



UNIVERSIDADE
FEDERAL DO CEARÁ

**UNIVERSIDADE FEDERAL DO CEARÁ
CENTRO DE CIÊNCIAS
DEPARTAMENTO DE COMPUTAÇÃO
PROGRAMA DE MESTRADO E DOUTORADO EM CIÊNCIA DA
COMPUTAÇÃO**

Samuel Nascimento de Araújo

Número de Dominação Romana em Grafos

**FORTALEZA – CE
Agosto de 2016**

Samuel Nascimento de Araújo

Número de Dominação Romana em Grafos

Dissertação de Mestrado apresentada ao Programa de Mestrado e Doutorado em Ciência da Computação, do Departamento de Computação da Universidade Federal do Ceará, como requisito parcial para obtenção do Título de Mestre em Ciência da Computação. Área de concentração: Teoria dos Grafos.

Orientador: Prof. Dr. Rudini Menezes Sampaio

FORTALEZA – CE

Agosto de 2016

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Biblioteca Universitária
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

A691n Araújo, Samuel Nascimento de.
Número de Dominação Romana em Grafos / Samuel Nascimento de Araújo. – 2016.
50 f. : il.

Dissertação (mestrado) – Universidade Federal do Ceará, Centro de Ciências, Programa de Pós-Graduação em Ciência da Computação, Fortaleza, 2016.
Orientação: Prof. Dr. Rudini Menezes Sampaio.

1. Computabilidade e Complexidade. 2. Teoria dos Grafos. I. Título.

CDD 005

SAMUEL NASCIMENTO DE ARAÚJO

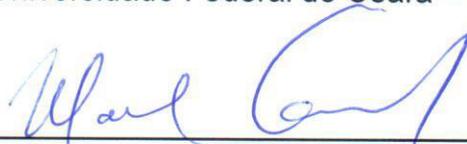
Número de Dominação Romana em Grafos

Dissertação submetida à Coordenação do Curso de Pós-Graduação em Ciência da Computação, da Universidade Federal do Ceará, como requisito para a obtenção do grau de Mestre em Ciência da Computação.

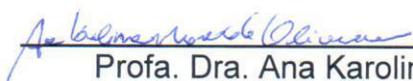
BANCA EXAMINADORA



Prof. Dr. Rudini Menezes Sampaio
(Presidente)
Universidade Federal do Ceará – UFC



Prof. Dr. Manoel Bezerra Campelo Neto
Universidade Federal do Ceará – UFC



Profa. Dra. Ana Karolinnna Maia de Oliveira
Universidade Federal do Ceará – UFC



Prof. Dr. Vinicius Fernandes dos Santos
Universidade Federal de Minas Gerais– UFMG

Fortaleza, 26 de agosto de 2016.

Agradecimentos

A minha família, em especial minha mãe, meu pai, Enio, Cilene, Lucas e Carol por estarem sempre presente nas horas em que mais precisei, me apoiar e reforçar todas as minhas decisões.

Ao meu orientador Rudini Menezes Sampaio por ser compreensivo, atencioso e um ótimo modelo de orientador e de pessoa. Agradeço pelas diversas ajudas que ele me forneceu nos momentos de entrave deste trabalho e por sua colaboração para finalização deste projeto assim como seus conselhos em diversos momentos em que necessitei.

Ao professor Manoel Campêlo, que sem ele não teria certeza de conseguir chegar até este ponto. Agradeço por suas difíceis mais excepcionais disciplinas que me fizeram ter outro ponto de vista sobre esta área de estudo que adoro. Um obrigado pessoal por seus emails encorajadores e seu apoio e confiança nas minhas habilidades.

Aos meus amigos no ParGO: Rennan, Rafael, Fábio, Márdson, Marcilon, Luiz Alberto, Eliezer, Arthur, Pablo, Paulo Henrique, Nicolás, Eurinardo, Jefferson, Adriano, Wladimir, Karol e Ernando por estarem sempre presentes no laboratório 2 (e muitas vezes, mesmo fora dele) para tirar dúvidas, compartilhar pensamentos, resolver problemas, dividir conquistas e emoções, dentre muitas outras coisas.

À Tatiane Fernandes por ser essa pessoa incrível e especial que sempre consegue ser a melhor, mesmo nas piores situações. Agradeço por ter estado por perto nos piores e melhores momentos e por fazer esses dois anos de mestrado valerem a pena, além das mil outras coisas que fez por mim.

Aos professores do grupo de pesquisa ParGO, por serem excelentes profissionais, e que por anos tenho como minha segunda família. Obrigado pelas excelentes aulas e por todos os momentos de descontração vividos nos eventos em que estivemos presente. Em especial agradeço a professora Cláudia Linhares e ao professor Victor Campos.

Aos amigos e funcionários do MDCC/UFC por sempre nos ajudar mesmo quando não percebemos. E um especial obrigado à Marcos Souza.

Às minhas amigas do Juazeiro do Norte: Lívia, Herlane, Jéssica, Mizraim e em especial a Bruna Dias que nos últimos 6 meses fizeram toda a diferença para que eu concluísse este trabalho.

À Carolina Oliveira e Erika Santos que sem o apoio e a persistência delas eu não estaria concluindo este trabalho. Sempre dispostas a conversar e que sempre se preocuparam com meu bem estar e minha saúde.

Ao CNPq pelo apoio financeiro.

“The only way to get smarter is by playing a smarter opponent”.

(Fundamentals of Chess 1883)

Resumo

No problema de dominação romana de um grafo, os vértices são atribuídos um valor de $\{0, 1, 2\}$, de tal maneira que cada vértice atribuído o valor 0 é adjacente a um vértice que foi atribuído o valor 2. O número de dominação romana é a menor soma possível de todos os valores dos vértices em tal atribuição. Nesta dissertação apresentamos a história do problema, e provamos que o número de dominação romana é NP-Completo mesmo para subgrafos induzidos de grids e é APX-difícil mesmo para grafos bipartidos com o grau máximo 4. Nós também provamos que o número de dominação romana é tratável por parâmetro fixo para grafos com treewidth local limitada, como grafos com grau máximo limitado ou genus limitado (como por exemplo grafos planares). Nós também obtivemos resultados de complexidade quando consideramos digrafos como entrada para o problema, tais como torneios bipartidos e digrafos planares.

Palavras-chave: Dominação, Dominação Romana, Complexidade Computacional, Grid, Digrafos.

Abstract

In a Roman domination of a graph, vertices are assigned a value from $\{0, 1, 2\}$ in such a way that every vertex assigned the value 0 is adjacent to a vertex assigned the value 2. The Roman domination number is the minimum possible sum of all values in such an assignment. In this dissertation, we show the history of the problem and prove that the Roman domination number is NP-Complete even for induced subgraphs of grids and it is APX-hard even for bipartite graphs with maximum degree 4. We also prove that the Roman domination number is fixed parameter tractable for graphs with bounded local treewidth, as graphs with bounded maximum degree or bounded genus (like planar graphs or toroidal graphs). We also obtain complexity results when we consider digraphs as an input for the problem such as bipartite tournaments and planar digraphs.

Key-words: Domination, Roman Domination, Computational Complexity, Grid, Digraphs.

Sumário

	Sumário	7
	Lista de ilustrações	8
1	INTRODUÇÃO	9
1.1	História da Dominação em Grafos	9
1.2	História e Motivação do Problema de Dominação Romana em Grafos	11
1.3	Conceitos Básicos	12
1.4	Resultados Existentes	13
1.4.1	Complexidade do Problema	13
1.4.2	Aproximabilidade	16
1.4.3	Complexidade Parametrizada	17
1.4.4	Propriedades da Dominação Romana	19
1.4.5	Limites Para o Número de Dominação Romana	20
1.4.6	Grafos para os quais $\gamma_R(G) = 2\gamma(G)$	20
2	RESULTADOS OBTIDOS EM GRAFOS	21
2.1	Complexidade de Subgrafos Induzidos de Grid	21
2.2	Tratabilidade por Parâmetro Fixo	26
3	RESULTADOS OBTIDOS EM DIGRAFOS	29
3.1	Dominação Romana de Digrafos Planares	29
3.2	Dominação Romana em Torneios Bipartidos	33
4	PROGRAMAÇÃO MATEMÁTICA	36
4.1	F_1 - The Set Covering Deployment Problem (SCDP)	36
4.2	F_2 - Differential de um Grafo	37
4.3	F_3 -Tática Binária de Proteção de Grafos	38
4.4	Comparativo Entre as Formulações	39
4.5	Estudo Poliédrico	41
4.5.1	Dimensão do Poliedro	42
4.5.2	Total Unimodularidade	45
5	CONCLUSÃO E TRABALHOS FUTUROS	47
	REFERÊNCIAS	49

Lista de ilustrações

Figura 1 – Rainhas	10
Figura 2 – Império Romano	11
Figura 3 – Redução do 3-SAT com cláusula $C_1 = \{X_1, \overline{X_2}, \overline{X_n}\}$	15
Figura 4 – Gadget da variável X_i	22
Figura 5 – Redução do Planar 3-SAT: $C_1 = (x_1, x_2), C_2 = (x_1, \overline{x_2}), C_3 = (\overline{x_1}, x_2)$	23
Figura 6 – Redução do PLANAR 3-SAT: $C_1 = (x_1 \vee x_2 \vee x_3), C_2 = (x_1 \vee \overline{x_2} \vee \overline{x_3})$	24
Figura 7 – Redução do PLANAR 3-SAT com exemplo de F_1 e F_2	25
Figura 8 – Grafo Bipartido Completo G	34

1 Introdução

Pelas regras do xadrez, uma peça denominada rainha pode, em um movimento, avançar qualquer número de espaços na vertical, horizontal e diagonal (assumindo que não existem outras peças no seu caminho). Na Figura 1 é possível ver um tabuleiro de xadrez padrão com uma rainha que pode se mover (ou atacar, ou dominar) qualquer quadrado marcado com "X".

Por volta do ano 1850, o problema de determinar qual o número mínimo de rainhas necessárias em um tabuleiro de xadrez, para que cada local do tabuleiro ou fosse ameaçado por uma rainha ou possuísse uma rainha nele, estava sendo estudado por entusiastas de xadrez em várias partes da Europa. Para melhor entender este problema, a Figura 1 mostra um exemplo onde precisamos de no máximo seis rainhas para dominar todas as regiões de um tabuleiro padrão (tamanho 8×8) de xadrez.

Encontrar esse número mínimo de peças de xadrez que domina todas as regiões do tabuleiro é considerado a origem do estudo de conjuntos dominantes em grafos. Porém, foi apenas em 1958 que Claude Berge definiu matematicamente a *dominação* de um grafo, e desde então mais de 1.200 artigos foram publicados sobre este tópico, e este número continua a crescer a cada ano.

No artigo intitulado "*Defend the Roman Empire!*" de 1999 [1], Ian Stewart discutiu uma estratégia que foi usada pelo Imperador Constantino para defender o Império Romano no período do século III. Motivado por este artigo, Cockayne et al. [2] introduziram a noção de dominação romana em grafos, que é uma variação do problema de dominação clássico. Com isso, surgiu uma nova motivação para expandir o conhecimento sobre esse tipo de problema.

Nesta dissertação de mestrado temos como objetivo apresentar o problema de dominação romana em grafos, juntamente com suas propriedades, e expor os resultados obtidos durante o período de pesquisa sobre o problema.

Para servir de introdução ao assunto, apresentamos na seção a seguir um breve histórico da origem do problema de dominação de um grafo que foi retirado do livro *Fundamentals of domination in graphs* de Haynes, Hedetniemi e Slater [3]. E na Seção 1.2 falamos sobre a história da dominação romana em grafos e algumas possíveis aplicações para o problema.

1.1 História da Dominação em Grafos

Apesar do estudo matemático de conjuntos dominantes ter iniciado por volta dos anos 1960, sua origem vem desde antes de 1862, quando Jaenisch [4] estudou o problema de

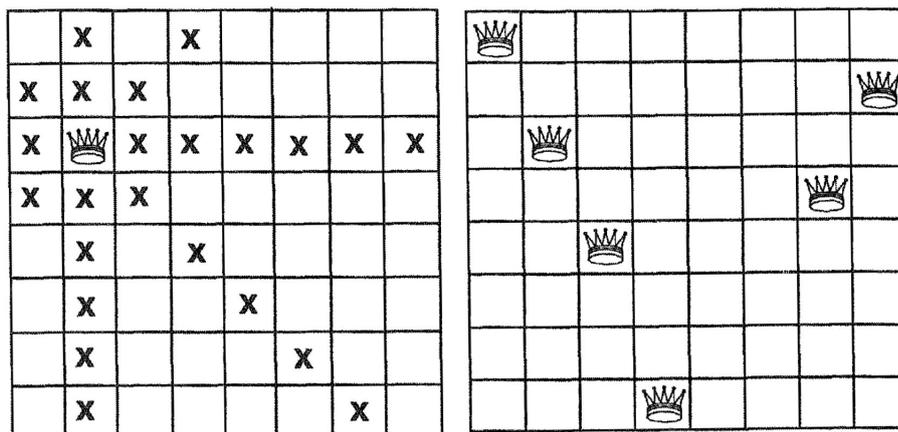


Figura 1 – Rainhas

encontrar o número mínimo de rainhas necessárias para cobrir (ou dominar) um tabuleiro de tamanho $n \times n$.

Como reportado por W. W. Rouse Ball em 1892 [5], entusiastas de xadrez no início do século XIX já estudavam os seguintes problemas:

1. **Cobertura** - Qual o número mínimo de peças de xadrez de um certo tipo necessárias para cobrir/atacar/dominar um tabuleiro de tamanho $n \times n$?
2. **Cobertura Independente** - Qual o número mínimo de peças de xadrez de um certo tipo necessárias para dominar um tabuleiro de tamanho $n \times n$, de tal forma que nenhuma peça ataca outra?
3. **Independência** - Qual o número máximos de peças de xadrez de um certo tipo que podem ser colocadas em um tabuleiro de tamanho $n \times n$ de modo que não exista uma peça que ataca outra?

Essas três perguntas foram estudados em detalhe pelos irmãos Yaglom em 1964, onde eles desenvolveram soluções para alguns desses problemas utilizando as peças de xadrez conhecidas como peão, bispo, cavalo e rei [6].

Em 1958 Claude Berge [7] escreveu um livro de teoria dos grafos onde ele definiu pela primeira vez o número de dominação de um grafo (embora ele tenha chamado de “Coeficiente de Estabilidade Externa”). Em 1962 Oystein Ore [8] usou pela primeira vez os termos “Conjunto Dominante” e “Número de Dominação” em seu livro de teoria dos grafos. Em 1977 Cockayne e Hedetniemi [9] publicaram um artigo comentando os poucos resultados sobre conjuntos dominantes em grafos que eram conhecidos na época, onde eles usaram pela primeira vez a notação $\gamma(G)$ para o número de dominação de um grafo G .

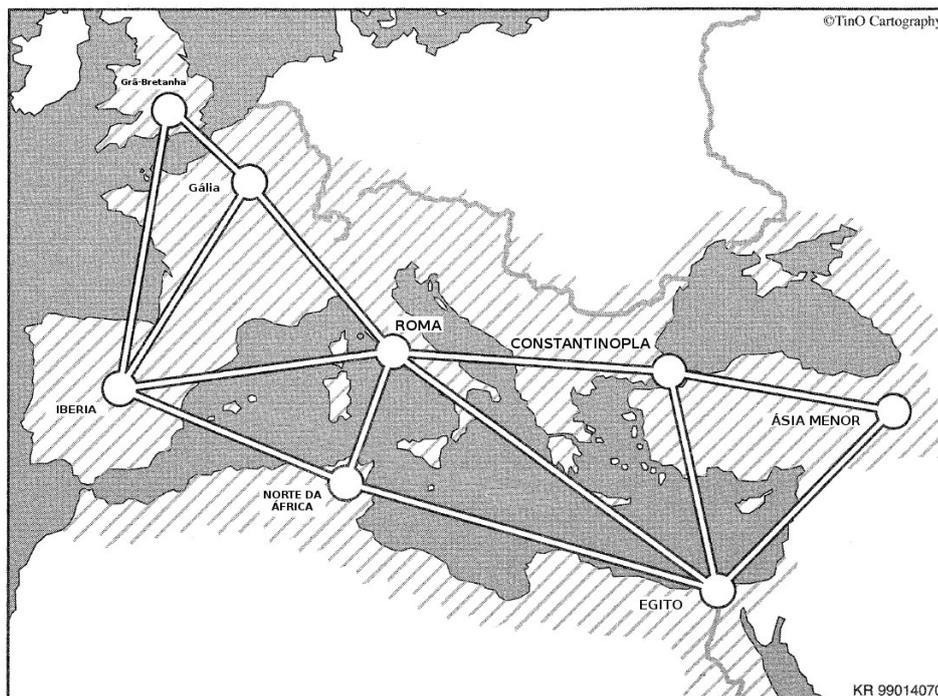


Figura 2 – Império Romano

1.2 História e Motivação do Problema de Dominação Romana em Grafos

No início do século III, o Império Romano estava sob ataque e Constantino, o Grande, tinha que decidir a melhor maneira de alocar suas tropas para proteger todas as regiões que possuía controle. As regiões que o Imperador Constantino controlava podem ser vistas na Figura 2, onde cada região é representada como um círculo (vértice) e um movimento ao longo de uma linha (aresta) entre regiões representa um “*passo*”. A tática do imperador foi colocar as tropas de tal maneira que cada região ou era protegida por sua própria tropa ou era passível de ser protegida por um território vizinho que continha duas tropas (uma das quais podia ser enviada para a região desprotegida em um passo se um conflito ocorresse).

O desafio do imperador era alocar apenas quatro tropas para defender as suas oito regiões. Ele escolheu colocar duas tropas em Roma e duas em sua nova capital, Constantinopla. Essa não foi uma escolha perfeita, pois deixa a Grã-Bretanha desprotegida. Existem pelo menos seis maneiras de melhorar essa tática que foram obtidas por meio de uma formulação matemática binária e apresentada no artigo de ReVelle e Rosing [10]. Por exemplo: colocar duas tropas em Constantinopla e duas na Ibéria. Poucos anos após essa decisão de Constantino, o Império Romano perdeu o controle da Grã-Bretanha (as causas foram certamente mais complexas do que podemos explicar em uma simples modelagem matemática).

O mesmo tipo de programação usado em [10] pode ser utilizado quando as pessoas

querem saber o melhor local de um cidade para colocar um novo hospital ou quartel de bombeiros, por exemplo. Muitas vezes esses problemas de otimização podem ser modelados usando a dominação romana ou uma de suas variantes. Assim como outros problemas de dominação em grafos, o problema de dominação romana também pode ser utilizado na alocação de servidores sem fio através de redes [11, 12] ou como estratégia militar, como foi mostrado em um artigo onde é apontado a utilidade da teoria dos grafos e problemas de otimização no mundo real [13].

Esta dissertação foi organizada da seguinte maneira:

Apresentamos neste capítulo as noções necessárias sobre grafos e digrafos que são usadas para definir formalmente o problema de dominação romana. Assim como uma revisão bibliográfica com importantes propriedades para compreender melhor o problema.

No Capítulo 2 apresentamos resultados obtidos sobre a complexidade do problema de dominação romana em subgrafos induzidos de grid, estudo que nos proporcionou a aceitação de um artigo no *VIII Latin-American Algorithms, Graphs and Optimization Symposium - LAGOS* em 2015. Além disso, apresentamos uma prova de que este problema está em \mathcal{FPT} para grafos com largura local em árvore limitada (*bounded local-threewidth*).

O Capítulo 3 traz resultados obtidos em parceria com Frédéric Havet¹ do *INRIA* em Sophia Antipolis, França. Estes resultados dizem respeito ao problema de dominação romana em digrafos, e para tal, apresentamos as definições necessárias sobre eles.

No Capítulo 4 fazemos um comparativo sobre diferentes formulações matemáticas do problema de dominação romana, assim como obtemos novas formulações e resultados de complexidade baseados em programação matemática.

O Capítulo 5 encerra nosso trabalho mostrando conclusões obtidas sobre o problema e possibilidades para trabalhos futuros.

1.3 Conceitos Básicos

Para o estudo do problema de dominação romana, precisamos de alguns conceitos sobre grafos que serão apresentados nesta seção. Para todas as definições não encontradas nesta seção, usaremos como referência as notações e definições em [14].

Definimos um *grafo* como um par ordenado $G = (V, E)$, onde V é um conjunto finito não vazio de *vértices* e E é um conjunto de pares não-ordenados (chamados *arestas*) de elementos distintos de V . Dado um grafo G , denotamos por $V(G)$ e $E(G)$ seus conjuntos de vértices e arestas, respectivamente. Dado um grafo $G = (V, E)$, definimos a *vizinhança aberta* de um vértice $v \in V$ como $N(v) = \{u \in V | uv \in E\}$ e a *vizinhança fechada* de v como $N[v] = N(v) \cup \{v\}$. Dado um subconjunto $S \subseteq V(G)$, seja $N(S) = \cup_{v \in S} N(v)$ e $N[S] = \cup_{v \in S} N[v]$. O *grau* de v é dado por $d(v) = |N(v)|$ e denotamos por $\Delta(G)$ o maior grau dentre os vértices em G .

¹ <http://www-sop.inria.fr/members/Frederic.Havet>

Dizemos que $G' = (V', E')$ é um *subgrafo* de G se $V' \subseteq V$ e $E' \subseteq E$. No caso particular onde E' contém todas as arestas $xy \in E(G)$ com $x, y \in V'$, dizemos que G' é *subgrafo induzido* de G . Nesse caso, dizemos que G' é induzido por V' e denotado por $G[V']$.

Uma *clique* é definida como um conjunto de vértices tal que existe uma aresta entre quaisquer dois vértices desse conjunto. Analogamente, um *conjunto independente* (também chamado *conjunto estável*) é definido como um conjunto de vértices onde não existe aresta entre quaisquer dois vértices. Um grafo G é *completo* se $V(G)$ é uma clique. Seja K_n o grafo completo com n vértices. Um grafo G é *bipartido* se $V(G)$ pode ser particionado em duas partes U e U' , que formam conjuntos independentes disjuntos de G . O grafo *bipartido completo* $K_{r,s}$ é o grafo bipartido com duas partes de tamanho r e s e todas as arestas entre as partes.

Sabendo o que são grafos e algumas de suas características, podemos agora definir problemas sobre eles. A noção de *conjunto dominante* está diretamente ligada ao problema de dominação romana, e por isso devemos defini-la formalmente. Um conjunto $S \subseteq V(G)$ é *dominante* sobre um conjunto $S' \subseteq V(G)$ se, para todo $v \in S'$, $N[v] \cap S \neq \emptyset$. Um conjunto $S \subseteq V(G)$ é dito dominante de G se é dominante sobre $V(G)$. O *número de dominação* $\gamma(G)$ de G é o tamanho do menor conjunto dominante de G .

Uma função $f : V(G) \rightarrow \{0, 1, 2\}$ é uma *função de dominação romana (FDR)* em um grafo G se todo vértice $u \in V(G)$ com $f(u) = 0$ possui um vizinho v com $f(v) = 2$. O peso de uma FDR f é o valor $\omega(f) = \sum_{u \in V} f(u)$, e o *número de dominação romana* $\gamma_R(G)$ de um grafo G é o menor peso entre as FDR's de G . Definimos ainda a partição ordenada (V_0, V_1, V_2) de V induzidas por f , onde $V_i = \{v \in V | f(v) = i\}$, para $i \in \{0, 1, 2\}$. Como existe uma correspondência 1-1 entre $f : V(G) \rightarrow \{0, 1, 2\}$ e a partição ordenada (V_0, V_1, V_2) , denotaremos $f = (V_0, V_1, V_2)$. Nessa representação, o seu peso é $\omega(f) = |V_1| + 2|V_2|$. Uma função de dominação romana é chamada de $\gamma_R(G)$ -*função* ou γ_R -*função* se ela é uma FDR de G e $\omega(f) = \gamma_R(G)$.

Observe que, dado um conjunto dominante V' de G , podemos obter uma FDR (V_0, V_1, V_2) tal que $V_0 = V(G) \setminus V'$, $V_1 = \emptyset$ e $V_2 = V'$, com peso $2 \cdot |V'|$. Por outro lado, dada uma FDR (V_0, V_1, V_2) de G , podemos obter um conjunto dominante $V' = V_1 \cup V_2$ de G .

1.4 Resultados Existentes

Apresentamos nesta seção um levantamento de importantes resultados obtidos no estudo do problema de dominação romana.

1.4.1 Complexidade do Problema

É importante mostrar a complexidade do problema em qualquer grafo, pois em capítulos posteriores provamos que restringindo o problema a certas classes de grafo, como por exemplo subgrafos induzidos de grid, o problema permanece difícil de ser resolvido.

Nosso método de prova é bastante similar ao que foi apresentado para grafos gerais por Hedetniemi et.al. [15] em 2000, e por isso um detalhamento maior desta prova é apresentado abaixo como em [16].

Definição.

Problema de Decisão do Número de Dominação Romana.

INSTÂNCIA: Um grafo $G = (V, E)$ e um inteiro positivo $k \leq |V|$.

PERGUNTA: G possui uma função de dominação romana f com peso

$$\omega(f) \leq k?$$

Para mostrar que o problema de decisão do número de dominação romana é NP-completo foi feita uma redução do problema 3-SAT [17], que é definido como segue:

Definição.

3-SATISFIABILITY (3-SAT).

INSTÂNCIA: Uma coleção de cláusulas $\mathcal{C} = \{C_1, \dots, C_m\}$ de um conjunto finito de variáveis $X = \{X_1, \dots, X_n\}$ tal que $|C_i| = 3$ para $1 \leq i \leq m$.

PERGUNTA: Existe uma atribuição de valores a X que satisfaça todas as cláusulas em \mathcal{C} ?

Em uma instância de 3-SAT, uma cláusula C_j consiste de exatamente três literais, onde um literal ou é uma variável $X_i \in X$ ou é seu complemento, que vamos denotar \overline{X}_i . Uma atribuição verdadeira é uma função $\phi : X \rightarrow \{VERDADEIRO, FALSO\}$ e uma cláusula é satisfeita por uma atribuição verdadeira ϕ se X_i é um literal na cláusula e $\phi(X_i) = VERDADEIRO$, ou se \overline{X}_i é um literal na cláusula e $\phi(X_i) = FALSO$ (que também podemos ver como $\phi(\overline{X}_i) = VERDADEIRO$).

Assim, seja I uma instância do 3-SAT como definido acima. Iremos criar um grafo $G = (V, E)$ de tal forma que I possui uma atribuição de valores satisfazível se e somente se existe uma função de dominação romana f em G com $\omega(f) \leq 2n$, onde n é o número de variáveis de I .

Introduzimos um vértice c_j para cada cláusula C_j e, para cada variável X_i , introduzimos um grafo bipartido completo $K_{2,3}$ com partes $R_i = \{x_i, \overline{x}_i\}$ e $S_i = \{u_i, w_i, z_i\}$. Para cada variável X_i , adicionamos a aresta (x_i, \overline{x}_i) e, para cada cláusula C_j , adicionamos as três arestas que vão de c_j para os vértices correspondentes aos literais da cláusula C_j . Um exemplo da construção do G é dado na Figura 3.

Proposição 1 (Hedetniemi et al. [15]). *Uma instância I do 3-SAT é satisfazível se e somente se no grafo $G = (V, E)$ existe uma função de dominação romana f com $\omega(f) \leq 2n$.*

Demonstração. Seja $\phi : X \rightarrow \{VERDADEIRO, FALSO\}$ uma atribuição de valores satisfazível para uma instância I do 3-SAT. Seja G o grafo obtido de I pela construção descrita anteriormente. Para cada variável X_i , faça $f(x_i) = 2$ se $\phi(X_i) = VERDADEIRO$ e

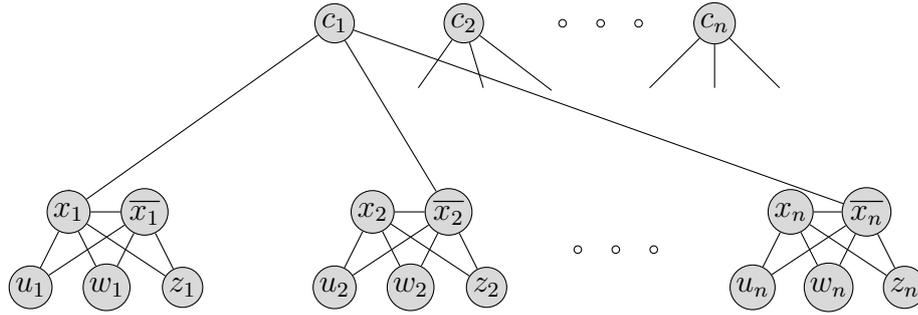


Figura 3 – Redução do 3-SAT com cláusula $C_1 = \{X_1, \overline{X_2}, \overline{X_n}\}$

faça $f(\overline{x}_i) = 2$ se $\phi(X_i) = \text{FALSO}$. Atribua $f(v) = 0$ para os demais vértices $v \in V(G)$. Como descrito anteriormente, seja $f = (V_0, V_1, V_2)$. Como todo vértice c_j possui pelo menos um vizinho em V_2 (pois ϕ marca pelo menos uma literal de C_j como *VERDADEIRO*) e x_i ou \overline{x}_i estão em V_2 e ambos são dominantes em $R_i \cup S_i$, temos que f é uma FDR com $\omega(f) = 2n$.

Assuma que G possui uma FDR $f = (V_0, V_1, V_2)$ com $\omega(f) \leq 2n$. Portanto, para cada variável X_i , precisamos ter $f(x_i) = 2$ ou $f(\overline{x}_i) = 2$ para que os vértices de S_i sejam dominados. Portanto $w(f) = 2n$ e por isso $f(c_j) = 0$ para toda cláusula C_j . Como o conjunto de vértices em V_2 é dominante sobre o conjunto de vértices que representam as cláusulas, temos que a instância do 3-SAT possui uma atribuição de valores satisfazível ϕ , onde $\phi(X_i) = \text{VERDADEIRO}$ se e somente se $f(x_i) = 2$. \square

Usando uma construção similar, Hedetniemi et.al. [15] mostrou que o problema do número de dominação romana é NP-Completo mesmo quando restrito a grafos cordais, bipartidos, split ou planares [15]. A Tabela 1 mostra os resultados de complexidade existentes para classes de grafo bem conhecidas.

Classe de Grafo	Complexidade	Referência
Árvores	Linear	Cockayne et al. - 2004
Bipartido	NP-Completo	Hedetniemi et al. - 2000
Cordal	NP-Completo	Hedetniemi et al. - 2000
Cografos	Linear	Liedloff et al. - 2005
Intervalo	Linear	Liedloff et al. - 2005
Largura em árvore limitada	Linear	Peng et al. - 2005
Planar	NP-Completo	Hedetniemi et al. - 2000
Split	NP-Completo	Hedetniemi et al. - 2000

Tabela 1 – Complexidade do problema de dominação romana para diferentes classes de grafo.

Para a proposição a seguir necessitamos da seguinte definição: um problema é dito *fortemente NP-difícil* se, mesmo se todos os parâmetros numéricos da instância do problema forem delimitados por um polinômio no tamanho da entrada, ele permanece NP-difícil.

Proposição 2 (Pagourtzis et al. [11]). *O número de Dominação Romana é fortemente NP-difícil mesmo que sua entrada seja um grafo planar G .*

Proposição 3 (Liedloff et al. [18]). *O número de Dominação Romana pode ser computado em tempo $O(n)$ em grafos de intervalo.*

Proposição 4 (Liedloff et al. [18]). *O número de Dominação Romana em um cografo pode ser computado em tempo linear.*

1.4.2 Aproximabilidade

Dado um problema de otimização P , seja $opt_P(I)$ o valor da solução ótima de alguma instância I de P e, para uma solução S de I , seja $val_P(I, S)$ o valor associado de S em I . Dada uma instância I de P e uma solução S de I o *fator de aproximação* $\mathcal{R}_P(I, S)$ é definido por:

$$\mathcal{R}_P(I, S) = \max \left\{ \frac{opt_P(I)}{val_P(I, S)}, \frac{val_P(I, S)}{opt_P(I)} \right\}.$$

Dada uma constante $r \geq 1$, um algoritmo r -aproximativo para P é um algoritmo que, aplicado a qualquer instância I de P , produz uma solução S tal que $\mathcal{R}(I, S) \leq r$. Se P tem um algoritmo r -aproximativo em tempo polinomial para alguma constante $r > 1$, dizemos que P pertence a APX . Se P tem um algoritmo r -aproximativo em tempo polinomial para qualquer constante $r > 1$, dizemos que P pertence a $PTAS$ (*polynomial time approximation scheme*). Se P tem um algoritmo r -aproximativo para qualquer constante $r > 1$ com tempo $O(n^c)$, onde c é uma constante fixa independente de r , dizemos que P pertence a $FPTAS$ (*full polynomial time approximation scheme*). Claramente $FPTAS \subseteq PTAS \subseteq APX$.

Uma redução de um problema P_1 para um problema P_2 consiste de um par (f, g) de funções computáveis em tempo polinomial tal que, para qualquer instância I de P_1 , (a) $f(I)$ é uma instância de P_2 , (b) $g(I, S)$ é uma solução viável de I , para qualquer solução viável S de $f(I)$.

Uma redução contínua [19] de P_1 para P_2 é uma tripla (f, g, γ) , onde (f, g) é uma redução de P_1 para P_2 e $\gamma \geq 1$ é uma constante, tal que, se $\mathcal{R}_{P_2}(f(I), S) \leq r$ ($r \geq 1$), então $\mathcal{R}_{P_1}(I, g(S)) \leq \gamma r$ para cada instância I de P_1 e para toda solução viável S de $f(I)$. Desta definição, P_2 em APX implica P_1 em APX . Além disso, se existe um algoritmo r -aproximativo em tempo polinomial para P_2 para algum $r \geq 1$, então existe um algoritmo γr -aproximativo em tempo polinomial para P_1 .

Uma redução preservadora de aproximação (*approximation preserving reduction*, ou simplesmente *AP-redução*) [20] de P_1 para P_2 é uma tripla (f, g, γ) , onde (f, g) é uma redução de P_1 para P_2 e γ é uma constante positiva, tal que, se $\mathcal{R}_{P_2}(f(I), S) \leq r$, então

$\mathcal{R}_{P_1}(I, g(S)) \leq 1 + \gamma(r - 1)$ para cada instância I de P_1 e para toda solução viável S de $f(I)$. Escrevemos $P_1 \leq_{AP} P_2$. Nessa redução, P_2 em PTAS implica P_1 em PTAS.

Dizemos que um problema P de otimização é *APX-difícil* se $Q \leq_{AP} P$ para todo problema Q em APX. Com isso, se P é APX-difícil e P está em PTAS, então $APX=PTAS$. Um problema é *APX-completo* se ele é APX e APX-difícil.

O tipo de redução usada mais frequentemente para provar que um problema é APX-difícil é a *L-redução* [20]. Dizemos que P_1 é L-reduzível para P_2 (denotado por $P_1 \leq_L P_2$) se existe uma quádrupla (f, g, α, β) , onde (f, g) é uma redução de P_1 para P_2 e α e β são constantes positivas tais que, para qualquer instância I de P_1 : (a) $|opt_{P_2}(f(I))| \leq \alpha |opt_{P_1}(I)|$, e (b) $|opt_{P_1}(I) - val_{P_1}(I, g(I, S))| \leq \beta |opt_{P_2}(f(I)) - val_{P_2}(f(I), S)|$ para qualquer solução viável S de $f(I)$.

Sabemos que, se $P_1 \leq_L P_2$ e P_1 é APX, então $P_1 \leq_{AP} P_2$ (Lema 8.2 de [20]). Além disso, se $P_1 \leq_{AP} P_2$ e P_1 é APX-difícil, então P_2 é também APX-difícil.

Proposição 5 (Pagourtzis et al. [11]). *O número de Dominação Romana pode ser aproximado por um fator $2 + 2 \ln n$, porém (a menos que $P = NP$) não existe algoritmo com aproximação $c \ln n$ para algum $0 < c < 2$.*

Proposição 6 (Pagourtzis et al. [11]). *O problema de Dominação Romana em grafos planares tem um esquema de aproximação em tempo polinomial (PTAS), porém (a menos que $P = NP$) não existe um esquema de aproximação de tempo totalmente polinomial (FPTAS).*

1.4.3 Complexidade Parametrizada

Dizemos que um algoritmo é FPT (*tratável com parâmetro fixo*) em um parâmetro $k = k(I)$ sobre suas entradas I se seu tempo de execução é $f(k(I)) \cdot |I|^c$, onde c é uma constante fixa e f é uma função computável [21]. Um *problema parametrizado* é um par (P, k) , onde P é um problema de decisão e $k = k(I)$ é um parâmetro sobre instâncias I de P . Dizemos que um problema parametrizado (P, k) é FPT se existe algoritmo FPT em k que decide P .

Quando dizemos que um problema de decisão P é FPT com relação a um parâmetro k , queremos dizer que (P, k) é FPT. Quando dizemos que um problema de decisão P é FPT (sem mencionar o parâmetro), subentende-se que P possui um parâmetro natural k e que (P, k) é FPT.

A notação $O^*(\cdot)$ se tornou um padrão para a representação da análise de tempo de algoritmos exatos. Ela significa suprimir não só constantes (como a notação mais comum $O(\cdot)$) mas também de partes polinomiais da função. Com isso, um problema parametrizado (P, k) está em FPT se e somente se existe algoritmo que decide P em tempo $O^*(f(k(I)))$ para alguma função f computável.

Proposição 7 (Fernau [22]). *O problema de decidir se um grafo tem uma FDR com peso no máximo k pode ser resolvido por um algoritmo FPT em tempo $O^*(2^{22.165\sqrt{k}})$ em grafos planares.*

Uma *árvore de decomposição* de um grafo G é uma dupla (T, \mathcal{X}) , onde T é uma árvore, \mathcal{X} contém um subconjunto $X_t \subseteq V(G)$ para cada nó t de T e: (i) todo vértice de G pertence a algum $X_t \in \mathcal{X}$, (ii) toda aresta tem suas extremidades em um mesmo $X_t \in \mathcal{X}$, e (iii) se $v \in X_i \cap X_j$, então $v \in X_t$ para todo nó t no único caminho entre i e j em T . A largura em árvore (*treewidth*) de (T, \mathcal{X}) é $\max\{|X_t| - 1 : t \in T, X_t \in \mathcal{X}\}$ e a largura em árvore $tw(G)$ de G é a menor largura em árvore entre todas as árvores de decomposição de G [23].

Proposição 8 (Fernau [22]). *O número de Dominação Romana de um grafo G pode ser determinado por um algoritmo FPT na largura em árvore $tw(G)$ em tempo $O(5^{tw(G)}|V(G)|)$.*

A seguir, adotaremos a notação de [23] para definir a hierarquia W de problemas parametrizados.

Uma redução FPT de um problema parametrizado (P_1, k_1) para outro (P_2, k_2) é uma dupla (f, g) tal que, para cada instância I de P_1 , (a) $f(I)$ é uma instância de P_2 , (b) f é computável por um algoritmo FPT com relação ao parâmetro k_1 e (c) g é uma função computável tal que $k_2(f(I)) \leq g(k_1(I))$. Com isso, se (P_2, k_2) é FPT, então (P_1, k_1) também é FPT.

Fórmulas de lógica proposicional são fórmulas lógicas construídas a partir de um conjunto de variáveis $\{x_1, \dots, x_n\}$ aplicando operadores \neg (negação), \wedge (conjunção) e \vee (disjunção). Dada uma classe Φ de fórmulas lógicas proposicionais, seja $\text{WSAT}(\Phi)$ o problema de decidir se, dada uma fórmula $\varphi \in \Phi$ e um inteiro k , existe uma atribuição satisfazível para φ com no máximo k variáveis verdadeiras. Seja $\text{p-WSAT}(\Phi)$ o problema parametrizado com $\text{WSAT}(\Phi)$ como problema de decisão e o inteiro k como parâmetro.

Para definir a hierarquia W , precisamos definir as classes $\Gamma_{t,d}$ e $\Delta_{t,d}$ de fórmulas lógicas proposicionais. Sejam $\Gamma_{0,d}$ e $\Delta_{0,d}$, respectivamente, as classes de fórmulas do tipo $\lambda_1 \wedge \dots \wedge \lambda_c$ e $\lambda_1 \vee \dots \vee \lambda_c$ para qualquer $1 \leq c \leq d$, onde λ_i é um literal (variável ou complemento de variável) para todo i . Para $t \geq 1$, sejam $\Gamma_{t,d}$ e $\Delta_{t,d}$ respectivamente as classes de fórmulas do tipo $\delta_1 \wedge \dots \wedge \delta_n$ e $\gamma_1 \vee \dots \vee \gamma_n$, para qualquer inteiro n , tal que $\delta_i \in \Delta_{t-1,d}$ e $\gamma_i \in \Gamma_{t-1,d}$ para todo i .

Por exemplo, $\Gamma_{2,1}$ é a classe de todas as fórmulas na FNC (forma normal conjuntiva, que são as instâncias do clássico problema SAT de decisão) e $\Gamma_{1,3}$ é a classe de todas as fórmulas na 3-FNC (que são as instâncias do clássico problema 3SAT de decisão).

Seja $W[t]$ a classe de problemas parametrizados que possuem uma redução FPT para $\text{p-WSAT}(\Gamma_{t,d})$ para algum $d \geq 1$. Dizemos que um problema parametrizado (P, k) é $W[t]$ -Difícil se existe uma redução FPT de $\text{p-WSAT}(\Gamma_{t,d})$ para (P, k) . Finalmente, um problema parametrizado é $W[t]$ -Completo se é $W[t]$ e $W[t]$ -Difícil.

Por exemplo, o problema parametrizado da clique de tamanho k é $W[1]$ -Completo e o do conjunto dominante de tamanho k é $W[2]$ -Completo.

Sabe-se que $FPT=W[0]$ e que $W[i] \subseteq W[j]$ para todo $i \leq j$ [23]. As classes na hierarquia W são também fechadas sob redução FPT.

Proposição 9 (Fernau [22]). *O problema de Dominação Romana é $W[2]$ -completo.*

1.4.4 Propriedades da Dominação Romana

Proposição 10 (Cockayne et al. [2]). *Para qualquer grafo G ,*

$$\gamma(G) \leq \gamma_R(G) \leq 2\gamma(G)$$

Proposição 11 (Cockayne et al. [2]). *Para qualquer grafo G com n vértices,*

$$\gamma(G) = \gamma_R(G) \Leftrightarrow G = \overline{K_n}$$

Seja $v \in S \subseteq V$. Um vértice u é dito *vizinho privado* (*private neighbor*) de v com respeito a S (denotado por u é um S -pn de v) se $u \in N[v] \setminus N[S \setminus \{v\}]$. Um S -pn de v é *vizinho privado externo* se ele é um vértice de $V - S$. Para simplificar a notação, dado um subgrafo H , dizemos que u é um H -pn de v se u é um $V(H)$ -pn de v .

Proposição 12 (Cockayne et al. [2]). *Seja $f = (V_0, V_1, V_2)$ uma FDR com peso $\gamma_R(G)$ de um grafo G e seja $H = G[V_0 \cup V_2]$. Então:*

- a) $G[V_1]$, o subgrafo induzido por V_1 , tem grau máximo 1.
- b) Não existe aresta de G entre V_1 e V_2 .
- c) Cada vértice de V_0 é adjacente a no máximo dois vértices de V_1 .
- d) V_2 é um conjunto dominante de $G[V_0 \cup V_2]$.
- e) Cada vértice $v \in V_2$ possui pelo menos dois H -pn's.
- f) Se v é isolado em $G[V_2]$ e tem exatamente um H -pn externo $w \in V_0$, então $N(w) \cap V_1 = \emptyset$.
- g) Seja k o número de vértices não isolados em $G[V_2]$ e seja $C = \{v \in V_0 : |N(v) \cap V_2| \geq 2\}$. Então $|V_0| \geq |V_2| + k + |C|$.

Proposição 13 (Cockayne et al. [2]). *Seja $f = (V_0, V_1, V_2)$ uma FDR com peso $\gamma_R(G)$ de um grafo G conexo e seja $H = G[V_0 \cup V_2]$. Então:*

- a) V_1 é um conjunto independente e $V_0 \cup V_2$ é uma cobertura de vértices.
- b) V_0 é um conjunto dominante sobre V_1 .
- c) Cada vértice de V_0 é adjacente a no máximo um vértices de V_1 .
- d) Se $v \in G[V_2]$ tem exatamente dois H -pn's w_1 e w_2 em V_0 , então não existem vértices $y_1, y_2 \in V_1$ tais que (y_1, w_1, v, w_2, y_2) é uma sequência de vértices de um caminho P_5 .
- e) $n_0 \geq \frac{3n}{7}$, e este limite é apertado até mesmo para árvores.

1.4.5 Limites Para o Número de Dominação Romana

Proposição 14 (Cockayne et al. [2]). *Para qualquer grafo G com n vértices e com grau máximo $\Delta(G)$,*

$$\gamma_R(G) \geq \frac{2n}{\Delta(G) + 1}$$

Proposição 15 (Cockayne et al. [2]). *Para um grafo G com n vértices e com grau mínimo $\delta(G)$,*

$$\gamma_R(G) \leq n \cdot \frac{2 + \ln((1 + \delta(G))/2)}{1 + \delta(G)}$$

Proposição 16 (Chambers et al. [24]). *Se G é grafo com n vértices, então*

$$\gamma_R(G) \leq 4n/5$$

Proposição 17 (Chambers et al. [24]). *Se G é grafo com n vértices, então*

$$\gamma_R(G) \leq n - \Delta(G) + 1$$

Proposição 18 (Favaron et al. [25]). *Seja $f = (V_0, V_1, V_2)$ qualquer $\gamma_R(G)$ -função de um grafo conexo G de ordem $n \geq 3$, então*

- $1 \leq |V_2| \leq \frac{2n}{5}$
- $0 \leq |V_1| \leq \frac{4n}{5} - 2$
- $\frac{n}{5} + 1 \leq |V_0| \leq n - 1$

1.4.6 Grafos para os quais $\gamma_R(G) = 2\gamma(G)$

Dado um grafo G , temos que $\gamma_R(G) \leq 2\gamma(G)$ da Proposição 10. Assim dizemos que G é um grafo romano se $\gamma_R(G) = 2\gamma(G)$. Sabemos que $P_{3k}, P_{3k+2}, C_{3k}, C_{3k+2}$ são grafos romanos para $k \geq 1$, e encontrar diferentes grafos romanos permanece um problema em aberto.

Proposição 19 (Cockayne et al. [2]). *Um grafo G é romano se e somente se ele tem uma FDR com peso $\gamma_R(G) f = (V_0, V_1, V_2)$ com $|V_1| = 0$.*

2 Resultados Obtidos em Grafos

2.1 Complexidade de Subgrafos Induzidos de Grid

O *grafo grid* é o grafo com conjunto de vértices $\mathbb{Z} \times \mathbb{Z}$ e conjunto de arestas $E = \{(i, j), (i', j')\} : |i - i'| + |j - j'| = 1, i, i', j, j' \in \mathbb{Z}\}$. Analisando o problema de dominação romana restrito a subgrafos induzidos de grid, nós provamos que computar $\gamma_R(G)$ permanece NP-difícil. Nesta seção apresentamos como chegamos a este resultado.

Teorema 2.1. *O problema de dominação romana é NP-completo mesmo para grafos planares bipartidos com grau máximo 4.*

Demonstração. É fácil ver que o problema de decisão está em NP. Para isso, dada uma instância (G, k) , basta receber como certificado uma partição (V_0, V_1, V_2) de $V(G)$ e verificar (em tempo polinomial) se $|V_1| + 2|V_2| \leq k$ e se V_2 é um conjunto dominante de V_0 .

A redução é feita a partir do PLANAR 3-SAT. Dado um conjunto $\mathcal{C} = \{C_1, \dots, C_m\}$ de cláusulas, o *grafo de \mathcal{C}* é o grafo que tem um vértice x_i e \bar{x}_i para cada variável X_i , um vértice c_j para cada cláusula C_j , uma aresta entre cada literal e seu complemento e tem uma aresta entre um literal e uma cláusula se e somente se a cláusula contém o literal em \mathcal{C} . Dizemos que \mathcal{C} é uma *fórmula planar*, se este grafo é planar. O PLANAR 3-SAT é o problema 3-SAT restrito a fórmulas planares. Dizemos que \mathcal{C} é uma *fórmula restrita* se qualquer cláusula tem no máximo 3 literais, cada variável aparece em no máximo 3 cláusulas e cada literal aparece em pelo menos 1 cláusula. Sabemos que o problema do PLANAR 3-SAT é NP-completo mesmo para fórmulas planares restritas (Teorema 2a de [26]).

Dado uma fórmula planar restrita \mathcal{C} com k variáveis e m cláusulas, obtemos um grafo G tal que \mathcal{C} é satisfazível se e somente se $\gamma_R(G) \leq 10k$. A partir do grafo de \mathcal{C} , construa G substituindo os vértices x_i e \bar{x}_i pelo *gadget* mostrado na Figura 4, onde temos dois vértices representando o literal X_i e dois vértices representando o literal \bar{X}_i , respectivamente $x_i^1, x_i^2, \bar{x}_i^1$ e \bar{x}_i^2 . O *gadget* de uma cláusula C_j continua a ser representado pelo vértice c_j . Se C_j contém X_i , crie uma aresta entre c_j e um vértice $l_i \in \{x_i^1, x_i^2\}$ de tal forma que $d(l_i) \leq 3$ (note que cada literal aparece em no máximo 2 cláusulas e temos 2 vértices representando cada literal). Analogamente, se C_j contém \bar{X}_i crie uma aresta entre c_j e um vértice $\bar{l}_i \in \{\bar{x}_i^1, \bar{x}_i^2\}$ tal que $d(\bar{l}_i) \leq 3$ após a adição da aresta. Na Figura 5 pode ser visto um exemplo desta construção onde temos as cláusulas $C_1 = (x_1, x_2)$, $C_2 = (x_1, \bar{x}_2)$ e $C_3 = (\bar{x}_1, x_2)$.

Lema 2.2. *O grafo G é planar bipartido com grau máximo 4.*

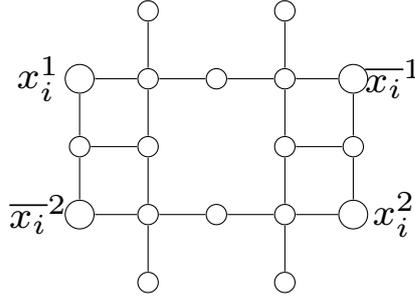


Figura 4 – Gadget da variável X_i

Demonstração. Em primeiro lugar, os gadgets de variável são planares e \mathcal{C} é uma fórmula planar, então G é planar. Em seguida, veja que todos os vértices tem grau máximo 4 pois em \mathcal{C} qualquer cláusula tem no máximo 3 literais e em cada gadget de variável X_i , por construção, temos que $d(l_i) \leq 3$ para todo $l_i \in \{x_i^1, x_i^2, \bar{x}_i^1, \bar{x}_i^2\}$ e $d(v) \leq 4$ para todo $v \neq l_i$. Consequentemente G possui grau máximo 4. Por fim, sejam $\{y_i^1, y_i^2, y_i^3, y_i^4, y_i^5, y_i^6, y_i^7, y_i^8\}$ os vértices no gadget da variável X_i que não são adjacentes a $l_i \in \{x_i^1, x_i^2, \bar{x}_i^1, \bar{x}_i^2\}$. Podemos particionar os vértices de G em dois conjuntos disjuntos S_1 e S_2 onde, sem perda de generalidade, $\{x_i^1, x_i^2, \bar{x}_i^1, \bar{x}_i^2, y_i^1, y_i^2, y_i^3, y_i^4, y_i^5, y_i^6, y_i^7, y_i^8\} \subseteq S_1$ para todo gadget de variável X_i e $\{V(G) \setminus S_1\} \subseteq S_2$ e com isso, concluímos que G é bipartido pois não existe $uv \in E(G)$ tal que $u, v \in S_k$ para $k \in \{1, 2\}$. \square

Não é difícil verificar que qualquer FDR de G tem peso pelo menos 10 em qualquer gadget de variável (note que, para dominar os vértices menores na Figura 4, precisamos de peso pelo menos 10). Além disso, não existe FDR de G com peso 10 em um gadget de variável com valor 2 em vértices x_i^1 ou x_i^2 e valor 2 em vértices \bar{x}_i^1 ou \bar{x}_i^2 .

Com isso, podemos considerar apenas dois tipos diferentes de função de dominação romana R_i e \bar{R}_i (chamadas de *boas*) em um gadget de variável X_i na prova a seguir: uma com $R_i(x_i^1) = R_i(x_i^2) = 2$ e $R_i(\bar{x}_i^1) = R_i(\bar{x}_i^2) = 0$, e a outra com $\bar{R}_i(\bar{x}_i^1) = \bar{R}_i(\bar{x}_i^2) = 2$ e $\bar{R}_i(x_i^1) = \bar{R}_i(x_i^2) = 0$.

Suponha que existe uma atribuição satisfazível verdadeira para \mathcal{C} . Então seja R a função de dominação romana obtida da atribuição onde $R(c_j) = 0$ para cada vértice de cláusula c_j e o valor dos vértices de cada gadget de variável X_i é o mesmo de R_i , se X_i é verdadeira, ou \bar{R}_i , caso contrário. Como \mathcal{C} possui uma atribuição satisfazível verdadeira, então para todo vértice de cláusula c_j existe pelo menos um vértice l_i em um gadget de variável X_i que é vizinho de c_j de tal forma que $R(l_i) = 2$. Com isso R tem peso $\omega(R) = 10k$. Logo $\gamma_R(G) \leq 10k$.

Por outro lado, se G tem uma função de dominação romana R com peso $\omega(R) \leq 10k$, então, como todo gadget de variável têm peso pelo menos 10, para qualquer vértice de cláusula c_j , temos que $R(c_j) = 0$ e C_j é adjacente a um vértice l_i de um gadget de variável com $R(l_i) = 2$. Então, atribuindo verdadeiro para X_i , se o gadget de variável x_i

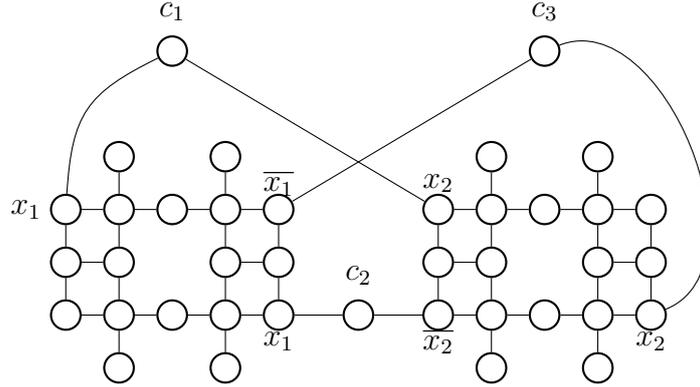


Figura 5 – Redução do Planar 3-SAT: $C_1 = (x_1, x_2), C_2 = (x_1, \bar{x}_2), C_3 = (\bar{x}_1, x_2)$

usa a função de dominação romana R_i , ou falso, caso contrário, todas as cláusulas são satisfeitas. \square

Teorema 2.3. *Computar o número de dominação romana é NP-difícil para subgrafos induzidos de grids.*

Demonstração. Dado uma fórmula planar restrita $\mathcal{C} = \{C_1, \dots, C_m\}$ com k variáveis, seja G o grafo bipartido planar com grau máximo 4 do Teorema 2.1. Sabemos que todo grafo planar com grau máximo 4 tem uma representação na grid tal que as arestas são descritas por segmentos (horizontais e verticais) das linhas no grafo grid. Tal representação pode ser construída em *low-order-polynomial time* [27]. Como o gadget de variável pode ser representado diretamente na grid sem alterações, a substituição será feita apenas nas arestas entre vértices de cláusula e vértices de literal por caminhos no grafo grid. Além disso, através de pequenos desvios (ilustrados na Figura 6) podemos garantir que o tamanho desses caminhos representando arestas entre vértices de cláusula e vértices de literal tem um tamanho múltiplo de 3 (número de vértices).

Note que qualquer dominação romana de um caminho de tamanho $3p$ tem peso pelo menos $2p$. Assim, denotamos por L_j a quantidade de vértices nos segmentos de c_j para os gadgets de variáveis em G dividido por 3. Isto é, se existem $3p$ vértices entre c_j e os gadgets de variável então $L_j = p$. Iremos fazer uma redução do PLANAR 3-SAT onde obtemos uma atribuição de valores satisfazível se e somente se

$$\gamma_R(G) \leq 10k + 2 \cdot \sum_{j \in \{1, \dots, m\}} L_j$$

Considere $L = \{l_1, l_2, \dots, l_{3p}\}$ como sendo um caminho entre um vértice de gadget x_i e um vértice de cláusula c_j , onde $p \in \mathbb{N}^+$, l_1 é vizinho de x_i , e l_{3p} é vizinho de c_j . Como visto no Teorema 2.1, só é preciso utilizar dois tipos diferentes de função de dominação romana R_i e \bar{R}_i para um gadget de variável x_i , e de maneira similar, na nossa prova só precisamos considerar dois tipos diferentes de função de dominação romana F_1 e F_2 em L .

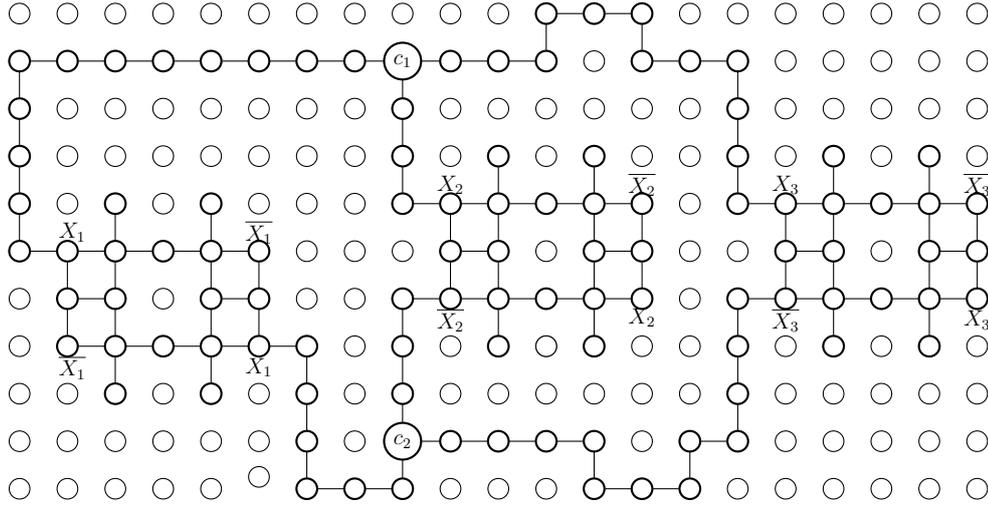


Figura 6 – Redução do PLANAR 3-SAT: $C_1 = (x_1 \vee x_2 \vee x_3)$, $C_2 = (x_1 \vee \overline{x_2} \vee \overline{x_3})$

$$F_1(l_q) = \begin{cases} 2, & \text{se } q \bmod 3 \equiv 0 \\ 0, & \text{c.c.} \end{cases} \quad F_2(l_q) = \begin{cases} 2, & \text{se } q \bmod 3 \equiv 2 \\ 0, & \text{c.c.} \end{cases}$$

Na Figura 7 mostramos um exemplo de F_1 e F_2 no qual um vértice v com cor preta significa $f(v) = 2$.

Suponha que existe uma atribuição satisfazível verdadeira para \mathcal{C} . Então seja R a função de dominação romana obtida da atribuição onde $R(c_j) = 0$ para cada vértice de cláusula c_j e o valor dos vértices de cada gadget de variável x_i é o mesmo de R_i , se x_i é verdadeira, ou $\overline{R_i}$, caso contrário. Dado que x_i está ligada a um vértice de cláusula c_j por um segmento L , dê o mesmo valor da função F_1 para L se x_i é verdadeira, e utilize F_2 caso contrário.

Perceba que, ao usar F_1 , o primeiro vértice de L pode ter valor 0, pois $R(x_i) = 2$; já o vértice c_j pode ter valor 0 pois o último vértice de L possui valor 2. E no caso onde escolhermos o valor de F_2 , os vértices definidos por l_q tal que $q \bmod 3 \equiv 0$ ou $q \bmod 3 \equiv 1$ são vizinhos de um vértice l_s , onde $s \bmod 3 \equiv 2$, o que implica que l_s possui valor 2. Neste último caso nenhum vértice de L interfere nos valores de x_i ou c_j .

Como \mathcal{C} possui uma atribuição satisfazível verdadeira, então para todo vértice de cláusula c_j existe pelo menos um vértice v_i em um gadget de variável x_i que possui um segmento L (com valores definidos por F_1) ligado a c_j de tal forma que $R(v_i) = 2 \Rightarrow R(c_j) = 0$. Com isso R tem peso $\omega(R) \leq 10k + 2 \cdot \sum_{j \in \{1, \dots, m\}} L_j$.

Por outro lado, se G tem uma função de dominação romana R com peso $\omega(R) \leq 10k + 2 \cdot \sum_{j \in \{1, \dots, m\}} L_j$, então, como todo gadget de variável têm peso pelo menos 10 e todo caminho de tamanho $3p$ tem peso pelo menos $2p$, para qualquer vértice de cláusula c_j , temos que $R(c_j) = 0$ e c_j é adjacente a um vértice l_q de um segmento L com $R(l_q) = 2$. Isso implica dizer que foram utilizados os mesmos valores de F_1 em L , caso contrário, teríamos

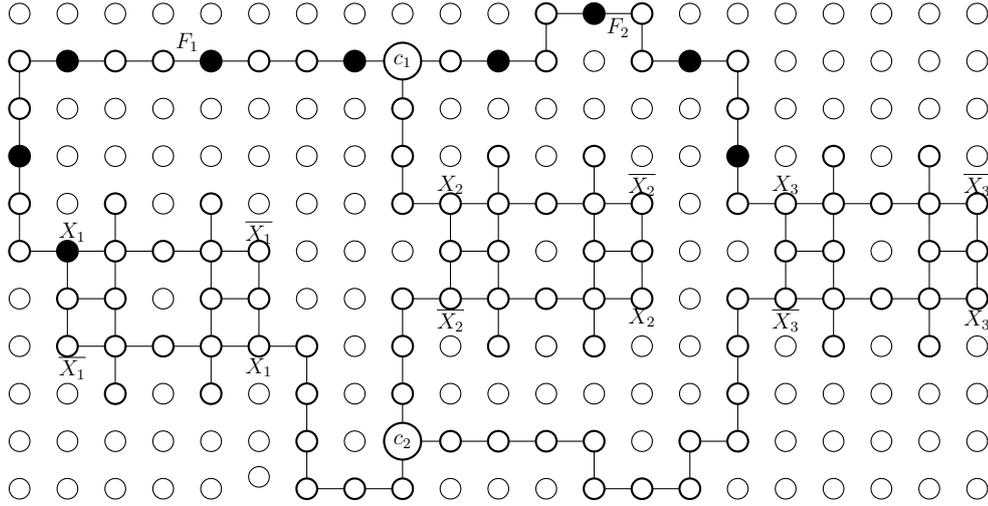


Figura 7 – Redução do PLANAR 3-SAT com exemplo de F_1 e F_2 .

$\omega(R) > 10k + 2 \cdot \sum_{j \in \{1, \dots, m\}} L_j$. Com isso, existe l_1 em L tal que $R(l_1) = 0$, e portanto l_1 é adjacente a um vértice v_i de um gadget de variável com $R(v_i) = 2$. Então, atribuindo verdadeiro para x_i , se o gadget de variável x_i usa a função de dominação romana R_i , ou falso, caso contrário, todas as cláusulas são satisfeitas.

Com isso concluímos que o problema do número de dominação Romana é NP-difícil em subgrafos induzidos de grid. \square

Teorema 2.4. *Determinar o número de dominação romana é APX-difícil mesmo em grafos bipartidos com grau máximo 4.*

Demonstração. O resultado deste teorema segue de uma L-redução do Max-2-Sat-3, que é o problema Max-2-Sat com fórmulas restritas. Assim,

Definição.

MAXIMUM 2-SATISFIABILITY (Max-2-Sat).

INSTÂNCIA: Uma coleção de cláusulas $\mathcal{C} = \{C_1, \dots, C_m\}$ de um conjunto finito de variáveis $X = \{X_1, \dots, X_k\}$ tal que $|C_i| = 2$ para $1 \leq i \leq m$.

PERGUNTA: Qual atribuição de valores de X satisfaz o maior número de cláusulas de \mathcal{C} ?

COMENTÁRIO: Nas instâncias de Max-2-Sat, cada cláusula C_j possui exatamente dois literais.

Sabe-se que o Max-2-Sat-3 (que é o Max-2-Sat em fórmulas restritas) é APX-Completo [20].

A redução é feita como mostrado no Teorema 2.1. Seja R uma dominação romana boa. Podemos afirmar que $\omega(R) = 10k + \sigma(R)$, onde $\sigma(R)$ é o número de vértices de cláusula com valor 1 em R . Seja $g(R)$ a atribuição das variáveis tal que X_i é verdadeiro,

se $R(X_i^1) = R(X_i^2) = 2$, e falso, caso contrário. Logo, $g(R)$ satisfaz $m - \sigma(R)$ cláusulas e, para qualquer atribuição satisfazendo ℓ cláusulas, podemos obter uma dominação Romana R' com $\omega(R') = 10k + (m - \ell)$. Seja ℓ_{opt} o valor máximo do número de cláusulas satisfeitas. Sabemos que $\ell_{opt} \geq m/2$. Isto implica que existe uma dominação romana boa R_{opt} onde

$$\begin{aligned} \omega(R_{opt}) &= 10k + (m - \ell_{opt}) \leq 20m + (m - \ell_{opt}) \leq 21m - \ell_{opt} \leq 42\ell_{opt} - \ell_{opt} \\ \omega(R_{opt}) &\leq 41\ell_{opt} \end{aligned}$$

pois, por definição $k \leq 2m$.

Além disso, $\omega(R) - \omega(R_{opt}) = \ell_{opt} - (m - \sigma(R))$. Com isto, concluímos que essa redução é uma L -redução com parametros $\alpha = 41$ e $\beta = 1$. \square

2.2 Tratabilidade por Parâmetro Fixo

Relembrando a Subseção 1.4.3, dizemos que o problema de dominação romana é FPT em tempo $O(t(n))$ em uma classe de grafo \mathcal{C} se, para qualquer p fixo, decidir se o número de dominação romana é no máximo p pode ser resolvido em tempo $O(t(n))$ para qualquer grafo em \mathcal{C} .

A largura local em árvore (*local-treewidth*) [28] de um grafo G é dado como uma função $ltw_G : \mathbb{N} \rightarrow \mathbb{N}$ que associa com qualquer $r \in \mathbb{N}$ a largura em árvore (*treewidth*) máxima de uma r -vizinhança em G . Isto é, $ltw_G(r) = \max_{v \in V(G)} \{tw(G[N_r(v)])\}$, onde $N_r(v)$ é o conjunto de vértices a distância máxima r de v . Dizemos que uma classe de grafo \mathcal{C} tem largura local em árvore (*local-treewidth*) limitada se existe uma função $f_{\mathcal{C}} : \mathbb{N} \rightarrow \mathbb{N}$ tal que, para todo $G \in \mathcal{C}$ e $r \in \mathbb{N}$, $ltw_G(r) \leq f_{\mathcal{C}}(r)$. Sabemos que grafos com genus limitado ou grau máximo limitado possuem largura local em árvore (*local-treewidth*) limitada [28]. Em particular, um grafo com grau máximo Δ tem $ltw_G(r) \leq \Delta^r$ e um grafo planar tem $ltw_G(r) \leq 3r - 1$ [29].

Em 2008, foi provado que o número de dominação romana pode ser obtido em tempo linear em grafos com distância hereditária usando o Teorema de Courcelle [30], baseado em lógica monádica de segunda ordem, explicada a seguir.

Fórmulas lógicas de *1º ordem* são obtidas a partir de fórmulas de lógica proposicional com a inclusão de quantificadores (\exists, \forall) sobre as variáveis. Por exemplo, a fórmula $Clique_k$ abaixo é de *1º ordem* e representa o problema Clique de decisão:

$$Clique_k := \exists x_1, \dots, \exists x_k \left(\bigwedge_{1 \leq i < j \leq k} x_i \neq x_j \wedge \bigwedge_{1 \leq i < j \leq k} E x_i x_j \right),$$

onde as variáveis x_1, \dots, x_k representam vértices e a relação $E x_i x_j$ indica se existe uma aresta entre os vértices x_i e x_j .

Fórmulas *monádicas de 2º ordem* são semelhantes as de *1º ordem*, mas permitem quantificação em conjuntos de elementos. Por exemplo, a fórmula $Color_k$ abaixo é monádica

de 2º ordem e representa o problema da k -coloração:

$$\begin{aligned} Color_k := \exists X_1, \dots, X_k \forall x, y \left(\bigvee_{i \in [k]} X_i x \wedge \bigwedge_{1 \leq i < j \leq k} \neg(X_i x \wedge X_j x) \right. \\ \left. \wedge \bigwedge_{i \in [k]} ((X_i x \wedge X_i y) \rightarrow \neg E x y) \right), \end{aligned}$$

onde a variável X_i representa o conjunto de vértices com a cor i e a relação $X_i x$ indica se $x \in X_i$.

O Teorema de Frick-Grohe [23] diz que todo problema de decisão que pode ser descrito por uma fórmula φ de 1º ordem possui algoritmo FPT em tempo $f(|\varphi|) \cdot (n+m)^2$ em grafos com largura local em árvore limitada, para alguma função computável f . O Teorema de Courcelle [23] diz que todo problema de decisão que pode ser descrito por uma fórmula φ monádica de 2º ordem é FPT no parâmetro $tw(G) + |\varphi|$.

Nesta seção, mostramos que o problema de decisão do número de dominação romana é de fato expressável em lógica de primeira ordem e, com a aplicação do Teorema de Frick e Grohe [31], provamos o seguinte:

Teorema 2.5. *Dominação romana é FPT em tempo $O((m+n)^2)$ para grafos com largura local em árvore limitada.*

Demonstração. Seja G um grafo. Dados inteiros k e ℓ , seja $Roman_{k,\ell}$ a seguinte expressão de primeira ordem, a qual é verdadeira se e somente se o grafo G possui uma dominação romana com k vértices com valor 2 e ℓ vértices com valor 1:

$$Roman_{k,\ell} := \exists x_1, \dots, x_{k+\ell} \left(\bigwedge_{1 \leq i < j \leq k+\ell} x_i \neq x_j \wedge \forall y \bigvee_{i \in \{1, \dots, k\}} E x_i y \vee \bigvee_{i \in \{1, \dots, k+\ell\}} (x_i = y) \right)$$

Agora seja $Roman_p$ a seguinte expressão de primeira ordem, a qual é verdadeira se e somente se G tem uma dominação romana com peso p :

$$Roman_p := \bigvee_{0 \leq k \leq \lfloor p/2 \rfloor} Roman_{k, p-2k}$$

Assim, o problema de decisão do número de dominação romana é expressável em lógica de primeira ordem. Além disso, $Roman_p$ usa no máximo $p^2/2$ variáveis e com isso, o tamanho da expressão $Roman_p$ está em função de p . Podemos então aplicar o teorema de Frick-Grohe (veja [31]) para provar que o problema de dominação romana é FPT em tempo $O((m+n)^2)$ em grafos com largura local em árvore limitada. \square

O teorema acima prova então a existência de algoritmos FPT para o problema da dominação romana em grafos com grau máximo limitado ou com genus limitado, o que inclui os grafos planares.

Influenciados por esse resultado, obtivemos um algoritmo FPT específico para o problema da dominação romana em grafos com grau máximo limitado.

Teorema 2.6. *Dominação romana é FPT em tempo $O((\Delta + 1)^k(m + n))$ para grafos com grau máximo Δ .*

Demonstração. Um grafo preto e branco consiste de um grafo tal que todo vértice está colorido com a cor preta ou branca. Vértices pretos são aqueles que ainda precisam ser dominados, enquanto vértices brancos são considerados já dominados.

Nosso algoritmo será através de uma árvore de busca. Cada nó da árvore de busca tem associado um grafo preto e branco e um parâmetro. O grafo da raiz é o grafo original com todos os vértices pretos e o parâmetro da raiz é o parâmetro k original. Uma folha é um nó cujo grafo associado tem todos seus vértices brancos ou cujo parâmetro é zero.

Seja h um nó não-folha da árvore de busca. Ramificamos h de acordo com um vértice preto v com menor grau no grafo associado ao nó h . Seja $N[v] = \{u_1, \dots, u_\ell\}$, onde $\ell = |N[v]|$. Com isso, o nó h terá $\ell + 1$ nós filhos h_0, h_1, \dots, h_ℓ na árvore de busca. No nó filho h_0 , atribui-se $f(v) = 1$, remove-se v do grafo associado e diminui-se o valor do parâmetro em 1. No nó filho h_i ($1 \leq i \leq \ell$), atribui-se $f(u_i) = 2$, remove-se u_i do grafo associado, diminui-se o valor do parâmetro em 2 e colorem-se todos os vértices em $N(u_i)$ de branco.

Se alguma folha tem todos os vértices brancos, retorne SIM. Caso contrário, retorne NÃO. Note que a altura da árvore é menor ou igual a k e que cada nó tem no máximo $\Delta + 1$ filhos. Portanto, a árvore tem no máximo $(\Delta + 1)^k$ nós e, como em cada nó, leva-se tempo $O(m + n)$, temos tempo total $O((\Delta + 1)^k(m + n))$. \square

Em [22], foi provado que o problema de dominação romana é FPT em grafos planares usando também o método da árvore de busca.

3 Resultados Obtidos em Digrafos

Um *digrafo* $D = (V(D), A(D))$ é constituído de um conjunto de vértices $V(D)$ e um conjunto de arcos $A(D)$, onde cada arco corresponde a um par ordenado de vértices. Se uv é um arco de D , então escrevemos $u \rightarrow v$, e dizemos que v é um *vizinho de saída* de u e u é um *vizinho de entrada* de v . Escrevemos $d_D^+(v)$ para o *grau de saída* de um vértice $v \in V$ e $d_D^-(v)$ para o seu *grau de entrada*. O mínimo e máximo grau de entrada e mínimo e máximo grau de saída de D são indicados por $\delta^- = \delta^-(D)$, $\Delta^- = \Delta^-(D)$, $\delta^+ = \delta^+(D)$, $\Delta^+ = \Delta^+(D)$, respectivamente. Para um vértice $v \in V(D)$, denotamos o conjunto de vizinhos de entrada e vizinhos de saída de v por $N^-(v)$ e $N^+(v)$, nesta ordem. Definimos $N^-[v] = N^-(v) \cup \{v\}$ e $N^+[v] = N^+(v) \cup \{v\}$ como a *vizinhança fechada de entrada* e *vizinhança fechada de saída* de $v \in V(D)$. Se $X \subseteq V(D)$, então $D[X]$ é o *subdigrafo induzido* por X . O *grafo subjacente* de um diágrafo D é o grafo G obtido substituindo cada arco de D por uma aresta não direcionada correspondente, ou seja, $V(G) = V(D)$ e $E(G) = \{uv | u \rightarrow v \in E(D) \text{ ou } v \rightarrow u \in E(D)\}$. Seja D um digrafo finito e simples com conjunto de vértices V e conjunto de arcos A , dizemos que D é um *grafo orientado* se ele não possui ciclos direcionados de tamanho 2.

Neste capítulo, nós introduzimos o conceito de *dominação romana em digrafos*. Dado um digrafo D , uma função $f : V(D) \rightarrow \{0, 1, 2\}$ é uma *Função de Dominação Romana (FDR)* se todo vértice $u \in V$ para o qual $f(u) = 0$ possui um vizinho de entrada $v \in V$ tal que $f(v) = 2$. O peso de uma FDR f é o valor $\omega(f) = \sum_{u \in V} f(u)$ e o *número de dominação romana*, denotado por $\gamma_R(D)$, é o peso mínimo de uma FDR em D . Definimos ainda as partições ordenadas (V_0, V_1, V_2) de $V(D)$ induzidas por f , onde $V_i = \{v \in V(D) | f(v) = i\}$, para $i = 0, 1, 2$. Como existe uma correspondência 1-1 entre $f : V(D) \rightarrow \{0, 1, 2\}$ e as partições ordenadas (V_0, V_1, V_2) , também denotamos $f = (V_0, V_1, V_2)$. Nessa representação, o seu peso é $\omega(f) = |V_1| + 2|V_2|$. Uma FDR f é chamada de $\gamma_R(D)$ -*função* se $\omega(f) = \gamma_R(D)$.

3.1 Dominação Romana de Digrafos Planares

Nesta seção, construímos um algoritmo de árvore de busca que coloca o problema da dominação romana restrito a digrafos planares em FPT. Algoritmos bastante semelhantes foram utilizados anteriormente para o problema da dominação em grafos planares [32] e para o problema da dominação romana também em grafos planares [22].

Aqui nós gostaríamos de introduzir a noção de um *digrafo preto e branco*, onde o conjunto de vértices $V(D)$ é dividido em dois conjuntos disjuntos B e W (*Black* e *White*), ou seja, $V(D) = B \uplus W$. Vértices pretos são aqueles que ainda precisam ser dominados,

enquanto vértices brancos são considerados já dominados. É importante notar que ainda é possível colocar o valor 2 em um vértice branco, para dominar os seus vértices vizinhos.

Cada nó da árvore de busca tem associado um digrafo preto e branco e um parâmetro. O digrafo da raiz é o digrafo original com todos os vértices pretos e o parâmetro da raiz é o parâmetro k original. Uma folha é um nó cujo digrafo associado tem todos seus vértices brancos ou cujo parâmetro é zero.

Seja h um nó da árvore de busca. Ramificamos h de acordo com um vértice preto v com menor grau de entrada no digrafo associado ao nó h . Seja $N^-[v] = \{u_1, \dots, u_\ell\}$, onde $\ell = |N^-[v]|$. Com isso, o nó h terá $\ell + 1$ nós filhos h_0, h_1, \dots, h_ℓ na árvore de busca. No nó filho h_0 , atribui-se $f(v) = 1$, remove-se v do digrafo associado e diminui-se o valor do parâmetro em 1. No nó filho h_i ($1 \leq i \leq \ell$), atribui-se $f(u_i) = 2$, remove-se u_i do digrafo associado, diminui-se o valor do parâmetro em 2 e colorem-se todos os vértices em $N^+(u_i)$ de branco.

Além disso, antes de ramificar um nó h , iremos aplicar as regras de redução abaixo em seu digrafo associado. Dizemos que um digrafo é *reduzido* se nenhuma dessas regras pode ser aplicada a ele.

- (R1) Se existe um vértice u com um vizinho de saída branco u_1 , então delete o arco $u \rightarrow u_1$.
- (R2) Delete um vértice branco u onde $N^+(u) \leq 2$.

Se (D, k) é uma instância de um problema e uma instância (D', k') é obtida a partir de (D, k) aplicando uma das regras de redução, então (D, k) tem uma solução se e somente se (D', k') tem uma solução. Nesse caso dizemos que o conjunto de regras de redução do problema é *sound*.

Lema 3.1. *As regras de redução (R1) e (R2) são sound.*

Demonstração. Seja $D = (B \uplus W, E)$ o digrafo preto e branco “original” e $D' = (B' \uplus W', E')$ o digrafo obtido aplicando uma vez a regra de redução correspondente. Vamos considerar as diferentes regras de redução, uma por uma:

- (R1) Seja D um digrafo que contém um vértice u com vizinho de saída branco u_1 . Uma vez que os valores de u e u_1 são independentes um do outro em D , então $R \subseteq B \uplus W$ é uma função dominação romana em D , se e somente se R é um função dominação romana em D' .
- (R2) Seja D um digrafo que contém um vértice branco u com $N^+(u) \leq 2$ e seja $R \subseteq B \uplus W$ uma função de dominação romana em D . Podemos ver que R restrito a D' é uma função dominação romana de D' se $R(u) = 0$. Como $u \in W$, $\exists w \in N^-(u)$ tal que $R(w) = 2$, então precisamos considerar apenas dois casos, $R(u) = 0$ e

$R(u) = 2$. Suponha que $N^+(u) \leq 1$ e $R(u) = 2$, é possível obter uma função dominação romana de menor custo redefinindo $R(u) = 0$ e se necessário, redefinindo $R(v) = 1$, tal que $v \in N^+(u)$. No caso em que $N^+(u) = 2$ considere $v_1, v_2 \in N^+(u)$ e uma vez que $R(u) = 2$, então $R(v_1) + R(v_2) \leq 2$ (caso contrário, redefinindo $R(u) = 0$ uma função de dominação romana menor pode ser obtida). No entanto, se $R(v_1) + R(v_2) \leq 2$, então, ou $R(v_1) = 0$ ou $R(v_2) = 0$ ou $R(v_1) = R(v_2) = 1$. Se $R(v_1) = R(v_2) = 1$, então podemos obter uma função de dominação romana menor redefinindo $R(u) = 0$. Por fim, suponha $R(v_1) = 0$ (o caso em que $R(v_2) = 0$ é análogo), neste caso, nós consideramos os três valores possíveis para $R(v_2)$. Se $R(v_2) = 2$ ou $R(v_2) = 1$ é possível obter uma função dominação romana menor com os valores $R(u) = 0$ e $R(v_1) = 1$. Se $R(v_2) = 0$ podemos obter uma outra função de dominação romana com os valores $R(u) = 0$, $R(v_1) = 1$ e $R(v_2) = 1$, sem alterar o peso de R . Assim, após estas alterações, R é uma função de dominação romana onde $R(u) = 0$ e concluímos então que $R(D) = R(D')$. Por outro lado, seja $R' \subseteq B \uplus W$ uma função de dominação romana em D' , como u é um vértice branco, então R' pode ser estendida para uma função dominação romana em D definindo $R'(u) = 0$. E com isso, $R'(D') = R'(D)$.

□

Lema 3.2. *Seja $D = (B \uplus W, E)$ um digrafo preto e branco. Se D é reduzido, então os vértices brancos formam um conjunto independente.*

Demonstração. O resultado segue diretamente da regra de redução (R1). □

Dados $A, B \subseteq V(D)$, denote por $E(A, B)$ o conjunto dos arcos de A para B . Afirmamos que o seguinte é verdade:

Afirmção 3.1.1. *Seja W_3 (resp. W_2, W_1) o conjunto dos vértices brancos com pelo menos 3 (resp. exatamente 2, exatamente 1) vizinhos de saída em B . Então:*

$$(i) |E(W_1, B)| = 0;$$

$$(ii) |E(W_2, B)| = 0;$$

$$(iii) |E(W_3, B)| < 6|B|.$$

Demonstração. Pela regra (R2), não existe $w \in W$, com exatamente um vértice $b \in B$ tal que $N^+(w) = \{b\}$ (ou exatamente dois vértices $b_1, b_2 \in B$ tal que $N^+(w) = \{b_1, b_2\}$). Portanto $W_1 = W_2 = \emptyset$, e com isso $|E(W_1, B)| = 0$ e $|E(W_2, B)| = 0$. Isto prova (i) e (ii).

Seja H o digrafo bipartido que compreende B, W_3 e todos os arcos de W_3 para B . Então $|E(W_3, B)| = E(H) \geq 3|W_3|$. Observe que H é planar (pois é um subdigrafo de D) e bipartido. Portanto seu grau médio é menor que 4. Assim, temos:

$$4 > \frac{2|E(H)|}{|W_3| + |B|} \geq \frac{6|W_3|}{|W_3| + |B|}.$$

A partir da desigualdade exterior, $6|W_3| < 4(|W_3| + |B|)$, então $|W_3| < 2|B|$. E da desigualdade restante

$$2|E(H)| < 4(|W_3| + |B|) < 12|B|$$

Portanto $|E(H)| < 6|B|$. Isto prova (iii). \square

Teorema 3.3. *Se $D = (B \uplus W, E)$ é um digrafo planar preto e branco reduzido, então existe um vértice preto $u \in B$ com $d^-(u) < 9$*

Demonstração. Usando Afirmação 3.1.1, não é difícil ver que

$$\sum_{v \in B} d_D^-(v) = \sum_{v \in B} d_B^-(v) + E(W, B) = \sum_{v \in B} d_B^-(v) + E(W_3, B).$$

Mas pela regra (R1), não há nenhum arco de B para W , então $d_D^+(v) = d_B^+(v)$ para todo $v \in B$. Assim, $\sum_{v \in B} d_D^+(v) = \sum_{v \in B} d_B^+(v) = \sum_{v \in B} d_B^-(v)$.

Deste modo

$$\sum_{v \in B} d_D^-(v) = \sum_{v \in B} d_D^+(v) + E(W_3, B) < \sum_{v \in B} (d_D^+(v) + 6)$$

por (iii).

Já que D é um digrafo planar preto e branco reduzido, então $D[B]$ é um digrafo planar também. Portanto

$$\sum_{v \in B} d_D^+(v) \leq 3|B| - 6 < 3|B|$$

O que quer dizer que

$$\sum_{v \in B} d_D^-(v) < \sum_{v \in B} (d_D^+(v) + 6) = \sum_{v \in B} d_D^+(v) + 6|B| < 3|B| + 6|B| = 9|B|$$

Consequentemente, há um vértice u em B tal que $d^-(u) < 9$. \square

Um algoritmo de árvore de busca simples iria agora escolher um vértice preto v de menor grau de entrada e ramificar de acordo com se $f(v) = 1$ ou $f(v) = 2$ para algum $u \in N^-[v]$ (esta ramificação reduz o parâmetro em 2 para cada u). De acordo com o Teorema 3.3, $N^-[v]$ contém no máximo nove vértices. Portanto, resolvendo a correspondente recorrência para o tamanho da árvore de busca $T(k) \leq T(k-1) + 9T(k-2)$ obtemos o seguinte:

Teorema 3.4. *O problema de dominação romana em digrafos planares pode ser resolvido em tempo $\mathcal{O}^*(3.5414^k)$.* \square

3.2 Dominação Romana em Torneios Bipartidos

Um cografo (ou *complement-reducible graph*, ou grafo livre de P_4) é um grafo definido pelos seguintes critérios:

1. K_1 é um cografo,
2. Se X é um cografo, seu complemento também é um cografo, e
3. Se X e Y são cografos, então a sua união $X \cup Y$ também é um cografo.

[18] provou que o número de dominação romana de um cografo pode ser calculado em tempo linear. Porém, provamos aqui que calcular o número de dominação romana em torneios bipartidos (que são cografos orientados) é NP-Completo.

O problema de decisão da dominação romana em digrafos é definido como:

DOMINAÇÃO ROMANA

INSTÂNCIA: Um digrafo $D = (V, A)$ e um inteiro positivo $k \leq |V|$.

PERGUNTA: D possui uma FDR f com peso $\omega(f) \leq k$?

Para mostrar que DOMINAÇÃO ROMANA é NP-Completo em torneios bipartidos, usamos uma redução do problema 3-SAT.

Teorema 3.5. *DOMINAÇÃO ROMANA é NP-Completo em torneios bipartidos.*

Demonstração. É fácil ver que o problema de decisão está em NP. Para isso, dada uma instância (D, k) , basta receber como certificado uma partição (V_0, V_1, V_2) de $V(D)$ e verificar (em tempo polinomial) se $|V_1| + 2|V_2| \leq k$ e se todo vértice de V_0 possui um vizinho de entrada em V_2 .

Seja I uma instância arbitrária do 3-SAT com o conjunto de cláusulas $\mathcal{C} = \{C_1, \dots, C_m\}$ e conjunto de variáveis $X = \{x_1, \dots, x_n\}$. Criamos um torneio bipartido completo D e uma FDR f tal que I tem uma atribuição satisfazível se e somente se $\omega(f) \leq 2n + 2$.

A partir de \mathcal{C} construímos um grafo bipartido completo G com dois conjuntos disjuntos S_1 e S_2 tal que $V(G) = S_1 \cup S_2$. Em S_1 , colocamos um vértice para cada literal x_i e \bar{x}_i . Em S_2 , criamos um vértice isolado w , um vértice para cada cláusula c_j e um conjunto de $2n + 3$ vértices Y_i para cada variável x_i . Colocamos uma aresta entre dois vértices u e v se e somente se $u \in S_1$ e $v \in S_2$ como mostrado na Figura 8.

Não é difícil ver que G é um grafo bipartido completo. Para construir o digrafo D , usamos a seguinte orientação das arestas de G :

- Todos os arcos de w vão de w para x , para todo $x \in S_1$.
- Orientamos a aresta de um vértice de literal $l_i \in \{x_i, \bar{x}_i\}$ para um vértice de cláusula c_j se e somente se $l_i \in C_j$ em \mathcal{C} . Caso contrário, orientamos a aresta de c_j para l_i .

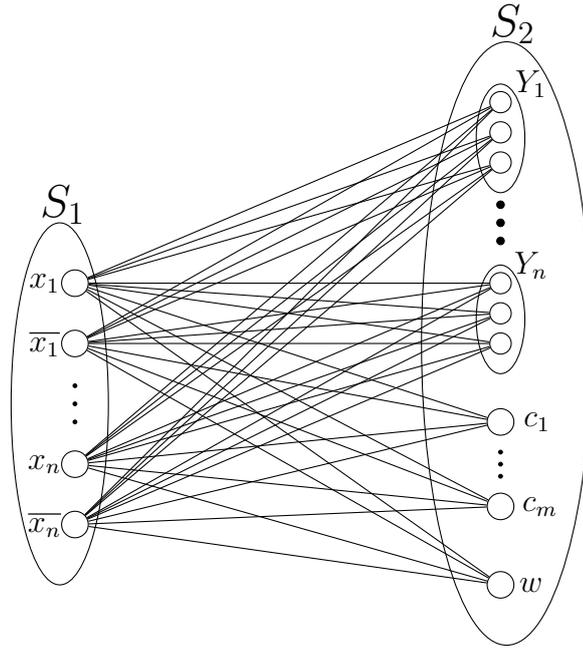


Figura 8 – Grafo Bipartido Completo G .

- Orientamos as arestas dos vértices x_i e \bar{x}_i para todos os vértices de Y_k se e somente se $i = k$. Caso contrário, orientamos as arestas dos vértices de Y_k para x_i e \bar{x}_i .

Afirmção 3.2.1. *A instância de 3-SAT tem uma atribuição verdadeira satisfazível se e somente se $\gamma_R(D) \leq 2n + 2$.*

Assuma que a instância I do 3-SAT tenha uma atribuição verdadeira satisfazível ϕ . Fixe $f(w) = 2$ e para cada variável x_i , coloque $f(x_i) = 2$ se $\phi(x_i) = VERDADEIRO$, caso contrário (se $\phi(x_i) = FALSO$) defina $f(\bar{x}_i) = 2$. Para todos os outros vértices v em D , coloque $f(v) = 0$. É fácil ver que a função f assim definida é uma função de dominação romana de peso $2n + 2$, uma vez que o valor de x_i ou de \bar{x}_i é igual a 2 e ambos os vértices dominam Y_i . Além disso, todos os vértices em S_1 são dominados por w , pois $f(w) = 2$. Por fim, cada vértice c_j tem pelo menos um vizinho de entrada $x \in S_1$ tal que $f(x) = 2$ pelo fato de ϕ atribuir pelo menos um literal para $VERDADEIRO$ em toda cláusula C_j .

Por outro lado, suponha que D tem uma função de dominação romana f tal que $\omega(f) \leq 2n + 2$. Para cada Y_i , temos que $f(x_i) = 2$ ou $f(\bar{x}_i) = 2$, caso contrário (se $f(x_i) < 2$ e $f(\bar{x}_i) < 2$), como $|Y_i| = 2n + 3$ e nenhum outro vértice é vizinho de entrada dos vértices em Y_i então $\omega(f) \geq 2n + 3$. Portanto, para todo $1 \leq i \leq n$ e todo $y \in Y_i$, temos $f(y) = 0$. Perceba que $d^-(w) = 0$, isto implica que $f(w) > 0$. Suponha $f(w) = 1$, para dominar todos os vértices em $S' = \{v \in S_1 : f(v) \neq 2\}$ ou existe $z \in S_2$ tal que $f(z) = 2$ e $S' \subseteq N^+(z)$ ou $f(v) = 1$ para todo $v \in S'$. Se existir z como mencionado acima, então $\gamma_R(D) \geq 2|S_1 \setminus S'| + f(w) + f(z)$. Claramente $|S_1 \setminus S'| \geq n$, então $\omega(f) \geq 2n + f(w) + f(z) \geq 2n + 3$. Se $f(v) = 1$ para todo $v \in S'$ então $\omega(f) \geq 2|S_1 \setminus S'| + f(w) + |S'|$. Como

$|S_1 \setminus S'| + |S'| = |S_1| = 2n$, podemos afirmar que $\omega(f) \geq 2|S_1 \setminus S'| + f(w) + |S'| \geq 2n + 3$. Portanto, $f(w) = 2$ e f tem peso $2n + 2$. Por conseguinte o conjunto de vértices $v \in S_1$ para o qual $f(v) = 2$ domina o conjunto de vértices de cláusula, ou então, $\omega(f) \geq 2n + 3$. Portanto, a instância dada do 3-SAT tem uma atribuição verdadeira satisfazível ϕ , onde $\phi(x_i) = VERDADEIRO$ se e somente se $f(x_i) = 2$.

□

4 Programação Matemática

4.1 F_1 - *The Set Covering Deployment Problem (SCDP)*

No artigo [10], os autores relacionam o problema de dominação romana com uma classe de problemas bem conhecida que é chamada de problemas de localização.

Usando programação inteira, foi apresentado no artigo mencionado acima, a primeira formulação para o problema de dominação romana definida por *The Set Covering Deployment Problem (SCDP)*. Esta formulação foi derivada de um problema bastante estudado, que é conhecido como *Location Set Covering Problem* ([33], [34] e [35]).

O *Location Set Covering Problem* é definido por uma demanda de nós e locais de instalações elegíveis espalhados numa rede ou em um plano, onde o problema é alocar o menor número de instalações, para que, todos os pontos de demanda tenham pelo menos uma instalação a uma distância padrão.

Na formulação SCDP, cada região (vértice) precisa ser protegida por uma ou mais tropas, ou precisa ser alcançada em um único passo (ou seja, existe uma aresta) por uma região que contém duas tropas. Procuramos o número mínimo de tropas para distribuir de tal forma que todas as regiões ou são protegidas ou são passíveis de serem protegidas (vizinhas de uma região com duas tropas).

$$X_u = \begin{cases} 1, & \text{Se a região } u \text{ contém uma ou duas tropas.} \\ 0, & \text{Caso contrário} \end{cases}$$

$$Y_u = \begin{cases} 1, & \text{Se a região } u \text{ contém duas tropas.} \\ 0, & \text{Caso contrário} \end{cases}$$

Em outras palavras, se $f = (V_0, V_1, V_2)$ é uma função de dominação romana em um grafo G onde os vértices representam regiões e arestas representam suas adjacências, então

$X_u \in \{1, 0\}$: A variável é 1 se a região $u \in V_1$ ou $u \in V_2$ e 0 caso contrário.

$Y_u \in \{1, 0\}$: A variável é 1 se a região $u \in V_2$ e 0 caso contrário.

Assim temos a seguinte formulação:

$$\text{Minimizar } \sum_{u \in V(G)} (X_u + Y_u)$$

$$\text{Sujeito a } X_u + \sum_{v \in N(u)} Y_v \geq 1 \quad \forall u \in V(G) \quad (4.1)$$

$$Y_u - X_u \leq 0 \quad \forall u \in V(G) \quad (4.2)$$

$$Y_u, X_u \in \{0, 1\} \quad \forall u \in V(G) \quad (4.3)$$

As restrições (4.1) garantem que toda região precisa ser protegida ou passível de ser protegida. Se $X_u = 1$, a região u é protegida independente das tropas em suas regiões vizinhas. Se $X_u = 0$, então pelo menos uma das suas regiões vizinhas deve conter duas tropas. Também é possível que a região u possua uma tropa defendendo-a e ainda assim exista uma região vizinha que contenha duas tropas (isto não é um problema pois não inviabiliza a possibilidade de pelo menos um desses eventos ocorrer em toda região $u \in V(G)$). As restrições do tipo (4.2) representam que não pode existir duas tropas em uma região u a menos que já exista pelo menos uma tropa em u . Por fim, as restrições do tipo (4.3) são restrições que garantem o valor binário positivo 0 ou 1 para toda região $u \in V(G)$.

A função objetivo equivale ao número de regiões com uma ou duas tropas mais o número de regiões com duas tropas. E este é exatamente o número de tropas alocadas. Minimizar a função objetivo, minimiza o número de tropas alocados satisfazendo as restrições (o que corresponde ao número de dominação romana de G).

4.2 F_2 - Differential de um Grafo

Seja $G = (V, E)$ um grafo, o *differential* de um conjunto de vértices $S \subseteq V$ é definido como $\partial(S) = |B(S)| - |S|$, onde $B(S)$ é o conjunto de vértices em $V \setminus S$ que tem pelo menos um vizinho em S . Mais informações sobre o differential podem ser vistas em [36].

Pelo Teorema 1 do artigo *The differential and the roman domination number of a graph* [36] temos que o número de dominação romana pode ser relacionado com o differential de um grafo G da seguinte maneira:

$$\gamma_R(G) = n - \partial(G),$$

onde $\partial(G) = \max\{\partial(S) : S \subseteq V\}$.

Portanto construímos uma formulação para o número de dominação romana baseada no differential de um grafo, que é apresentada a seguir:

$$\text{Maximizar } \sum_{u \in V(G)} (X_u - Y_u)$$

$$\text{Sujeito a } X_u + Y_u \leq 1 \quad \forall u \in V(G) \quad (4.4)$$

$$X_u - \sum_{v \in N(u)} Y_v \leq 0 \quad \forall u \in V(G) \quad (4.5)$$

$$Y_u, X_u \in \{0, 1\} \quad \forall u \in V(G) \quad (4.6)$$

$$X_u = \begin{cases} 1, & \text{Se } u \in B(S) \\ 0, & \text{Caso contrário} \end{cases}$$

$$Y_u = \begin{cases} 1, & \text{Se } u \in S \\ 0, & \text{Caso contrário} \end{cases}$$

Esta formulação nos fornece $\partial(G)$, pois as restrições do tipo (4.4) nos afirmam que um vértice u não pode estar em $B(S)$ e em S simultaneamente, enquanto que o conjunto de restrições (4.5) garante que se $u \in B(S)$ então ele possui pelo menos um vizinho em S . Assim, obtemos $\partial(G)$ como resultado de nossa formulação, e, conseqüentemente, $n - \partial(G)$ é o resultado para o problema de dominação romana em G .

Perceba que se compararmos essa formulação com uma FDR, os conjuntos S e $B(S)$ são iguais as partições V_2 e V_0 respectivamente.

4.3 F_3 -Tática Binária de Proteção de Grafos

No trabalho de Burger, Villiers e Vuuren [37], intitulado “*A binary programming approach towards achieving effective graph protection*” [37], foi introduzida uma nova formulação onde computar o número de dominação romana de um grafo G é formulado como um programa binário da seguinte maneira:

$$\text{Minimizar } \sum_{u \in V(G)} (X_u + 2Y_u)$$

$$\text{Sujeito a } X_u + Y_u + \sum_{v \in N(u)} Y_v \geq 1 \quad \forall u \in V(G) \quad (4.7)$$

$$X_u + Y_u \leq 1 \quad \forall u \in V(G) \quad (4.8)$$

$$Y_u, X_u \in \{0, 1\} \quad \forall u \in V(G) \quad (4.9)$$

$$X_u = \begin{cases} 1, & \text{Se } u \in V_1 \\ 0, & \text{Caso contrário} \end{cases}$$

$$Y_u = \begin{cases} 1, & \text{Se } u \in V_2 \\ 0, & \text{Caso contrário} \end{cases}$$

O conjunto de restrições (4.7) garante que cada vértice v_i não ocupado (i.e. para o qual $X_i = Y_i = 0$) é dominado por pelo menos um vértice vizinho no conjunto Y (i.e. contendo duas tropas). O conjunto de restrições (4.8) garantem que os conjuntos X e Y são disjuntos (i.e. nenhum vértice pode ser interpretado como sendo ocupado por uma tropa e por duas ao mesmo tempo). E por fim, V_1 e V_2 representam os conjuntos de uma partição ordenada (V_0, V_1, V_2) de G para o número de dominação romana.

4.4 Comparativo Entre as Formulações

Dado um grafo $G = (V, E)$, em uma comparação entre as três formulações apresentadas podemos construir uma análise inicial com relação à quantidade de variáveis e restrições de cada formulação, como pode ser visto na Tabela 2.

Formulação	Variáveis	Restrições
F1	$2 V $	$2 V $
F2	$2 V $	$2 V $
F3	$2 V $	$2 V $

Tabela 2 – Comparativo entre as formulações apresentadas

Fazendo uma comparação mais aprofundada, podemos perceber que as três formulações são equivalentes. Mostraremos a seguir, as mudanças de variáveis necessárias para observar esta equivalência.

- $F_1 \Leftrightarrow F_2$

Substituindo $X_u = 1 - X'_u$ em F_1 obtemos:

$$\begin{aligned} (1 - X'_u) + \sum_{v \in N(u)} Y_v &\geq 1 && \Leftrightarrow && X'_u - \sum_{v \in N(u)} Y_v &\leq 0 \\ Y_u - (1 - X'_u) &\leq 0 && \Leftrightarrow && X'_u + Y_u &\leq 1 \\ 0 &\leq (1 - X'_u) \leq 1 && \Leftrightarrow && 0 &\leq X'_u \leq 1 \end{aligned}$$

Isto mostra que F_1 e F_2 são iguais.

- $F_1 \Leftrightarrow F_3$

Substituindo $X_u = X''_u + Y_u$ em F_1 obtemos:

$$\begin{aligned} X_u + \sum_{v \in N(u)} Y_v &\geq 1 && \Leftrightarrow && X''_u + Y_u + \sum_{v \in N(u)} Y_v &\geq 1 \\ Y_u - (X''_u + Y_u) &\leq 0 && \Leftrightarrow && X''_u &\geq 0 \\ 0 &\leq (X''_u + Y_u) \leq 1 && \Leftrightarrow && X''_u + Y_u &\leq 1 \end{aligned}$$

Provando que F_1 e F_3 são iguais.

Consequentemente, por transitividade, podemos ver que F_1 , F_2 e F_3 são iguais (mesmo tendo interpretações distintas).

Seja t_i o tempo de resolução de uma instância resolvida utilizando a formulação F_i , $i \in \{1, 2, 3\}$. Um estudo feito por análise empírica sobre as formulações, mostrou-nos que usualmente $t_3 \leq t_j, j \in \{1, 2\}$. Com essa informação, nós propomos a seguinte melhoria a ser feita em F_3 :

$$\begin{aligned}
& \text{Minimizar} && \sum_{u \in V(G)} (X_u + 2Y_u) \\
& \text{Sujeito a} && X_u + Y_u + \sum_{v \in N(u)} Y_v \geq 1 && \forall u \in V(G) \\
& && X_u, Y_u \in \{0, 1\} && \forall u \in V(G)
\end{aligned}$$

$$\begin{aligned}
X_u &= \begin{cases} 1, & \text{Se } u \in V_1 \\ 0, & \text{Caso contrário} \end{cases} \\
Y_u &= \begin{cases} 1, & \text{Se } u \in V_2 \\ 0, & \text{Caso contrário} \end{cases}
\end{aligned}$$

Seja F'_3 a formulação apresentada acima, que é uma relaxação daquela definida na Seção 4.3. Na verdade, obtemos o seguinte teorema:

Teorema 4.1. *O valor ótimo da função objetivo de F'_3 é igual ao valor ótimo da função objetivo de F_3 .*

Demonstração. Seja (\bar{x}, \bar{y}) uma solução viável de F'_3 . Defina (\hat{x}, \hat{y}) da seguinte maneira:

$$\begin{aligned}
\hat{y}_i &= \bar{y}_i, \quad i \in V(G) \\
\hat{x}_i &= \begin{cases} \bar{x}_i, & \text{Se } \bar{x}_i + \bar{y}_i \leq 1 \\ 0, & \text{Se } \bar{x}_i + \bar{y}_i = 2 \end{cases} \quad \forall i \in V(G)
\end{aligned}$$

Mostramos agora que (\hat{x}, \hat{y}) é viável para F_3 considerando dois casos para $i \in V(G)$:

1. $\bar{x}_i = 0$ ou $\bar{y}_i = 0$:

$$\begin{aligned}
\hat{x}_i + \hat{y}_i + \sum_{j \in N(i)} \hat{y}_j &= \bar{x}_i + \bar{y}_i + \sum_{j \in N(i)} \bar{y}_j \geq 1 \\
\hat{x}_i + \hat{y}_i &= \bar{x}_i + \bar{y}_i = \max\{\bar{x}_i, \bar{y}_i\} \leq 1,
\end{aligned}$$

Mostrando que (4.7) e (4.8) são satisfeitas.

2. $\bar{x}_i = 1$ e $\bar{y}_i = 1$:

$$\begin{aligned}
\hat{x}_i + \hat{y}_i + \sum_{j \in N(i)} \hat{y}_j &= 0 + 1 + \sum_{j \in N(i)} \bar{y}_j \geq 1 \\
\hat{x}_i + \hat{y}_i &= 0 + 1 \leq 1
\end{aligned}$$

Portanto, seja S'_{F_3} o valor da função objetivo de uma solução ótima (\bar{x}_i, \bar{y}_i) de F'_3 e (\hat{x}_i, \hat{y}_i) sua solução correspondente em F_3 . Seja S_{F_3} o valor da função objetivo de uma solução ótima de F_3 . Então:

$$S_{F_3} \leq \sum_{i \in V(G)} (\hat{x}_i + 2\hat{y}_i) \leq \sum_{i \in V(G)} (\bar{x}_i + 2\bar{y}_i) = S'_{F_3}$$

Por outro lado, como F'_3 é uma relaxação de F_3 , pois apenas removemos a restrição (4.8), podemos concluir que:

$$S'_{F_3} \leq S_{F_3}$$

Ou seja, $S'_{F_3} = S_{F_3}$. □

Veja que o potencial de F'_3 está no fato de possuir apenas $|V|$ restrições (metade de F_3), levando ao uso de menos espaço de memória e à redução do número de operações por iteração de um método de resolução.

4.5 Estudo Poliédrico

Um **poliedro** em \mathbb{R}^n é um conjunto P definido pela interseção de um número finito de semiespaços. Em outras palavras, um subconjunto $P \subseteq \mathbb{R}^n$ é um poliedro se $P = \{x \in \mathbb{R}^n : Ax \leq b\}$, para alguma matriz $A \in \mathbb{R}^{m \times n}$ e algum vetor $b \in \mathbb{R}^m$. A matriz (A, b) é chamada de uma representação do poliedro P e podemos usar $P(A, b)$ para denotar essa representação.

Notação. Dado um poliedro $P(A, b)$, onde $A \in \mathbb{R}^{m \times n}$ e $b \in \mathbb{R}^m$, vamos denotar por:

- $(a^i)^T, b_i$: i -ésima linha de A e i -ésima componente de b
- $M = \{1, \dots, m\}$: Um conjunto de índices de 1 a m das linhas de A
- $M^= = \{i \in M : (a^i)^T x = b_i, \forall x \in P\}$
- $M^< = M \setminus M^= = \{i \in M : (a^i)^T x < b_i, \text{ para algum } x \in P\}$
- $(A^=, b^=)$: submatriz de (A, b) com linhas indexadas por $M^=$
- $(A^<, b^<)$: submatriz de (A, b) com linhas indexadas por $M^<$

Um ponto $x \in P$ é dito um ponto interno de P , denotado por $x \in ir(P)$, se $A^< x < b^<$. Por outro lado, $x \in P$ é ponto interior de P , denotado por $x \in int(P)$, se $Ax < b$.

4.5.1 Dimensão do Poliedro

Proposição 20 (Ferreira e Wakabayashi [38]). *Se $P(A, b) \subseteq \mathbb{R}^n$ e $P \neq \emptyset$ então $\dim(P) = n - \text{posto}(A^\circ)$.*

Demonstração. Da álgebra linear, sabemos que $\dim(\mathcal{N}(A^\circ)) = n - \text{posto}(A^\circ)$. Seja $r = \dim(P)$ e $s = \dim(\mathcal{N}(A^\circ))$. Queremos então mostrar que $r = s$.

$r \leq s$ Como $\dim(P) = r$, então existem x^0, x^1, \dots, x^r vetores afim-independentes em P . Logo, $x^1 - x^0, x^2 - x^0, \dots, x^r - x^0$ são vetores linearmente independentes. E esses vetores pertencem a $\mathcal{N}(A^\circ)$ pois $A^\circ(x^i - x^0) = 0, \forall i \in \{1, 2, \dots, r\}$. Portanto $s \geq r$.

$s \leq r$ Como $P \neq \emptyset$, existe $\bar{x} \in \text{ir}(P)$ pelo Teorema 2.7.1 de [38]. Se $s = 0$, o resultado é trivial pois $r \geq 0$ e $r \leq s$. Caso contrário, seja x^1, x^2, \dots, x^s uma base para $\mathcal{N}(A^\circ)$. Existe $\epsilon_i > 0$ tal que $\bar{x} + \epsilon_i x^i \in P$, para $i \in \{1, 2, \dots, s\}$ [38]. Tome $\epsilon = \min\{\epsilon_i : i \in \{1, 2, \dots, s\}\} > 0$. Então, $\bar{x}, \bar{x} + \epsilon x^1, \bar{x} + \epsilon x^2, \dots, \bar{x} + \epsilon x^s$ são $s+1$ pontos afim-independentes de P , pois $\epsilon x^1, \epsilon x^2, \dots, \epsilon x^s$ são linearmente independentes. Logo $r \geq s$.

□

Seja G um grafo com n vértices, dado como entrada do problema de dominação romana, e $P_G = \{(x, y) \in \mathbb{R}^{n+n} : A(x, y) \leq b\}$ o poliedro que representa a formulação apresentada na Seção 4.3. Por definição

$$A = \begin{bmatrix} -I & (-I - \hat{A}) \\ I & I \end{bmatrix} \text{ e } b = \begin{bmatrix} -\hat{1} \\ \hat{1} \end{bmatrix}$$

onde $I \in \mathbb{R}^{n \times n}$ é uma matriz identidade, $\hat{1}$ é um vetor de tamanho n onde todas as componentes tem valor 1 e \hat{A} é a matriz de adjacência de G .

Lema 4.2. *Se G possui k vértices isolados então $\dim(P_G) \leq 2n - k$.*

Demonstração. Suponha que $v_i \in V(G)$ é um vértice isolado em G . Temos pela i -ésima linha que

$$(a^i)^T x \leq -1 \implies -(a^i)^T x \geq 1.$$

Como v_i é um vértice isolado, a i -ésima linha de \hat{A} é nula, e portanto

$$-(a^i)^T = (a^{i+n})^T$$

Sabendo que $(a^{i+n})^T x \leq 1$, isto implica que

$$-(a^i)^T x = (a^{i+n})^T x = 1.$$

Perceba que a afirmação acima é válida para todo i onde v_i é um vértice isolado em G . Além disso, $\{a^{i+n} : i = 1, \dots, n\}$ é um conjunto linearmente independente. E como G possui k vértices isolados, então $\text{posto}(A^{\bar{=}}) \geq k$ e pela Proposição 20 $\dim(P_G) = 2n - \text{posto}(A^{\bar{=}}) \leq 2n - k$, portanto

$$\dim(P_G) \leq 2n - k.$$

□

Lema 4.3. $\text{Posto}(A^{\bar{=}})$ é igual ao número de vértices isolados em G .

Demonstração. Seja u um vértice isolado em G . Para este vértice temos as seguintes restrições:

$$x_u + y_u \geq 1 \quad (4.7) \quad e$$

$$x_u + y_u \leq 1 \quad (4.8)$$

Logo, (4.7) e (4.8) implicam na igualdade $x_u + y_u = 1$ para o vértice u . Como tais igualdades são linearmente independentes, temos que $\text{posto}(A^{\bar{=}}) \geq k$, onde k é igual ao número de vértices isolados.

Para mostrar que $\text{posto}(A^{\bar{=}}) \leq k$, considere $\pi^t x + \theta^t y = \pi_0$ uma igualdade de P_G . Queremos mostrar que ela é combinação linear das igualdades formadas pelas restrições dos vértices isolados em G , ou seja:

- $\pi_u = \theta_u, \quad \forall u \in V(G)$, se u é isolado
- $\pi_u = \theta_u = 0, \quad \forall u \in V(G)$, se u não é isolado

Considere os pontos (\bar{x}, \bar{y}) e (\tilde{x}, \tilde{y}) tais que:

$$\begin{cases} \bar{x}_w = 1, & \forall w \in V(G) \\ \bar{y}_w = 0, & \forall w \in V(G) \end{cases} \quad \begin{cases} \tilde{x}_w = 1, & \forall w \in V(G) \setminus \{u\} \\ \tilde{y}_w = 0, & \forall w \in V(G) \setminus \{u\} \\ \tilde{x}_u = 0 \\ \tilde{y}_u = 1 \end{cases}$$

Perceba que $(\bar{x}, \bar{y}) \in P_G$ e $(\tilde{x}, \tilde{y}) \in P_G$. então:

$$\pi^t(\bar{x} - \tilde{x}) + \theta^t(\bar{y} - \tilde{y}) = 0$$

$$\pi_u - \theta_u = 0$$

$$\pi_u = \theta_u$$

Suponha agora que u possui pelo menos um vizinho $v \in V(G)$. Defina (\hat{x}, \hat{y}) tal que:

$$\begin{cases} \hat{x}_w = 1, & \forall w \in V(G) \setminus \{u, v\} \\ \hat{y}_w = 0, & \forall w \in V(G) \setminus \{v\} \\ \hat{x}_u = \hat{x}_v = 0 \\ \hat{y}_v = 1 \end{cases}$$

Como $(\hat{x}, \hat{y}) \in P_G$, então:

$$\pi^t(\bar{x} - \hat{x}) + \theta^t(\bar{y} - \hat{y}) = 0$$

$$\pi_u + \pi_v - \theta_v = 0$$

$$\pi_u = 0$$

□

Teorema 4.4. G possui k vértices isolados se e somente se $\dim(P_G) = 2n - k$.

Demonstração. Suponha que G possui k vértices isolados. Sabemos pelo Lema 4.2 que $\dim(P_G) \leq 2n - k$. Mais que isso, pela prova desse lema para todo vértice $u \in V(G)$ isolado, temos o conjunto de igualdades $X_u + Y_u = 1$ e pela prova do Lema 4.3 qualquer igualdade na forma $\pi^t x + \theta^t y = \pi_0$ é combinação linear do conjunto de igualdades mencionado acima.

Por outro lado, se $\dim(P_G) = 2n - k$, a Proposição 20 nos diz que $\dim(P_G) = 2n - \text{posto}(A^=)$. Portanto, $\text{posto}(A^=) = k$ e pelo Lema 4.3 o $\text{posto}(A^=)$ é igual ao número de vértices isolados em G . Concluimos então que G possui k vértices isolados.

□

Dizemos que um poliedro $P \subseteq \mathbb{R}^n$ tem dimensão plena se $\dim(P) = n$, i.e.

$$P \subseteq \{x \in \mathbb{R}^n : a^T x = \alpha\} \Rightarrow a = 0, \alpha = 0$$

Corolário 4.4.1. *Seja G um grafo e P_G o poliedro definido acima, então G não possui vértices isolados se e somente se P_G possui dimensão plena.*

Demonstração. Este corolário é consequência direta do Teorema 4.4. Se G não tem vértice isolado, então $\dim(P_G) = 2n - k$ e $k = 0$ e portanto, $\dim(P_G) = 2n$. Já no caso onde $\dim(P_G) = 2n$, supondo por contradição que existe pelo menos um vértice isolado, o Teorema anterior nos diz que $\dim(P_G) = 2n - k$, o que é um absurdo $\forall k \geq 1$.

□

4.5.2 Total Unimodularidade

Definição. Uma matriz é **Totalmente Unimodular (TU)** se e somente se, o determinante de toda submatriz quadrada é 0, 1 ou -1.

Proposição 21 ([38]). Seja I a representação da matriz identidade. Assim, os seguintes itens são equivalentes:

- A é TU.
- A^T é TU.
- $\begin{bmatrix} A \\ I \end{bmatrix}$ é TU.
- $\begin{bmatrix} A & I \end{bmatrix}$ é TU.
- $\begin{bmatrix} A \\ -A \end{bmatrix}$ é TU.
- $\begin{bmatrix} A & -A \end{bmatrix}$ é TU.

Proposição 22 ([38]). Seja A uma matriz TU, então $\{x \in \mathbb{R}^n : Ax \leq b, x \geq 0\}$ é inteiro.

Demonstração. Seja x um vértice de $\{x \in \mathbb{R}^n : Ax \leq b, x \geq 0\}$. Então x pode ser particionado em (x^1, x^2) , dado $Bx^1 = b, x^2 = 0$ para alguma submatriz invertível de (A, I) . Pela regra de Cramer, temos que:

$$Bx^1 = b \iff x^1 = B^{-1}b \iff \forall i : x_i^1 = \frac{\det(B^i)}{\det(B)}$$

onde B^i é a matriz onde cada coluna é igual a sua correspondente em B , exceto pela sua i -ésima coluna, que é substituída por b .

Logo, como A é uma matriz TU, x será inteiro pois $\det(B)$ é igual a +1 ou -1. \square

Teorema 4.5. Se G possui matriz de adjacência TU então o número de dominação romana pode ser calculado em tempo polinomial.

Demonstração. A formulação apresentada na Seção 4.2 pode ser visto como $\max\{c^t x : Ax \leq b\}$. E com isso, podemos representar as restrições da seguinte maneira:

$$Ax \leq b \Rightarrow \begin{bmatrix} I & I \\ I & -M \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix} \leq \begin{bmatrix} \hat{1} \\ 0 \end{bmatrix}$$

onde M representa a matriz de adjacência do grafo G dado como instância.

Note que usando a Proposição 21, A é TU se e somente se M é TU. Ou seja, temos que todo grafo que possui matriz de adjacência totalmente unimodular pode ser resolvido de forma ótima com a relaxação do problema utilizando variáveis com valor real.

\square

Corolário 4.5.1. Se G é uma floresta ou um grafo unicíclico com o tamanho de seu ciclo divisível por 4 então o número de dominação romana de G pode ser resolvido em tempo polinomial.

Demonstração. Em [39] é mostrado que toda floresta e todo grafo unicíclico com o tamanho de seu ciclo divisível por 4 possuem matrizes de adjacência totalmente unimodular. Portanto, pelo Teorema 4.5, sabemos que o problema de dominação romana para essas classes de grafo pode ser resolvido de forma ótima em tempo polinomial.

□

5 Conclusão e Trabalhos Futuros

Esta dissertação de mestrado tem como centro o estudo do problema de dominação romana. Baseado em um problema originalmente criado por entusiastas do xadrez, a dominação em grafos tem sido um assunto bastante explorado por mais de um século. Durante este período de tempo, muitas variantes e generalizações deste tema apareceram, e com elas veio a renovação do interesse na pesquisa sobre a dominação em grafos. Isso não foi diferente com o problema de dominação romana, o qual possui um significado histórico datado do início do século III.

Atualmente já existem vários artigos sobre o problema de dominação romana e suas variantes. Ele foi originalmente sugerido por Ian Stewart [1] e posteriormente foi mais trabalhado por ReVelle, Toregas e Falkson [35].

Sabemos que existem algoritmos em tempo linear que podem resolver o problema em algumas classes de grafo, como por exemplo, cografos, árvores e grafos de intervalo. Porém, em um caso geral, o problema de encontrar o número de dominação romana é NP-difícil.

Conseguimos provar que o problema permanece de fato NP-difícil, mesmo quando restringimos a entrada do problema a subgrafos induzidos de grid. Com relação a algoritmos aproximativos, foi demonstrado que o número de dominação romana é APX-difícil, mesmo em grafos bipartidos com grau máximo 4.

Apesar de já existir um resultado sobre FPT em grafos planares, nós utilizamos de um método de lógica de segunda ordem para mostrar que o número de dominação romana é FPT em tempo $O(n^2)$ para grafos com largura em árvore local (*local-treewidth*) limitada e com isso construímos um algoritmo em tempo $O((\Delta + 1)^k(m + n))$ que comprova o fato de que o problema de dominação romana é FPT para grafos com grau máximo limitado Δ .

Ao considerar o problema quando temos orientação nas arestas, observamos que o problema de dominação romana em digrafos planares pode ser resolvido em tempo $\mathcal{O}^*(3.5414^k)$ e, além disso, é NP-completo em torneios bipartidos. Isso mostra uma disparidade, pois o problema pode ser resolvido em tempo linear em cografos.

Utilizamos também uma abordagem de modelagem matemática e análise de poliedro e conseguimos provar que:

1. O grafo não possui vértices isolados se e somente se o poliedro definido pelas restrições do problema possui dimensão plena.
2. Dado um grafo que é uma floresta ou um grafo unicíclico com o tamanho de seu ciclo divisível por 4, então o problema de dominação romana pode ser resolvido em tempo polinomial.

Assim, tendo feito um amplo estudo do problema, sugerimos como trabalhos futuros aplicações de teoremas bem conhecidos de Dominação na formulação melhorada apresentada no Capítulo 4.

Referências

- 1 STEWART, I. Defend the roman empire! *Scientific American*, v. 281, p. 136–138, 1999.
- 2 COCKAYNE, E. J. et al. Roman domination in graphs. *Discrete Mathematics*, Elsevier, v. 278, n. 1, p. 11–22, 2004.
- 3 HAYNES, T. W.; HEDETNIEMI, S.; SLATER, P. *Fundamentals of domination in graphs*. [S.l.]: CRC Press, 1998.
- 4 JAENISCH, C. d. *Applications de l'Analyse Mathematique an Jenudes Echechs*. [S.l.]: Petrograd, 1862.
- 5 BALL, W. W. R. *Mathematical recreations and problems of past and present times*. [S.l.]: Macmillan, 1892.
- 6 WHYBURN, L. Challenging mathematical problems with elementary solutions. vol. 1, combinatorial analysis and probability theory. am yaglom and im yaglom. translated from the russian edition (1954) by james mccawley, jr. revised and edited by basil gordon. holden-day, san francisco, 1964. viii+ 231 pp. illus. In: AMERICAN ASSOCIATION FOR THE ADVANCEMENT OF SCIENCE. [S.l.], 1965.
- 7 BERGE, C. Theory of graphs and its applications. 1962. *Methuen, London*.
- 8 ORE, O.; ORE, Y. *Theory of graphs*. [S.l.]: American Mathematical Society Providence, RI, 1962. v. 38.
- 9 COCKAYNE, E. J.; HEDETNIEMI, S. T. Towards a theory of domination in graphs. *Networks*, Wiley Online Library, v. 7, n. 3, p. 247–261, 1977.
- 10 REVELLE, C. S.; ROSING, K. E. Defendens imperium romanum: a classical problem in military strategy. *American Mathematical Monthly*, JSTOR, p. 585–594, 2000.
- 11 PAGOURTZIS, A. et al. *Server placements, Roman domination and other dominating set variants*. [S.l.]: Springer, 2002.
- 12 SHANG, W.; HU, X. The roman domination problem in unit disk graphs. In: *Computational Science–ICCS 2007*. [S.l.]: Springer, 2007. p. 305–312.
- 13 ARQUILLA, J.; FREDRICKSEN, H. Graphing an optimal grand strategy. *Military Operations Research*, Military Operations Research Society, v. 1, n. 3, p. 3–17, 1995.
- 14 WEST, D. B. *Introduction to graph theory*. 2. ed. [S.l.]: Pearson, 2001.
- 15 HEDETNIEMI, S. et al. *Roman domination in graphs II. Slides and notes from presentation at Ninth Quadrennial International Conference on Graph Theory, Combinatorics, Algorithms, and Applications*. 2000.
- 16 JR, P. A. D. *Applications and variations of domination in graphs*. Tese (Doutorado) — Citeseer, 2000.

- 17 GAREY, M. R.; JOHNSON, D. S. *Computers and Intractability; A Guide to the Theory of NP-Completeness*. New York, NY, USA: W. H. Freeman & Co., 1990. ISBN 0716710455.
- 18 LIEDLOFF, M. et al. Roman domination over some graph classes. In: _____. *Graph-Theoretic Concepts in Computer Science: 31st International Workshop, WG 2005, Metz, France, June 23-25, 2005, Revised Selected Papers*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005. p. 103–114. ISBN 978-3-540-31468-4. Disponível em: http://dx.doi.org/10.1007/11604686_10.
- 19 CRESCENZI, P. A short guide to approximation preserving reductions. In: IEEE. *Computational Complexity, 1997. Proceedings., Twelfth Annual IEEE Conference on (Formerly: Structure in Complexity Theory Conference)*. [S.l.], 1997. p. 262–273.
- 20 AUSIELLO, G. et al. *Complexity and approximation: Combinatorial optimization problems and their approximability properties*. [S.l.]: Springer Science & Business Media, 2012.
- 21 DOWNEY, R. G.; FELLOWS, M. R. *Parameterized complexity*. [S.l.]: Springer Science & Business Media, 2012.
- 22 FERNAU, H. Roman domination: a parameterized perspective†. *International Journal of Computer Mathematics*, Taylor & Francis, v. 85, n. 1, p. 25–38, 2008.
- 23 FLUM, J.; GROHE, M. *Parameterized Complexity Theory*. [S.l.]: Springer, 2010.
- 24 CHAMBERS, E. W. et al. Extremal problems for roman domination. *SIAM Journal on Discrete Mathematics*, SIAM, v. 23, n. 3, p. 1575–1586, 2009.
- 25 FAVARON, O. et al. On the roman domination number of a graph. *Discrete Mathematics*, Elsevier, v. 309, n. 10, p. 3447–3451, 2009.
- 26 DAHLHAUS, E. et al. The complexity of multiterminal cuts. *SIAM Journal on Computing*, SIAM, v. 23, n. 4, p. 864–894, 1994.
- 27 GAREY, M. R.; JOHNSON, D. S. The rectilinear steiner tree problem is np-complete. *SIAM Journal on Applied Mathematics*, SIAM, v. 32, n. 4, p. 826–834, 1977.
- 28 EPPSTEIN, D. Diameter and treewidth in minor-closed graph families. *Algorithmica*, Springer, v. 27, n. 3-4, p. 275–291, 2000.
- 29 BODLAENDER, H. L. A partial k-arboretum of graphs with bounded treewidth. *Theoretical computer science*, Elsevier, v. 209, n. 1, p. 1–45, 1998.
- 30 COURCELLE, B. Graph structure and monadic second-order logic: Language theoretical aspects. In: _____. *Automata, Languages and Programming: 35th International Colloquium, ICALP 2008, Reykjavik, Iceland, July 7-11, 2008, Proceedings, Part I*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008. p. 1–13. ISBN 978-3-540-70575-8. Disponível em: http://dx.doi.org/10.1007/978-3-540-70575-8_1.
- 31 FLUM, J.; GROHE, M. The parameterized complexity of counting problems. *SIAM Journal on Computing*, SIAM, v. 33, n. 4, p. 892–922, 2004.

- 32 ALBER, J. et al. A refined search tree technique for dominating set on planar graphs. *Journal of Computer and System Sciences*, Elsevier, v. 71, n. 4, p. 385–405, 2005.
- 33 TOREGAS, C. et al. The location of emergency service facilities. *Operations Research, Informis*, v. 19, n. 6, p. 1363–1373, 1971.
- 34 TOREGAS, C.; REVELLE, C. Binary logic solutions to a class of location problem*. *Geographical Analysis*, Wiley Online Library, v. 5, n. 2, p. 145–155, 1973.
- 35 REVELLE, C.; TOREGAS, C.; FALKSON, L. Applications of the location set-covering problem. *Geographical analysis*, Wiley Online Library, v. 8, n. 1, p. 65–76, 1976.
- 36 BERMUDO, S.; FERNAU, H.; SIGARRETA, J. M. The differential and the roman domination number of a graph. *Applicable Analysis and Discrete Mathematics*, n. 00, p. 3–3, 2014.
- 37 BURGER, A.; VILLIERS, A. D.; VUUREN, J. V. A binary programming approach towards achieving effective graph protection. In: *Proceedings of the 42nd Annual Conference of the Operations Research Society of South Africa, Stellenbosch*. [S.l.: s.n.], 2013. p. 19–30.
- 38 FERREIRA, C. E.; WAKABAYASHI, Y. *Combinatória poliédrica e planos-de-corte faciais*. [S.l.]: UNICAMP-Instituto de Computacao, 1996.
- 39 AKBARI, S.; KIRKLAND, S. On unimodular graphs. *Linear algebra and its applications*, Elsevier, v. 421, n. 1, p. 3–15, 2007.