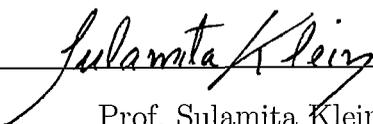


COGRAFOS-(K, L): CARACTERIZAÇÃO E RECONHECIMENTO

Raquel de Souza Francisco Bravo

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

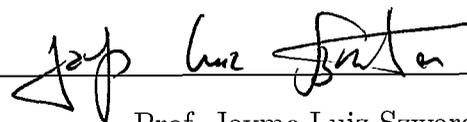
Aprovada por:



Prof. Sulamita Klein, D.Sc.



Prof. Loana Tito Nogueira, D.Sc.



Prof. Jayme Luiz Szwarcfiter Ph.D.



Prof. Nair Maria Maia de Abreu, D.Sc.

RIO DE JANEIRO, RJ - BRASIL

MARÇO DE 2006

BRAVO, RAQUEL DE SOUZA
FRANCISCO

Cografos- (k, l) : Caracterização e
Reconhecimento [Rio de Janeiro] 2006

XIII, 98 p. 29,7 cm (COPPE/UFRJ,
M.Sc., Engenharia de Sistemas e Computação,
2006)

Dissertação – Universidade Federal do Rio
de Janeiro, COPPE

1 - Cografos

2 - Reconhecimento de cografos

3 - Cografo- (k, l)

I. COPPE/UFRJ II. Título (série)

Existe um só sucesso: o de podermos viver a nossa vida como desejamos.

(Christopher Morley)

Agradecimentos

Antes de tudo, agradeço a Deus, criador e dono de toda a ciência, e mesmo assim se preocupa conosco, meras criaturas, mas incondicionalmente amadas por Ele e chamadas de *filhos*. A Ele, toda honra e glória.

A meu querido esposo e amigo Renato, maior responsável pela minha decisão em fazer este mestrado e seguir carreira acadêmica. Obrigada por muitas vezes ter me ajudado nas tarefas domésticas para que eu pudesse estudar. Enfim, por ser o melhor marido que alguma mulher poderia sequer imaginar em ter e me fazer a mais feliz das esposas.

Aos meus pais, pelo incentivo, apoio, exemplo, amor e carinho com que me educaram, nunca medindo esforços para me proporcionar o melhor que poderiam em todos os aspectos da vida. Serei sempre grata a vocês. Agradeço, também, ao meu irmão, por todo o carinho. Eu amo vocês!

Às professoras Sulamita Klein e Loana Nogueira, muito mais que orientadoras, amigas muito dedicadas, por fornecerem uma direção a este trabalho, pela paciência de ouvirem minhas demonstrações nem sempre muito claras, pelas dicas e conselhos sempre pertinentes.

Aos Profs. Jayme Szwarcfiter e Nair Abreu, por aceitarem fazer parte desta banca, reunindo valor inestimável a este trabalho. Ao Prof. Jayme, em especial, por ter me acolhido nesta área de pesquisa.

Aos meus amigos Danilo e Rodrigo Hausen, pela paciência e dedicação, pois sempre recorria a eles para pedir ajuda relacionada ao Latex.

À minha amiga Priscila Petito, por ter me ajudado a fazer as apresentações que ocorreram no decorrer do mestrado e por estar sempre presente nas idas e vindas a UFRJ.

Aos professores da COPPE-Sistemas e aos colegas de pós-graduação, obrigada pela amizade e coleguismo.

A CAPES, pelo suporte financeiro durante todo o tempo de pós-graduação.

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

COGRAFOS- (K, L) : CARACTERIZAÇÃO E RECONHECIMENTO

Raquel de Souza Francisco Bravo

Março/2006

Orientador: Sulamita Klein

Programa: Engenharia de Sistemas e Computação

Neste trabalho, consideramos o problema de particionar um grafo em k conjuntos independentes e l cliques, conhecido como o problema da (k, l) -partição ou o problema dos grafos- (k, l) , o qual foi introduzido por Brandstädt, e generalizado por Feder, Hell, Klein e Motwani como o problema da M -partição. Brandstädt provou que, dado um grafo G , é NP-completo decidir se G é um grafo- (k, l) para $k \geq 3$ ou $l \geq 3$. Em particular, consideramos uma subclasse de grafos perfeitos: os cografos, que consistem de grafos sem P_4 (caminho induzido com 4 vértices), e apresentamos uma caracterização dos cografos- (k, l) em termos de subgrafos proibidos, i. e., obstruções, além de um algoritmo linear para reconhecer tal classe. O algoritmo linear baseia-se no fato de que cliques máximas e conjuntos independentes máximos podem ser encontrados em tempo linear quando restritos a cografos.

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

(K, L) -COGRAPHS: CHARACTERIZATION AND RECOGNITION

Raquel de Souza Francisco Bravo

March/2006

Advisor: Sulamita Klein

Department: Systems Engineering and Computer Science

In this work we consider the problem of partitioning a graph into k independent sets and l cliques, known as the (k, l) -partition problem, which was introduced by Brandstädt, and generalized by Feder, Hell, Klein and Motwani as the M-partition problem. Brandstädt proved that, given a graph G , it is NP-complete to decide if G is a (k, l) -graph for $k \geq 3$ or $l \geq 3$. In particular, we consider a subclass of perfect graphs: the cographs, which consist of graphs without P_4 (induced path with 4 vertices), and we present a characterization of the (k, l) -cographs in terms of forbidden subgraphs, i. e., obstructions. Moreover, we give a linear time algorithm to recognize such class, which is based on the fact that maximum cliques and maximum independent sets can be found in linear time for cographs. .

Conteúdo

1	Introdução	1
1.0.1	Definição do Problema	2
1.1	Organização do trabalho	4
1.2	Grafos	5
1.3	Complexidade de Algoritmos	9
2	Cografos	12
2.1	Cografos e suas propriedades estruturais	13
2.1.1	Decomposição modular	14
2.1.2	Co-árvore	15
2.2	Caracterização de cografos	23
3	Reconhecimento de cografos	31
3.1	Algoritmo linear	32
3.2	Complexidade	60
3.3	Alguns problemas algorítmicos	60
4	Cografo-(k, l)	68
4.1	Grafos- (k, l)	69
4.2	Caracterização dos cografos- (k, l)	70
4.2.1	Uma recente caracterização de cografos	75

4.2.2	Cografos-(1, 2) e Cografos-(2, 2)	79
4.2.3	Caracterização dos cografos- (k, l)	83
4.3	Reconhecimento dos cografos- (k, l)	85
4.3.1	Algoritmo de reconhecimento dos cografos- (k, l)	85
4.3.2	Corretude	85
4.3.3	Complexidade	87
5	Conclusão	88
5.0.4	Trabalhos Futuros	89
A	Apêndice do capítulo 2	91

Lista de Figuras

2.1	Exemplo do cografo G e sua respectiva co-árvore	15
2.2	Cografos com no máximo 4 vértices	16
2.3	Os possíveis grafos conexos com 4 vértices a menos de isomorfismos	17
2.4	O complemento dos possíveis grafos conexos com 4 vértices a menos de isomorfismos	17
2.5	A co-árvore de G rotulada	20
2.6	Grafo G_1 desconexo e sua respectiva co-árvore	20
2.7	Ilustração da árvore T e sua sub-árvore T' construída a partir da remoção do vértice y de T para o item (i)	22
2.8	Ilustração da árvore T e sua sub-árvore T' construída com a remoção do vértice y de T para o caso (a) do item (ii)	22
2.9	Ilustração da árvore T e sua subárvore T' construída com a remoção do vértice y de T para o caso (b) do item (ii)	23
2.10	$G[u, v, x, y]$	29
2.11	Caminho com 4 vértices	30
3.1	Um cografo e sua co-árvore correspondente	34
3.2	Ilustração do P_4 induzido pelos vértices b, c, x e d , referente ao caso 1.1 do item (i).	37

3.3	Ilustração do P_4 induzido pelos vértices b, c, a e x , referente ao caso 1.2 do item (i)	38
3.4	Ilustração do P_4 formado pelos vértices b, a, x e c , referente ao caso 2 do item (i)	38
3.5	Ilustração do P_4 formado pelos vértices b, c, a e x , referente ao caso 3 do item (i)	39
3.6	Ilustração do P_4 formado pelos vértices b, a, c e x , referente ao caso 4 do item (i)	40
3.7	Ilustração do P_4 formado pelos vértices x, a, β e b , referente ao caso 2.1 do item (ii)	42
3.8	Ilustração do P_4 formado pelos vértices b, c, a e x , referente ao caso 2.1 do item (ii)	42
3.9	Ilustração do P_4 formado pelos vértices d, x, a e c , referente ao caso 3.1 do item (ii)	43
3.10	Ilustração do P_4 formado pelos vértices e, x, a e b , referente ao caso 3.2.1 do item (ii)	44
3.11	Ilustração do P_4 formado pelos vértices e, d, a e x , referente ao caso 3.2.2 do item (ii)	44
3.12	Ilustração do P_4 formado pelos vértices b, a, x e d , referente ao caso 2.1 do item (iii)	45
3.13	Ilustração do P_4 formado pelos vértices e, x, a e b , referente ao caso 2.2.1 do item (iii)	46
3.14	Ilustração do P_4 formado pelos vértices e, c, a e x , referente ao caso 2.2.2 do item (iii)	46
3.15	Ilustração do P_4 formado pelos vértices b, c, x e d , referente ao caso 1.1 do item (v)	48

3.16	Ilustração do P_4 formado pelos vértices b, c, a e x , referente ao caso 1.2 do item (v)	48
3.17	Ilustração do P_4 induzido pelos vértices b, e, a e x , referente ao caso 2 do item (v)	48
3.18	Ilustração do P_4 formado pelos vértices b, a, x e d , referente ao caso 1 do item (vi)	49
3.19	Ilustração do P_4 formado pelos vértices b, a, x e c , referente ao caso 2.1 do item (vi)	50
3.20	Ilustração do P_4 formado pelos vértices c, e, a e x , referente ao caso 2.2 do item (vi)	50
3.21	Ilustração do caso 1.1 da demonstração do teorema 3.1	51
3.22	Ilustração do caso 1.2 da demonstração do teorema 3.1	51
3.23	Ilustração do caso 1.3 da demonstração do teorema 3.1	51
3.24	Ilustração do caso 2.1.1 da demonstração do teorema 3.1	52
3.25	Ilustração do caso 2.1.2 da demonstração do teorema 3.1	52
3.26	Ilustração do caso 2.1.3 da demonstração do teorema 3.1	53
3.27	Ilustração do caso 2.2.1 da demonstração do teorema 3.1	53
3.28	Ilustração do caso 2.2.2 da demonstração do teorema 3.1	54
3.29	Ilustração do caso 2.2.3 da demonstração do teorema 3.1	54
3.30	O cografo G e sua co-árvore T . Os símbolos $*$ e $(*)$ referem-se ao procedimento <i>Marcar</i>	57
3.31	Cografo $G' = G + x$ e sua co-árvore T'	58
3.32	O grafo $G' + z$ não é um cografo (os vértices b, a, z e c formam um P_4)	59
3.33	Cografo G e sua respectiva co-árvore T	65
3.34	Resultados dos algoritmos para as cliques maximais e máximas: (a) Algoritmo 3.4; (b) Algoritmo 3.5; (c) Algoritmo 3.6;	66

3.35	Resultados dos algoritmos para os kernels e conjuntos independentes máximos: (a) Algoritmo 3.4; (b) Algoritmo 3.5; (c) Algoritmo 3.6;	67
4.1	2^*K_2	71
4.2	$G[K_k] \subseteq G[V \setminus S^*]$	73
4.3	Ilustração do caso (ii), isto é, s_k é não-adjacente ao vértice s_1^*	74
4.4	Ilustração da demonstração do teorema 4.8, implicação (i) \Rightarrow (iii).	77
4.5	P_4 induzido pelos vértices x, y, z e w de $V(G)$	78
4.6	3^*K_2 , onde as arestas ausentes são opcionais	79
4.7	3^*K_2 particionado em dois conjuntos independentes, S^* e $V \setminus S^*$	80
4.8	3^*K_3 , onde as arestas ausentes são opcionais	81
4.9	$(l + 1)^*K_{k+1}$, onde as arestas ausentes são opcionais	83
4.10	Uma partição em k conjuntos independentes	85
A.1	Grafo G e sua respectiva árvore de decomposição modular . .	94

Capítulo 1

Introdução

O estudo em grafos é um tópico muito importante em Ciência da Computação. A importância do assunto tanto no campo teórico, implicando no crescimento do número de publicações e artigos especializados no assunto, como na prática, em computação e na solução de problemas reais, se reflete na grande quantidade de problemas existentes na área, ainda em processo de estudo.

Um dos objetivos nesta área consiste em resolver, computacionalmente, problemas reais ou não, que possam ser modelados como grafos. De maneira mais simples e clara, ao dizer “resolver computacionalmente” estamos interessados não apenas em resolver o problema através de um computador digital mas, além disso, produzir algoritmos que resolvam o problema e que possam ser executados em tempo razoável (mais adiante explicamos esta noção de tempo razoável). No entanto, para muitos problemas em grafos, não são conhecidos algoritmos para resolver tais problemas neste tempo razoável.

Uma maneira de solucionar os problemas difíceis em grafos é restringí-los a classes especiais de grafos. Nesta dissertação, estudamos as propriedades de uma importante subclasse dos grafos perfeitos - os cografos - e usamos

suas propriedades estruturais para desenvolver um algoritmo linear para um problema no contexto de partição em grafos que é *NP*-completo para grafos em geral.

1.0.1 Definição do Problema

Neste trabalho consideramos um problema de partição de grafos. Estudo esse que tem despertado muito interesse devido às pesquisas em grafos perfeitos [17] e também pela procura de algoritmos eficientes para o reconhecimento e estudo de problemas de otimização de determinadas classes de grafos. Muitos desses problemas de partição de grafos tem como objetivo particionar o conjunto dos vértices de um grafo em subconjuntos V_1, V_2, \dots, V_k onde $V_1 \cup V_2 \cup \dots \cup V_k = V$ e $V_i \cap V_j = \emptyset, i \neq j, 1 \leq i \leq k$ e $1 \leq j \leq k$, exigindo-se, porém, algumas propriedades sobre esses subconjuntos de vértices.

Estas propriedades podem ser *internas* (relações entre os vértices de um determinado subconjunto) ou *externas* (relações entre os subconjuntos de vértices). Dentre as propriedades internas, podemos citar como exemplo, aquela que exige que os vértices de cada subconjunto V_i sejam completamente adjacentes (V_i é uma clique) ou completamente não-adjacentes (V_i é um conjunto independente). Nas propriedades externas, por sua vez, as exigências são feitas sobre os pares (V_i, V_j) , isto é, V_i e V_j podem ser completamente adjacentes (todos os vértices de V_i são adjacentes a todos os vértices de V_j) ou não-adjacentes (todos os vértices de V_i são não-adjacentes a todos os vértices de V_j).

Como exemplo, podemos citar um dos problemas mais famosos que se insere neste contexto, o problema da k -coloração, no qual deseja-se verificar se é possível particionar os vértices de um dado grafo em k conjuntos inde-

pendentes V_1, \dots, V_k (sem restrições externas). Sabe-se que esse problema é polinomial para $k \leq 2$ e NP-completo para $k \geq 3$ [25].

Outro problema bastante conhecido de particionamento de grafos consiste em verificar se um dado grafo G é split, ou equivalentemente, verificar se o conjunto dos vértices de G pode ser particionado em dois subconjuntos, dos quais um é um conjunto independente e o outro é uma clique.

Provou-se que o reconhecimento de grafos split pode ser realizado em tempo linear [17]. Recentemente, uma generalização de grafos split foi proposta por Brandstädt, que definiu uma nova classe de grafos, a classe dos *grafos- (k, l)* , a qual também chamamos de grafos split generalizados, como sendo aquela formada pelos grafos cujo conjunto de vértices pode ser particionado em k conjuntos independentes e l cliques. Brandstädt em [3, 4] considerou, em particular, as classes de grafos- $(2, 1)$, grafos- $(1, 2)$ e grafos- $(2, 2)$, apresentando algoritmos polinomiais para reconhecê-las. Feder *et al.* em [14] também apresentaram algoritmos polinomiais para o reconhecimento destas classes que surgiram como sub-produto de algoritmos de partição em subgrafos densos e esparsos. Por outro lado, Brandstädt [3] provou que o problema de reconhecer grafos- (k, l) , para $k \geq 3$ ou $l \geq 3$, é um problema NP-completo. Como exemplo, podemos considerar a classe dos grafos- $(k, 0)$, que corresponde ao problema de reconhecer se um dado grafo é k -colorível (como dito anteriormente o reconhecimento desta classe é NP-completo para $k \geq 3$).

Sendo NP-completo o reconhecimento de grafos- (k, l) , para $k \geq 3$ ou $l \geq 3$, restringimos este problema à classe dos cografos, i.e., dado um cografo G e dois inteiros não-negativos k e l , resolvemos o problema de reconhecer se G é um grafo- (k, l) .

Mais especificamente, caracterizamos os cografos- (k, l) através de estruturas contidas no mesmo e provamos o seguinte resultado:

Teorema 1.1. *(Caracterização) Um grafo G é um grafo- (k, l) se e somente se não contém nenhum dos cografos da família $(l + 1)^*K_{k+1}$ como subgrafo induzido. (Mais a frente iremos definir quem é a família $(l + 1)^*K_{k+1}$)*

Vale mencionar que tal resultado é uma generalização do resultado de [13], onde são apresentadas caracterizações dos cografos- $(1, 2)$ e dos cografos- $(2, 2)$. Além disso, outros trabalhos já foram desenvolvidos no contexto de partição em grafos e de partição em grafos com listas para grafos cordais [21, 22, 23].

A próxima seção descreve como este trabalho está organizado.

1.1 Organização do trabalho

Este trabalho está dividido em cinco capítulos. No capítulo 1, apresentamos os objetivos, as definições e notações utilizadas no decorrer desta dissertação. No capítulo 2, introduzimos a classe dos cografos, apresentamos algumas de suas propriedades e uma prova detalhada do teorema de caracterização dos cografos. No capítulo 3, descrevemos um algoritmo em tempo linear para reconhecer tal classe. A contribuição principal desta dissertação é abordada no capítulo 4, onde definimos os grafos- (k, l) e apresentamos uma caracterização e um algoritmo em tempo linear para o reconhecimento dos cografos- (k, l) . Finalmente, no capítulo 5, fornecemos as conclusões desta dissertação e algumas propostas de continuidade do trabalho.

A seguir, estabelecemos algumas definições e notações que serão utilizadas ao longo deste trabalho.

1.2 Grafos

Um *grafo simples* é um par ordenado $G = (V, E)$, onde V é um conjunto finito não-vazio de *vértices*, denotado por $V(G)$ e E é um conjunto de pares não-ordenados de vértices distintos, chamados *arestas*, e denotado por $E(G)$. Uma aresta entre dois vértices u e v de $V(G)$ será denotada por (u, v) . Utilizaremos a notação $n = |V(G)|$ e $m = |E(G)|$ para denotarmos a cardinalidade de $V(G)$ e $E(G)$, respectivamente. No decorrer deste trabalho, iremos utilizar a denominação grafos para denotar o que definimos como grafos simples.

Um grafo G é dito *trivial* se $|V(G)| = 1$, isto é G possui um único vértice.

Um vértice u é *adjacente* a outro vértice v em G se $(u, v) \in E(G)$. Neste caso, dizemos que u e v são *vizinhos* em G , e que a aresta $e = (u, v)$ é *incidente* a u e a v , ou que tem *extremos* u e v . Caso $(u, v) \notin E(G)$, dizemos que u e v são não-adjacentes.

Denotamos por $N(u)$ o conjunto de vértices adjacentes a u em G e tal conjunto é chamado de *vizinhança* de u , e por $N[u]$ o conjunto $N(u) \cup \{u\}$ e chamado de *vizinhança fechada* de u . Um vértice u é dito *universal* quando $N(u) = V(G) - \{u\}$. Os vértices u, v são *gêmeos* se $N(u) - \{u, v\} = N(v) - \{u, v\}$. Dois vértices gêmeos são ditos *verdadeiros* se eles são adjacentes, isto é, $(u, v) \in E$ e, são ditos *falsos*, caso contrário. Na literatura, irmãos fortes e fracos são, também, chamados de gêmeos verdadeiros e falsos, respectivamente.

O *grau* de um vértice $v \in V(G)$, denotado por $d_G(v)$, é o número de arestas incidentes ao vértice v .

Um grafo H é um *subgrafo* de um grafo G se $V(H) \subseteq V(G)$ e $E(H) \subseteq E(G)$. Dado um conjunto de vértices $Y \subseteq V(G)$, $Y \neq \emptyset$, o *subgrafo de G induzido* por Y , denotado por $G[Y]$ é o subgrafo H de G tal que $V(H) = Y$ e

$E(H)$ é o conjunto das arestas de G que têm ambos os extremos em Y . Neste trabalho consideremos apenas subgrafos induzidos e portanto, quando nos referirmos a subgrafo estaremos assumindo implicitamente que são subgrafos induzidos.

Um *passeio* em um grafo G é uma sequência de vértices, cada qual adjacente ao vértice que lhe sucede na sequência.

Um *caminho* num grafo G é um passeio $P = v_1, v_2, \dots, v_k$, onde os v_i 's são vértices (dois a dois distintos). Uma *corda* em P é uma aresta que liga dois vértices não-consecutivos de P . Um *caminho induzido* é um caminho sem cordas, e denotado por P_k o caminho induzido por k vértices. Dizemos que um grafo é *livre de P_k* quando não contém um P_k como subgrafo.

Um passeio v_1, \dots, v_k, v_{k+1} , é denominado *ciclo* quando v_1, \dots, v_k for um caminho, $k \geq 3$ e $v_1 = v_{k+1}$.

Um grafo G é dito *cíclico* quando G contém um ciclo como subgrafo induzido. Caso contrário, dizemos que G é *acíclico*.

Um conjunto S é *maximal (minimal)* em relação a uma determinada propriedade P se S satisfaz P , e todo conjunto S' que contém propriamente S (que está contido propriamente em S) não satisfaz P .

Definimos a *união* de dois grafos $G_1 = (V_1, E_1)$ e $G_2 = (V_2, E_2)$ como sendo o grafo $G = G_1 \cup G_2 = (V_1 \cup V_2, E_1 \cup E_2)$. Da mesma forma, definimos a operação "*join*" de G_1 e G_2 , denotado por $G_1 \oplus G_2 = (V_1 \cup V_2, E_1 \cup E_2 \cup \{(x, y) : x \in V_1 \text{ e } y \in V_2\})$.

Um grafo G é *completo* se quaisquer dois vértices distintos de G são adjacentes. Denotamos por K_n o grafo completo com n vértices.

Um conjunto de vértices C de um grafo G é uma *clique* se $G[C]$ é um grafo completo. Denotamos por $\omega(G)$ o *tamanho da clique máxima*, isto é:

$$\omega(G) = \max \{ |V'| \mid V' \subseteq V \text{ e } V' \text{ é uma clique de } G \}$$

Um conjunto de vértices I de um grafo G é um *conjunto independente* se $G[I]$ é um grafo sem arestas. Definimos por $\alpha(G)$ o tamanho do *conjunto independente máximo*, isto é:

$$\alpha(G) = \max \{ |V'| \mid V' \subseteq V \text{ e } V' \text{ é um conjunto independente de } G \}$$

Uma *coloração* de um grafo G é uma partição de $V(G)$ onde cada classe da partição é um conjunto independente. Uma *k-coloração* é uma partição de $V(G)$ em k classes. O *número cromático* de G , denotado por $\chi(G)$, é o menor k para o qual existe uma k -coloração de G . Neste caso, dizemos que o grafo G é *k-cromático* ou *k-colorível*.

Um grafo G é *conexo* se para todo par de vértices distintos v e w de G existe um caminho de v a w . Caso contrário, G é dito *desconexo*. Um *componente conexo* de G é um subgrafo maximal conexo de G .

Sejam $v, w \in V(G)$. A *distância* entre v e w em G , denotada por $d_G(v, w)$, é o comprimento do menor caminho entre v e w em G .

A *excentricidade* de um vértice $v \in V(G)$, denotada por $ex_G(v)$, é o valor:

$$ex_G(v) = \max_{w \in V(G)} d_G(v, w)$$

O *diâmetro* de um grafo é o valor $\max_{v \in V(G)} ex(v)$.

Uma *árvore* T é um grafo acíclico e conexo. Uma árvore T é denominada *enraizada* quando algum vértice $v \in V(T)$ é escolhido como especial. Este vértice é então chamado de *raiz* da árvore.

Sejam v, w dois vértices de uma árvore enraizada T de raiz R . Se v pertence ao caminho de R a w em T , então denominamos v *ancestral* de w e w *descendente* de v . Em particular, se $(v, w) \in E(T)$ então v é *pai* de w ,

denotada por $\text{pai}(w)$, sendo w o filho de v , denotada por $\text{filho}(v)$. Sejam z o pai de v e v o pai de w , denominamos z o avô de w , denotada por $\text{avô}(w)$. Dois vértices que possuem o mesmo pai são chamados de irmãos. A raiz R de uma árvore não possui pai, enquanto que todo vértice $v \neq R$ possui um único pai. Uma folha é um vértice que não possui filhos. Denomina-se *nível* de um vértice v de uma árvore a distância da raiz R a v .

O *complemento* de um grafo G , denotado por \bar{G} , é o grafo que possui o mesmo conjunto de vértices de G e tal que dois vértices são adjacentes em \bar{G} se e somente se são não-adjacentes em G .

Um *kernel* de um grafo é um conjunto independente maximal. Observe que $S \subseteq V$ é um kernel se e somente se S é uma clique maximal em \bar{G} .

Dizemos que uma classe de grafos C é *hereditária* se todo subgrafo induzido de um grafo em C também pertence a C .

Um grafo G é dito *bipartido* quando seu conjunto de vértices pode ser particionado em dois subconjuntos V_1, V_2 , tais que toda aresta de G une um vértice de V_1 a outro de V_2 , isto é, podemos particionar $V(G)$ em dois conjuntos independentes. Um grafo é dito *bipartido completo* se é bipartido e possui uma aresta para cada par de vértices v_1, v_2 , sendo $v_1 \in V_1$ e $v_2 \in V_2$. Denotamos por $K_{n,m}$ o grafo bipartido completo, onde $|V_1| = n$ e $|V_2| = m$. Um grafo é *k-partido* quando seu conjunto de vértices pode ser particionado em k conjuntos independentes. Um grafo é *split* se o seu conjunto de vértices pode ser particionado em um conjunto independente e uma clique.

Um *emparelhamento* $M \subseteq E(G)$ é um conjunto de arestas de G , duas a duas não-adjacentes. Um vértice $v \in V(G)$ é *saturado* por um emparelhamento $M \subseteq E(G)$, quando v for incidente a alguma aresta de M . Um emparelhamento é *perfeito* quando todos os seus vértices forem saturados.

Uma *cobertura* k de um grafo G é um subconjunto de vértices que é incidente a todas as arestas.

Um grafo G é *perfeito* quando para todo subgrafo induzido H de G temos que o número cromático de H é igual ao tamanho da clique máxima de H , isto é, $\chi(H) = \omega(H)$. Berge [2] em 1960 conjecturou que um grafo G é perfeito se e somente se ele é berge (G e \overline{G} não admitem ciclo ímpar induzido maior ou igual a 5) e ficou conhecida como a Conjectura forte dos grafos perfeitos. Tal conjectura ficou em aberto por 35 anos, tendo sido recentemente provada por Chudnovsky, Robertson, Seymour e Thomas [9].

1.3 Complexidade de Algoritmos

Uma área da Ciência da Computação que teve um grande e rápido crescimento no últimos 25 anos é o campo de *Algoritmos e Teoria da Complexidade* - em particular, o desenvolvimento e análise de algoritmos computacionais. O objetivo de pesquisa nesta área é o de estudar a natureza dos problemas que podem ser resolvidos através de um computador digital, para fornecer soluções para problemas resolvíveis, assim como classificá-los em categorias dependendo do seu grau de dificuldade ou intratabilidade.

Formalmente, um problema algorítmico π consiste de um conjunto D de todas as possíveis entradas para o problema, chamado *conjunto de instâncias*, e de uma *questão* Q sobre estas instâncias. Resolver um problema algorítmico é desenvolver um algoritmo cuja entrada é uma instância do problema e cuja saída é uma resposta à questão do problema.

Um problema é dito de *decisão* quando a questão exige uma resposta do tipo SIM ou NÃO. Como exemplo, seja π o seguinte problema: “Dado um grafo G , reconhecer se G é um cografo.” O conjunto de instâncias de π é

obviamente o conjunto de todos os grafos. O problema π pode ser assim esquematizado:

Instância genérica de π : um grafo G .

Questão: G é cografo?

Fica evidente que o problema π acima é um problema de decisão, em particular, um *problema de reconhecimento*. Resolver π significa elaborar um algoritmo de reconhecimento de cografos.

Dizemos que um algoritmo é *polinomial* quando sua complexidade de tempo (medida do número de passos que o algoritmo efetua) é uma função polinomial no tamanho da sua entrada. Os problemas de decisão para os quais existem algoritmos polinomiais constituem a classe P. Tais problemas são chamados *polinomiais*.

Um problema de decisão é *não-determinístico polinomial* quando qualquer instância que produz resposta SIM possui um *certificado* sucinto, isto é, uma comprovação de que a resposta SIM é de fato verificável em tempo polinomial no tamanho da instância. Esta classe de problemas de decisão é a classe NP.

A classe Co-NP é formada pelos problemas que possuem um certificado sucinto para as instâncias que produzem resposta NÃO.

Sejam $\pi_1(D_1, Q_1)$ e $\pi_2(D_2, Q_2)$ dois problemas de decisão. Uma *transformação* ou *redução polinomial* de π_1 em π_2 é uma função $f : D_1 \rightarrow D_2$ tal que as seguintes condições são satisfeitas:

- f pode ser calculada em tempo polinomial;
- para toda instância $I \in D_1$, tem-se que I produz resposta SIM para π_1 se e somente se $f(I)$ produz resposta SIM para π_2 .

Um problema de decisão π pertence à classe *NP-completo* quando as seguintes condições são satisfeitas:

- $\pi \in \text{NP}$;
- para todo problema $\pi' \in \text{NP}$ existe uma transformação polinomial de π' em π .

Um problema pertencente à classe *NP-completo* é chamado *NP-completo*. Para provar que um certo problema π é NP-completo, basta mostrar que $\pi \in \text{NP}$ e que existe uma transformação de um problema NP-completo π' em π .

Analogamente, prova-se que um problema de decisão π pertence à classe *Co-NP-completo* (e, neste caso, π é dito *Co-NP-completo*) quando $\pi \in \text{Co-NP}$ e existe um problema π' (Co-)NP-completo tal que:

- se π' é NP-completo, existe uma função f que pode ser calculada em tempo polinomial tal que para toda instância I' de π' , tem-se que I' produz SIM para π' se e somente se $I = f(I')$ produz NÃO para π ;
- se π' é Co-NP-completo, existe uma função f que pode ser calculada em tempo polinomial tal que para toda instância I' de π' , tem-se que I' produz NÃO para π' se e somente se $I = f(I')$ produz NÃO para π .

Como fonte de referências para esta seção, indicamos [16, 37].

Capítulo 2

Cografos

Os cografos surgiram em diversas áreas da Matemática, sendo objeto de pesquisa de vários cientistas independentemente. Esta independência fez com que vários sinônimos para o termo cografo surgissem, dentre os quais podemos citar: grafos D^* [24], grafos sem P_4 , grafos HD [36] e grafos redutíveis por complemento [10].

Na década de 70, H. Lerchs [26, 27] introduziu o termo cografo, tal como conhecemos hoje. H. Lerchs estudou suas propriedades estruturais e algorítmicas, definindo assim a classe dos cografos. Como uma extensão deste estudo, L. Stewart [34] desenvolveu um algoritmo com complexidade $O(n^2)$ para reconhecer tal classe. Posteriormente, um algoritmo mais eficiente, com complexidade linear, foi introduzido por Corneil *et al.* [11]. Outros algoritmos para reconhecer cografos com complexidade linear são conhecidos [20], [8]. Em particular, Habib e Paul [20] descreveram um algoritmo de dois passos: primeiramente, uma técnica de refinamento para produzir uma ordenação especial dos vértices é usada e, na segunda fase, executa-se um simples teste para verificar se um dado grafo é um cografo, utilizando tal ordenação.

Um cografo pode ser obtido a partir de um conjunto de grafos triviais, através de um número finito de operações de união e “join”. Ele pode ser representado de forma única através de uma árvore especial, que recebe o nome de co-árvore. Considerando-se tal representação, obtém-se diversas propriedades e características de um cografo, o que auxilia na solução de problemas considerados difíceis em outras classes de grafos.

Este capítulo está organizado da seguinte forma: na seção 2.1 definimos formalmente os cografos e introduzimos propriedades estruturais dos cografos. Na seção 2.2 apresentamos uma caracterização clássica de cografos dada por Corneil, Lerchs e Burlingham [10] e uma demonstração detalhada desta caracterização.

2.1 Cografos e suas propriedades estruturais

Definição 2.1. *Os cografos são definidos recursivamente [10] através da seguinte definição formal:*

- (i) *O grafo trivial (K_1) é um cografo;*
- (ii) *Se G_1, \dots, G_k são cografos, então a sua união $G_1 \cup G_2 \cup \dots \cup G_k$ também é cografo;*
- (iii) *Se G é cografo, então \overline{G} também é cografo.*

Um resultado de suma importância, provado em [26], afirma que os cografos podem ser representados por uma árvore de decomposição única. Esta representação é a chave para o reconhecimento linear dos cografos [11] e é abordada brevemente neste capítulo e com mais detalhes no apêndice A.

2.1.1 Decomposição modular

A decomposição modular facilita a resolução de diversos problemas em certas classes de grafos, e algoritmos para encontrá-la tem uma vasta história [29].

Definição 2.2. [1] Um módulo de um grafo G é um subconjunto M de vértices de $V(G)$ tal que cada vértice de $V(G) \setminus M$ ou é adjacente a todo vértice de M ou a nenhum vértice de M .

Definição 2.3. Um módulo M é dito trivial se $|M| = 1$ ou $M = V(G)$.

Definição 2.4. Um grafo G é dito primal (ou primo) se G possui apenas módulos triviais.

A decomposição modular é um processo para decompor um grafo. Em cada etapa, o subgrafo que está sendo decomposto é um módulo do grafo original. Cada um destes subgrafos é decomposto recursivamente. O processo continua até que todos os subgrafos que estão sendo decompostos contenham um único vértice.

Através da decomposição modular de um grafo G é possível construir uma árvore correspondente a G cuja estrutura é de grande importância para algumas aplicações. Dentre estas aplicações, citamos a construção de algoritmos eficientes para reconhecer diversas classes de grafos. Em particular, estaremos interessados em mostrar como a decomposição modular pode ser utilizada para reconhecer os cografos.

A decomposição modular de um grafo G particiona $V(G)$ em módulos, os quais podem ser classificados em *paralelo*, *serial* e *vizinhaça*. Tal classificação é feita de acordo com a conectividade de G e \overline{G} .

A decomposição modular de um grafo não trivial G é usado recursivamente para definir a árvore de decomposição modular única. O módulo M

é *paralelo* se $G[M]$ é desconexo, M é *serial* se $\overline{G[M]}$ é desconexo e M é *vizinhança* se $G[M]$ e $\overline{G[M]}$ são ambos conexos.

Árvore de decomposição

De acordo com o citado acima, a árvore de decomposição possui os módulos paralelo, serial e vizinhança. Os vértices do grafo G correspondem as folhas da árvore de decomposição. Cada módulo M de G está associado a um nó da árvore, cujas folhas descendentes de M correspondem aos vértices pertencentes ao módulo M .

No decorrer desta dissertação estaremos interessados em um caso particular da árvore de decomposição modular: o caso em que a árvore contém somente os módulos paralelo e serial. Vejamos na figura 2.1 uma representação de um grafo G (cografo), onde sua decomposição modular possui apenas os módulos paralelo e serial.

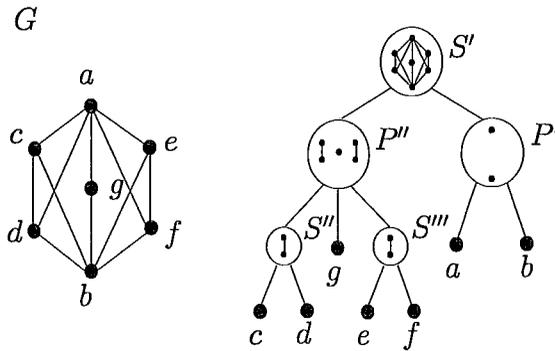


Figura 2.1: Exemplo do cografo G e sua respectiva co-árvore

2.1.2 Co-árvore

Definimos a co-árvore T de um grafo G como sendo a árvore de decomposição que contém apenas os módulos serial e paralelo.

O teorema dado a seguir caracteriza os cografos através de sua árvore de decomposição.

Teorema 2.5. [10] *Dado um grafo G , as seguintes afirmações são equivalentes:*

(i) G é cografo;

(ii) G não contém um P_4 como subgrafo;

(iii) O complemento de todo subgrafo conexo não trivial de G é desconexo.

Demonstração. (i) \Rightarrow (ii) A definição de cografos nos fornece uma forma construtiva de gerar todos os cografos, através das operações de união e complemento, a partir do K_1 .

Em particular, a figura 2.2 representa a família dos cografos com no máximo quatro vértices.

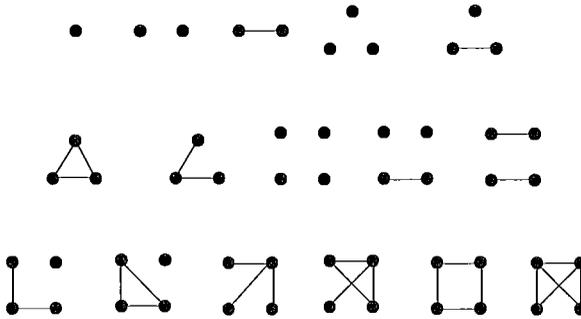


Figura 2.2: Cografos com no máximo 4 vértices

Observe que como o P_4 não pertence à família dos cografos com menos de cinco vértices então os demais cografos com mais de 4 vértices também não conterão o P_4 como subgrafo, já que estes são gerados pelas operações de união e complemento, e $P_4 = \overline{P_4}$.

Portanto, G não contém um P_4 como subgrafo.

(ii) \Rightarrow (iii) Se G tem menos de 4 vértices, a prova segue facilmente.

Se $n = 4$ com G e \overline{G} conexos, então G e \overline{G} são isomorfos ao P_4 , vide figuras 2.3 e 2.4 (Observe que G_1 é o único grafo da figura 2.3 cujo complemento também é conexo).

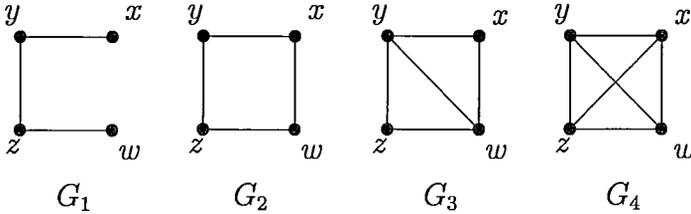


Figura 2.3: Os possíveis grafos conexos com 4 vértices a menos de isomorfismos

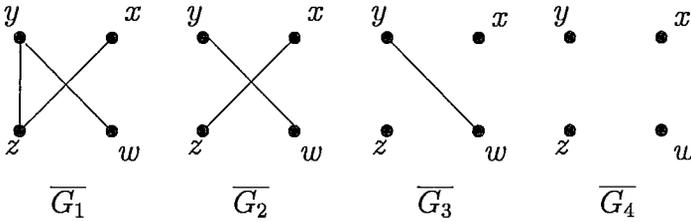


Figura 2.4: O complemento dos possíveis grafos conexos com 4 vértices a menos de isomorfismos

A prova será feita por contrapositiva ($\neg iii \Rightarrow \neg ii$).

Suponha $n \geq 5$.

Seja G' um subgrafo conexo de G , com pelo menos 5 vértices, cujo complemento é conexo.

Seja $e = (a, b)$ uma aresta de G' . Então, $(a, b) \notin E(\overline{G'})$. No entanto, $\overline{G'}$ também é conexo, logo, existe um caminho de a a b em $\overline{G'}$. Seja P um caminho mínimo de a a b em $\overline{G'}$.

Se $|P| > 3$ então $\overline{G'}$ contém um P_4 induzido, finalizando, assim, a prova.

Se $|P| = 3$ então considere $P = (a, x, b)$ em $\overline{G'}$.

Uma vez que G' é conexo e não possui as arestas (a, x) e (x, b) , então deverão existir pelo menos um dos caminhos, P' ou P'' em G' , sendo P' o caminho de a a x e P'' o caminho de b a x .

Se $|P'| > 3$ ou $|P''| > 3$, então G' contém um P_4 induzido, concluindo a prova.

Se $|P'| = 3$ e $|P''| = 3$ então sejam $P' = (a, x', x)$ e $P'' = (x, x'', b)$. Suponhamos, sem perda de generalidade, que exista o caminho P' . Pode-se verificar que $(b, x') \notin E(G')$, caso contrário o complemento do subgrafo $G[x, x', a, b]$ seria desconexo. Mas neste caso, $G[x, x', a, b]$ induz um P_4 .

Portanto, o complemento de todo subgrafo conexo não trivial de G é desconexo.

(iii) \Rightarrow (i) Iniciamos a prova observando que se G satisfaz (iii) então \overline{G} também satisfaz. De fato, suponha que \overline{G} não satisfaz (iii).

Seja \overline{H} um componente conexo não trivial de \overline{G} tal que $\overline{(\overline{H})} = H$ também é conexo. Como H é um subgrafo conexo não trivial de G e por hipótese, G satisfaz (iii) então \overline{H} é desconexo. Contradição.

Procederemos a prova por indução no número de vértices de G .

Para $n \leq 3$, o resultado vale trivialmente.

Suponha o resultado válido para grafos com menos de n vértices e seja G um grafo com n vértices.

Se G for desconexo temos, pela hipótese indutiva, que cada componente conexo G' de G é um cografo, pois $|V(G')| < n$. Logo, G é cografo.

Se G for conexo então, pela hipótese, \overline{G} é desconexo. Portanto, cada componente conexo G' de \overline{G} é um cografo, pois $|V(G')| < n$. Logo, \overline{G} é um cografo, implicando em G ser cografo. \square

A equivalência $(i) \Leftrightarrow (iii)$ do teorema 2.5 nos fornece uma caracterização de cografos através de sua árvore de decomposição, que afirma que G é um cografo se e somente se G possui uma co-árvore, dado pelo corolário a seguir:

Corolário 2.6. [10] *Um grafo G é um cografo se e somente se para todo subgrafo H de G com pelo menos dois vértices, exatamente uma das condições abaixo é satisfeita:*

(i) H é desconexo;

(ii) \overline{H} é desconexo;

□

Deste ponto em diante, representaremos a co-árvore de um grafo G da seguinte maneira: as folhas representarão os vértices do grafo e os nós internos serão rotulados por 0 ou 1, onde os nós rotulados por 0 serão chamados *nós tipo-0* e corresponderão ao módulo paralelo, e os nós rotulados por 1 serão chamados *nós tipo-1* e corresponderão ao módulo serial. Para facilitar a representação, iremos adotar a raiz como sendo sempre um nó tipo-1. Além disso, a raiz terá somente um filho se e somente se o grafo representado for desconexo.

Dado um cografo G e sua respectiva co-árvore T , a seguinte propriedade é facilmente verificada a partir da definição de co-árvore.

Propriedade 2.7. *Dois vértices x e y de G são adjacentes (não-adjacentes) se e somente se o menor ancestral comum a x e a y em T for um nó tipo-1 (nó tipo-0).*

Demonstração. É fácil verificar que x e y são adjacentes (não-adjacentes) se e somente se pertencem a um mesmo componente conexo do grafo original G

(\overline{G}) , se e somente se o menor ancestral comum aos dois vértices é um módulo serial (paralelo), i.e., um nó tipo-1 (nó tipo-0). \square

A figura 2.5 representa a mesma co-árvore da figura 2.1, porém já rotulada. E a figura 2.6 representa um cografo G_1 desconexo e sua respectiva co-árvore.

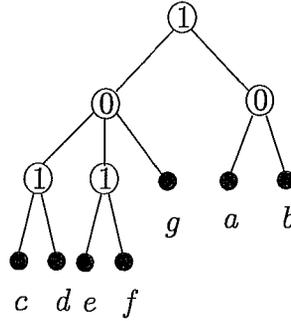


Figura 2.5: A co-árvore de G rotulada

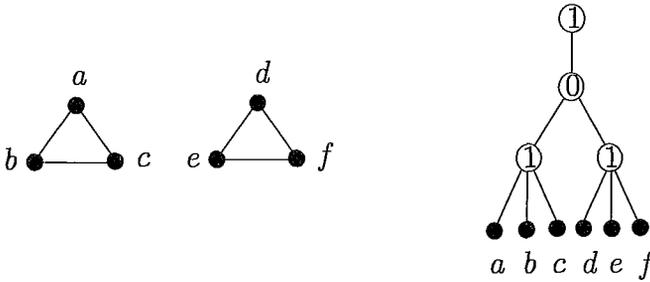


Figura 2.6: Grafo G_1 desconexo e sua respectiva co-árvore

A seguir, algumas definições e propriedades importantes dos cografos são apresentadas:

Definição 2.8. *Seja $G = (V, E)$ um grafo.*

Dizemos que G tem a propriedade clique-kernel se para toda clique maximal C de G e para todo kernel K de G , $|C \cap K| = 1$.

Denotaremos o conjunto de cliques maximais e kernels de G por ξ_G e χ_G , respectivamente. Além disso, $\xi_G(x)$ denota o conjunto de cliques maximais que contém o vértice x e $\xi_G(\bar{x})$ denota o conjunto de cliques maximais que não contém o vértice x . Analogamente, $\chi_G(x)$ denota o conjunto de kernels que contém o vértice x e $\chi_G(\bar{x})$ denota o conjunto de kernels que não contém o vértice x .

Lema 2.9. *Todo subgrafo de um cografo é um cografo.*

Vale lembrar que o termo subgrafo empregado em todo texto possui o mesmo sentido que subgrafo induzido.

Demonstração. Seja G um cografo com n vértices.

Para $n \leq 3$, o lema segue imediatamente da definição de cografos.

Podemos, portanto, assumir que $n \geq 4$.

Como todo subgrafo de um cografo G pode ser obtido removendo-se um vértice por vez, é portanto suficiente mostrar que a remoção de um único vértice de um cografo resulta em um cografo.

Sejam $G = (V, E)$ um cografo e T a co-árvore associada a G . O subgrafo $G' = V \setminus \{y\}$, onde $y \in V$, é um cografo se e somente se G' possuir uma co-árvore T' associada a ele (corolário 2.6).

A construção de T' será feita examinando x ($\text{pai}(y)$ em T). Para isto, considere os seguintes casos:

(i) x tem mais de dois filhos em T .

Como x tem mais de dois filhos em T então podemos remover y da co-árvore, continuando assim com uma co-árvore com as “mesmas” características que a anterior. Logo, a construção de T' é feita simplesmente pela remoção de y em T , como mostra a figura 2.7.

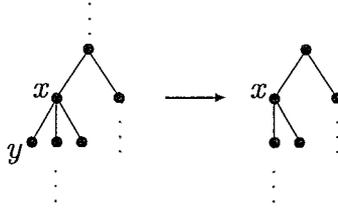


Figura 2.7: Ilustração da árvore T e sua sub-árvore T' construída a partir da remoção do vértice y de T para o item (i)

(ii) x tem exatamente dois filhos, y e y' , em T .

(a) Se y' é folha, a construção de T' será feita simplesmente pela remoção de x e y de T , e conectando y' ao pai de x , como mostra a figura 2.8.

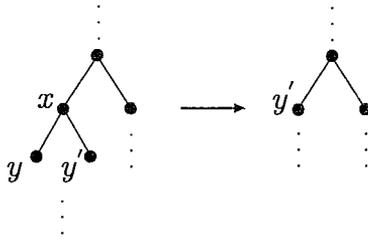


Figura 2.8: Ilustração da árvore T e sua sub-árvore T' construída com a remoção do vértice y de T para o caso (a) do item (ii)

(b) Se y' não é folha, a construção de T' será feita com a remoção de x , y e y' de T , e conectando os filhos de y' ao pai de x , como mostra a figura 2.9.

Nos casos acima, T' é uma co-árvore que representa de maneira única o subgrafo G' .

□

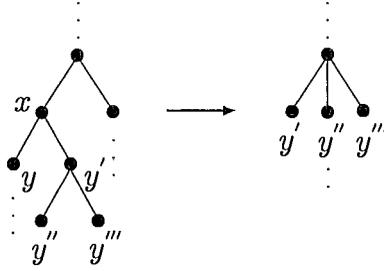


Figura 2.9: Ilustração da árvore T e sua subárvore T' construída com a remoção do vértice y de T para o caso (b) do item (ii)

2.2 Caracterização de cografos

Dadas as diversas classes de grafos com suas propriedades estruturais particulares, é importante determinar se um grafo pertence ou não a uma certa classe. Para isto, busca-se identificar características e propriedades que sejam exclusivas da classe. A classe dos cografos possui uma estrutura própria e muito forte, o que permite uma fácil identificação.

A seguir apresentamos algumas definições que serão necessárias no decorrer desta seção.

Definição 2.10. *Um grafo G é um grafo dacey quando para toda clique maximal C de G e para todo par de vértices distintos u e v tal que $C \subseteq N(u) \cup N(v)$ temos que $(u, v) \in E(G)$.*

Definição 2.11. *Um grafo G é dacey hereditário (DH) se e somente se para todo subgrafo $G' \subseteq G$, G' é um grafo dacey.*

O teorema a seguir, apresentado por Corneil, Lerchs e Burlingham [10], estabelece a equivalência entre sete diferentes caracterizações de cografos.

Teorema 2.12. [10] *Dado um grafo G , as seguintes afirmações são equivalentes:*

- (1) G é um cografo;
- (2) Todo subgrafo não trivial de G possui pelo menos um par de gêmeos;
- (3) Todo subgrafo de G tem a propriedade clique-kernel;
- (4) G não contém um P_4 como subgrafo;
- (5) O complemento de todo subgrafo conexo não trivial de G é desconexo;
- (6) G é um grafo DH;
- (7) Todo subgrafo conexo de G tem diâmetro menor ou igual a dois.

Demonstração. (1) \Rightarrow (2) : (G é um cografo \Rightarrow todo subgrafo não trivial de G possui pelo menos um par de gêmeos)

Sejam G um cografo e T sua respectiva co-árvore. Seja G' um subgrafo não trivial de G .

Seja T' a co-árvore correspondente a G' .

Como G' é um grafo não trivial então $|V'| \geq 2$, implicando em dizer que a co-árvore T' possui pelo menos um nó interno adjacente a pelo menos duas folhas, e como quaisquer duas folhas descendentes de um mesmo pai são irmãos, então qualquer subgrafo não trivial de G possui pelo menos um par de gêmeos.

(2) \Rightarrow (3) : (Todo subgrafo não trivial de G possui pelo menos um par de gêmeos \Rightarrow todo subgrafo de G tem a propriedade clique-kernel)

Esta prova será feita por indução no número p de vértices de H .

Se $p = 1$ temos que a clique maximal C e o kernel K possuem apenas um vértice, e portanto, $|C \cap K| = 1$.

No passo geral da indução, vamos supor que todos os subgrafos com p vértices possuem a propriedade clique-kernel e provaremos que todos os subgrafos com $p + 1$ vértices também possuem tal propriedade.

Sejam H um subgrafo de G com $p + 1$ vértices, C uma clique maximal de H e K um kernel de H .

Pela hipótese, temos que H possui pelo menos um par de gêmeos, x e x' . Considere o subgrafo $H' = H \setminus \{x'\}$. H' possui p vértices, logo pela hipótese indutiva, H' possui a propriedade clique-kernel.

Com o objetivo de concluir a prova, observamos que é possível expressar as cliques maximais e os kernels de H em termos das cliques maximais e dos kernels de H' .

(i) Se x, x' são gêmeos verdadeiros;

Toda clique maximal de H' que não contém x é uma clique maximal de H , pois ao adicionarmos o vértice x' em H' , x' não será adjacente a todos os vértices da clique maximal de H , já que x não é adjacente a tais vértices e x, x' são gêmeos. Logo, $\xi_H(\bar{x}) = \xi_{H'}(\bar{x})$.

Toda clique maximal de H' contendo x é uma clique maximal de H com a adição de x' , pois $N(x) = N(x')$ e x, x' são adjacentes, já que x, x' são gêmeos verdadeiros. Daí, $\xi_H(x) = \xi_{H'}(x) + x'$

Todo kernel de H' que não contém x é um kernel de H , pois se x não pertence ao kernel, isto quer dizer que x é adjacente a algum vértice y do kernel, então ao adicionarmos x' em H' teremos que x' também será adjacente a y , já que x, x' são gêmeos, logo x' também não pertence ao kernel, concluindo então que $\chi_H(\bar{x}) = \chi_{H'}(\bar{x})$.

Todo kernel de H' que contém x permanece um kernel em H (já que x' é adjacente a x); para qualquer kernel de H' contendo x , um novo kernel é

obtido em H ao substituir x por x' . Portanto, $\chi_H(x) = \chi_{H'}(x)$ e $\chi_H(x') = \chi_{H'}(x) - x + x'$.

Observemos que $\xi_H = \xi_H(x) \cup \xi_H(\bar{x})$ e $\chi_H = \chi_H(x) \cup \chi_H(\bar{x})$.

Agora, examinaremos a interseção de uma clique maximal arbitrária $C \in \xi_H$ e um kernel arbitrário $K \in \chi_H$. Antes, porém, relembramos que qualquer clique e qualquer conjunto independente possuem, no máximo, um vértice em comum. Temos quatro possibilidades a serem analisadas:

$$(1) C \in \xi_H(x) \text{ e } K \in \chi_H(x)$$

Neste caso, $|C \cap K| = 1$.

$$(2) C \in \xi_H(x) \text{ e } K \in \chi_H(\bar{x})$$

Temos que $C \in \xi_H(x) = \xi_{H'}(x) + x'$ e $K \in \chi_H(\bar{x}) = \chi_{H'}(\bar{x})$.

A inclusão do vértice x' alterou apenas o conjunto das cliques maximais, através da adição de x' em cada clique maximal C do conjunto das cliques maximais de H' que contém x . Além disso, vale lembrar que, $x' \notin \chi_H(\bar{x})$. Logo, pela hipótese indutiva, $|C \cap K| = 1$,

$$(3) C \in \xi_H(\bar{x}) \text{ e } K \in \chi_H(\bar{x})$$

Temos que $C \in \xi_H(\bar{x}) = \xi_{H'}(\bar{x})$ e $K \in \chi_H(\bar{x}) = \chi_{H'}(\bar{x})$.

Como os vértices, tanto da clique maximal C quanto os vértices do kernel K do grafo H , mantiveram-se os mesmos que os vértices da clique maximal e do kernel de H' então, pela hipótese indutiva, $|C \cap K| = 1$.

$$(4) C \in \xi_H(\bar{x}) \text{ e } K \in \chi_H(x)$$

Temos que $C \in \xi_H(\bar{x}) = \xi_{H'}(\bar{x})$ e $K \in \chi_H(x) = \chi_{H'}(x) - x + x'$.

Pela hipótese indutiva, é fácil ver que $|C \cap K| = 1$, já que temos que a interseção da clique maximal C' de H' com o kernel K' de H' é um, e $x \notin C' \cap K'$, visto que $x \notin C'$. Conclui-se, então, que a interseção de $C \cap K$ permanece a mesma ao adicionar o vértice x' em K' e remover o vértice x de K' . Portanto, $|C \cap K| = 1$.

Depois de analisarmos as 4 possibilidades, podemos concluir que, para gêmeos verdadeiros, todo subgrafo de G tem a propriedade clique-kernel.

A análise para o caso em que x, x' são gêmeos falsos é feita de maneira análoga.

(ii) Se x, x' são gêmeos falsos;

Toda clique maximal de H' que não contém x é uma clique maximal de H , pois se x não pertence a clique maximal isto quer dizer que x não é adjacente a algum vértice y da clique maximal então, ao adicionarmos o vértice x' em H' , teremos que x' também não é adjacente a y , já que x, x' são gêmeos, logo x' não pertence a clique maximal. Concluimos então que $\xi_H(\bar{x}) = \xi_{H'}(\bar{x})$.

Toda clique maximal de H' que contém x permanece uma clique maximal em H (já que x' é não-adjacente a x); para qualquer clique maximal de H' contendo x , uma nova clique maximal é obtida em H ao substituir x por x' . Portanto, $\xi_H(x) = \xi_{H'}(x)$ e $\xi_H(x') = \xi_{H'}(x) - x + x'$

Todo kernel de H' que não contém x é um kernel de H pois ao adicionarmos o vértice x' em H' , x' é adjacente a algum vértice do kernel de H , já que x também o é e eles são gêmeos, isto é, $\chi_H(\bar{x}) = \chi_{H'}(\bar{x})$.

Todo kernel de H' que contém x é um kernel de H com a adição do vértice x' , pois $N(x) = N(x')$ e x, x' são não-adjacentes, já que x, x' são gêmeos falsos. Portanto, $\chi_H(x) = \chi_{H'}(x) + x'$.

Podemos observar que, $\xi_H = \xi_H(x) \cup \xi_H(\bar{x})$ e $\chi_H = \chi_H(x) \cup \chi_H(\bar{x})$.

A prova é idêntica ao caso (i) onde as noções de clique maximal e kernel são trocadas, e portanto, a prova será omitida.

De (i) e (ii) concluímos que, se todo subgrafo de G tem pelo menos um par de gêmeos então todo subgrafo de G tem a propriedade clique-kernel.

(3) \Rightarrow (4) : (Todo subgrafo de G tem a propriedade clique-kernel $\Rightarrow G$ não contém um P_4 como subgrafo)

Suponhamos, por absurdo, que G contém um $P_4 = (x, y, z, w)$ como subgrafo. O subgrafo formado pelo P_4 não possui a propriedade clique-kernel: considerando a clique maximal C formada pelos vértices y e z e o kernel K formado pelos vértices x e w , verificamos que $|C \cap K| = 0$, contradizendo a hipótese, pois $|C \cap K| = 1$, para todo subgrafo de G .

Portanto, G não contém um P_4 como subgrafo.

(4) \Rightarrow (5) : (G não contém um P_4 como subgrafo \Rightarrow o complemento de todo subgrafo conexo não trivial de G é desconexo)

A prova desta implicação foi mostrada no teorema 2.5, implicação (ii) \Rightarrow (iii).

(5) \Rightarrow (6) : (O complemento de todo subgrafo conexo não trivial de G é desconexo $\Rightarrow G$ é DH)

Seja G um grafo tal que o complemento de todo subgrafo (não trivial) conexo de G seja desconexo.

Suponhamos, por absurdo, que G não é DH , i.e., existe um subgrafo G' que não é um grafo dacey.

Como G' não é um grafo dacey então temos que existe uma clique maximal C tal que $C \subseteq N(u) \cup N(v)$ e $(u, v) \notin E(G)$.

Seja $x \in V(C)$. Como $x \in N(u) \cup N(v)$, temos que $(x, u) \in E(G)$ ou $(x, v) \in E(G)$.

Seja Z o conjunto dos vértices de C que são adjacentes tanto ao vértice u quanto ao vértice v , isto é, $Z = \{z \in V(C) \mid (z, u) \in E(C) \text{ e } (z, v) \in E(C)\}$.

Considere $C' = C \setminus Z$. Podemos, portanto, particionar a clique C' em dois conjuntos de vértices, X e Y , de forma que o conjunto X é formado pelos vértices que são adjacentes apenas ao vértice u e o conjunto Y é formado pelos vértices que são adjacentes apenas ao vértice v . Observe que $C' = X \cup Y$ e $X \cap Y = \emptyset$. Além disso, $X \neq \emptyset$ e $Y \neq \emptyset$, caso contrário, C não seria maximal.

Logo, existem vértices $x \in X$ e $y \in Y$, tal que $(x, u) \in E(G)$, $(x, v) \notin E(G)$, $(y, v) \in E(G)$ e $(y, u) \notin E(G)$. Pela Figura 2.10 vemos que $G[x, y, u, v]$ induz um subgrafo conexo, isomorfo a um P_4 , cujo complemento é também conexo, contradizendo nossa hipótese.

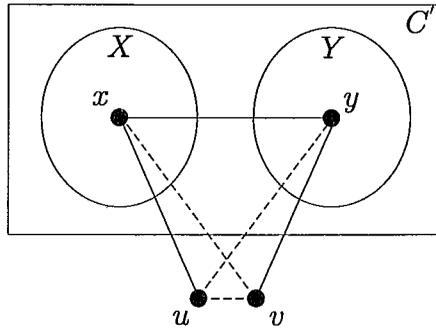


Figura 2.10: $G[u, v, x, y]$

(6) \Rightarrow (7) : (G é $DH \Rightarrow$ todo subgrafo conexo de G tem diâmetro menor ou igual a dois)

Suponhamos que G é DH .

Suponhamos, por absurdo, que nem todo subgrafo conexo de G tem diâmetro menor ou igual a dois.

Seja G' um subgrafo de G com diâmetro maior ou igual a três. Logo, existem pelo menos dois vértices $u, v \in V(G')$ tal que a distância entre u e

v é maior ou igual a três. Sem perda de generalidade, tomemos u e v dois vértices de G' cuja distância seja três. Observe que, neste caso, $(u, v) \notin E(G)$.

Denote por $P = (u, w_1, w_2, v)$ o caminho, de tamanho três, entre os vértices u e v , como mostra a figura 2.11.

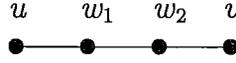


Figura 2.11: Caminho com 4 vértices

Considere $C = \{w_1, w_2\}$. Observe que $N(u) \supseteq \{w_1\}$, $N(v) \supseteq \{w_2\}$. Logo, $C \subseteq N(u) \cup N(v)$.

Como G é DH , $(u, v) \in V(G)$, já que $C \subseteq N(u) \cup N(v)$, contradizendo o fato de $(u, v) \notin E(G)$.

Logo, todo subgrafo de G tem diâmetro menor ou igual a dois.

(7) \Rightarrow (4) : (Se todo subgrafo conexo de G tem diâmetro menor ou igual a dois $\Rightarrow G$ não contém um P_4 como subgrafo)

Suponhamos, por absurdo, que todo subgrafo conexo de G tem diâmetro menor ou igual a dois e G contém um P_4 como subgrafo.

P_4 é conexo com diâmetro igual a três. Absurdo.

(5) \Rightarrow (1) : (O complemento de todo subgrafo conexo não trivial de G é desconexo $\Rightarrow G$ é um cografo.)

A prova desta implicação foi demonstrada no teorema 2.5, implicação (iii) \Rightarrow (i). □

Observação 2.13. *Se G é um cografo então G é um grafo perfeito.*

De fato, todo cografo G é perfeito, pois G e seu complemento, \overline{G} , não possuem P_4 , como mostrado no teorema 2.12, logo não possuem ciclo ímpar induzido de tamanho maior ou igual a cinco.

Capítulo 3

Reconhecimento de cografos

Como vimos na seção 2.2, são conhecidas diversas caracterizações de cografos, isto é, características estruturais dos cografos, de forma que se um dado grafo G apresentar uma dessas características então pode-se concluir que G é um cografo. Neste ponto, faz-se necessária a seguinte pergunta: como verificar (ou reconhecer), em tempo polinomial, se um dado grafo G possui alguma dessas características?

Felizmente, a classe dos cografos apresenta algoritmos de reconhecimento em tempo linear. Este capítulo destina-se a apresentar e descrever um desses algoritmos [11], assim como justificar sua complexidade. Além disso, mostramos que vários problemas *NP*-difíceis podem ser resolvidos eficientemente quando restritos à classe dos cografos.

Na seção 3.1 descrevemos um algoritmo linear para reconhecer cografos. Na seção 3.2 analisamos sua complexidade e, por último, na seção 3.3 apresentamos dois algoritmos: um para resolver o problema da clique máxima e outro para o problema do conjunto independente máximo, os quais serão utilizados no capítulo 4 para demonstrar a contribuição principal desta dissertação.

3.1 Algoritmo linear

Várias das equivalências do Teorema 2.12 possuem algoritmos polinomiais para seu reconhecimento, como é o caso das estruturas P_4 induzidas. Por isso, é possível construir algoritmos que reconheçam um cografo e construam sua co-árvore correspondente em tempo polinomial.

A seguir descrevemos um algoritmo linear para reconhecer cografos.

Idéia do algoritmo:

Dado que todo subgrafo de um cografo também é um cografo, o algoritmo constrói a co-árvore T de um cografo G vértice a vértice. Isto é, uma vez que tenhamos um cografo G e sua co-árvore T , verificamos se $G + x$ é cografo, onde x é um novo vértice a ser introduzido em G . Para isso, analisa-se as relações de adjacência de x com os vértices de G usando T .

O conjunto M , resultado do procedimento *Marcar*, descrito a seguir, provê condições necessárias e suficientes para que $G + x$ seja um cografo, se G o for. O algoritmo inicia com a co-árvore para dois vértices, e incrementa os vértices restantes na co-árvore um a um.

O algoritmo enunciado a seguir, dado por Corneil *et al.* [11] reconhece se um dado grafo G é cografo.

Antes de apresentar o algoritmo, introduzimos a seguinte notação. Seja w um nó da árvore T (com raiz R), denotamos por $f(w)$ o número de filhos de w em T e $mf(w)$ o número atual de filhos de w que foram marcados e, subsequentemente, desmarcados. Vale mencionar que o valor de $mf(w)$ será automaticamente atualizado pelo procedimento *Marcar*, dado a seguir.

O seguinte procedimento usa a informação de adjacência do novo vértice x para “marcar” ou então “desmarcar”, quando apropriado, os nós em T .

Entrada: A co-árvore T

Procedimento Marcar(x)

```
1 para cada nó  $v$  de  $T$  faça
2    $mf(v) := 0$ ;
3 fim_para;
4 Marcar todas as folhas de  $T$  que são adjacentes a  $x$ ;
5 para cada nó  $u$  marcado de  $T$  com  $f(u) = mf(u)$  faça
6   desmarcar o vértice  $u$ ;
7    $mf(u) := 0$ ;
8   se  $u \neq R$  então
9     marcar o vértice  $w$ , onde  $w$  é o pai de  $u$ ;
10     $mf(w) := mf(w) + 1$ ;
11    Inserir  $u$  no início da lista de filhos marcados e desmarcados de
12      $w$ ;
13 fim_se;
14 fim_para;
15 se algum vértice continuar marcado e  $f(R) = 1$  então
16   marcar a raiz  $R$ ;
17 fim_se;
```

Denote por M o conjunto de nós internos de T que continuam marcados após o procedimento *Marcar* ter sido executado, e sejam α o nó em M com o maior nível em T e β o nó em $M \setminus \{\alpha\}$ com o maior nível em T .

Dizemos que um nó marcado γ tipo-1 é *propriamente marcado* se e somente se $mf(\gamma) = f(\gamma) - 1$.

Um *caminho alternante legítimo* em uma co-árvore marcada é um caminho alternante de nós tipo-1 propriamente marcados e nós tipo-0 desmarcados, cujos extremos do caminho são nós tipo-1.

O procedimento *Marcar* é ilustrado pelo grafo G e sua co-árvore T da figura 3.1 ao considerar a adição do vértice x adjacente aos vértices a , d , e e f de G . Na figura, associamos $*$ com os nós marcados e $(*)$ com os nós que foram marcados e, subsequentemente, desmarcados. Nesta mesma figura, observamos que os nós R e $\text{pai}(a)$ são nós marcados e os nós a , d , e , f e $\text{pai}(d)$ são desmarcados. Além disso, temos $M = \{R, \text{pai}(a)\}$, $\alpha = \text{pai}(a)$ e $\beta = R$.

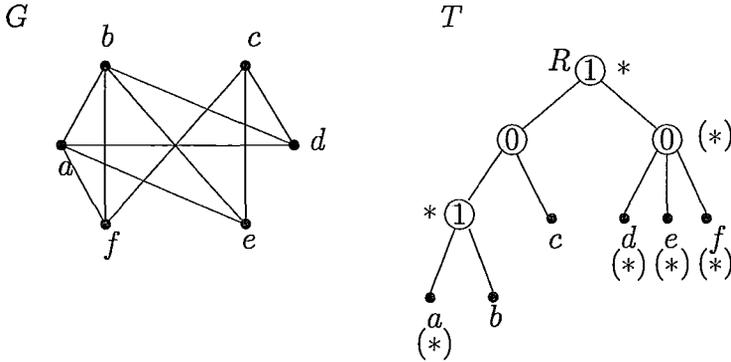


Figura 3.1: Um cografo e sua co-árvore correspondente

O teorema seguinte mostra como utilizar o conjunto marcado M para determinar se $G + x$ é ou não um cografo. Utilizaremos este teorema como base para o algoritmo linear de reconhecimento dos cografos.

Teorema 3.1. [11] *Se G é um cografo com a co-árvore T então $G + x$ é um cografo se e somente se satisfaz uma das seguintes condições:*

1. M é vazio

2. (a) $M \setminus \{\alpha\}$ consiste exatamente dos nós tipo-1 do caminho alternante legítimo (possivelmente vazio) terminando em R , e
- (b) ou α é um nó tipo-0 cujo pai é β ou α é um nó tipo-1 cujo avô, se existir, é β .

Demonstração. (\Rightarrow) Seja G um cografo com a co-árvore T . Suponhamos que $G + x$ é um cografo.

Vamos supor, por absurdo, que as condições 1 e 2 não sejam satisfeitas, isto é, $M \neq \emptyset$ e além disso, pelo menos uma das afirmações seguintes ocorre.

- (i) $M \setminus \{\alpha\}$ contém um nó tipo-0;
- (ii) Existe um nó tipo-1 em $M \setminus \{\alpha\}$ que não é propriamente marcado;
- (iii) Existe $\gamma \neq R$ em $M \setminus \{\alpha\}$ tal que o avô de γ não está em $M \setminus \{\alpha\}$;
- (iv) Os vértices de $M \setminus \{\alpha\}$ não formarão um caminho até R ;
- (v) α é um nó tipo-0 cujo pai não é β ;
- (vi) α é um nó tipo-1 cujo avô não é β .

Por definição de M e pelo procedimento *Marcar*, temos os seguintes fatos:

Fato 3.2. *Para todo vértice $y \in M$, existem pelo menos duas folhas descendentes de y tal que uma é adjacente a x e a outra é não-adjacente a x .*

Observe que, se não existe pelo menos uma folha descendente de $y \in M$ adjacente a x então, pelo procedimento *Marcar*, y não seria um nó marcado, e se não existe pelo menos uma folha não-adjacente a x , então teríamos $mf(y) = f(y)$, e y teria sido marcado pelo procedimento *Marcar*.

Fato 3.3. *Toda folha da co-árvore T que foi marcada pelo procedimento Marcar será desmarcada até o fim da execução do procedimento Marcar.*

O fato 3.3 é fácil de ser verificado, pois se alguma folha t da árvore T foi marcada e não foi desmarcada, isto implica em dizer que $f(t) \neq mf(t)$, isto é, $mf(t) < f(t)$, mas isto é uma contradição, já que $f(t) = 0$ e $mf(t) \in \mathbb{N}$.

Fato 3.4. *Se um nó interno γ da co-árvore T não foi marcado pelo procedimento Marcar então existe pelo menos uma folha descendente de γ que é não-adjacente a x .*

A verificação do fato 3.4 é simples. Suponhamos, por absurdo, que todas as folhas descendentes de γ são adjacentes a x . Pelo fato 3.3 e pelo procedimento *Marcar*, todos os descendentes de γ terão sido marcados e, subsequentemente, desmarcados. Consequentemente, γ será marcado pelo procedimento *Marcar*.

Usando estes fatos, mostraremos que todas as seis condições anteriores implicam na existência de um P_4 induzido em $G + x$.

(i) $M \setminus \{\alpha\}$ contém um nó tipo-0;

Sejam γ um nó tipo-0 de $M \setminus \{\alpha\}$ e δ o menor ancestral comum a α e γ em T (possivelmente $\delta = \gamma$).

Existem 4 casos a serem considerados, dependendo dos rótulos 0 ou 1 de α e δ .

Caso 1: α e δ são ambos nós tipo-0;

Como $\alpha \in M$ então, pelo fato 3.2, existem pelo menos duas folhas a e b , descendentes de α , tal que $(a, x) \in E(G + x)$ e $(b, x) \notin E(G + x)$. Da mesma forma, como $\gamma \in M$ temos que existe pelo menos uma folha d descendente de γ tal que $(d, x) \in E(G + x)$.

Seja c uma folha descendente do pai de α , que não é descendente de α .

Podemos observar que $\text{pai}(\alpha)$ é um nó tipo-1 e portanto não pode ser δ .

Pela propriedade 2.7, temos que $(a, c) \in E(G)$, $(b, c) \in E(G)$, $(c, d) \notin E(G)$, $(a, b) \notin E(G)$, $(a, d) \notin E(G)$ e $(b, d) \notin E(G)$.

Agora, temos dois casos a considerar:

(1.1) c é adjacente a x ;

Logo, existe um P_4 induzido formado pelos vértices b, c, x e d , como mostra a figura 3.2

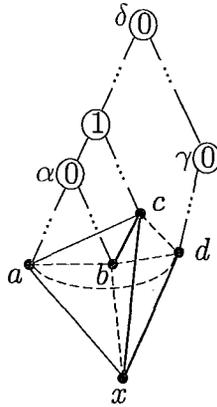


Figura 3.2: Ilustração do P_4 induzido pelos vértices b, c, x e d , referente ao caso 1.1 do item (i).

(1.2) c é não-adjacente a x ;

Logo, existe um P_4 induzido formado pelos vértices b, c, a e x , como mostra a figura 3.3

No caso em que $\delta = \gamma$, tomamos d como sendo uma folha descendente de γ , mas que não seja descendente de θ , onde θ é um filho de γ que está no caminho de α a γ .

Caso 2: α é um nó tipo-1 e δ é um nó tipo-0;

Temos que $\alpha \in M$ então, pelo fato 3.2, existem pelo menos duas folhas, a e b , descendentes de α tal que $(a, x) \in E(G + x)$ e $(b, x) \notin E(G + x)$. E

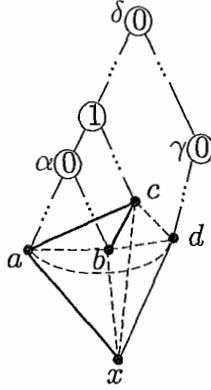


Figura 3.3: Ilustração do P_4 induzido pelos vértices b, c, a e x , referente ao caso 1.2 do item (i)

como $\gamma \in M$ temos que existe pelo menos uma folha c descendente de γ tal que $(c, x) \in E(G + x)$.

É fácil ver que b, a, x, c induzem um P_4 já que, pela propriedade 2.7, $(a, b) \in E(G)$, $(a, c) \notin E(G)$ e $(b, c) \notin E(G)$.

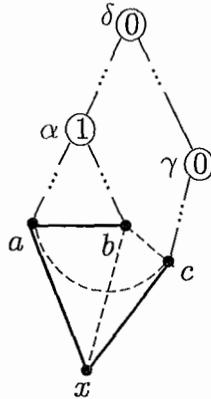


Figura 3.4: Ilustração do P_4 formado pelos vértices b, a, x e c , referente ao caso 2 do item (i)

No caso em que $\delta = \gamma$, tomamos c como sendo uma folha descendente de γ , mas que não seja descendente de θ , onde θ é um filho de γ que está no caminho de α a γ .

Caso 3: α é um nó tipo-0 e δ é um nó tipo-1;

Como $\alpha \in M$ então, pelo fato 3.2, existem pelo menos duas folhas, a e b , descendentes de α tal que $(a, x) \in E(G + x)$ e $(b, x) \notin E(G + x)$. E como $\gamma \in M$ temos que existe pelo menos uma folha c descendente de γ tal que $(c, x) \notin E(G + x)$.

Pela propriedade 2.7, $(a, b) \notin E(G)$, $(a, c) \in E(G)$ e $(b, c) \in E(G)$. Logo, os vértices b, c, a, x induzem um P_4 , como mostra a figura 3.5.

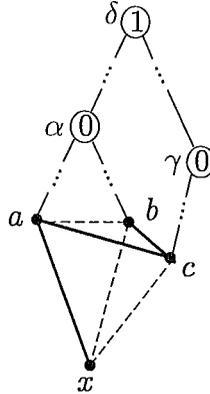


Figura 3.5: Ilustração do P_4 formado pelos vértices b, c, a e x , referente ao caso 3 do item (i)

Caso 4: α e δ são ambos nós tipo-1;

Como $\alpha \in M$ então, pelo fato 3.2, existe pelo menos uma folha a descendente de α tal que $(a, x) \notin E(G + x)$. Da mesma maneira, como $\gamma \in M$ temos que existem pelo menos duas folhas, b e c , descendentes de γ tal que $(b, x) \notin E(G + x)$ e $(c, x) \in E(G + x)$.

Pela propriedade 2.7, temos que $(a, b) \in E(G)$, $(a, c) \in E(G)$ e $(b, c) \notin E(G)$.

Concluimos assim, que $G + x$ possui um P_4 induzido pelos vértices b , a , c e x , como mostra a figura 3.6.

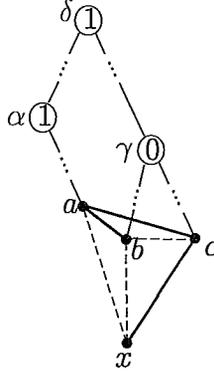


Figura 3.6: Ilustração do P_4 formado pelos vértices b , a , c e x , referente ao caso 4 do item (i)

As provas dos demais casos são feitas de maneira análoga. Como uma das contribuições desta dissertação, descrevemos as mesmas com todos os detalhes, visto que não encontramos na literatura as demonstrações de tais casos.

(ii) Existe um nó tipo-1 em $M \setminus \{\alpha\}$ que não é propriamente marcado;

Seja δ um nó tipo-1 em $M \setminus \{\alpha\}$ que não é propriamente marcado, isto é, $mf(\delta) \leq f(\delta) - 2$.

Como $mf(\delta) \leq f(\delta) - 2$, δ possui pelo menos dois filhos, γ e β , que no final do procedimento *Marcar* ficaram marcados ou nem sequer foram marcados. Observe que γ e β são nós tipo-0, já que δ é tipo-1 e é pai de γ e β .

Temos, portanto, três possibilidades a serem analisadas:

Caso 1: γ e β ambos ficaram marcados após o procedimento *Marcar*;

Neste caso, $\gamma, \beta \in M$, já que γ e β ficaram marcados após o procedimento *Marcar*, com pelo menos um desses dois nós pertencente a $M \setminus \{\alpha\}$. Logo, temos um nó tipo-0 em $M \setminus \{\alpha\}$. Pelo item (i), provado anteriormente, segue que existe um P_4 induzido em $G + x$.

Caso 2: Um nó foi marcado e o outro nó nem foi marcado pelo procedimento *Marcar*;

Sem perda de generalidade, suponhamos que o nó γ ficou marcado e o nó β nem foi marcado. Como γ ficou marcado temos que $\gamma \in M$.

Se $\gamma \neq \alpha$ então γ é um nó tipo-0 pertencente a $M \setminus \{\alpha\}$, implicando na existência de um P_4 induzido em $G + x$, como foi provado no item (i).

Se $\gamma = \alpha$ temos que $\gamma \in M$ e pelo fato 3.2, existem pelo menos duas folhas descendentes de γ tal que $(a, x) \in E(G + x)$ e $(b, x) \notin E(G + x)$.

Como β nem foi marcado então duas possibilidades podem ser analisadas:

(2.1) β é uma folha;

Neste caso, $(\beta, x) \notin E(G + x)$, caso contrário, β teria sido marcado. Além disso, pela propriedade 2.7, $(a, \beta) \in E(G)$, $(b, \beta) \in E(G)$ e $(a, b) \notin E(G)$. Logo, os vértices x, a, β e b de $G + x$ induzem um P_4 , como mostra a figura 3.7.

(2.2) β não é uma folha;

Como β não é folha e nem foi marcado então, pelo fato 3.4, existe pelo menos uma folha c descendente de β tal que $(c, x) \notin E(G + x)$.

Pela propriedade 2.7, podemos observar que $(a, b) \notin E(G)$, $(a, c) \in E(G)$ e $(b, c) \in E(G)$.

Logo, os vértices b, c, a e x de $G + x$ induzem um P_4 , como mostra a figura 3.8.

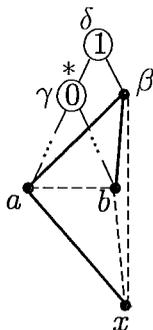


Figura 3.7: Ilustração do P_4 formado pelos vértices x , a , β e b , referente ao caso 2.1 do item (ii)

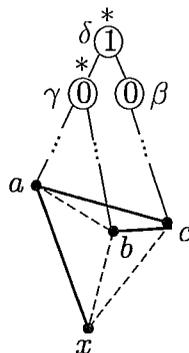


Figura 3.8: Ilustração do P_4 formado pelos vértices b , c , a e x , referente ao caso 2.1 do item (ii)

Caso 3: γ e β ambos nem foram marcados pelo procedimento *Marcar*;

Como γ não foi marcado então, pelo fato 3.4, existe pelo menos uma folha b descendente de γ tal que $(b, x) \notin E(G+x)$. Da mesma maneira, temos que β possui pelo menos uma folha c descendente tal que $(c, x) \notin E(G+x)$. Temos que $\delta \in M$ então, pelo fato 3.2, existe pelo menos uma folha a descendente de δ tal que $(a, x) \in E(G+x)$.

Seja Ω o menor ancestral comum a δ e α ($\delta \neq \alpha$ já que $\delta \in M \setminus \{\alpha\}$).

Considere as duas seguintes possibilidades:

(3.1) Ω é um nó tipo-0;

Como $\alpha \in M$ então, pelo fato 3.2, temos que existe uma folha d descendente de α tal que $(d, x) \in E(G+x)$.

Observe que, pela propriedade 2.7, $(a, b) \in E(G)$, $(a, c) \in E(G)$, $(b, c) \in E(G)$, $(a, d) \notin E(G)$, $(b, d) \notin E(G)$ e $(c, d) \notin E(G)$.

Logo, $G+x$ possui um P_4 induzido formado pelos vértices d , x , a e c , como mostra a figura 3.9

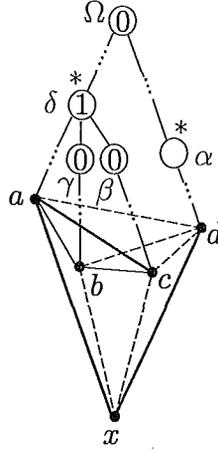


Figura 3.9: Ilustração do P_4 formado pelos vértices d, x, a e c , referente ao caso 3.1 do item (ii)

(3.2) Ω é um nó tipo-1;

Como Ω e δ são nós tipo-1 então Ω não é pai de δ . Seja e uma folha descendente do pai de δ , tal que e não é descendente de δ .

Temos que $\alpha \in M$ então, pelo fato 3.2, existe uma folha d descendente de α tal que $(d, x) \notin E(G + x)$.

Observe que, pela propriedade 2.7, $(a, e) \notin E(G)$, $(e, d) \in E(G)$, $(b, e) \notin E(G)$, $(a, b) \in E(G)$, $(a, d) \in E(G)$, e $(b, d) \in E(G)$.

Como não sabemos se e é ou não adjacente a x então temos duas possibilidades a serem consideradas:

(3.2.1) e é adjacente a x ;

Logo, existe um P_4 induzido formado pelos vértices e, x, a e b , como mostra a figura 3.10.

(3.2.2) e é não-adjacente a x ;

Então existe um P_4 induzido formado pelos vértices e, d, a e x , como mostra a figura 3.11.

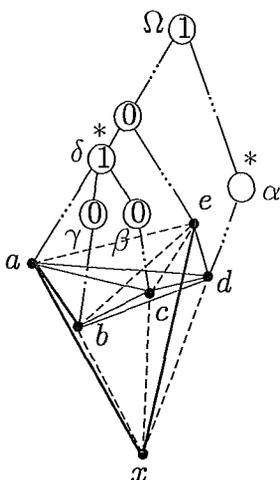


Figura 3.10: Ilustração do P_4 formado pelos vértices e, x, a e b , referente ao caso 3.2.1 do item (ii)

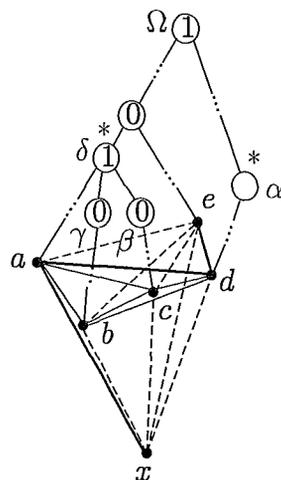


Figura 3.11: Ilustração do P_4 formado pelos vértices e, d, a e x , referente ao caso 3.2.2 do item (ii)

(iii) Existe $\gamma \neq R$ em $M \setminus \{\alpha\}$ tal que o avô de γ não está em $M \setminus \{\alpha\}$;

Para provarmos a existência de um P_4 em $G + x$, devemos analisar duas possibilidades:

(1) γ não tem avô;

Se γ não tem avô então γ possui como pai a raiz R . Como R é um nó tipo-1 então γ é um nó tipo-0, mas foi provado no item (i) se γ é um nó tipo-0, existe um P_4 induzido em $G + x$.

(2) γ tem avô;

Seja δ o avô de γ . Temos que $\delta \notin M \setminus \{\alpha\}$.

Observe que $\delta \neq \alpha$, pois δ não é o nó de M com o maior nível (δ é avô de $\gamma \in M \setminus \{\alpha\}$).

Como foi provado no item (i) se $M \setminus \{\alpha\}$ contiver um nó tipo-0, então $G + x$ contém um P_4 induzido. Vamos então analisar apenas o caso em que

$\gamma \in M \setminus \{\alpha\}$ é um nó tipo-1. Neste caso, δ também é um nó tipo-1, já que δ é avô de γ .

Como $\gamma \in M$ então, pelo fato 3.2, existem pelo menos duas folhas, a e b , descendentes de γ tal que $(a, x) \in E(G+x)$ e $(b, x) \notin E(G+x)$. Da mesma forma, existem pelo menos duas folhas, c e d , descendentes de α tal que $(c, x) \notin E(G+x)$ e $(d, x) \in E(G+x)$.

Vamos analisar duas possibilidades:

(2.1) α e γ possuem como menor ancestral o pai de γ .

Como o pai de γ é tipo-0 então, pela propriedade 2.7, $(a, d) \notin E(G)$ e $(b, d) \notin E(G)$.

Logo, existe um P_4 induzido formado pelos vértices b, a, x e d , como mostra a figura 3.12

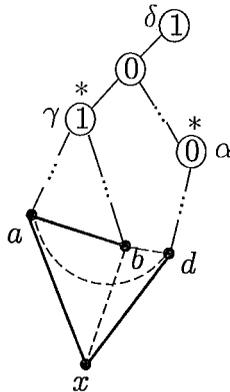


Figura 3.12: Ilustração do P_4 formado pelos vértices b, a, x e d , referente ao caso 2.1 do item (iii)

(2.2) α e γ não possuem como menor ancestral o pai de γ .

Seja Ω o menor ancestral comum a α e γ .

Se Ω for um nó tipo-0 então a análise é análoga ao caso em que α e γ possuem como menor ancestral o pai de γ .

Se Ω for um nó tipo-1 então seja e uma folha descendente do pai de γ que não é descendente de γ .

Pela propriedade 2.7, $(a, b) \in E(G)$, $(a, c) \in E(G)$, $(b, c) \in E(G)$, $(c, e) \in E(G)$, $(a, e) \notin E(G)$, $(b, e) \notin E(G)$.

Agora, temos que analisar dois casos:

(2.2.1) e é adjacente a x ;

Logo, existe um P_4 induzido pelos vértices e, x, a e b , como mostra a figura 3.13.

(2.2.2) e é não-adjacente a x ;

Neste caso, existe um P_4 induzido pelos vértices e, c, a e x , como mostra a figura 3.14

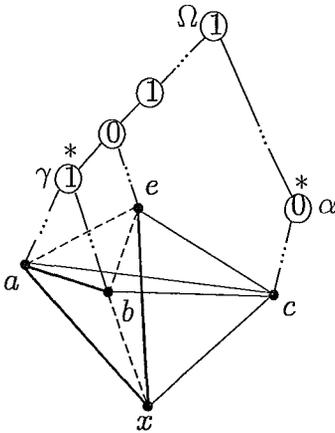


Figura 3.13: Ilustração do P_4 formado pelos vértices e, x, a e b , referente ao caso 2.2.1 do item (iii)

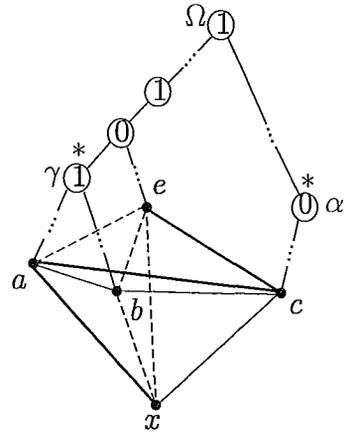


Figura 3.14: Ilustração do P_4 formado pelos vértices e, c, a e x , referente ao caso 2.2.2 do item (iii)

(iv) Os vértices de $M \setminus \{\alpha\}$ não formam um caminho até R ;

Se os vértices de $M \setminus \{\alpha\}$ não formam um caminho até R , isto implica em dizer que existe pelo menos um nó tipo-1, δ , (provado no item (i) que

$M \setminus \{\alpha\}$ deve possuir somente nós tipo-1 para que não haja a presença de P_4 no caminho que não está marcado, isto é, $\delta \notin M \setminus \{\alpha\}$.

Observe que δ é avô de algum nó tipo-1 γ que está em $M \setminus \{\alpha\}$. Como $\gamma \neq R$ e $\gamma \in M \setminus \{\alpha\}$ tal que $\delta \notin M \setminus \{\alpha\}$ então a análise do item (iv) é análoga à análise do item (iii).

(v) α é um nó tipo-0 cujo pai não é β ;

Sejam α o nó de maior nível de M , β o nó de maior nível de $M \setminus \{\alpha\}$ e δ o menor ancestral comum a α e a β .

Seja c uma folha descendente do pai de α que não é descendente a α .

Observe que β é um nó tipo-1, pois foi provado no item (i) que $M \setminus \{\alpha\}$ não contém um nó tipo-0, caso contrário, existe um P_4 induzido em $G + x$.

Como α é um nó marcado então, pelo fato 3.2, existem pelo menos duas folhas, a e b , descendentes de α tal que $(a, x) \in E(G+x)$ e $(b, x) \notin E(G+x)$. Da mesma forma, existem pelo menos duas folhas, d e e , descendentes de β tal que $(d, x) \in E(G+x)$ e $(e, x) \notin E(G+x)$.

Temos duas possibilidades a considerar:

(1) δ é um nó tipo-0;

Pela propriedade 2.7 temos que $(a, c) \in E(G)$, $(b, c) \in E(G)$, $(a, b) \notin E(G)$, $(a, d) \notin E(G)$, $(b, d) \notin E(G)$ e $(c, d) \notin E(G)$.

(1.1) c é adjacente a x ;

Neste caso, existe um P_4 induzido pelos vértices b, c, x e d , como mostra a figura 3.15.

(1.2) c é não-adjacente a x ;

Neste caso, existe um P_4 induzido pelos vértices b, c, a e x , como mostra a figura 3.16.

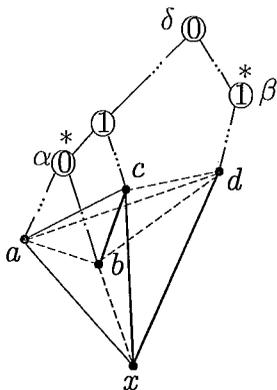


Figura 3.15: Ilustração do P_4 formado pelos vértices b, c, x e d , referente ao caso 1.1 do item (v)

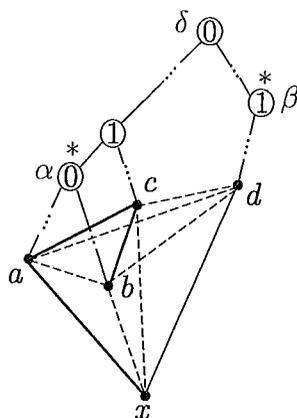


Figura 3.16: Ilustração do P_4 formado pelos vértices b, c, a e x , referente ao caso 1.2 do item (v)

Pela propriedade 2.7, temos que $(a, e) \in E(G)$, $(b, e) \in E(G)$ e $(a, b) \notin E(G)$. Logo, existe um P_4 induzido pelos vértices b, e, a e x , como mostra a figura 3.17.

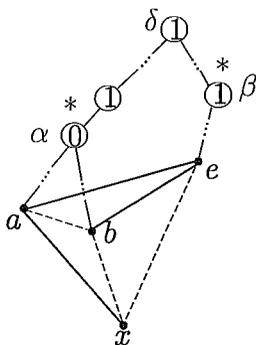


Figura 3.17: Ilustração do P_4 induzido pelos vértices b, e, a e x , referente ao caso 2 do item (v)

(vi) α é um nó tipo-1 cujo avô não é β ;

Sejam α o nó de maior nível de M , β o nó de maior nível de $M \setminus \{\alpha\}$ e δ o menor ancestral comum a α e a β .

Temos que β é um nó tipo-1, pois pelo item (i), $M \setminus \{\alpha\}$ não contém nó tipo-0, caso contrário, existe um P_4 induzido em $G + x$.

Como $\alpha \in M$ então, pelo fato 3.2, existem pelo menos duas folhas, a e b , descendentes a α tal que $(a, x) \in E(G + x)$ e $(b, x) \notin E(G + x)$. Da mesma maneira, existem pelo menos duas folhas, d e e , descendentes a β tal que $(d, x) \in E(G + x)$ e $(e, x) \notin E(G + x)$.

Considere as duas seguintes possibilidades:

(1) δ é um nó tipo-0;

Pela propriedade 2.7, temos que $(a, b) \in E(G)$, $(a, d) \notin E(G)$ e $(b, d) \notin E(G)$.

Logo, existe um P_4 induzido pelos vértices b, a, x e d , como mostra a figura 3.18.

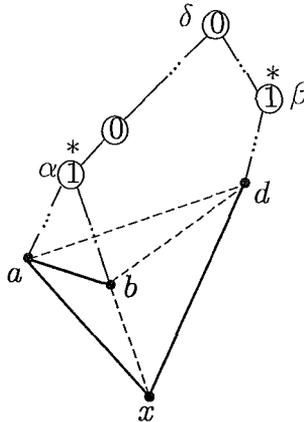


Figura 3.18: Ilustração do P_4 formado pelos vértices b, a, x e d , referente ao caso 1 do item (vi)

(2) δ é um nó tipo-1;

Seja c uma folha descendente do pai de α que não é descendente a α .

Pela propriedade 2.7, temos que $(a, b) \in E(G)$, $(a, e) \in E(G)$, $(b, e) \in E(G)$, $(c, e) \in E(G)$, $(a, c) \notin E(G)$ e $(b, c) \notin E(G)$.

(2.1) c é adjacente a x ;

Neste caso, existe um P_4 induzido pelos vértices b, a, x e c , como mostra a figura 3.19.

(2.2) c é não-adjacente a x ;

Neste caso, existe um P_4 induzido pelos vértices c, e, a e x , como mostra a figura 3.20.

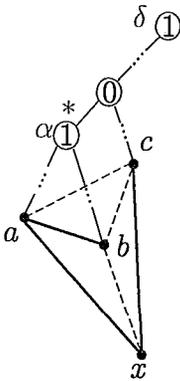


Figura 3.19: Ilustração do P_4 formado pelos vértices b, a, x e c , referente ao caso 2.1 do item (vi)

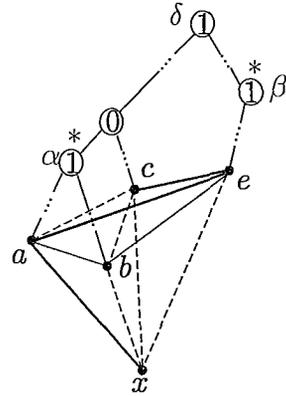


Figura 3.20: Ilustração do P_4 formado pelos vértices c, e, a e x , referente ao caso 2.2 do item (vi)

(\Leftarrow) Suponhamos que uma das condições seja satisfeita.

Seja G um cografo e T sua co-árvore correspondente.

Provaremos esta parte da demonstração pela construção de T' , a co-árvore correspondente a $G + x$.

(1) Se a primeira condição for satisfeita (M é vazio), temos então 3 casos a serem analisados;

(1.1) Se G e $G + x$ são ambos conexos então x pode ser adicionado como um filho da raiz R , como mostra a figura 3.21.

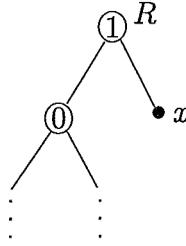


Figura 3.21: Ilustração do caso 1.1 da demonstração do teorema 3.1

(1.2) Se G e $G + x$ são ambos desconexos então x pode ser adicionado como um filho do único filho da raiz R , como mostra a figura 3.22.

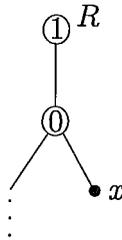


Figura 3.22: Ilustração do caso 1.2 da demonstração do teorema 3.1

(1.3) Se $G + x$ é desconexo, mas G é conexo então a raiz de R e x serão filhos do único nó que é filho da nova raiz R' , como mostra a figura 3.23.

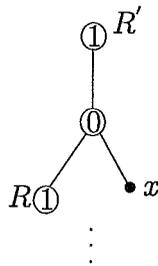


Figura 3.23: Ilustração do caso 1.3 da demonstração do teorema 3.1

(2) Se a segunda condição for satisfeita, temos que M é diferente de vazio, isto é, existe um nó marcado $\alpha \in M$ com o maior nível na árvore;

Seja A os filhos de α que foram marcados e, subsequentemente, desmarcados pelo procedimento *Marcar*. Analogamente, seja B os filhos de α que nem foram marcados pelo procedimento *Marcar*. Como $\alpha \in M$ temos que $|A| \geq 1$ e $|B| \geq 1$, pois $f(\alpha) \neq mf(\alpha)$.

A construção da co-árvore T é feita com base na análise dos dois seguintes casos:

Caso 1: α é um nó tipo-0;

Neste caso, os elementos de A e B ou são folhas ou são nós tipo-1.

(2.1.1) Se $|A| = 1$ e $a \in A$ é uma folha então adicionamos um novo nó tipo-1 no lugar de a e fazemos a e x filhos deste nó, como mostra a figura 3.24.

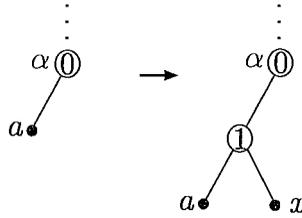


Figura 3.24: Ilustração do caso 2.1.1 da demonstração do teorema 3.1

(2.1.2) Se $|A| = 1$ e $a \in A$ é um nó tipo-1 então simplesmente adicionamos x como um novo filho de a , como mostra a figura 3.25.

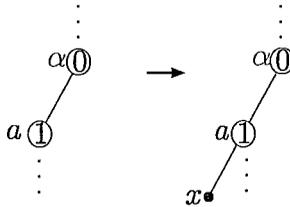


Figura 3.25: Ilustração do caso 2.1.2 da demonstração do teorema 3.1

(2.1.3) No caso em que $|A| > 1$, removemos todos os elementos de A de α e adicionamos um novo nó tipo-1 neste lugar. Os filhos deste novo nó serão o vértice x e um novo nó tipo-0 com os elementos de A como filhos, como mostra a figura 3.26.

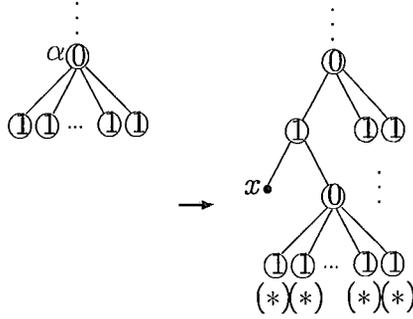


Figura 3.26: Ilustração do caso 2.1.3 da demonstração do teorema 3.1

Caso 2: α é um nó tipo-1.

Neste caso, os elementos de A e B ou são folhas ou são nós tipo-0.

(2.2.1) Se $|B| = 1$ e $b \in B$ é uma folha então adicionamos um novo nó tipo-0 no lugar de b e fazemos b e x filhos deste nó, como mostra a figura 3.27.

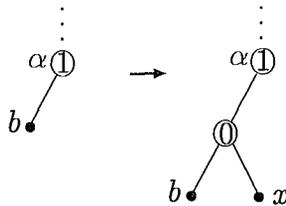


Figura 3.27: Ilustração do caso 2.2.1 da demonstração do teorema 3.1

(2.2.2) Se $|B| = 1$ e $b \in B$ é um nó tipo-0 então adicionamos x como um novo filho de b , como mostra a figura 3.28.

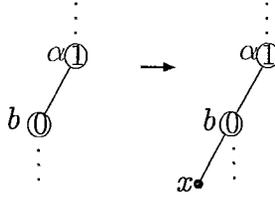


Figura 3.28: Ilustração do caso 2.2.2 da demonstração do teorema 3.1

(2.2.3) Se $|B| > 1$ então retiramos todos os elementos de B e adicionamos um novo nó tipo-0 neste lugar. Os filhos deste novo nó serão o vértice x e um novo nó tipo-1 com os elementos de B como filhos, como mostra a figura 3.29. Para vermos que T' é uma representação correta de

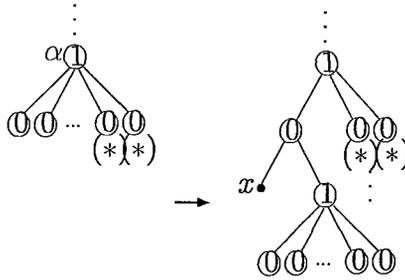


Figura 3.29: Ilustração do caso 2.2.3 da demonstração do teorema 3.1

$G + x$, observamos que as alterações de T refletem somente nas adjacências de x com os vértices da subárvore enraizada por α , e o fato de termos um caminho alternante legítimo de α até R nos garante que todas as outras adjacências de x estão representadas corretamente. Os vértices de G preservam as adjacências em $G + x$. \square

A seguir, descrevemos o algoritmo que reconhece cografos, cuja entrada consiste de um grafo G com seus vértices arbitrariamente indexados v_1, \dots, v_n .

Algoritmo 3.2: Reconhece-cografos

```
1 Criar um novo nó tipo-1  $R$  ;
2 se  $(v_1, v_2) \in E(G)$  então
3     adicionar  $v_1, v_2$  como filhos de  $R$  ;
4     Senão criar um novo nó,  $N$ , tipo-0 e adicionar  $N$  como filho de  $R$  ;
5         adicionar  $v_1, v_2$  como filhos de  $N$  ;
6 para  $x = v_3$  até  $v_n$  faça
7     Marcar( $x$ ) ;
8     se todos os nós de  $T$  foram marcados e desmarcados então
9         adicionar  $x$  como filho de  $R$  ;
10        vá para o próximo vértice  $x$  ;
11    se nenhum nó de  $T$  foi marcado então
12        se  $f(R) = 1$  então
13            adicionar  $x$  como filho do único filho de  $R$  ;
14            senão criar um novo nó tipo-1,  $R$ , com um filho (um novo nó tipo-0) e
15                dois netos:  $x$  e a antiga raiz  $R$  ;
16                vá para o próximo vértice  $x$  ;
17     $u :=$  Encontrar-nó; // Este nó é encontrado através do procedimento 3.3
18    se rótulo( $u$ ) = 0 (1) então
19        se  $|A| = 1$  ( $|B| = 1$ ) então
20            se  $w \in A$  ( $w \in B$ ) é uma folha então
21                adicionar um novo nó tipo-1 (tipo-0) no lugar de  $w$  e torne  $w$  e  $x$ 
22                filhos deste nó ;
23                senão adicionar  $x$  como um novo filho de  $w$  ;
24            senão remover todos os elementos de  $A(B)$  a partir de  $u$  e adicioná-los
25            como filhos de um novo nó  $y$  com rótulo( $y$ ) = rótulo( $u$ ) ;
26        se  $u$  é um nó tipo-0 então
27            adicionar um novo nó tipo-1 como um filho de  $u$  e os filhos deste novo
28            nó tipo-1 são  $x$  e  $y$  ;
29            senão remover  $u$  de seu pai e adicionar  $y$  em seu lugar ;
30            adicionar um novo nó tipo-0 como um filho de  $y$  e os filhos deste
31            novo nó tipo-0 são  $x$  e  $u$  ;
32        vá para o próximo vértice  $x$  ;
```

Procedimento Encontrar-nó

```
1 // Inicializar e verificar a raiz.
2  $y := \emptyset$  ;
3 se  $R$  não foi marcado então
4    $G + x$  não é um cografo; //condição (iii)
5 senão se  $mf(R) \neq f(R) - 1$  então
6    $y := R$ ;
7 desmarcar  $R$ ,  $mf(R) := 0$ ,  $u := w$  e  $w := R$  ;
8 // Checa se o caminho de  $u$  até  $w$  é um caminho alternante legítimo.
9 enquanto existirem vértices marcados em  $T$  faça
10 escolher arbitrariamente um vértice marcado  $u$  ;
11 se  $y \neq \emptyset$  então
12    $G + x$  não é um cografo; //condições (i) ou (ii)
13 se rótulo ( $u$ ) = 1 então
14   se  $mf(u) \neq f(u) - 1$  então
15      $y := u$  ;
16   se pai( $u$ ) é marcado então
17      $G + x$  não é um cografo; //condições (i) ou (vi)
18   senão  $t := avô(u)$  ;
19 senão  $y := u$  e  $t := pai(u)$  ;
20 desmarcar  $u$  e  $mf(u) := 0$  ;
21 //Verifica se o caminho de  $u$  a  $w$  faz parte do caminho alternante legítimo de  $u$  a  $R$ 
22 enquanto  $t \neq w$  faça
23 se  $t = R$  então
24    $G + x$  não é um cografo; //condição (iv)
25 se  $t$  não está marcado então
26    $G + x$  não é um cografo; //condições (iii) ou (v) ou (vi)
27 se  $mf(t) \neq f(t) - 1$  então
28    $G + x$  não é um cografo; //condição (ii)
29 se pai( $t$ ) está marcado então
30    $G + x$  não é um cografo; //condição (i)
31 desmarcar  $t$  ,  $mf(t) := 0$  e  $t := avô(t)$  ;
32  $w := u$  ;
```

Para melhor entendimento do algoritmo 3.2, vale mencionar que u denota o nó de maior nível examinado até o momento; w denota o nó tipo-1 de maior nível examinado antes de u ; y representa um nó tipo-1 marcado que não é propriamente marcado ou um nó tipo-0 marcado, caso exista em T . Além disso, denotamos por A o conjunto de filhos de u que foram marcados e, subsequentemente, desmarcados e B o conjunto de filhos de u que não foram marcados.

Observe que na linha 16 do algoritmo 3.2 fazemos uma chamada ao procedimento 3.3 que, encontra u (i.e., o vértice marcado de maior nível de T), caso $G + x$ seja um cografo.

Exemplo: Execução do algoritmo para o caso em que $G+x$ é cografo

Como ilustração, vamos executar o algoritmo 3.2 para verificar se $G + x$ é um cografo, dado que G é um cografo com sua co-árvore correspondente T , como mostra a figura 3.30. Além disso, x é um novo vértice adicionado ao grafo G adjacente aos vértices a, d, e e f , como mostra a figura 3.31 (a).

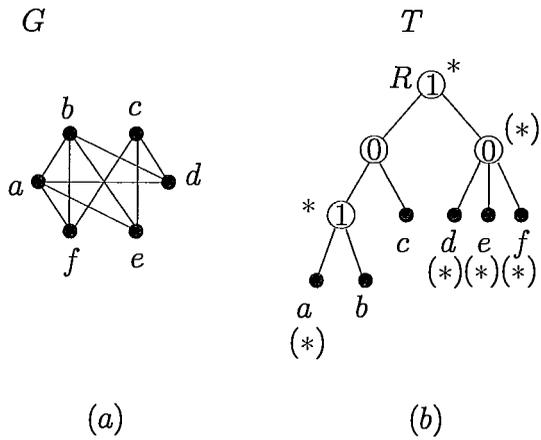
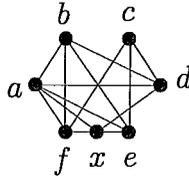
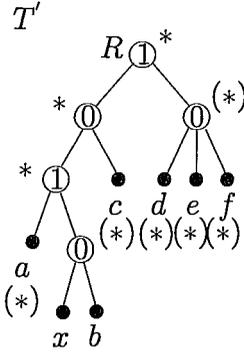


Figura 3.30: O cografo G e sua co-árvore T . Os símbolos $*$ e $(*)$ referem-se ao procedimento *Marcar*

$$G' = G + x$$



(a)



(b)

Figura 3.31: Cografo $G' = G + x$ e sua co-árvore T'

Ao final do procedimento *Marcar*, os únicos vértices marcados são os vértices $\text{pai}(a)$ e R , enquanto que os vértices a , d , e , f e $\text{pai}(d)$ são vértices que foram marcados e, subsequentemente, desmarcados, como mostra a figura 3.30 (b).

O procedimento 3.3 desmarca R , e $u := w := R$ no bloco 2 – 7. Além disso, no bloco 9 – 31 existe apenas um vértice marcado para ser examinado: $u := \text{pai}(a)$. Como u é um nó tipo-1 com $mf(u) = d(u) - 1$ e o $\text{pai}(u)$ não está marcado, então, pelo bloco 11 – 20, $t := R$ e $\text{pai}(a)$ é desmarcado. O bloco 22 – 31 não é executado, já que $t = w = R$. Na linha 32, atualizamos $w := \text{pai}(a)$ para execução da próxima volta do bloco 9 – 31, mas esta volta não será executada, uma vez que não temos mais nenhum vértice marcado. O procedimento 3.3 termina com $u = \text{pai}(a)$, indicando, assim, que $G + x$ é um cografo. Voltando para o bloco 17 – 26 do algoritmo 3.2, identificamos que $A = \{a\}$ e $B = \{b\}$. Como u é nó tipo-1, $|B| = 1$ e $b \in B$ é uma folha então, pelo algoritmo, temos apenas que adicionar um novo nó tipo-0 no lugar de b e tomar b e x filhos desse nó tipo-0. A co-árvore resultante T' correspondente ao cografo $G + x$ é mostrada na figura 3.31 (b).

Execução do algoritmo no caso em que $G' + z$ não é um cografo

Agora, consideramos o grafo G' da figura 3.31 (a), com adição do vértice z adjacente a a, c, d, e, f .

Após a execução do procedimento *Marcar*, os vértices $\text{pai}(a)$, $\text{pai}(c)$ e R permanecem marcados, e os vértices a, c, d, e, f e $\text{pai}(d)$ são os vértices que foram marcados e, subsequentemente, desmarcados.

O procedimento 3.3 desmarca R , $u := w$ e $w := R$ no bloco 2–7. Assuma que $\text{pai}(c)$ tenha sido escolhido no bloco 9–31, i.e., $u := \text{pai}(c)$. Como $\text{pai}(c)$ é um nó tipo-0 então finalizamos o bloco 11–20 com $y = \text{pai}(c)$, $t = R$ e $\text{pai}(c)$ desmarcado. O bloco 22–31 não será executado, pois $w = t = R$. Na linha 32, atualizamos $w := \text{pai}(c)$, logo, temos $u := \text{pai}(a)$ e verificamos que, no bloco 11–20, $G' + z$ não é um cografo, já que $y \neq \emptyset$, indicando assim que $M \setminus \{\alpha\}$ contém um nó tipo-0 (condição (i) da prova do teorema).

Se $u := \text{pai}(a)$ fosse escolhido no bloco 9–31 ao invés de $\text{pai}(c)$, também concluiríamos que $G' + z$ não é um cografo no bloco 11–20, onde rótulo(u)= 1 e $\text{pai}(u)$ está marcado.

A figura 3.32 mostra que $G' + z$ não é um cografo, pois possui um P_4 como subgrafo, conforme indicado.

$G' + z$

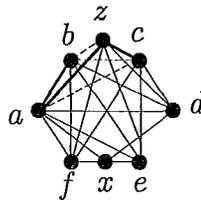


Figura 3.32: O grafo $G' + z$ não é um cografo (os vértices b, a, z e c formam um P_4)

3.2 Complexidade

A análise para o algoritmo é baseada no tempo $O(d_G(x))$ gasto para adicionar x a T , onde $d_G(x)$ é o grau de x em $G + x$.

Como todos os nós internos de T , exceto possivelmente a raiz, têm pelo menos dois filhos, sabemos que o procedimento *Marcar* irá examinar apenas $O(d_G(x))$ nós. Para cada um desses nós, o procedimento é feito em tempo constante, e portanto o procedimento *Marcar* pode ser executado em tempo $O(d_G(x))$. É claro que $|M|$ é também limitado por $O(d_G(x))$, e como o procedimento 3.3 examina cada nó marcado em tempo constante, o tempo para este procedimento é $O(|M|) = O(d_G(x))$.

Todas as três alterações (todos os nós de T foram marcados e desmarcados, nenhum nó de T foi marcado ou existem pelo menos um nó marcado e um nó que nem foi marcado) podem ser feitas em tempo constante.

No procedimento *Marcar* mantemos uma lista dos filhos que foram marcados e, subsequentemente, desmarcados, e portanto, eles podem ser acessados em tempo $O(d_G(x))$. Logo, todas as três modificações podem ser executadas em tempo $O(d_G(x))$.

Assim, obtém-se o tempo necessário para a realização de cada operação, e o tempo para o algoritmo de reconhecimento de um cografo é $O(n + m)$, sendo portanto linear.

3.3 Alguns problemas algorítmicos

Devido à representação única dos cografos através da co-árvore, diversos problemas que são de difícil solução para grafos em geral, tornam-se polinomiais para cografos, conforme [26], [10], e [11].

Como exemplo, podemos citar os problemas de encontrar as cliques maximais e máximas de um cografo, assim como os conjuntos independentes maximais e máximos de cografos.

A seguir, descrevemos como utilizar a co-árvore T correspondente a um cografo G para resolver o número de cliques maximais, o tamanho da clique máxima e o número cromático, e uma fórmula geradora para o conjunto das cliques maximais.

Os algoritmos que apresentaremos são executados a partir das folhas, percorrendo a árvore, e efetuando operações específicas em cada nó até alcançar a raiz, onde o resultado apropriado é então obtido. Em cada nó interno é realizada uma determinada operação, de acordo com o rótulo do nó (tipo-0 ou tipo-1).

Os algoritmos de clique e conjunto independentes (maximais ou máximos) são ambos executados da mesma forma. A diferença entre eles se dá apenas na operação efetuada em cada nó da co-árvore, invertendo as operações realizadas nos nós tipo-1 com as operações nos nós tipo-0.

Algoritmo 3.4: Número de cliques maximais de G

Seja i um nó interno da co-árvore T e sejam i_1, \dots, i_r seus filhos. Denote por $G_{i_1}, G_{i_2}, \dots, G_{i_r}$ os subgrafos de G induzidos pelas folhas descendentes de i_1, \dots, i_r , respectivamente.

O algoritmo percorre os nós de T , partindo das folhas até a raiz. Ao analisar cada nó interno i , o algoritmo encontra as cliques maximais do subgrafo de G induzido pelas folhas descendentes de i .

No passo inicial, cada folha j é uma clique maximal do subgrafo de G induzido por j . No passo geral, ao analisar um nó interno i , o algoritmo já terá encontrado as cliques maximais dos subgrafos $G_{i_1}, G_{i_2}, \dots, G_{i_r}$. Para encon-

trar as cliques maximais do subgrafo G_i induzido pelas folhas descendentes de i , o algoritmo considera os dois seguintes casos:

- i é um nó tipo-0: neste caso, os subgrafos G_{i_1}, \dots, G_{i_r} são componentes conexos de G e, portanto, as cliques maximais de G_i são as cliques maximais de cada G_{i_k} , $k = 1, \dots, r$. Logo, o número de cliques maximais até o momento, i.e, do subgrafo G_i , será a soma do número de cliques maximais de seus filhos.

- i é um nó tipo-1: neste caso, os subgrafos G_{i_1}, \dots, G_{i_r} são componentes conexos de \bar{G} e, portanto, cada clique maximal de G_{i_p} é completamente adjacente a cada clique maximal de G_{i_q} , com $p \neq q$. Logo, o número de cliques maximais de G_i será o produto entre o número de cliques maximais de seus filhos.

Entrada: A co-árvore T

Algoritmo 3.4: Número de cliques maximais de G

```
1  Atribua o valor 1 para as folhas de  $T$  ;
2  //  $j$  é o nível da co-árvore
3  para  $i = j - 1$  até 0 faça
4    para cada nó interno do nível  $i$  faça
5      se o nó é tipo-1 então
6         $aux =$  multiplicação do valor de seus filhos ;
7      senão
8         $aux =$  soma do valor dos seus filhos ;
9      fim_se;
10   fim_para;
11 fim_para;
12 Atribua o  $aux$  para o nó atual ;
```

Algoritmo 3.5: Tamanho da clique máxima e o número cromático de G

Ao analisar um nó tipo-0, com r filhos, a clique máxima do subgrafo de G induzido por suas folhas descendentes será o tamanho da maior clique entre as cliques encontradas para seus filhos, isto é, $a = \max_{k=1}^r a_k$.

Ao analisar um nó tipo-1 com s filhos, a clique máxima do subgrafo de G induzido por suas folhas descendentes será aquela formada pela união das maiores cliques de cada filho. Então, o tamanho da clique máxima ao analisar um nó tipo-1 será a soma do tamanho das maiores cliques de cada filho.

Entrada: A co-árvore T

Algoritmo 3.5: Tamanho da clique máxima e o número cromático

```
1  Atribua o valor 1 para as folhas de  $T$  ;
2  //  $j$  é o nível da co-árvore
3  para  $i = j - 1$  até 0 faça
4      para cada nó interno do nível  $i$  faça
5          se o nó é tipo-1 então
6               $aux =$  soma do valor de seus filhos ;
7          senão
8               $aux =$  o valor máximo dentre seus filhos ;
9          fim_se;
10     fim_para;
11 fim_para;
12 Atribua o  $aux$  para o nó atual ;
```

Algoritmo 3.6: Fórmula geradora para o conjunto das cliques maximais de G

No que segue, o operador \wedge indica que seus operandos são adjacentes no cografo G , enquanto que o operador \vee indica que seus operandos são

não-adjacentes no cografo G . Então, a fórmula geradora indica todas as adjacências e não-adjacências do cografo, determinando, assim, as cliques maximais. Com esta fórmula geradora, é possível obter o tamanho da clique máxima, o número de cliques maximais e máximas, e todas as cliques maximais e máximas.

Temos que os filhos do nó tipo-0 são não-adjacentes entre si, então a fórmula é construída colocando o símbolo \vee entre cada um dos filhos, e os filhos do nó tipo-1 são adjacentes entre si, logo a fórmula é construída colocando o símbolo \wedge entre cada um dos filhos.

Entrada: A co-árvore T

Algoritmo 3.6: Fórmula geradora para o conjunto das cliques maximais de G

```

1 //  $j$  é o nível da co-árvore
2 para  $i = j - 1$  até 0 faça
3   para cada nó interno do nível  $i$  faça
4     se o nó é tipo-1 então
5        $aux =$  conjunção das fórmulas geradoras de seus filhos
        usando  $\wedge$  ;
6     senão
7        $aux =$  disjunção das fórmulas geradoras de seus filhos
        usando  $\vee$  ;
8     fim_se;
9   fim_para;
10 fim_para;
11 Atribua o  $aux$  para o nó atual ;

```

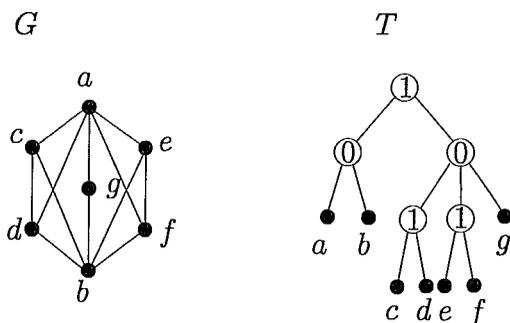


Figura 3.33: Cografo G e sua respectiva co-árvore T

Como exemplo, considere G o cografo da Figura 3.33. A Figura 3.34 mostra graficamente a execução dos algoritmos 3.4, 3.5 e 3.6 no cografo G .

- Número de cliques maximais: 6;
- Tamanho da clique máxima e número cromático: 3;
- Fórmula geradora: $(a \vee b) \wedge [(c \wedge d) \vee (e \wedge f) \vee g]$. A fórmula indica as cliques maximais: $\{a, c, d\}$, $\{b, c, d\}$, $\{a, e, f\}$, $\{b, e, f\}$, $\{a, g\}$ e $\{b, g\}$, e as cliques máximas que são as quatro primeiras.

Como dito anteriormente, todos os algoritmos são executados da mesma maneira, a única diferença entre eles é na operação efetuada em cada nó da co-árvore. Se invertermos as operações dos nós tipo-1 com as operações dos nós tipo-0 nos algoritmos citados acima, obtemos um resultado correspondente para os kernels e conjuntos independentes máximos.

Na figura 3.35 ilustramos graficamente a execução dos algoritmos 3.4, 3.5 e 3.6 quando modificados para kernels, encontrando, assim, o número de kernels, o tamanho do conjunto independente máximo e a fórmula geradora para o conjunto de kernels do cografo da figura 3.33.

- Número de kernels: 5;

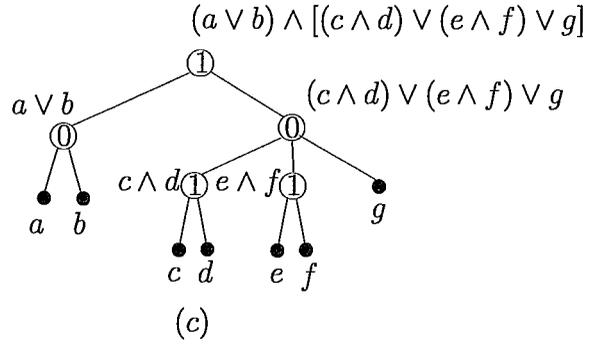
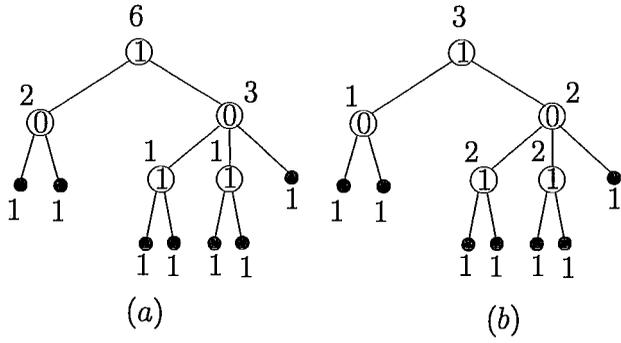


Figura 3.34: Resultados dos algoritmos para as cliques maximais e máximas: (a) Algoritmo 3.4; (b) Algoritmo 3.5; (c) Algoritmo 3.6;

- Tamanho do conjunto independente máximo: 3;
- Fórmula geradora: $(a \wedge b) \vee [(c \vee d) \wedge (e \vee f) \wedge g]$. A fórmula indica os kernels: $\{c, e, g\}$, $\{c, f, g\}$, $\{d, e, g\}$, $\{d, f, g\}$ e $\{a, b\}$, e os conjuntos independentes máximos são os quatro primeiros.

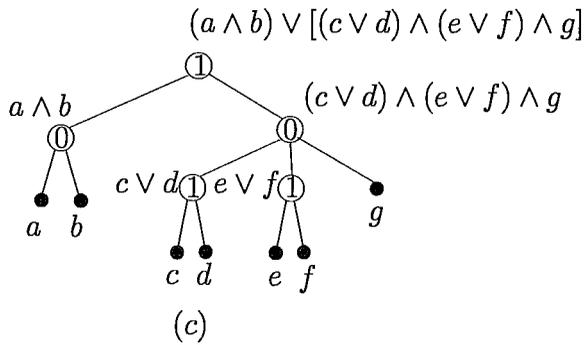
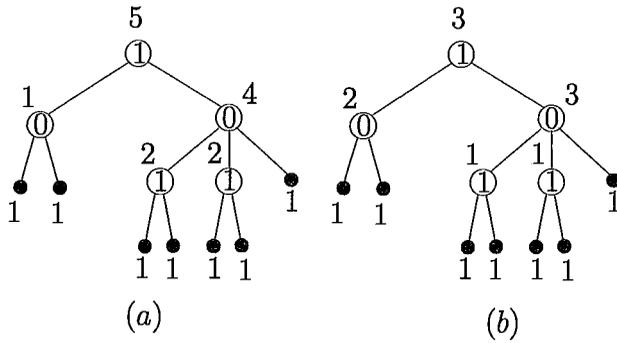


Figura 3.35: Resultados dos algoritmos para os kernels e conjuntos independentes máximos: (a) Algoritmo 3.4; (b) Algoritmo 3.5; (c) Algoritmo 3.6;

Capítulo 4

Cografo- (k, l)

Um problema combinatório muito estudado atualmente é o problema da partição em grafos, que consiste em verificar se o conjunto de vértices de um dado grafo G pode ser particionado em subconjuntos V_1, \dots, V_k , com cada um deles satisfazendo certas propriedades. Pode-se, por exemplo, exigir que cada subconjunto V_i seja uma clique ou um conjunto independente (restrições internas); ou ainda, exigir propriedades entre cada par de subconjuntos (restrições externas); por exemplo, que o pares V_i, V_j sejam completamente adjacentes ou não-adjacentes, isto é, cada vértice, v_i , do subconjunto V_i , é adjacente ou não-adjacente a todos os vértices v_j do subconjunto V_j , respectivamente.

Um problema atual que se insere neste contexto de partição em grafos é o problema dos grafos- (k, l) , que consiste em reconhecer se um dado grafo G pode ser particionado em k conjuntos independentes e l cliques (sem restrições externas).

Neste capítulo definimos os grafos- (k, l) e apresentamos alguns resultados no contexto dos cografos- (k, l) .

4.1 Grafos- (k, l)

Em [3], Brandstädt definiu uma nova classe de grafos, a classe dos grafos- (k, l) , classe esta que corresponde a uma generalização dos grafos split (grafos cujo conjunto de vértices pode ser particionado em um conjunto independente e uma clique) [17], como pode ser visto pela definição a seguir.

Definição 4.1. *Um grafo G é grafo- (k, l) , ou simplesmente (k, l) , se o conjunto de vértices de G pode ser particionado em k conjuntos independentes e l cliques.*

Na definição dada acima, é importante ressaltar que as cliques não são necessariamente maximais, além disso, alguns dos k conjuntos independentes ou das l cliques podem ser vazios.

Ainda em [3, 4], Brandstädt apresentou um algoritmo polinomial para reconhecer as classes $(2, 1)$, $(1, 2)$ e $(2, 2)$, e mostrou que reconhecer os grafos- (k, l) para $k \geq 3$ ou $l \geq 3$ é um problema *NP*-completo. Feder *et al.*[14] também propuseram algoritmos polinomiais para essas classes, que surgiram como sub-produto de algoritmos de partição em subgrafos densos e esparsos.

Uma vez que o reconhecimento de grafos- (k, l) é *NP*-completo para $k \geq 3$ ou $l \geq 3$, o estudo de grafos- (k, l) para certas subclasses de grafos torna-se uma alternativa na tentativa de reconhecer os grafos- (k, l) .

Muitos trabalhos já foram desenvolvidos nesta linha, a saber o reconhecimento de grafos cordais- (k, l) , isto é, dado um grafo cordal G verificar se G é (k, l) . Em [22] uma caracterização e um algoritmo de reconhecimento com complexidade $O(n(n + m))$ para os grafos cordais- (k, l) são dados. Outros resultados no contexto de partição de grafos cordais, onde restrições externas são consideradas, pode ser encontrado em [14].

4.2 Caracterização dos cografos- (k, l)

Este trabalho visa resolver o problema dos grafos- (k, l) quando restrito à classe dos cografos, caracterizando e reconhecendo os cografos- (k, l) .

Uma propriedade importante dos cografos (e portanto, cografos- (k, l)) é que estes possuem a propriedade de complementaridade, isto é, um grafo G é (k, l) se e somente se \overline{G} é um grafo- (l, k) . Esta propriedade é de fácil verificação, já que o complemento de um grafo é também um grafo e se G pode ser particionado em k conjuntos independentes e l cliques então o complemento de G pode ser particionado em k cliques e l conjuntos independentes.

Observação 4.2. *Dado um grafo G qualquer, temos que se G é (k, l) então $G \cup S$ é $(k + 1, l)$, onde S é um conjunto independente. Da mesma forma, se G é (k, l) então $G \cup K$ é $(k, l + 1)$, onde K é uma clique.*

A recíproca da observação 4.2 vale, em particular, para cografos, resultado este mostrado por Demange *et al.* [13] e apresentado na subseção 4.2.1.

Antes de apresentarmos nosso resultado principal, vamos estabelecer a seguinte notação:

Sejam p, r inteiros positivos, denotamos por p^*K_r o grafo formado por p cópias de um K_r , com algumas arestas proibidas e outras opcionais entre os K_r 's. Mais precisamente, para $i = 1, \dots, p$, C_i denota uma clique de tamanho r , e v_i^j os vértices da clique i , para $j = 1, \dots, r$. As únicas arestas proibidas entre duas cliques C_i e $C_{i'}$ são as arestas do tipo $(v_i^j, v_{i'}^j)$, $\forall j = 1, \dots, r$, $\forall i, i' = 1, \dots, p$, com $i \neq i'$. As demais arestas entre as cliques são opcionais, i.e, podem existir ou não.

Observe que como algumas arestas do p^*K_r podem ou não existir, um p^*K_r corresponde não apenas a um grafo, mas a uma família de subgrafos.

Como exemplo, a figura 4.1 representa os grafos da família 2^*K_2 .

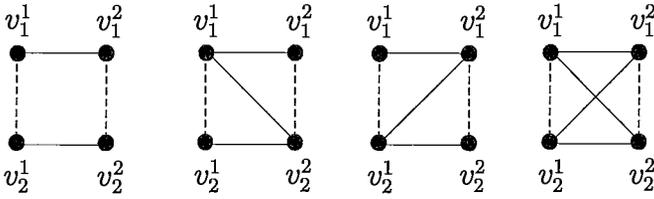


Figura 4.1: 2^*K_2

A seguir, algumas propriedades importantes de cografos são apresentadas:

Lema 4.3. [12] *Todo componente conexo de um cografo sem triângulo, isto é, sem K_3 induzido, é um grafo bipartido completo.*

Demonstração. Seja G' um componente conexo de um cografo G sem triângulo.

Devemos mostrar que G' é bipartido completo.

Pelo lema 2.9, temos que G' é um cografo, e pelo teorema 2.12, G' não possui P_4 e, portanto, não possui nenhum ciclo C_k , para $k \geq 5$. Como, por hipótese, G' não possui triângulos, então G' não possui nenhum ciclo ímpar, logo pela caracterização de grafo bipartido (G' é bipartido se e somente se não possui ciclos ímpares), G' é bipartido.

Sejam A e B as bipartições de G' , i.e, $V(G') = A \cup B$ (lembre-se que A e B são conjuntos independentes).

A prova de que G' é bipartido completo segue por indução em n , o número de vértices de G' .

Para $n \leq 3$ o resultado segue trivialmente, já que G' é conexo.

Suponhamos que o resultado seja verdadeiro para grafos com menos que n vértices. Seja G' um grafo bipartido com $|V(G')| = n$.

Suponha por absurdo, que existem dois vértices $a \in A$ e $b \in B$ tal que $(a, b) \notin E(G')$. Como $|V(G')| \geq 4$ e $G' \setminus \{b\}$ é bipartido completo (pela

hipótese indutiva), então existe pelo menos um vértice $b' \in B$, $b \neq b'$, tal que $(a, b') \in E(G')$ (na verdade, a é adjacente a todos os vértices de B). Por outro lado, G' é conexo, logo existe $a' \in A$, $a' \neq a$ tal que $(a', b) \in E(G')$, como $(a, a') \notin E(G')$ e $(b, b') \notin E(G')$, então $G[a, b, a', b']$ é um P_4 induzido. Contradição, logo $(a, b) \in E(G')$. \square

Lema 4.4. *Todo componente conexo de um cografo sem K_p é $(p-1)$ -partido.*

Demonstração. Seja G' um componente conexo de um cografo G sem K_p .

Como cografos são perfeitos e G' não possui K_p , $\omega(G') \leq p-1$. Logo, $\chi(G') \leq p-1$.

Portanto, G' é $(p-1)$ -partido. \square

O resultado a seguir, contribuição desta dissertação, auxilia na demonstração da nova caracterização de cografos, dada por Demande *et al.* [13].

Lema 4.5. [6] *Sejam G um cografo e S^* um conjunto independente maximal de G .*

Se $G[V \setminus S^]$ contém um K_k como subgrafo então G contém um K_{k+1} como subgrafo.*

Demonstração. Sejam G um cografo e S^* um conjunto independente maximal de G .

Suponha que $G[V \setminus S^*]$ possui um K_k como subgrafo. Neste caso, são necessários pelo menos k conjuntos independentes para particionar $G[V \setminus S^*]$ em conjuntos independentes. Sejam $S_1, S_2, \dots, S_k \subseteq V \setminus S^*$, onde cada S_i , $i = 1, \dots, k$ contém um vértice do K_k , como mostra a figura 4.2.

A prova segue por indução em k (k denota o tamanho do subgrafo completo):

Como base da indução é fácil ver que para cliques de tamanho um o resultado segue imediatamente.

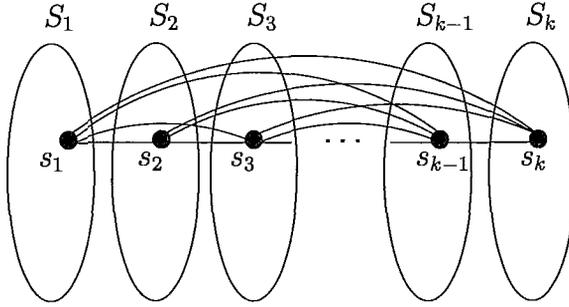


Figura 4.2: $G[K_k] \subseteq G[V \setminus S^*]$

Suponhamos que o resultado é válido para cliques de tamanho, no máximo, $k - 1$, isto é, se $G[V \setminus S^*]$ contém um K_i então G contém um K_{i+1} como subgrafo, $\forall i = 1, \dots, k - 1$.

Suponha que $G' = G[V \setminus S^*]$ contém um K_k como subgrafo. Pelo mesmo argumento explicado anteriormente, são necessários, pelo menos, k conjuntos independentes para particionar G' em conjuntos independentes.

Sejam $S_1, S_2, \dots, S_k \subseteq V \setminus S^*$, onde cada S_i contém um vértice do K_k . $G[S_1 \cup S_2 \cup \dots \cup S_{k-1}]$ contém um K_{k-1} e logo, pela hipótese indutiva, temos que $G[S^* \cup S_1 \cup \dots \cup S_{k-1}]$ contém um K_k como subgrafo. Denote por C tal subgrafo. Sejam $s_1, s_2, \dots, s_{k-1}, s_1^*$ os vértices de C , com $s_i \in S_i$, $\forall i = 1, \dots, k - 1$, e $s_1^* \in S^*$.

Daí, temos duas possibilidades a serem analisadas:

- (i) $(s_k, s_1^*) \in E(G)$.

Neste caso, o resultado segue imediatamente.

- (ii) $(s_k, s_1^*) \notin E(G)$

Neste caso, como S^* é maximal, s_k deve ser adjacente a algum outro vértice $s_2^* \in S^*$.

Como $(s_k, s_1^*) \notin E(G)$, então $(s_2^*, s_{k-1}) \in E(G)$, caso contrário, $G[s_1^*, s_{k-1}, s_k, s_2^*]$ induziria um P_4 , contradizendo o fato de G ser cografo, como pode ser visto pela figura 4.3.

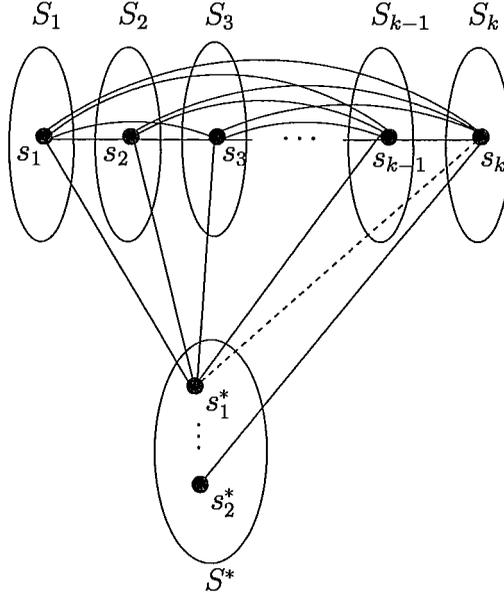


Figura 4.3: Ilustração do caso (ii), isto é, s_k é não-adjacente ao vértice s_1^*

Repetindo este mesmo procedimento para $G[s_1^*, s_i, s_k, s_2^*]$, concluímos que $(s_2^*, s_i) \in E(G)$, para $i = 1, \dots, k - 2$. Então, G contém um K_{k+1} como subgrafo.

Após analisadas as duas possibilidades podemos concluir que G contém um K_{k+1} como subgrafo. □

De maneira equivalente, temos o seguinte lema:

Lema 4.6. [6] *Sejam G um cografo e K uma clique maximal de G .*

Se $G[V \setminus K]$ contém um \overline{K}_k como subgrafo então G contém um \overline{K}_{k+1} como subgrafo.

4.2.1 Uma recente caracterização de cografos

Recentemente, uma nova caracterização de cografos, foi apresentada por [12]. Tal caracterização é de extrema importância para a caracterização dos cografos- (k, l) . Antes de enunciá-la, a seguinte notação faz-se necessária.

Definição 4.7. *Seja G um grafo. Denotamos por $\alpha_{k,l}(G)$ a cardinalidade máxima de um subgrafo de G que é (k, l) .*

Temos, por convenção, que $\alpha_{0,0}(G) = 0$. Em [12], Demange *et al.* denominaram $\alpha_{1,1}(G)$ como o número split-independente de G e passou a representar tal número simplesmente pela notação $\alpha_S(G)$.

Teorema 4.8. [12] *Seja $G = (V, E)$ um grafo, $G[V']$ um subgrafo de G e K um clique máxima de $G[V']$.*

As seguintes afirmações são equivalentes para todo grafo G :

(i) G é um cografo,

(ii) $\forall V' \subseteq V, \alpha_S(G[V']) = \alpha(G[V']) + \omega(G[V']) - 1,$

(iii) $\forall V' \subseteq V, \text{ se } G[V'] \text{ é um grafo-}(k, l), \text{ com } l \geq 1 \text{ então } G[V' \setminus K] \text{ é um grafo-}(k, l - 1).$

Demonstração. (i) \Rightarrow (ii) Seja G um cografo e $G' \subseteq G$.

Pelo teorema 2.12, G satisfaz a propriedade clique-kernel, i.e., para toda clique maximal C de G e todo kernel K de G , $|C \cap K| = 1$. Portanto, não existe nenhum par disjunto de clique maximal e kernel de G , e obviamente, esta propriedade é hereditária.

Logo, $\alpha_{1,1}(G[V']) = \alpha(G[V']) + \omega(G[V']) - 1.$

(ii) \Rightarrow (i) Seja $\alpha_{1,1}(G[V']) = \alpha(G[V']) + \omega(G[V']) - 1, \forall V' \subseteq V$

Suponhamos, por absurdo, que G contém um P_4 como subgrafo induzido.

Observe que $\alpha(P_4) = 2$, $\omega(P_4) = 2$ e $\alpha_s(P_4) = 4 = \alpha(P_4) + \omega(P_4)$, contradizendo o fato de $\alpha_s(G[V']) = \alpha(G[V']) + \omega(G[V']) - 1$. Contradição, logo G é cografo.

(i) \Rightarrow (iii) Seja G um cografo.

Suponhamos que $\forall V' \subseteq V$, $G[V']$ é um grafo- (k, l) .

Tome $(S_1 \cup \dots \cup S_k \cup C_1 \cup \dots \cup C_l)$ como sendo uma partição de V' em k conjuntos independentes e l cliques.

Seja K uma clique máxima de $G[V']$. K pode ser decomposta em duas partes: K_C e K_S , de forma que $K = K_C \cup K_S$, $K_C = K \cap (C_1 \cup \dots \cup C_l)$ e $K_S = K \cap (S_1 \cup \dots \cup S_k)$.

Temos que K_S tem no máximo k vértices, já que $\forall i = 1, \dots, k$, S_i é um conjunto independente. Sem perda de generalidade, podemos então escrever $K_S = \{x_1, \dots, x_s\}$, onde $x_i \in S_i$ e $s \leq k$.

Uma ilustração da prova é dada na figura 4.4, onde a clique K é mostrada pelo conjunto com o sombreamento mais claro e as linhas pontilhadas descrevem os conjuntos independentes de $G[V']$.

Por outro lado, existe um conjunto de vértices L com $|L| \leq |K_S| = s$ tal que $K_C \cup L$ é clique maximal de $C_1 \cup \dots \cup C_l$.

Daí, pela contrapositiva do lema 4.6, podemos dizer que $(C_1 \cup \dots \cup C_l) \setminus (K_C \cup L)$ é um grafo- $(0, l - 1)$, pois $(C_1 \cup \dots \cup C_l)$ é um grafo- $(0, l)$.

Sabemos que $(S_{s+1} \cup \dots \cup S_k)$ é um grafo- $(k - s, 0)$, então é suficiente mostrar que $L \cup [(S_1 \cup \dots \cup S_s) \setminus K_S]$ é um grafo- $(s, 0)$ para a obtenção do teorema, isto é, $G[V' \setminus K]$ é um grafo- $(k, l - 1)$.

Para tal, suponhamos, por absurdo, que $L \cup [(S_1 \cup \dots \cup S_s) \setminus K_S]$ não é um grafo- $(s, 0)$, isto implica em dizer que $L \cup [(S_1 \cup \dots \cup S_s) \setminus K_S]$ contém uma clique K' de tamanho $s + 1$.

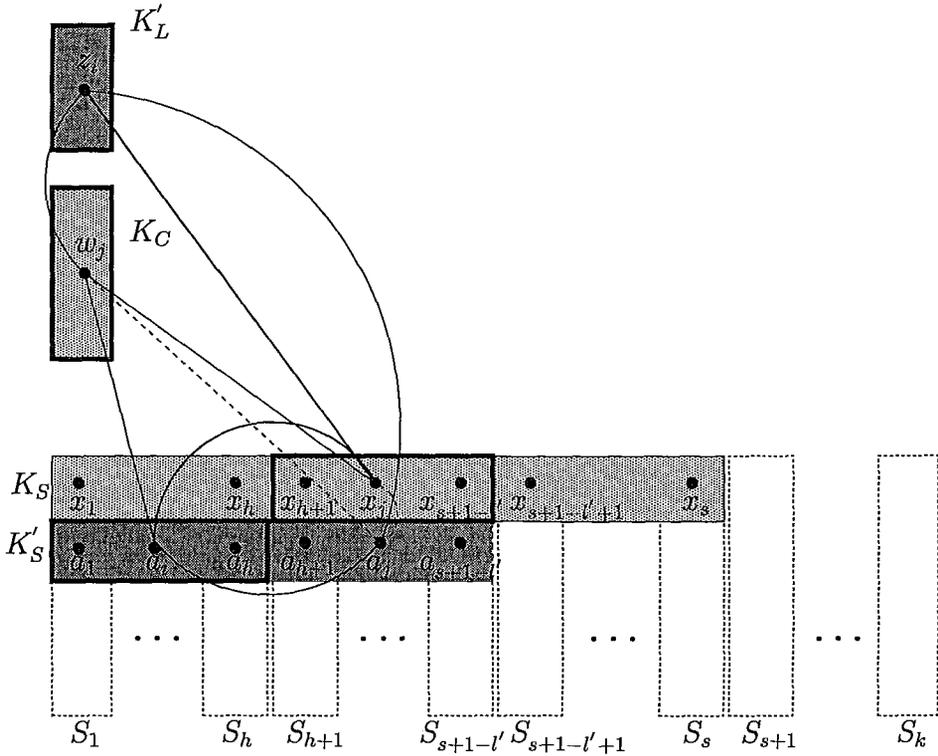


Figura 4.4: Ilustração da demonstração do teorema 4.8, implicação (i) \Rightarrow (iii).

Temos que $K' \cap L \neq \emptyset$, já que $S_1 \cup S_2 \cup \dots \cup S_s$ é um grafo- $(s, 0)$ e $|K'| = s + 1$. Da mesma forma, $K' \cap [(S_1 \cup S_2 \cup \dots \cup S_s) \setminus K_S] \neq \emptyset$, pois $|K'| > |L|$.

Particionaremos, agora, a clique K' em duas partes: K'_L e K'_S , onde $K' = K'_L \cup K'_S$, $K'_L = K' \cap L = \{z_1, \dots, z_{l'}\}$ e $K'_S = K' \cap [(S_1 \cup \dots \cup S_s) \setminus K_S] = \{a_1, \dots, a_h, a_{h+1}, \dots, a_{s+1-l'}\}$, onde $\forall i \leq h$, $\{a_i\} \cup K_C$ é uma clique e $\forall i \geq h + 1$, $\exists w_i \in K_C$ tal que $(a_i, w_i) \notin E$ (possivelmente $h = s + 1 - l'$).

Na figura 4.4, K' é mostrada pelo conjunto com o sombreamento mais escuro.

Observemos que $\forall h, h+1 \leq j \leq s+1-l', \forall i \leq l'$, se $a_j z_i w_j x_j$ não for um P_4 então a aresta $(x_j, z_i) \in E$. Além disso, $\forall i \leq h, \forall j \geq h+1$, se $a_j a_i w_j x_j$ não é um P_4 então a aresta $(a_i, x_j) \in E$.

As observações acima são fáceis de serem verificadas já que G não contém o P_4 como subgrafo.

Observe, também, que se $K_C \cup K'_L \cup \{x_1, \dots, x_{s+1-l'}\}$ tem $|K_C| + s + 1$ vértices de forma que não seja uma clique então $h \geq 1$. A verificação desta observação é dada por: se $h = 0$ então $K_C \cup K'_L \cup \{x_1, \dots, x_{s+1-l'}\}$ é uma clique, pois os vértices $\{x_1, \dots, x_{s+1-l'}\}$ não existem, e $K_C \cup K'_L$ é uma clique, o que justifica a existência de ao menos um vértice no conjunto K_S .

É suficiente então ressaltar que as duas primeiras observações mostram que $K_C \cup K'_L \cup \{a_i, i \leq h\} \cup \{x_j, h+1 \leq j \leq s+1-l'\}$ é uma clique de cardinalidade $|K_C| + s + 1$, mas como $|K| = |K_C| + |K_S| = |K_C| + s$ então a cardinalidade de $K_C \cup K'_L \cup \{a_i, i \leq h\} \cup \{x_j, h+1 \leq j \leq s+1-l'\}$ é $|K| + 1$, o que contradiz a hipótese, pois K é uma clique máxima de $G[V']$.

Então, $L \cup [(S_1 \cup \dots \cup S_s) \setminus K_S]$ não contém uma clique de tamanho $s + 1$, logo $L \cup [(S_1 \cup \dots \cup S_s) \setminus K_S]$ é um grafo- $(s, 0)$, concluindo então que $G[V' \setminus K]$ é um grafo- $(k, l - 1)$.

(iii) \Rightarrow (i) Suponhamos, por absurdo, que G contém pelo menos um P_4 como subgrafo, formado pelas arestas (x, y) , (y, z) e (z, w) , como mostra a figura 4.5.

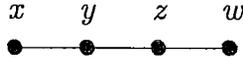


Figura 4.5: P_4 induzido pelos vértices x, y, z e w de $V(G)$

É fácil ver que o P_4 é um grafo-(1,1), onde a clique $C = \{y, z\}$ e o conjunto independente $K = \{x, w\}$. Tomando os vértices x e y como clique máxima teremos que $G[V(P_4) \setminus \{x, y\}]$ não é um grafo-(1,0).

Logo, G é um cografo. □

O principal resultado desta dissertação é uma generalização dos resultados a seguir [12].

4.2.2 Cografos-(1,2) e Cografos-(2,2)

Os próximos teoremas desta subsecção caracterizam os cografos-(1,2) e (2,2) e foram apresentados por Demange *et al.* [12].

Teorema 4.9. [12] *Seja G um cografo.*

*G é um grafo-(1,2) se e somente se não contém nenhum dos cografos da família 3^*K_2 , como mostra a Figura 4.6.*

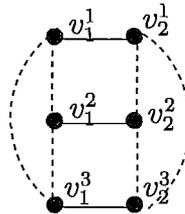


Figura 4.6: 3^*K_2 , onde as arestas ausentes são opcionais

Demonstração. (\Rightarrow) Sejam G um cografo-(1,2) e S^* um conjunto independente máximo de G .

Suponhamos, por absurdo, que G contém pelo menos um dos cografos da família 3^*K_2 como subgrafo.

Pelo teorema 4.8 temos que se G é um cografo-(1, 2) então $G[V \setminus S^*]$ é um cografo-(0, 2). Tome $S^* = \{v_1^1, v_1^2, v_1^3\}$. É fácil ver que $G[V \setminus S^*]$ não é (0, 2). Contradição.

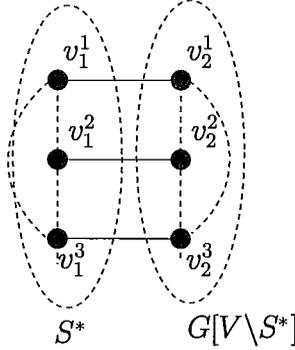


Figura 4.7: 3^*K_2 particionado em dois conjuntos independentes, S^* e $V \setminus S^*$

Logo, G não contém nenhum dos cografos da família 3^*K_2 como subgrafo.

(\Leftarrow) Seja G um cografo. Suponhamos que G não contém nenhum dos cografos da família 3^*K_2 como subgrafo.

Suponha, por absurdo, que G não é um grafo-(1, 2).

Seja S^* um conjunto independente máximo de G e S' um conjunto independente máximo de $G' = G[V \setminus S^*]$.

Temos que $G'' = G[S^* \cup S']$ é bipartido e S^* é um conjunto independente máximo de G'' . Daí, S' é uma cobertura mínima de $G[S^* \cup S']$, isto é, os vértices de S' cobrem todos os vértices de $G[S^* \cup S']$ (observe que todos os vértices de S' são adjacentes a algum vértice de S^* , caso contrário, S^* não seria um conjunto independente máximo; além disso, $|S^*| \geq |S'|$), logo pelo Teorema de König, que afirma que se G é bipartido então o tamanho do emparelhamento máximo é igual ao tamanho da cobertura mínima, temos que o emparelhamento máximo de $G[S^* \cup S']$ possui tamanho $|S'|$. Como G não contém nenhum dos cografos da família 3^*K_2 como subgrafo, temos que

Portanto, G não contém nenhum dos cografos da família 3^*K_3 como subgrafo.

(\Leftarrow) A prova será feita pela contrapositiva, i.e, dado um cografo G , se G não é um grafo-(2, 2) então G contém pelo menos um dos cografos da família 3^*K_3 como subgrafo.

Sejam G um cografo e S^* um conjunto independente máximo de G .

Suponhamos que G não é um grafo-(2, 2).

Temos, pela observação 4.2, que $G[V \setminus S^*]$ não é um grafo-(1, 2). Se $G[V \setminus S^*]$ não é um grafo-(1, 2) então, pelo teorema 4.9, $G[V \setminus S^*]$ contém pelo menos um dos cografos, K , da família 3^*K_2 como subgrafo, logo $\exists S_1 \cup S_2 \subseteq V \setminus S^*$ tal que S_1 e S_2 são conjuntos independentes e $|S_1| = |S_2| = 3$. Temos que $G[S_1 \cup S_2]$ admite um emparelhamento perfeito, formado pelo subgrafo K de G .

Como $G[S_1 \cup S_2]$ não é um grafo-(1, 2) então, pelo teorema 4.8, $G' = G[S^* \cup S_1 \cup S_2]$ não é um grafo-(2, 2).

Observe que G' não é bipartido, já que S^* , S_1 e S_2 são conjuntos independentes, $G[S_1 \cup S_2]$ é um grafo bipartido e para todo vértice v de S_1 e para todo vértice w de S_2 , v é adjacente a algum vértice de S^* e w também o é.

Como G' não é bipartido então pelo lema 4.3, G' contém um triângulo (a', b', c') com $a' \in S^*$, $b' \in S_1$ e $c' \in S_2$. Temos que os vértices a' , b' e c' formam uma clique máxima de G' , logo pelo teorema 4.8, $G'' = [(S^* \cup S_1 \cup S_2) \setminus \{a', b', c'\}]$ não é um grafo-(2, 1). Da mesma maneira, G'' também contém um triângulo (a'', b'', c'') , com $a'' \in S^*$, $b'' \in S_1$ e $c'' \in S_2$. Os vértices a'' , b'' e c'' formam uma clique máxima de G'' então pelo teorema 4.8, $G''' = G[(S^* \cup S_1 \cup S_2) \setminus \{a', b', c', a'', b'', c''\}]$ não é um grafo-(2, 0). Consequentemente, G''' também contém um triângulo (a''', b''', c''') , com $a''' \in S^*$, $b''' \in S_1$ e $c''' \in S_2$.

Daí, o subgrafo $G[\{a', b', c', a'', b'', c'', a''', b''', c'''\}]$ induz um cografo pertencente à família 3^*K_3 .

Logo, se G não é um grafo-(2, 2) então G contém pelo menos um dos cografos da família 3^*K_3 como subgrafo. \square

4.2.3 Caracterização dos cografos-(k, l)

Nesta subseção apresentamos uma caracterização para cografos-(k, l) em termos de subgrafos proibidos.

O teorema de caracterização de cografos-(k, l), apresentado a seguir, assim como o lema 4.5 apresentado anteriormente, correspondem às contribuições principais desta dissertação.

Teorema 4.11. *Um cografo G é um grafo-(k, l) se e somente se não contém nenhum dos cografos da família $(l + 1)^*K_{k+1}$ como subgrafo, como mostra a figura 4.9.*

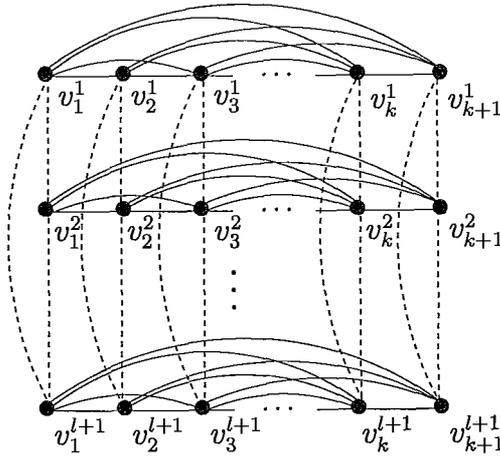


Figura 4.9: $(l + 1)^*K_{k+1}$, onde as arestas ausentes são opcionais

Demonstração. (\Rightarrow) Suponha que G contém pelo menos um dos cografos F da família $(l+1)^*K_{k+1}$, como subgrafo. Seja S_i^* um conjunto independente máximo de F , consistindo de $l+1$ vértices correspondentes de F , um de cada clique. É fácil ver que S_i^* é, de fato, um conjunto independente máximo de F . Pelo teorema 4.8, se G é (k, l) então $G[V \setminus S_i^*]$ é $(k-1, l)$. Repetindo este processo (k vezes) restarão $l+1$ vértices correspondentes de F . O subgrafo induzido por estes vértices induzem um conjunto independente com $l+1$ vértices, que não pode ser particionado em l cliques. Logo, como $F \subseteq G$ não é um grafo- (k, l) então G também não o é.

(\Leftarrow) A prova segue por indução em $k+l$.

Para $k+l=1$ o resultado segue imediatamente.

Assuma que o resultado é válido para o caso em que particionamos G em, no máximo, $k+l-1$ subconjuntos (i.e., com o número total de conjuntos independentes e cliques totalizando, no máximo, $k+l-1$).

Suponha que G não é um grafo- (k, l) . Seja S^* um conjunto independente máximo de G . Pela observação 4.2, $G' = G[V \setminus S^*]$ não é um grafo- $(k-1, l)$. Logo, pela hipótese indutiva, temos que G' contém pelo menos um cografo F da família $(l+1)^*K_k$ como subgrafo.

Sejam $S_1, S_2, \dots, S_{k-1}, S_k \subseteq V \setminus S^*$ conjuntos independentes, cada um deles formado pelos vértices correspondentes no subgrafo F , como mostra a figura 4.10.

Então, S^* é um conjunto independente máximo de $G[S^* \cup S_1 \cup \dots \cup S_k]$.

Pelo lema 4.5, temos que $G[S^* \cup S_1 \cup \dots \cup S_k]$ contém pelo menos um dos cografos da família $(l+1)^*K_{k+1}$ como subgrafo. \square

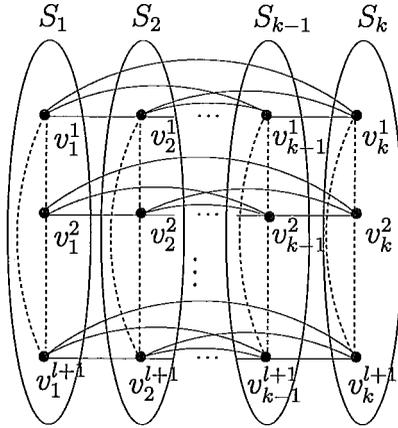


Figura 4.10: Uma partição em k conjuntos independentes

4.3 Reconhecimento dos cografos- (k, l)

Iniciamos esta seção observando que o teorema 4.8 nos dá um algoritmo simples para o reconhecimento dos cografos- (k, l) .

Considere k e l fixos. O algoritmo está descrito abaixo.

4.3.1 Algoritmo de reconhecimento dos cografos- (k, l)

Seja G um cografo. Denotamos por S_i o conjunto independente máximo de G_i , e C_j a clique máxima de G_j .

Entrada: O cografo $G = (V, E)$

4.3.2 Corretude

Teorema 4.12. *O algoritmo 4.1 está correto.*

Demonstração. Claramente, se o algoritmo encontra uma partição de G em k conjuntos independentes e l cliques então G é um grafo- (k, l) .

Algoritmo 4.1: Algoritmo para reconhecimento dos cografos- (k, l)

```
1  $G_1 := G$ ;  
2 para  $i = 1$  até  $k$  faça  
3   Encontre  $S_i$ ;  
4    $G_{i+1} := G_i \setminus S_i$ ;  
5    $|V'| := |V| - |S_i|$ ;  
6 para  $j = k + 1$  até  $k + l$  faça  
7   Encontre  $C_j$ ;  
8    $G_{j+1} := G_j \setminus C_j$ ;  
9    $|V'| := |V| - |C_j|$ ;  
10 se  $|V'| = 0$  então  
11    $G$  é um grafo- $(k, l)$ ;  
12 senão  $G$  não é um grafo- $(k, l)$ ;
```

Agora, temos que provar que se o algoritmo não encontra tal partição então G contém pelo menos um dos cografos da família $(l + 1)^*K_{k+1}$ como subgrafo.

A prova será dada por indução em $k + l$.

Para $k + l = 1$ o resultado segue imediatamente.

Seja o resultado verdadeiro quando particionamos um cografo G em no máximo $k + l$ subconjuntos, isto é, em no máximo k conjuntos independentes e l cliques.

Suponha que o algoritmo falha quando particionamos o cografo G em $k + l + 1$ subconjuntos: $k + 1$ conjuntos independentes e l cliques. Isto significa que existe pelo menos um vértice $v' \in V(G)$ que não pode ser inserido em nenhum dos $k + 1$ conjuntos independentes S_1, S_2, \dots, S_{k+1} , porque v' é adjacente a algum vértice de cada conjunto, isto é, $\exists s_i \in S_i$, com

$i = 1, \dots, k + 1$, tal que $(s_i, v') \in V(G)$, e nem inserido em nenhuma das l cliques $C_{k+2}, C_{k+3}, \dots, C_{k+1+l}$, porque v' é não-adjacente a algum vértice de cada conjunto, isto é, $\exists c_j \in C_j$, com $j = k + 2, \dots, k + 1 + l$, tal que $(c_j, v') \notin V(G)$.

Seja $H = G[V \setminus S^*]$, $H \subseteq G$. Observe que $S^* = S_1$

É fácil ver que ao executar o algoritmo em H , encontramos os mesmos k conjuntos independentes S_2, \dots, S_{k+1} , e as mesmas cliques $C_{k+2}, \dots, C_{k+1+l}$, como antes. Entretanto, ao particionar H em k conjuntos independentes e l cliques, restará um vértice $v' \in V[H]$, implicando em H conter pelo menos um dos cografos da família $(l+1)^*K_{k+1}$ como subgrafo, por hipótese indutiva. Como S_1 é um conjunto independente máximo de G então, aplicando o lema 4.5, concluímos que G contém pelo menos um dos cografos da família $(l+1)^*K_{k+2}$ como subgrafo. \square

4.3.3 Complexidade

Teorema 4.13. *O algoritmo 4.1 reconhece os cografos- (k, l) em tempo linear.*

Demonstração. Para provar a complexidade do algoritmo 4.1 vale simplesmente mencionar que o conjunto independente máximo e a clique máxima de um cografo podem ser encontrados em tempo linear [10]. Como estas são as principais operações executadas pelo algoritmo 4.1, concluímos que o algoritmo pode ser executado em tempo $O((k+l)(n+m))$, ou seja, o algoritmo 4.1 tem complexidade linear, já que k e l são fixos. \square

Capítulo 5

Conclusão

Esta dissertação apresentou um estudo detalhado sobre a classe dos cografos. Estudo este que suscitou na resolução de um determinado problema *NP*-completo de partição de grafos em geral, quando restritos à classe dos cografos.

Como contribuições deste trabalho, podemos destacar principalmente:

- Um teorema de caracterização, em termos de estruturas proibidas, de cografos- (k, l) , apresentado no capítulo 4 na seção 4.2 e que generaliza os resultados de [13] onde os cografos- $(1, 2)$ e os cografos- $(2, 2)$ são caracterizados;
- Um algoritmo linear e simples para o reconhecimento dos cografos- (k, l) ;
- Enunciamos e provamos o teorema 4.8 que pôde ser utilizado na demonstração detalhada da nova caracterização de cografos, provado por [13];

Vale mencionar que os resultados descritos acima foram apresentados em dois congressos, um no âmbito internacional [6] e outro no âmbito

nacional [7]. Além disso, tais resultados foram submetidos à revista Discrete Applied Mathematics e encontra-se em fase de análise.

- Demonstramos o teorema 2.12, que é uma caracterização dos cografos e o teorema 3.1, que é a corretude do algoritmo de reconhecimento dos cografos que não foram encontrados na literatura detalhadamente.

5.0.4 Trabalhos Futuros

Como continuação desta dissertação, propomos os seguintes problemas:

- Resolver o problema dos grafos- (k, l) para outras classes de grafos que contenham os cografos, como por exemplo, para grafos P_4 -reduzíveis (i.e., todo vértice de G pertence a no máximo um P_4) e para grafos P_4 -esparso (i.e., todo subgrafo de G com 5 vértices induz no máximo um P_4). Observe que *cografo* \subset P_4 -reduzíveis \subset P_4 -esparso.
- Resolver o problema da M -partição, ou seja, com restrições internas e externas, para cografos.

Seja M uma matriz simétrica de ordem $m \times m$, onde cada elemento $M_{i,j}$ é igual a 0, 1, ou *. Uma *partição* M de um grafo G , ou uma *M -partição*, é uma partição do conjunto de vértices $V(G)$ em m partes A_1, A_2, \dots, A_m , tal que A_i é estável (isto é, independente) se $M_{i,i} = 0$, completo (isto é, uma clique) se $M_{i,i} = 1$ ou sem nenhuma restrição interna se $M_{i,i} = *$, e tal que A_i e A_j são completamente não-adjacentes se $M_{i,j} = 0$, ou completamente adjacentes se $M_{i,j} = 1$, ou sem restrição externa se $M_{i,j} = *$.

- Resolver o problema de partição- (k, l) com listas para cografos.

O problema de partição com listas é um problema ainda mais geral que o problema da M -partição e, uma instância deste problema é um grafo G e uma coleção de listas $L(x)$, $x \in V(G)$, cada lista sendo um conjunto de partes (uma parte é um dos subconjuntos de vértices da partição). Uma solução para a M -partição com listas é uma solução para a M -partição correspondente, tal que cada vértice x pertence a uma das partes $A_i \in L(x)$.

Apêndice A

Apêndice do capítulo 2

A *decomposição modular* foi descoberta independentemente por Möhring [31] e Muller e Spinrad[32], e é um processo para decompor um grafo. Em qualquer estágio, o subgrafo que está sendo decomposto é chamado de *módulo*. No passo geral, cada um dos subgrafos é decomposto recursivamente, e este processo continua até que todos os subgrafos que estão sendo decompostos contenham somente um vértice.

A noção de módulo surgiu naturalmente de diferentes estruturas combinatórias [30]. A decomposição modular é muitas vezes o primeiro passo algorítmico para muitos problemas de grafos incluindo reconhecimento, decisão e problemas de otimização. De fato, a decomposição modular teve um papel muito importante para várias classes de grafos com algoritmos de reconhecimento (cografos [11], grafos de intervalo [31], grafos de permutação [33] e outras classes de grafos perfeitos [5, 17]), e no problema de orientação transitiva [15, 28]. No que segue, abordamos apenas algumas propriedades e definições a respeito de decomposição modular que serão utilizadas no decorrer desta dissertação.

A seguir, algumas propriedades e definições importantes de decomposição modular são apresentadas.

Definição A.1. *Um módulo de G é um conjunto de vértices M de $V(G)$ tal que todos os vértices de M tem a mesma vizinhança em $V(G)\setminus M$, i.e., cada vértice em $V(G)\setminus M$ ou é adjacente a todos os vértices de M ou a nenhum.*

Definição A.2. [5] *Seja $G = (V, E)$ um grafo.*

- $V' \subseteq V$ é dito um módulo trivial de G se $V' = V$ ou $|V'| = 1$, e V' é um módulo próprio se $V' \neq V$.
- G é um grafo primo (também denominado grafo primal) se G contém somente módulos triviais. Caso contrário, G é decomposto.
- Dois módulos V', V'' são sobrepostos se os conjuntos $V' \cap V''$, $V' \setminus V''$, e $V'' \setminus V'$ são todos não-vazios.
- Um módulo V' é forte se para todo módulo V'' , os módulos V' e V'' são não coincidentes, isto é, $V' \cap V'' = \emptyset$ ou $V' \subseteq V''$ ou $V'' \subseteq V'$.
- Dois vértices $x, y \in V$ são gêmeos se $\{x, y\}$ é um módulo de G , ou seja, x e y têm a mesma vizinhança em G . Os gêmeos x, y são gêmeos verdadeiros se $(x, y) \in E$, caso contrário x, y são gêmeos falsos.

Proposição A.3. [5] *Sejam $G = (V, E)$ um grafo e V', V'' dois módulos de G . Então as seguintes propriedades são válidas:*

- (i) $V' \cap V''$ é um módulo.
- (ii) Se $V'' \not\subseteq V'$, então $V' \setminus V''$ é um módulo.
- (iii) Se $V' \cap V'' \neq \emptyset$, então $V' \cup V''$ é um módulo.

(iv) Para todo $V''' \subseteq V$, o conjunto $V' \cap V'''$ é um módulo de $G[V''']$.

Corolário A.4. [5] Sejam U_1 , U_2 e V' subconjuntos de V . Se V' é um módulo dos subgrafos $G[U_1 \cup V']$ e $G[U_2 \cup V']$, então V' é um módulo do subgrafo $G[U_1 \cup U_2 \cup V']$.

Através da decomposição modular de um grafo G é possível construir uma árvore correspondente a G cuja estrutura é de extrema importância para algumas aplicações em grafos, e tal árvore denomina-se *árvore de decomposição modular*. Dentre estas aplicações, citamos a construção de algoritmos eficientes para reconhecer diversas classes de grafos [5, 11, 17, 31, 33].

A decomposição modular de um grafo G divide um grafo em módulos, e tal classificação é feita de acordo com a conectividade de G e \overline{G} .

Teorema A.5. (Gallai [15], Habib [18], Habib, Maurer [19], Sumner [35])
Seja $G = (V, E)$ um grafo com pelo menos dois vértices. Então exatamente uma das seguintes condições é verdadeira:

- *Módulo Paralelo:* Se G é desconexo e \overline{G} é conexo, então G pode ser decomposto em componentes conexos de G ;
- *Módulo Serial:* Se G é conexo e \overline{G} é desconexo, então G pode ser decomposto em componentes conexos de \overline{G} ;
- *Módulo Vizinhança:* Se G e \overline{G} são conexos então existe algum $V' \subseteq V$ e uma única partição P de V tal que:

(a) $|V'| > 3$,

(b) $G[V']$ é um subgrafo primal maximal de G ,

(c) Para toda classe S da partição P , S é um módulo e $|S \cap V'| = 1$.

De acordo com o teorema acima, a árvore de decomposição modular possui três tipos de módulos: módulo paralelo, módulo serial e módulo vizinhança. Cada vértice de G corresponderá a uma folha da árvore de decomposição modular. Cada módulo M de G estará associado a um nó da árvore, cujas folhas descendentes de M corresponderão aos vértices pertencentes ao módulo M . Mais precisamente, a árvore de decomposição modular é construída da seguinte maneira: consideremos o módulo $M = V(G)$. Se $|M| = 1$, então a decomposição modular de G será o único vértice de $V(G)$. Caso contrário, M é um módulo paralelo, serial ou vizinhança. Se M é um módulo paralelo, criamos um vértice P na árvore, cujos filhos de P correspondem à decomposição modular das componentes conexas de $G[M]$. Se M é um módulo serial, criamos um vértice S na árvore e inserimos como filhos deste vértice as decomposições modulares das componentes conexas de $\overline{G[M]}$. Se M é um módulo vizinhança, criamos um vértice N na árvore e inserimos como filhos deste vértice as decomposições modulares dos submódulos primais maximais de M . A árvore de decomposição modular de um grafo G é única, a menos de isomorfismos [26].

Como exemplo, a Figura A.1 exibe o grafo G e a sua árvore de decomposição modular.

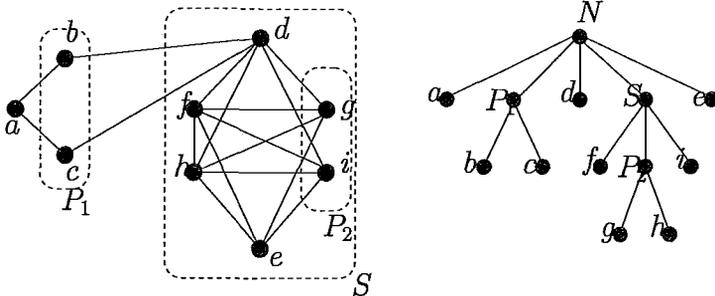


Figura A.1: Grafo G e sua respectiva árvore de decomposição modular

Bibliografia

- [1] M. F. Andrielli and C. P. de Mello. Conjunto fonte máximo em grafos de comparabilidade. Technical Report IC-96-07, icunicamp, 1996.
- [2] C. Berge. Les problèmes de coloration en théorie des graphes. *Inst. Stat. Univ. Paris*, pages 123–160, 1960.
- [3] A. Brandstädt. Partitions of graphs into one or two independent sets and cliques. *Discrete Mathematics*, 152(1-3):47–54, 1996.
- [4] A. Brandstädt. Corrigendum. *Discrete Mathematics*, 186:295, 1998.
- [5] A. Brandstädt, V. B. Le, and J. P. Spinrad. *Graph classes: A survey*. SIAM monographs on discrete mathematics and applications, Philadelphia, 4 edition, 1999.
- [6] R. S. F. Bravo, L. T. Nogueira, and S. Klein. Cographs (k,l) -partitionable. In *7th International Colloquium on Graph Theory*, volume 22, pages 277–280. *Electronic Notes in Discrete Mathematics*, 2005.
- [7] R. S. F. Bravo, L. T. Nogueira, and S. Klein. Obstruções de cografos (k,l) . In *Simpósio Brasileiro de Pesquisa Operacional*, pages 2361–2369. *Anais do XXXVII SBPO*, 2005.

- [8] A. Bretscher, D. G. Corneil, M. Habib, and C. Paul. A simple linear time lexbfs cograph recognition algorithm. In *29th WG Workshop*, pages 119–130, 2003.
- [9] M. Chudnovsky, N. Robertson, P. D. Seymour, and R. Thomas. Progress on perfect graphs. *Math. Programming*, B(97):405–422, 2003.
- [10] D. G. Corneil, H. Lerchs, and L. S. Burlingham. Complement reducible graphs. *Discrete Applied Mathematics*, 3:163–174, 1981.
- [11] D. G. Corneil, Y. Perl, and L. K. Stewart. A linear recognition algorithm for cographs. *SIAM Journal on Computing*, 14(4):926–934, 1985.
- [12] T. Ekim and D. de Werra. On split-coloring problems. *Journal of Combinatorial Optimization*, 10 (3):211–225, 2005.
- [13] T. Ekim, D. de Werra, and M. Demange. Partitioning cographs into cliques and stable sets. *Discrete Optimization*, 2:145–153, 2005.
- [14] T. Feder, P. Hell, S. Klein, and R. Motwani. Complexity of graph partition problems. In R. E. Miller F. W. Thatcher, editor, *Proceedings of the 31st Annual ACM Symposium on the Theory of Computing STOC'99*, pages 464–472. Plenum Press, 1999.
- [15] T. Gallai. Transitiv orientierbaren graphen. *Acta Math. Acad. Sci. Hung*, 18:25–66, 1967.
- [16] M. R. Garey and D. S. Johnson. *Computers and Intractability: A guide to the theory of NP-completeness*. New York, 1979.
- [17] M. C. Golumbic. *Algorithmic graph theory and perfect graphs*. Academic Press, New York, 1980.

- [18] M. Habib. Substitution of structures combinatories. Ph.d. thesis., Université Pierre et Marie Curie, Paris VI, France, 1981.
- [19] M. Habib and M. C. Maurer. On the x-join decomposition for undirected graphs. *Discrete Applied Mathematics*, 3:198–207, 1979.
- [20] M. Habib and C. Paul. A simple linear time algorithm for cograph recognition. *Discrete Applied Mathematics*, 145:183–197, 2005.
- [21] P. Hell, T. Feder, L. T. Nogueira, S. Klein, and F. Protti. List matrix partition of chordal graphs. *Theoretical Computer Science*, 349:52–66, 2005.
- [22] P. Hell, S. Klein, L. T. Nogueira, and F. Protti. Partitioning chordal graphs into independent sets and cliques. *Discrete Applied Mathematics*, 141:185–194, 2004.
- [23] P. Hell, S. Klein, L. T. Nogueira, and F. Protti. Packing r -cliques in weighted chordal graphs. *Annals of OR*, 138:179–187, 2005.
- [24] H. A. Jung. On a class of posets and the corresponding comparability graphs. *J. Combin. Theory Ser.*, 24 (B):125–133, 1978.
- [25] R. M. Karp. *Reducibility among combinatorial problems*. Complexity of computer computations, New York, 1972.
- [26] H. Lerchs. *On cliques and kernels*. Dept. of Comp. Sci., University of Toronto, 1971.
- [27] H. Lerchs. *On the clique-kernel structure of graphs*. Dept. of Comp. Sci., University of Toronto, 1972.

- [28] R. M. McConnell and J. P. Spinrad. Modular decomposition and transitive orientation. *Discrete Mathematics*, 201:189–241, 1999.
- [29] R. M. McConnell and J. P. Spinrad. *Linear-time modular decomposition and efficient transitive orientation of comparability graphs*.
- [30] R. H. Möhring and F. J. Radermacher. Substitution decomposition for discrete structures and connections with combinatorial optimization. *Annals of Discrete Mathematics*, 19:257–356, 1984.
- [31] R. H. Möhring and F. J. Radermacher. Algorithmic aspects of comparability graphs and interval graphs. In I. Rival, editor, *Graphs and Orders*, pages 41–102. D. Reidel Pub. Comp., 1985.
- [32] J. Muller and J. Spinrad. Incremental modular decomposition. *Journal ACM*, 36:1–19, 1989.
- [33] A. Pnueli, A. Lempel, and S. Even. Transitive orientation of graphs and identification of permutation graphs. *Canad. J. Math.*, 23:160–175, 1971.
- [34] L. Stewart. Cographs, a class of tree representable graphs. Master thesis, University of Toronto, Toronto, Ontario, 1978.
- [35] D. P. Sumner. Graphs indecomposable with respect to the x -join. *Discrete Mathematics*, 6:281–298, 1973.
- [36] D. P. Sumner. Dacey graphs. *J. Australian Math. Soc.*, 18(4):492–502, 1974.
- [37] J. L. Szwarcfiter. *Grafos e algoritmos computacionais*. Rio de Janeiro, 1983.