

UNIVERSIDADE FEDERAL FLUMINENSE

SIMONE COSTA DE SOUZA ESPOSITO

**Transversais-Clique de Ciclos em Grafos de Distância
Hereditária**

NITERÓI

2013

UNIVERSIDADE FEDERAL FLUMINENSE

SIMONE COSTA DE SOUZA ESPOSITO

Transversais-Clique de Ciclos em Grafos de Distância Hereditária

Dissertação de Mestrado submetida ao Programa de Pós-Graduação em Computação da Universidade Federal Fluminense como requisito parcial para a obtenção do título de Mestre. Área de concentração: Redes de Computadores.

Orientador:
Loana Tito Nogueira

Co-orientador:
Fábio Protti

NITERÓI

2013

Transversais-Clique de Ciclos em Grafos de Distância Hereditária

Simone Costa de Souza Esposito

Dissertação de Mestrado submetida ao Programa de Pós-Graduação em Computação da Universidade Federal Fluminense como requisito parcial para a obtenção do título de Mestre.

Aprovada por:

Profª. Dra. Loana Tito Nogueira / IC-UFF (Presidente)

Prof. Dr. Fábio Protti / UFF

Prof. Dr. Luidi Simonetti / UFF

Prof. Dra. Celina Miraglia Herrera de Figueiredo / UFRJ

Niterói, Julho de 2013.

Agradecimentos

Primeiramente, agradeço a Deus, pela realização deste sonho.

Ao meu esposo, Antonio Marco, por todo apoio e compreensão, principalmente na minha gravidez. Não posso esquecer de todas as vezes que me acompanhou nas idas e vindas de Niterói. Obrigada, amor, por cuidar de mim de uma forma tão especial!

Ao meu querido filho, Daniel, que esteve comigo nas últimas provas do mestrado e me deu a coragem necessária para avançar. Agora, já não mais em meu ventre, seu sorriso continua me dando força para vencer os obstáculos e não desanimar.

Aos meus pais, Antonio e Rosângela, pelos valores, educação e incentivo dados. Em especial, a minha mãe, que nos últimos meses, não mediu esforços em me ajudar para que este trabalho fosse concluído. Agradeço, também, ao meu irmão, Robson que sempre me incentivou.

A minha orientadora, Loana Nogueira, pela dedicação e paciência demonstrada em todo momento. Principalmente, por acreditar em mim. Ela realmente será um exemplo a seguir. Com certeza, sua orientação foi o melhor presente que recebi neste mestrado. Da mesma maneira, ao meu co-orientador, professor Fábio Protti, pela grande colaboração e disponibilidade para a conclusão deste trabalho.

Aos amigos que, direta ou indiretamente, contribuíram para que este trabalho fosse concluído.

Resumo

Um grafo G é de distância hereditária se G é conexo e todo caminho induzido em G é isométrico, isto é, as distâncias em qualquer subgrafo induzido conexo são as mesmas que as no grafo original. Os grafos de distância hereditária foram introduzidos e primeiramente caracterizados por Howorka e formam uma importante subclasse dos grafos perfeitos.

Neste trabalho, consideramos o problema de verificar se um dado grafo de distância hereditária G admite uma clique que intercepte todos os ciclos do grafo, ou seja, estudamos transversais-cliques de ciclos em grafos de distância hereditária.

Mais formalmente, um *transversal-clique de ciclos*, ou simplesmente *cct* (do inglês *clique-cycle transversal*), de um grafo G , é uma clique $T \subseteq V(G)$ tal que $T \cap V(C) \neq \emptyset$ para todo ciclo C de G . Logo, um grafo admite cct se e somente se puder ser particionado em uma floresta e um subgrafo completo. Por esta razão, tais grafos são também chamados de *grafos*-(\mathcal{F}, \mathcal{C}).

Grafos que admitem cct podem ser reconhecidos em tempo polinomial. Entretanto, nenhuma caracterização de tais grafos é conhecida, mesmo para grafos perfeitos. Por esta razão, como contribuição principal desta dissertação, descrevemos uma caracterização de grafos de distância hereditária que admitem cct através de uma família de subgrafos induzidos proibidos. Este resultado amplia estudos similares para grafos cordais e cografos. Além disso, mostramos que grafos de distância hereditária que admitem cct podem ser reconhecidos em tempo linear por esta caracterização.

Palavras-chave: Grafos de distância hereditária, transversal de ciclos, transversal-clique de ciclos.

Abstract

A graph G is distance-hereditary if it is connected and every induced path is isometric, that is, the distance between any two vertices in any induced connected subgraph equals their distance in the graph. Distance-hereditary graphs were introduced and first characterized by Howorka and form an important subclass of the perfect graphs.

In this work, we consider the problem of verifying if a distance-hereditary graph G admits a clique that intersects all of its cycles, or equivalently, we study clique cycle-transversal in distance-hereditary graphs.

More formally, a *clique cycle-transversal*, or simply *cct*, of a graph G , is a clique $T \subseteq V(G)$ such that $T \cap V(C) \neq \emptyset$ for every cycle C of G . Thus, a graph admits a cct if and only if it can be partitioned into a forest and a complete subgraph. By this reason, such graphs are also called $(\mathcal{F}, \mathcal{C})$ -graphs.

Graphs admitting a cct can be recognized in polynomial time. However, no structural characterization for distance-hereditary graphs admitting a cct was known, even for perfect graphs. For this reason, as the main contribution of this work, we describe a characterization of distance-hereditary graphs with cct in terms of forbidden induced subgraphs. This result extends similar studies for chordal graphs and cographs. Furthermore, we show that distance-hereditary graphs admitting a cct can be recognized in linear time as a consequence of our characterization.

Keywords: Distance-hereditary graphs, clique transversal, clique cycle-transversal.

Palavras-chave

1. Grafos de distância hereditária
2. Grafos- $(2, 1)$
3. Transversal de ciclos
4. Transversal-clique de ciclos
5. Grafos- $(\mathcal{F}, \mathcal{C})$.

Sumário

Lista de Figuras	viii
1 Introdução	1
1.1 Grafos - Definições e Notações	3
1.1.1 Lógica de segunda ordem monádica	8
1.2 Complexidade de Algoritmos	9
1.3 Organização do Trabalho	10
2 Grafos de Distância Hereditária	11
2.1 Introdução	11
2.2 Definições e Caracterização	11
2.2.1 Clique-width	14
2.3 Relação dos grafos de distância hereditária com outras famílias de grafos .	18
2.3.1 Cografos	19
2.3.1.1 Propriedades Estruturais	19
2.3.1.2 Decomposição Modular	20
2.3.1.3 Co-árvore	21
2.4 Reconhecimento de grafos de distância hereditária	22
3 Transversais de Ciclos	27
3.1 Introdução	27
3.2 Transversal de Ciclos	27
3.2.1 Transversais de ciclos ímpares	28
3.3 Grafos- (k, l)	29

3.3.1	Grafos- $(2, 1)$	30
3.4	Grafos- $(\mathcal{F}, \mathcal{C})$	31
3.4.1	Reconhecimento de grafos- $(\mathcal{F}, \mathcal{C})$	31
4	Transversais-clique de Ciclos em Grafos de Distância Hereditária	33
4.1	Introdução	33
4.2	Caracterização dos grafos de distância hereditária que admitem cct	36
4.2.1	Caso 1: y é um gêmeo verdadeiro de x	37
4.2.2	Caso 2: y é um gêmeo falso de x	42
5	Conclusão	55
5.1	Trabalhos Futuros	56
	Referências	57

Lista de Figuras

1.1	Principais classes e contribuições da dissertação	3
2.1	Exemplos de grafos de distância hereditária.	11
2.2	Casa, buraco, dominó, e leque.	12
2.3	Exemplo de sequência de poda.	14
2.4	Exemplo de árvore de poda.	14
2.5	Exemplo de um cografo e sua co-árvore.	21
2.6	Layout de distância.	24
4.1	Casos da Proposição 4.1	35
4.2	Subgrafos proibidos para grafos de distância hereditária com cct.	36
4.3	Conjuntos Q_2 , Q_1 e $N_1(x)$	37
4.4	Caso 1 - Possíveis ciclos com y em G'	37
4.5	Vizinhança de Q_2 em $N_1(x)$ contendo dois vértices a e b	39
4.6	Vizinhança de Q_2 em $N_1(x)$ contendo um vértice a , existência de C_4 <i>aucv</i> e $bv \in E$	39
4.7	Ciclos em $G[N^2(x) \cup R]$	40
4.8	Vértices $a, b \in N(x)$, $ab \notin E(G)$ e ciclos do tipo (A_1) e (A_2)	41
4.9	Vértices $a, b \in N(x)$, $ab \notin E(G)$ e ciclo do tipo (B_3)	41
4.10	Vértices $a, b \in N(x)$, $ab \notin E(G)$, ciclo do tipo (B_3) e a e b adjacentes a u e v	42
4.11	Caso 2 - Possíveis ciclos com y em G'	43
4.12	Vizinhança de Q_2 em $N_1(x)$ contendo três vértices a, b, c	44
4.13	Vizinhança de Q_2 em $N_1(x)$ contendo dois vértices a, b e outro vértice $c \in$ $N_1(x)$	44

4.14	Vizinhança de Q_2 em $N_1(x)$ contendo dois vértices a, b com a não adjacente a a' e b a b' em Q_1	45
4.15	Vizinhança de Q_2 em $N_1(x)$ contendo um vértice a	45
4.16	Vizinhança de Q_2 em $N_1(x)$ contendo um vértice a e $bc \in E(G)$	46
4.17	Vizinhança de Q_2 em $N_1(x)$ contendo um vértice a e triângulo auv com $v \in N^2(x)$	46
4.18	Vizinhança de Q_2 em $N_1(x)$ contendo um vértice a e C_4 $auvw$ com $u, v \in N^2(x)$ e $w \in R$	46
4.19	C_4 abc com $a, b \in N(x)$ e $c \in N^2(x)$ e um vértice não adjacente d em Q_1	47
4.20	C_4 abc com $a, b \in N(x)$ e dois vértices não adjacentes d, d' em Q_1	47
4.21	Vértice e em $N^2(x)$ adjacente aos vértices a e b em $N(x)$	48
4.22	Vértice d em um C_3 duv com $u, v \in N^2(x)$	49
4.23	Vértice d em um C_4 $duvw$ com $u, v \in N^2(x)$ e $w \in R$	49
4.24	Vértice d em um C_4 com $d, z \in N(x)$ e $u, v \in N^2(x)$	50
4.25	Vértice u adjacente a três vértices a_1, a_2, a_3 em S	50
4.26	Vértice $u \in N^2(x)$ e ciclo do tipo (A_1) ou (A_2)	50
4.27	Vértice $u \in N^2(x)$ e ciclo do tipo (B_1)	51
4.28	Vértice $u \in N^2(x)$ e ciclo do tipo (B_2) ou (D_1)	51
4.29	Vértice $u \in N^2(x)$ e ciclo do tipo (B_3)	51
4.30	Vértice $u \in N^2(x)$ e ciclo do tipo (C_1) ou (D_2)	51
4.31	Vértice $a' \in Q_1$	52
4.32	Principais classes e contribuições da dissertação	54

Capítulo 1

Introdução

Atualmente, a teoria dos grafos é uma importante ferramenta matemática com aplicação em diversas áreas de conhecimento, tais como: Computação, Genética, Química, Engenharia, etc. Podemos verificar esta importância através do aumento de estudos, publicações e artigos sobre o assunto.

O objetivo da teoria dos grafos é modelar problemas de natureza combinatória de forma a resolvê-los computacionalmente. No entanto, esses problemas são geralmente NP-difíceis e, portanto, não existem algoritmos eficientes para resolvê-los, a menos que $P = NP$. Uma maneira de solucionar os problemas difíceis em grafos é restringi-los a classes especiais de grafos.

Como exemplo, podemos citar o problema de encontrar um transversal mínimo de ciclos em um grafo. Um *transversal de ciclos* de um grafo G é um subconjunto $T \subseteq V(G)$ tal que $T \cap V(C) \neq \emptyset$ para todo ciclo C de G . Sendo assim, temos que $V(G) \setminus T$ é uma floresta (ou seja, não contém ciclo). Desta maneira, encontrar um transversal mínimo de ciclos é encontrar o menor número de vértices que ao serem removidos torna o grafo acíclico.

Encontrar um transversal mínimo de ciclos em um grafo é NP-difícil devido ao resultado geral em [43], que diz que o problema de encontrar o número mínimo de vértices de um grafo G cuja deleção resulta em um subgrafo satisfazendo uma propriedade hereditária π em subgrafos induzidos é NP-difícil. No entanto, este pode ser solucionado em tempo linear quando restrito a classe dos cografos [5].

Muitos problemas em grafos podem ser considerados no contexto de transversais mínimos de ciclos, como: encontrar um transversal mínimo de ciclos ímpares (que é equivalente a encontrar um subgrafo bipartido induzido máximo), ou o problema de encontrar um

transversal mínimo de triângulo (que é equivalente a encontrar um subgrafo induzido máximo livre de triângulo).

Nesta dissertação, não estamos interessados em transversais mínimos de ciclos, tratamos, especificamente, de transversais de ciclos que têm a propriedade adicional de ser cliques, chamados *transversais-clique de ciclos*, ou simplesmente *cct* (do inglês *clique-cycle transversal*). Um grafo G admite *cct* se e somente se puder ser particionado em uma floresta e um subgrafo completo. Por esta razão, tais grafos são também chamados de *grafos*-(\mathcal{F}, \mathcal{C}).

Os *grafos*-(\mathcal{F}, \mathcal{C}) formam uma subclasse dos grafos-(2, 1). Os grafos-(2, 1) são um caso particular dos grafos-(k, l) (grafos cujo conjunto de vértices pode ser particionado em k conjuntos independentes e l cliques), quando $k = 2$ e $l = 1$.

Em [3, 4, 7], Brandst  d apresentou um algoritmo polinomial de reconhecimento para os grafos-(2, 1) e mostrou que reconhecer grafos-(k, l)    um problema NP-Completo para $k \geq 3$ ou $l \geq 3$. Por este motivo, muitos trabalhos t  m considerado fam  lias especiais de grafos-(k, l) que podem ser reconhecidos eficientemente; por exemplo, grafos cordais-(k, l) [31], cografos-(k, l) [19, 20, 22] e P_4 -esparsos-(k, l) [8].

Grafos-(\mathcal{F}, \mathcal{C}) podem ser reconhecidos em tempo polinomial utilizando o algoritmo de reconhecimento dos grafos-(2, 1). No entanto, n  o se conhece na literatura nenhuma caracteriza  o por subgrafos proibidos para grafos gerais tanto para a classe dos grafos-(\mathcal{F}, \mathcal{C}) quanto para a classe dos grafos-(2, 1).

Neste trabalho, consideramos o problema de reconhecer se um grafo G    (\mathcal{F}, \mathcal{C}), ou seja, verificar se G admite um transversal-clique de ciclos, numa importante subclasse dos grafos perfeitos, os grafos de dist  ncia heredit  ria. Um grafo G    de dist  ncia heredit  ria se G    conexo e as dist  ncias em qualquer subgrafo induzido conexo s  o as mesmas que as no grafo original.

Mais especificamente, como contribui  o principal desta disserta  o, fornecemos uma caracteriza  o de grafos de dist  ncia heredit  ria que admitem um transversal-clique de ciclos atrav  s de um conjunto finito de subgrafos induzidos proibidos, resultado este publicado em 2013 [6]. Um estudo similar sobre transversais-clique de ciclos em cografos, uma subclasse dos grafos de dist  ncia heredit  ria, foi realizado em [5].

Al  m disso, como decorr  ncia do teorema principal deste trabalho, caracterizamos os grafos de dist  ncia heredit  ria que s  o grafos-(2, 1) atrav  s de subgrafos induzidos proibidos e apresentamos uma outra prova para o resultado apresentado em [5], ou seja, caracterizamos, t  mb  m atrav  s de subgrafos proibidos, os cografos que admitem um

transversal-clique de ciclos.

Por último, mostramos que grafos de distância hereditária que admitem cct podem ser reconhecidos em tempo linear através da caracterização fornecida neste trabalho.

Para ilustrar, segue na Figura 1.1, um diagrama com as principais classes de grafos estudadas neste trabalho. Observe que as setas mostram o relacionamento entre estas classes: $(\mathcal{F}, \mathcal{C}) \subseteq (2, 1)$ e cografos \subseteq grafos de distância hereditária. As principais contribuições desta dissertação estão indicadas pela linha contínua.

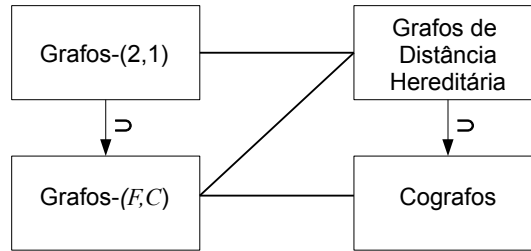


Figura 1.1: Principais classes e contribuições da dissertação

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

1.1 Grafos - Definições e Notações

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 $E(G)$. Se $e \in E(G)$ então e será denotada por $e = (u, v)$ ou $e = uv$. Utilizamos 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 .

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 $N(u) \cup \{u\}$ é chamado de *vizinhança fechada* de u .

O *grau* de um vértice $v \in V(G)$, denotado por $d(v)$, é o número de arestas incidentes ao vértice v . Um vértice u é dito universal quando $N(u) = V(G) - \{u\}$, ou seja, é um vértice adjacente a todos os outros vértices do grafo. Um vértice pendente é um vértice de grau 1.

Dois vértices x e y são gêmeos verdadeiros se e somente se $N[x] = N[y]$ e gêmeos falsos se e somente se $N(x) = N(y)$.

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), e $(v_i, v_{i+1}) \in E(G)$, $1 \leq i \leq k-1$.

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

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

Uma *corda* em um caminho P é uma aresta que liga dois vértices não-consecutivos de P . Um *caminho induzido* é um caminho sem cordas. Denotamos por P_k o caminho induzido por k vértices. Dizemos que um grafo é P_k -free quando não contém um P_k como subgrafo induzido.

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 $v_1 = v_{k+1}$. O tamanho de um ciclo é o número de arestas no ciclo.

O *ciclo sem corda* com n vertices é denotado por C_n . O grafo C_n para $n \geq 5$ é um *buraco*.

Um grafo G é dito *cíclico* quando G contém um ciclo como subgrafo. 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 .

Um grafo G é *completo* se quaisquer dois vértices distintos de G são adjacentes. Denotamos por K_n o grafo completo com n vértices. Os grafos K_1 e K_3 são chamados *grafo trivial* e *triângulo*, respectivamente.

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 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 consideraremos apenas subgrafos induzidos e portanto, quando nos referirmos a subgrafo estaremos assumindo implicitamente que são subgrafos induzidos.

Um conjunto de vértices I de um grafo G é um *conjunto independente* se $G[I]$ é um grafo sem arestas. Denotamos por I_n o conjunto independente com n vértices. Um conjunto independente I é dito *maximal* se para qualquer conjunto independente I' tal que $I \subseteq I'$ então $I = I'$. Definimos por $\alpha(G)$ o tamanho do *conjunto independente máximo*, isto é:

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

Um conjunto de vértices C de um grafo G é uma *clique* se $G[C]$ é um grafo completo. Denotamos por K_k uma clique de k vértices. Uma clique C é dita *maximal* se para qualquer clique C' tal que $C \subseteq C'$ então $C = C'$. 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 *núcleo* de um grafo é um conjunto independente maximal. Observe que $S \subseteq V$ é um núcleo se e somente se S é uma clique maximal em \overline{G} .

O *complemento* de um grafo G , denotado por \overline{G} , é o grafo que possui o mesmo conjunto de vértices de G e tal que dois vértices são adjacentes em \overline{G} se e somente se não são adjacentes em 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*. Uma *componente conexa* de G é um subgrafo maximal conexo de G .

Uma *floresta* é um grafo acíclico. Uma *árvore* T é uma floresta conexa. 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. Uma árvore não enraizada é também denominada *árvore livre*.

Sejam v, w dois vértices de uma árvore enraizada T de raiz r . Suponhamos que v pertença ao caminho de r a w em T , então denominamos v *ancestral* de w e w *descendente* de v .

Temos também que 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 $\text{filho}(v)$.

Dois vértices que possuem o mesmo pai são chamados de *irmãos*. A raiz de uma árvore não possui pai, enquanto que todo vértice $v \neq r$ possui um único. Uma *folha* é um vértice que não possui filhos e um vértice que possui pelo menos um, é chamado de vértice *interno*. Os vértices de uma árvore também são chamados de *nós*.

Se H é um subgrafo induzido de G então nós dizemos que G *contém* H , caso contrário G é livre de H .

Seja $X = (V_X, E_X)$ e $Y = (V_Y, E_Y)$ dois grafos tais que $V_X \cap V_Y = \emptyset$. As operações "+" e " \cup " são definidas como segue: a *união disjunta* $X \cup Y$, algumas vezes tratada simplesmente como *união*, é o grafo com o conjunto de vértices $V_X \cup V_Y$ e o conjunto de arestas $E_X \cup E_Y$; a operação *junção* $X + Y$ é o grafo com o conjunto de vértices $V_X \cup V_Y$ e o conjunto de arestas $E_X \cup E_Y \cup \{xy \mid x \in V_X, y \in V_Y\}$. Dois grafos são *disjuntos* se eles não têm vértice em comum.

Dois conjuntos X e Y se *interceptam* ou (se *sobrepõem*) se $X \cap Y \neq \emptyset$, $Y \setminus X \neq \emptyset$, e $X \setminus Y \neq \emptyset$.

Dois conjuntos X e Y são *comparáveis* se X é um subconjunto de Y ou Y é um subconjunto de X , caso contrário X e Y são *incomparáveis*.

Dois grafos X e Y são ditos *isomorfos*, denotado por $X \approx Y$, quando existe uma função bijetora entre os conjuntos de vértices de X e Y $f : V(X) \rightarrow V(Y)$ de tal forma que quaisquer dois vértices u e v de X são adjacentes em X se e somente se $f(u)$ e $f(v)$ são adjacentes em Y .

Uma *casa* é o grafo com os vértices a, b, c, d, e e as arestas ab, bc, cd, ad, ae, be .

Um *leque* ou *gema* é o grafo com os vértices a, b, c, d, e e as arestas $ab, bc, cd, ae, be, ce, de$.

Um *dominó* é o grafo com os vértices a, b, c, d, e, h e as arestas $ab, bc, cd, ad, be, eh, ch$.

Um *grafo split* é um grafo cujo conjunto de vértices pode ser particionado em um conjunto independente e uma clique. É bem conhecido que G é um grafo split se e somente se G é livre de $(2K_2, C_4, C_5)$ [24].

Um grafo G é *perfeito* quando seu número cromático de todo subgrafo induzido é igual ao tamanho da clique máxima, isto é, $\chi(G) = \omega(G)$. Temos uma caracterização de extrema importância para os grafos perfeitos, na qual diz que G é perfeito se e somente se G é um grafo *Berge*, isto é, G não admite ciclo ímpar induzido maior ou igual a 5 e o seu

complemento, \overline{G} , também não admite ciclo ímpar induzido maior ou igual a 5.

Um grafo 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. Tal partição (V_1, V_2) é chamada *bipartição* do grafo. 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 $|V_2| = m$. É fato conhecido que um grafo é bipartido se e somente se não possui um ciclo ímpar como subgrafo.

Um *cografo* é um grafo que não contém P_4 , isto é, não contém caminho induzido com 4 vértices como subgrafo. No capítulo 2, daremos outras definições e caracterizações desta classe de grafos.

Um *grafo cordal* é um grafo que não contém C_k , para $k \geq 4$.

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

Um *grafo de distância hereditária* é um grafo em que as distâncias em qualquer subgrafo conexo induzido são as mesmas no grafo original. Apresentaremos detalhadamente tais grafos no capítulo 2 desta dissertação.

Um *grafo threshold* é um grafo que pode ser construído a partir de um único vértice por repetidas aplicações das seguintes operações: (a) adicionar um único vértice isolado ao grafo; (b) adicionar um único vértice universal ao grafo. É bem conhecido que G é um grafo threshold se e somente se G é livre de $(2K_2, C_4, P_4)$ [10].

Seja \mathcal{H} uma determinada família de grafos. Um \mathcal{H} -subgrafo de G é um subgrafo induzido de G isomorfo a um membro de \mathcal{H} . Um grafo é \mathcal{H} -livre se não contém um \mathcal{H} -subgrafo. Um \mathcal{H} -transversal de um grafo G é um subconjunto $T \subseteq V(G)$ tal que T intercepta todo \mathcal{H} -subgrafo de G . Claramente, se T é um \mathcal{H} -transversal de G então $G - T$ é \mathcal{H} -livre. Além disso, se T é mínimo então $G - T$ é um subgrafo induzido \mathcal{H} -livre de G máximo.

Se $T \cap V(C) \neq \emptyset$ para o ciclo C , dizemos que T *cobre* C ou C é coberto por T . Dizemos ainda que, um vértice $x \in V(G)$ *vê* um ciclo C quando x é adjacente a algum vértice $y \in V(C)$. Da mesma forma, se x vê uma aresta uv , então ou x é vizinho de u ou vizinho de v ou vizinho de ambos.

1.1.1 Lógica de segunda ordem monádica

Nesta seção, vamos introduzir alguns conceitos de lógica de segunda ordem monádica. Com este objetivo, apresentamos, inicialmente, as definições de lógica de primeira ordem e lógica de segunda ordem.

A lógica de primeira ordem, conhecida também como cálculo de predicados de primeira ordem, é um sistema lógico que estende a lógica proposicional. A lógica de primeira ordem em relação à lógica proposicional acrescenta a quantificação, ou seja, a linguagem da lógica de primeira ordem vai permitir quantificar os elementos do domínio.

A lógica de segunda ordem é uma extensão da lógica de primeira ordem uma vez que podemos quantificar não só os elementos do domínio mas também os predicados (propriedades sobre os elementos). Já a lógica de segunda ordem monádica é uma restrição da lógica de segunda ordem, onde é possível quantificar apenas as variáveis de primeira ordem e os predicados unários, ou seja, predicados que admitem apenas um argumento.

Fórmulas da lógica de segunda ordem monádica, ou simplesmente fórmulas MSOL (do inglês *monadic second order logic*), são construídas a partir de fórmulas atômicas utilizando conectivos booleanos ($\wedge, \vee, \neg, \rightarrow, \leftrightarrow$), quantificadores sobre variáveis individuais ($\exists x, \forall x$) e quantificadores sobre variáveis de conjunto ($\exists X, \forall X$).

A Lógica de segunda ordem monádica pode ser usada para descrever propriedades de grafos, como: relações entre grafos, relações entre vértices de grafos, etc. O domínio (ou universo) de discurso, conjunto sobre os quais se pode quantificar, contém vértices e arestas. A incidência entre vértices e arestas é representada por uma relação binária.

Nas fórmulas da lógica de segunda ordem monádicas apresentadas nesta dissertação, assumimos um número infinito de variáveis individuais $\{x, y, z, w, \dots\}$ e de variáveis de conjuntos $\{T, X, Y, \dots\}$. As fórmulas atômicas utilizadas são: $T(x)$, que se verdadeira significa que o vértice x é um elemento do conjunto T e $A(x, y)$, que se verdadeira significa que o vértice x é adjacente ao vértice y .

Por exemplo, verificar que um grafo G é bipartido pode ser representado pela seguinte fórmula de segunda ordem monádica:

$$\exists X(\forall u, v(u \in X \wedge A(u, v) \rightarrow v \notin X) \wedge \forall u, v(u \notin X \wedge A(u, v) \rightarrow v \in X)).$$

1.2 Complexidade de Algoritmos

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 .

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:

1. f pode ser calculada em tempo polinomial;
2. 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:

1. $\pi \in NP$;
2. para todo problema $\pi' \in NP$ 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 $\pi \in NP$ e que existe uma transformação de um problema NP-completo π' em π .

Como fonte de referência para esta seção, indicamos [26, 42].

1.3 Organização do Trabalho

Este trabalho está dividido da seguinte forma.

No Capítulo 1, além da introdução, apresentamos as definições e notações utilizadas no decorrer desta dissertação.

No Capítulo 2, introduzimos as principais definições e caracterizações dos grafos de distância hereditária. Abordamos o relacionamento desta classe com outras classes de grafos e apresentamos uma importante subclasse, os cografos. Ainda neste capítulo, descrevemos um algoritmo de reconhecimento para a classe e estudamos sua complexidade.

No Capítulo 3, introduzimos o conceito de transversais de ciclos e definimos os grafos- (k, l) e uma de suas subclasses, os grafos- $(2, 1)$, além de estudar os grafos- $(\mathcal{F}, \mathcal{C})$ e o conceito de transversal-clique de ciclos.

No Capítulo 4, apresentamos o principal resultado desta dissertação, a caracterização dos grafos de distância hereditária que admitem transversal-clique de ciclos através de subgrafos proibidos.

Finalmente, no Capítulo 5, concluímos esta dissertação, apresentando os resultados obtidos e os problemas que pretendemos abordar futuramente.

Capítulo 2

Grafos de Distância Hereditária

2.1 Introdução

Neste capítulo, temos como objetivo apresentar as principais definições e caracterizações dos grafos de distância hereditária. Da mesma maneira, abordamos os conceitos de uma importante subclasse, os cografos. Além disso, descrevemos um algoritmo de reconhecimento para a classe e estudamos sua complexidade.

A seguir, vamos introduzir alguns conceitos e caracterizações dos grafos de distância hereditária.

2.2 Definições e Caracterização

Os grafos de distância hereditária foram introduzidos por Howorka (1977) que apresentou as primeiras definições e caracterizações para esta classe de grafos [32].

Definição 2.1 *Um grafo G é de distância hereditária se para qualquer subgrafo conexo induzido H de G , a distância entre quaisquer dois vértices x e y em H é a mesma que em G , ou seja $d_H(x, y) = d_G(x, y)$, isto é, todo caminho induzido em G é isométrico.*

Na Figura 2.1, apresentamos alguns exemplos de grafos de distância hereditária.

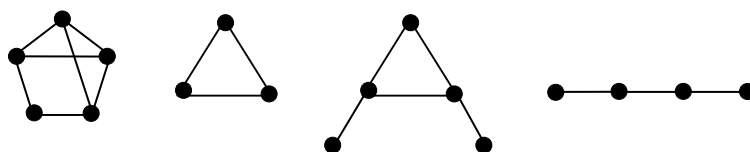


Figura 2.1: Exemplos de grafos de distância hereditária.

No que segue, introduzimos algumas caracterizações bem conhecidas de grafos de distância hereditária, dadas por Bandelt e Mulder [2] e Hammer e Maffray [30].

Bandelt e Mulder mostraram em [2] uma caracterização por subgrafos proibidos para esta classe de grafos.

Teorema 2.1 [2] *Um grafo G é de distância hereditária se e somente se é livre de (casa, buraco, dominó e leque).*

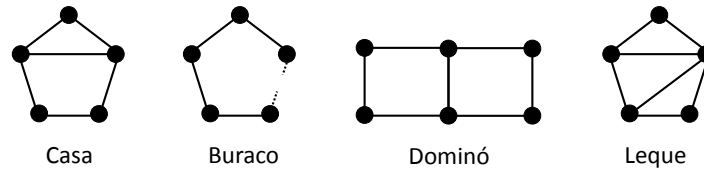


Figura 2.2: Casa, buraco, dominó, e leque.

Além disso, também mostraram como um grafo de distância hereditária G pode ser construído recursivamente. Veja Teorema abaixo.

Teorema 2.2 [2] *Um grafo G é de distância hereditária se e somente se G pode ser gerado a partir de um único vértice por uma sequência de extensões consistindo de operações para adicionar um vértice pendente, um gêmeo falso ou um gêmeo verdadeiro.*

O Teorema 2.2 nos garante que dado um grafo $G = (V, E)$ de distância hereditária e $x \in V(G)$, um grafo de distância hereditária G' pode ser construído a partir de G adicionando-se o vértice y adjacente a:

- (1) um único vértice x (y é um vértice pendente);
- (2) $N(x)$ (y é um gêmeo falso);
- (3) $N[x]$ (y é um gêmeo verdadeiro).

Estas operações são chamadas, geralmente, de *extensões de um vértice*.

Assim como podemos construir recursivamente um grafo de distância hereditária através de *extensões de um vértice*, este também pode ser desconstruído "invertendo-se" este procedimento.

Hammer e Maffray [30] propuseram um algoritmo de reconhecimento linear para esta classe de grafos através da sequência de decomposição de grafos de distância hereditária, a qual chamaram de *sequência de poda*. Além disso, mostraram que um grafo conexo G possui uma sequência de poda se e somente se G é um grafo de distância hereditária.

Definição 2.2 *Uma sequência de poda é uma ordenação total $\sigma = [x_1, \dots, x_n]$ de vértices e uma sequência $S = [s_1, \dots, s_{n-1}]$ de palavras, tal que para $1 \leq i \leq n-1$, s_i é uma das seguintes palavras:*

- (x_i, P, x_j) , $i < j$, se em G_i , $N(x_i) = \{x_j\}$ (x_i é um vértice pendente);
- (x_i, F, x_j) , $i < j$, se em G_i , $N(x_i) = N(x_j)$ (x_i e x_j são gêmeos falsos);
- (x_i, T, x_j) , $i < j$, se em G_i , $N[x_i] = N[x_j]$ (x_i e x_j são gêmeos verdadeiros);

onde G_i denota o subgrafo induzido $G[V \setminus \{x_1, \dots, x_{i-1}\}]$.

Podemos entender mais claramente o processo de poda do grafo através do Teorema 2.3, também demonstrado em [30].

Teorema 2.3 *Um grafo G é de distância hereditária se e somente se todo subgrafo induzido de G tem ou um vértice pendente ou um par de gêmeos.*

Este teorema significa que: dado qualquer grafo de distância hereditária G com n vértices $1, 2, \dots, n$, para $2 \leq i \leq n$, o subgrafo de G induzido pelos vértices $\{1, 2, \dots, i\}$ é obtido do subgrafo induzido por $\{1, 2, \dots, i-1\}$ tornando o vértice i um vértice pendente ou um gêmeo falso ou um gêmeo verdadeiro de j . Desta maneira, cada palavra s_i da sequência (s_2, \dots, s_n) é respectivamente iPj , ou iFj , ou iTj para algum inteiro $j < i$. E, esta sequência de palavras, chamamos de sequência de poda.

Alternativamente, qualquer sequência de poda pode ser representada por uma árvore enraizada chamada *árvore de poda*.

Definição 2.3 *Seja G um grafo com n vértices denotados por v_1, \dots, v_n , e seja $S = \{s_2, \dots, s_n\}$ uma sequência de poda de G . A árvore de poda T correspondente a sequência de poda S de G é construída como segue:*

- (1) *Seja T_1 a árvore consistindo de um único vértice raiz v_1 e $i = 1$;*

- (2) $i \leftarrow i + 1$. Se $i > n$ então $T \leftarrow T_n$ e pare;
- (3) Seja $s_i = v_i P v_j$ (respectivamente $s_i = v_i F v_j$, ou $s_i = v_i T v_j$) então T_i é a árvore obtida de T_{i-1} adicionando-se o vértice v_i e tornando-o o filho mais a direita do vértice v_j , rotulando a aresta $v_i v_j$ em p (respectivamente f ou t);
- (4) Volte ao passo 2;

Por exemplo, $(2P1, 3P2, 4T1, 5F2, 6P4)$ é uma sequência de poda do grafo de distância hereditária da Figura 2.3 que pode ser representada pela árvore de poda da Figura 2.4.

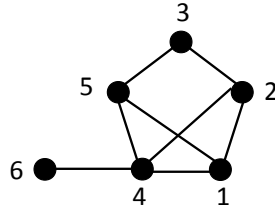


Figura 2.3: Exemplo de sequência de poda.

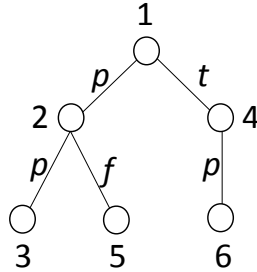


Figura 2.4: Exemplo de árvore de poda.

A próxima seção tem como objetivo introduzir o conceito de *clique-width* e apresentar a *clique-width* máxima de um grafo de distância hereditária, um importante resultado para esta dissertação.

2.2.1 Clique-width

A *clique-width* é um parâmetro que mede, em certo sentido, a complexidade de um grafo. Este parâmetro é definido via um processo de construção do grafo onde apenas um número limitado de rótulos estão disponíveis; vértices que compartilham o mesmo rótulo em um determinado momento do processo de construção devem ser tratados da mesma maneira nos passos subsequentes.

No que segue, apresentamos alguns conceitos para o entendimento deste parâmetro.

Um grafo G é *rotulado* em vértices (ou arestas) quando a cada vértice (ou aresta) estiver associado exatamente um rótulo. Neste trabalho, utilizaremos a denominação grafo rotulado para denotar o que definimos como grafo rotulado em vértices.

Um k -grafo é um grafo rotulado com rótulos em $\{1, 2, \dots, k\}$. Um k -grafo G é representado por uma estrutura (V, E, V_1, \dots, V_k) , onde V e E são os conjuntos de vértices e arestas, respectivamente, e V_1, \dots, V_k formam uma partição de V , tal que V_i é o conjunto de vértices com rótulo i em G .

Definimos quatro operações em grafos rotulados. Primeiro, denotamos por $i(v)$ a operação de criar um novo vértice v e associar, a este, o rótulo i . A segunda operação, denotada por $G \oplus H$, é união disjunta de dois grafos rotulados G e H . As outras duas operações são definidas a seguir.

Denotamos por $\eta_{i,j} = (V, E', V_1, \dots, V_k)$, a conexão de vértices com rótulos i e j onde $E' = E \cup \{(u, v) : u \in V_i, v \in V_j\}$, e por $\rho_{i \rightarrow j} = (V, E, V'_1, \dots, V'_k)$, a substituição do rótulo i para o rótulo j , onde $V'_i = \emptyset$, $V'_j = V_j \cup V_i$, e $V'_p = V_p$ para $p \neq i, j$.

Todo grafo pode ser definido por uma expressão algébrica usando estas quatro operações. Por exemplo, o grafo consistindo de dois vértices isolados x e y pode ser definido pela expressão $1(x) \oplus 1(y)$, e o grafo com dois vértices adjacentes x e y pode ser definido pela expressão $\eta_{1,2}(1(x) \oplus 2(y))$.

A noção de *clique-width* de grafos foi introduzida por Courcelle, Engelfriet e Rozenberg em [13]. A *clique-width* de um grafo G , denotada por $cwd(G)$, é definida como o número mínimo de rótulos necessários para construir G através das quatro operações de grafo definidas anteriormente.

Além disso, uma expressão construída a partir destas operações usando k rótulos diferentes é chamada uma k -expressão. Desta forma, a *clique-width* de um grafo G é o menor k tal que existe uma k -expressão definindo G . Dizemos que uma k -expressão t define um grafo G se G é equivalente ao grafo rotulado obtido de t .

Por exemplo, a *clique-width* de uma clique com quatro vértices é no máximo 2, uma vez que esta pode ser definida pela sequência abaixo que resulta numa 2-expressão:

Criação do vértice w com o rótulo 1

$1(w)$

Criação do vértice x com o rótulo 2

$$2(x)$$

União disjunta dos vértices x e w

$$1(w) \oplus 2(x)$$

Criação de arestas entre os vértices com rótulos 1 e 2

$$(\eta_{1,2}(1(w) \oplus 2(x)))$$

Alteração dos vértices com rótulo 2 para o rótulo 1

$$p_{2 \rightarrow 1}(\eta_{1,2}(1(w) \oplus 2(x)))$$

Criação do vértice v com rótulo 2 e realização da operação união disjunta deste vértice com o grafo construído até a etapa anterior

$$(2(v) \oplus p_{2 \rightarrow 1}(\eta_{1,2}(1(w) \oplus 2(x))))$$

Criação de arestas entre os vértices com rótulos 1 e 2

$$(\eta_{1,2}(2(v) \oplus p_{2 \rightarrow 1}(\eta_{1,2}(1(w) \oplus 2(x)))))$$

Alteração dos vértices com rótulo 2 para o rótulo 1

$$p_{2 \rightarrow 1}(\eta_{1,2}(2(v) \oplus p_{2 \rightarrow 1}(\eta_{1,2}(1(w) \oplus 2(x)))))$$

Criação do vértice u com rótulo 2 e realização da operação união disjunta deste vértice com o grafo construído até a etapa anterior

$$(2(u) \oplus p_{2 \rightarrow 1}(\eta_{1,2}(2(v) \oplus p_{2 \rightarrow 1}(\eta_{1,2}(1(w) \oplus 2(x)))))$$

Criação de arestas entre os vértices com rótulos 1 e 2

$$(\eta_{1,2}(2(u) \oplus p_{2 \rightarrow 1}(\eta_{1,2}(2(v) \oplus p_{2 \rightarrow 1}(\eta_{1,2}(1(w) \oplus 2(x)))))$$

Alteração dos vértices com rótulo 2 para o rótulo 1

$$p_{2 \rightarrow 1}(\eta_{1,2}(2(u) \oplus p_{2 \rightarrow 1}(\eta_{1,2}(2(v) \oplus p_{2 \rightarrow 1}(\eta_{1,2}(1(w) \oplus 2(x)))))$$

No exemplo acima, note que a última operação de substituição dos vértices com rótulo 2 para o rótulo 1 foi realizada com intuito de padronizar os rótulos, conforme realizado nas etapas anteriores, embora, esta não seja necessária na expressão que define uma clique com quatro vértices.

Agora, vamos apresentar as definições relacionadas ao Teorema 2.4, demonstrado por Golumbic e Rotics em [28], que nos fornece a clique-width máxima de grafos de distância hereditária.

Definição 2.4 *Seja T uma árvore enraizada, e seja x um nó de T . Denotamos por T_x a sub-árvore de T de raiz x .*

Definição 2.5 *Seja G um grafo possuindo uma sequência de poda S , e seja T a árvore de poda correspondente a S . Sejam u e v dois vértices quaisquer de G . Dizemos que v é um descendente gêmeo de u se $v = u$, ou v é um descendente de u em T tal que todas as arestas do caminho entre v e u são rotuladas em t ou f .*

Antes de enunciarmos o Teorema 2.4, ainda é necessário apresentar a Afirmação 2.1, que é a base do teorema.

Afirmação 2.1 [28] *Para cada nó interno de uma árvore de poda T , existe uma 3-expressão t_x que define o grafo rotulado G' , tal que $G' = G[T_x]$, todos os descendentes gêmeos de x são rotulados com 2 em G' , e todos os outros vértices de G' são rotulados com 1.*

Além disso, seja x qualquer nó interno de T cujos filhos ordenados da esquerda para a direita são x_1, \dots, x_l , a expressão t_x pode ser construída seguindo os passos abaixo.

- (1) $e_{l+1} \leftarrow 2(x); i \leftarrow l + 1;$
- (2) $i \leftarrow i - 1$. Se $i = 0$ então $t_x \leftarrow e_1$ e pare;
- (3) Se x_i é um vértice pendente de x então $e_i \leftarrow p_{3 \rightarrow 1}(\eta_{2,3}(e_{i+1} \oplus p_{2 \rightarrow 3}(t_{x_i})))$;
- (4) Se x_i é um gêmeo verdadeiro de x então $e_i \leftarrow p_{3 \rightarrow 2}(\eta_{2,3}(e_{i+1} \oplus p_{2 \rightarrow 3}(t_{x_i})))$;
- (5) Se x_i é um gêmeo falso de x então $e_i \leftarrow t_{x_i} \oplus e_{i+1}$;
- (6) Volte ao passo 2;

Para exemplificar esta afirmação e utilizando como referência a árvore de poda da Figura 2.4, seguem abaixo os passos para a construção da 3-expressão t do grafo de distância hereditária da Figura 2.3.

$$e_3 \leftarrow 2(1); i \leftarrow 3$$

$$i \leftarrow 2$$

$$e_2 \leftarrow p_{3 \rightarrow 2}(\eta_{2,3}(e_3 \oplus p_{2 \rightarrow 3}(t_{x_2})))$$

$$t_{a_2} \leftarrow \eta_{2,1}(2(4) \oplus 1(6))$$

$$e_2 \leftarrow p_{3 \rightarrow 2}(\eta_{2,3}(2(1) \oplus \eta_{3,1}(3(4) \oplus 1(6))))$$

$$i \leftarrow 1$$

$$e_1 \leftarrow p_{3 \rightarrow 1}(\eta_{2,3}(e_2 \oplus p_{2 \rightarrow 3}(t_{x_1})))$$

$$t_{x_1} \leftarrow \eta_{2,1}(2(5) \oplus 2(2) \oplus 1(3))$$

$$e_1 \leftarrow p_{3 \rightarrow 1}(\eta_{2,3}(p_{3 \rightarrow 2}(\eta_{2,3}(2(1) \oplus \eta_{3,1}(3(4) \oplus 1(6)))) \oplus p_{2 \rightarrow 3}(\eta_{2,1}(2(5) \oplus 2(2) \oplus 1(3)))))$$

$$e_1 \leftarrow p_{3 \rightarrow 1}(\eta_{2,3}(p_{3 \rightarrow 2}(\eta_{2,3}(2(1) \oplus \eta_{3,1}(3(4) \oplus 1(6)))) \oplus \eta_{3,1}(3(5) \oplus 3(2) \oplus 1(3))))$$

$$t \leftarrow e_1$$

Como verificado na Afirmação 2.1, existe uma 3—expressão t que define um grafo de distância hereditária G . Sendo assim, podemos apresentar a clique-width máxima dos grafos de distância hereditária através do Teorema 2.4 abaixo.

Teorema 2.4 [28] *Todo grafo de distância hereditária possui clique-width no máximo 3.*

Além dos grafos de distância hereditária, outras classes de grafos possuem clique-width limitada, por exemplo: todos os grafos completos K_n , $n \geq 2$, possuem clique-width igual a 2, os cografos, são os grafos com clique-width no máximo 2 e as árvores possuem clique-width no máximo 3 [15].

2.3 Relação dos grafos de distância hereditária com outras famílias de grafos

Os grafos de distância hereditária formam uma subclasse dos grafos perfeitos [2, 30]. Como vimos no capítulo anterior, um grafo é perfeito se e somente se não contém, como subgrafo induzido, ciclo ímpar de tamanho maior ou igual a cinco ou o seu complemento.

Através desta caracterização, verificamos facilmente que todo grafo de distância hereditária G é perfeito. Note que, se G é um grafo de distância hereditária, então G não contém buraco, sendo assim, G também não contém ciclo ímpar maior ou igual a 5 (C_5 , C_7 , ...)

e $\overline{C_5}$ como subgrafo induzido. Além disso, G não contém leque, portanto, G também não contém o complemento de ciclos ímpares maiores do que 5 ($\overline{C_7}, \overline{C_9}, \dots$) como subgrafo induzido, portanto, G é perfeito.

Por outro lado, a classe dos grafos de distância hereditária contém uma importante classe de grafos, os cografos (conforme explicado na próxima seção). Desta maneira, temos que: cografos \subseteq grafos de distância hereditária \subseteq grafos perfeitos.

2.3.1 Cografos

Os cografos surgiram em diversas áreas da matemática. Sendo objeto de estudo de vários pesquisadores independentemente. Por isso, existem vários sinônimos para o termo cografo na literatura, dentre os quais podemos citar: grafos livres de P_4 , grafos D^* , grafos HD e, grafos redutíveis por complemento. O termo cografo, tal como conhecemos hoje, foi introduzido, na década de 70, por H. Lerchs que estudou suas propriedades estruturais e algorítmicas [35].

Definição 2.6 [11] *Os cografos são definidos recursivamente através da seguinte definição formal:*

- (i) *O grafo trivial é um cografo, isto é, o K_1 é um cografo;*
- (ii) *Se (G_1, \dots, G_k) são cografos, então a 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.*

Como vimos na Definição 2.6, um cografo pode ser obtido a partir de um conjunto de grafos triviais, através de um número finito de operações de união e junção. Além disso, ele pode ser representado através de uma árvore de decomposição única, chamada co-árvore [35]. Esta representação é a chave para o reconhecimento de tal classe, com complexidade linear, introduzido por Corneil *et al.* [12].

2.3.1.1 Propriedades Estruturais

Nesta seção, apresentamos algumas propriedades estruturais dos cografos sem realizar a demonstração das mesmas.

Teorema 2.5 [2] *Um grafo G é um cografo se e somente se G pode ser gerado de um único vértice por uma sequência de extensões consistindo de operações para adicionar um gêmeo falso ou um gêmeo verdadeiro.*

Através do Teorema 2.5, verificamos que os cografos podem ser construídos da mesma forma que os grafos de distância hereditária (Veja Teorema 2.2). A única diferença é que, no caso dos cografos, não se utiliza a operação de adição de vértices pendentes. Portanto, todo cografo é um grafo de distância hereditária. Este fato também pode ser verificado pelo teorema abaixo.

Teorema 2.6 [2] *Um grafo G é um cografo se e somente se G é a união disjunta de grafos de distância hereditária com diâmetro no máximo 2.*

Nas próximas seções abordaremos com mais detalhes a decomposição modular e a co-árvore.

2.3.1.2 Decomposição Modular

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.

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 definição abaixo, podemos entender o conceito de módulo.

Definição 2.7 [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 .*

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

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

2.3.1.3 Co-árvore

Como mencionado anteriormente, um cografo pode ser representado por uma árvore de decomposição, chamada co-árvore, a qual é única a menos de isomorfismo.

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

Teorema 2.7 [11] *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.

Observamos, portanto, que o complemento de todo subgrafo conexo não trivial de um cografo G é desconexo. Sendo assim, a árvore de decomposição modular de G contém apenas os módulos serial e paralelo.

A partir deste ponto em diante representaremos a co-árvore de um grafo G da seguinte forma: as folhas representarão os vértices do grafo e os nós da árvore serão rotulados por 0 ou 1. A raiz receberá rótulo 1, os filhos de um nó com rótulo 1 serão rotulados 0 e os filhos de um nó 0, receberão rótulo 1. Ou seja, os nós internos em cada caminho da raiz até as folhas na árvore alternarão entre 0 e 1, onde os nós rotulados por 0 corresponderão ao módulo paralelo, e os nós rotulados por 1 corresponderão ao módulo serial. Por isso, os nós internos da co-árvore serão chamados de *nós tipo-0* ou *nós tipo-1*. Outra característica importante é que cada nó interno tem pelo menos dois filhos (exceto o cografo trivial).

Para exemplificar, veja na Figura 2.5 um cografo e sua co-árvore.

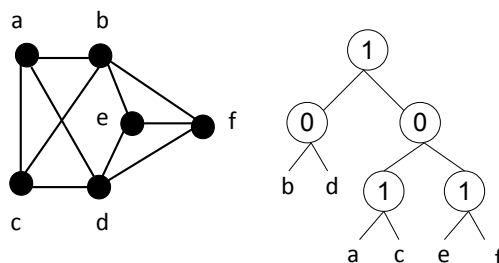


Figura 2.5: Exemplo de um cografo e sua co-árvore.

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

Propriedade 2.1 *Dois vértices x e y de G são adjacentes (não-adjacentes) se e somente se o menor ancestral comum a x e a y for serial (paralelo).*

Em outras palavras, dois vértices x e y de G são adjacentes se e somente se o menor ancestral comum a x e a y for um *nó tipo-1*.

Além disso, dois vértices que tem o mesmo nó pai em uma co-árvore são gêmeos verdadeiros (gêmeos falsos) se e somente se o nó pai é um *nó tipo-1* (respectivamente *nó tipo-0*).

2.4 Reconhecimento de grafos de distância hereditária

Como mencionado anteriormente, um algoritmo com complexidade linear para reconhecimento de grafos de distância hereditária através da sequência de poda foi proposto por Hammer e Maffray em [30]. O resultado deste algoritmo é apresentar uma sequência de poda se e somente se o grafo de entrada é de distância hereditária. No entanto, Damiand [16] encontrou um contra-exemplo para o mesmo, mostrando que o algoritmo responderia que existe uma sequência de poda embora o grafo dado não fosse de distância hereditária.

Em 2001, Damiand, Habib e Paul [17] apresentaram a correção do algoritmo proposto por Hammer e Maffray. O algoritmo corrigido, além de calcular a sequência de poda do grafo de entrada (Algoritmo 1), verifica sua corretude através do Algoritmo 3 mantendo a complexidade linear do algoritmo inicial.

Este algoritmo de reconhecimento de grafos de distância hereditária é baseado no Teorema 2.8 demonstrado por Bandelt e Mulder em [2]. Antes de enunciarmos o Teorema, vamos precisar da definição de *layout de distância*, apresentada abaixo.

O *layout de distância* de um grafo conexo $G = (V, E)$ em um vértice $u \in V$ é a coleção de conjuntos L_0, L_1, \dots, L_t onde $t = \max_{v \in V} d_G(u, v)$ e $L_i = \{v \in V : d_G(u, v) = i\}$ para $0 \leq i \leq t$. Para qualquer vértice x e para $1 \leq i \leq t$, denotamos por $N_i(x) = N(x) \cap L_i$. Agora, seja x um vértice de L_i , chamamos de *grau interno* de x a cardinalidade de $N_{i-1}(x)$.

Teorema 2.8 [2] *Seja G um grafo conexo e L_1, \dots, L_k seja o layout de distância de um vértice arbitrário v de G . Então G é um grafo de distância hereditária se e somente se as condições seguintes são satisfeitas para qualquer $1 \leq i \leq k$:*

- (1) *Se x e y pertencem a mesma componente conexa de $G[L_i]$, então $N_{i-1}(x) = N_{i-1}(y)$;*
- (2) *$G[L_i]$ é um cografo;*

- (3) Se $x \in N_{i-1}(u)$ e $y \in N_{i-1}(u)$ estão em diferentes componentes conexas X e Y de $G[L_{i-1}]$, então $X \cup Y \subseteq N(u)$ e $N_{i-2}(x) = N_{i-2}(y)$;
- (4) Se x e y estão em diferentes componentes conexas de $G[L_i]$, então $N_{i-1}(x)$ e $N_{i-1}(y)$ são ou disjuntos ou um dos dois conjuntos está contido no outro;
- (5) Se $x \in N_{i-1}(u)$ e $y \in N_{i-1}(u)$ estão na mesma componente conexa C de $G[L_{i-1}]$, então os vértices de C não adjacentes a u são ou adjacentes a ambos x e y ou a nenhum deles;

Algoritmo 1 Reconhecimento-dhg (G)

Entrada: Grafo $G = (V, E)$;
 Saída: Sequência de poda (S, σ) sss G é um grafo de distância hereditária;
 $j \leftarrow 1$;
 Computar a distância L_1, \dots, L_k de um vértice arbitrário v ;
para $i \leftarrow k$ até 1 **faça**
 para cada componente conexa cc de $G[L_i]$ **faça**
 $z \leftarrow \text{Reconhecimento-cografo}(G[cc], j)$;
 reduza cc em z ;
 $j \leftarrow j + |cc| - 1$
fim para
 ordene os vértices de $G[L_i]$ em ordem crescente de grau interno
para cada vértice x de L_i com grau = 1 **faça**
 Seja y o único vizinho de x ;
 $\sigma(j) \leftarrow x$ e $s_j \leftarrow (xPy)$;
 $j \leftarrow j + 1$;
fim para
se $i \neq 1$ e grau interno $\neq 1$ **então**
 para cada $x \in L_i$ em ordem crescente de grau interno **faça**
 $y \leftarrow \text{Reconhecimento-cografo}(G[N_{i-1}(x)], j)$;
 reduza $N_{i-1}(x)$ em y ;
 $j \leftarrow j + |N_{i-1}(x)| - 1$;
 $\sigma(j) \leftarrow x$ e $s_j \leftarrow (xPy)$;
 $j \leftarrow j + 1$;
 fim para
fim se
fim para

Conforme Teorema 2.8 (2), se G é um grafo de distância hereditária e L_1, \dots, L_k é o layout de distância de um vértice arbitrário v , então $G[L_i]$ é um cografo. Desta maneira, como verificamos no Algoritmo 1, o algoritmo de reconhecimento de grafos de distância hereditária utiliza, como subrotina, um algoritmo de reconhecimento de cografos. Este algoritmo, descrito no Algoritmo 2, usa uma co-árvore para computar a sequência de poda de um cografo.

Como exemplo, a Figura 2.6 apresenta o layout de distância (conjuntos L_1 e L_2) do vértice 1 do grafo de distância hereditária da Figura 2.3. Observe que $G[L_1]$ e $G[L_2]$ são cografos.

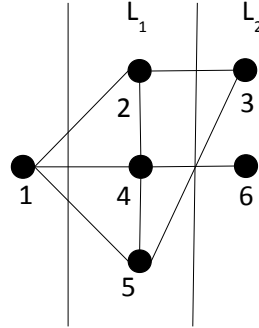


Figura 2.6: Layout de distância.

Além disso, utilizando este mesmo grafo como entrada para o Algoritmo 1, temos como resultado, a seguinte sequência de poda: ($S=[6P4, 5F2, 3P2, 4T2, 2P1]$, $\sigma=[6, 5, 3, 4, 2, 1]$).

Agora, através dos Teoremas 2.9, 2.10 e 2.11 vamos analisar a complexidade total dos algoritmos apresentados.

Teorema 2.9 [30] *O Algoritmo 2 computa a sequência de poda de um cografo G em $O(n+m)$.*

Com o objetivo de obter complexidade linear para o Algoritmo 2, temos que construir a co-árvore em tempo linear. Para isso, podemos utilizar o clássico algoritmo de Corneil, Perl e Stewart [12] ou um algoritmo mais recente de Bretscher, Corneil, Habib e Paul [9]. O restante do algoritmo pode ser comparado a um percurso pós-ordem em uma árvore. Como o tamanho da co-árvore é $O(n+m)$, a complexidade total é $O(n+m)$.

A complexidade total do Algoritmo 1 é demonstrada também em [17] através do Teorema 2.10.

Teorema 2.10 [17] *O Algoritmo 1 calcula a sequência de poda de G se e somente se G é um grafo de distância hereditária. Além disso, o Algoritmo 1 pode ser executado em $O(n+m)$.*

Finalmente, apresentamos o Algoritmo 3, utilizado na etapa de verificação, com algumas mudanças de notação a fim de facilitar o entendimento. Neste algoritmo, cada vértice e sua vizinhança são visitados no máximo duas vezes, mantendo a complexidade linear.

Algoritmo 2 Reconhecimento-cografo (G, j)

Entrada: Cografo G ;
 Saída: Sequência de poda (S, σ) e o último vértice da sequência sss G é um cografo;
 Compute a co-árvore T de G (usando um algoritmo de reconhecimento de cografo);
 Seja A o conjunto dos nós de T que possuem apenas folhas como descendentes;
enquanto $A \neq \emptyset$ **faça**
 Escolha um nó arbitrário $N \in A$;
 Escolha um filho arbitrário x de N ;
 para cada filho $y \neq x$ de N **faça**
 se N é um *nó tipo-1* **então**
 $\sigma(j) = y$ e $s_j \leftarrow (yTx)$
 senão
 $\sigma(j) = y$ e $s_j \leftarrow (yFx)$
 fim se
 $j \leftarrow j + 1$;
 fim para
 Substitua N por x em T ;
 se x é a raiz de T **então**
 Retorne: x é o último vértice da sequência
 fim se
 se O nó pai N tem apenas folhas como descendentes **então**
 Adicione N a A
 fim se
fim enquanto

Algoritmo 3 Verificação(G, σ)

Entrada: Grafo $G = (V, E)$ e uma sequência (S, σ) onde $S = [s_1, \dots, s_{n-1}]$;
 Saída: Verdadeiro se e somente se S é uma sequência de poda de G ;
para $j = n - 1$ até 1 **faça**
 $xQy \leftarrow s_j$
 se $Q = P$ **então**
 se $|N(x) \setminus \{x_1, \dots, x_{j-1}\}| \neq 1$ ou $(x, y) \notin E$ **então**
 return false;
 fim se
 senão
 se $N(x) \setminus \{x_1, \dots, x_{j-1}\} \neq N(y) \setminus \{x_1, \dots, x_{j-1}\}$ **então**
 se $N[x] \setminus \{x_1, \dots, x_{j-1}\} \neq N[y] \setminus \{x_1, \dots, x_{j-1}\}$ **então**
 return false;
 fim se
 fim se
 fim se
 return true;
fim para

Vale ressaltar que para reconhecer grafos de distância hereditária utilizamos os Algoritmos 1 e 3 combinados. Assim, temos o seguinte resultado.

Teorema 2.11 [17] *Usando o Algoritmo 1 e o Algoritmo 3, grafos de distância hereditária podem ser reconhecidos em tempo linear.*

Além dos algoritmos apresentados anteriormente, Gioan e Paul, em 2007, apresentaram um algoritmo dinâmico para inserção e remoção de vértices para grafos de distância hereditária com complexidade $O(d)$ (com d sendo o grau do vértice envolvido) [27]. O algoritmo para inserção de vértices pode ser usado para obter um algoritmo de reconhecimento (estático) de grafos de distância hereditária. A técnica utilizada baseou-se na decomposição split [27].

Em 2009, Nakano, Uehara e Uno, também apresentaram em [39], um algoritmo para reconhecimento de grafos de distância hereditária com complexidade linear. Neste caso, árvores de prefixos foram utilizadas para manter a vizinhança dos vértices com a finalidade de encontrar mais eficientemente os nós que representam os gêmeos falsos e verdadeiros do grafo. Embora, sua complexidade para reconhecer grafos de distância hereditária seja a mesma do algoritmo apresentado anteriormente nesta dissertação, $O(n + m)$.

Capítulo 3

Transversais de Ciclos

3.1 Introdução

Neste capítulo, temos como objetivo apresentar o conceito de transversais de ciclos.

Inicialmente, abordaremos o problema de encontrar transversais de ciclos mínimos abrangendo, adicionalmente, o problema de encontrar transversais de ciclos ímpares.

Além disso, definiremos a classe de grafos denominada grafos- (k, l) e, uma de suas subclasses, a classe dos grafos- $(2, 1)$.

Por fim, estudaremos os grafos- $(\mathcal{F}, \mathcal{C})$ e introduziremos o conceito de transversais-cliques de ciclos. Este conceito será fundamental para o entendimento dos demais capítulos desta dissertação.

3.2 Transversal de Ciclos

Um *transversal de ciclos* de um grafo G é um subconjunto $T \subseteq V(G)$ tal que $T \cap V(C) \neq \emptyset$ para todo ciclo C de G , ou equivalentemente, todo ciclo em G contém pelo menos um vértice em T . Sendo assim, temos que $V(G) \setminus T$ é uma floresta (ou seja, não contém ciclo).

Na literatura, o problema de decisão nomeado transversal de ciclos é formulado como segue: dado um grafo G e um inteiro positivo k , existe um transversal de ciclos T de G satisfazendo $|T| \leq k$? Desta maneira, encontrar o transversal mínimo de ciclos é encontrar o menor valor de k que torna o grafo acíclico após a remoção de T . Além disso, encontrar o transversal mínimo de ciclos é equivalente a encontrar a floresta induzida máxima.

Encontrar um transversal mínimo de ciclos em um grafo é NP-difícil devido ao resultado geral em [43], que diz que o problema de encontrar o número mínimo de vértices de um

grafo G cuja deleção resulta em um subgrafo satisfazendo uma propriedade hereditária π em subgrafos induzidos é NP-difícil. Este resultado implica que outros problemas envolvendo transversais mínimos de ciclos também são NP-difíceis, por exemplo, o problema de encontrar transversal mínimo de ciclos ímpares (que é equivalente a encontrar um subgrafo bipartido induzido máximo), ou o problema de encontrar um transversal mínimo de triângulo (que é equivalente a encontrar um subgrafo induzido máximo livre de triângulo).

No entanto, este torna-se polinomial quando restrito a famílias específicas de grafos, tais como grafos *de intervalo* ($O(n + m)$) [38], grafos *de permutação* ($O(nm)$) [36], grafos *de cocomparabilidade* ($O(n^2m)$) [18], grafos *bipartidos convexos*¹ ($O(n^2m)$) [18], entre outros. Além destes, existem também algoritmos exatos para encontrar um transversal mínimo de ciclos em grafos gerais com n vértices com complexidade $O((1.8899)^n)$ e $O((1.7548)^n)$ em [40] e [25], respectivamente.

Além disso, recentemente, pesquisadores apresentaram outros estudos sobre transversais de ciclos em várias famílias de grafos. Por exemplo, em [29], os autores estudaram o problema de encontrar transversais de C_k (isto é, um subconjunto de vértices que interceptam todos os ciclos induzidos sem corda com k vértices), para um inteiro fixo k , em grafos com grau limitado; entre outros resultados, eles descreveram um algoritmo polinomial para encontrar transversais mínimos de C_4 em grafos com grau máximo três.

Em [5], Brandstädt, Brito, Klein, Nogueira e Protti provaram que encontrar um transversal mínimo de ciclos em grafos perfeitos é NP-completo, mesmo para grafos bipartidos com grau máximo 4. Eles também mostraram que este problema é solucionado em tempo linear para cografos.

3.2.1 Transversais de ciclos ímpares

Outro problema que se insere neste contexto e de grande interesse devido a sua conexão com a teoria dos grafos perfeitos é encontrar transversais de ciclos ímpares de tamanho no máximo k .

Um transversal de ciclos ímpares para um grafo G é um subconjunto $S \subseteq V(G)$ tal que $G \setminus S$ é bipartido. Ou seja, todo ciclo de comprimento ímpar contém pelo menos um vértice em S . Desta maneira, o objetivo é determinar se existe um subconjunto S de tamanho no máximo k tal que $G \setminus S$ é bipartido.

¹Seja G um grafo bipartido com bipartição (X, Y) . A ordenação \prec de X possui a propriedade de adjacência se para todo $y \in Y$, $N(y)$ consiste dos vértices que são consecutivos na ordenação \prec de X . Um grafo bipartido G é convexo se existe uma ordenação de X ou Y que satisfaz a propriedade de adjacência.

Observe que o problema de encontrar transversais mínimos de ciclos ímpares é equivalente a encontrar o maior subgrafo bipartido induzido. O nome transversal de ciclos ímpares vem do fato de que um grafo é bipartido se e somente se não possui ciclo ímpar.

Em 2004, Reed, Smith e Vetta [41] provaram que encontrar um conjunto mínimo de vértices tal que ao removê-los torna o grafo bipartido pode ser resolvido em $O(4^k kmn)$ em um grafo com n vértices e m arestas, onde k é o número de vértices a remover. Posteriormente, foi observado por Hüffner em [33] que o tempo de execução do algoritmo apresentado por Reed *et al.* em 2004 era na verdade $O(3^k kmn)$. Recentemente, Lokshtanov propôs um algoritmo com tempo de execução $O(2.32^k n^{O(1)})$ usando técnicas de programação linear [37].

Antes de iniciarmos o conceito de transversal-clique de ciclos, vamos introduzir uma classe de grafos definida por Brandstädt em [3], denominada grafos- (k, l) .

3.3 Grafos- (k, l)

Um grafo G é um grafo- (k, l) se seu conjunto de vértices pode ser particionado em k conjuntos independentes e l cliques. Mais formalmente, denotamos por (k, l) o conjunto de todos os grafos $G = (V, E)$ para os quais existe uma partição- (k, l) do conjunto de vértices V .

Uma partição- (k, l) de G é uma partição $S_1, \dots, S_k, C_1, \dots, C_l$ de V com os conjuntos independentes S_j , para $j \in \{1, \dots, k\}$ e cliques C_i , para $i \in \{1, \dots, l\}$. É importante ressaltar que alguns dos k conjuntos independentes ou das l cliques podem ser vazios e, além disso, as cliques não são necessariamente maximais.

Observe que grafos- (k, l) são uma generalização natural de grafos split uma vez que estes são precisamente grafos- $(1, 1)$ (obtidos quando $k = l = 1$). Por outro lado, a classe de grafos bipartidos corresponde aos grafos- $(2, 0)$. Além disso, quando $l = 0$, temos os grafos- $(k, 0)$, que correspondem aos grafos que admitem uma k -coloração.

Brandstädt em [3, 4, 7] apresentou um algoritmo polinomial de reconhecimento $O((n + m)^2)$ para grafos- $(2, 1)$ e um algoritmo de reconhecimento $O((n + \overline{m})^2)$ para grafos- $(1, 2)$, onde \overline{m} denota o número de arestas no complemento. Feder, Hell, Klein e Motwani em [23] também apresentaram algoritmos polinomiais para o reconhecimento desta classe. No entanto, Brandstädt provou que o problema de reconhecimento de grafos- (k, l) é NP-Completo para $k \geq 3$ ou $l \geq 3$. A prova de NP-completude utiliza uma redução do

problema da 3-coloração ao problema de reconhecer grafos- (k, l) .

Por este motivo, muitos trabalhos têm considerado famílias especiais de grafos- (k, l) que podem ser reconhecidos eficientemente; por exemplo, grafos cordais- (k, l) [31], cografos- (k, l) [19, 20, 22], e grafos perfeitos- (k, l) [21].

Outros estudos também foram realizados para esta classe de grafos. Por exemplo, em [31] os autores deram uma caracterização por subgrafos proibidos de grafos cordais- (k, l) . Além disso, como informado anteriormente, também apresentaram um algoritmo de reconhecimento para tais grafos com complexidade $O(n(m + n))$. Mais especificamente, este algoritmo de reconhecimento encontra o valor mínimo de l tal que G é um grafo- (k, l) . Quando $k = 1$, isto é, quando há uma partição com um conjunto independente e uma clique, o algoritmo é mais eficiente. Neste caso, possui complexidade $O(m + n)$.

Além disso, Feder *et al.* [23] generalizaram o problema da (k, l) -partição como o problema da M -partição, isto é, particionar o conjunto de vértices de um grafo em m partes A_1, A_2, \dots, A_m tal que cada A_i é um conjunto independente ou completo e, ainda, cada par A_i, A_j são completamente não adjacentes ou completamente adjacentes entre si.

3.3.1 Grafos- $(2, 1)$

Um grafo é $(2, 1)$ se pode ter seu conjunto de vértices particionado em dois conjuntos independentes e uma clique, ou equivalentemente, particionado em um grafo bipartido e uma clique. Através desta definição, como um grafo G é bipartido se e somente se não possui ciclo ímpar, podemos dizer que um grafo é $(2, 1)$ se contém uma clique que intercepta todo ciclo ímpar, pois, caso contrário, não seria possível obter a bipartição induzida pelos vértices de G que não pertencem à clique.

Observe que um grafo- $(2, 1)$ tem uma partição- $(2, 1)$: S_1, S_2, C , em que S_1 e S_2 são os conjuntos independentes e C é a clique. Desta maneira, para todo vértice $v \in C$, a não vizinhança de v , denotada por $\overline{N}(v)$, pertence a partição- $(2, 0)$. E, para todo vértice $v \in I_1 \cup I_2$, a vizinhança de v pertence a partição- $(1, 1)$.

Assim sendo, os vértices de um grafo- $(2, 1)$ podem ser classificados de acordo com as suas condições de vizinhanças. Uma consequência imediata é que se G possui um vértice $v \in V$ com $N(v) \notin (1, 1)$ e $\overline{N}(v) \notin (2, 0)$, então G não é um grafo- $(2, 1)$. Este foi o princípio para o reconhecimento para grafos- $(2, 1)$ e grafos- $(1, 2)$ utilizado por Brandstäd em [3].

Desta forma, para os vértices que preenchem exatamente uma destas condições, podemos determinar em que parte da partição- $(2, 1)$ eles pertencem (e se existe tal partição). Isso

significa que a decisão se G é um grafo- $(2,1)$ pode ser iniciada da seguinte maneira:

- (1) Seja $C^F := \{v : N(v) \notin (1,1) \text{ e } \overline{N}(v) \in (2,0)\}$. Checar se C^F é uma clique, ou seja, checar se os vértices cuja vizinhança não pertence a $(1,1)$ e pertencem a $(2,0)$ formam uma clique.
- (2) Sejam $S_1^F \cup S_2^F := \{v : N(v) \in (1,1) \text{ e } \overline{N}(v) \notin (2,0)\}$. Checar se $S_1^F \cup S_2^F$ é bipartido, ou seja, checar se os vértices cuja vizinhança pertence a $(1,1)$ e não pertencem a $(2,0)$ formam um conjunto bipartido.

Se (1) ou (2) falhar, G não é um grafo- $(2,1)$. Caso contrário, verificam-se os vértices que preenchem ambas as condições, 1 e 2. Seja R o conjunto de tais vértices.

$$R := \{v : N(v) \in (1,1) \text{ e } \overline{N}(v) \in (2,0)\}.$$

Se R é \emptyset , então C^F, S_1^F, S_2^F é uma partição- $(2,1)$ de G . Senão, realiza-se uma extensão da clique e dos conjuntos bipartidos e uma nova verificação é realizada para checar se C^F permanece uma clique e $S_1^F \cup S_2^F$ permanece bipartido. Para mais detalhes sobre o procedimento para a realização da extensão da partição veja em [3].

3.4 Grafos- $(\mathcal{F}, \mathcal{C})$

Quando um transversal de ciclos $T \subseteq V(G)$ tem a propriedade adicional de induzir uma clique, dizemos que T é um *transversal-clique de ciclos* ou simplesmente *cct*. Desta maneira, um grafo G admite um cct se e somente se puder ser particionado em uma floresta e um subgrafo completo; por esta razão, como tais grafos admitem uma partição floresta-clique, são chamados de *grafos- $(\mathcal{F}, \mathcal{C})$* . Em alguns trabalhos, estes grafos também são chamados de grafos- $(\mathcal{C}, \mathcal{F})$ [5].

Observe que os grafos- $(\mathcal{F}, \mathcal{C})$ formam uma subclasse dos grafos- $(2,1)$ (grafos cujo conjunto de vértices pode ser particionado em um grafo bipartido e uma clique). Isto se deve pelo seguinte fato: como uma floresta é um grafo acíclico então toda floresta é um grafo bipartido.

3.4.1 Reconhecimento de grafos- $(\mathcal{F}, \mathcal{C})$

Iniciamos esta seção observando que os grafos- $(\mathcal{F}, \mathcal{C})$ podem ser facilmente reconhecidos em tempo polinomial. Como vimos anteriormente, os grafos- $(\mathcal{F}, \mathcal{C})$ formam uma subclasse dos grafos- $(2,1)$. Sendo assim, podemos utilizar um algoritmo de reconhecimento de grafos- $(2,1)$ como parte do algoritmo para reconhecer grafos- $(\mathcal{F}, \mathcal{C})$. Note que se G não

for um grafo- $(2, 1)$, G também não é um grafo- $(\mathcal{F}, \mathcal{C})$. A estratégia para reconhecer um grafo- $(\mathcal{F}, \mathcal{C})$ inicialmente verifica se G é um grafo- $(2, 1)$, o que pode ser feito em tempo polinomial. Em caso afirmativo, testa para cada clique candidata Q de uma partição- $(2, 1)$ de G , se $G - Q$ é acíclico (o que pode ser feito em tempo linear). Se este teste falhar para todas as cliques Q , então G não é um grafo- $(\mathcal{F}, \mathcal{C})$, caso contrário, G é um grafo- $(\mathcal{F}, \mathcal{C})$. Para concluir o argumento, afirmamos que o número de cliques candidatas Q é polinomial uma vez que G é um grafo- $(2, 1)$. Seja (B, Q) uma partição- $(2, 1)$ de $V(G)$ onde B induz um subgrafo bipartido e Q é uma clique. Agora, seja (B', Q') uma outra partição- $(2, 1)$ de $V(G)$. Então $|Q' \setminus Q| \leq 2$ e $|Q \setminus Q'| \leq 2$, caso contrário $G[B]$ ou $G[B']$ conteria um triângulo, que é impossível. Portanto, podemos gerar em tempo polinomial todas as outras cliques candidatas Q' de Q .

O Algoritmo 4 a seguir descreve os passos para o reconhecimento dos grafos- $(\mathcal{F}, \mathcal{C})$.

Algoritmo 4 Reconhecimento grafos- $(\mathcal{F}, \mathcal{C})$

```

Entrada: Grafo  $G = (V, E)$ ;
se  $G$  é  $(2, 1)$  então
    para cada Clique  $C$  de uma partição- $(2, 1)$  faça
        se  $G - C$  é acíclico então
            Retorne " $G$  é um grafo- $(\mathcal{F}, \mathcal{C})$ ";
        senão
            Retorne " $G$  não é um grafo- $(\mathcal{F}, \mathcal{C})$ ";
        fim se
    fim para
fim se

```

A corretude do algoritmo baseia-se no seguinte fato: se não é possível encontrar uma clique que seja transversal de ciclos ímpares, isto é, uma clique C que ao ser removida de uma grafo G , torna o grafo G bipartido, então também não é possível encontrar uma clique C tal que $G - C$ é uma floresta, ou seja, não possui ciclos.

Capítulo 4

Transversais-clique de Ciclos em Grafos de Distância Hereditária

4.1 Introdução

Como já vimos no Capítulo 2, um grafo é de distância hereditária se as distâncias em qualquer subgrafo conexo são as mesmas que as no grafo original. Vimos também que um transversal de ciclo é um subconjunto de vértices que intercepta todos os ciclos do grafo. Se este subconjunto de vértices tem a propriedade adicional de ser clique, dizemos que é um transversal-clique de ciclos.

Este capítulo tem como objetivo realizar um estudo sobre transversais-clique de ciclos em grafos de distância hereditária, ou seja, verificar se um grafo de distância hereditária G admite uma clique que intercepte todos os ciclos do grafo. Como resultado principal, apresentamos a caracterização de tais grafos através de subgrafos proibidos.

Para isso, inicialmente, apresentaremos uma caracterização e algumas notações e propriedades dos grafos de distância hereditária, também chamados de grafos livres de $HHDG$, que serão fundamentais para o desenvolvimento deste capítulo.

Conforme Teoremas 2.1 e 2.2, temos o seguinte resultado:

Teorema 4.1 [2] *São equivalentes as seguintes afirmações para qualquer grafo G :*

- (i) G é um grafo de distância hereditária.
- (ii) G pode ser construído a partir de um único vértice adicionando-se repetidamente um vértice pendente, um gêmeo falso, ou um gêmeo verdadeiro, respectivamente.
- (iii) G é livre de (casa, buraco, dominó, leque). (Veja Figura 2.2.)

Seja $G = (V, E)$ um grafo e $x \in V$, utilizaremos, deste ponto em diante, a notação $N^2(x) = \{y \in V \mid \text{dist}_G(x, y) = 2\}$, para representar o conjunto dos vértices de G que distam exatamente 2 em relação a x em G . Além disso, denotamos por $R = V \setminus (N[x] \cup N^2(x))$, os vértices de G que possuem uma distância maior do que 2 de x .

As seguintes propriedades em relação aos grafos de distância hereditária também serão úteis para o nosso resultado.

Proposição 4.1 *Seja G um grafo de distância hereditária e $u, v \in N^2(x)$.*

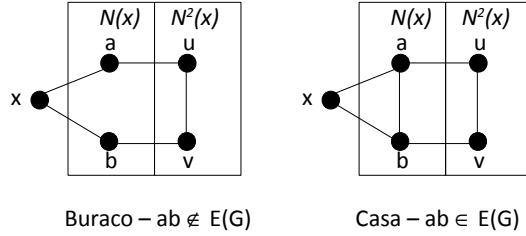
- (i) *Se $uv \in E(G)$ ou $uv \notin E(G)$ mas são conectados por um caminho em $N^2(x) \cup R$ então $N(u) \cap N(x) = N(v) \cap N(x)$.*
- (ii) *Se $N(x)$ é um conjunto independente então para $u, v \in N^2(x)$ com $uv \notin E$, $N(u) \cap N(x)$ e $N(v) \cap N(x)$ não se sobrepõem.*

Prova.

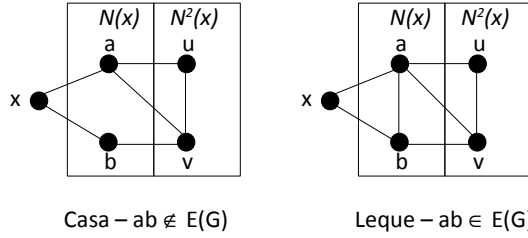
(i): Se $uv \in E(G)$ então u, v não podem ter vizinhanças incomparáveis em $N(x)$ uma vez que G é livre de (casa, buraco) (Figura 4.1(a)). Além disso, como G é livre de (casa, leque) (Figura 4.1(b)), as vizinhanças de u e v em $N(x)$ não podem conter propriamente uma a outra. Para finalizar, temos que considerar o caso que $uv \notin E(G)$ mas existe um caminho em $N^2(x) \cup R$. Para isso, vamos considerar o menor caminho entre u e v : se o primeiro vértice do caminho for $u' \in N^2(x)$ então u e u' têm a mesma vizinhança já que $u, u' \in E(G)$ e a prova deve iniciar para u' e v . Desta maneira, vamos assumir sem perda de generalidade que o primeiro vértice do caminho seja $r \in R$. Neste caso, temos três possibilidades. Na primeira, $rv \in E(G)$, sendo assim, como G é livre de buraco (Figura 4.1(c)), u, v não podem ter vizinhanças incomparáveis em $N(x)$ e, além disso, uma vez que G é livre de (casa, dominó) (Figura 4.1(d)), as vizinhanças de u e v em $N(x)$ não podem conter propriamente uma a outra. Na segunda possibilidade, r é adjacente a um vértice $u' \in N^2(x)$ que não pode ser adjacente a u pois não seria um caminho mínimo e, agora, u e u' devem ter a mesma vizinhança caso contrário temos uma contradição pois ur e $ru' \in E(G)$ e a prova deve iniciar para u' e v . Por último, se no caminho entre u e v , r for adjacente a um outro vértice $r' \in R$ teremos um buraco em todos os casos, o que é uma contradição. Isto prova a Proposição 4.1 (i).

(ii): Suponha, por contradição, que as vizinhanças de u e v em $N(x)$ se sobrepõem. Neste caso, sejam a, b, c vértices em $N(x)$ tais que $au \in E(G)$, $av \notin E(G)$, $bu, bv \in E(G)$,

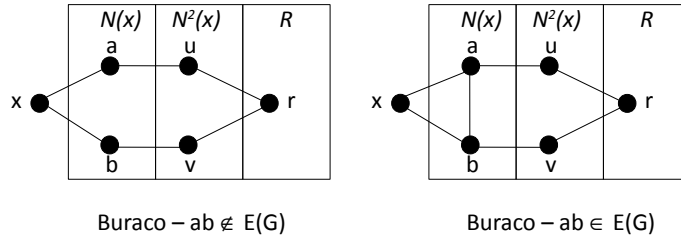
$cu \notin E(G)$ e $cv \in E(G)$. Então x, a, b, c, u, v induzem um dominó em G (Figura 4.1(e)), que é uma contradição. Isto prova a Proposição 4.1 (ii). ■



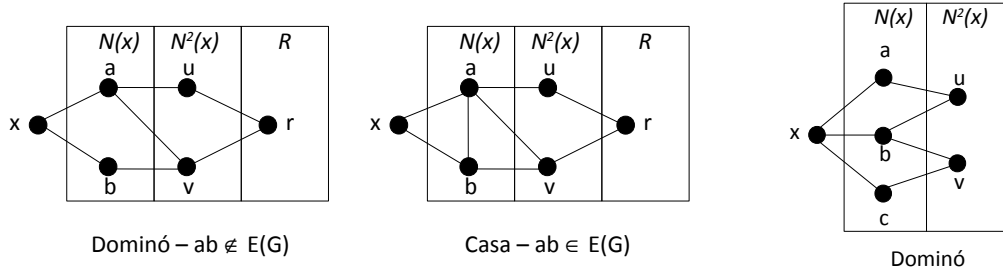
(a) Vizinhanças de u e v em $N(x)$ são incomparáveis e $uv \in E(G)$.



(b) Vizinhanças de u e v em $N(x)$ contém propriamente uma a outra e $uv \in E(G)$.



(c) Vizinhanças de u e v em $N(x)$ são incomparáveis e $uv \notin E(G)$.



(d) Vizinhanças de u e v em $N(x)$ contém propriamente uma a outra e $uv \notin E(G)$.

(e) Vizinhanças de u e v em $N(x)$ se sobrepõem.

Figura 4.1: Casos da Proposição 4.1

4.2 Caracterização dos grafos de distância hereditária que admitem cct

O próximo teorema caracteriza os grafos de distância hereditária que admitem transversal-clique de ciclos através de subgrafos proibidos. Sejam G_1, \dots, G_{12} os grafos da Figura 4.2.

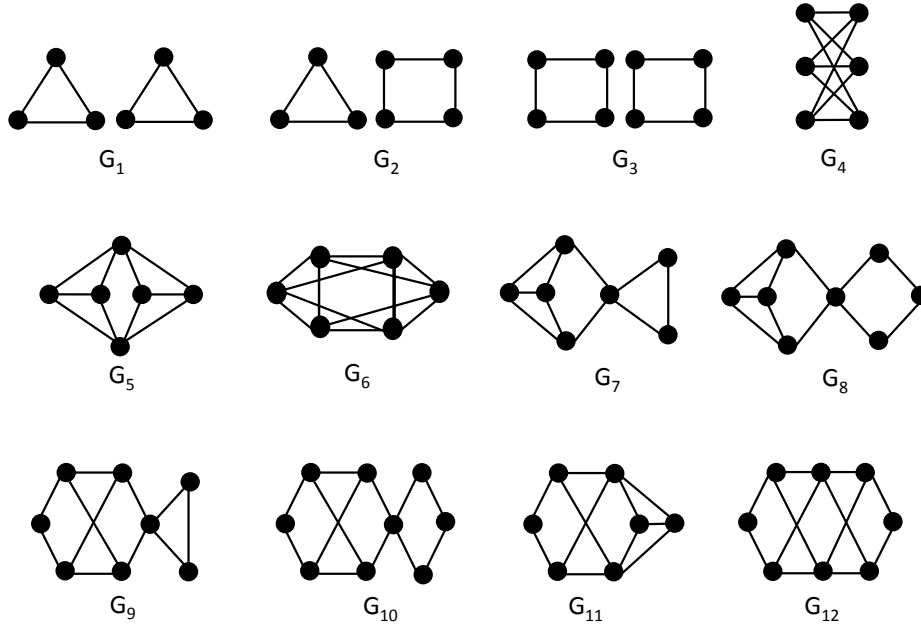


Figura 4.2: Subgrafos proibidos para grafos de distância hereditária com cct.

Teorema 4.2 [6] *Seja G um grafo de distância hereditária. Então G admite um transversal-clique de ciclos se e somente se G é livre de (G_1, \dots, G_{12}) .*

Prova. É fácil verificar, através da análise da Figura 4.2, que os grafos G_1, \dots, G_{12} não admitem cct. Além disso, também é fácil ver que tais subgrafos são minimais, uma vez que a retirada de qualquer vértice faz que estes admitam cct. Para a recíproca do teorema, seja G' um grafo de distância hereditária livre de (G_1, \dots, G_{12}) . Pelo Teorema 4.1, G' pode ser construído a partir de um grafo de distância hereditária G , por aplicar uma das três operações do Teorema 4.1 (ii). Como adicionar um vértice pendente y a um vértice x em G não cria ciclos com y , pode-se restringir a prova aos dois casos seguintes: G' resulta de G por ou adicionar um gêmeo verdadeiro ou um gêmeo falso y ao vértice x em G , e em ambos os casos, nós temos que mostrar que G' tem um cct.

Para isso, assumimos, por indução, que G tem um cct Q . Além disso, como o conjunto de vértices $V(G)$ pode ser particionado em $\{x\} \cup N(x) \cup N^2(x) \cup R$, definimos os conjuntos $Q_1 = Q \cap N(x)$, $Q_2 = Q \cap N^2(x)$ e $N_1(x) = N(x) \setminus Q_1$, necessários para a realização das provas dos casos 1 e 2 (Veja Figura 4.3).

	$N(x)$	$N^2(x)$	R	
y	Q_1	Q_2		Q
x	$N_1(x)$			

Figura 4.3: Conjuntos Q_2 , Q_1 e $N_1(x)$.

4.2.1 Caso 1: y é um gêmeo verdadeiro de x .

Seja G' o grafo resultante de G ao adicionar um gêmeo verdadeiro y a x em G .

Neste caso, os ciclos possíveis com y em G' são triângulos xya para $a \in N(x)$, triângulos yab para $a, b \in N(x)$, $ab \in E(G)$, e C_4 's $yabc$ para $a, b \in N(x)$, $ab \notin E(G)$, $c \in N^2(x)$ (Veja Figura 4.4). Se $x \in Q$ ou, de forma mais geral, $Q \subseteq N[x]$, então $Q \cup \{y\}$ é um cct de G' . Logo, temos que considerar o caso em que $x \notin Q$. Uma vez que o triângulo xab é coberto por Q , o triângulo yab também é coberto por Q , e de forma similar para o C_4 $yabc$, onde $xabc$ também é um C_4 em G coberto por Q . Logo, temos que lidar apenas com triângulos xya .

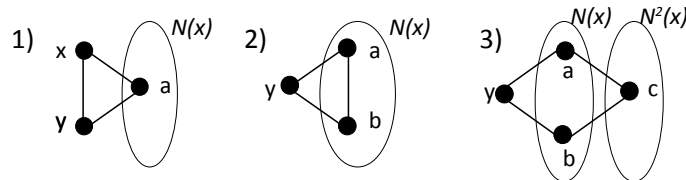


Figura 4.4: Caso 1 - Possíveis ciclos com y em G' .

Para verificar que é possível encontrar um cct em G' quando $x \notin Q$, iniciamos provando

que, neste caso, $G[N(x)]$ é um grafo *split* através da Afirmação 4.1.

Afirmação 4.1 *Se $x \notin Q$ então $G[N(x)]$ é um grafo split com partição $(N_1(x), Q_1)$.*

Prova da Afirmação 4.1. Para cada aresta $ab \in G[N(x)]$, xab é um triângulo. Portanto, a ou b está em Q e $N_1(x)$ é um conjunto independente. Como Q_1 é uma clique, conclui-se a prova da afirmação. ■

Além disso, para provar que existe um cct em G' que cobre os triângulos xya que foram criados ao adicionar um gêmeo verdadeiro a x , precisamos analisar os ciclos existentes em G a fim de construir o cct de G' a partir de Q .

Como G é livre de (G_1, G_2, G_3) , R induz um subgrafo livre de ciclos em G . Desta forma, a análise deste caso deve ser dividida em dois subcasos: o caso em que $G[N^2(x) \cup R]$ é livre de ciclos e o caso em que $G[N^2(x) \cup R]$ contém ciclo.

Caso 1.1: $G[N^2(x) \cup R]$ é livre de ciclos.

Afirmação 4.2 *Se y é um gêmeo verdadeiro de x , $x \notin Q$ e $G[N^2(x) \cup R]$ é livre de ciclos então G' admite um cct.*

Prova da Afirmação 4.2. Como Q é uma clique e $G[N^2(x) \cup R]$ é livre de ciclos, Q_2 contém no máximo dois vértices, caso contrário, teríamos um ciclo em $N^2(x)$. Se $Q_2 = \emptyset$ então $Q \cup \{y\}$ é um cct de G' . Por outro lado, se um vértice $u \in Q_2$ não possui vizinhos em $N_1(x)$ então todo ciclo contendo u também contém um vértice de Q_1 , i.e., $Q \setminus \{u\}$ ainda é um cct de G . Logo, assumimos, sem perda de generalidade, que todo vértice em Q_2 tem um vizinho em $N_1(x)$.

Se $Q_2 = \{u, v\}$, a vizinhança de Q_2 em $N_1(x)$ não pode conter dois vértices a e b , caso contrário, pela Afirmação 4.1 e Proposição 4.1, os vértices x, y, a, b, u, v induzem novamente um G_5 , mostrado na Figura 4.5. Portanto, u e v tem precisamente um vizinho $a \in N_1(x)$ que deve ser adjacente a todos os vértice de Q_1 , caso contrário, se a não for adjacente a um vértice $b \in Q_1$ então os vértices x, y, a, b, u, v induzem G_5 . Logo, $(Q \setminus \{u, v\} \cup \{a, y\})$ é um cct de G' .

Se $Q_2 = \{u\}$, consideram-se dois subcasos:

(i) O vértice u tem um vizinho $a \in N_1(x)$ que não é adjacente a algum vértice $b \in Q_1$. Afirmamos que todo ciclo C em G contendo a e u mas que não contém vértices de Q_1

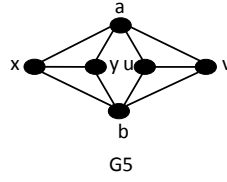


Figura 4.5: Vizinhança de Q_2 em $N_1(x)$ contendo dois vértices a e b .

deve conter também x . Isto porque se C não contém x então, pela Proposição 4.1, C é um triângulo auv com $v \in N^2(x)$ ou um C_4 $aucv$ com $c \in N_1(x)$, $v \in N^2(x)$ ou um C_4 com a, u , um vértice $v \in N^2(x)$ e um vértice $w \in R$. No primeiro caso, pela Proposição 4.1, os vértices x, y, a, b, u, v induzem G_5 . O segundo caso não pode ocorrer uma vez que a existência do ciclo $aucv$ implica na existência do ciclo $axcv$ em G , não coberto por Q . No terceiro caso, pela Proposição 4.1, $bv \in E$ e agora, os vértices x, y, a, b, u, v, w induzem G_{11} , conforme figura 4.6, uma contradição.

Isto implica que $(Q \setminus \{u\}) \cup \{x, y\}$ é um cct de G' .

(ii) Todo vizinho $a \in N_1(x)$ de u é adjacente a todos os vértices de Q_1 . Então $Q \cup \{a\}$ é um cct de G para algum $a \in N_1(x)$ e, como pela Afirmação 4.1 $N_1(x)$ é um conjunto independente, todos os outros vizinhos $a' \in N_1(x)$ de u não são adjacentes a vértices de $Q_1 \cup \{a\}$. Através de um argumento análogo ao aplicado em (i), concluímos que todo ciclo C em G contendo a', u mas que não contém vértices de $Q_1 \cup \{a\}$ deve também conter x .

Conclui-se que $(Q \setminus \{u\}) \cup \{x, y, a\}$ é um cct de G' . Isto completa a prova da Afirmação 4.2.

■

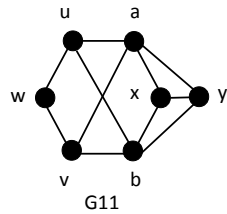


Figura 4.6: Vizinhança de Q_2 em $N_1(x)$ contendo um vértice a , existência de C_4 $aucv$ e $bv \in E$.

Caso 1.2: $G[N^2(x) \cup R]$ contém ciclo.

No que segue, suponha que $G[N^2(x) \cup R]$ contém um ciclo C . Pode-se verificar que este ciclo é de um dos tipos abaixo (Veja Figura 4.7):

- (A₁) C tem exatamente um vértice u em $N^2(x)$, e C é um C_4 .
- (A₂) C tem exatamente um vértice u em $N^2(x)$, e C é um C_3 .
- (B₁) C tem exatamente dois vértices u, v em $N^2(x)$, $uv \in E(G)$, e C é um C_4 .
- (B₂) C tem exatamente dois vértices u, v em $N^2(x)$, $uv \in E(G)$, e C é um C_3 .
- (B₃) C tem exatamente dois vértices u, v em $N^2(x)$, $uv \notin E(G)$, e C é um C_4 .
- (C₁) C é um C_4 com exatamente três vértices u, v, w em $N^2(x)$ (que formam um P_3 em $N^2(x)$).
- (D₁) C é um C_3 em $N^2(x)$.
- (D₂) C é um C_4 em $N^2(x)$.

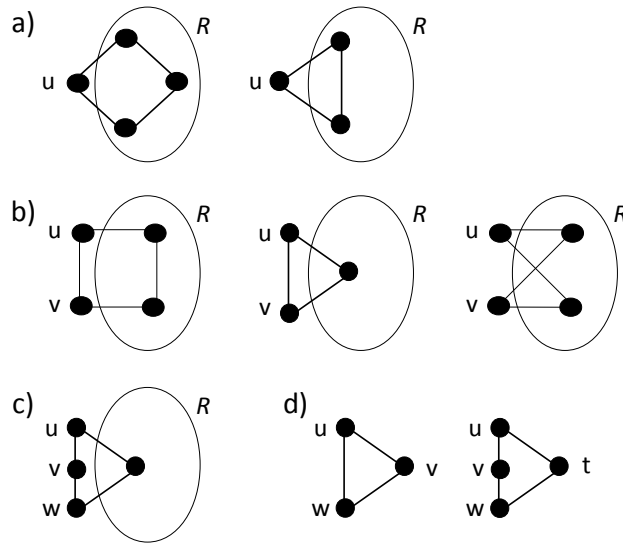


Figura 4.7: Ciclos em $G[N^2(x) \cup R]$.

Inicialmente, provamos que $N(x)$ é uma clique, através da Afirmação 4.3 abaixo.

Afirmação 4.3 *Se y é um gêmeo verdadeiro de x e existe ciclo em $G[N^2(x) \cup R]$ então $N(x)$ é uma clique.*

Prova da Afirmação 4.3. Suponha por contradição que existam os vértices $a, b \in N(x)$ e $ab \notin E(G)$. Como G' é livre de (G_1, G_2) , a e b devem ver todo ciclo em $G[N^2(x) \cup R]$. Se

C é um ciclo do tipo (A_1) ou (A_2) em $G[N^2(x) \cup R]$, i.e., com exatamente um vértice u em $N^2(x)$ então a e b são adjacentes a u e obtemos G_7 ou G_8 (Figura 4.8) - uma contradição. Se C é do tipo (B_3) , $uv \notin E(G)$, da mesma maneira, a e b tem que ver C , e se a e b não são adjacentes tanto a u quanto a v então existe ou um buraco ou um dominó ou G_8 , como verificado detalhadamente através da Figura 4.9. Logo, a e b são adjacentes a ambos u e v , i.e., G' contém como subgrafo induzido G_{11} (Veja Figura 4.10) - uma contradição.

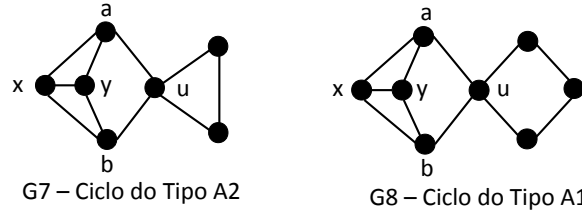


Figura 4.8: Vértices $a, b \in N(x)$, $ab \notin E(G)$ e ciclos do tipo (A_1) e (A_2) .

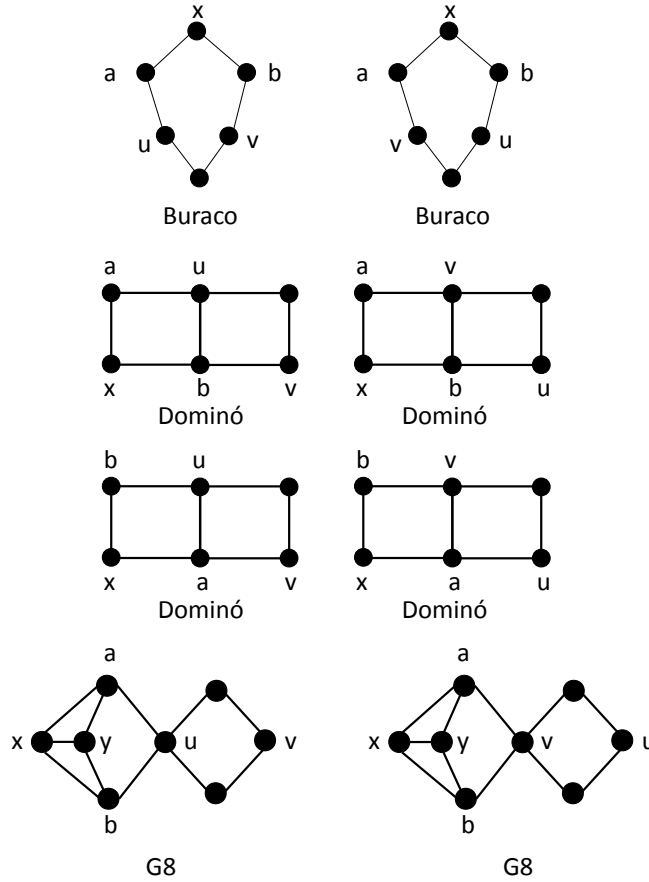


Figura 4.9: Vértices $a, b \in N(x)$, $ab \notin E(G)$ e ciclo do tipo (B_3) .

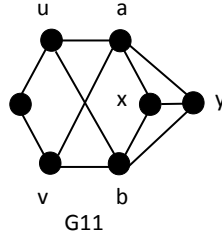


Figura 4.10: Vértices $a, b \in N(x)$, $ab \notin E(G)$, ciclo do tipo (B_3) e a e b adjacentes a u e v .

Analizamos os casos restantes considerando a seguinte situação: Se a é adjacente ao vértice u e b é adjacente ao vértice $v \neq u$ em $N^2(x)$ tal que existe um caminho entre u e v em $N^2(x)$ então, pela Proposição 4.1, a e b veem uma aresta comum $u'v'$; mas então G' contém G_5 com x, y, a, b, u', v' - uma contradição. Isto conclui a Afirmação 4.3. ■

Continuando a prova deste caso, se $a \in N(x)$ então existe um triângulo xya e a deve ver todo ciclo em $G[N^2(x) \cup R]$ uma vez que G é livre de (G_1, G_2, G_3) . Ainda mais, provamos na Afirmação 4.4 que a é adjacente a todo vértice em Q_2 .

Afirmação 4.4 *Se y é um gêmeo verdadeiro de x e existe ciclo em $G[N^2(x) \cup R]$ então todo vértice em $N(x)$ é adjacente a todo vértice em Q_2 .*

Prova da Afirmação 4.4. Como G' é livre de (G_1, G_2, G_3) , qualquer vértice $a \in N(x)$ é adjacente a pelo menos um vértice u em cada ciclo de $G[N^2(x) \cup R]$. Sabe-se, também, pela Proposição 4.1, que todos os vértices de Q_2 tem a mesma vizinhança em $N(x)$, e o vértice u é adjacente a a . Logo, todos os vértices em Q_2 são adjacentes a todos os vértices em $N(x)$, que conclui a Afirmação 4.4. ■

Desta maneira, como pela Afirmação 4.3 que $N(x)$ é uma clique, conclui-se que se há um ciclo em $G[N^2(x) \cup R]$ e $x \notin Q$ então $N(x) \cup Q_2$ é um cct de G' , o que finaliza a prova do Caso 1. ■

4.2.2 Caso 2: y é um gêmeo falso de x .

A partir de agora, seja G' o grafo resultante de G ao adicionar um gêmeo falso y a x em G . Novamente, por indução, supomos que G tem um cct Q . Os ciclos possíveis com y em

G' são triângulos yab para $a, b \in N(x)$, $ab \in E(G)$, C_4 's $yabc$ para $a, b \in N(x)$, $ab \notin E(G)$, $c \in N^2(x)$, e C_4 's $xyab$ para $a, b \in N(x)$ (Veja Figura 4.11).

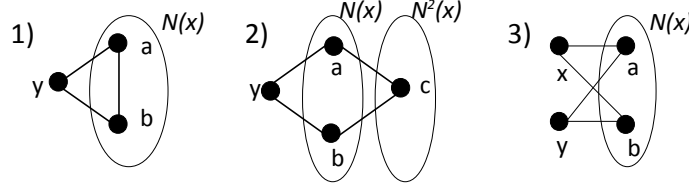


Figura 4.11: Caso 2 - Possíveis ciclos com y em G' .

Se $|N(x)| = 1$ então Q é também um cct de G' pois a inclusão de y em G não cria novos ciclos em G' . Por isso, assumimos que $|N(x)| \geq 2$.

Em primeiro lugar, provamos através da Afirmação 4.5 que $G'[N(x)]$ é um grafo threshold. Este fato é essencial para a prova deste caso, além de fortalecer a Afirmação 4.1.

Afirmação 4.5 $G'[N(x)]$ é um grafo threshold.

Prova da Afirmação 4.5. Como G' é um grafo de distância hereditária, $N(x)$ é livre de P_4 , e uma vez que G' é livre de G_5 e G_6 , $N(x)$ é livre de $2K_2$ e C_4 , i.e., $G'[N(x)]$ é um grafo threshold, o que conclui a Afirmação 4.5. ■

Recorde que $V(G)$ pode ser particionado em $\{x\} \cup N(x) \cup N^2(x) \cup R$ e, como G' é livre de (G_1, G_2, G_3) , R induz um subgrafo livre de ciclos em G' . Sendo assim, igualmente ao caso anterior, dividimos a análise em dois subcasos: $G[N^2(x) \cup R]$ é livre de ciclos e $G[N^2(x) \cup R]$ contém ciclo.

Caso 2.1: $G[N^2(x) \cup R]$ é livre de ciclos.

Vamos mostrar que também para este caso, G' tem um cct.

Sabe-se que G tem um cct Q e, como antes, definimos $Q_1 = Q \cap N(x)$, $Q_2 = Q \cap N^2(x)$ e $N_1(x) = N(x) \setminus Q_1$. Pela suposição do caso 2.1, Q_2 pode conter no máximo dois vértices. Além disso, como na prova da Afirmação 4.2, podemos assumir que todo vértice em Q_2 tem um vizinho em $N_1(x)$ e, se $Q_2 \neq \emptyset$ então $x \notin Q$.

Caso 2.1.1: $|Q_2| = 2$.

Se $Q_2 = \{u, v\}$, dividimos a prova em três subcasos. Relembre que pela Proposição 4.1, u e v tem a mesma vizinhança em $N_1(x)$.

(i) Se u, v possuem três vizinhos a, b, c no conjunto independente $N_1(x)$ então, como mostra a Figura 4.12, os vértices x, y, u, a, b, c (vértices x, y, v, a, b, c , respectivamente) induzem G_4 em G' , que é impossível.

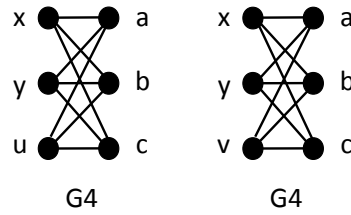


Figura 4.12: Vizinhança de Q_2 em $N_1(x)$ contendo três vértices a, b, c .

(ii) Se u, v possuem exatamente dois vizinhos a, b em $N_1(x)$ então não existe outro vértice c , $c \in N_1(x)$, caso contrário x, y, a, b, c, u, v induzem G_{11} em G' (Veja Figura 4.13). Além disso, como G' é livre de G_4 e pela Afirmação 4.5, $G'[N(x)]$ é um grafo threshold, ou a ou b é adjacente a todos os vértices de Q_1 , caso contrário, se a não é adjacente a a' e b não é adjacente a b' em Q_1 , então, conforme Figura 4.14, ou G' contém G_4 (se $a' = b'$) ou existe um $P_4 ab'a'b$ em $N(x)$. Suponha que a é adjacente a todos os vértices de Q_1 então todo ciclo em G' contendo y também contém algum vértice em $Q \cup \{a\}$. Desta forma temos que $Q \cup \{a\}$ é um cct de G' .

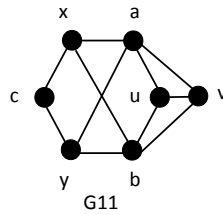


Figura 4.13: Vizinhança de Q_2 em $N_1(x)$ contendo dois vértices a, b e outro vértice $c \in N_1(x)$.

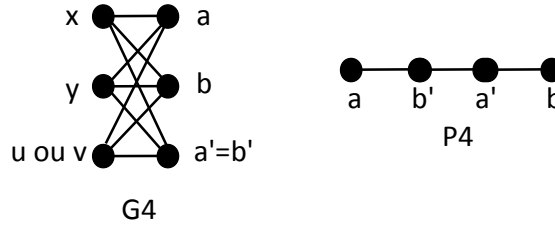


Figura 4.14: Vizinhança de Q_2 em $N_1(x)$ contendo dois vértices a, b com a não adjacente a a' e b a b' em Q_1 .

(iii) Se u, v tem exatamente um vizinho a em $N_1(x)$, vamos analisar a vizinhança de a :

Se a não é adjacente a algum vértice $b \in Q_1$ então não existe outro vértice $c \in N_1(x)$ senão, como mostra a Figura 4.15, os vértices x, y, a, b, c, u, v induzem G_{11} , se $bc \notin E(G)$, ou c, b, x, u, a induzem casa, se $bc \in E(G)$ e, portanto, todo ciclo em G' contendo y também contém algum vértice em Q , i.e., Q é ainda um cct em G' .

Se a é adjacente a todos os vértices em Q_1 , todo ciclo contendo u ou v também contém algum vértice em $Q_1 \cup \{a\}$ e, portanto, $(Q \setminus \{u, v\}) \cup \{a, y\}$ é um cct de G' . Neste caso, y também é necessário no cct devido aos C_4 's formados com x, y e dois vértices de $N_1(x)$.

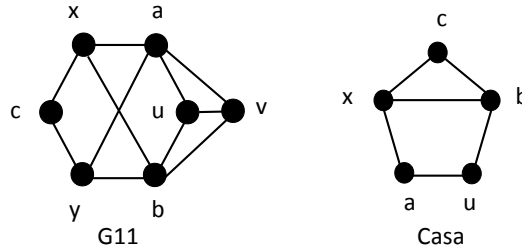


Figura 4.15: Vizinhança de Q_2 em $N_1(x)$ contendo um vértice a .

Caso 2.1.2: $|Q_2| = 1$.

Seja $Q_2 = \{u\}$. Como G' é livre de G_4 , u possui no máximo dois vizinhos em $N_1(x)$. Se u tem dois vizinhos a, b em $N_1(x)$ então, como $G'[N(x)]$ é um grafo threshold (Afirmação 4.5) e G' é livre de G_4 , um deles, que chamaremos de a , deve ser adjacente a todos os vértices de Q_1 , e isto significa que todo ciclo contendo u também contém algum vértice em $Q_1 \cup \{a\}$ (recorde que $G[N^2(x) \cup R]$ livre de ciclos), i.e., $(Q \setminus \{u\}) \cup \{a, y\}$ é um cct de G' .

Se u tem precisamente um vizinho a em $N_1(x)$ e a é adjacente a todos os vértices em Q_1 , novamente $(Q \setminus \{u\}) \cup \{a, y\}$ é um cct de G' ; caso contrário, a não é adjacente a algum vértice b em Q_1 , e a análise segue considerando os dois casos seguintes:

(i') Se $N_1(x) = \{a\}$ então todo ciclo em G' contendo y também contém um vértice de Q_1 , e portanto Q é um cct de G' .

(ii') Se $N_1(x)$ contém um vértice $c \neq a$, temos que $bc \notin E(G)$ (caso contrário, verificamos na Figura 4.16 que os vértices x, a, b, c, u induzem casa). Desta maneira, vamos mostrar que não há ciclo C em G contendo a e u sem vértices de Q_1 , pois, se existir tal ciclo C então pela Proposição 4.1, ele deve ser um triângulo auv com $v \in N^2(x)$, e então os vértices x, y, a, b, c, u, v induzem o grafo G_{11} , ou é um C_4 $auvw$ com $u, v \in N^2(x)$ e $w \in R$ mas então existe G_{12} ou dominó em G' (Veja Figura 4.18). Assim, conclui-se que $(Q \setminus \{u\}) \cup \{y\}$ is a cct of G' .

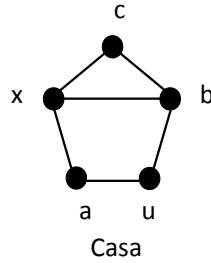


Figura 4.16: Vizinhança de Q_2 em $N_1(x)$ contendo um vértice a e $bc \in E(G)$.

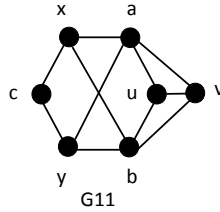


Figura 4.17: Vizinhança de Q_2 em $N_1(x)$ contendo um vértice a e triângulo auv com $v \in N^2(x)$.

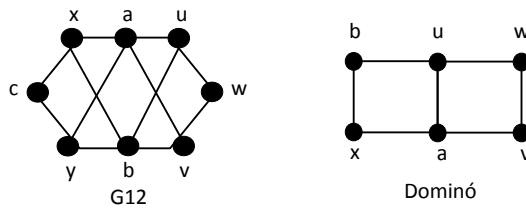


Figura 4.18: Vizinhança de Q_2 em $N_1(x)$ contendo um vértice a e C_4 $auvw$ com $u, v \in N^2(x)$ e $w \in R$.

Caso 2.1.3: $Q_2 = \emptyset$.

Neste caso, $Q \subseteq N[x]$. Se existir um cct Q of G com $x \notin Q$ então $Q \cup \{y\}$ é um cct de G' e este caso está finalizado. Portanto, temos que mostrar que no Caso 2.1.3, G tem um cct Q com $Q \subseteq N(x)$.

Em G , existem dois tipos de ciclos contendo x : Triângulos xab com $a, b \in N(x)$ e C_4 's abc com $a, b \in N(x)$ e $c \in N^2(x)$. Sabemos que, pela Afirmação 4.5, $N(x)$ induz um grafo threshold em G' , e em particular, este é particionado em uma clique Q_1 e um conjunto independente $N_1(x)$. Então Q_1 (e em geral, toda clique maximal em $N(x)$) cobre todo triângulo xab uma vez que $ab \in E$.

O caso em que existe um C_4 abc com $a, b \in N(x)$ e $c \in N^2(x)$ em G é mais complexo. Assume-se que existe tal C_4 e este não está coberto por Q_1 . Se o vértice a (vértice b , respectivamente) for adjacente a todos os vértices em Q_1 então a clique $Q_1 \cup \{a\}$ ($Q_1 \cup \{b\}$, respectivamente) cobrirá abc também. Caso contrário, ambos os vértices a e b não são adjacentes a vértices de Q_1 . Uma vez que pela Afirmação 4.5, $N(x)$ é livre de P_4 , a e b tem um não vizinho comum, d , em Q_1 . Segue que $cd \notin E$ (senão, conforme Figura 4.19, x, y, c, a, b, d induz G_4). Se a e b não são adjacentes a outro vértice $d' \in Q_1$ então x, y, d, d', a, b, c induz G_{11} (Figura 4.20), uma contradição. Logo, um dos vértices a e b , nomearemos a , tem no máximo um não vizinho, que nomearemos d , em Q_1 e, como $N(x)$ é um grafo threshold, sem perda de generalidade, a vizinhança de b em Q_1 está contida na vizinhança de a em Q_1 ; e como vimos anteriormente, b não é adjacente a d e, da mesma forma, c não é adjacente a d .

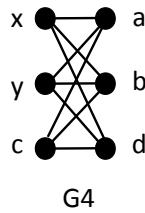


Figura 4.19: C_4 abc com $a, b \in N(x)$ e $c \in N^2(x)$ e um vértice não adjacente d em Q_1 .

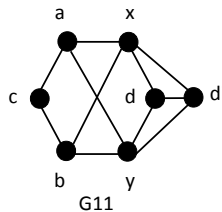


Figura 4.20: C_4 abc com $a, b \in N(x)$ e dois vértices não adjacentes d, d' em Q_1 .

Com a finalidade de mostrar que $Q_1 \setminus \{d\} \cup \{a\}$ é um cct de G , a Afirmação 4.6 analisa os vizinhos de d em $N^2(x)$.

Afirmação 4.6 *Sejam a, b e c vértices do C_4 abc não coberto por Q_1 com $a, b \in N(x)$ e $c \in N^2(x)$. Seja d um vértice de Q_1 não vizinho de a, b e c . Seja e um vizinho de d em $N^2(x)$. Então e não é adjacente aos vértices a, b e c .*

Prova da Afirmação 4.6. Iniciamos por observar que se $ce \in E$, pela Proposição 4.1 (i), c e e tem a mesma vizinhança em $N(x)$, o que é impossível, uma vez que e é adjacente a d e c não é adjacente a d . Portanto, $ce \notin E$. Neste caso, pela Proposição 4.1 (ii), se e for adjacente a um dos vértices a e b , ele deve ser adjacente a ambos. Desta maneira, verificamos na Figura 4.21 que os vértices x, y, e, a, b, d induzem G_4 - uma contradição. Logo, e também não deve ser adjacente a ambos os vértices a e b . ■

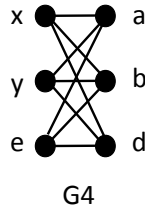


Figura 4.21: Vértice e em $N^2(x)$ adjacente aos vértices a e b em $N(x)$.

Agora, suponha que $Q'_1 := (Q_1 \setminus \{d\}) \cup \{a\}$ não é um cct de G . Note que $N'_1(x) = N(x) \setminus Q'_1$ é independente. Então, existe um ciclo em G cujo único vértice de Q_1 é d . Obviamente, se C for um ciclo contendo d e uma aresta em $N(x)$ então Q'_1 cobre C uma vez que $N'_1(x)$ é independente. Logo, temos que considerar ciclos sem arestas em $N(x)$.

Inicialmente, considere um C_3 duv com $u, v \in N^2(x)$. Então pela Afirmação 4.6, u e v não são adjacentes a a, b e c , e agora, junto com y , G' contém G_9 (Figura 4.22), uma contradição. Se d estiver em um C_4 $duvw$ com $u, v \in N^2(x)$ e $w \in R$ então, muito similarmente, junto com y , G' contém G_{10} (Figura 4.23), uma contradição. Logo, d não está contido em tais ciclos.

Se C for um C_4 com $d, z \in N(x)$ e $u, v \in N^2(x)$ então, novamente pela Afirmação 4.6, u e v não são adjacentes a a, b e c . Então z deve ser adjacente a a e b , caso contrário existe casa ou G_{12} em G' , junto com y , mas, então os vértices $xadz$ induzem uma casa, que é uma contradição. Veja na Figura 4.24 tais contradições.

Da mesma forma, isto também ocorre quando d estiver em um C_4 $xdqu$, com $q \in N'_1(x)$ e $u \in N^2(x)$.

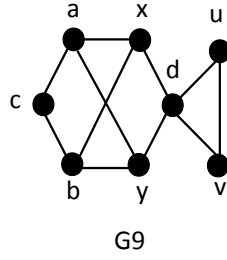


Figura 4.22: Vértice d em um C_3 duv com $u, v \in N^2(x)$.

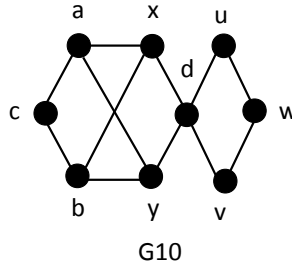


Figura 4.23: Vértice d em um C_4 $duvw$ com $u, v \in N^2(x)$ e $w \in R$.

Esta contradição final mostra que, no Caso 2.1.3, existe um cct Q de G sem x , e logo, há um cct $Q \cup \{y\}$ em G' . ■

Caso 2.2: $G[N^2(x) \cup R]$ contém ciclo.

Assim como no caso 1.2, vamos assumir que $G[N^2(x) \cup R]$ contém um ciclo C . Consequentemente, temos que $x \notin Q$.

Iniciamos este caso provando, através da Afirmação abaixo, que $N(x)$ é livre de I_3 .

Afirmação 4.7 *Se y é um gêmeo falso de x e $G[N^2(x) \cup R]$ contém ciclo então $N(x)$ é livre de I_3 .*

Prova da Afirmação 4.7. Assuma que $G[N^2(x) \cup R]$ contém ciclo e $N(x)$ contém um conjunto independente de três vértices a_1, a_2, a_3 . Seja S um conjunto independente maximal em $N(x)$ contendo a_1, a_2, a_3 . Pela Proposição 4.1, para vértices $u, v \in N^2(x)$ na mesma componente conexa de $G[N^2(x) \cup R]$, suas vizinhanças em S são iguais. Além disso, nenhum vértice $u \in N^2(x)$ é adjacente a três vértices em S , caso contrário, como mostra a Figura 4.25, G_4 está contido em G' . Logo, para todo par de vértices $u, v \in N^2(x)$ na mesma componente conexa, $N(u) \cap S = N(v) \cap S \subseteq \{a_1, a_2\}$.

Se $u \in N^2(x)$ estiver em um ciclo do tipo (A_1) ou (A_2) então x, y, a_1, a_2, a_3, u e os vértices restantes do ciclo induzem G_2, G_3, G_9 ou G_{10} (Veja detalhadamente na Figura 4.26). Se o

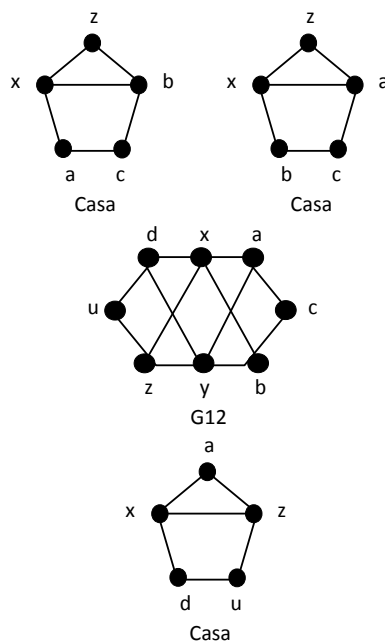


Figura 4.24: Vértice d em um C_4 com $d, z \in N(x)$ e $u, v \in N^2(x)$.

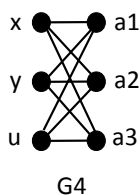


Figura 4.25: Vértice u adjacente a três vértices a_1, a_2, a_3 em S .

ciclo com $u, v \in N^2(x)$ for do tipo (B_1) , G' contém casa (Figura 4.27); se for do tipo (B_2) ou (D_1) , existe G_2 ou G_{11} em G' (Figura 4.28); se for do tipo (B_3) , G' contém G_3 ou G_{12} (Figura 4.29); e finalmente, se for do tipo (C_1) ou (D_2) , G' contém G_3 ou G_{11} (Figura 4.30). Isto finaliza a Afirmação 4.7. ■

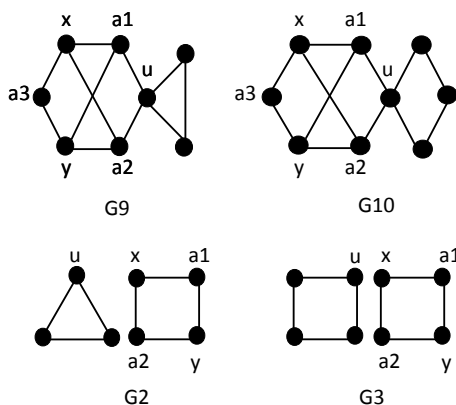
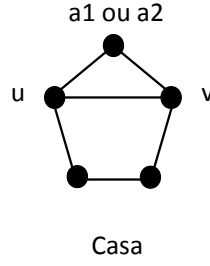
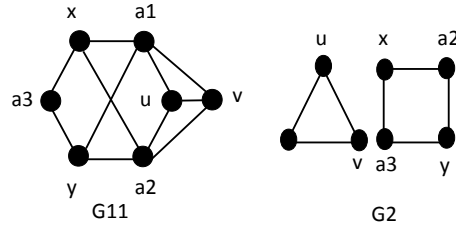
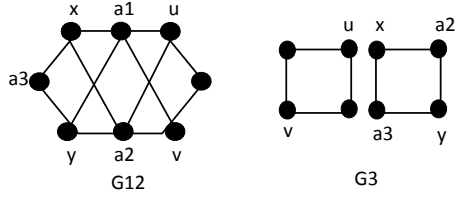
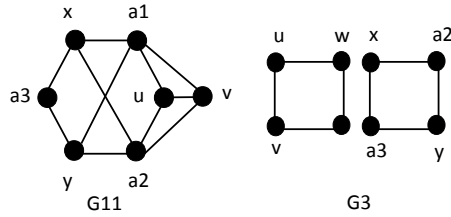


Figura 4.26: Vértice $u \in N^2(x)$ e ciclo do tipo (A_1) ou (A_2) .

Figura 4.27: Vértice $u \in N^2(x)$ e ciclo do tipo (B_1) .Figura 4.28: Vértice $u \in N^2(x)$ e ciclo do tipo (B_2) ou (D_1) .Figura 4.29: Vértice $u \in N^2(x)$ e ciclo do tipo (B_3) .Figura 4.30: Vértice $u \in N^2(x)$ e ciclo do tipo (C_1) ou (D_2) .

Concluimos que se y é um gêmeo falso de x e $G[N^2(x) \cup R]$ contém ciclo então pela Afirmação 4.7, $N_1(x)$ contém no máximo dois vértices. Se $|N_1(x)| \leq 1$ então Q é um cct de G' . Se $N_1(x) = \{a, b\}$ então, pela Afirmação 4.5, um deles, que chamaremos de a , é adjacente a todos os vértices de Q_1 . Além disso, pela Proposição 4.1, ou $Q_2 \cup \{a\}$ é uma clique e então, $Q \cup \{a\}$ é um cct de G' , ou a não é adjacente a vértices de Q_2 e então, seja C um ciclo em $G[N^2(x) \cup R]$ e seja $u \in V(C) \cap Q_2$. Como $xyab$ é um C_4 e G é livre de

(G_2, G_3) , existe uma aresta ligando $xyab$ e C . Também pela Proposição 4.1, concluímos que b é adjacente a todos os vértices de $C \cap N^2(x)$ e, portanto, em Q_2 . Agora, se existir algum vértice $a' \in Q_1$ então x, a, a', b, u , conforme Figura 4.31, induz ou uma casa ou um leque - uma contradição. Portanto $Q_1 = \emptyset$ e $Q_2 \cup \{b\}$ é um cct de G' . Isto finaliza a prova do Caso 2 e logo, também a prova do Teorema 4.2. ■

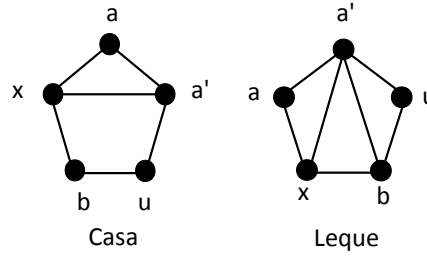


Figura 4.31: Vértice $a' \in Q_1$.

Como uma consequência direta do Teorema 4.2, obtemos uma outra prova do resultado em [5]:

Corolário 4.1 *Se G é um cografo então G admite um cct se e somente se G é livre de (G_1, \dots, G_6) .*

Prova. Analisando os grafos G_1 a G_6 , é fácil verificar que os mesmos não admitem cct. Por outro lado, sabemos que os cografos formam uma subclasse dos grafos de distância hereditária. Logo, os subgrafos proibidos dos cografos estão contidos nos subgrafos proibidos dos grafos de distância hereditária. Além disso, G é livre de (G_7, \dots, G_{12}) porque todos eles contém P_4 . Desta forma, pelo Teorema 4.2, o corolário segue. ■

Como verificamos, o Corolário 4.1 fornece uma caracterização dos cografos que admitem cct por subgrafos proibidos. Já o Corolário 4.2, abaixo, caracteriza os grafos de distância hereditária que são grafos-(2,1) também através de subgrafos induzidos proibidos.

Corolário 4.2 *Seja G um grafo de distância hereditária. Então G é um grafo (2,1) se e somente se G é livre de (G_1, G_5, G_6, G_7) .*

Prova. Os grafos G_1, G_5, G_6 e G_7 não são grafos-(2,1). Por outro lado, assuma que G é livre de (G_1, G_5, G_6, G_7) e G não é um grafo-(2,1). Seja G' um subgrafo induzido minimal

de G que não é um grafo-(2,1). Note que ser um grafo-(2,1) é equivalente a admitir uma clique que intercepta todo ciclo ímpar. Logo, G' não admite um cct. Porém, como consequência do resultado apresentado em [34], para um grafo de distância hereditária, ser um grafo minimal que não é (2,1) implica em ser um grafo minimal que não admite cct. Desta maneira, pelo Teorema 4.2, G' é isomorfo a um dos grafos G_1, G_2, \dots, G_{12} . Como $G_2, G_3, G_4, G_8, G_9, G_{10}, G_{11}$, e G_{12} são grafos-(2,1), segue que G contém G_1, G_5, G_6 , ou G_7 como subgrafo induzido. ■

Por último, vamos mostrar que verificar um cct em um grafo de distância hereditária pode ser feito em tempo linear. Para isso, apresentamos um importante resultado de Courcelle *et al.* [14] através do Teorema 4.3.

Teorema 4.3 [14] *Em grafos que possuem clique-width de no máximo k , onde k é fixo, todo problema de decisão e otimização que pode ser representado em lógica de segunda ordem monádica possui um algoritmo linear.*

Observe que o problema de verificar que um grafo de distância hereditária G admite um cct pode ser representado pela seguinte fórmula MSOL:

$$\begin{aligned} & \exists T(\\ & \forall x \forall y \forall z (A(x, y) \wedge A(x, z) \wedge A(y, z) \Rightarrow T(x) \vee T(y) \vee T(z)) \\ & \wedge \\ & \forall x \forall y \forall z \forall w (A(x, y) \wedge A(y, z) \wedge A(z, w) \wedge A(w, x) \Rightarrow T(x) \vee T(y) \vee T(z) \vee T(w))) \end{aligned}$$

Mais especificamente, a fórmula acima verifica se existe um transversal de ciclos T em G . Como G é um grafo de distância hereditária, então G não contém C_n , $n \geq 5$, como subgrafo induzido. Sendo assim, pelo menos um dos vértices de cada K_3 e C_4 deve pertencer a T . Além disso, T tem a propriedade adicional de induzir uma clique, o que também pode ser representado por uma fórmula MSOL.

Desta maneira, como vimos no Teorema 2.4 que todo grafo de distância hereditária possui *clique-width* no máximo 3, segue o Teorema abaixo.

Teorema 4.4 *Verificar que um grafo de distância hereditária admite cct pode ser feito em tempo linear.*

Observe também que é possível escrever em lógica de segunda ordem monádica, ou mais especificamente, em lógica de primeira ordem, os subgrafos proibidos da Figura 4.2. Como exemplo, a fórmula abaixo representa o subgrafo G_1 .

$$\begin{aligned} & \forall u \forall v \forall w \forall x \forall y \forall z \\ & \neg (A(u, v) \wedge A(u, w) \wedge A(v, w) \wedge \\ & A(x, y) \wedge A(x, z) \wedge A(y, z) \wedge \\ & \neg A(u, x) \wedge \neg A(u, y) \wedge \neg A(u, z) \wedge \\ & \neg A(v, x) \wedge \neg A(v, y) \wedge \neg A(v, z) \wedge \\ & \neg A(w, x) \wedge \neg A(w, y) \wedge \neg A(w, z)) \end{aligned}$$

Sendo assim, a caracterização de grafos de distância hereditária apresentada nesta dissertação também produz um algoritmo linear. Isto implica no Teorema 4.5.

Teorema 4.5 *Verificar que um grafo de distância hereditária é livre de (G_1, \dots, G_{12}) pode ser feito em tempo linear.*

Assim como ilustrado na introdução, segue na Figura 4.32, o diagrama com as principais contribuições desta dissertação atualizado com os respectivos teorema e corolários apresentados.

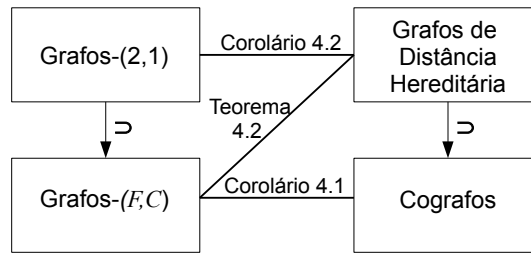


Figura 4.32: Principais classes e contribuições da dissertação

Capítulo 5

Conclusão

Nesta dissertação apresentamos um estudo detalhado sobre transversais-clique de ciclos em grafos de distância hereditária. Ou seja, consideramos o problema de verificar se um dado grafo de distância hereditária G admite uma clique que intercepte todos os seus ciclos.

Para isso, apresentamos as principais definições e caracterizações dos grafos de distância hereditária, assim como, descrevemos um algoritmo de reconhecimento e estudamos sua complexidade. Além disso, introduzimos o conceito de transversais de ciclos e definimos os grafos- (k, l) e os grafos- $(2, 1)$, bem como estudamos os grafos- $(\mathcal{F}, \mathcal{C})$ e o conceito de transversais-clique de ciclos.

Como resultado principal desta dissertação, apresentamos, uma caracterização dos grafos de distância hereditária que admitem transversal-clique de ciclos através de uma família de subgrafos proibidos, resultado este publicado em 2013 [6].

Além disso, como consequência direta do teorema principal, caracterizamos os grafos de distância hereditária que são grafos- $(2, 1)$ através de subgrafos induzidos proibidos e, obtivemos uma outra prova para o resultado apresentado em [5], caracterizando os cografos que admitem cct também por subgrafos proibidos.

Por último, mostramos que grafos de distância hereditária que admitem cct podem ser reconhecidos em tempo linear através da caracterização fornecida neste trabalho.

5.1 Trabalhos Futuros

Como continuação deste trabalho, propomos estudar os seguintes problemas:

- (1) Caracterizar e reconhecer grafos- $(\mathcal{F}, \mathcal{C})$ para outras classes de grafos, por exemplo subclasses de grafos perfeitos, tais como: grafos fortemente cordais (grafos cordais em que todos os seus ciclos de tamanho maior do que 5 têm uma corda ímpar, ou seja, uma corda entre vértices não consecutivos em um ciclo e que estão separados por uma distância ímpar); grafos perfeitamente ordenáveis (grafos que podem ser ordenados de tal maneira que um algoritmo de coloração guloso é ótimo em todo subgrafo induzido); e, até mesmo para a própria classe dos grafos perfeitos.
- (2) Encontrar a família de subgrafos proibidos para a classe dos grafos de distância hereditária- (k, l) .
- (3) Encontrar uma caracterização por subgrafos proibidos para os grafos- $(2, 1)$ e para os grafos- $(\mathcal{F}, \mathcal{C})$.

Referências

- [1] M. F. Andrielli and C. P. de Mello. Conjunto fonte máximo em grafos de comparabilidade. Relatório técnico, Instituto de Computação, Universidade de Campinas, Julho 1996.
- [2] Hans-Jürgen Bandelt and Henry Martijn Mulder. Distance-hereditary graphs. *Journal of Combinatorial Theory, Ser. B*, 41:182–208, 1986.
- [3] Andreas Brandstädt. Partitions of graphs into one or two independent sets and cliques. *Discrete Mathematics*, 152(1-3):47–54, 1996.
- [4] Andreas Brandstädt. Corrigendum. *Discrete Mathematics*, 186(1-3):295, 1998.
- [5] Andreas Brandstädt, Synara Brito, Sulamita Klein, Loana Tito Nogueira, and Fábio Protti. Cycle transversals in perfect graphs and cographs. *Theoretical Computer Science*, 469:15–23, 2013.
- [6] Andreas Brandstädt, Simone Esposito, Loana Tito Nogueira, and Fábio Protti. Clique cycle-transversals in distance-hereditary graphs. In *VII Latin-American Algorithms, Graphs and Optimization Symposium (LAGOS'13)*, Electronic Notes in Discrete Mathematics, Playa del Carmen, Mexico, a ser publicado.
- [7] Andreas Brandstädt, Van Bang Le, and Thomas Szymczak. The complexity of some problems related to graph 3-colorability. *Discrete Applied Mathematics*, 89(1-3):59–73, 1998.
- [8] Raquel S. F. Bravo, Sulamita Klein, Loana Tito Nogueira, and Fábio Protti. Characterization and recognition of p_4 -sparse graphs partitionable into k independent sets and l cliques. *Discrete Applied Mathematics*, 159(4):165–173, 2011.
- [9] Anna Bretscher, Derek G. Corneil, Michel Habib, and Christophe Paul. A simple linear time lexbfs cograph recognition algorithm. *SIAM Journal on Discrete Mathematics*, 22(4):1277–1296, 2008.
- [10] V. Chvátal and P.L. Hammer. Set-packing and threshold graphs. *Univ. Waterloo Res. Report, CORR*, 73–21, 1973.
- [11] D. G. Corneil, H. Lerchs, and L. S. Burlingham. Complement reducible graphs. *Discrete Applied Mathematics*, 3:163–174, 1981.
- [12] Derek G. Corneil, Yehoshua Perl, and Lorna K. Stewart. A linear recognition algorithm for cographs. *SIAM Journal on Computing*, 14(4):926–934, 1985.
- [13] Bruno Courcelle, Joost Engelfriet, and Grzegorz Rozenberg. Handle-rewriting hypergraph grammars. *Journal of Computer and System Sciences*, 46(2):218–270, 1993.

- [14] Bruno Courcelle, Johann A. Makowsky, and Udi Rotics. Linear time solvable optimization problems on graphs of bounded clique-width. *Theory of Computing Systems*, 33(2):125–150, 2000.
- [15] Bruno Courcelle and Stephan Olariu. Upper bounds to the clique width of graphs. *Discrete Applied Mathematics*, 101(1-3):77–114, 2000.
- [16] G. Damiand. Quelques propriétés des graphes distances héréditaires. Dissertação de mestrado, Université Montpellier II, Junho 1997.
- [17] Guillaume Damiand, Michel Habib, and Christophe Paul. A simple paradigm for graph recognition: application to cographs and distance hereditary graphs. *Theoretical Computer Science*, 263(1-2):99–111, 2001.
- [18] Y. Daniel Liang and Maw-Shang Chang. Minimum feedback vertex sets in cocomparability graphs and convex bipartite graphs. *Acta Informatica*, 34(5):337–346, 1997.
- [19] Raquel de Souza Francisco, Sulamita Klein, and Loana Tito Nogueira. Characterizing (k, l) -partitionable cographs. *Electronic Notes in Discrete Mathematics*, 22:277–280, 2005.
- [20] Marc Demange, Tinaz Ekim, and Dominique de Werra. Partitioning cographs into cliques and stable sets. *Discrete Optimization*, 2(2):145 – 153, 2005.
- [21] Tomás Feder and Pavol Hell. Matrix partitions of perfect graphs. *Discrete Mathematics*, 306(19-20):2450–2460, 2006.
- [22] Tomás Feder, Pavol Hell, and Winfried Hochstättler. Generalized colourings (matrix partitions) of cographs. *Graph Theory in Paris*, pages 149–167, 2007.
- [23] Tomás Feder, Pavol Hell, Sulamita Klein, and Rajeev Motwani. Complexity of graph partition problems. In *Proceedings of the thirty-first annual ACM symposium on Theory of computing*, STOC '99, pages 464–472, Atlanta, Georgia, USA, 1999.
- [24] S. Földes and P.L. Hammer. Split graphs. *Congr. Numer.*, 19:311–315, 1977.
- [25] Fedor V. Fomin, Serge Gaspers, and Artem V. Pyatkin. Finding a minimum feedback vertex set in time $O(1.7548^n)$. In *Proceedings of the 2nd International Workshop on Parameterized and Exact Computation (IWPEC 2006)*, pages 184–191, Zürich, Switzerland, 2006.
- [26] M. R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.
- [27] Gioan, Emeric, Paul, and Christophe. Dynamic distance hereditary graphs using split decomposition. In *Proceedings of the 18th international conference on Algorithms and computation*, ISAAC'07, pages 41–51, Sendai, Japan, 2007. Springer-Verlag.
- [28] Martin Charles Golumbic and Udi Rotics. On the clique-width of some perfect graph classes. *International Journal of Foundations of Computer Science*, 11(3):423–443, 2000.

- [29] Marina Groshaus, Pavol Hell, Sulamita Klein, Loana Tito Nogueira, and Fábio Protti. Cycle transversals in bounded degree graphs. *Discrete Mathematics & Theoretical Computer Science*, 13(1):45–66, 2011.
- [30] P. L. Hammer and F. Maffray. Completely separable graphs. *Discrete Applied Mathematics*, 27(1-2):85–99, April 1990.
- [31] Pavol Hell, Sulamita Klein, Loana Tito Nogueira, and Fábio Protti. Partitioning chordal graphs into independent sets and cliques. *Discrete Applied Mathematics*, 141(1-3):185–194, 2004.
- [32] E. Howorka. A characterization of distance-hereditary graphs. *Q J Math*, 28(4):417–420, 1977.
- [33] Falk Hüffner. Algorithm engineering for optimal graph bipartization. In *Proceedings of the 4th international conference on Experimental and Efficient Algorithms*, WEA’05, pages 240–252, Santorini Island, Greece, 2005.
- [34] João Thompson. Particionando grafos de distância hereditária em conjuntos independentes e cliques. Dissertação de mestrado, Universidade Federal Fluminense, Março 2013.
- [35] H. Lerchs. On cliques and kernels. Technical report, Dept. of Comp. Sci., University of Toronto, 1971.
- [36] Y. Daniel Liang. On the feedback vertex set problem in permutation graphs. *Information Processing Letters*, 52(3):123 – 129, 1994.
- [37] Daniel Lokshtanov, N. S. Narayanaswamy, Venkatesh Raman, M. S. Ramanujan, and Saket Saurabh. Faster parameterized algorithms using linear programming. *The Computing Research Repository (CoRR)*, abs/1203.0833, 2012.
- [38] Chin Lung Lu and Chuan Yi Tang. A linear-time algorithm for the weighted feedback vertex problem on interval graphs. *Inf. Process. Lett.*, 61(2):107–111, 1997.
- [39] Shin-Ichi Nakano, Ryuhei Uehara, and Takeaki Uno. A new approach to graph recognition and applications to distance-hereditary graphs. *Journal of Computer Science and Technology*, 24(3):517–533, 2009.
- [40] Igor Razgon. Exact computation of maximum induced forest. In *Proceedings of the 10th Scandinavian conference on Algorithm Theory*, SWAT’06, pages 160–171, Riga, Latvia, 2006.
- [41] Bruce A. Reed, Kaleigh Smith, and Adrian Vetta. Finding odd cycle transversals. *Oper. Res. Lett.*, 32(4):299–301, 2004.
- [42] J. L. Szwarcfiter. *Grafos e Algoritmos Computacionais*. Editora Campus, 1986.
- [43] Mihalis Yannakakis. Node- and edge-deletion NP-complete problems. In *Proceedings of the tenth annual ACM symposium on Theory of computing*, STOC ’78, pages 253–264, San Diego, California, USA, 1978.