência de 3-folha e grafos estritamente de intervalo são, por sua vez, incluídos na hierarquia como subclasses dos grafos estritamente cordais.

### 5.1 A classe

Os *grafos estritamente cordais*, também denominados *grafos bloco duplicados*, são aqueles obtidos pela adição de zero ou mais gêmeos verdadeiros a cada vértice de um grafo bloco.

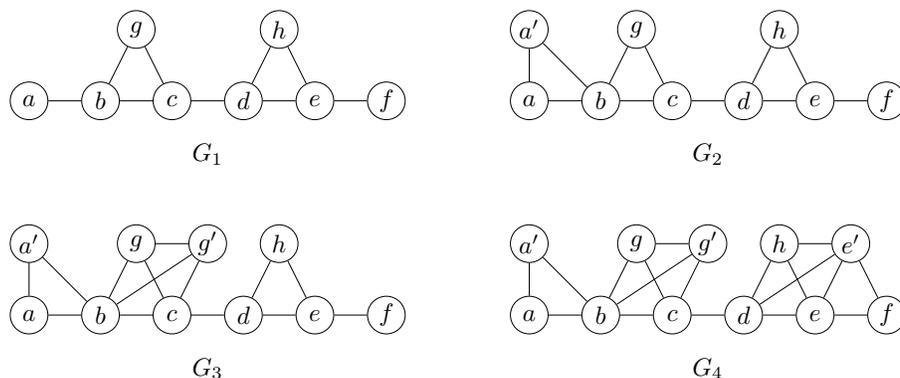


Figura 5.1: Grafos estritamente cordais

Na Figura 5.1 ilustramos esta construção. O grafo  $G_1$  é um grafo bloco e, a partir dele, incluiu-se um gêmeo verdadeiro do vértice  $a$ ,  $a'$ , obtendo-se o grafo  $G_2$ . Adicionamos então um gêmeo verdadeiro do vértice  $g$ ,  $g'$ , e obtemos  $G_3$ . A partir de  $G_3$ , acrescentamos um gêmeo verdadeiro do vértice  $e$ ,  $e'$ , e, finalmente, obtemos o grafo  $G_4$ . Os quatro grafos são exemplos de grafos estritamente cordais.

As caracterizações da classe, uma utilizando propriedades dos separadores minimais de vértices e a outra por subgrafos proibidos, podem ser reunidas no seguinte teorema.

**Teorema 5.1.** *Seja  $G = (V, E)$  um grafo cordal. As seguintes afirmativas são equivalentes:*

1.  $G$  é um grafo estritamente cordal.
2. Para quaisquer separadores minimais de vértices distintos  $S, S' \in \mathbb{S}$ , tem-se que  $S \cap S' = \emptyset$ .
3.  $G$  é um grafo livre de gema e de seta.

Demonstração:

(1  $\rightarrow$  2) Seja  $G$  um grafo estritamente cordal obtido a partir de um grafo bloco  $G'$ . Todos os separadores minimais de vértices de um grafo bloco têm cardinalidade 1, pelo Teorema 3.12. Pela definição de grafo estritamente cordal, somente gêmeos verdadeiros podem ser incluídos em  $G'$ . Se o novo vértice  $v$  é gêmeo de um vértice simplicial  $w$ , a clique maximal que contém  $w$  é aumentada e  $v$  é também um vértice simplicial de  $G$ . Caso contrário,  $w$  pertence a um separador minimal de vértices  $S$  e tanto  $S$  quanto as cliques maximais as quais  $w$  pertence são aumentadas. Desta forma, a interseção de dois separadores minimais de vértices distintos de  $G$  é vazia.

(2  $\rightarrow$  3) O grafo gema e o grafo seta têm três cliques maximais e dois separadores minimais de vértices  $S$  e  $S'$ . Como em ambos os grafos  $S \cap S' \neq \emptyset$ ,  $G$  é livre de gema e de seta.

(3  $\rightarrow$  1) Considere  $G$  um grafo cordal livre de gema e de seta. Como o fato de ser livre de gema e de seta é uma propriedade hereditária para remoção de vértices, podemos assumir que  $G$  não possui gêmeos verdadeiros e mostrar que, nesse caso,  $G$  é um grafo bloco. Vamos supor que  $G$  não é um grafo bloco. O grafo  $G$  é cordal então  $G$  contém um diamante, pelo Teorema 3.12. Sejam  $a$  e  $b$  os vértices de grau 3 e  $c$  e  $d$  os de grau 2 do diamante. Como os vértices  $a$  e  $b$  não podem ser gêmeos verdadeiros,  $G$  tem outro vértice  $v$  adjacente ou a  $a$  ou a  $b$  mas não a ambos. Vamos considerar que  $\{a, v\} \in E$ . Se  $v$  não é adjacente nem a  $c$  nem a  $d$ ,  $G$  contém um grafo seta. Se  $v$  é adjacente a somente um deles,  $G$  contém um grafo gema. Se  $v$  é adjacente a ambos,  $v$ ,  $c$ ,  $d$  e  $b$  induzem um ciclo de comprimento 4 em  $G$ . Assim,  $G$  não é cordal. Contradição. Logo,  $G$  é um grafo estritamente cordal. ■

O grafo da Figura 5.2 possui nove cliques maximais e seis separadores minimais de vértices  $S_1 = \{b, h\}$ ,  $S_2 = \{c, i\}$ ,  $S_3 = \{r, s\}$ ,  $S_4 = \{o\}$ ,  $S_5 = \{d, j\}$  e  $S_6 = \{e, k\}$ . Podemos, usando o segundo item do Teorema 5.1, verificar de maneira direta que o grafo é estritamente cordal pois  $S_i \cap S_j = \emptyset$ ,  $i, j = 1, \dots, 6$ ,  $i \neq j$ .

Retomando um resultado apresentado no Capítulo 3, Seção 3.3, é fácil verificar que os grafos bloco, as árvores, os grafos caterpillar e os grafos caminho são subclasses dos grafos estritamente cordais.

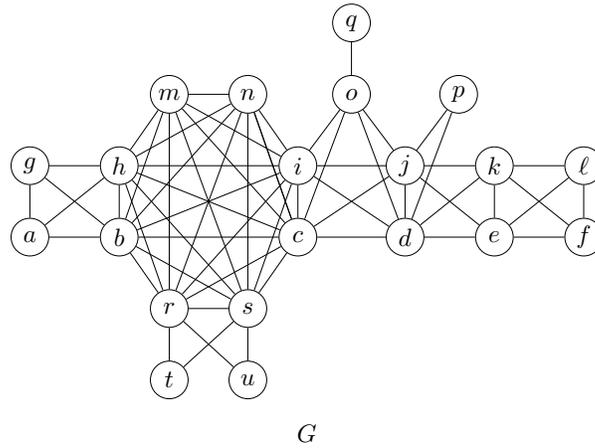


Figura 5.2: Grafo estritamente cordal

## 5.2 Reconhecimento

Baseado no Teorema 5.1, um algoritmo de reconhecimento pode ser apresentado. Considere um grafo cordal  $G = (V, E)$  e seu conjunto  $\mathbb{S}$  de separadores minimais de vértices; o algoritmo deve reconhecer se há interseção dos mesmos. É estabelecido um rótulo  $D(v)$  para cada um dos vértices do grafo, inicializado com zeros. Na  $i$ -ésima iteração, o  $smv$   $S_i$  é considerado e os rótulos de seus vértices recebem o índice  $i$ . Caso o rótulo seja anteriormente diferente de zero, isso significa que o vértice já apareceu em outro separador e o grafo não é estritamente cordal.

---

### Algoritmo 5.1. Reconhecer-Estritamente-Cordal

---

**Entrada:** grafo  $G$  e  $\mathbb{S} = \{S_1, \dots, S_k\}$ ;  
**Saída:**  $G$  é estritamente cordal ou não;  
**Início**  
*EstritamenteCordal*  $\leftarrow$  *true*;  
**Para**  $i = 1, \dots, n$  **faça**  $D(i) \leftarrow 0$ ;  
**Para**  $i = 1, \dots, k$  **faça** % analisar separadores minimais  
**Para**  $v \in S_i$  **faça**  
**Se**  $D(v) \neq 0$  **então**  
*EstritamenteCordal*  $\leftarrow$  *false*;  
**senão**  $D(v) \leftarrow i$ ;  
**Fim.**

---

Mais uma vez vamos nos basear no Teorema 2.3 para justificar a complexidade tempo de  $O(n + m)$  do Algoritmo 5.1. O algoritmo inicializa  $D$  e depois percorre os separadores minimais de vértices que, como são subconjuntos das cliques maximais, o que tem complexidade de pior caso de  $O(n + m)$ .

Para o grafo da Figura 5.2, mostramos o acompanhamento do algoritmo na tabela a seguir. A variável booleana *EstritamenteCordal* permanece *true* até o final da execução.

$i$	$S_i$	$D$																				
		$a$	$b$	$c$	$d$	$e$	$f$	$g$	$h$	$i$	$j$	$k$	$\ell$	$m$	$n$	$o$	$p$	$q$	$r$	$s$	$t$	$u$
1	$\{b, h\}$	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
2	$\{c, i\}$	0	1	2	0	0	0	0	1	2	0	0	0	0	0	0	0	0	0	0	0	0
3	$\{r, s\}$	0	1	2	0	0	0	0	1	2	0	0	0	0	0	0	0	0	3	3	0	0
4	$\{o\}$	0	1	2	0	0	0	0	1	2	0	0	0	0	0	4	0	0	3	3	0	0
5	$\{d, j\}$	0	1	2	5	0	0	0	1	2	5	0	0	0	0	4	0	0	3	3	0	0
6	$\{e, k\}$	0	1	2	5	6	0	0	1	2	5	6	0	0	0	4	0	0	3	3	0	0

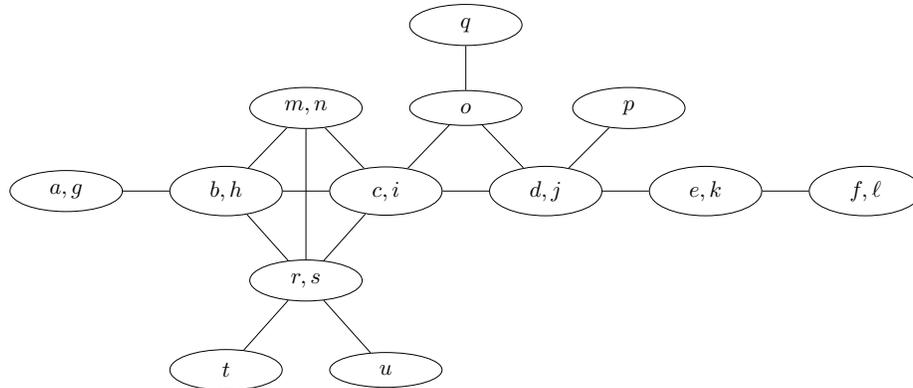
### 5.3 $C_c(G)$ de grafos estritamente cordais

O grafo de cliques críticas é a ferramenta estrutural introduzida no Capítulo 2, Seção 2.5, especialmente adequada para os grafos estritamente cordais. Nessa seção veremos algumas propriedades e um algoritmo para sua construção.

#### 5.3.1 Propriedades

Relembrando a definição, uma clique  $C$  de  $G$  é uma *clique crítica* quando todos os seus vértices são gêmeos verdadeiros e é maximal para a propriedade. O *grafo de cliques críticas*  $C_c(G)$  é o grafo cujo conjunto de vértices são as cliques críticas e tal que duas cliques críticas  $C$  e  $C'$  são adjacentes sempre que um vértice de  $C$  for adjacente em  $G$  a algum vértice em  $C'$ .

O grafo estritamente cordal  $G$  da Figura 5.2 possui treze cliques críticas  $C_1 = \{a, g\}$ ,  $C_2 = \{b, h\}$ ,  $C_3 = \{m, n\}$ ,  $C_4 = \{r, s\}$ ,  $C_5 = \{t\}$ ,  $C_6 = \{u\}$ ,  $C_7 = \{c, i\}$ ,  $C_8 = \{d, j\}$ ,  $C_9 = \{o\}$ ,  $C_{10} = \{q\}$ ,  $C_{11} = \{p\}$ ,  $C_{12} = \{e, k\}$ ,  $C_{13} = \{f, \ell\}$ . Na Figura 5.3, temos o grafo de cliques críticas  $C_c(G)$ .



$C_c(G)$

Figura 5.3: Grafo de cliques críticas

Uma propriedade das cliques críticas dos grafos da classe é vista a seguir.

**Teorema 5.2.** *Seja  $G = (V, E)$  um grafo estritamente cordal não completo. Uma clique crítica qualquer de  $G$  é formada por:*

1. *vértices simpliciais de uma clique maximal de  $G$  (se existirem);*
2. *vértices de um separador minimal de vértices de  $G$ .*

Demonstração:

Pela Propriedade 2.5, toda clique maximal é a união de cliques críticas. Um vértice  $v \in V$  de um grafo estritamente cordal ou é um vértice simplicial ou pertence a um único *smv*  $S$ . No primeiro caso, se existir,  $v$  e seus gêmeos verdadeiros pertencem à mesma clique crítica. No segundo caso, pelo fato de  $G$  ser estritamente cordal, todos os vértices pertencentes a  $S$  são gêmeos verdadeiros pois eles separam as mesmas cliques maximais. ■

Para os grafos estritamente cordais, o grafo de cliques críticas tem uma estrutura bem específica, possibilitando a caracterização a seguir.

**Teorema 5.3.** *Um grafo cordal  $G$  é estritamente cordal se e somente se  $C_c(G)$  é um grafo bloco.*

Demonstração:

( $\rightarrow$ ) Considere uma seqüência  $\langle Q_1, \dots, Q_q \rangle$  das cliques maximais de  $G$  obtida pelo Algoritmo 2.3. Vamos construir indutivamente o grafo  $C_c(G)$  a partir desta.

Quando  $q = 1$ ,  $G$  é um grafo completo e  $C_c(G)$  é um grafo com um único vértice e é um grafo bloco.

Supor que as  $(q - 1)$  cliques maximais da seqüência já foram consideradas, isto é,  $C_c(G')$  é um grafo bloco sendo  $G' = G [Q_1 \cup \dots \cup Q_{q-1}]$ . Vamos mostrar que após a inclusão das cliques críticas da  $q$ -ésima clique maximal, o grafo  $C_c(G) = C_c(G [Q_1 \cup \dots \cup Q_q])$  é um grafo bloco.

A clique maximal  $Q_q$  tem pelo menos um vértice simplicial (lembrando que o primeiro vértice de um *eep* é um vértice simplicial de  $G$ ). Pelo Teorema 5.2, existe um único vértice  $C$  em  $C_c(G')$  tal que  $C \cap Q_q \neq \emptyset$ .

Dois casos devem ser analisados:

1.  $C \cap Q_q$  é um vértice de  $C_c(G')$ . Então, um único vértice  $Q_q \setminus C$  deve ser incluído em  $C_c(G')$  e adjacente somente a  $C \cap Q_q$ .
2.  $C \cap Q_q$  está contido em um vértice de  $C_c(G')$ .
  - (a) O vértice  $Q_q \setminus C$  deve ser incluído.
  - (b) O vértice  $C$  deve ser particionado em dois outros,  $C \setminus (C \cap Q_q)$  e  $C \cap Q_q$ .

Com relação as adjacências:

- (a)  $Q_q \setminus C$  é adjacente unicamente a  $C \cap Q_q$ .
- (b) Os vértices adjacentes a  $C$  em  $C_c(G')$  são adjacentes tanto a  $C \setminus (C \cap Q_q)$  quanto a  $C \cap Q_q$ , formando uma nova clique maximal em  $C_c(G)$ .

Em ambos os casos,  $C_c(G)$  é um grafo bloco.

( $\leftarrow$ ) Se  $G$  não é um grafo estritamente cordal então ele possui o grafo gema  $G'$  e/ou o grafo seta  $G''$  como subgrafos induzidos. Ambos os grafos não são grafos

bloco e são tais que  $C_c(G') = G'$  e  $C_c(G'') = G''$ . ■

Vamos exemplificar a prova do Teorema 5.3 (Figura 5.4). Considere o grafo  $G$  da Figura 5.2 e a sequência  $\langle Q_1, \dots, Q_9 \rangle$ , sendo  $Q_1 = \{a, b, g, h\}$ ,  $Q_2 = \{b, c, h, i, m, n, r, s\}$ ,  $Q_3 = \{r, s, t\}$ ,  $Q_4 = \{r, s, u\}$ ,  $Q_5 = \{c, d, i, j, o\}$ ,  $Q_6 = \{o, q\}$ ,  $Q_7 = \{d, j, p\}$ ,  $Q_8 = \{d, e, j, k\}$  e  $Q_9 = \{e, f, k, \ell\}$ . Vejamos como seria a inclusão da clique maximal  $Q_5$  no grafo  $G' = G[Q_1 \cup \dots \cup Q_4]$ . A clique  $Q_5$  contém nesse momento duas cliques críticas  $C = \{c, i\}$  (clique crítica já existente em  $C_c(G')$ ) e  $C' = \{d, j, o\}$  (de vértices simpliciais). Exatamente o caso 1 da demonstração.

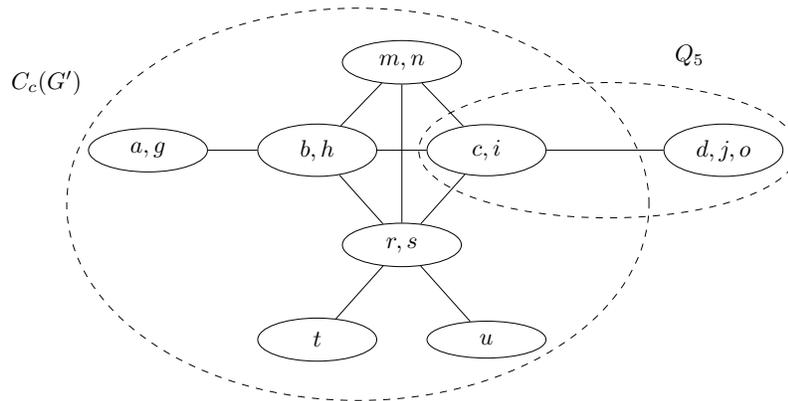


Figura 5.4: Obtendo  $C_c(G[Q_1 \cup \dots \cup Q_5])$

Vamos observar outra inclusão, a da clique  $Q_6$  em  $G' = G[Q_1 \cup \dots \cup Q_5]$ . O vértice  $\{q\}$  deve ser incluído, o vértice  $\{d, j, o\}$  deve ser particionado em dois,  $\{d, j\}$  e  $\{o\}$ , e as adjacências devem ser restauradas conforme o caso 2 da demonstração, observe a Figura 5.5.

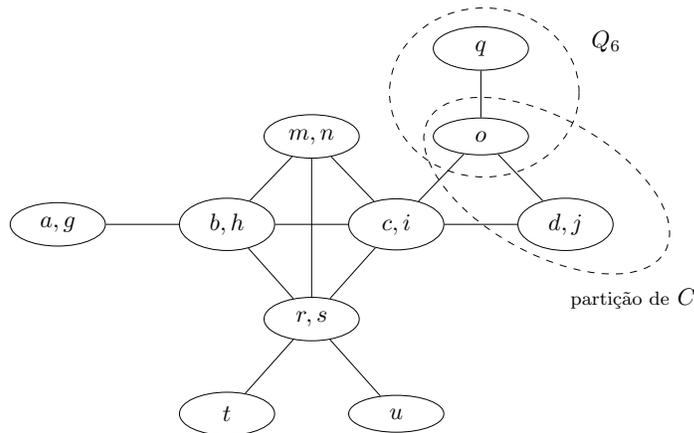


Figura 5.5: Obtendo  $C_c(G[Q_1 \cup \dots \cup Q_6])$

**Corolário 5.1.** *Seja  $G$  um grafo estritamente cordal. Toda clique maximal de  $C_c(G)$  tem no máximo um vértice simplicial.*

### 5.3.2 Algoritmo de construção

Como visto no Capítulo 2, Seção 2.5, um algoritmo para a construção de um grafo de cliques críticas para grafos em geral tem complexidade de tempo de  $O(n^2)$ . Baseado no Teorema 5.2, um algoritmo para determinação de  $C_c(G) = (V_c, E_c)$  de um grafo estritamente cordal  $G = (V, E)$  pode ser apresentado com complexidade de tempo linear. O algoritmo utiliza um rótulo  $clique(v)$  que indica a qual clique crítica o vértice  $v$  pertence. Primeiramente, as cliques críticas relativas aos separadores minimais de vértices de  $G$  são incluídas em  $V_c$ ; o valor de  $clique(v)$  é determinado para os vértices dessas cliques. Depois, cada clique maximal é percorrida em busca de vértices simpliciais (isto é, vértices que ainda têm o rótulo igual a 0). Se existem vértices simpliciais, então uma nova clique crítica é inicializada. Em seguida, as arestas do grafo são percorridas e as arestas de  $C_c(G)$  são estabelecidas. O algoritmo executa três percursos: o primeiro nos separadores minimais de vértices, o segundo nas cliques maximais e o terceiro nas arestas do grafo. Cada um desses percursos tem complexidade de pior caso de  $O(n + m)$ . Note que, ao fim do algoritmo, como cada vértice de um grafo estritamente cordal pertence apenas a uma única clique,  $clique(v)$  indica a clique crítica à qual o vértice  $v$  pertence.

**Algoritmo 5.2.** *Construir-grafo-cliques-críticas*

---

**Entrada:** grafo estritamente cordal  $G = (V, E)$ ,  
 $\mathbb{Q} = \{Q_1, \dots, Q_q\}$ ,  
 $\mathbb{S} = \{S_1, \dots, S_k\}$ ;  
**Saída:**  $C_c(G) = (V_c, E_c)$ ;  
**Início**  
 $V_c \leftarrow \emptyset$ ;  $E_c \leftarrow \emptyset$ ;  
**Para**  $i = 1, \dots, k$  **faça**                   % cliques críticas formadas por  $smv$   
     $C_i \leftarrow S_i$ ;  
     $V_c \leftarrow V_c \cup \{C_i\}$ ;  
    **Para**  $v \in S_i$  **faça**  $clique(v) \leftarrow i$ ;  
     $indice \leftarrow k$ ;  
    **Para**  $i = 1, \dots, q$  **faça**  
        determinar  $Simp$ , conjunto de vértices simpliciais de  $Q_i$ ;  
        **Se**  $Simp \neq \emptyset$  **então**               % cliques críticas formadas por simpliciais  
             $indice \leftarrow indice + 1$ ;  
             $C_{indice} \leftarrow Simp$ ;  
             $V_c \leftarrow V_c \cup \{C_{indice}\}$ ;  
            **Para**  $v \in Simp$  **faça**  $clique(v) \leftarrow indice$ ;  
    **Para**  $\{v, w\} \in E$  **faça**  
         $E_c \leftarrow E_c \cup \{\{C_{clique(v)}, C_{clique(w)}\}\}$ ;  
        remover arestas duplicadas em  $E_c$ ;  
**Fim.**

---

Vamos acompanhar o algoritmo para o grafo da Figura 5.2, observando as tabelas.

$i$	$C_i$	clique																				
		$a$	$b$	$c$	$d$	$e$	$f$	$g$	$h$	$i$	$j$	$k$	$\ell$	$m$	$n$	$o$	$p$	$q$	$r$	$s$	$t$	$u$
1	$\{b, h\}$	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
2	$\{c, i\}$	0	1	2	0	0	0	0	1	2	0	0	0	0	0	0	0	0	0	0	0	0
3	$\{r, s\}$	0	1	2	0	0	0	0	1	2	0	0	0	0	0	0	0	0	3	3	0	0
4	$\{o\}$	0	1	2	0	0	0	0	1	2	0	0	0	0	4	0	0	3	3	0	0	0
5	$\{d, j\}$	0	1	2	5	0	0	0	1	2	5	0	0	0	0	0	4	0	3	3	0	0
6	$\{e, k\}$	0	1	2	5	6	0	0	1	2	5	6	0	0	0	0	4	0	3	3	0	0

Ao final da primeira iteraç o, temos  $indice = 6$ ,  $V_c = \{C_1, \dots, C_6\}$  e ainda  $E_c = \emptyset$ . Os valores dos v rtices j  rotulados por *clique* n o aparecem na tabela a seguir pois esses n o ser o modificados.

$i$	$Q_i$		$indice$	$C_{indice}$	clique									
	$Simp$				$a$	$f$	$g$	$\ell$	$m$	$n$	$p$	$q$	$t$	$u$
1	$\{a, g\}$	$\{b, h\}$	7	$\{a, g\}$	7	0	7	0	0	0	0	0	0	0
2	$\{m, n\}$	$\{b, c, h, i, r, s\}$	8	$\{m, n\}$	7	0	7	0	8	8	0	0	0	0
3	$\{t\}$	$\{r, s\}$	9	$\{t\}$	7	0	7	0	8	8	0	0	9	0
4	$\{u\}$	$\{r, s\}$	10	$\{u\}$	7	0	7	0	8	8	0	0	9	10
5	$\emptyset$	$\{c, d, i, j, o\}$												
6	$\{q\}$	$\{o\}$	11	$\{q\}$	7	0	7	0	8	8	0	11	9	10
7	$\{p\}$	$\{d, j\}$	12	$\{p\}$	7	0	7	0	8	8	12	11	9	10
8	$\emptyset$	$\{d, e, j, k\}$												
9	$\{f, \ell\}$	$\{e, k\}$	13	$\{f, \ell\}$	7	13	7	13	8	8	12	11	9	10

Ap s a terceira iteraç o, obtemos como sa da o grafo de cliques cr ticas sendo

$$V_c = \{C_1, \dots, C_{13}\} \text{ e}$$

$$E_c = \{\{C_1, C_2\}, \{C_1, C_3\}, \{C_1, C_7\}, \{C_1, C_8\}, \\ \{C_2, C_3\}, \{C_2, C_4\}, \{C_2, C_5\}, \{C_2, C_8\}, \\ \{C_3, C_8\}, \{C_3, C_9\}, \{C_3, C_{10}\}, \{C_4, C_5\}, \\ \{C_4, C_{11}\}, \{C_5, C_6\}, \{C_5, C_{12}\}, \{C_6, C_{13}\}\}.$$

Como coment rio final podemos notar que o grafo de cliques cr ticas tende a ser um grafo bem mais simples do que o grafo original. Isto ocorre naturalmente pois  $C_c(G)$   , num certo sentido, uma compacta o do grafo  $G$  em que os excessos (os v rtices g meos verdadeiros) s o retirados.

## 5.4 Um problema: domina o

O grafo de cliques cr ticas pode tamb m ser utilizado para solucionar problemas no grafo original. Nosso exemplo   cl ssico na literatura, domina o dos v rtices, para o qual apresentamos duas varia es. Primeiramente, recordamos alguns conceitos conhecidos.

Um conjunto  $D \subseteq V$  de um grafo  $G = (V, E)$    um *conjunto dominante* se  $N[D] = V$  ou, equivalentemente, para todo  $v \in V \setminus D$ ,  $|N(v) \cap D| \geq 1$ . Um conjunto dominante  $D$    um *conjunto dominante minimal* quando nenhum de seus subconjuntos pr prios   um conjunto dominante.

**Teorema 5.4.** *Se  $G = (V, E)$  é conexo e  $D$  é um conjunto dominante minimal então  $V \setminus D$  é um conjunto dominante.*

O número de dominação  $\gamma(G)$  é a menor cardinalidade dentre todas as cardinalidades dos conjuntos dominantes do grafo  $G$  e o número de dominação superior  $\Gamma(G)$  é a maior cardinalidade dentre todas dos conjuntos dominantes minimais.

Observe que o conjunto formado por um elemento de cada um dos separadores minimais de vértices de um grafo cordal  $G$  o domina.

**Teorema 5.5.** *Se  $G$  é um grafo cordal então  $\gamma(G) \leq |\mathbb{S}|$ .*

Para grafos cordais, três resultados imediatos, porém importantes, estão enunciados a seguir.

**Propriedade 5.1.** *Qualquer conjunto dominante minimal de vértices não possui gêmeos verdadeiros.*

**Propriedade 5.2.** *Existe um conjunto dominante minimal de cardinalidade mínima de  $G$  composto somente por vértices pertencentes aos separadores minimais de vértices de  $G$ .*

**Propriedade 5.3.** *Toda clique simplicial de  $G$  tem um vértice em algum conjunto dominante minimal de  $G$ .*

Para grafos estritamente cordais, outras propriedades são obtidas.

**Propriedade 5.4.**  $\frac{|\mathbb{S}|}{2} \leq \gamma(G)$ .

Esta cota inferior é imediata, basta assumir que os separadores sejam dois a dois adjacentes, isto é, existe uma aresta com extremidades em cada um deles.

**Propriedade 5.5.** *Existe uma correspondência biunívoca entre os separadores minimais de vértices de  $G$  e de  $C_c(G)$ .*

Isso nos sugere que, ao resolver o problema de dominação para o grafo de cliques críticas, estaremos resolvendo também o problema para o grafo original.

**Teorema 5.6.** *Se  $G$  é um grafo estritamente cordal então  $\gamma(G) = \gamma(C_c(G))$ .*

Demonstração:

Seja  $\gamma(C_c(G)) = k$  e  $\mathcal{D} = \{C_1, \dots, C_k\}$  um de seus conjuntos dominantes minimais. Cada clique crítica  $C_i$  domina a si mesma e a qualquer clique crítica adjacente a ela. Considere o conjunto  $D = \{v_1, \dots, v_k\}$  tal que  $v_i \in C_i$ ,  $i = 1, \dots, k$ . Cada  $v_i$  domina exatamente os vértices da união de todas as cliques críticas dominadas por  $C_i$ , isto é,  $N[v_i] = \bigcup_{X \in N[C_i]} X$ . Como as cliques críticas determinam uma partição do conjunto de vértices de  $G$  (Propriedade 2.4), temos que  $D$  domina  $G$ . Além disso,  $D$  é um conjunto dominante minimal em  $G$  pois, caso contrário,  $\mathcal{D}$  não seria minimal em  $C_c(G)$ . ■

Podemos provar de forma análoga a igualdade para o número de dominação superior.

**Teorema 5.7.** *Se  $G$  é um grafo estritamente cordal então  $\Gamma(G) = \Gamma(C_c(G))$ .*

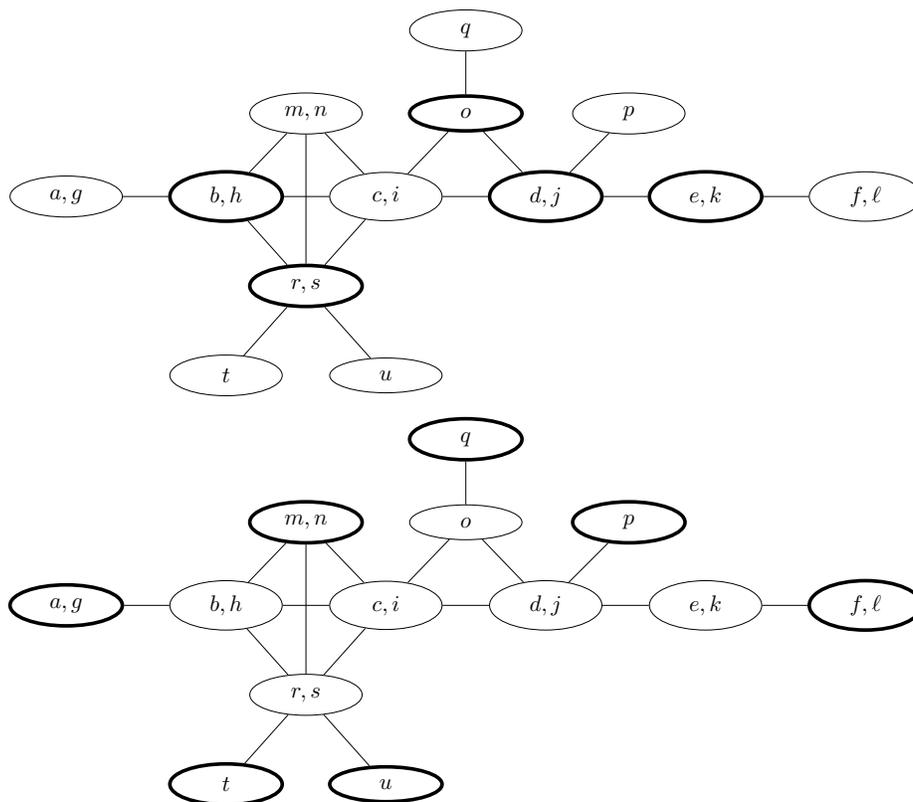


Figura 5.6: Conjuntos dominantes minimais em  $C_c(G)$

Conhecendo-se o número de dominação e o número de dominação superior do grafo  $C_c(G)$ , os resultados valem para os invariantes do grafo estritamente cordal original. Por exemplo, considere o grafo  $C_c(G)$  (Figura 5.3) do grafo  $G$  da Figura 5.2. O conjunto

$$\{\{b, h\}, \{r, s\}, \{d, j\}, \{e, k\}, \{o\}\}$$

é dominante minimal de menor cardinalidade e  $\gamma(C_c(G)) = \gamma(G) = 5$ . Já o conjunto

$$\{\{a, g\}, \{f, \ell\}, \{m, n\}, \{p\}, \{q\}, \{t\}, \{u\}\}.$$

dominante minimal de maior cardinalidade e  $\Gamma(C_c(G)) = \Gamma(G) = 7$  (ver Figura 5.6). Voltando ao grafo original  $G$ , escolhamos arbitrariamente um vértice de cada uma das cliques críticas que compõem um conjunto dominante minimal de  $C_c(G)$  e obtemos um conjunto dominante de cardinalidade mínima, por exemplo,  $\{b, r, d, e, o\}$  e máxima, por exemplo,  $\{a, f, m, p, q, t, u\}$  (ver Figura 5.7).

Na verdade, os Teoremas 5.6 e 5.7 nos dizem um pouco mais, como mostra a próxima propriedade.

**Propriedade 5.6.** *Considere um grafo de cliques críticas  $C_c(G)$  de um grafo estritamente cordal  $G$ . Quando determinamos  $\gamma(C_c(G))$  e  $\Gamma(C_c(G))$ , determinamos não só  $\gamma(G)$  e  $\Gamma(G)$ , mas também os números de dominação e dominação superior de toda uma classe de grafos que tenha o grafo de cliques críticas isomorfo ao grafo  $C_c(G)$ .*

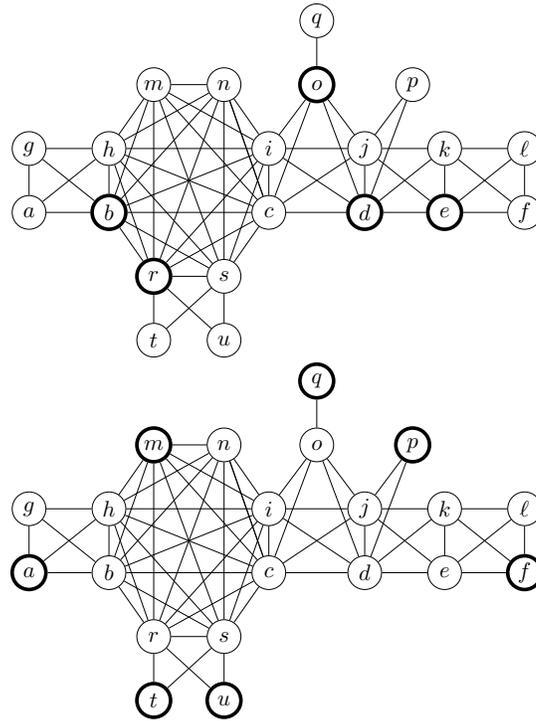


Figura 5.7: Conjuntos dominantes minimais em  $G$

Considere o grafo bloco  $G$  da Figura 5.8 como sendo um grafo de cliques críticas de um grafo estritamente cordal com 5 cliques críticas. Os conjuntos  $\{C_2, C_3\}$  e  $\{C_1, C_4, C_5\}$  são dominantes minimais; e os números de dominação são  $\gamma(G) = 2$  e  $\Gamma(G) = 3$ . O grafo  $G$  é grafo de cliques críticas tanto do grafo  $G_1$  quanto de  $G_2$ , apresentados na mesma figura. Ambos têm os referidos números de dominação.

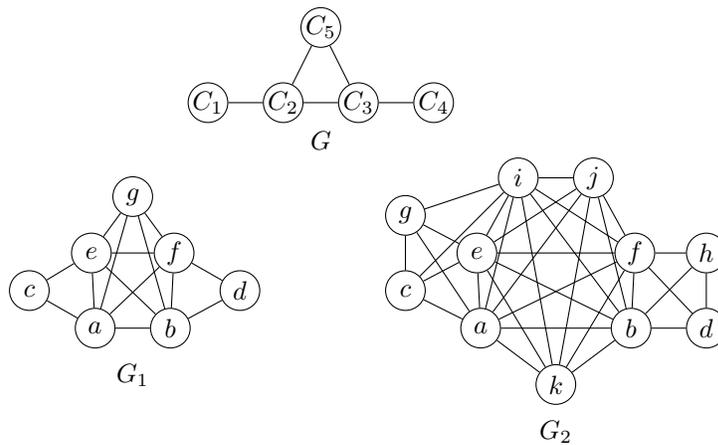


Figura 5.8: Grafos com os mesmos invariantes de dominação

### 5.4.1 Dominação conexa

Um conjunto dominante conexo de  $G$  é um conjunto dominante  $D$  cujo subgrafo induzido  $G[D]$  é conexo. A cardinalidade do menor conjunto dominante conexo é o número de dominação conexa  $\gamma_c(G)$ . Trivialmente, temos que  $\gamma(G) \leq \gamma_c(G)$ .

Dado um grafo  $G$  estritamente cordal, é possível considerar primeiramente o número de dominação conexa de seu grafo de cliques críticas  $C_c(G)$  e depois obter alguns resultados para o grafo  $G$  propriamente dito. Lembramos que  $C_c(G)$ , nesse caso, é um grafo bloco.

**Teorema 5.8.** *Seja  $G$  um grafo bloco não completo. Então  $\gamma_c(G) = |\mathbb{S}|$ .*

**Corolário 5.2.** *Seja  $G$  um grafo estritamente cordal não completo.*

*Então,  $\gamma_c(C_c(G)) = |\mathbb{S}|$ .*

Podemos demonstrar, de forma análoga ao Teorema 5.6, o seguinte resultado.

**Teorema 5.9.** *Se  $G$  é um grafo estritamente cordal então  $\gamma_c(G) = \gamma_c(C_c(G))$ .*

Vamos voltar ao grafo  $G$  da Figura 5.2 e ao seu grafo de cliques críticas da Figura 5.3. A Figura 5.9 exibe o conjunto dominante conexo de  $C_c(G)$

$$\{\{b, h\}, \{r, s\}, \{c, i\}, \{o\}, \{d, j\}, \{e, k\}\}.$$

Temos que  $\gamma_c(C_c(G)) = \gamma_c(G) = 6$  e um conjunto dominante conexo de  $G$  é, por exemplo,  $\{b, s, c, o, d, e\}$ .

### 5.4.2 Grafos bem dominados

Um grafo  $G$  é um grafo bem dominado se todos os conjuntos dominantes minimais de vértices têm a mesma cardinalidade, isto é,  $\gamma(G) = \Gamma(G)$ .

O próximo teorema caracteriza os grafos estritamente cordais bem dominados.

**Teorema 5.10.** *O grafo estritamente cordal  $G = (V, E)$  é bem dominado se e somente se as cliques simpliciais de  $G$  formam uma partição de  $V$ .*

Demonstração:

Considere  $\mathbb{Q}_S \subseteq \mathbb{Q}$  o conjunto de cliques simpliciais de  $G$ .

( $\rightarrow$ ) Seja  $\gamma(G) = \Gamma(G) = k$  e  $D = \{v_1, \dots, v_k\}$  um conjunto dominante do grafo. Se  $\mathbb{Q}_S$  não é uma partição de  $V$ , dois casos devem ser analisados.

1.  $\bigcup_{Q \in \mathbb{Q}_S} Q = V$  mas existem  $Q, Q' \in \mathbb{Q}_S$ ,  $Q \neq Q'$  e  $Q \cap Q' \neq \emptyset$ , isto é, existe um  $smv$   $S = Q \cap Q'$  e, sem perda de generalidade, existem vértices simpliciais  $v_i, v_j \in D$ ,  $i \neq j$ , com  $v_i \in Q$ ,  $v_j \in Q'$  e separados por  $S$ . Então eles poderiam ser substituídos por qualquer  $s \in S$  e teríamos um novo conjunto dominante.
2.  $\bigcup_{Q \in \mathbb{Q}_S} Q \subset V$ ; então existem cliques maximais  $Q$  e  $Q'$  não simpliciais,  $Q \neq Q'$  e  $Q \cap Q' \neq \emptyset$ , isto é, existe um  $smv$   $S$  que não está contido em nenhuma clique simplicial. Um vértice de  $S$  deve ser incluído em  $D$ .

Em qualquer um dos casos, existe um conjunto dominante minimal  $D'$  tal que  $|D'| \neq k$  e  $\gamma(G) \neq \Gamma(G)$ . Contradição.

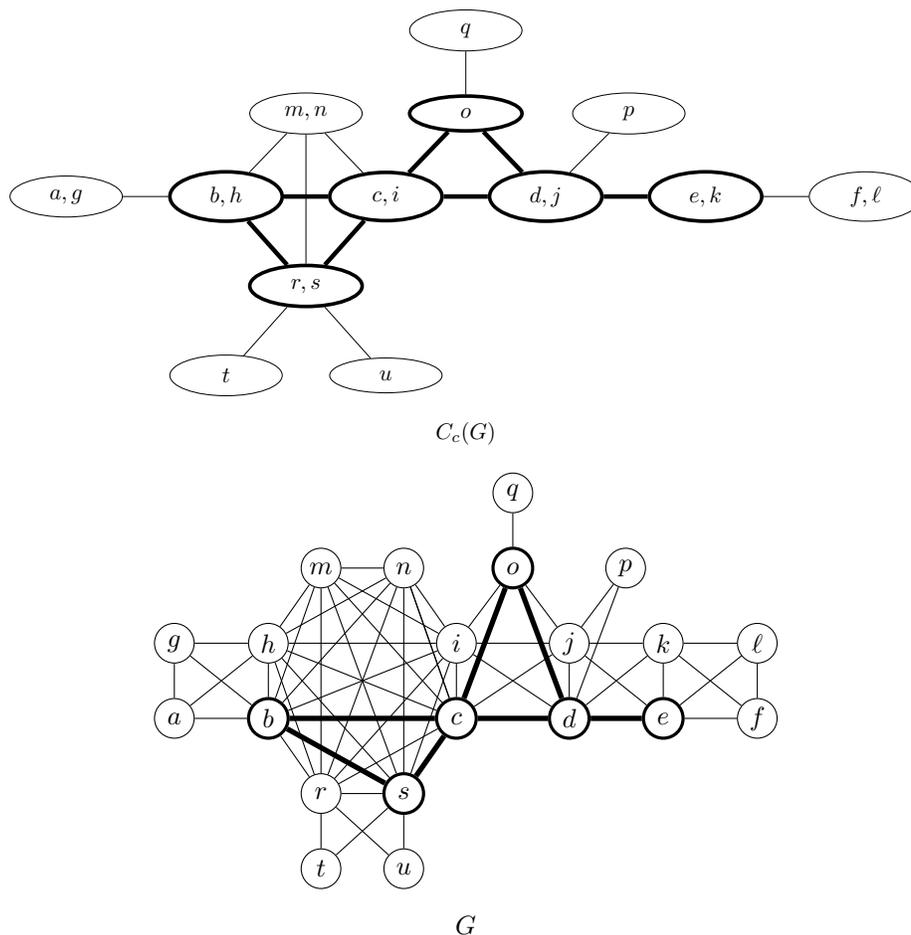


Figura 5.9: Conjuntos dominantes conexos em  $C_c(G)$  e em  $G$

( $\leftarrow$ ) Se  $\mathcal{Q}_S$  é uma partição de  $V$  então existe um conjunto  $D$  dominante minimal tal que  $|D| = |\mathcal{Q}_S|$ . Mas  $D$  tem cardinalidade mínima (não há vértices a retirar) e máxima (não há vértices a incluir). Então  $\gamma(G) = \Gamma(G)$ . ■

O grafo  $G$  da Figura 5.2 não é bem dominado. Já vimos que  $\gamma(G) = 5$  e  $\Gamma(G) = 7$  e, de fato, suas 7 cliques simpliciais  $Q_1, Q_2, Q_3, Q_4, Q_6, Q_7$  e  $Q_9$  não são uma partição do conjunto de vértices (caso 1 da demonstração do teorema). Na Figura 5.10, temos um grafo estritamente cordal também com 7 cliques maximais e tal que o conjunto das cliques simpliciais  $\{\{a, b, g\}, \{c, h, k\}, \{d, i, \ell\}, \{e, f, j\}\}$  é uma partição do conjunto de vértices. O grafo é bem dominado por 4 vértices, por exemplo, pelo conjunto  $\{b, c, d, e\}$ .

Finalizamos com mais uma propriedade de  $C_c(G)$ .

**Propriedade 5.7.** O grafo estritamente cordal  $G = (V, E)$  é bem dominado se e somente se  $C_c(G)$  é bem dominado.

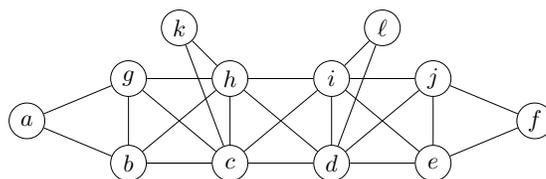


Figura 5.10: Grafo bem dominado

### 5.4.3 Dominação e novas subclasses

Uma releitura possível para os problemas específicos de dominação é que eles podem ser vistos como restrições para definir (sub)classes. Por exemplo, um grafo  $G$  é bem dominado quando  $\gamma(G) = \Gamma(G)$ . Essa condição pode ser utilizada para definir a subclasse dos *grafos estritamente cordais bem dominados*.

O mesmo pode ser feito para o invariante da dominação conexa. Podemos considerar os grafos estritamente cordais tais que  $\gamma(G) = \gamma_c(G)$  ou tais que  $\gamma_c(G) = \Gamma(G)$ . Vamos ficar com o primeiro caso. O fato de um grafo  $G$  satisfazer a igualdade  $\gamma(G) = \gamma_c(G)$  o seleciona como elemento de uma certa subclasse.

Na Figura 5.11, o grafo  $G_1$  é tal que  $\gamma(G_1) = \gamma_c(G_1) = 2$ ;  $G_3$  possui  $\gamma(G_3) = \Gamma(G_3) = 3$ ;  $G_2$  satisfaz as duas igualdades e  $G_4$  nenhuma delas.

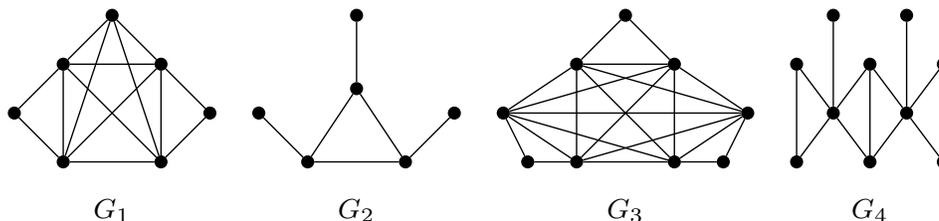


Figura 5.11: Subclasses via critérios de dominação

## 5.5 Grafos AC

Nesta seção vamos abordar uma subclasse dos grafos cordais, os grafos AC, definidos a partir de uma restrição no grafo de interseção de cliques. Um *grafo AC* é um grafo cujo grafo de interseção de cliques é acíclico. Na Figura 5.12 temos um exemplo. O grafo de cliques é uma árvore; por conseguinte, sua árvore de cliques é única.

É possível estabelecer restrições sobre a pertinência dos vértices de um grafo cordal em relação às cliques maximais. Temos a seguinte caracterização conhecida que segue esta linha.

**Teorema 5.11.** *Um grafo  $G$  é um grafo AC se e somente se é cordal e cada vértice pertence a no máximo duas cliques maximais.*

Outra caracterização da classe encontrada na literatura é por subgrafos proibidos.

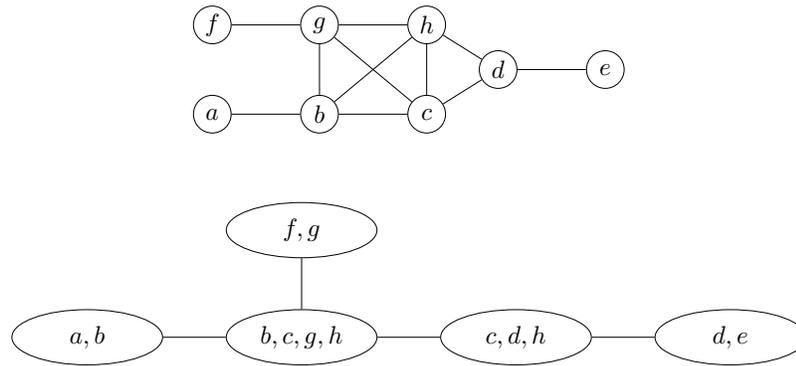


Figura 5.12: Grafo AC e seu grafo de cliques

**Teorema 5.12.** *Um grafo cordal  $G$  é um grafo AC se e somente se é livre de gema e de garra.*

Grafos AC também podem ser caracterizados pelos separadores minimais de vértices, o que nos permite estabelecer a relação da classe com os grafos estritamente cordais.

**Teorema 5.13.** *Um grafo cordal  $G$  é um grafo AC se e somente se para quaisquer separadores minimais de vértices distintos  $S, S' \in \mathbb{S}$ ,  $S \cap S' = \emptyset$  e para todo  $S \in \mathbb{S}$ ,  $\mu(S) = 1$ .*

Demonstração:

( $\rightarrow$ ) Sejam  $S \neq S'$  separadores minimais de vértices tais que  $S \subset Q$  e  $S' \subset Q'$  sendo  $Q$  e  $Q'$  cliques maximais. Supor que  $S \cap S' \neq \emptyset$ . Existe uma clique maximal  $Q''$  tal que  $S \cap S' \subset Q''$ . Assim, existe  $v \in S \cap S'$ , com  $v \in Q$ ,  $v \in Q'$  e  $v \in Q''$ , contrariando o Teorema 5.11.

Se  $\mu(S) \neq 1$ , vamos considerar, sem perda de generalidade, que seja igual a 2. Existem três clique maximais  $Q$ ,  $Q'$  e  $Q''$  tais que  $S = Q \cap Q' = Q \cap Q''$  e, conseqüentemente,  $Q' \cap Q'' \neq \emptyset$ , formando um ciclo no grafo de cliques de  $G$ .

( $\leftarrow$ ) Imediato. ■

O Teorema 5.13 nos permite concluir diretamente que a classe dos grafos AC é a interseção da classe dos grafos estritamente cordais com a dos grafos ur-cordais.

**Corolário 5.3.** *Todo grafo AC é um grafo ur-cordal e estritamente cordal.*

### 5.5.1 Reconhecimento

O Teorema 5.11 sugere um método imediato para determinar se um grafo  $G$  é ou não um grafo AC. Considere um grafo cordal  $G = (V, E)$  e seu conjunto  $\mathbb{Q}$  de cliques maximais. O algoritmo utiliza um rótulo *contador*( $v$ ) inicializado com zeros para cada um dos vértices de  $G$ . O número de ocorrências do vértice nas cliques maximais é armazenado em *contador*. Se esse ultrapassar dois, isso significa que o grafo não é AC.



árvore  $T$  é uma raiz 3-folha. Veremos que os grafos potência de 3-folha formam uma subclasse dos grafos estritamente cordais.

Na Figura 5.14, temos a árvore raiz 3-folha  $T$  do grafo  $G$  da Figura 5.10.

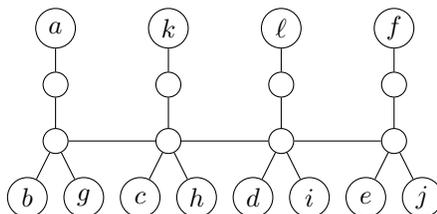


Figura 5.14: Árvore raiz 3-folha

Observamos nas tabelas a seguir que as distâncias dos vértice  $a$ ,  $b$  e  $i$  para os seus vizinhos em  $G$  na árvore  $T$  são menores ou iguais a 3. Isso ocorre para todos os vértices de  $G$ .

$dist_T$	$b$	$g$
$a$	3	3

$dist_T$	$a$	$c$	$g$	$h$
$b$	3	3	2	3

$dist_T$	$c$	$d$	$e$	$h$	$j$	$l$
$i$	3	2	3	3	3	3

Existem na literatura inúmeros resultados para os grafos potência de 3-folha. Em particular, os que se seguem são bem interessantes.

**Teorema 5.15.** *Um grafo cordal  $G$  é uma potência de 3-folha se e somente se  $G$  é obtido pela substituição dos vértices de uma árvore por cliques.*

**Teorema 5.16.** *Um grafo cordal  $G$  é uma potência de 3-folha se e somente se  $G$  é livre de gema, de seta e de touro.*

Pelos Teoremas 5.1 e 5.16, temos como decorrência:

**Corolário 5.4.** *Todo grafo potência de 3-folha é um grafo estritamente cordal.*

**Teorema 5.17.** *Seja  $G$  uma potência de 3-folha. Então, toda clique maximal de  $G$  tem seus vértices em no máximo duas cliques críticas.*

Demonstração:

Se dois vértices  $u$  e  $v$  de  $G$  são adjacentes a um mesmo vértice interno (que não é uma folha) em uma árvore raiz 3-folha  $T$  de  $G$ , então  $dist_T(u, v) = 2$  e eles possuem a mesma vizinhança em  $G$  (são gêmeos verdadeiros) pois suas distâncias para as outras folhas de  $T$  são iguais. Portanto, vértices que pertencem a cliques críticas distintas não podem ser adjacentes a um mesmo vértice interno de  $T$ . Assim, se os vértices de uma clique maximal  $Q$  de  $G$  são elementos de pelo menos três cliques críticos, esses vértices precisam ser adjacentes a pelo menos três diferentes vértices internos de  $T$ . Dois destes três vértices internos de  $T$  têm distância igual a dois, o que acarreta que suas folhas têm distância quatro. Contradição. ■

Apresentamos mais uma caracterização utilizando o grafo de cliques críticas que é a base de um algoritmo eficiente de reconhecimento.

**Teorema 5.18.** *Um grafo cordal  $G = (V, E)$  é uma potência de 3-folha se e somente se  $C_c(G)$  é uma árvore.*

Demonstração:

( $\rightarrow$ ) Se  $C_c(G)$  não é uma árvore então existe um ciclo formado pelas cliques críticas  $C, C'$  e  $C''$  com vértices  $u \in C, v \in C'$  e  $w \in C''$  adjacentes dois a dois em  $G$ . Temos que os vértices  $u, v$  e  $w$  estão em uma mesma clique maximal  $Q$  de  $G$ , o que contraria o Teorema 5.17.

( $\leftarrow$ ) Vamos construir uma raiz 3-folha  $T$  para  $G$  a partir da sua árvore de cliques críticas. Cada vértice de  $C_c(G)$  corresponde a um vértice interno de  $T$ ; vamos adicionar tantas folhas quantos forem os vértices da clique crítica em questão, adjacentes à mesma. Este novo grafo  $T$  é uma árvore, já que só foram acrescentadas folhas na árvore  $C_c(G)$ . Quaisquer dois vértices de  $u, v \in V$  são adjacentes em  $G$  quando pertencem à mesma clique crítica ( $dist_T(u, v) = 2$ ) ou pertencem a duas cliques críticas adjacentes em  $C_c(G)$  ( $dist_T(u, v) = 3$ ). Logo,  $T$  é uma raiz 3-folha de  $G$  e, conseqüentemente,  $G$  é um grafo potência de 3-folha. ■

Na Figura 5.15 temos o grafo de cliques críticas do grafo  $G$  da Figura 5.10.

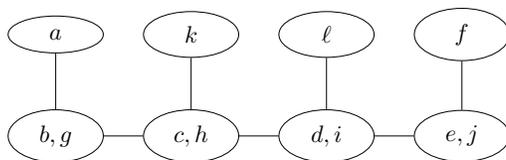


Figura 5.15: Grafo de cliques críticas

**Propriedade 5.8.** *O grafo de cliques críticas de um grafo potência de 3-folha é a árvore mencionada no Teorema 5.15.*

### 5.6.1 Reconhecimento

O reconhecimento de um grafo estritamente cordal  $G$  como sendo um grafo potência de 3-folha pode ser feito em dois passos.

**Passo 1.** Construir  $C_c(G)$  usando o Algoritmo 5.2.

**Passo 2.** Verificar se  $C_c(G)$  é uma árvore. Em caso afirmativo,  $G$  é um grafo potência de 3-folha.

Ambos os passos têm complexidade de tempo linear.

## 5.7 Grafos estritamente de intervalo

Nessa seção tratamos de uma classe recentemente definida, grafos que são simultaneamente estritamente cordais e de intervalo, os *grafos estritamente de intervalo* (*GEI*). Na Figura 5.16 temos um exemplar da classe.

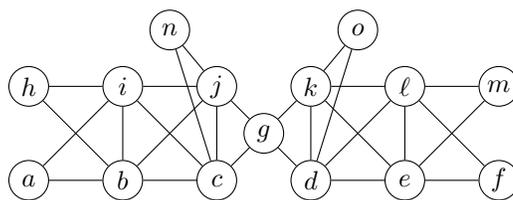


Figura 5.16: Grafo estritamente de intervalo

Uma primeira caracterização decorre do Corolário 3.1; é imediato que um grafo  $G$  é um grafo estritamente de intervalo quando  $G$  é completo ou  $\ell(G) = 2$ . Temos também uma caracterização por subgrafos proibidos. O teorema a seguir reúne essas caracterizações.

**Teorema 5.19.** *Seja  $G = (V, E)$  um grafo estritamente cordal não completo. As seguintes afirmativas são equivalentes:*

1.  $G$  é um grafo estritamente de intervalo.
2.  $G$  é livre de 2-rede e de garra dupla.
3.  $\ell(G) = 2$ .

Demonstração:

(1  $\rightarrow$  2) Se  $G$  é um grafo estritamente de intervalo,  $G$  é um grafo de intervalo. Então  $G$  é livre de  $AT$  (Teorema 3.1). A 2-rede e a garra dupla são grafos que possuem um  $AT$ . Então  $G$  é livre de 2-rede e de garra dupla.

(2  $\rightarrow$  3) O grafo  $G$  é estritamente cordal e como tal livre de gema e de seta. Como é também livre de 2-rede e de garra dupla, ciclos maiores do que 3 e guarda-chuva, ele tem todos os subgrafos proibidos de um grafo de intervalo (Teorema 3.3). E, pelo Corolário 3.1,  $\ell(G) = 2$ .

(3  $\rightarrow$  1) Imediato. ■

**Corolário 5.5.** *Toda clique maximal de um grafo estritamente de intervalo  $G$  contém no máximo dois separadores minimais de vértices.*

### 5.7.1 Reconhecimento

A caracterização apresentada no Teorema 5.19 nos permite desenvolver um algoritmo linear para o reconhecimento de um grafo estritamente de intervalo. Num primeiro passo, verificamos se  $G$  é um grafo estritamente cordal. Depois, considerando o terceiro item do Teorema 5.19, a folhagem mínima de  $G$  é calculada e analisada. Precisamos então conhecer a folhagem mínima de um grafo estritamente cordal. Para tal, vamos utilizar o grafo de cliques críticas, como mostra o próximo teorema.

**Teorema 5.20.** *Se  $G$  é um grafo estritamente cordal então  $\ell(G) = \ell(C_c(G))$ .*

Demonstração:

Pelo Teorema 2.17, verifica-se diretamente. ■

Recaímos então no problema de determinar a folhagem mínima de um grafo bloco. O seguinte resultado é encontrado na literatura, lembrando que  $Simp$  é o conjunto de vértices simpliciais de um grafo.

**Teorema 5.21.** *Seja  $G$  um grafo bloco não completo. Então  $\ell(G) = \max\{2, r(G)\}$  sendo  $r(G)$  o número de separadores minimais de vértices de  $G$  que não o são em  $G[V \setminus Simp]$ .*

O algoritmo de reconhecimento consta de dois passos.

**Passo 1.** Determinar o grafo  $C_c(G)$ .

**Passo 2.** Determinar  $\ell(C_c(G))$ .

Se  $\ell(C_c(G)) = 2$  então  $G$  é  $GEI$ .

Para executar o Passo 2, é preciso construir o grafo  $G'$ , resultante da remoção dos vértices simpliciais de  $C_c(G)$ . Observe que os separadores minimais de vértices de um grafo bloco são conjuntos unitários. Então, um  $smv$  de  $C_c(G)$ , que não é  $smv$  em  $G'$ , é simplicial em  $G'$ . Conclui-se que basta encontrar os vértices simpliciais de  $G'$  para termos a solução do problema.

O algoritmo baseado nesses passos é linear pois a construção do grafo de cliques críticas (Algoritmo 5.2), a construção de  $G'$  e a determinação do número de vértices simpliciais têm solução em tempo linear.

## 5.8 Comparando classes

Essa seção é dedicada ao estudo mais detalhado do relacionamento de algumas subclasses de grafos cordais mencionadas no texto. Vamos comparar inicialmente duas classes: a dos grafos estritamente de intervalo (subclasse dos grafos estritamente cordais) e dos grafos de intervalo próprio (subclasse dos grafos ur-intervalo), ambas subclasses dos grafos de intervalo. Depois exibimos exemplos para quatro das subclasses dos grafos estritamente cordais.

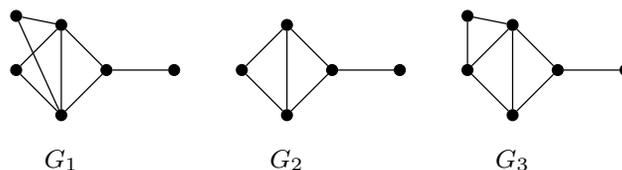
### 5.8.1 Grafos estritamente de intervalo $\times$ grafos de intervalo próprio

Já vimos no Capítulo 4 que grafos de intervalo próprio ( $GIP$ ) são uma importante subclasse de grafos de intervalo. Qual será então a relação entre esses grafos e a classe dos grafos estritamente de intervalo? Não é difícil perceber que as duas classes são distintas. Além disso,  $GIP \not\subseteq GEI$  e  $GEI \not\subseteq GIP$ .

O próximo teorema apresenta como podemos distinguir grafos pertencentes às duas classes.

**Teorema 5.22.** *Seja  $G$  um grafo de intervalo.*

1. *Se  $G$  é um  $GEI$  então  $G$  é um  $GIP$  se e somente se para todo separador minimal de vértices  $S \in \mathbb{S}$ ,  $\mu(S) = 1$ .*

Figura 5.17:  $GEI \times GIP$ 

2. Se  $G$  é um  $GIP$  então  $G$  é um  $GEI$  se e somente se para quaisquer separadores minimais de vértices distintos  $S, S' \in \mathbb{S}$ ,  $S \cap S' = \emptyset$ .

Devido a suas características, podemos chamar a classe dos grafos  $GEI \cap GIP$  de *ur-estritamente de intervalo*. Observe que um grafo ur-estritamente de intervalo é um grafo  $AC$ .

Na Figura 5.17, o grafo  $G_1 \in GEI \setminus GIP$ ,  $G_2 \in GEI \cap GIP$  e  $G_3 \in GIP \setminus GEI$ .

### 5.8.2 Quatro subclasses

Nessa seção vamos comparar quatro subclasses de grafos estritamente cordais: grafos bloco, grafos  $AC$ , grafos potência de 3-folha e grafos estritamente de intervalo. É fácil perceber que as classes são todas distintas.

**Teorema 5.23.** *Não existe um grafo cordal  $G$  tal que  $G$  pertença simultaneamente às classes  $AC$  e potência de 3-folha e que não pertença à classe dos grafos estritamente de intervalo ou à classe dos grafos bloco.*

Demonstração:

Seja  $G$  um grafo  $AC$  e potência de 3-folha. Pelo Teorema 5.15, sabemos que  $G$  pode ser obtido pela substituição dos vértices de uma árvore por cliques. Então, sem perda de generalidade, uma árvore é um grafo potência de 3-folha. Como cada vértice de um grafo  $AC$  pertence a, no máximo, duas cliques maximais, cada vértice da árvore só pode ter, no máximo, dois vizinhos. Então,  $G$  é um caminho. Porém, nesse caso,  $G$  é também um grafo estritamente de intervalo e um grafo bloco. ■

**Teorema 5.24.** *Não existe um grafo cordal  $G$  tal que  $G$  pertença simultaneamente às classes bloco,  $AC$  e potência de 3-folha e que não pertença à classe dos grafos estritamente de intervalo.*

Demonstração:

Os grafos bloco cujos grafos de cliques críticas são árvores, são eles mesmos árvores. Assim, se  $G$  é um bloco e um grafo potência de 3-folha então  $G$  é uma árvore. Se além disso exigirmos que  $G$  seja um grafo  $AC$  então  $G$  é um caminho, pois este é a única árvore unicamente representável. Mas todo caminho é um grafo estritamente de intervalo. Logo, não existe um grafo que seja um bloco, potência de 3-folha e  $AC$  que não seja estritamente de intervalo. ■

Na Figura 5.18 temos um diagrama das interseções das quatro subclasses, exemplificadas com os respectivos grafos. Por abuso de linguagem, usamos  $\emptyset$  no diagrama

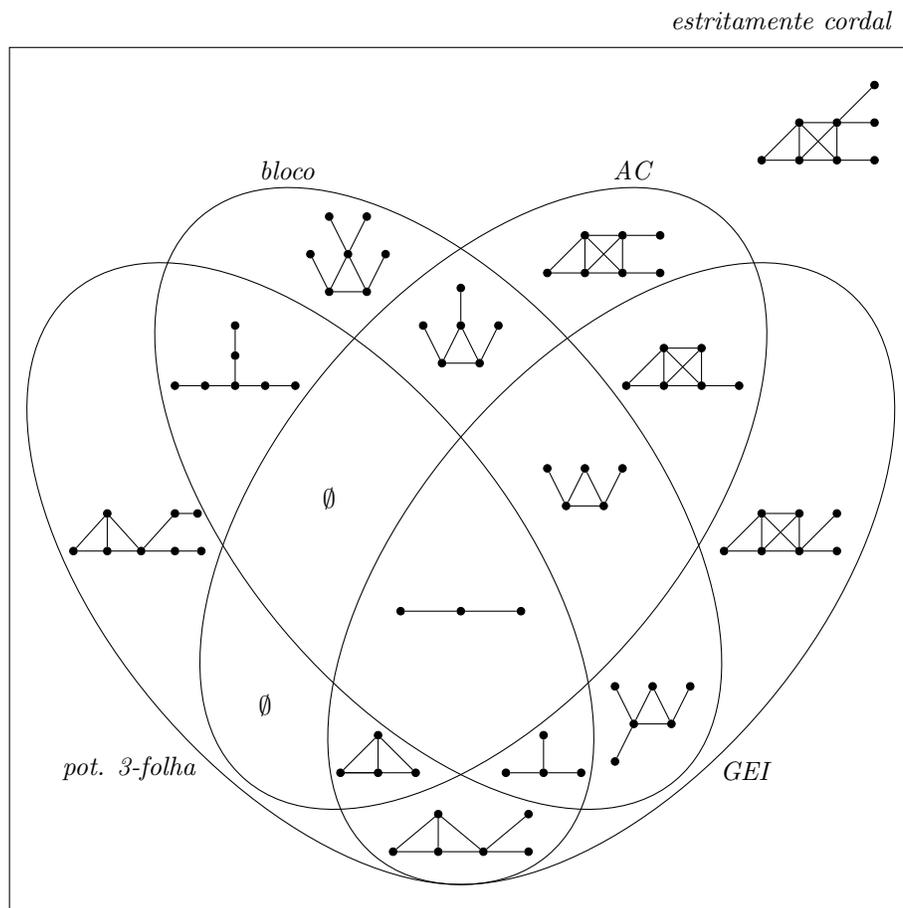


Figura 5.18: Exemplos para as interseções das 4 subclasses

para indicar que a interseção é vazia. Pretende-se que os outros exemplos sejam autoexplicativos.

## 5.9 A hierarquia

A Figura 5.19 ilustra a hierarquia dos grafos estudados no texto. Note a exibição das três linhas de classes abordadas: os grafos de intervalo, os grafos cordais unicamente representáveis e os grafos ptolemaicos. São mostradas, para cada uma das linhas, as interseções de subclasses, dando assim ênfase ao amplo relacionamento entre as mesmas.

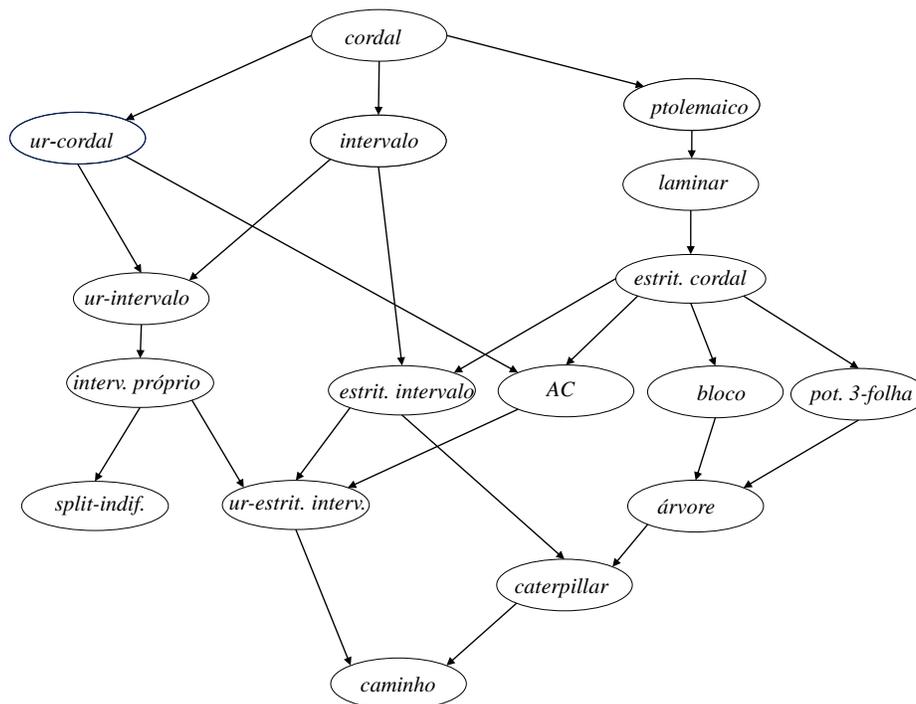


Figura 5.19: Hierarquia final

### 5.10 Exercícios

1. Considere um grafo  $G$  com as cliques maximais abaixo. O grafo  $G$  é estritamente cordal? Sim ou não, justifique.

$$\{\{a, b, c, d, e\}, \{b, c, d, e, f\}, \{f, g, h, i, j\}, \{f, i, j, k\}\}$$

2. Use a caracterização apresentada no Teorema 5.3 para decidir se o grafo  $G$  da Figura 5.20 é estritamente cordal.

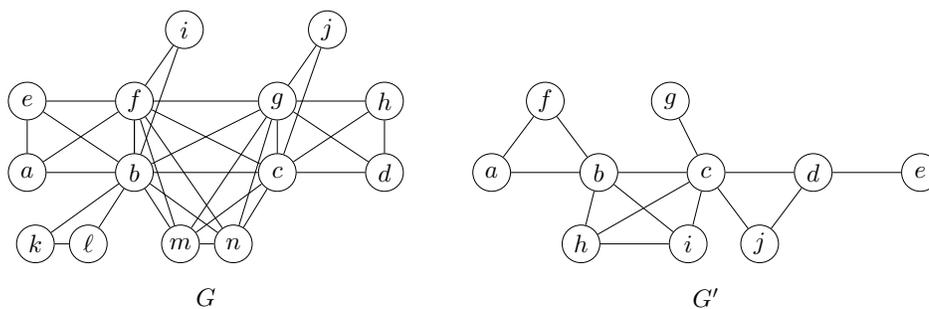


Figura 5.20: Grafos

3. Apresente um exemplo de um grafo estritamente cordal que não seja ur-cordal. Mostre duas árvores de cliques distintas para o grafo.
4. Considere os grafos  $G_1$  e  $G_2$  da Figura 5.8. Indique conjuntos dominantes minimais de cardinalidades mínima e máxima.
5. Determine  $\gamma(G')$ ,  $\Gamma(G')$  e  $\gamma_c(G')$  para o grafo  $G'$  da Figura 5.20.
6. Indique um grafo  $G$  que exemplifique o caso 2 do Teorema 5.10.
7. Apresente um algoritmo para o reconhecimento dos grafos estritamente cordais bem dominados e estabeleça a sua complexidade.
8. O grafo  $G_1$  da Figura 5.1 não é um grafo potência de 3-folha. Use o Teorema 5.16 para justificar essa afirmação.
9. O grafo da Figura 5.10 é um grafo estritamente de intervalo? Justifique.
10. Considere o diagrama da Figura 5.21. Ele está correto? Justifique. Em caso afirmativo, apresente grafos para exemplificar as cinco possibilidades.

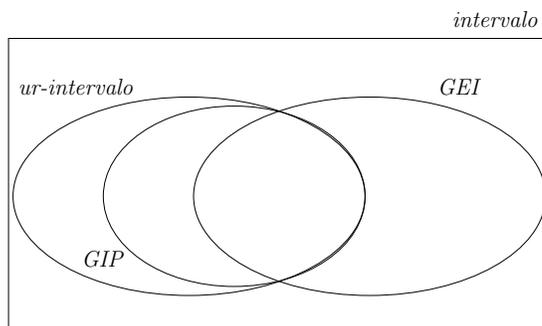


Figura 5.21: Subclasses de grafos de intervalo

11. Justifique a pertinência de cada um dos grafos da Figura 5.18 aos conjuntos indicados.

## 5.11 Notas bibliográficas

Grafos estritamente cordais foram definidos em 2005 por Kennedy [40], baseados em propriedades de hipergrafos, mas já haviam sido introduzidos em 2002 como *grafos bloco duplicados* por Golubic e Peled [30].

A caracterização dos grafos estritamente cordais por subgrafos proibidos é devida a Dom *et al.* [20]; relacionando-a ao grafo de cliques críticas, é devida a Brandstädt e Wagner [11]. O Teorema 5.1 foi apresentado por Markenzon e Waga [55].

O livro de Haynes *et al.* [36] é um texto fundamental sobre dominação. Sampathkumar e Walikar [67] definiram conjunto dominante conexo e Chen *et al.* [16] determinaram o número de dominação conexa para grafos bloco. Quanto a grafos bem dominados, o assunto foi introduzido por Finbow *et al.* [24]. No trabalho de Topp e Volkmann [73] podemos encontrar algumas caracterizações de grafos bloco

bem dominados com uma forte relação com grafos bem cobertos. Dominação e dominação conexa para os grafos estritamente cordais foram abordados por Markenzon e Waga [56].

Os grafos  $AC$  foram definidos e caracterizados por Blair e Ravi [4] e Blair [2].

A noção de grafo potência de  $k$ -folha e de árvore raiz  $k$ -folha foram introduzidas e estudadas por Nishimura *et al.* [61]. As caracterizações da classe foram apresentadas por Dom *et al.* [20] (subgrafos proibidos), por Brandstädt e Le [8] (Teorema 5.15) e por Kennedy *et al.* [41]. Brandstädt e Wagner introduziram os grafos potência de  $(k, \ell)$ -folha [10] e provaram que os grafos estritamente cordais são exatamente os grafos potência de  $(4, 6)$ -folha [11].

A classe dos grafos estritamente de intervalo foi definida em Markenzon e Waga [58]. O Teorema 5.21 é devido a Lin *et al.* [48].



# Bibliografia

- [1] BANDELT, H.-J.; MULDER, H. M. Distance-hereditary graphs. *Journal of Combinatorial Theory, Series B*, v. 41, p. 182–208, 1986.
- [2] BLAIR, J. R. S. The efficiency of ac graphs. *Discrete Applied Mathematics*, v. 44, p. 119–138, 1993.
- [3] BLAIR, J. R. S.; PEYTON, B. An introduction to chordal graphs and clique trees. In *Graph Theory and Sparse Matrix Computation*, IMA, v. 56, p. 1–29, 1993.
- [4] BLAIR, J. R. S.; RAVI, S. S. Tr-matching for chordal graphs. In: MEDANIC, J. V. (Ed.). : Proceedings of the Twenty-Seventh Annual Allerton Conference on Communication, Control, and Computing, 1989. p. 72–81.
- [5] BODLAENDER, H. L. et al. Simple max-cut for split-indifference graphs and graphs with few  $p_4$ 's. *Lecture Notes in Computer Science*, v. 3059, p. 87–99, 2004.
- [6] BOGART, K. P.; WEST, D. B. A short proof that “proper = unit”. *Discrete Mathematics*, v. 201, p. 21–23, 1999.
- [7] BOOTH, K. S.; LUEKER, G. S. Testing for the consecutive ones property, interval graphs, and graph planarity using pq-tree algorithms. *Journal of Computer and System Sciences*, v. 13, p. 335–379, 1976.
- [8] BRANDSTÄDT, A.; LE, V. B. Structure and linear time recognition algorithm of 3-leaf powers. *Information Processing Letters*, v. 98, p. 133–138, 2006.
- [9] BRANDSTÄDT, A.; LE, V. B.; SPINRAD, J. *Graph Classes - a Survey*. Philadelphia, PA: SIAM Monographs in Discrete Mathematics and Applications, 1999.
- [10] BRANDSTÄDT, A.; WAGNER, P. On  $(k, \ell)$ -leaf powers. In: KUČERA, L.; KUČERA, A. (Ed.). MFCS: LNCS 4708, 2007. p. 525–535.
- [11] BRANDSTÄDT, A.; WAGNER, P. Characterising  $(k, \ell)$ -leaf powers. *Discrete Applied Mathematics*, v. 158, p. 110–122, 2010.
- [12] BRASSARD, G.; BRATLEY, P. *Algorithmics: Theory and Practice*. Englewood Cliffs, New Jersey: Prentice Hall, 1988.
- [13] BUNEMAN, P. A characterization of rigid circuit graphs. *Discrete Mathematics*, v. 9, p. 205–212, 1974.
- [14] CAMPOS, C. N. et al. The total chromatic number of split-indifference graphs. *Discrete Mathematics*, v. 312, p. 2690–2693, 2012.

- [15] CHANDRAN, L. S. et al. Generating and characterizing the perfect elimination orderings of a chordal graph. *Theoretical Computer Science*, v. 307, p. 303–317, 2003.
- [16] CHEN, X. g.; SUN, L.; XING, H. m. Characterization of graphs with equal domination and connected domination numbers. *Discrete Mathematics*, v. 289, p. 129–135, 2004.
- [17] CORNEIL, D. G. A simple 3-sweep lbf algorithm for the recognition of unit interval graphs. *Discrete Applied Mathematics*, v. 138, p. 371–379, 2004.
- [18] CORNEIL, D. G.; OLARIU, S.; STEWART, L. The lbf structure and recognition of interval graphs. *SIAM Journal on Discrete Mathematics*, v. 23, p. 1905–1953, 2009.
- [19] DIESTEL, R. *Graph Theory: Graduate texts in mathematics 173*. 2. ed. New York: Springer-Verlag, 2000.
- [20] DOM, M. et al. Error compensation in leaf root problems. *Lecture Notes in Computer Science*, v. 3341, p. 389–401, 2004.
- [21] FARBER, M. Characterizations of strongly chordal graphs. *Discrete Mathematics*, v. 43, p. 173–189, 1983.
- [22] FARBER, M. Domination, independent domination, and duality in strongly chordal graphs. *Discrete Applied Mathematics*, v. 7, p. 115–130, 1984.
- [23] FIGUEIREDO, C. M. H. de; MEIDANIS, J.; MELLO, C. P. de. A linear-time algorithm for proper interval graph recognition. *Information Processing Letters*, v. 56, p. 179–184, 1995.
- [24] FINBOW, B. H. A.; NOWAKOWSKI, R. J. Well-dominated graphs: a collection of well-covered ones. *Ars Combinatoria*, v. 25A, p. 5–10, 1988.
- [25] GALINIER, P.; HABIB, M.; PAUL, C. Chordal graphs and their clique graphs. *Lecture Notes in Computer Science*, v. 1017, p. 385–371, 1995.
- [26] GAREY, M. R.; JOHNSON, D. S. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York: W.H. Freeman and Company, 1979.
- [27] GAVRIL, F. The intersection graphs of subtrees in trees are exactly the chordal graphs. *Journal of Combinatorial Theory, Series B*, v. 16, p. 47–56, 1974.
- [28] GILMORE, P. C.; HOFFMAN, A. J. A characterization of comparability graphs and of interval graphs. *Canadian Journal of Mathematics*, v. 16, p. 539–548, 1964.
- [29] GOLUBIC, M. C. *Algorithmic Graph Theory and Perfect Graphs: Annals of discrete mathematics 57*. 2. ed. New York: Academic Press, 2004.
- [30] GOLUBIC, M. C.; PELED, U. N. Block duplicate graphs and a hierarchy of chordal graphs. *Discrete Applied Mathematics*, v. 124, p. 67–71, 2002.
- [31] GROSS, J.; YELLEN, J. *Graph Theory and its Applications*. [S.l.]: CRC Press series on Discrete Mathematics and its Applications, 1998.
- [32] HABIB, M. et al. Lex-bfs and partition refinement, with applications to transitive orientation, interval graph recognition. *Theoretical Computer Science*, v. 234, p. 59–84, 2000.

- [33] HABIB, M.; STACHO, J. Reduced clique graphs of chordal graphs. *European Journal of Combinatorics*, v. 33, p. 712–735, 2012.
- [34] HARA, H.; TAKEMURA, A. Boundary cliques, clique trees and perfect sequences of maximal cliques of a chordal graph. *METR 2006-41, Department of Mathematical Informatics, University of Tokyo*, 2006.
- [35] HARARY, F. A characterization of block graphs. *Canadian Mathematical Bulletin*, v. 6, p. 1–6, 1963.
- [36] HAYNES, T. W.; HEDETNIEMI, S. T.; SLATER, P. J. *Fundamentals of Domination in Graphs*. New York: Marcel Dekker, 1998.
- [37] HOWORKA, E. A characterization of of ptolemaic graphs. *Journal of Graph Theory*, v. 5, p. 323–331, 1981.
- [38] HSU, W. L.; MA, T. H. Fast and simple algorithms for recognizing chordal comparability graphs and interval graphs. *SIAM Journal on Computing*, v. 28, p. 1004–1020, 1999.
- [39] IBARRA, L. A simple algorithm to find hamiltonian cycles in proper interval graphs. *Information Processing Letters*, v. 109, p. 1105–1108, 2009.
- [40] KENNEDY, W. Strictly chordal graphs and phylogenetic roots. *Master Thesis, University of Alberta*, 2005.
- [41] KENNEDY, W.; LIN, G.; YAN, G. Strictly chordal graphs are leaf powers. *Journal of Discrete Algorithms*, v. 4, p. 511–525, 2006.
- [42] KORTE, N.; MOHRING, R. H. An incremental linear-time algorithm for recognizing interval graphs. *SIAM Journal on Computing*, v. 18, p. 68–81, 1989.
- [43] KRATSCH, D. et al. Certifying algorithms for recognizing interval graphs and permutation graphs. *SIAM Journal on Computing*, v. 36, p. 326–353, 2006.
- [44] KUMAR, P. S.; MADHAVAN, C. E. V. Clique tree generalization and new subclasses of chordal graphs. *Discrete Applied Mathematics*, v. 117, p. 109–131, 2002.
- [45] LEKKERKERKER, C. G.; BOLAND, J. Representation of a finite graph by a set of intervals on the real line. *Fundamenta Mathematicæ*, v. 51, p. 45–64, 1962.
- [46] LI, P.; WU, Y. Maximal neighborhood search and rigid interval graphs. *Journal of Graph Algorithms and Applications*, v. 17, p. 245–264, 2013.
- [47] LIN, G.; KEARNEY, P. E.; JIANG, T. Phylogenetic k-root and steiner k-root. *Lecture Notes in Computer Science*, v. 1969, p. 539–551, 2000.
- [48] LIN, I.-J.; MCKEE, T. A.; WEST, D. B. The leafage of a chordal graph. *Disquisitiones Mathematicæ. Graph Theory*, v. 18, p. 23–48, 1998.
- [49] LOOGES, P. J.; OLARIU, S. Optimal greedy algorithms for indifference graphs. *Computers & Mathematics with Applications*, v. 25, p. 15–25, 1993.
- [50] MARKENZON, L.; JUSTEL, C. M.; PACIORNIK, N. Subclasses of  $k$ -trees: characterization and recognition. *Discrete Applied Mathematics*, v. 154, p. 818–825, 2006.

- [51] MARKENZON, L.; PEREIRA, P. R. C. One-phase algorithm for the determination of minimal vertex separators of chordal graphs. *International Transactions in Operational Research*, v. 17, p. 683–690, 2010.
- [52] MARKENZON, L.; PEREIRA, P. R. C.; WAGA, C. F. E. M.  $k$ -separator chordal graphs: leafage and subfamilies. *International Transactions in Operational Research*, v. 20, p. 681–688, 2013.
- [53] MARKENZON, L.; VERNET, O. *Representações Computacionais de Grafos*. São Paulo: Notas em Matemática Aplicada 24 SBMAC, 2006.
- [54] MARKENZON, L.; VERNET, O.; PEREIRA, P. R. C. Codificação de árvores: Conceitos e algoritmos. In: . ERMAC: Vitória, ES, 2005. p. 1–28.
- [55] MARKENZON, L.; WAGA, C. F. E. M. New results on ptolemaic graphs. *Discrete Applied Mathematics*, v. 196C, p. 135–140, 2015.
- [56] MARKENZON, L.; WAGA, C. F. E. M. Propriedades e aplicações de grafos de cliques críticas. In: . XLVII SBPO: Porto de Galinhas, Brazil, 2015. p. 3371–3378.
- [57] MARKENZON, L.; WAGA, C. F. E. M. Revisiting the recognition of proper interval graphs. *Journal of Combinatorial Mathematics and Combinatorial Computing*, aceito, 2016.
- [58] MARKENZON, L.; WAGA, C. F. E. M. Strictly interval graphs: Characterization and linear time recognition. *Electronic Notes in Discrete Mathematics*, v. 52, p. 181–188, 2016.
- [59] MCHUGH, J. A. *Algorithmic Graph Theory*. Englewood Cliffs, New Jersey: Prentice Hall Inc., 1990.
- [60] NETTO, P. O. B. *Grafos: Teoria, Modelos, Algoritmos*. 5. ed. São Paulo: Edgard Blücher, 2009.
- [61] NISHIMURA, N.; RAGDE, P.; THILIKOS, D. On graph powers for leaf-labeled trees. *Journal of Algorithms*, v. 42, p. 69–108, 2002.
- [62] ORTIZ, C.; MACULAN, N.; SZWARCFITER, J. L. Characterizing and edge-colouring split-indifference graphs. *Discrete Applied Mathematics*, v. 82, p. 209–217, 1998.
- [63] PANDA, B. S.; DAS, S. A linear time recognition algorithm for proper interval graphs. *Information Processing Letters*, v. 87, p. 153–161, 2003.
- [64] ROBERTS, F. S. *Indifference graphs: Proof techniques in graph theory*. New York: Academic Press, 1969.
- [65] ROBERTS, F. S. On the compatibility between a graph and a simple order. *Journal of Combinatorial Theory*, v. 11, p. 28–38, 1971.
- [66] ROSE, D. J.; TARJAN, R. E.; LUEKER, G. Algorithmic aspects of vertex elimination on graphs. *SIAM Journal on Computing*, v. 5, p. 266–283, 1976.
- [67] SAMPATHKUMAR, E.; WALIKAR, H. B. The connected domination number of a graph. *Journal of Mathematical and Physical Sciences*, v. 13, p. 607–613, 1979.

- [68] SAWADA, J.; SPINRAD, J. P. From a simple elimination ordering to a strong elimination ordering in linear time. *Information Processing Letters*, v. 86, p. 299–302, 2003.
- [69] SHIBATA, Y. On the tree representation of chordal graphs. *Journal of Graph Theory*, v. 12, p. 421–428, 1988.
- [70] SPINRAD, J. P. *Efficient Graph Representations*: The fields institute for research in mathematical sciences. Providence, Rhode Island: American Mathematical Society, 2003.
- [71] SZWARCFITER, J. L. *Grafos e Algoritmos Computacionais*. Rio de Janeiro: Editora Campus, 1984.
- [72] SZWARCFITER, J. L.; MARKENZON, L. *Estruturas de Dados e seus Algoritmos*. 3a. ed.. ed. Rio de Janeiro: LTC - Livros Técnicos e Científicos, 2010.
- [73] TOPP, J.; VOLKMANN, L. Well covered and well dominated block graphs and unicyclic graphs. *Mathematica Pannonica*, v. 1/2, p. 55–66, 1990.
- [74] UEHARA, R.; UNO, Y. Laminar structure of ptolemaic graphs with applications. *Discrete Applied Mathematics*, v. 157, p. 1533–1543, 2009.
- [75] WEST, D. B. *Introduction to Graph Theory*. 2nd. ed. Englewood Cliffs, New Jersey: Prentice Hall, 2001.

# Índice

- árvore, 2
  - geradora, 2
  - raiz  $k$ -folha, 78
  - subárvore, 2
- árvore de cliques, 20
  - valorada  $C_\nu(G)$ , 24
  - propriedades, 23
- adjacência, 1
- algoritmo, 13
  - árvore de cliques, 21
  - ciclo hamiltoniano, 51
  - clique maximais, 17
  - grafo de cliques críticas, 69
  - percurso em largura lexicográfica, 13
  - separadores minimais de vértices, 23
  - sequência gulosa, 33
  - teste de *eep*, 15
- algoritmo de reconhecimento, 7
  - de intervalo próprio, 49
  - estritamente cordal, 65
  - estritamente de intervalo, 82
  - potência de 3-folha, 80
  - $AC$ , 78
- aresta, 1
- asteroide triplo  $AT$ , 32
- caminho, 2
  - euleriano, 2
  - hamiltoniano, 2
- caminho de cliques, 32
- ciclo
  - euleriano, 2
  - hamiltoniano, 2
- classe de grafos, 6
  - caracterização, 7
  - definição, 7
  - hierarquia, 41
  - inclusão, 7
- clique, 2, 16
  - base, 53
  - crítica, 26, 66
  - limítrofe, 17
  - máxima, 2, 16
  - maximal, 16
  - simplicial, 17
- componente, 2
  - conexa, 2
- condição, 5
  - necessária e suficiente, 6
  - de clique, 46
  - de vizinhança, 46
  - necessária, 5
  - suficiente, 5
- condicional, 5
- conjunto
  - $\mathcal{P}$ -conjunto, 4
  - minimal, 4
  - maximal, 4
- conjunto asteroidal, 32
  - cardinalidade máxima  $a(G)$ , 32
- conjunto de adjacência monótona, 14
  - representação por, 14
- conjunto dominante, 2, 70
  - conexo, 74
  - minimal, 70
- conjunto independente, 2
  - maximal, 2
- contagem, 7
- corda, 2
- diâmetro, 2
- distância, 2
- dominação, 70
- enumeração, 7
- esquema de eliminação, 11
  - bicompátível *eeb*, 15
  - forte, 16
  - perfeita *eep*, 12
  - simples, 16
- extensão de vértice único, 37
  - sequência *sext*, 37

- família laminar, 38
- folha, 2
- folhagem mínima  $\ell(G)$ , 20
- gêmeos falsos, 1
- gêmeos verdadeiros, 1
- grafo, 1
  - bloco duplicado, 64
  - split, 53
  - $k$ -conexo, 2
  - acíclico, 2
  - bem dominado, 74
  - biconexo, 2
  - bipartido, 6
  - bipartido completo, 6
  - caterpillar, 6
  - ciclo, 6
  - completo, 6
  - conexo, 2
  - de circuito rígido, 8
  - de subárvores, 8
  - desconexo, 2
  - estrela, 6
  - fortemente cordal, 16
  - hamiltoniano, 7
  - indiferença, 46
  - planar, 7
  - potência, 78
  - regular, 1
  - roda, 6
  - triangularizado, 8
  - ur-estritamente de intervalo, 83
- grafo bloco, 6, 40, 68
- grafo caminho, 6
- grafo cordal, 8
  - unicamente representável, 35
  - laminar, 39
- grafo de cliques  $C(G)$ , 18
- grafo de cliques críticas  $C_c(G)$ , 26
  - grafo estritamente cordal, 66
- grafo de cliques reduzido  $C_r(G)$ , 24
- grafo de interseção, 7
  - da família, 7
- grafo de interseção de cliques  $C(G)$ , 18
- grafo de intervalo, 31
  - unicamente representável, 35
  - indiferença, 46
  - unitário, 46
- grafo de intervalo próprio, 35, 46
  - ciclo hamiltoniano, 50
- grafo estritamente cordal, 41, 64
- grafo estritamente de intervalo, 80
- grafo potência de  $k$ -folha, 78
  - potência de 2-folha, 79
  - potência de 3-folha, 79
- grafo ptolemaico, 36
- grafo split-indiferença, 53
  - contagem, 56
- grafo ur-cordal, 35
- grafo ur-intervalo, 35
- grafo  $AC$ , 76
- grau, 1
  - máximo, 1
- hierarquia, 7
- isomorfismo de grafos, 3
- laço, 1
- matriz de adjacência, 3
- modelo de intervalo, 31
- número de clique  $\omega(G)$ , 2
- número de dominação  $\gamma(G)$ , 71
- número de dominação conexa  $\gamma_c(G)$ , 74
- número de dominação superior  $\Gamma(G)$ , 71
- número de independência  $\beta_0(G)$ , 2
- ordem, 1
- percurso em largura lexicográfica, 12
- problema
  - da contagem, 56
- problema de reconhecimento, 7
- proposição, 5
  - contrapositiva, 5
- propriedade, 4, 23
  - árvore geradora de peso máximo  $PAM$ , 24
  - da interseção de cliques  $PIC$ , 23
  - da subárvore induzida  $PSI$ , 23
  - hereditária, 4
  - sequência de cliques  $RIP$ , 24
  - super-hereditária, 4
- representação, 4
  - por conjuntos de adjacência, 4
  - por matriz de adjacência, 4
- separador, 19
  - $uv$ -separador, 19
  - $uv$ -separador minimal, 19
  - de vértices, 19

- minimal de vértices - *smv*, 19
- minimal do grafo, 19
- multiplicidade, 20
- sequência, 11
- sequência de vértices, 11, 33
  - canônica, 47
  - gulosa, 33, 47
- subgrafo, 2
  - induzido, 2
  - maximal, 2
  - proibido, 29
- tamanho, 1
- teorema de caracterização, 6
- vértice, 1
  - compatível, 16
  - pendente, 1
  - simples, 16
  - simplicial, 2
  - universal, 1
- vizinhança, 1
  - aberta, 1
  - de um subconjunto, 1
  - fechada, 1