



UNIVERSIDADE FEDERAL DO CEARÁ
CENTRO DE CIÊNCIAS
DEPARTAMENTO DE COMPUTAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO
DOUTORADO EM COMPUTAÇÃO

EURINARDO RODRIGUES COSTA

RESULTADOS EM JOGOS DE COLORAÇÃO E PERSEGUIÇÃO EM GRAFOS

FORTALEZA

2022

EURINARDO RODRIGUES COSTA

RESULTADOS EM JOGOS DE COLORAÇÃO E PERSEGUIÇÃO EM GRAFOS

Tese apresentada ao Programa de Pós-Graduação em Ciência da Computação do Centro de Ciências da Universidade Federal do Ceará, como requisito parcial à obtenção do título de doutor em Ciência da Computação. Área de Concentração: Teoria dos Grafos (Teoria da Computação).

Orientador: Prof. Dr. Rudini M. Sampaio.

FORTALEZA

2022

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

- C871r Costa, Eurinardo Rodrigues.
Resultados em jogos de coloração e perseguição em grafos / Eurinardo Rodrigues Costa. – 2022.
104 f. : il. color.
- Tese (doutorado) – Universidade Federal do Ceará, Centro de Ciências, Programa de Pós-Graduação em
Ciência da Computação, Fortaleza, 2022.
Orientação: Prof. Dr. Rudini Menezes Sampaio.
1. jogo da coloração. 2. jogo da coloração gulosa. 3. jogo do espião. 4. complexidade computacional. I.
Título.

CDD 005

EURINARDO RODRIGUES COSTA

RESULTADOS EM JOGOS DE COLORAÇÃO E PERSEGUIÇÃO EM GRAFOS

Tese apresentada ao Programa de Pós-Graduação em Ciência da Computação do Centro de Ciências da Universidade Federal do Ceará, como requisito parcial à obtenção do título de doutor em Ciência da Computação. Área de Concentração: Teoria dos Grafos (Teoria da Computação).

Aprovada em: 02/09/2022.

BANCA EXAMINADORA

Prof. Dr. Rudini M. Sampaio (Orientador)
Universidade Federal do Ceará (UFC)

Prof. Dra. Ana Karolinnna Maia de Oliveira
Universidade Federal do Ceará (UFC)

Prof. Dr. Carlos Vinicius Gomes Costa Lima
Universidade Federal do Cariri (UFCA)

Prof. Dr. Nicolas de Almeida Martins
Universidade da Integração Internacional da
Lusofonia Afro-Brasileira (UNILAB)

Prof. Dr. Thiago Braga Marcilon
Universidade Federal do Cariri (UFCA)

À minha família, por sua paciência em me aturar.
De modo mais especial, à minha esposa Dayana
e minhas filhas Maria Sofia e Catarina.

AGRADECIMENTOS

Primeiramente, gostaria de agradecer a Deus por sua infinita bondade em ter dado para nós o Seu Filho Jesus Cristo que veio para remissão dos pecados do mundo. Também quero agradecer à Santíssima Virgem Maria e a São José por terem intercedido por mim nos momentos de aflição e por serem exemplos de santidade em minha vida.

Agradeço imensamente aos meus pais Maria e Cícero (*in memoriam*) por terem me dado a vida e, principalmente, agradeço à minha mãe por ter trabalhado tanto e feito todo o esforço do mundo para me criar e me dar uma boa educação desde a partida do meu pai Cícero ao meus dois anos. Agradeço também à minha irmã Eurifran e ao meu cunhado Nonato por serem pessoas maravilhosas. Não posso esquecer dos meus tios Lúcia, Verônica, Luciene, Eliene, Evaristo (*in memoriam*), Cazuzá, meu avô Vicente (*in memoriam*) e minha avó Ana que cuidaram de mim e me ensinaram tantas coisas boas. Obrigado aos meus primos Jorge e Eduardo por terem dividido tantos momentos comigo durante minha infância e adolescência.

Obrigado à minha esposa Dayana por ter me compreendido nesse tempo dedicado ao doutorado e, principalmente, por ter escolhido fazer parte de todo resto da minha vida. Agradeço também às minhas filhas Maria Sofia e Catarina que, mesmo sem compreender, me deram muita força e disposição para dar continuidade no doutorado.

Meu agradecimento especial ao meu orientador Rudini por sua paciência e por ter acreditado na minha capacidade.

Obrigado aos meus amigos da Capela Imaculada Conceição, Carlos, Vera, Joel, Diego, Kamilla e todos os outros que são pessoas de Deus e se empenham verdadeiramente na construção do Reino do Senhor. Obrigado pelos momentos de descontração em meio ao turbilhão de coisas e compromissos que eu tive durante esse tempo. Meu muito obrigado ao Padre Fatico (*in memoriam*), exemplo de sacerdote e bom formador.

Obrigado também aos meus amigos do NEMO, Pablo, Tatiane, Marcio e Bonfim, pela força dada nos momentos mais difíceis do doutorado.

Obrigado a todos os professores do grupo PARCO que me acompanham desde minha graduação.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001, Fundação Cearense de Apoio ao Desenvolvimento Científico e Tecnológico (FUNCAP) e Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

“Vinde a mim os que estais cansados sob o peso
de vosso fardo e vos darei descanso.”

(Jesus Cristo)

RESUMO

Nesta tese, estudamos três jogos em grafos. Para o Jogo da Coloração, respondemos uma questão aberta de longa data proposta por Bodlaender em 1991: o número cromático de jogo é PSPACE-difícil. Mostramos também que o número guloso de jogo é PSPACE-difícil, no Jogo da Coloração Gulosa. De fato, mostramos que ambos os problemas, do Jogo da Coloração e do Jogo da Coloração Gulosa, são PSPACE-Completos, mesmo se o número de cores é o número cromático. Além disso, provamos que o número guloso de jogo é igual ao número cromático para várias superclasses de cografos, estendendo o resultado de Havet e Zhu de 2013. Para o Jogo do Espião, obtemos um limite superior para o produto forte de dois grafos gerais e mostramos exemplos de grafos grades Rei que chegam nesse limite e outros grafos grades Rei que o número de guarda é menor que esse limite. Obtemos também o valor exato do número de guarda no produto lexicográfico de dois grafos gerais para cada distância $d \geq 2$. Do ponto de vista algorítmico, mostramos o resultado positivo: se o número k de guardas é fixo, o Jogo do Espião é solucionável em tempo polinomial $O(n^{3k+2})$ para cada velocidade $s \geq 2$ e distância $d \geq 0$. Em outras palavras, o Jogo do Espião está em XP quando o parâmetro é o número de guardas. Usando o algoritmo XP, obtemos um algoritmo FPT para grafos com poucos P_4 's. Como resultado negativo, provamos que o Jogo do Espião é W[2]-difícil mesmo em grafos bipartidos quando o parâmetro é o número de guardas, para cada velocidade $s \geq 2$ e distância $d \geq 0$, estendendo, deste modo, o resultado de Cohen et al. de 2018.

Palavras-chave: jogo da coloração; jogo da coloração gulosa; jogo do espião; complexidade computacional.

ABSTRACT

In this thesis we study three games in graphs. In the graph coloring game we answer a long-standing open question proposed by Bodlaender in 1991: the game chromatic number is PSPACE-hard. In the greedy coloring game we also prove that the game Grundy number is PSPACE-hard. In fact, we prove that both problems, the graph coloring game and the greedy coloring game, are PSPACE-Complete even if the number of colors is the chromatic number. Moreover, we prove that the game Grundy number is equal to the chromatic number for several superclasses of cographs, extending a result of Havet and Zhu in 2013. In the spy game we obtain an upper bound on the strong product of two general graphs and obtain examples of King grids that match this bound and other examples for which the guard number is smaller. We also obtain the exact value of the guard number in the lexicographic product of two general graphs for any distance $d \geq 2$. From the algorithmic point of view, we prove a positive result: if the number k of guards is fixed, the spy game is solvable in polynomial time $O(n^{3k+2})$ for every speed $s \geq 2$ and distance $d \geq 0$. In other words, the spy game is XP when parameterized by the number of guards. This XP algorithm is used to obtain an FPT algorithm on graphs with few P_4 's. As a negative result, we prove that the spy game is W[2]-hard even in bipartite graphs when parameterized by the number of guards, for every speed $s \geq 2$ and distance $d \geq 0$, extending the hardness result of Cohen et al. in 2018.

Keywords: graph coloring game; greedy coloring game; spy game; computational complexity.

LISTA DE FIGURAS

Figura 1 – Grafo exemplo para os jogos de coloração	12
Figura 2 – Possíveis jogadas no Jogo da Coloração	13
Figura 3 – Configuração de jogo para o Jogo da Coloração no grafo da Figura 1	14
Figura 4 – Segunda configuração de jogo para o Jogo da Coloração no grafo da Figura 1	14
Figura 5 – Possíveis jogadas para o Jogo da Coloração Gulosa	15
Figura 6 – Quem começar perde, no Jogo da Coloração ou no Jogo da Coloração Gulosa	16
Figura 7 – Grafo exemplo para o Jogo do Espião	21
Figura 8 – Possíveis jogadas para o Jogo do Espião no grafo da Figura 7	22
Figura 9 – Exemplo de um grafo	26
Figura 10 – Uma árvore enraizada no vértice a	29
Figura 11 – Um grafo bipartido	30
Figura 12 – Superfícies em \mathbb{R}^3	30
Figura 13 – K_7 no quadrado	31
Figura 14 – Hierarquia de grafos com poucos P_4 's	32
Figura 15 – Exemplo de aranha magra e aranha gorda	33
Figura 16 – Exemplo: $3SAT \leq_p$ COBERTURA DE VÉRTICES	37
Figura 17 – Exemplos de circuitos booleanos	44
Figura 18 – Grafo F_1 : ferramenta para demonstrar o Teorema 5	49
Figura 19 – Exemplo de redução para o Jogo da Coloração	53
Figura 20 – Grafo F_2 : ferramenta para demonstrar o Teorema 6	58
Figura 21 – Exemplo de redução para o Jogo da Coloração Gulosa	59
Figura 22 – Grafo split G com partição (C, S)	61
Figura 23 – P_4 -esparso: G_1, G_2 e G_3	63
Figura 24 – P_4 -esparso: G_k	63
Figura 25 – $\chi(G_k) = 2k, \Gamma(G_k) = 3k$ e $\Gamma_g(G_k) = 2k$	64
Figura 26 – Grafo $\overline{P_5}$	65
Figura 27 – Exemplos de grades Rei $P_3 \boxtimes P_5$ e $P_5 \boxtimes P_3$	69
Figura 28 – Exemplo de estratégia vencedora em $P_3 \boxtimes P_3$ e $P_5 \boxtimes P_5$ com dois guardas	71
Figura 29 – Exemplo de estratégia vencedora para o espião em $P_4 \boxtimes P_4$ e $P_6 \boxtimes P_6$ com três guardas	73
Figura 30 – Exemplos de produto cartesiano $P_3 \square P_5$ e $P_5 \square P_3$	74

Figura 31 – Exemplo de grade Rei $P_3 \cdot P_5$	75
Figura 32 – $P_5 \square P_5$ e $P_5 \cdot P_5$ com uma estratégia vencedora para o espião contra um guarda e $(s, d) = (2, 1)$	76
Figura 33 – Exemplo de redução para o Lema 18	86
Figura 34 – Exemplo de redução para o Lema 19	88
Figura 35 – Exemplo de redução para o Lema 20	90
Figura 36 – Exemplo de redução para o Lema 21	92
Figura 37 – Exemplo de redução para o Lema 22	93
Figura 38 – Exemplo de redução para o Lema 23	95

SUMÁRIO

1	INTRODUÇÃO	11
1.1	Jogos de coloração em grafos	12
1.2	Jogo do Espião	20
1.3	Organização do texto	24
2	CONCEITOS BÁSICOS	26
2.1	Introdução à Teoria dos Grafos	26
2.1.1	<i>Definições básicas</i>	26
2.1.2	<i>Coloração em Grafos</i>	28
2.1.3	<i>Classes de grafos</i>	29
2.1.4	<i>Grafos com poucos P_4's</i>	31
2.2	Introdução à Teoria da Complexidade	34
2.3	Complexidade Parametrizada	39
2.3.1	<i>Hierarquia W</i>	42
3	JOGOS DE COLORAÇÃO EM GRAFOS	45
3.1	O número cromático de jogo é PSPACE-Difícil	45
3.2	O número guloso de jogo é PSPACE-Difícil	55
3.3	O número guloso de jogo em grafos split	59
3.4	Número guloso de jogo em grafos com poucos P_4's	62
4	JOGO DO ESPIÃO	68
4.1	Jogo do Espião em grades Rei e produtos de grafos	68
4.2	O Jogo do Espião está em XP quando o parâmetro é o número de guardas	77
4.3	Jogo do Espião está em FPT em grafos com poucos P_4's	79
4.4	O Jogo do Espião é $W[2]$-difícil em grafos bipartidos	84
5	CONCLUSÕES E TRABALHOS FUTUROS	98
5.1	Considerações finais para os jogos de coloração em grafos	98
5.2	Considerações finais para o Jogo do Espião	99
	REFERÊNCIAS	101

1 INTRODUÇÃO

Jogos de tabuleiro e enigmas matemáticos acompanham a humanidade desde muito tempo. Muitos autores tentaram registrar os “enigmas matemáticos” mais conhecidos de sua época. Entre eles, destaca-se H.E. Dudeney e seu livro “*Amusements in Mathematics*” (“Diversão em Matemática”) (DUDENEY, 1917) de 1917, com uma lista impressionante de 430 enigmas. Muitos deles ainda são estudados hoje em dia, como o problema “Puzzle with Pawns” (“Quebra-cabeça com Peões”) que se estuda em Geometria Computacional com o nome de “*No-three-in-line problem*” (“Problema sem três linhas”).

Entre 1956 e 1981, Martin Gardner escreveu a coluna “*Mathematical Games*” (“Jogos de Matemática”) da *Scientific American*, sendo um grande divulgador de jogos matemáticos nesses 25 anos. Um dos problemas estudados nesta tese, o Jogo de Coloração, foi proposto pela primeira vez no último ano de publicação de sua coluna. Desde então, muitos jogos foram estudados do ponto de vista computacional. Um dos pioneiros é o Problema TQBF “*Totally quantified Boolean formula*” (“Fórmula Booleana Totalmente Quantificada”), também chamado QSAT “*Quantified SAT*” (“SAT quantificado”) que foi o primeiro problema provado ser PSPACE-Completo, uma noção de complexidade computacional voltada para jogos que é definida nas próximas seções (STOCKMEYER; MEYER, 1973) (veja também (FORTNOW, 2005) para mais detalhes). Nesta tese, trabalhamos com uma variante desse jogo lógico, chamada POS-CNF.

Muitos jogos combinatórios foram pesquisados do ponto de vista computacional. Para maiores informações, o livro “*Games, Puzzles and Computation*” (“Jogos, Quebra-cabeças e Computação”) de 2009 dos autores Hearn e Demaine (HEARN; DEMAINE, 2009) possui uma coletânea de vários jogos computacionais e demonstrações de suas complexidades. Outro exemplo é o livro “*The Game of Cops and Robbers on Graphs*” (“O Jogo de Polícia e Ladrão em Grafos”) de 2011 dos autores Bonato e Nowakowski (BONATO; NOWAKOVSKI, 2011), que estuda Jogos de Perseguição em grafos, principalmente o Jogo de Polícia e Ladrão e algumas de suas variantes.

Essa literatura é uma pequena amostra de como essa área de pesquisa teórica em jogos computacionais é bastante ativa. Esta tese contribui neste sentido investigando dois Jogos de Coloração (Seção 1.1), a saber o Jogo da Coloração clássico e o Jogo da Coloração Gulosa, e também investigando um Jogo de Perseguição em Grafos, a saber, o Jogo do Espião (Seção 1.2).

1.1 Jogos de coloração em grafos

Nesta seção, apresentamos dois jogos de coloração em grafos. O Jogo de Coloração que foi descrito por Gardner em 1981 em sua coluna “*Mathematical Games*” na *Scientific American* (GARDNER, 1981) e o Jogo da Coloração Gulosa que foi introduzido em (HAVET; ZHU, 2013).

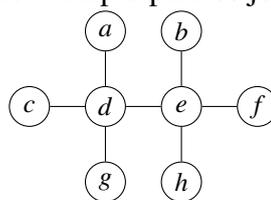
Sejam G um grafo simples finito e C um conjunto de inteiros não negativos com k elementos. No *Jogo da Coloração* em G , dois jogadores, Alice e Bob, alternam suas jogadas com as seguintes regras:

1. Alice faz a primeira jogada do início do jogo;
2. uma jogada consiste em escolher um vértice v qualquer de G que não esteja colorido e, em seguida, colorir v com uma cor de C que não tenha sido atribuída a nenhum dos vértices vizinhos de v já coloridos; e
3. Alice vence o jogo se todos os vértices forem coloridos. Caso contrário, Bob vence o jogo.

O *Jogo da Coloração Gulosa* segue do Jogo da Coloração com a modificação de que o vértice escolhido em cada jogada deve ser colorido pelo menor inteiro possível de C .

Denotamos por $A(v, c)$ a jogada em que Alice escolhe o vértice v e o colore com a cor c . De modo análogo, denotamos por $B(v, c)$ a jogada em que Bob escolhe o vértice v e o colore com a cor c . Para exemplo dos dois jogos, observe o grafo da Figura 1.

Figura 1 – Grafo exemplo para os jogos de coloração



Fonte: elaborado pelo autor.

Considerando o Jogo da Coloração com o conjunto de cores $C = \{1, 2, 3\}$, conjunto no qual pode-se fazer uma bijeção com as cores azul, verde e amarelo, na Figura 2 temos um exemplo das seguintes jogadas (separadas em rodadas de duas jogadas cada):

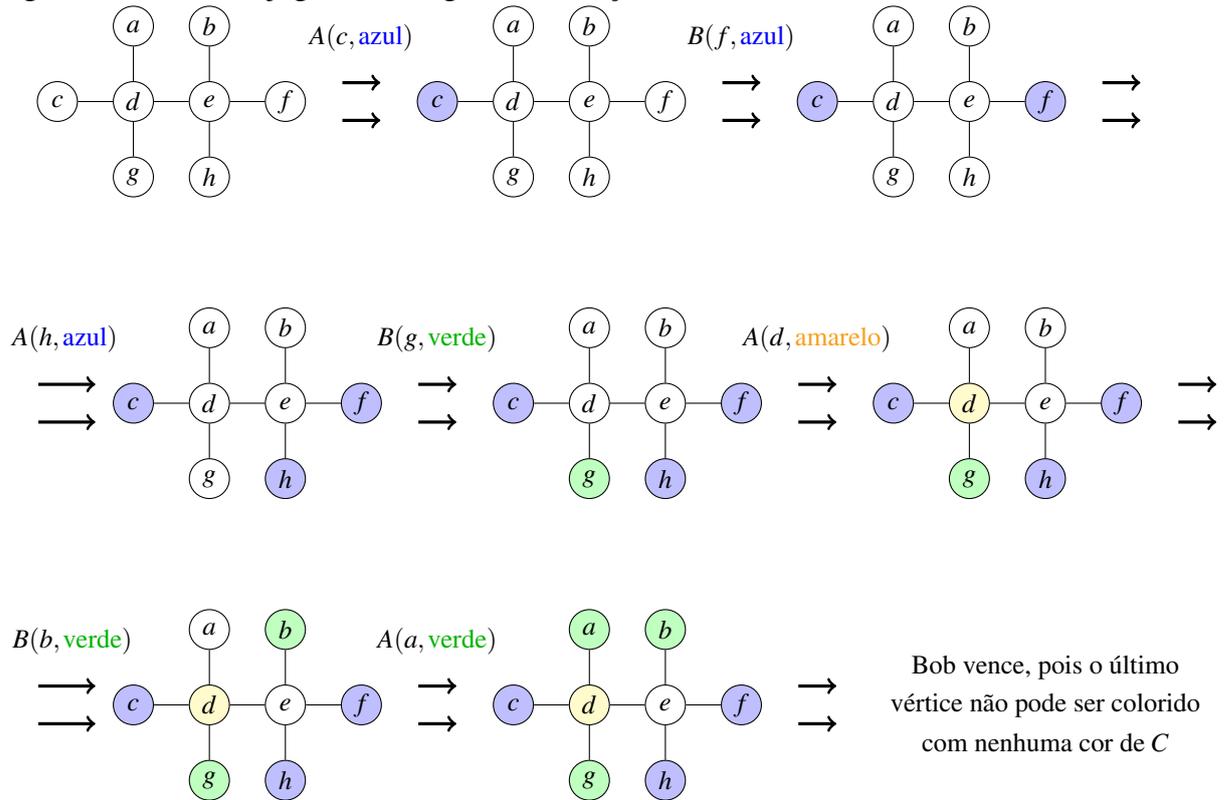
Rodada 1: $A(c, \text{azul}) - B(f, \text{azul})$

Rodada 2: $A(h, \text{azul}) - B(g, \text{verde})$

Rodada 3: $A(d, \text{amarelo}) - B(b, \text{verde})$

Rodada 4: $A(a, \text{verde}) -$ Bob vence, pois e não pode ser colorido com nenhuma cor de C .

Figura 2 – Possíveis jogadas no Jogo da Coloração



Uma *estratégia* para um jogador é uma função que, para cada configuração possível durante o jogo, informa a próxima jogada. Dizemos que uma estratégia é uma *estratégia vencedora* se, independente da estratégia usada pelo adversário, o jogador sempre sai vitorioso. Em geral, a descrição de uma estratégia não precisa considerar todas as configurações de um jogo, mas as configurações possíveis diante da estratégia do próprio jogador. Por exemplo, se Alice colore primeiramente um certo vértice v com a cor 1 em sua estratégia, não será necessário descrever a estratégia de Alice em configurações onde v está colorido com outra cor.

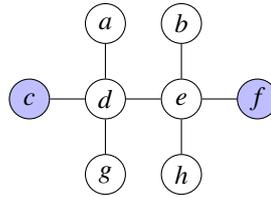
O interessante do grafo da Figura 1 é que se Alice iniciar o jogo colorindo o vértice c , então Bob possui uma estratégia vencedora. Supondo, sem perda de generalidade, que as duas primeiras jogadas sejam na primeira rodada:

Rodada 1: $A(c, \text{azul}) - B(f, \text{azul})$.

Até essa jogada temos a configuração de jogo da Figura 3. Então basta Bob fazer com que os vizinhos dos vértices centrais d e e sejam coloridos com as cores **verde** e **amarelo**. Para entendermos melhor, podemos analisar os casos para as respostas de Bob na segunda rodada:

- Rodada 2: $A(d, \text{verde ou amarelo}) - B(b, \text{amarelo ou verde})$. Bob vencerá o jogo, pois e não poderá ser colorido com nenhuma cor de C .

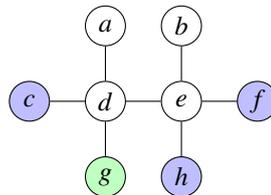
Figura 3 – Configuração de jogo para o Jogo da Coloração no grafo da Figura 1



Fonte: elaborado pelo autor.

- Rodada 2: Alice colore e . Análogo ao item anterior.
- Rodada 2: $A(h, \text{verde ou amarelo}) - B(d, \text{amarelo ou verde})$. Bob vencerá o jogo, pois e não poderá ser colorido com nenhuma cor de C .
- Rodada 2: $A(h, \text{azul}) - B(g, \text{verde})$. Nesta configuração de jogo (veja a Figura 4), perceba que Bob vencerá o jogo se Alice não colorir d , pois Bob poderá colorir e ou a com a cor **amarelo** e o vértice d não poderá ser colorido com nenhuma cor de C . Logo, Alice, na Jogada 5, deverá colorir o vértice d com a cor **amarelo** e Bob vencerá colorindo b com a cor **verde**, pois e não poderá ser colorido com as cores de C .
- Rodada 2: Alice colore a ou b ou g . Análogo aos dois casos anteriores.

Figura 4 – Segunda configuração de jogo para o Jogo da Coloração no grafo da Figura 1



Fonte: elaborado pelo autor.

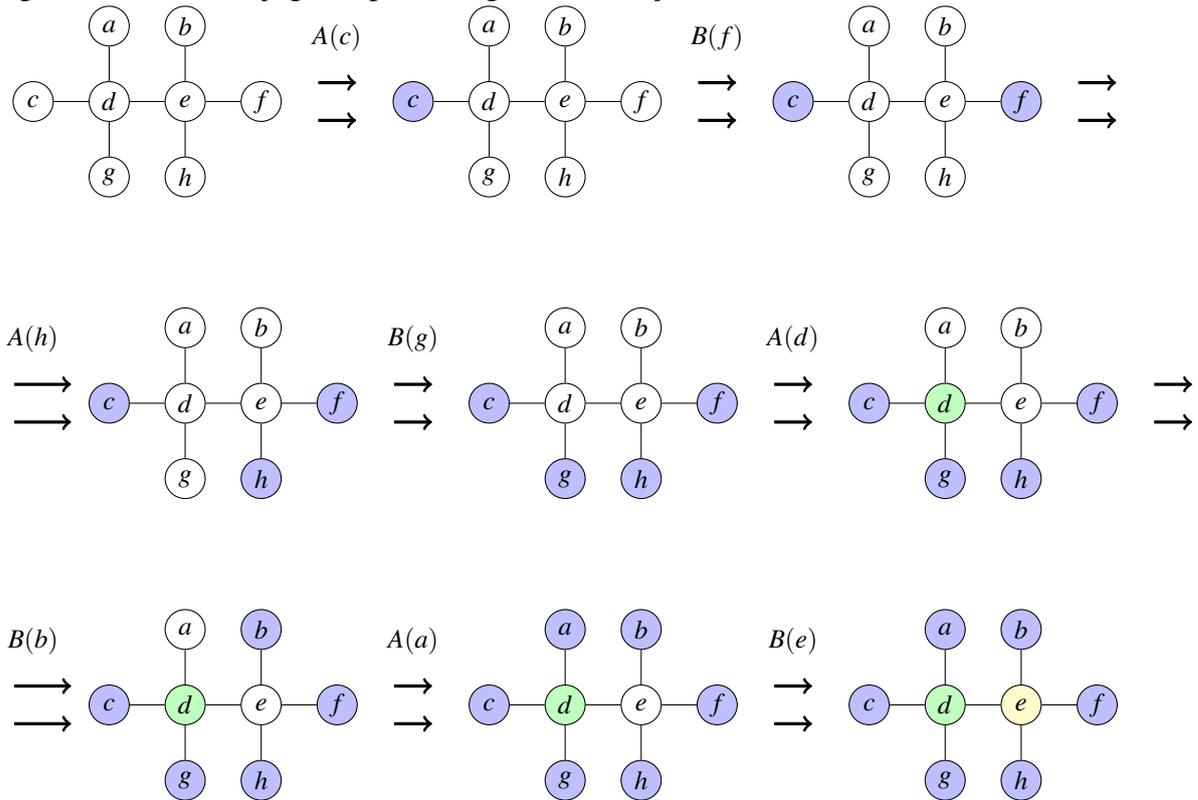
Agora consideramos o Jogo da Coloração Gulosa. Denotamos por $A(v)$ a jogada em que Alice colore o vértice v (lembrando, com a menor cor possível). De modo análogo, denotamos $B(v)$ a jogada em que Bob colore o vértice v com a menor cor possível. Note que Alice possui uma estratégia vencedora no grafo da Figura 1, mesmo iniciando o jogo colorindo o vértice c . Uma simples estratégia para Alice é colorir os vértices centrais d e e em suas próximas jogadas. Segue abaixo um exemplo de sequência de jogadas que pode ser visualizada na Figura 5 (considere que 1 é a cor **azul**, 2 é a **verde** e 3 é a cor **amarela**):

Rodada 1: $A(c)$ (cor 1/**azul**) – $B(f)$ (cor 1/**azul**)

Rodada 2: $A(h)$ (cor 1/**azul**) – $B(g)$ (cor 1/**azul**)

Rodada 3: $A(d)$ (cor 2/**verde**) – $B(b)$ (cor 1/**azul**)

Figura 5 – Possíveis jogadas para o Jogo da Coloração Gulosa



Rodada 4: $A(a)$ (cor 1/azul) – $B(e)$ (cor 3/amarelo)

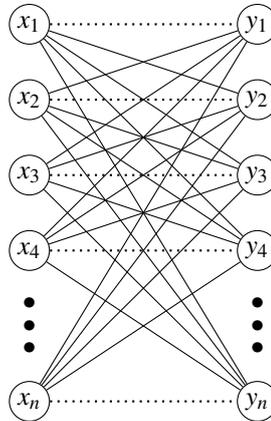
Alice vence o jogo, pois o grafo inteiro foi colorido com azul, verde e amarelo.

Algumas variantes desses jogos são possíveis e já foram investigadas em alguns artigos. Por exemplo, para $X \in \{Alice, Bob\}$ e $Y \in \{Alice, Bob, ninguém\}$, sejam $g_{X,Y}$ e $gg_{X,Y}$ as variantes do Jogo da Coloração e do Jogo da Coloração Gulosa, com a diferença que X começa o jogo e Y pode passar jogadas. Por exemplo, se $Y = X = Bob$, então temos que Bob começa o jogo e pode passar jogadas em $g_{X,Y}$ e $gg_{X,Y}$. Note que, em todas essas variantes, no máximo um jogador pode passar jogadas, inclusive a primeira jogada. Além disso, sejam $cg_{X,Y}$ e $cgg_{X,Y}$ as variantes do Jogo da Coloração e do Jogo da Coloração Gulosa, com a diferença que X começa o jogo, Y pode passar jogadas e o subgrafo induzido pelos vértices coloridos deve ser conexo em qualquer momento do jogo. Quando nenhum dos jogadores pode passar jogadas, podemos omitir o subscrito Y .

Em muitos jogos, a primeira jogada pode dar algumas vantagens a um dos jogadores, mas, tanto no Jogo da Coloração como no Jogo da Coloração Gulosa, existem grafos em que o jogador que faz a primeira jogada perde, isto é, o segundo jogador possui uma estratégia vencedora. Um exemplo é o grafo da Figura 6, onde temos os vértices x_i e y_i , para $i = 1, \dots, n$, e

cada x_i é vizinho de y_j , para $i = 1, \dots, n$ e $j = 1, \dots, n$, exceto quando $i = j$.

Figura 6 – Quem começar perde, no Jogo da Coloração ou no Jogo da Coloração Gulosa



Fonte: elaborado pelo autor.

Nota: grafo que possui os vértices x_i e y_i , para $i = 1, \dots, n$, e cada x_i é vizinho de y_j , para $i = 1, \dots, n$ e $j = 1, \dots, n$, exceto quando $i = j$.

Considere o grafo da Figura 6 e $C = \{1, 2\}$. Pela simetria do grafo, podemos assumir, sem perda de generalidade, que, na primeira jogada, o vértice x_1 é escolhido e colorido com a cor 1:

1. Se Bob iniciar na variante do Jogo da Coloração, então Alice ganha o jogo com a seguinte estratégia:
 - a) Após Bob colorir x_i com a cor 1, Alice colore y_i com cor 2.
2. Se Bob iniciar na variante do Jogo da Coloração Gulosa, então Alice ganha o jogo com a seguinte estratégia:
 - a) Após Bob colorir x_i (cor 1), Alice colore x_j (cor 1) com $i \neq j$.
3. Se Alice iniciar o jogo, então Bob consegue forçar mais que as duas cores de C (note que ele consegue até n cores) em ambos os jogos com a seguinte estratégia:
 - a) Se Alice colorir x_i , então Bob colore y_i com a mesma cor de x_i para $i = 1, \dots, n$.
 - b) Se Alice colorir y_i , então Bob colore x_i com a mesma cor de y_i para $i = 1, \dots, n$.

Perceba que, nas variantes dos jogos em que Bob inicia, após Alice fazer a segunda jogada, a cor de x_k ($k = 1, \dots, n$) será forçadamente 1, enquanto y_k ($k = 1, \dots, n$) terá forçadamente cor 2.

Perceba também com esse exemplo que o poder de passar jogadas pode ser vantajoso para um jogador, pois, nesse exemplo, basta evitar a primeira jogada para vencer. Apesar disso,

veremos nesta tese algumas classes de grafos em que, mesmo que Bob possa passar jogadas, Alice consegue garantir o número mínimo de cores (número cromático) no Jogo da Coloração Gulosa.

Dizemos que um jogo com dois jogadores é de *informação perfeita* se ambos os jogadores têm pleno conhecimento das jogadas do seu oponente. Se o jogo sempre finaliza após um número finito de jogadas dizemos que o jogo é *finito*. Note que, em grafos finitos, os jogos de coloração definidos nesta seção são finitos, pois o jogo termina após todos os vértices serem coloridos, e que são de informação perfeita, pois cada jogador consegue sempre ver o grafo inteiro e todos os vértices já coloridos, bem como as cores utilizadas em cada vértice já colorido.

A seguir temos o famoso Teorema de Zermelo.

Teorema 1.1.1. (ZERMELO, 1913) *Em um jogo finito de informação perfeita que não possui empate e em que dois jogadores alternam suas jogadas, temos que um dos jogadores possui uma estratégia vencedora.*

Pelo Teorema de Zermelo, concluímos que ou Alice ou Bob possui uma estratégia vencedora tanto para o Jogo de Coloração quanto para o Jogo de Coloração Gulosa e suas variantes definidas anteriormente. Com isso, surge a motivação de investigar parâmetros que envolvem uma estratégia vencedora para um dos jogadores, por exemplo, o menor número de cores tal que Alice possui uma estratégia vencedora. Nesta tese, nós nos concentramos nas versões originais do Jogo de Coloração e do Jogo de Coloração Gulosa.

Seja G um grafo. O *número cromático de jogo* $\chi_g(G)$ e o *número guloso de jogo* $\Gamma_g(G)$ são: a menor cardinalidade do conjunto de cores C no qual Alice possui uma estratégia vencedora no Jogo da Coloração e no Jogo de Coloração Gulosa, respectivamente. Nesta tese focamos nesses dois parâmetros para os jogos de coloração em grafos.

O Jogo da Coloração foi considerado pela primeira vez por Brams no contexto de coloração de mapas e descrito por Gardner em 1981 em sua coluna “*Mathematical Games*” na *Scientific American* (GARDNER, 1981). Permaneceu despercebido até que Bodlaender (BODLAENDER, 1991) reinventou-o em 1991 como “*Jogo de Construção de Colorir*”. Bodlaender deixou a complexidade do problema em aberto. Em suas palavras, “*a complexidade do jogo de construção de colorir é um interessante problema em aberto*”.

Desde então, o Jogo da Coloração tornou-se bem ativo em pesquisas:

- Em 1993, Faigle et al. (FAIGLE *et al.*, 1993) provaram que $\chi_g(G) \leq 4$ em florestas.

- Em 1994, Kierstead e Trotter (KIERSTEAD; TROTTER, 1994) mostraram que $\chi_g(G) \leq 7$ em grafos periplanares (em inglês “*outerplanar graphs*”).
- Em 1999, Dinski e Zhu (DINSKI; ZHU, 1999) provaram que $\chi_g(G) \leq k(k+1)$ para um grafo qualquer com número cromático acíclico k .
- Em 2000, Zhu mostrou que $\chi_g(G) \leq 3k+2$ para k -árvores parciais (ZHU, 2000).
- Em 2007, Sidorowicz (SIDOROWICZ, 2007) mostrou que $\chi_g(G) \leq 5$ em cacti.
- Em 2008, para grafos planares, Zhu (ZHU, 2008) mostrou que $\chi_g(G) \leq 17$.
- Ainda em 2008, Bohman, Frieze e Sudakov (BOHMAN *et al.*, 2008) pesquisaram o comportamento assintótico de $\chi_g(G_{n,p})$ para grafos aleatórios $G_{n,p}$.
- Em 2014, Sekiguchi (SEKIGUCHI, 2014) mostrou que $\chi_g(G) \leq 13$ se a cintura é pelo menos 4.
- Em 2018, Nakprasit et al. (NAKPRASIT; NAKPRASIT, 2018) mostraram que $\chi_g(G) \leq 5$ se a cintura é pelo menos 7.

Apesar disso, pouco progresso foi obtido na complexidade do problema do Jogo da Coloração. Em 2015, Dunn et al. (DUNN *et al.*, 2015) obtiveram um progresso parcial na complexidade do número cromático de jogo em florestas.

O Jogo da Coloração Gulosa e o número guloso de jogo foram introduzidos por Havet e Zhu (HAVET; ZHU, 2013). Eles mostraram que $\Gamma_g(G) \leq 3$ em florestas e $\Gamma_g(G) \leq 7$ em 2-árvores parciais.

Seja G um grafo. Uma *coloração própria* nos vértices G é uma coloração em que cada vértice possui cor distinta dos seus vizinho. Denotamos por $\chi(G)$ o *número cromático* de G que é o menor número de cores de uma coloração própria nos vértices de G . Se seguirmos uma ordem nos vértice de G e colorirmos os vértices sempre com a menor cor possível considerando a coloração própria, então obtemos uma *coloração gulosa*. Denotamos por $\Gamma(G)$ o *número guloso* de G que é o maior número de cores que podem ser usadas em uma coloração gulosa. Claramente, $\chi(G) \leq \chi_g(G)$ e

$$\chi(G) \leq \Gamma_g(G) \leq \Gamma(G). \quad (1.1)$$

De fato, veja que podemos chegar em uma coloração com $\chi(G)$ através do Jogo da Coloração Gulosa, nas jogadas, basta ir colorindo primeiro os vértices de cor 1, depois os vértices de cor 2, ... , até, por fim, colorir os últimos vértices com cor $\chi(G)$. Neste caso podemos pensar que ambos os jogadores seguem a melhor estratégia para Alice. Note que também podemos chegar

em uma coloração com $\Gamma(G)$ através do Jogo da Coloração Gulosa. Neste caso podemos pensar que ambos os jogadores seguem a melhor estratégia para o Bob.

Uma questão interessante é se $\chi_g(G)$ ou $\Gamma_g(G)$ são limitados superiormente por uma função de $\chi(G)$. A resposta para essa pergunta é NÃO, como podemos ver pelo grafo G da Figura 6 que $\chi(G) = 2$, enquanto $\chi_g(G) = \Gamma_g(G) = n$. Outra questão é se $\Gamma_g(G)$ é limitado inferiormente por uma função de $\Gamma(G)$ que, por sua vez, também é respondida com NÃO, uma vez que no grafo G' (grafo da Figura 6 com mais um vértice isolado) temos que $\Gamma(G') = n$, enquanto $\Gamma_g(G') = 2$. Para mais detalhes, veja (HAVET; ZHU, 2013).

Havet e Zhu (HAVET; ZHU, 2013) também propuseram duas questões a respeito do número guloso de jogo:

Problema 5 de (HAVET; ZHU, 2013): $\chi_g(G)$ pode ser limitado superiormente por uma função de $\Gamma_g(G)$?

Problema 6 de (HAVET; ZHU, 2013): é verdade que $\Gamma_g(G) \leq \chi_g(G)$ para um grafo G qualquer?

Em 2015, Krawczyk e Walczak (KRAWCZYK; WALCZAK, 2015) responderam negativamente ao Problema 5 de (HAVET; ZHU, 2013). Em nossos conhecimentos, o Problema 6 de (HAVET; ZHU, 2013) ainda está em aberto, mesmo quando relaxado para: $\Gamma_g(G)$ é limitado superiormente por uma função de $\chi_g(G)$?

Existem poucos resultados para determinação exata de alguns desses parâmetros, mesmo em classes de grafos bem restritas. Em 2008, Bartnick et al. obtiveram valores exatos para o número cromático de jogo do produto cartesiano de K_2 por um caminho, um ciclo ou um grafo completo (BARTNICKI *et al.*, 2008). Em 2013, Havet e Zhu mostraram que $\Gamma_g(G) = \chi(G)$ para cografos (sabe-se que $\Gamma(G) = \chi(G)$ (HAVET; ZHU, 2013) em cografos e pela equação 1.1 temos que $\Gamma_g(G) = \chi(G)$). Nesta tese, estendemos esse resultado provando que $\Gamma_g(G) = \chi(G)$ para superclasses de cografos, como grafos P_4 -esparso, P_4 -tidy e P_4 -laden. Para essas classes de grafos, mostramos também que $\Gamma(G)$ pode ser maior que $\chi(G)$ o quanto quisermos e, além disso, Alice possui uma estratégia vencedora com $\chi(G)$ cores mesmo na variante do Jogo de Coloração Gulosa em que Bob pode começar o jogo e pode passar jogadas.

Em 1991, Bodlaender nomeou o Jogo da Coloração como “Jogo de Construção para Colorir” e em suas palavras afirmou “a complexidade do Jogo de Construção para Colorir é um interessante problema aberto”. Em 2015, Dunn et al. (DUNN *et al.*, 2015) investigaram a complexidade e afirmaram, em suas palavras, que “*mais de duas décadas depois e essa*

questão permanece aberta”. Nesta tese, respondemos essa questão mostrando que ambos os jogos, Jogo da Coloração e Jogo da Coloração Gulosa, são PSPACE-Completos mesmo se o número de cores é igual ao número cromático, isto é, mostramos que decidir se $\chi_g(G) = \chi(G)$ é PSPACE-Completo e decidir se $\Gamma_g(G) = \chi(G)$ também é PSPACE-Completo.

Nossos resultados citados nos dois parágrafos anteriores foram apresentados em 2019 no *Latin & American Algorithms, Graphs and Optimization Symposium* (LAGOS) e publicados em seus anais *Electronic Notes in Theoretical Computer Science* (COSTA *et al.*, 2019) como resumo estendido e no periódico *Theoretical Computer Science* como artigo completo (COSTA *et al.*, 2020). Depois desses resultados, outras variantes do Jogo de Coloração também foram provados PSPACE-Completos, como as variantes $g_{X,Y}$ e cg_X em (LIMA *et al.*, 2022b) e as variantes $gg_{X,Y}$ e cgg_X em (LIMA *et al.*, 2022a), onde $X \in \{Alice, Bob\}$ e $Y \in \{Alice, Bob, ninguém\}$.

1.2 Jogo do Espião

Nesta seção, apresentamos o Jogo do Espião, um jogo de perseguição em grafos introduzido em (COHEN *et al.*, 2016).

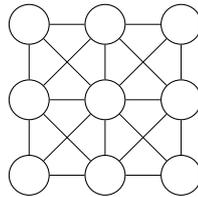
Sejam G um grafo finito e os inteiros $s \geq 1$, $d \geq 0$ e $k \geq 1$ que são a velocidade do espião, a distância de vigilância dos guardas e o número de guardas, respectivamente. O *Jogo do Espião* é um jogo de dois jogadores no grafo G com k guardas e um espião que ocupam alguns vértices de G . Os guardas e o espião podem ocupar o mesmo vértice (inclusive todos os guardas e o espião no mesmo vértice ao mesmo tempo). Um jogador controla os guardas e o outro controla o espião. É um jogo de informação perfeita: cada jogador tem o pleno conhecimento das posições dos elementos controlados pelo outro jogador. Inicialmente, o espião é colocado em algum vértice de G e, em seguida, os k guardas são colocados em alguns vértices de G . No processo do jogo, primeiro o espião pode mover-se ao longo de no máximo s arestas e, em seguida, cada guarda pode mover-se ao longo de no máximo uma aresta. O jogo continua com o revezamento entre movimentos do espião e dos guardas nos vértices do grafo. O espião ganha se, em algum momento do jogo, após a jogada dos guardas, estiver em um vértice a uma distância maior que d de cada guarda. Caso contrário, o jogador que controla os guardas vence o jogo, isto é, sempre existe pelo menos um guarda a uma distância no máximo d do espião.

Podemos considerar que o Jogo do Espião é um jogo finito, pois possui um número finito de configurações em um grafo finito. Por exemplo, podemos considerar que os guardas vencem se, após uma jogada do espião, for repetida uma configuração já jogada. Deste modo,

pelo Teorema de Zermelo 1.1.1, um dos jogadores possui uma estratégia vencedora. Seja então o número de guarda $gn_{s,d}(G)$ o menor número k de guardas tal que os guardas possuem uma estratégia vencedora para “vigiar” o espião com velocidade s a uma distância d , contra qualquer estratégia do espião.

Como exemplo do Jogo do Espião considere o grafo da Figura 7. Suponha que temos dois guardas. Considere que o espião é E , os dois guardas são 1 e 2, velocidade do espião $s = 2$ e que a distância de vigilância é $d = 0$. Veja que os guardas possuem uma estratégia vencedora que consiste em deixar um dos guardas no vértice central e o outro guarda no vértice em que o espião estiver. Caso o espião mova-se para o vértice central então os dois guardas movem-se para o vértice central. Caso o espião mova-se para um vértice que não seja o vértice central, um guarda no vértice central move-se para o vértice do espião e o outro guarda move-se para o vértice central. A Figura 8 mostra possíveis jogadas em que os guardas seguem essa estratégia no grafo da Figura 7.

Figura 7 – Grafo exemplo para o Jogo do Espião

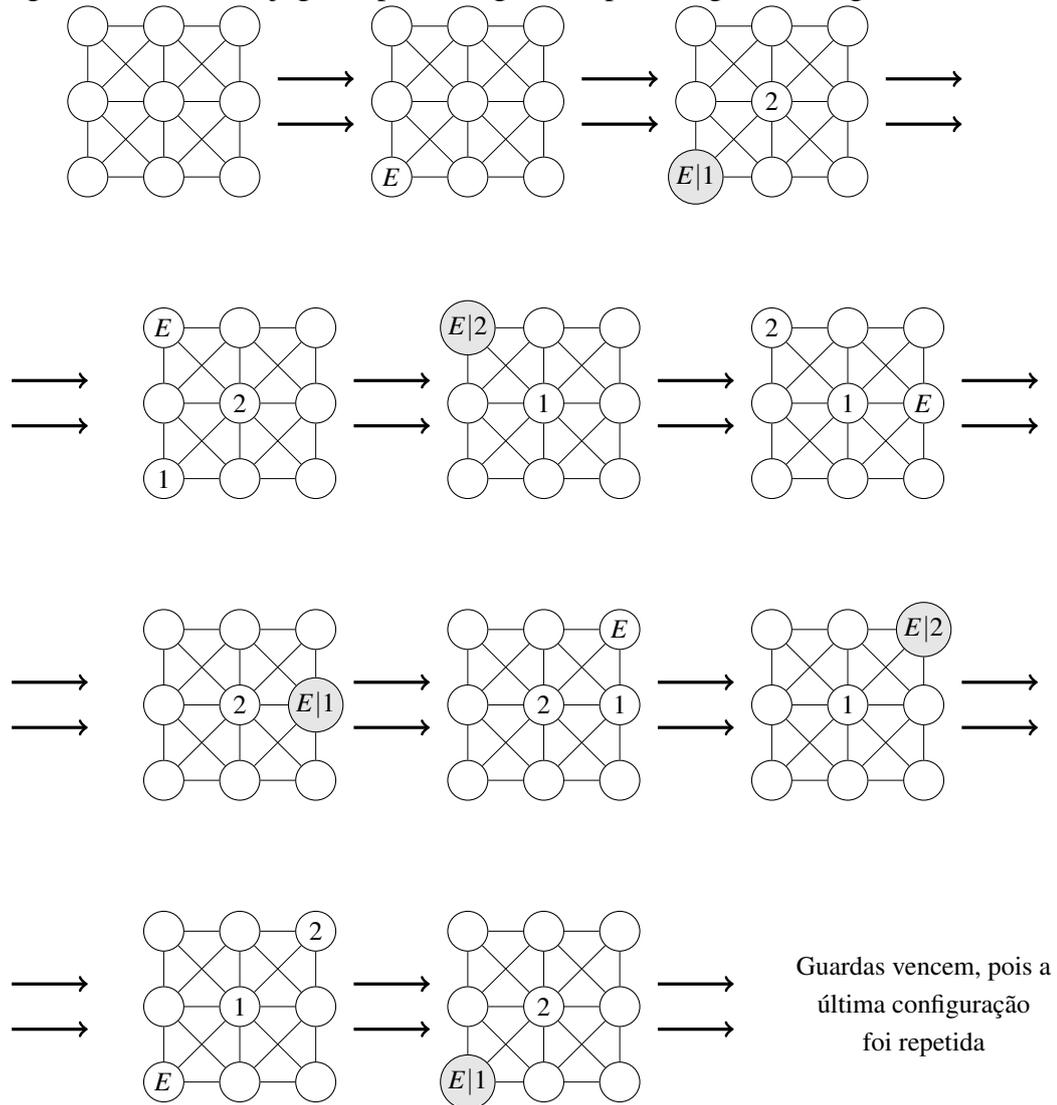


Fonte: elaborado pelo autor.

Ao introduzir o Jogo do Espião em 2016 (COHEN *et al.*, 2016), os autores apresentaram ainda uma variante em que os guardas se posicionam primeiro. Neste caso, uma vez que o espião pode escolher o vértice inicial a uma distância maior que d de qualquer guarda, as regras do jogo precisam ser ligeiramente modificadas: o controlador dos guardas vence o jogo se, depois de um número finito de jogadas, é garantido que o espião permaneça sempre a uma distância máxima d de pelo menos um guarda. Do mesmo modo, o jogador que controla o espião vence o jogo se sempre puder alcançar um vértice a uma distância maior que d de cada guarda. Foi provado que, se o espião vence contra k guardas quando é o primeiro a jogar, então ele também vence contra k guardas quando é o segundo a jogar. A volta nem sempre é verdadeira.

Um jogo clássico relacionado ao Jogo do Espião é o *Jogo de Polícia e Ladrão* (em inglês “*Cops and Robber*”) que é equivalente ao Jogo do Espião com velocidade $s = 1$ nesta versão em que o espião é colocado depois dos guardas e o jogo termina quando um guarda ocupar o mesmo vértice do espião (pois o espião não poderá mais fugir deste guarda, já que

Figura 8 – Possíveis jogadas para o Jogo do Espião no grafo da Figura 7



Fonte: elaborado pelo autor.

Nota: considere que o espião é E , os dois guardas são 1 e 2, velocidade do espião $s = 2$ e que a distância de vigilância é $d = 0$.

ambos tem a mesma velocidade). Neste jogo, o espião é o ladrão, enquanto os guardas são os policiais. Para velocidade $s > 1$, a equivalência não é verdadeira e os jogos são significativamente diferentes. Por isso, costuma-se considerar a velocidade do espião $s \geq 2$ no Jogo do Espião. De modo análogo ao número de guarda, no Jogo de Polícia e Ladrão temos o *número policial* (em inglês “*cop number*”) $cn(G)$ que é o menor número de policiais necessários para vencer em G (AIGNER; FROMME, 1984). Para mais informações, aconselhamos o excelente livro “*The Game of Cops and Robbers on Graphs*” (BONATO; NOWAKOVSKI, 2011).

Provou-se em (COHEN *et al.*, 2018) que, se k guardas vencem o Jogo do Espião na versão em que o espião é o primeiro a jogar, então $k + cn(G) - 1$ guardas vencem o Jogo do Espião na versão em que o espião é o segundo a jogar (lembre-se que $cn(G)$ é o número policial

de G no Jogo de Polícia e Ladrão). Existem exemplos em que esse valor é ótimo. Por exemplo, em qualquer grafo G , $gn_{1,0}(G) = 1$, pois um guarda pode ser colocado na posição inicial do espião e segui-lo para sempre. Por outro lado, na versão em que os guardas são colocados primeiro, o Jogo do Espião com velocidade $s = 1$ e distância $d = 0$ de vigilância é equivalente ao Jogo de Polícia e Ladrão e portanto, sendo $k = 1$, são necessários $k + cn(G) - 1 = cn(G)$ guardas para vencer.

Outro jogo conhecido bem relacionado ao Jogo do Espião é o *Jogo da Dominação Eterna* (GODDARD *et al.*, 2005; GOLDWASSER; KLOSTERMEYER, 2008; KLOSTERMEYER; MACGILLIVRAY, 2009; KLOSTERMEYER; MYNHARDT, 2011). Nesse jogo, k defensores devem ocupar alguns vértices de um grafo G . Em cada jogada, um atacante escolhe um vértice v de G e os defensores podem mover-se ao longo de uma aresta de modo que pelo menos um defensor esteja a uma distância de no máximo d (inteiro dado) de v . Esse jogo é equivalente ao Jogo do Espião quando a velocidade s do espião é pelo menos o diâmetro do grafo (ou seja, o espião consegue atingir qualquer vértice em uma única jogada).

Cohen *et al.* (COHEN *et al.*, 2020) mostraram que o número de guarda é computável em tempo polinomial para árvores. Em outro trabalho, Cohen *et al.* (COHEN *et al.*, 2018) mostraram que o número de guarda é NP-difícil para grafos gerais, mas para grafos direcionados é PSPACE-difícil mesmo em grafos direcionados acíclicos. No caso de grafos direcionados, a definição é levemente diferente: o espião e os guardas movem-se seguindo a direção das arestas, mas a distância de vigilância é considerada no grafo não direcionado subjacente (ignorando as direções das arestas).

Voltando aos grafos não direcionados, (COHEN *et al.*, 2020) obtiveram o valor exato do número de guarda para caminhos P_n com n vértices para qualquer velocidade $s \geq 2$ e distância $d \geq 0$:

$$gn_{s,d}(P_n) = \left\lceil \frac{n}{2d + 2 + \lfloor \frac{2d}{s-1} \rfloor} \right\rceil.$$

Os autores deixaram ainda como problema futuro para investigação do número de guarda em grades: “*muitas questões permanecem abertas, como a caracterização do número de guarda em outras classes de grafos, por exemplo, em grades*”.

Nesta tese iniciamos o estudo do número de guarda em grades Rei e em produtos de grafos, como o produto cartesiano, o produto lexicográfico e o produto forte e mostramos um limitante superior para o número de guarda do produto forte de dois grafos quaisquer.

Obtemos também o valor exato para o produto lexicográfico de dois grafos quaisquer. Além disso, provamos que é possível decidir em tempo polinomial $O(n^{3k+2})$ se o espião possui uma estratégia vencedora quando o número de guardas é igual ao número fixo k , para toda velocidade $s \geq 2$ e distância $d \geq 0$. Em outras palavras, o Jogo do Espião está em XP quando o parâmetro é o número de guardas. Como resultado negativo, provamos que o Jogo do Espião é W[2]-difícil mesmo em grafos bipartidos quando o parâmetro é o número de guardas, para toda velocidade $s \geq 2$ e distância $d \geq 0$, estendendo, deste modo, os resultados de (COHEN *et al.*, 2018).

Nossos resultados citados acima acerca do Jogo do Espião foram apresentados em 2021 no *International Computing and Combinatorics Conference (COCOON)* (COSTA *et al.*, 2022a) e publicados na *Theoretical Computer Science* como artigo completo (COSTA *et al.*, 2022b).

1.3 Organização do texto

Esta tese está organizada como segue. No Capítulo 2, apresentamos alguns conceitos básicos em Teoria dos Grafos, Teoria da Complexidade clássica e Teoria da Complexidade Parametrizada que são úteis para os demais capítulos.

O Capítulo 3 se concentra no problema do Jogo da Coloração. Nesse capítulo, respondemos a questão de Bodlaender de 1991, mostrando que determinar o número cromático de jogo é PSPACE-completo e que o mesmo ocorre para o número guloso de jogo, isto é, mostramos que ambos os jogos, Jogo da Coloração e Jogo da Coloração Gulosa, são PSPACE-completos mesmo se o número de cores é o número cromático. Em seguida, mostramos que número guloso de jogo é igual ao número cromático para grafos split e várias classes de cografos, estendendo um resultado de Havet e Zhu de 2013.

O Capítulo 4 se concentra no Jogo do Espião. Nesse capítulo, mostramos um limite superior para o número de guarda no produto forte de dois grafos gerais e, em seguida, mostramos exemplos de grafos grades Rei nos quais o número de guarda é igual ao limite e outros exemplos de grades Rei em que o número de guarda é estritamente menor que o limite. Obtemos também o valor exato do número de guarda no produto lexicográfico de dois grafos gerais. Como resultado positivo mostramos que, se o número de guardas é um inteiro fixo k , então o Jogo do Espião é solucionável em tempo $O(n^{3k+2})$ para qualquer velocidade $s \geq 1$ e qualquer distância de vigilância $d \geq 0$ por um algoritmo XP. Ou seja, o Jogo do Espião está em XP quando o parâmetro é o número de guardas. Utilizando esse algoritmo XP obtemos um algoritmo FPT para grafos

com poucos P_4 's. Como resultado negativo, mostramos que o Jogo do Espião é $W[2]$ -difícil, mesmo em grafos bipartidos, quando o parâmetro é o número de guardas, para cada velocidade $s \geq 2$ e cada distância de vigilância $d \geq 0$, estendendo os resultados de Cohen et al. de 2016.

No Capítulo 5, descrevemos nossas considerações finais acerca dos objetos de estudos desta tese.

2 CONCEITOS BÁSICOS

Neste capítulo, apresentamos alguns conceitos básicos em Teoria dos Grafos, Teoria da Complexidade Clássica e Teoria da Complexidade Parametrizada que são necessários para uma boa compreensão dos demais capítulos.

2.1 Introdução à Teoria dos Grafos

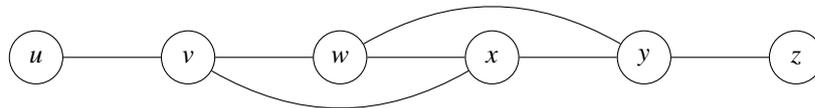
Nesta seção, apresentamos algumas definições e resultados básicos em Teoria dos Grafos. Para outras definições e resultados veja (BONDY; MURTY, 2008).

2.1.1 Definições básicas

Um grafo simples G consiste de um par ordenado $G = (V, E)$, onde V (ou $V(G)$) é um conjunto de elementos, chamados vértices, e E (ou $E(G)$) é um conjunto de elementos, chamados arestas, onde uma aresta é um subconjunto de dois vértices distintos de V , chamados de extremidades da aresta. Um grafo simples $G = (V, E)$ é *finito* quando V é finito. Nesta tese, trabalharemos apenas com grafos simples finitos e, por simplicidade, usaremos apenas o termo grafo. Também por simplicidade, podemos denotar uma aresta $a = \{x, y\}$ (aresta com extremidade nos vértices x e y) apenas como $a = xy$.

Em uma representação gráfica de um grafo, os vértices são representados como círculos e as arestas são representadas como linhas (contínuas e não necessariamente retas) ligando as duas extremidades da aresta. A Figura 9 mostra o exemplo do grafo $G = (V, E)$, onde $V = \{u, v, w, x, y, z\}$ e $E = \{uv, vw, vx, wx, wy, xy, yz\}$.

Figura 9 – Exemplo de um grafo



Fonte: elaborado pelo autor.

Seja $G = (V, E)$ um grafo. Dizemos que dois vértices $x, y \in V(G)$ são *vizinhos* (ou *adjacentes*) em G caso $xy \in E(G)$. A *vizinhança* (ou os *vizinhos*) de um vértice $x \in V(G)$ é o conjunto $N(x) = \{y \in V(G) \mid xy \in E(G)\}$ e a *vizinhança fechada* de x é o conjunto $N[x] = N(x) \cup \{x\}$. A *vizinhança de um conjunto* $A \subseteq V(G)$ é o conjunto $N(A) = \bigcup_{x \in A} N(x) - A$ e a *vizinhança fechada do conjunto* A é o conjunto $N[A] = N(A) \cup A$. Por exemplo, para o grafo da Figura 9

temos que $N(w) = \{v, x, y\}$, $N[w] = \{w, v, x, y\}$, $N(\{w, x\}) = \{v, y\}$ e $N[\{w, x\}] = \{w, x, v, y\}$.

Sejam $G = (V, E)$ um grafo e um vértice $v \in V(G)$. Denotamos por $d(v) = |N(v)|$ o *grau* do vértice v . Denotamos por $\Delta(G)$ o *grau máximo* de G , isto é, $\Delta(G) = \max\{d(v) \mid v \in V(G)\}$. Analogamente, $\delta(G)$ denota o *grau mínimo* de G , isto é, $\delta(G) = \min\{d(v) \mid v \in V(G)\}$. Por exemplo, para o grafo G' da Figura 9, temos que $d(u) = 1$, $d(w) = 3$, $d(v) = 3$, $\Delta(G') = 3$ e $\delta(G') = 1$.

O complemento \overline{G} de um grafo G é o grafo com mesmo conjunto de vértices $V(\overline{G}) = V(G)$ tal que $uv \in E(\overline{G})$ se e só se $uv \notin E(G)$ para quaisquer dois vértices distintos $u, v \in V(G)$. Um subconjunto de vértices de um grafo é dito *clique* quando são dois a dois adjacentes. Denotamos por K_n o grafo, chamado *grafo completo*, que contém n vértices e que $V(K_n)$ é uma clique. Um subconjunto de vértices de um grafo é dito *estável* (ou *independente*) quando são não-adjacentes dois a dois. Dizemos que subconjunto de arestas é dito *conjunto independente de arestas* (ou *emparelhamento*) se suas arestas não possuem extremidades em comum. Um emparelhamento M é dito *perfeito*, se todo vértice do grafo é extremidade de alguma aresta de M . Dado um grafo $G = (V, E)$, denotamos por $\omega(G)$ o tamanho de um maior subconjunto de vértices de G que formam uma clique. Por exemplo, para o grafo G' da Figura 9, temos que $\omega(G') = 3$, pois temos o conjunto $\{w, x, y\}$ que é clique e não há outra com tamanho quatro em G' .

Sejam os grafos $G = (V, E)$ e $H = (V', E')$. Dizemos que H é *subgrafo* de G caso $V' \subseteq V$ e $E' \subseteq E$. Por exemplo, temos que o grafo $P = (\{u, v, w, x, y\}, \{uv, vw, wx, xy\})$ é um subgrafo do grafo da Figura 9. Dizemos que H é um *subgrafo induzido* de G se H é subgrafo de G e, para cada par de vértices x e y de H , $xy \in E(H)$ se e somente se $xy \in E(G)$. Veja que o grafo P não é subgrafo induzido do grafo da Figura 9, pois não possui as arestas vx e wy . Caso $V(H)$ seja uma clique e H subgrafo de G , dizemos que $V(H)$ *induz* uma clique em G . Seja $X \subseteq V(G)$. Denotamos por $G[X]$ o subgrafo induzido de G por X . Por exemplo, se G' é o grafo da Figura 9 e $X = \{w, x, y\}$, então $G'[X] = (\{w, x, y\}, \{wx, xy, yw\})$ é um grafo completo de tamanho 3 e podemos dizer que X induz uma clique de tamanho 3.

Um *caminho* $P = (V, E)$ é um grafo em que todos os seus vértices podem ser colocados em uma sequência $(v_0, v_1, v_2, \dots, v_k)$ de modo que existem apenas as arestas $v_i v_{i+1} \in E$, para $i = 0, 1, \dots, k-1$. Podemos denotar P ainda por $v_0 \cdots v_k$. Dizemos que v_0 e v_k são *vértices extremos* ou as *extremidades* do caminho P ; e dizemos que v_i é vértice interno para $i = 1, \dots, k-1$. No grafo G' da Figura 9, temos os subgrafos $P' = uvwxy$, $P'' = uvxy$ e $P''' = uvxwy$ (dentre outros)

que são caminhos e, por simplicidade, dizemos que existe um caminho entre u e y em G' (pois existe P' tal que u e y são extremidades de P' em G'). Note que o subgrafo induzido por $V(P')$ não é um caminho, pois possui 6 arestas enquanto possui 5 vértices (segundo a definição um caminho com 5 vértices deve possuir exatamente 4 arestas). O *tamanho* (ou *comprimento*) de um caminho é igual ao número de vértices que ele possui menos um, isto é, sua quantidade de arestas. Denotamos por P_n um caminho que possui n vértices.

Um *ciclo* $C = (V, E)$ (ou *caminho fechado*) é um grafo com 3 ou mais vértices, em que todos os seus vértices podem ser colocados em uma sequência $(v_0, v_1, v_2, \dots, v_k)$ de modo que existem apenas as arestas $v_0v_k \in E$ e $v_iv_{i+1} \in E$, para $i = 0, 1, \dots, k-1$. O *tamanho de um ciclo* é igual ao número de seus vértices. Veja que um ciclo de tamanho n possui n vértices, enquanto um caminho de tamanho n possui $n+1$ vértices. Denotamos por C_n um ciclo de tamanho n . Por exemplo, para o grafo G' representado na Figura 9, temos os ciclos $G[\{v, w, x\}]$ e $G[\{w, x, y\}]$ que são dois C_3 .

Dizemos que um vértice v de um grafo G é *universal* se ele é adjacente a todos os outros vértices de G . Um grafo é *conexo* caso exista um caminho entre cada par de seus vértices. Um grafo é *desconexo* caso não seja conexo. Quando um grafo é desconexo, seus subgrafos conexos maximais são chamados de *componentes conexas*. A *distância* entre dois vértices em um grafo é o tamanho mínimo de um caminho entre eles. O *diâmetro* de um grafo conexo G é a maior distância possível entre dois vértices quaisquer do grafo.

2.1.2 Coloração em Grafos

Um dos principais problemas que motivou o estudo de coloração em grafos foi o PROBLEMA DAS QUATRO CORES, que consiste em saber se é possível colorir qualquer mapa geográfico com apenas quatro cores, onde regiões vizinhas têm que possuir cores distintas. Modelando o problema para grafos, podemos pensar que as regiões são os vértices de um grafo e que temos uma aresta entre dois vértices que correspondem a regiões vizinhas no mapa. Neste problema estamos interessados em colorir os vértices.

Uma k -coloração de um grafo G é uma função $c : V(G) \mapsto S$, onde S é um conjunto de inteiros de 1 a k , que chamamos de *cores*. Caso quaisquer dois vértices adjacentes tenham cores distintas na k -coloração, dizemos que ela é *própria* (ou seja, cada cor representa um conjunto independente do grafo).

Seja G um grafo. Definimos o *número cromático* de G , denotado por $\chi(G)$, o menor

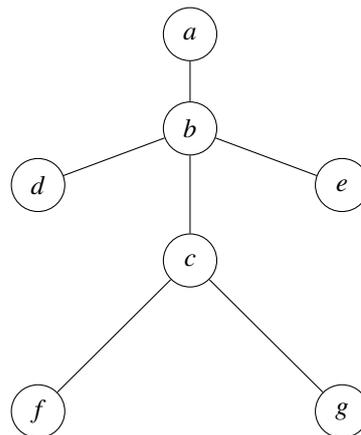
k tal que existe uma k -coloração própria para G . Definimos o *número cocromático* de G o menor k tal que existe uma k -coloração (não necessariamente própria) para G tal que cada um dos seus subconjuntos de vértices coloridos com a mesma cor representa um conjunto independente ou uma clique de G .

2.1.3 Classes de grafos

Nesta seção definimos algumas classes de grafos.

Uma *árvore* é um grafo conexo e que não possui ciclos. Se um grafo T é uma árvore, então é sabido que $|E(T)| = |V(T)| - 1$. Um vértice em uma árvore é também chamado de *nó*. Os nós em uma árvore que possuem apenas um vizinho são chamados de *folhas* e os demais são chamados de *nós internos*. Podemos escolher um nó r , chamado nó raiz, em uma árvore e chamá-la de *árvore enraizada em r* . Seja T uma árvore enraizada em r e um vértice $v \in V(T)$. Um *descendente* do nó v é um nó u tal que v é um vértice interno ao caminho entre os vértices r e u (note que temos apenas um único caminho entre r e u , uma vez que se existem dois caminhos distintos teríamos um ciclo e logo T não seria uma árvore). Um *filho* de v é um descendente que é vizinho de v . Se u é filho de v , dizemos que v é *pai* de u . Seja T' a árvore da Figura 10 enraizada no vértice a . Temos que c , d e e são filhos de b e que a é pai de b .

Figura 10 – Uma árvore enraizada no vértice a



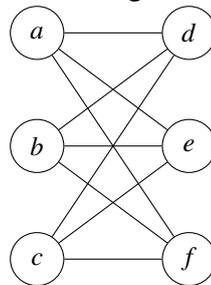
Fonte: elaborado pelo autor.

Um *cacto* é um grafo em que cada uma de suas arestas está em no máximo um ciclo. Portanto toda árvore é um cacto, uma vez que não possui ciclos.

Um grafo $G = (V, E)$ é dito *bipartido* se V pode ser particionado em dois conjuntos (X, Y) tais que a vizinhança de qualquer vértice em X (resp. Y) está contida em Y (resp. X). Na Figura 11, temos um grafo bipartido cujas partes são $A = \{a, b, c\}$ e $B = \{d, e, f\}$. Note

ainda que esse grafo possui o maior número de arestas possível para um grafo bipartido, pois se adicionarmos alguma aresta o grafo torna-se não bipartido. Deste modo, o grafo da Figura 11 é chamado de grafo *bipartido completo*. Denotamos por $K_{m,n}$ o grafo bipartido completo com partes A e B tais que $m = |A|$ e $n = |B|$. O grafo da Figura 11 é o grafo $K_{3,3}$.

Figura 11 – Um grafo bipartido

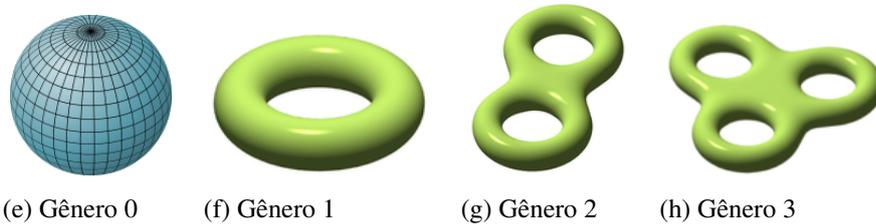


Fonte: elaborado pelo autor.

Um grafo $G = (V, E)$ é *planar* caso exista uma representação gráfica de G no plano Cartesiano sem cruzamento de arestas. Veja que o grafo da Figura 10 é um grafo planar, pois a própria figura é uma representação gráfica sem cruzamento de arestas. Também é fácil ver que o grafo K_4 (grafo completo com 4 vértices) é planar. Já o grafo $K_{3,3}$ da Figura 11 e o grafo K_5 não são planares, pois não há representação gráfica deles no plano cartesiano sem cruzamento de arestas.

Dado um inteiro $k \geq 0$, seja S_k a superfície em \mathbb{R}^3 que possui k buracos. Dizemos que k é o *gênero* (“genus” em inglês) da superfície S_k . A Figura 12 mostra exemplos das superfícies S_0 , S_1 , S_2 e S_3 , também chamadas de esfera, toro, 2-toro e 3-toro respectivamente.

Figura 12 – Superfícies em \mathbb{R}^3



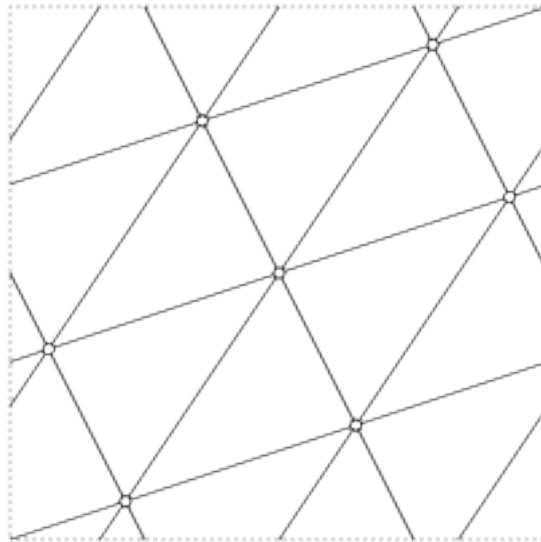
Fonte: Wikipedia (2021).

O *gênero* de um grafo é o menor k tal que o grafo pode ser representado na superfície S_k de modo que não exista cruzamento de arestas. Sabe-se que todo grafo planar pode ser representado na superfície de uma esfera (S_0) sem cruzamento de arestas e, portanto, todo grafo planar possui gênero 0. Sabe-se que todo grafo possui um gênero: intuitivamente, o grafo pode

ser representado numa esfera e, para cada cruzamento de aresta, faz-se um “buraco” na superfície com o intuito de remover tal cruzamento.

Um grafo *toroidal* é um grafo com gênero 1, isto é, pode ser representado na superfície S_1 (toro) sem cruzamento de arestas. Um grafo toroidal bem conhecido é o K_7 , o qual pode ser desenhado num quadrado como na Figura 13 com a ideia de juntar os lados opostos do quadrado para obter o toro com o K_7 desenhado sem cruzamento de arestas.

Figura 13 – K_7 no quadrado



Fonte: Steven (2008).

2.1.4 Grafos com poucos P_4 's

Dois grafos são *isomorfos* se existe uma correspondência biunívoca (um-a-um) entre os vértices dos dois grafos de modo preserve a relação entre vértices vizinhos. Dizemos que um grafo G é livre de um grafo H quando H não é isomorfo a nenhum subgrafo de G . Um *cografo* é um grafo livre de P_4 induzido. Esta classe foi introduzida por (CORNEIL *et al.*, 1981). Um grafo G é P_4 -*esparso* se cada conjunto de cinco vértices em G induz no máximo um P_4 . Esta classe foi introduzida por (JAMISON; OLARIU, 1992).

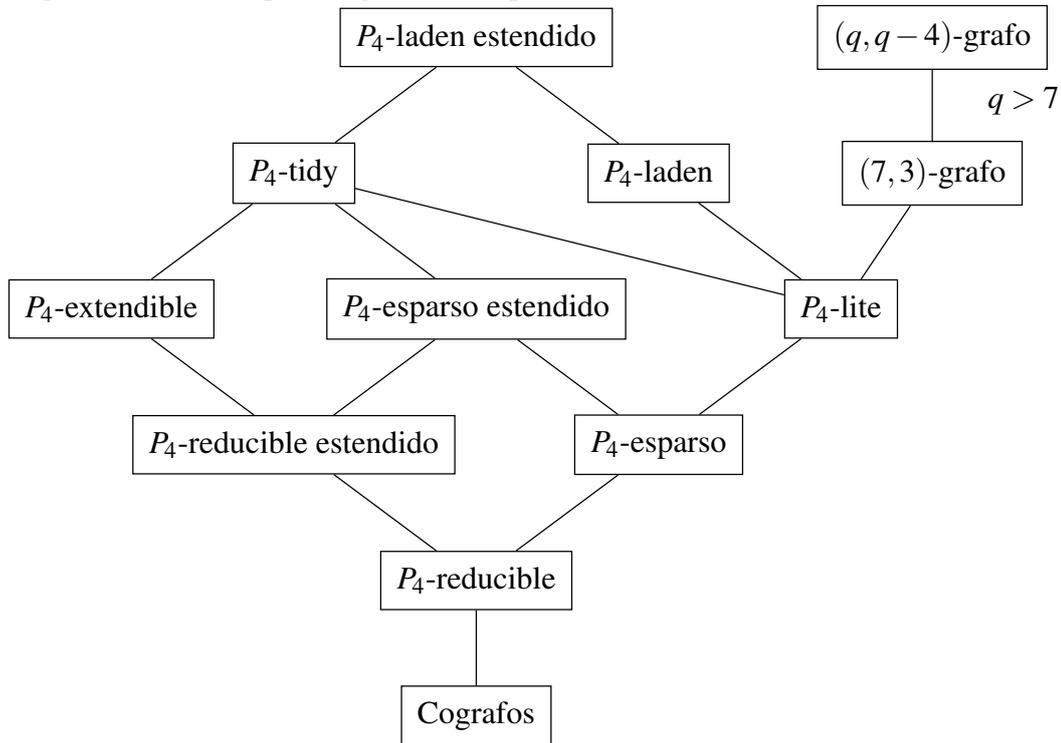
Um grafo G é um $(q, q - 4)$ -grafo para algum inteiro $q \geq 4$ se cada subconjunto com q vértices induz no máximo $q - 4$ P_4 's. Cografos e grafos P_4 -esparso são exatamente $(4, 0)$ -grafos e $(5, 1)$ -grafos.

Dizemos que um grafo é P_4 -*laden estendido* se qualquer subgrafo induzido com no máximo seis vértices e que contém mais de dois P_4 's induzidos é livre de $\{2K_2, C_4\}$ (grafo que não possui dois K_2 independentes e nem C_4 como subgrafo induzido, também conhecidos como

grafos pseudo-split). Essa classe de grafos foi introduzida em (GIAKOUMAKIS, 1996).

A motivação para desenvolver algoritmos para grafos P_4 -laden estendidos e $(q, q-4)$ -grafos está no fato de que eles estão no topo de uma hierarquia amplamente estudada que contém classes contendo grafos com poucos P_4 's, incluindo cografos, P_4 -esparsos, P_4 -lite, P_4 -laden e P_4 -tidy. Veja a Figura 14. Sendo assim, resolver problemas interessantes de modo eficiente para grafos P_4 -laden estendidos e $(q, q-4)$ -grafos implica imediatamente em algoritmos eficientes para todas essas classes.

Figura 14 – Hierarquia de grafos com poucos P_4 's



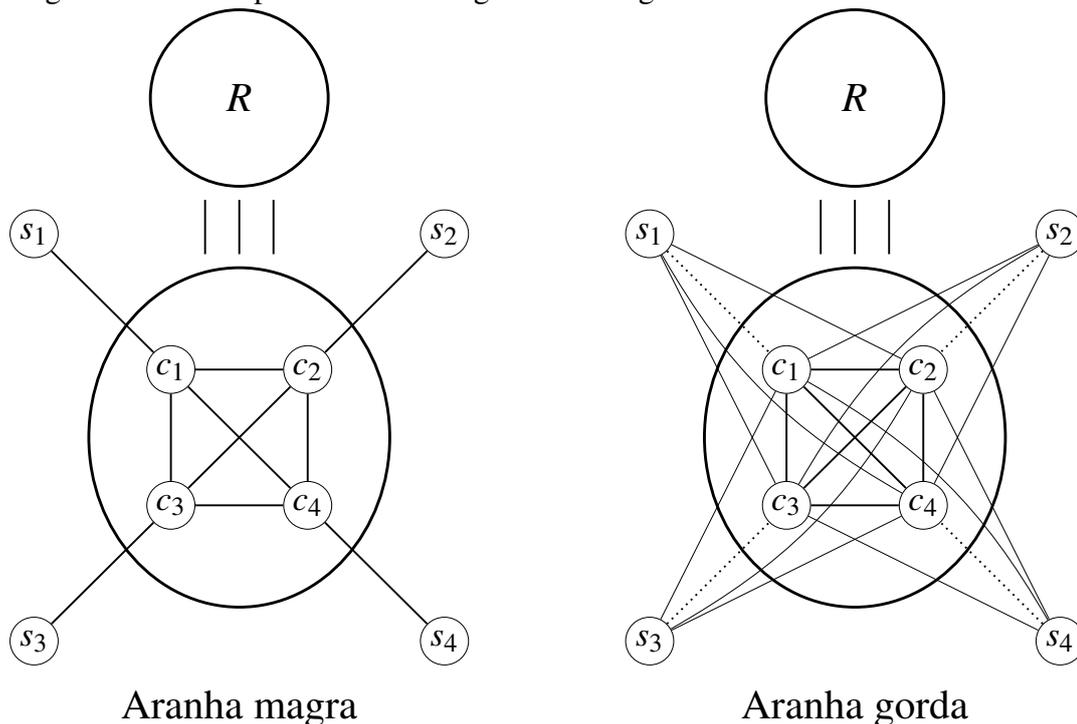
Fonte: elaborado pelo autor.

Todas essas classes possuem decomposições estruturais importantes em termos de operações como união disjunta, junção, aranha, quase-aranha e pseudo-split. Sejam os grafos $G_1 = (V_1, E_1)$ e $G_2 = (V_2, E_2)$ com conjuntos de vértices disjuntos, a *união disjunta* de G_1 e G_2 é o grafo $G_1 \cup G_2 = (V_1 \cup V_2, E_1 \cup E_2)$ e a *junção* de G_1 e G_2 é o grafo $G_1 \vee G_2 = (V_1 \cup V_2, E_1 \cup E_2 \cup \{uv \mid u \in V_1, v \in V_2\})$. Sabe-se que todo cografo com mais de um vértice é uma união disjunta ou junção de dois cografos (CORNEIL *et al.*, 1981).

Um grafo (dito *aranha*) admite uma decomposição chamada de *aranha* se seu conjunto de vértices admite uma partição (R, C, S) , onde $C = \{c_1, \dots, c_k\}$ e $S = \{s_1, \dots, s_k\}$, com $k \geq 2$, são respectivamente uma clique e um conjunto estável; s_i é adjacente a c_j se e somente se $i = j$ (uma aranha magra), ou s_i é adjacente a c_j se e somente se $i \neq j$ (uma aranha gorda); e

cada vértice de R é adjacente a cada vértice de C e não-adjacente a cada vértice de S . Veja na Figura 15 um exemplo de aranha magra e outro de aranha gorda. Note que o complemento de uma aranha magra é uma aranha gorda, e vice-versa. Um grafo é dito *quase-aranha* se é obtido de um grafo aranha pela substituição de no máximo um vértice de $C \cup S$ por um K_2 ou um $\overline{K_2}$ (mantendo a mesma vizinhança do vértice que foi trocado). Claramente, cada aranha é uma quase-aranha.

Figura 15 – Exemplo de aranha magra e aranha gorda



Fonte: elaborado pelo autor.

Nota: a esquerda temos uma aranha magra e à direita uma aranha gorda. Para ambas considere (R, C, S) , onde $C = \{c_1, c_2, c_3, c_4\}$ e $S = \{s_1, s_2, s_3, s_4\}$.

Jamison e Olariu (JAMISON; OLARIU, 1992) provaram que, se um grafo G com mais de um vértice é P_4 -esparso, então G é a união disjunta ou a junção de dois grafos P_4 -esparso, ou G é uma aranha (R, C, S) tal que $G[R]$ é um grafo P_4 -esparso.

Dizemos que um grafo G é p -conectado (caminho conectado), se, para cada partição de $V(G)$ em X e Y , existe um P_4 com vértices em ambas as partes X e Y (ou seja, tem vértices em X e também em Y). Uma p -componente é um subgrafo p -conectado maximal. Dizemos que um grafo p -conectado H é *separável*, se seus vértices possuem uma bipartição (H_1, H_2) tal que cada P_4 com vértices em ambas partes da bipartição possuem seus vértices extremos em H_1 e os outros (vértices internos do P_4) estão em H_2 . Perceba que, se G é uma aranha (R, C, S) , então $G[C \cup S]$ é uma p -componente separável com bipartição (S, C) .

Dizemos que um grafo G é *pseudo-split* se seu conjunto de vértices possui uma partição (R, C, S) tal que C induz uma clique, S induz um conjunto independente e, se R é não vazio, então R induz um C_5 e cada vértice de R é adjacente a cada vértice de C e é não-adjacente a cada vértice de S . Note que o complemento de um pseudo-split é também um pseudo-split.

O teorema a seguir mostra uma decomposição para grafos P_4 -laden estendidos, a qual pode ser obtida em tempo $O(m+n)$.

Teorema 1. (GIAKOUMAKIS, 1996) *Um grafo G com mais de um vértice é P_4 -laden estendido se e somente se exatamente uma das condições a seguir é satisfeita:*

- (a) G é a união disjunta ou a junção de dois grafos não vazios que são P_4 -laden estendidos.
- (b) G é um quase-aranha ou um grafo pseudo-split (R, C, S) tal que R é vazio ou $G[R]$ é um grafo P_4 -laden estendido;
- (c) G é isomorfo a C_5 , P_5 ou $\overline{P_5}$.

2.2 Introdução à Teoria da Complexidade

Um *problema de decisão* é um problema que, dada uma instância, retorna SIM ou NÃO a uma pergunta sobre a instância. Por exemplo, podemos definir o problema de decisão para o problema da cobertura de vértices do seguinte modo:

Problema COBERTURA DE VÉRTICES

Instância: um grafo G e inteiro positivo k .

Pergunta: existe um subconjunto S com no máximo k vértices de G de modo que cada aresta de G possui alguma extremidade em S ?

Definimos a *Classe P* como o conjunto dos problemas de decisão que possuem algoritmos polinomiais que os resolvam.

Definimos a *Classe NP* como o conjunto dos problemas de decisão que são “verificáveis” em tempo polinomial. Em outras palavras, se for dado um “certificado” (ou “prova”) de que a instância tem resposta SIM no problema de decisão, é possível verificar em tempo polinomial que o “certificado” é válido (ou seja, com o certificado temos a prova que a instância tem de fato resposta SIM). Como exemplo, mostramos abaixo que o problema COBERTURA DE VÉRTICES pertence a classe NP

Lema 1. (GAREY; JOHNSON, 1979) COBERTURA DE VÉRTICES pertence a classe NP.

Demonstração. Para certificado de uma instância (G, k) , considere um subconjunto S de vértices do grafo G . O algoritmo verificador para o certificado S deve verificar:

- (a) se $|S| \leq k$ e
- (b) se toda aresta de G possui pelo menos uma extremidade em S .

Para o item (a), basta contar os elementos de S , o que leva tempo $O(k) = O(n)$. Para o item (b), basta percorrer todas as arestas de G (existem $O(n^2)$ arestas) e verificar se alguma de suas extremidades está em S ($|S| = k \leq n$) o que resulta num tempo total de $O(kn^2) = O(n^3)$.

Logo podemos verificar se S é cobertura para G em tempo polinomial $O(n) + O(n^3) = O(n^3)$ e, portanto, COBERTURA DE VÉRTICES pertence a NP. \square

Um problema de decisão A se *reduz polinomialmente* a um problema de decisão B (denotamos $A \leq_p B$ e dizemos que A se reduz polinomialmente a B) se existe uma função f (algoritmo) polinomial que, para cada instância I de A , obtém uma instância $f(I)$ de B de modo que: I é SIM em A se e somente se $f(I)$ é SIM em B .

Um problema de decisão B é NP-difícil se, para todo $A \in \text{NP}$, temos que $A \leq_p B$, isto é, qualquer problema em NP se reduz polinomialmente a B . Um problema é NP-completo se é NP-difícil e se pertence a NP. Em resumo:

1. B é NP-difícil se $A \leq_p B$ para todo $A \in \text{NP}$ e
2. B é NP-completo se B é NP-difícil e $B \in \text{NP}$.

Um resultado clássico muito importante na Teoria da Complexidade é o que se segue.

Teorema 2. (GAREY; JOHNSON, 1979) *Se um problema B é NP-completo e $B \in \text{P}$, então $\text{P} = \text{NP}$. Além disso, se $\text{P} = \text{NP}$, então todo problema NP-completo está em P .*

A questão “ $\text{P} = \text{NP}$?” é uma das mais importantes e antigas da Ciência da Computação, que continua sem resposta. É também um dos 7 problemas do milênio do Instituto Clay de Matemática, com um prêmio de 1 milhão de dólares para quem resolvê-lo. De acordo com o Teorema 2, uma forma de resolver tal questão é mostrar a existência de um problema $B \in \text{P}$ que seja NP-completo. Mostrando-se que não existe um problema $B \in \text{P}$ que seja NP-completo podemos concluir, pelo Teorema 2, que $\text{P} \neq \text{NP}$.

Outra importância do Teorema 2, além de ajudar a resolver a questão “ $\text{P} = \text{NP}$?”, é que podemos classificar os problemas pelo grau de dificuldade, pois, uma vez que mostramos que um problema é NP-completo, também estamos mostrando que esse problema não possui algoritmo polinomial conhecido e que encontrar esse algoritmo polinomial é bem difícil, se

ele existir. Dizemos que um problema em P é “tratável” (ou “fácil”) de resolver enquanto os problemas NP-difíceis são “intratáveis” (ou “difíceis”) de resolver.

Com isso, percebe-se a importância de se mostrar que um problema é NP-completo. Um modo bastante utilizado para isso é descrito no teorema abaixo.

Teorema 3. (GAREY; JOHNSON, 1979) *Se B é NP-completo, $C \in \text{NP}$ e $B \leq_p C$, então C é NP-completo.*

Com o Teorema 3, podemos mostrar que um problema é NP-completo sem utilizar diretamente a definição. A dificuldade passa a ser encontrar um problema NP-completo e uma redução polinomial desse problema para o problema que queremos mostrar ser NP-completo.

O primeiro problema mostrado ser NP-completo foi o Problema SAT, definido a seguir. Dizemos que uma fórmula lógica está na *forma normal conjuntiva* se consiste de conjunções (operador lógico “e”, denotado por \wedge) de cláusulas, onde uma cláusula consiste de disjunções (operador “ou”, denotado por \vee) de literais (variável lógica ou complemento de variável lógica). Por exemplo, a fórmula lógica $\phi = (x_1 \vee x_2 \vee x_3) \wedge (\bar{x}_1 \vee x_2 \vee \bar{x}_3)$, que possui 2 cláusulas com 3 literais cada, está na forma normal conjuntiva, onde x_1 , x_2 e x_3 são variáveis lógicas. Dizemos que uma fórmula lógica na forma normal conjuntiva é *satisfatível* se existe uma atribuição de V ou F às variáveis de modo que o resultado da fórmula com essa atribuição seja V. Temos que ϕ é satisfatível, pois resulta em V quando todas suas variáveis são valoradas com V.

Problema SAT

Instância: uma fórmula ϕ na forma normal conjuntiva.

Pergunta: ϕ é satisfatível?

Partindo do PROBLEMA SAT e utilizando o Teorema 3, pode-se concluir que o PROBLEMA 3SAT (definido a seguir) também é NP-completo.

Problema 3SAT

Instância: Fórmula lógica ϕ na forma normal conjuntiva com 3 literais por cláusula.

Pergunta: ϕ é satisfatível?

A principal ideia para a redução do SAT para o 3SAT é que: para as cláusulas com menos de três literais, podemos duplicar algum dos literais dessa cláusula até obter três literais nela; e para cada cláusula com mais de três literais podemos fazer uma troca semelhante como no exemplo a seguir, $(x_1 \vee x_2 \vee x_3 \vee x_4)$ trocamos por $(x_1 \vee x_2 \vee y) \wedge (\bar{y} \vee x_3 \vee x_4)$, onde y é uma

nova variável para a fórmula do 3SAT. Note que essa redução é polinomial e que a fórmula do problema SAT é satisfatível se e só se a fórmula obtida do 3SAT também é satisfatível.

Para finalizar esta seção, mostraremos que o problema COBERTURA DE VÉRTICES é NP-completo a partir do problema 3SAT.

Teorema 4. (GAREY; JOHNSON, 1979) COBERTURA DE VÉRTICES é NP-completo.

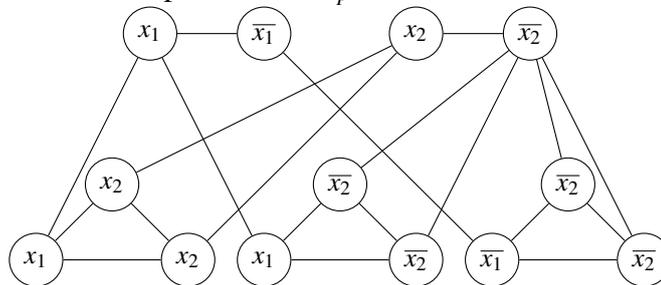
Demonstração. Do Lema 1 temos que COBERTURA DE VÉRTICES está em NP e sabemos que 3SAT é NP-completo. Mostramos agora uma redução polinomial do 3SAT para o COBERTURA DE VÉRTICES considerando que (ϕ) é uma instância do 3SAT e (G, k) a instância do COBERTURA DE VÉRTICES que será gerada a partir de ϕ . Considere que v é a quantidade de variáveis de ϕ e c é a quantidade de cláusulas de ϕ .

Construção:

- para cada variável x de ϕ , criar vértices x e \bar{x} ; e uma aresta $x\bar{x}$ em G (chamamos os vértices x e \bar{x} de vértices variáveis);
- para cada cláusula de $(x \vee y \vee z)$ de ϕ , criar um ciclo de tamanho 3 (x, y, z) em G (chamamos os vértices x, y e z de vértices cláusulas);
- ligar os vértices variáveis aos vértices cláusulas que correspondem ao mesmo literal em ϕ ;
- e
- $k = v + 2c$.

Veja o exemplo de construção do grafo na Figura 16 para $\phi = (x_1 \vee x_2 \vee x_2) \wedge (x_1 \vee \bar{x}_2 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee \bar{x}_2 \vee \bar{x}_2)$.

Figura 16 – Exemplo: $3SAT \leq_p$ COBERTURA DE VÉRTICES



Fonte: elaborado pelo autor.

Nota: $\phi = (x_1 \vee x_2 \vee x_2) \wedge (x_1 \vee \bar{x}_2 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee \bar{x}_2 \vee \bar{x}_2)$. Os vértices de cima foram criados a partir das variáveis x_1 e x_2 , enquanto os vértices de baixo (que induzem um ciclo de tamanho 3) foram criados a partir das cláusulas. As arestas de cima para baixo ligam vértices de literais correspondem.

Veja que G possui $2v + 3c$ vértices e $v + 6c$ arestas e logo sua construção será polinomial. Portanto (G, k) é gerada em tempo polinomial.

Agora mostramos que ϕ é satisfatível se e somente se o grafo G , construído a partir de ϕ , possui uma cobertura de tamanho k (que é igual a $v + 2c$).

Se ϕ é satisfatível, então existe uma valoração V ou F válida para as variáveis de ϕ em que cada cláusula de ϕ é V . Escolha os vértices variáveis de G que correspondem aos literais V da valoração e, para cada cláusula C , escolha dois vértices cláusulas de C de modo que o terceiro vértice cláusula corresponda a um literal V na valoração. Veja que temos $v + 2c = k$ vértices escolhidos. Note que cada aresta de G possui uma extremidade que é um vértice escolhido: para as arestas criadas em (a) temos que exatamente um dos literais x ou \bar{x} foi escolhido; para as arestas criadas nos ciclos de tamanho três em (b) temos que dois vértices de cada ciclo foram escolhidos; e para cada aresta criada em (c) temos que ela tem uma extremidade que é vértice variável u_v e a outra extremidade que é um vértice cláusula u_c e vemos que se u_v não é escolhido é porque corresponde a um literal F para a valoração e, nessa situação, u_c é escolhido, isto é, pelo menos uma das extremidades é um vértice escolhido. Desse modo, os vértices escolhidos formam uma cobertura de vértices de tamanho k para G .

Suponha que temos uma cobertura C , com tamanho $k \leq v + 2c$, para o grafo G (grafo construído a partir de ϕ). Veja que necessariamente temos que qualquer cobertura de vértice C' de G deve cobrir cada aresta criada em (a) com pelo menos um vértice e cada aresta dos ciclos criados (b) com pelo menos dois vértices. Isso implica que, em cada cobertura de vértices de G tem que possuir tamanho pelo menos $v + 2c$. Logo, $v + 2c \leq |C| = k \leq v + 2c$ e, portanto $|C| = v + 2c$. Como temos $v + 2c$ vértices em C necessariamente temos que C possui exatamente um vértice para cada estrutura criada em (a) e exatamente dois vértices para cada estrutura criada em (b). Para os vértices de C que foram criados em (a), dê a valoração ao seu literal correspondente como V e o seu literal oposto como F . Como temos que, para cada cláusula, existe um vértice cláusula que não está na cobertura e a aresta criada em (c) que sai dele está coberta pelo vértice variável (para o qual valoramos seu literal como V), então cada cláusula possui pelo menos um literal V . Portanto, temos que essa valoração satisfaz ϕ .

Utilizando o Teorema 3 concluímos nosso resultado. □

Até agora focamos na complexidade de tempo dos algoritmos referentes ao tempo de execução. Contudo existe também complexidade de espaço que é análogo a complexidade de tempo para células de memória necessária para um algoritmo, em vez de tempo de execução. Similarmente às classes P , NP , NP -difícil e NP -completa, temos as classes $PSPACE$, $NPSPACE$, $PSPACE$ -completa e $PSPACE$ -difícil, respectivamente. Contudo, na complexidade de espaço,

$PSPACE = NPSPACE$. Sabemos também que $P \subseteq NP \subseteq PSPACE$.

O primeiro problema provado ser PSPACE-Completo foi o TQBF “*Totally quantified Boolean formula*” (“Fórmula Booleana Totalmente Quantificada”), também chamado QSAT “*Quantified SAT*” (“SAT quantificado”), em 1973 (STOCKMEYER; MEYER, 1973), definido a seguir. É dada como instância uma fórmula lógica com variáveis de modo que toda variável está quantificada com \exists (existe) ou \forall (para todo) no início da fórmula. O objetivo é decidir se a fórmula é verdadeira ou falsa. Por exemplo, a fórmula $\forall x_1 \exists x_2 (x_1 \vee x_2) \wedge (x_1 \vee \bar{x}_2)$ é falsa, pois tomando x_1 falso, a fórmula será sempre falsa, independente do valor de x_2 . Por outro lado, a fórmula $\exists x_1 \forall x_2 (x_1 \wedge x_2) \vee (x_1 \wedge \bar{x}_2)$ é verdadeira, pois tomando x_1 verdadeiro, a fórmula será sempre verdadeira, independente do valor de x_2 .

O problema TQBF pode ser visto como um jogo com dois jogadores: V e F. O jogador V só pode atribuir valores a variáveis quantificadas com \exists e o jogador F só pode atribuir valores a variáveis quantificadas com \forall . Além disso, a atribuição de valores deve seguir a ordem da quantificação no início da fórmula. O objetivo do jogador V é tornar a fórmula verdadeira, enquanto o objetivo do jogador F é tornar a fórmula falsa.

Após a prova da PSPACE-Completude do problema TQBF, vários outros problemas foram provados PSPACE-Completos, vários deles voltados para jogos com dois jogadores. Um dos mais antigos, por exemplo, é o jogo NODE-KAYLES em que dois jogadores alternam jogadas selecionando vértices de um grafo de modo que os vértices selecionados induzem um conjunto independente no grafo. O último jogador a conseguir jogar vence o jogo (obtendo um conjunto independente maximal). Sabe-se que decidir qual jogador possui uma estratégia vencedora no NODE-KAYLES é um problema PSPACE-Completo (SCHAEFER, 1978), provado em 1978.

2.3 Complexidade Parametrizada

Com a motivação para tratar problemas NP-difíceis e classificá-los quanto ao grau de dificuldade de resolução, Downey e Fellows introduziram a Teoria da Complexidade Parametrizada (recomendamos a leitura o livro (DOWNEY; FELLOWS, 2012)).

Seguimos abaixo as definições do livro (FLUM; GROHE, 2006). Um *parâmetro* k para um problema computacional Q é uma função que atribui um número natural $k(x)$ para cada instância x do problema Q . Quando a instância x do problema estiver clara no contexto, podemos escrever simplesmente k ao invés de $k(x)$. Um *problema parametrizado* é um par (Q, k) , onde

Q é um problema de decisão e k é um parâmetro do problema Q . Vemos a seguir uma versão parametrizada do PROBLEMA 3-COLORAÇÃO parametrizado pelo grau máximo do grafo da instância.

Problema 3-COLORAÇÃO(Δ)

Instância: grafo G .

Parâmetro: $\Delta(G)$ (grau máximo de G).

Pergunta: existe uma coloração própria de vértices de G usando no máximo 3 cores?

Vemos a seguir a definição de um problema parametrizado da cobertura de vértices.

Problema COBERTURA DE VÉRTICES(k)

Instância: grafo G e inteiro k .

Parâmetro: inteiro k .

Pergunta: existe um subconjunto de vértices $S \subseteq V(G)$, de tamanho no máximo k , em que cada aresta de G possui uma extremidade em S ?

Vemos a seguir a definição do problema parametrizado da clique.

Problema CLIQUE(k)

Instância: grafo G e inteiro k .

Parâmetro: inteiro k .

Pergunta: existe um subconjunto de vértices $S \subseteq V(G)$, de tamanho no máximo k , em que todos os vértices de S são adjacentes entre si?

Vemos a seguir a definição do problema parametrizado do conjunto dominante.

Problema DOMINANTE(k)

Instância: grafo G e inteiro k

Parâmetro: inteiro k

Pergunta: existe um subconjunto de vértices $S \subseteq V(G)$, de tamanho no máximo k , em que cada vértice de $G - S$ possui um vizinho em S ?

Também é possível lidar com parametrização em uma quantidade constante de parâmetros, k_1, k_2, \dots, k_c . Nesse caso, considera-se que o parâmetro k do problema parametrizado é a soma $k = k_1 + \dots + k_c$. Um exemplo seria o problema DOMINANTE(Δ, k) que tem a mesma instância e a mesma pergunta de DOMINANTE(k), mas é parametrizado por $\Delta(G)$ e k , ou seja, tem $\Delta(G) + k$ como parâmetro.

Dado um problema computacional Q , um algoritmo XP com relação a um parâmetro k do problema Q é um algoritmo que executa em tempo $O(f(k) \cdot n^{g(k)})$, onde n é o tamanho da

representação da instância do problema Q , k é o parâmetro e f e g são funções computáveis. Definimos a *Classe* XP como o conjunto dos problemas parametrizados que possuem algoritmos XP. Ou seja, são os problemas parametrizados que possuem algoritmos polinomiais no tamanho da instância quando os parâmetros são fixos, isto é, o tamanho de suas representações é assintoticamente constante em relação ao tamanho n da representação da instância.

Dado um problema computacional Q , um *algoritmo* FPT (tratável por parâmetro fixo) com relação a um parâmetro k do problema Q é um algoritmo que executa em tempo $O(f(k) \cdot n^{O(1)})$ (ou simplesmente $O^*(f(k))$), onde k é o parâmetro, n é o tamanho da representação da instância do problema parametrizado e f é uma função computável. Definimos a *Classe* FPT como o conjunto dos problemas parametrizados que possuem algoritmos FPT.

Temos o análogo da redução polinomial da Teoria da Complexidade Clássica. Sejam (Q, k) e (Q', k') dois problemas parametrizados. Uma *redução* FPT (também chamada de *redução parametrizada* ou *transformação paramétrica*) de (Q, k) para (Q', k') (denotada por $(Q, k) \leq_{\text{FPT}} (Q', k')$) é um algoritmo R que, para toda instância x de Q , produz uma instância $x' = R(x)$ de Q' tal que

1. x é SIM em Q se e somente se x' é SIM em Q' ,
2. existe uma função computável g tal que $k'(x') \leq g(k(x))$ para toda instância x de Q ,
3. R é computável por um algoritmo FPT (com relação ao parâmetro k).

Outras definições e resultados semelhantes à Teoria da Complexidade mostramos a seguir.

Lema 2 (Preservação da tratabilidade por parâmetro fixo). *Se $(Q, k) \leq_{\text{FPT}} (Q', k')$ e $(Q', k') \in \text{FPT}$, então $(Q, k) \in \text{FPT}$.*

Duas classes importantes de problemas parametrizados são as *Classes* $W[1]$ e $W[2]$. A definição original dessas classes depende da definição de problemas em circuitos booleanos e, para isso, são necessárias várias notações adicionais, que deixaremos para a subseção seguinte. Fornecemos a princípio definições equivalentes (FLUM; GROHE, 2006) e mais simples dessas classes:

- $W[1]$ é a classe dos problemas parametrizados que possuem uma redução FPT para o problema $\text{CLIQUE}(k)$.
- $W[2]$ é a classe dos problemas parametrizados que possuem uma redução FPT para o problema $\text{DOMINANTE}(k)$.

Assim como $P \subseteq NP$ na Complexidade Clássica, temos que $FPT \subseteq W[1] \subseteq W[2]$ na Complexidade Parametrizada.

Para $t \in \{1, 2\}$, dizemos que um problema parametrizado (Q', k') é $W[t]$ -difícil se para todo problema $(Q, k) \in W[t]$, temos que $(Q, k) \leq_{FPT} (Q', k')$. Além disso, (Q', k') é $W[t]$ -completo se $(Q', k') \in W[t]$ e (Q', k') é $W[t]$ -difícil.

Do mesmo modo que existe a conjectura de que $P \neq NP$ na Teoria da Complexidade clássica, também existe a conjectura $FPT \neq W[1] \neq W[2]$ na Teoria da Complexidade Parametrizada. Sabe-se que se um problema $W[1]$ -completo estiver em FPT , então $FPT = W[1]$ e se um problema $W[2]$ -completo estiver em $W[1]$, então $W[1] = W[2]$. Sabe-se ainda que COBERTURA DE VÉRTICES(k) pertence a FPT e que, pela definição dada acima das Classes $W[1]$ e $W[2]$, o problema CLIQUE(k) é $W[1]$ -completo e o problema DOMINANTE(k) é $W[2]$ -completo.

2.3.1 Hierarquia W

Para fornecer a definição original das Classes $W[1]$ e $W[2]$, mencionadas na seção anterior, é preciso antes definir *circuito booleano*. Um circuito booleano é um grafo direcionado acíclico no qual os vértices são rotulados da seguinte maneira:

- (a) todo vértice com grau de entrada 0 é um *nó de entrada*,
- (b) todo vértice com grau de entrada 1 é um *nó de negação*,
- (c) todo vértice com grau de entrada pelo menos 2 é um *nó de conjunção* ou um *nó de disjunção*.

Os termos negação, conjunção e disjunção estão associados aos operadores lógicos clássicos \neg , \wedge e \vee , respectivamente. Além disso, existe apenas um vértice com grau de saída 0 que também é rotulado como *nó de saída*. Dizemos que um nó é *grande* se tem grau de entrada pelo menos 3 (ou seja, só pode ser de conjunção ou disjunção).

Por simplicidade, como o circuito booleano é acíclico, costuma-se representá-lo sem as orientações das arestas, assumindo o sentido natural dos nós de entrada para o nó de saída, de cima para baixo, como exemplifica a Figura 17.

A *profundidade* (em inglês “*depth*”) de um circuito booleano é o tamanho máximo de um caminho de um nó de entrada para o nó de saída. O *entrelaçamento* (em inglês “*wef*”) de um circuito booleano é o número máximo de nós grandes em um caminho de um nó de entrada para o nó de saída.

Associar valores 0-1 (verdadeiro-falso) aos nós de entrada determina o valor de cada

nó do circuito da maneira esperada, aplicando os operadores lógicos. Com o termo atribuição, estaremos nos referindo a uma atribuição de valores 0-1 aos nós de entrada de um circuito booleano dado. Dizemos que uma atribuição satisfaz o circuito booleano se o valor do nó de saída é 1.

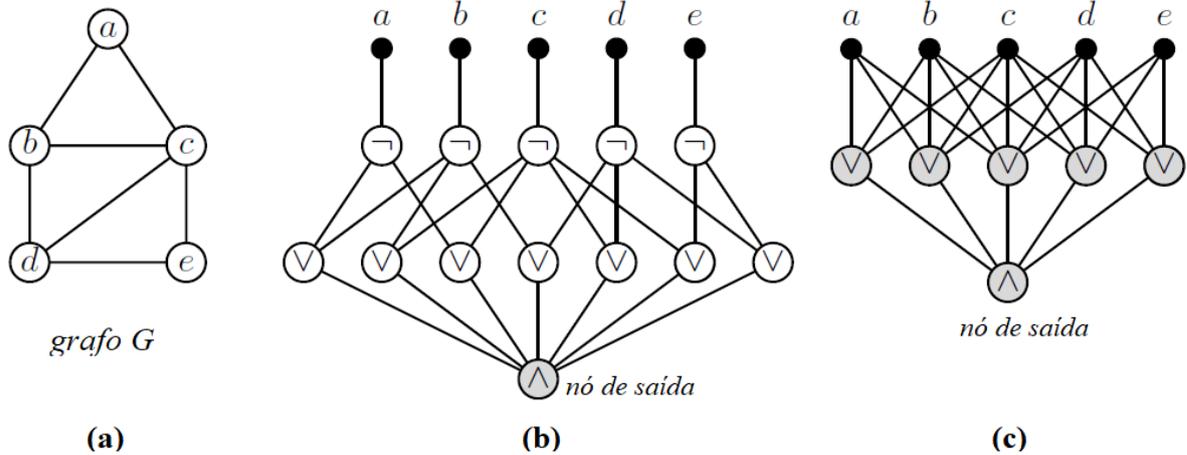
As Figuras 17 (b) e (c) mostram circuitos booleanos associados aos problemas de conjunto independente e conjunto dominante do grafo G da Figura 17 (a), respectivamente. Na Figura 17 (b) cada vértice preto é ligado a apenas um vértices ' \neg ' e ambos são referente a um vértice de G , enquanto cada vértice ' \vee ' é referente a uma aresta de G e é ligado aos dois vértices ' \neg ' referentes a suas extremidades; e o vértice ' \wedge ' é ligado a todos os vértices ' \vee '. Note que, na Figura 17 (b), se pegarmos os vértices a e b como 1 (verdadeiro) logo o primeiro vértice ' \vee ' (da esquerda pra direita) será 0 (falso) e logo o vértice de saída ' \wedge ' será 0. Como isso ocorre para cada par vértices adjacentes de G , então, para o vértice de saída ser 1, todos os vértices de entrada com 1 devem ser correspondentes a um conjunto independente em G . Para obter um circuito booleano para o problema $\text{CLIQUE}(k)$, basta tomar o circuito do conjunto independente, como na Figura 17 (b), relacionado ao complemento do grafo G da Figura 17 (a). Já na Figura 17 (c) cada vértice preto corresponde a um vértice de G , cada vértices ' \vee ' corresponde a vizinhança fechada de um vértice de G e o vértice ' \wedge ' é ligado a todos os vértices ' \vee '. Para termos 0 no vértice de saída no circuito da Figura 17 (c), então pelo menos um vértice ' \vee ' deverá ter todos seus vértices de entrada 0, isto é, deverá existe uma vizinhança fechada de algum vértice de G com todos seus vértices 0. Deste modo, para termos 1 no vértice de saída devemos ter pelo menos um vértice de entrada 1 na vizinhança fechada de cada vértice de G , vemos aqui, que esses vértices de entrada 1 formam um conjunto dominante de G .

Decidir se um circuito booleano é satisfatível, ou seja, se existe uma atribuição que o satisfaz, é um problema NP-Completo, pois o Problema 3SAT é um caso particular: toda instância do 3SAT pode ser representada em um circuito booleano de profundidade de no máximo 3 e entrelaçamento de no máximo 2 (conjunção de disjunções).

Seja WCD (derivado do inglês "*Weighted Circuit Satisfiability*") o problema ponderado de satisfatibilidade de circuitos: dado um circuito booleano C e um inteiro k , decidir se existe uma atribuição que satisfaça o circuito C com exatamente k valores iguais a 1.

Com isso, definimos a classe $W[t]$, onde $t \geq 1$ é um inteiro, como a classe dos problemas parametrizados que possuem uma redução FPT para o problema WCD restrito a circuitos com entrelaçamento t , parametrizado por k (número de valores 1 nos nós de entrada).

Figura 17 – Exemplos de circuitos booleanos



Fonte: Adaptação de (CYGAN *et al.*, 2015).

Nota: exemplos de circuitos booleanos para decidir se um conjunto de vértices é independente ou dominante em um grafo G . Nós grandes (que entram na contagem do entrelaçamento) estão em cinza: (a) Grafo G com 5 vértices e 7 arestas; (b) Circuito booleano de profundidade 3 e entrelaçamento 1, que é satisfeito apenas por **conjuntos independentes** de G (cada nó de disjunção corresponde a uma aresta de G e tem grau de entrada 2); e (c) Circuito de profundidade 2 e entrelaçamento 2 satisfeito apenas por **conjuntos dominantes** de G (cada nó de disjunção corresponde à vizinhança fechada de um vértice de G).

Note que $W[t] \subseteq W[t + 1]$ para todo inteiro $t \geq 1$.

Os circuitos booleanos da Figura 17 mostram que é possível representar o problema do Conjunto Independente (e consequentemente o problema $\text{CLIQUE}(k)$) em circuitos booleanos com entrelaçamento 1 e o problema $\text{DOMINANTE}(k)$ em circuitos com entrelaçamento 2. Ou seja, $\text{CLIQUE}(k)$ e $\text{DOMINANTE}(k)$ estão em $W[1]$ e em $W[2]$, respectivamente.

Sabe-se que $\text{CLIQUE}(k)$ é $W[1]$ -Difícil, ou seja, todo problema em $W[1]$ possui uma redução FPT para ele, e que $\text{DOMINANTE}(k)$ é $W[2]$ -Difícil, ou seja, todo problema em $W[2]$ possui uma redução FPT para ele (FLUM; GROHE, 2006; DOWNEY; FELLOWS, 2013).

3 JOGOS DE COLORAÇÃO EM GRAFOS

Neste capítulo apresentamos nossos resultados acerca de dois jogos de coloração em grafos: Jogo da Coloração e Jogo da Coloração Gulosa. Mostramos que ambos os jogos são PSPACE-completos, mesmo se o número de cores é o número cromático. Além disso, estendemos um resultado de Havet e Zhu (HAVET; ZHU, 2013) provando que o número guloso de jogo é igual ao número cromático para grafos split e várias super classes de cografos. Nossos resultados deste capítulo foram apresentados em 2019 no *Latin & American Algorithms, Graphs and Optimization Symposium* (LAGOS) e publicados em seus anais *Electronic Notes in Theoretical Computer Science* (COSTA *et al.*, 2019), como resumo estendido, e no periódico *Theoretical Computer Science* como artigo completo (COSTA *et al.*, 2020). Esses resultados resolveram um problema em aberto há mais de 28 anos e foram obtidos com o auxílio do pesquisador Ronan Soares (DEMA-UFC) e seu aluno de mestrado na época, Victor Lage. Dada a importância dos resultados obtidos (a resolução de um problema em aberto há quase 30 anos), os orientadores decidiram que os resultados deveriam constar em ambos os trabalhos de pós-graduação, a saber, a dissertação de mestrado de Victor Lage (PESSOA, 2019) e esta tese de doutorado.

3.1 O número cromático de jogo é PSPACE-Difícil

Uma das dificuldades em se provar a PSPACE-completude do Jogo da Coloração está na própria definição do problema de decisão. Como mencionado por Zhu (ZHU, 1999), o Jogo da Coloração “*apresenta algumas propriedades estranhas*” e a seguinte questão, aparentemente “ingênua”, permanece em aberto até hoje (Questão 1 de (ZHU, 1999)):

- Alice possui uma estratégia vencedora para o Jogo da Coloração com $k + 1$ cores se ela já possui uma estratégia vencedora com k cores?

Com isso, podemos definir os dois problemas de decisão para o Jogo da Coloração:

Problema 1 do Jogo da Coloração

Instância: um grafo G e um inteiro positivo k .

Pergunta: $\chi_g(G) \leq k$?

Problema 2 do Jogo da Coloração

Instância: um grafo G e um inteiro positivo k .

Pergunta: Alice possui uma estratégia vencedora com apenas k cores?

Os Problemas 1 e 2 do Jogo da Coloração são equivalentes se e somente se a Questão 1 de (ZHU, 1999) é respondida com SIM. Para verificar isso, note que, se a Questão 1 de (ZHU, 1999) tem resposta NÃO, então os problemas não são equivalentes, pois existiriam grafos tais que Alice possui uma estratégia vencedora com menos de k cores, mas não possui com k cores. Assuma agora que a Questão 1 de (ZHU, 1999) tem resposta SIM. Logo se $\chi(G) \leq k$, então Alice possui uma estratégia com $k' \leq k$ cores e, pela Questão 1, também tem com k cores. Por outro lado, se Alice possui uma estratégia com k cores, então $\chi(G) \leq k$. Ou seja, os problemas seriam equivalentes.

Nesta seção, mostramos que o seguinte problema (mais restrito) do Jogo da Coloração é PSPACE-completo.

Problema 3 do Jogo da Coloração

Instância: um grafo G e o número cromático $\chi(G)$.

Pergunta: $\chi_g(G) = \chi(G)$?

Facilmente percebe-se que os Problemas 1 e 2 são generalizações do Problema 3, uma vez que ambos os Problemas 1 e 2 são equivalentes para $k = \chi(G)$. Para verificar isso, note que $\chi_g(G) \leq k = \chi(G)$ se e somente se $\chi_g(G) = \chi(G)$, que é verdade se e somente se Alice possui uma estratégia vencedora com $k = \chi(G)$ cores. Então a PSPACE-dificuldade do Problema 3 do Jogo da Coloração implica na PSPACE-dificuldade dos Problemas 1 e 2 do Jogo da Coloração.

Antes de provar a PSPACE-completude, vamos provar que estão em PSPACE. Até onde sabemos, nenhum trabalho possui a definição explícita desses problemas de decisão ou a prova de pertinência deles em PSPACE.

Lema 3. *Os Problemas 1, 2 e 3 do Jogo de Coloração estão em PSPACE.*

Demonstração. Primeiro apresentamos o Algoritmo 1 abaixo, que é recursivo. Vamos mostrar que ele decide o Problema 2 em espaço polinomial. Consideramos que o conjunto C de cores é o conjunto $\{1, \dots, k\}$, onde k é dado na instância.

Claramente o Algoritmo 1 decide o Problema 2 verificando todas as possibilidades de jogadas. Para avaliarmos a complexidade de espaço observamos que a profundidade da recursão é, no máximo, o número de jogadas possíveis no grafo G , isto é, a quantidade de vértices n (temos um vértice colorido por jogada). Em cada nível precisamos armazenar apenas as cores dos n vértices e, deste modo, o espaço será $O(m + n)$. Portanto, o Algoritmo 1 tem espaço linear

Algoritmo 1: Resolve o Problema 2 do Jogo de Coloração em espaço polinomial

Entrada: Grafo G , inteiro positivo k , jogada (Alice ou Bob).

Saída: responde SIM, caso exista uma estratégia vencedora para Alice em G com k cores e, em caso contrário, NÃO.

```

1 se  $G$  está totalmente colorido com  $k$  cores então
2   └─ retorne SIM
3 se vértice que não pode ser colorido com uma cor de 1 a  $k$  então
4   └─ retorne NÃO
5 se jogada = Alice então
6   ┌─ para cada vértice não colorido  $v \in V(G)$  faça
7     ┌─ para cada cor  $i = 1, \dots, k$  faça
8       ┌─ se  $i$  não é cor de vértice vizinho de  $v$  então
9         ┌─ colorir  $v$  com cor  $i$ 
10        ┌─ se Algoritmo 1 ( $G, k, Bob$ ) = SIM então
11          └─ retorne SIM
12        └─ descolorir  $v$ 
13      └─ retorne NÃO
14 se jogada = Bob então
15   ┌─ para cada vértice não colorido  $v \in V(G)$  faça
16     ┌─ para cada cor  $i = 1, \dots, k$  faça
17       ┌─ se  $i$  não é cor de vértice vizinho de  $v$  então
18         ┌─ colorir  $v$  com cor  $i$ 
19        ┌─ se Algoritmo 1 ( $G, k, Alice$ ) = NÃO então
20          └─ retorne NÃO
21        └─ descolorir  $v$ 
22      └─ retorne SIM
  
```

de execução. Consequentemente, o Problema 3 também está em PSPACE, visto que $\chi(G)$ é dado na instância.

Por fim, considere o Problema 1. Note que podemos resolvê-lo utilizando o Algoritmo 1 do Problema 2 para todo $k' = \chi(G), \dots, k$. Isto é, se o Algoritmo 1 retorna SIM para algum k' , então o Problema 1 tem resposta SIM também. Caso contrário, será NÃO. Uma vez que o Problema 2 está em PSPACE, então o Problema 1 também está em PSPACE. \square

Para mostrar que o Problema 3 do Jogo da Coloração é PSPACE-completo, obtemos uma redução do PROBLEMA POS-CNF definido a seguir, que é PSPACE-completo (SCHAEFER, 1978).

No PROBLEMA POS-CNF, temos um conjunto $\{X_1, \dots, X_N\}$ de N variáveis e uma fórmula na forma normal conjuntiva com M cláusulas C_1, \dots, C_M que são formadas apenas por

literais positivos (isto é, os literais das cláusulas não são complemento de variáveis lógicas). Dois jogadores (Alice e Bob) alternam suas jogadas atribuindo valor verdadeiro ou falso a uma variável ainda sem valor. Alice ganha se e somente se a fórmula é verdadeira após todas as N variáveis terem recebido um valor verdadeiro ou falso. Claramente, uma vez que existem apenas variáveis positivas, podemos assumir que Alice e Bob apenas atribuem às variáveis escolhidas os valores Verdadeiro (V) e Falso (F), respectivamente. Podemos assumir que nenhuma cláusula contém todas as variáveis, uma vez que esta cláusula é trivialmente satisfeita.

Como exemplo, considere a fórmula $\phi = (X_1 \vee X_2) \wedge (X_1 \vee X_3) \wedge (X_2 \vee X_4) \wedge (X_3 \vee X_4)$.

Note que Bob possui uma estratégia vencedora para ϕ :

1. Se Alice jogar X_1 **V**, Bob joga X_4 **F** e vence, pois Alice não consegue salvar $(X_2 \vee X_4)$ e $(X_3 \vee X_4)$ em uma jogada;
2. Se Alice jogar X_2 **V**, Bob joga X_3 **F** e vence, pois Alice não consegue salvar $(X_1 \vee X_3)$ e $(X_3 \vee X_4)$ em uma jogada;
3. Se Alice jogar X_3 **V**, Bob joga X_2 **F** e vence, pois Alice não consegue salvar $(X_1 \vee X_2)$ e $(X_2 \vee X_4)$ em uma jogada.
4. Se Alice jogar X_4 **V**, Bob joga X_1 **F** e vence, pois Alice não consegue salvar $(X_1 \vee X_2)$ e $(X_1 \vee X_3)$ em uma jogada;

No lema a seguir, provamos uma propriedade importante do PROBLEMA POS-CNF, que será bastante útil na redução para o Problema do Jogo de Coloração.

Lema 4. *Para qualquer instância do PROBLEMA POS-CNF, o jogador que tem uma estratégia vencedora continua a ter uma estratégia vencedora para a mesma instância na variante do jogo POS-CNF em que seu adversário pode passar jogadas.*

Demonstração. Suponha que um dos jogadores, Alice ou Bob, chamado a partir de agora de jogador principal, possui uma estratégia vencedora para alguma instância do POS-CNF. Considere a partir de agora a variante do Jogo POS-CNF em que o oponente do jogador principal pode passar jogadas. Como observado antes, suponha ainda que, ao escolher uma variável, o jogador principal e seu oponente irão atribuir seus valores padrão às variáveis: Alice sempre atribui V e Bob sempre atribui F.

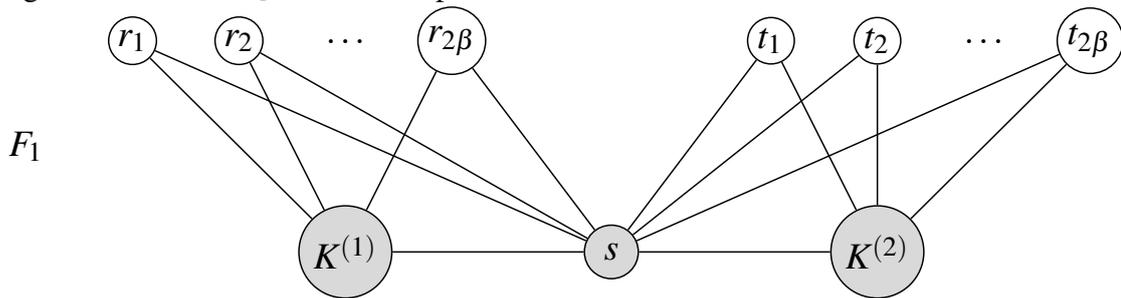
Se o oponente do jogador principal passou uma jogada no meio do jogo, o jogador principal assume que seu oponente escolheu alguma das variáveis ainda não selecionadas e continua a jogar seguindo sua estratégia vencedora do jogo original. Se o oponente escolher essa variável em algum outro momento do jogo, simplesmente assumamos que outra variável

ainda não selecionada foi escolhida e continue a jogar seguindo a estratégia vencedora do jogo original. Caso todas as variáveis tenham sido selecionadas, incluindo as “assumidas” pelo jogador principal, então a fórmula já tem o valor esperado (V se o jogador principal for Alice, e F se o jogador principal for Bob), pois o jogador principal seguiu sua estratégia vencedora do jogo original, e simplesmente continue selecionando variáveis em qualquer ordem, com o simples objetivo de terminar o jogo. \square

Um resultado análogo ao do Lema 4 não é válido para qualquer jogo. Por exemplo, no Jogo da Coloração e no Jogo da Coloração Gulosa cuja instância é o grafo bipartido completo após a remoção de um emparelhamento perfeito, conforme demonstramos na Figura 6 da Seção 1.1.

Mostraremos no Teorema 5 que o Problema 3 do Jogo da Coloração é PSPACE-completo. Antes disso, mostramos uma importante ferramenta para a redução do Teorema 5, que é o grafo F_1^β da Figura 18, onde $\beta \geq 2$ é um número inteiro. O grafo F_1^β possui um vértice universal s , duas cliques $K^{(1)}$ e $K^{(2)}$ com $\beta - 1$ vértices cada e dois conjuntos independentes $I^{(1)} = \{r_1, \dots, r_{2\beta}\}$ e $I^{(2)} = \{t_1, \dots, t_{2\beta}\}$ com 2β vértices cada. Cada vértice de $I^{(i)}$ é adjacente a todos os vértices de $K^{(i)}$ para $i \in \{1, 2\}$. Começamos provando que, com $2\beta - 1$ cores, Alice vence o jogo em F_1^β se e somente se ela começa o jogo e, neste caso, sua primeira jogada deve ser no vértice s . Por simplicidade, denotaremos apenas F_1 , uma vez que o valor β estará claro no contexto.

Figura 18 – Grafo F_1 : ferramenta para demonstrar o Teorema 5



Fonte: elaborado pelo autor.

Nota: grafo F_1 . $K^{(1)}$ e $K^{(2)}$ são cliques de tamanho $\beta - 1$. Os vértices $r_1, \dots, r_{2\beta}$ são adjacentes a qualquer vértice de $K^{(1)}$. Os vértices $t_1, \dots, t_{2\beta}$ são adjacentes a cada vértice de $K^{(2)}$. O vértice s é universal.

Lema 5. *Alice tem uma estratégia vencedora para o grafo F_1 da Figura 18 no Jogo da Coloração com $2\beta - 1$ cores se e somente se ela começa o jogo e, neste caso, ela deve colorir o vértice s em sua primeira jogada. Além disso, existe uma estratégia vencedora para Alice mesmo se Bob puder passar jogadas.*

Demonstração. Lembramos que $A(v, c)$ denota a jogada em que Alice escolhe o vértice v e o colore com a cor c e, de modo análogo, $B(v, c)$ denota a jogada que Bob escolhe o vértice v e o colore com a cor c . Do grafo F_1 , faremos $I^{(1)} = \{r_1, \dots, r_{2\beta}\}$, $I^{(2)} = \{t_1, \dots, t_{2\beta}\}$, $K^{(1)} = \{u_1, u_2, v_3, \dots, u_{\beta-1}\}$ e $K^{(2)} = \{v_1, v_2, v_3, \dots, v_{\beta-1}\}$. Considere $C = \{1, 2, \dots, 2\beta - 1\}$ o conjunto das cores que serão utilizadas no jogo.

Intuitivamente vemos que se tivermos inicialmente pelo menos β cores distintas nos vértices do conjunto independente $I^{(1)}$, então F_1 será colorido com pelo menos 2β cores, pois teremos as β cores de $I^{(1)}$ somadas às β cores distintas necessárias para colorir a clique $K^{(1)} \cup \{s\}$. Analogamente para $I^{(2)}$. Desse modo, uma estratégia para Bob é colorir β vértices de $I^{(1)}$ ou $I^{(2)}$ com cores distintas, vencendo o jogo, já que nessa situação são necessárias mais que do que $2\beta - 1$ cores para colorir F_1 .

Vemos também que, se as cliques $K^{(1)} \cup \{s\}$ e $K^{(2)} \cup \{s\}$ forem coloridas antes dos conjuntos independentes $I^{(1)}$ e $I^{(2)}$, usando no máximo $2\beta - 1$ cores, então F_1 poderá ser colorido com $2\beta - 1$ cores, pois os vértices de $I^{(1)}$ e $I^{(2)}$ têm grau $\beta < 2\beta - 1$. Desse modo, uma estratégia para Alice é colorir os vértices das cliques $K^{(1)} \cup \{s\}$ e $K^{(2)} \cup \{s\}$ primeiro e deixar os vértices de $I^{(1)}$ e $I^{(2)}$ para depois.

Contudo, ainda não sabemos quem vencerá o jogo, pois tudo depende do que será colorido primeiro, um conjunto independente $I^{(1)}$ (ou $I^{(2)}$) com β cores (Bob vencendo) ou as cliques $K^{(1)} \cup \{s\}$ e $K^{(2)} \cup \{s\}$ coloridas primeiro (Alice vencendo). Com isso, Alice possui a estratégia de colorir o mais rápido possível os vértices das cliques de F_1 e Bob os vértices dos conjuntos independentes, pois, em caso contrário um estará ajudando com a estratégia do seu adversário. Assim, consideramos que, nas jogadas iniciais, Alice colore vértices das cliques e Bob colore vértices dos conjuntos independentes.

Caso Bob inicie o jogo, facilmente vemos que ele vence colorindo β vértices de $I^{(1)}$ com cores distintas, pois $I^{(1)}$ tem 2β vértices. Segue um exemplo até a β -ésima jogada de Bob, considerando que Alice joga apenas na clique $K^{(1)} \cup \{s\}$:

Rodada 1: $B(r_1, 1) - A(s, 2)$

Rodada 2: $B(r_2, 3) - A(u_1, 4)$

Rodada 3: $B(r_3, 5) - A(u_2, 6)$

\vdots $\quad \quad \quad \vdots$

Rodada i : $B(r_i, 2i - 1) - A(u_{i-1}, 2i)$

\vdots $\quad \quad \quad \vdots$

Rodada β : $B(r_\beta, 2\beta - 1) - \text{Alice não consegue colorir o vértice } u_{\beta-1}$

Segue outro exemplo até a β -ésima jogada de Bob, considerando que Alice joga apenas no conjunto independente $I^{(1)}$:

Rodada 1: $B(r_1, 1) - A(r_2, 1)$

Rodada 2: $B(r_3, 2) - A(r_4, 2)$

Rodada 3: $B(r_5, 3) - A(r_6, 3)$

\vdots $\quad \quad \quad \vdots$

Rodada i : $B(r_{2i-1}, i) - A(r_{2i}, i)$

\vdots $\quad \quad \quad \vdots$

Rodada β : $B(r_{2\beta-1}, \beta) - \text{Bob consegue } \beta \text{ cores distintas em } I^{(1)}$

Nesses exemplos Bob consegue colorir β vértices de $I^{(1)}$ com cores distintas de C e vence o jogo, como já mencionado.

Agora assuma que Alice inicia o jogo. Caso a primeira jogada de Alice seja colorir um vértice de $K^{(1)}$, então Bob segue a estratégia de colorir os β vértices de $I^{(2)}$ com cores distintas (como mencionado acima). Analogamente temos o mesmo para o início em algum vértice de $K^{(2)}$. Podemos observar que, se Alice não iniciar o jogo no vértice s , então perderá.

Assuma então que Alice inicia o jogo e colore o vértice s primeiro. A partir daí, se Bob colorir um vértice de $I^{(1)} \cup K^{(1)}$, Alice colore um vértice de $K^{(1)}$ (se houver algum não-colorido; caso contrário, ela colore um vértice de $K^{(2)}$). Temos o análogo se Bob colorir um vértice de $I^{(2)} \cup K^{(2)}$.

O exemplo a seguir ilustra essa estratégia:

Rodada 1: $A(s, 1) - B(r_1, 2)$

Rodada 2: $A(u_1, 3) - B(r_2, 4)$

Rodada 3: $A(u_2, 5) - B(r_3, 6)$

\vdots $\quad \quad \quad \vdots$

Rodada i : $A(u_{i-1}, 2i - 1) - B(r_i, 2i)$

\vdots $\quad \quad \quad \vdots$

Rodada $(\beta - 1)$: $A(u_{\beta-2}, 2\beta - 3) - B(r_{\beta-1}, 2\beta - 2)$

Rodada β : $A(u_{\beta-1}, 2\beta - 1) - B(r_\beta, 2)$

Além disso, facilmente vemos que, se Bob passar jogadas, Alice pode manter sua estratégia, colorindo as cliques primeiro e assim vencendo o jogo.

□

Teorema 5. Dado um grafo G e o número cromático $\chi(G)$, decidir se $\chi_g(G) = \chi(G)$ é PSPACE-completo. Consequentemente, se k é um inteiro, decidir se $\chi_g(G) \leq k$ ou decidir se Alice possui uma estratégia vencedora com k cores são problemas PSPACE-completos.

Demonstração. Pelo Lema 3 temos que os três problemas de decisão estão em PSPACE. Agora mostraremos a construção do grafo seguida de um exemplo na Figura 19. Dada uma fórmula ϕ de POS-CNF com N variáveis X_1, \dots, X_N e M cláusulas C_1, \dots, C_M , sejam:

- p_j o tamanho da cláusula C_j (para $j = 1, \dots, M$),
- $p = \max_{j=1, \dots, M} \{p_j\}$ e
- $\beta = 2N + M$.

Veja que cada cláusula possui no máximo p variáveis.

A seguir construímos o grafo G tal que $\chi(G) = 2\beta - 1$ e, além disso, $\chi_g(G) = 2\beta - 1$ se e somente se Alice tem uma estratégia vencedora para ϕ .

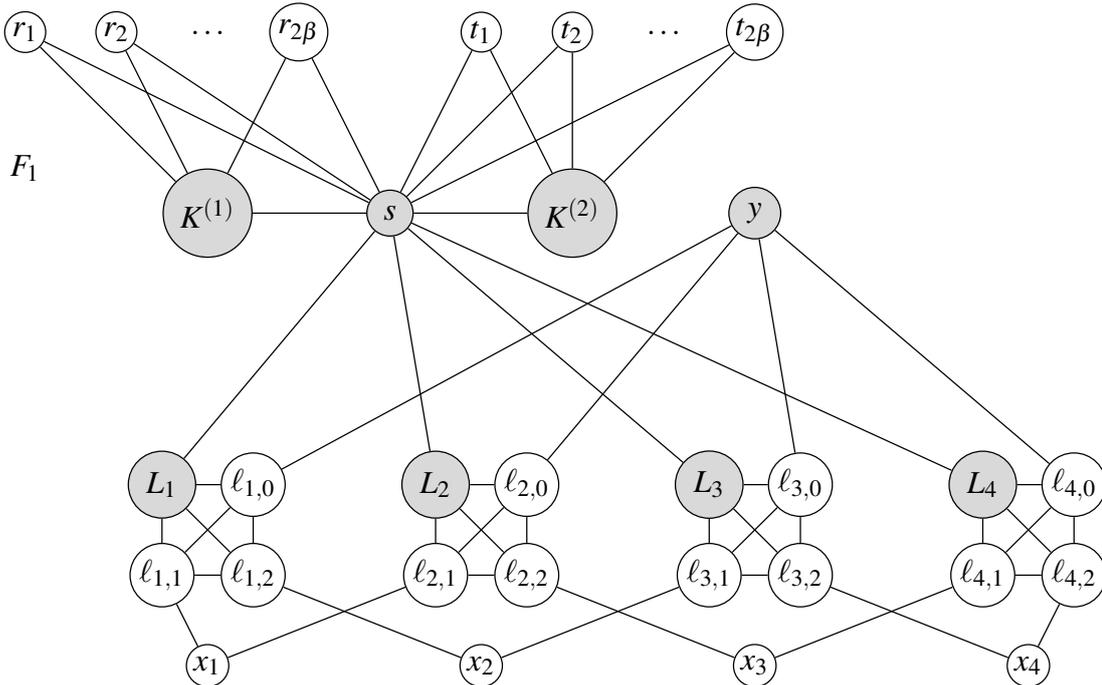
Construção de G , que é vazio inicialmente:

- Adicione F_1^β considerando $\beta = 2N + M$ (grafo da Figura 18);
- Adicione um vértice y ;
- Para cada variável X_i , adicione um vértice x_i (chamamos x_i de *vértice variável*);
- Para cada cláusula C_j adicione uma clique (chamamos de *clique cláusula*) com os seguintes vértices:
 - $\ell_{j,1}, \dots, \ell_{j,p_j}$, referentes às variáveis de C_j ,
 - $\ell_{j,0}$ (que não é referente a nenhuma variável),
 - Adicione uma clique L_j com tamanho $2(\beta - p_j) - 3 \geq 2N + M$,
- Conecte cada $\ell_{j,k}$ a x_i com uma aresta se e somente se ambos são referentes a mesma variável, para $j = 1, \dots, M$, $i = 1, \dots, N$ e $k = 1, \dots, p_j$;
- Conecte $\ell_{j,0}$ ao vértice y para cada $j = 1, \dots, M$;
- Para cada vértice $\ell_{j,k}$ ($j = 1, \dots, M$ e $k = 0, \dots, p_j$), substitua-o por dois vértices gêmeos $\ell'_{j,k}$ e $\ell''_{j,k}$ (ou seja, são adjacentes entre si e aos vizinhos de $\ell_{j,k}$).
- Conecte todos os vértices de cada L_j a s (vértice de F_1) para $j = 1, \dots, M$.

Vemos que cada clique cláusula possui exatamente $2\beta - 1$ vértices.

A Figura 19 mostra a construção do grafo G para $\phi = (X_1 \vee X_2) \wedge (X_1 \vee X_3) \wedge (X_2 \vee X_4) \wedge (X_3 \vee X_4)$. Lembramos que Bob tem uma estratégia vencedora para essa fórmula ϕ . Nesse exemplo, temos $N = 4$ variáveis, $M = 4$ cláusulas, $p = 2$, $\beta = 2N + M = 12$ e as cliques L_1, \dots, L_M possuem $2(\beta - p) - 3 = 17$ vértices cada.

Figura 19 – Exemplo de redução para o Jogo da Coloração



Fonte: elaborado pelo autor.

Nota: construção de G a partir da fórmula $\phi = (X_1 \vee X_2) \wedge (X_1 \vee X_3) \wedge (X_2 \vee X_4) \wedge (X_3 \vee X_4)$. Lembramos que cada vértice $\ell_{j,k}$ representa dois vértices gêmeos $\ell'_{j,k}$ e $\ell''_{j,k}$; L_1, L_2, L_3 e L_4 são cliques com 17 vértices. Bob possui uma estratégia vencedora com 2β cores no Jogo da Coloração.

Veja que temos uma coloração própria nos vértices de G com $2\beta - 1$. Basta colorir:

- s com a cor 1,
- y e x_i com a cor $2\beta - 1$ (para $i = 1, \dots, n$),
- Os vértices das cliques $K^{(1)}$ e $K^{(2)}$ de F_1 com as cores de 2 até β ,
- Os vértices dos conjuntos independentes de F_1 com a cor $\beta + 1$,
- $\ell'_{j,k}$ e $\ell''_{j,k}$ (para $j = 1, \dots, M$) com as cores $2k + 1$ e $2k + 2$ (para $k = 0, \dots, p_j$),
- E os vértices de L_j com as cores $2p_j + 3, \dots, 2\beta - 1$.

Uma vez que cada clique cláusula possui exatamente $2\beta - 1$ vértices, então $\chi(G) = 2\beta - 1$.

Do Lema 5, na primeira jogada, Alice deve colorir o vértice s de F_1 , caso contrário F_1 não pode ser colorido com $2\beta - 1$ cores. Vamos assumir então que Alice colore o vértice s em sua primeira jogada com a cor 1. Além disso, note que, com exceção do vértice s e dos vértices de cada clique cláusula, todo vértice tem grau menor que $2\beta - 1 = 6N + 2M - 1$. Ou seja, todo vértice poderá ser colorido no decorrer do jogo usando $2\beta - 1$ cores, com exceção dos vértices de cada clique cláusula (lembrando que s será colorido na primeira jogada). Assim, o jogo se resumirá a conseguir colorir ou não os vértices de cada clique cláusula.

Mostramos agora que, se Bob possui uma estratégia vencedora no jogo POS-CNF,

então ele tem uma estratégia vencedora no jogo de coloração (ou seja, $\chi_g(G) > 2\beta - 1$). Assuma que Bob ganha no jogo POS-CNF e Alice começa o jogo colorindo s com a cor 1. Bob pode usar a seguinte estratégia. Na sua primeira jogada, Bob colore y com a cor 1 e em suas próximas jogadas faz o seguinte: se Alice colorir um vértice em $N[x_i]$ para algum i , Bob considera que ela valorou X_i como Verdadeiro no jogo POS-CNF e colore com cor 1 o vértice x_j representando o literal X_j escolhido por ele em sua estratégia vencedora do jogo POS-CNF; se Alice não colorir um vértice em $N[x_i]$ para algum i , então Bob joga como se Alice passasse sua jogada no jogo POS-CNF. Uma vez que Bob possui uma estratégia vencedora no jogo POS-CNF, em algum momento todos os literais de uma cláusula estarão valorados com Falso. Isso significa que todos os vizinhos externos de uma clique cláusula estarão coloridos com a cor 1. Uma vez que uma clique cláusula tem $2\beta - 1$ vértices que não podem ser coloridos com a cor 1, vemos que $\chi_g(G) > 2\beta - 1$.

Mostramos agora que, se Alice tem uma estratégia vencedora no jogo POS-CNF, então ela tem uma estratégia vencedora no jogo de coloração (ou seja, $\chi_g(G) = 2\beta - 1$). Veja que cada vértice da clique cláusula tem grau exatamente $2\beta - 1$ (uma vez que possui apenas um vizinho fora da clique cláusula). Para termos cada clique cláusula colorida com as cores de $\{1, \dots, 2\beta - 1\}$, Alice deve garantir que todas as cores que aparecem na vizinhança externa da clique cláusula também apareçam dentro da clique cláusula. Por outro lado, uma estratégia de Bob é fazer com que todos os vizinhos externos de uma clique cláusula sejam coloridos com a mesma cor 1 de s (que representará Falso em POS-CNF) para impedir que Alice use a cor 1 de s dentro da clique cláusula.

Como mencionado, assuma que Alice vence o jogo POS-CNF e que sua primeira jogada é colorir o vértice s de F_1 com a cor 1. Se Bob joga em um vértice de F_1 , então Alice segue a estratégia vencedora em F_1 , pelo Lema 5. Assuma, portanto, apenas jogadas fora de F_1 .

Primeiro mostraremos que o vértice y será colorido até a terceira jogada. Suponha que Bob não colore y com a cor 1 na sua primeira jogada. Então, na próxima jogada, Alice pode garantir que y recebe uma cor distinta de 1. Depois disso, se Bob colorir $\ell'_{i,0}$ ou $\ell''_{i,0}$ para algum i e a clique cláusula correspondente não possui um vértice colorido com a cor 1, então Alice deve colorir um vértice da clique cláusula com a cor 1. Caso contrário, se não existe vértice variável colorido, Alice colore esse vértice variável com uma cor distinta de 1.

Uma vez que cada clique L_j tem pelo menos $2N + M$ vértices, então Alice consegue colorir todos os vértices de variáveis antes que Bob possa colorir todos os vértices de algum

L_j . Com isso, Alice garante que todas as cores dos vértices variáveis aparecem em cada clique cláusula e, deste modo, Alice vence.

Portanto, assumamos que Bob colore o vértice y com cor 1. Com isso, Alice pode usar a seguinte estratégia:

- (1) Se Bob colorir um vértice de F_1 , então Alice usa a estratégia vencedora em F_1 do Lema 5;
- (2) Se Bob colorir algum vértice gêmeo obtido de um vértice $\ell_{i,j}$, então Alice colore o outro vértice gêmeo (obtido do vértice $\ell_{i,j}$) com a cor de menor valor possível.
- (3) Se Bob colorir algum vértice de L_j , então Alice continua o jogo como se Bob tivesse passado sua jogada no jogo POS-CNF, significando que ela escolhe um vértice x_j e colore-o com uma cor diferente de 1, onde X_j é o literal escolhido por ela na estratégia vencedora no jogo POS-CNF;
- (4) Se Bob colorir um vértice x_i , então Alice joga como se Bob tivesse valorado X_i como Falso no jogo POS-CNF;
- (5) Caso não ocorra nenhuma das opções anteriores, então Alice escolhe um vértice qualquer de G e colore com a menor cor possível.

Seguindo esta estratégia, cada clique cláusula tem um vértice colorido com 1. Com isso, uma vez que cada clique L_j tem pelo menos $3N + M$ vértices, Alice e Bob podem terminar colorindo cada clique cláusula usando as cores $1, \dots, 2\beta - 1$. \square

3.2 O número guloso de jogo é PSPACE-Difícil

Resultados similares aos do problema de coloração podem ser obtidos para o número guloso de jogo. Ao contrário do problema do Jogo da Coloração, o problema do jogo de coloração gulosa satisfaz a seguinte proposição:

Proposição 1. *Se Alice possuir uma estratégia vencedora com k cores no problema da coloração gulosa, então Alice também possui uma estratégia com $k + 1$ cores.*

Essa proposição é direta, uma vez que uma estratégia vencedora com k cores no jogo da coloração gulosa pode ser vista como uma estratégia vencedora com $k + 1$ cores na qual não se usa a cor $k + 1$, visto que a coloração seja gulosa.

Portanto podemos definir o problema de decisão para o jogo da coloração gulosa do seguinte modo:

Problema 1 do Jogo da Coloração Gulosa

Instância: um grafo G e um inteiro positivo k

Pergunta: $\Gamma_g(G) \leq k$? Isto é, Alice tem uma estratégia vencedora usando k cores no jogo da coloração gulosa?

Nesta seção, mostramos que o Problema 2 a seguir do Jogo da Coloração Gulosa, que é mais restrito, é PSPACE-completo:

Problema 2 do Jogo da Coloração Gulosa

Instância: um grafo G e o número cromático $\chi(G)$

Pergunta: $\Gamma_g(G) = \chi(G)$? Isto é, Alice tem uma estratégia vencedora usando $\chi(G)$ cores no Jogo da Coloração Gulosa?

Facilmente vemos que o Problema 1 do Jogo da Coloração Gulosa é uma generalização do Problema 2 do Jogo da Coloração Gulosa (basta fazer $k = \chi(G)$). Então a PSPACE-dificuldade do Problema 2 do Jogo da Coloração Gulosa implica na PSPACE-dificuldade do Problema 1 do Jogo da Coloração Gulosa. No que se segue, provamos que ambos os problemas estão em PSPACE.

Lema 6. *Os Problemas 1 e 2 do Jogo da Coloração Gulosa estão em PSPACE.*

Demonstração. Seguiremos com a mesma ideia de demonstração do Lema 3. Primeiro apresentamos o Algoritmo 2 e, em seguida, mostramos que ele decide o Problema 1 do Jogo da Coloração Gulosa em espaço polinomial. Como antes, consideramos que o conjunto C de cores é o conjunto $\{1, \dots, k\}$, onde k é dado na instância.

Claramente o Algoritmo 2 decide o Problema 1 do Jogo da Coloração Gulosa verificando todas as possibilidades de jogadas. Para avaliarmos a complexidade de espaço, observamos que a profundidade da recursão é, no máximo, o número de jogadas possíveis no grafo G , isto é, a quantidade de vértice n (temos um vértice colorido por jogada). Em cada nível precisamos armazenar apenas as cores dos n vértices e, deste modo, o espaço será $O(m + n)$. Portanto, o Algoritmo 2 tem espaço linear de execução e o Problema 1 do Jogo da Coloração Gulosa está em PSPACE.

Por fim, considere o Problema 2 do Jogo da Coloração Gulosa. Note que podemos resolvê-lo em espaço linear utilizando o Algoritmo 2 fazendo $k = \chi(G)$. Portanto, o Problema 2 do Jogo da Coloração Gulosa também está em PSPACE. \square

Obtemos uma redução do problema POS-CNF (muito similar à redução do Teorema 5 para o problema do Jogo da Coloração). Uma ferramenta importante para a redução é o grafo

Algoritmo 2: Resolve o Problema 1 do Jogo de Coloração Gulosa em espaço polinomial

Entrada: Grafo G , inteiro positivo k , jogada (Alice ou Bob).

Saída: responde SIM, caso exista uma estratégia vencedora para Alice em G com k cores no Jogo da Coloração Gulosa e, em caso contrário, NÃO.

```

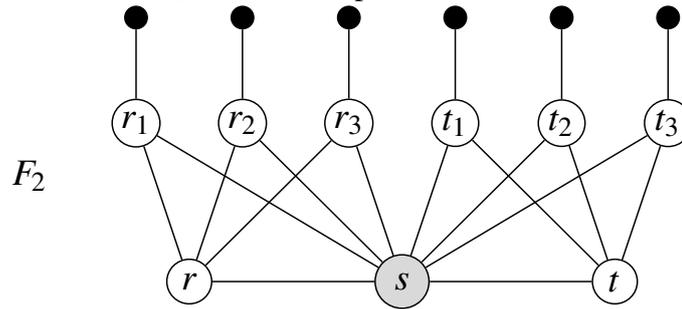
1 se  $G$  está totalmente colorido com  $k$  cores então
2   └─ retorne SIM
3 se existe vértice que não pode ser colorido com uma cor de 1 a  $k$  então
4   └─ retorne NÃO
5 se jogada = Alice então
6   ┌─ para cada vértice não colorido  $v \in V(G)$  faça
7     ┌─  $i \leftarrow 1$ 
8     ┌─ enquanto  $i$  é cor de vértice vizinho de  $v$  faça
9       └─  $i \leftarrow i + 1$ 
10    ┌─ colorir  $v$  com cor  $i$ 
11    ┌─ se Algoritmo 1 ( $G, k, Bob$ ) = SIM então
12      └─ retorne SIM
13    └─ descolorir  $v$ 
14  └─ retorne NÃO
15 se jogada = Bob então
16  ┌─ para cada vértice não colorido  $v \in V(G)$  faça
17    ┌─  $i \leftarrow 1$ 
18    ┌─ enquanto  $i$  é cor de vértice vizinho de  $v$  faça
19      └─  $i \leftarrow i + 1$ 
20    ┌─ colorir  $v$  com cor  $i$ 
21    ┌─ se Algoritmo 1 ( $G, k, Alice$ ) = NÃO então
22      └─ retorne NÃO
23    └─ descolorir  $v$ 
24  └─ retorne SIM

```

F_2 da Figura 20. Começamos mostrando que, com 3 cores, Alice ganha o jogo em F_2 se e somente se ela iniciar com a primeira jogada e, neste caso, sua primeira jogada é colorir o vértice s .

Lema 7. *Alice possui uma estratégia vencedora para o grafo F_2 da Figura 20 no Jogo da Coloração Gulosa com 3 cores se e somente se ela começar o jogo. Neste caso, ela deve colorir o vértice s na sua primeira jogada.*

Demonstração. Suponha que Alice joga primeiro colorindo r , r_1 , r_2 ou r_3 (cor 1 considerando o Jogo de Coloração Gulosa). Então, Bob pode colorir o vértice t_2 (cor 1). Com isso, s e t não podem ser coloridos com a cor 1 e, deste modo, Bob pode forçar t_1 ou t_3 a serem coloridos com

Figura 20 – Grafo F_2 : ferramenta para demonstrar o Teorema 6

Fonte: elaborado pelo autor.

Nota: com 3 cores, quem jogar primeiro ganha no Jogo da Coloração Gulosa.

a cor 4, colorindo um dos dois vértices pretos com a cor 1 na sua próxima jogada. Temos o análogo quando jogar primeiro colorindo t, t_1, t_2 ou t_3 .

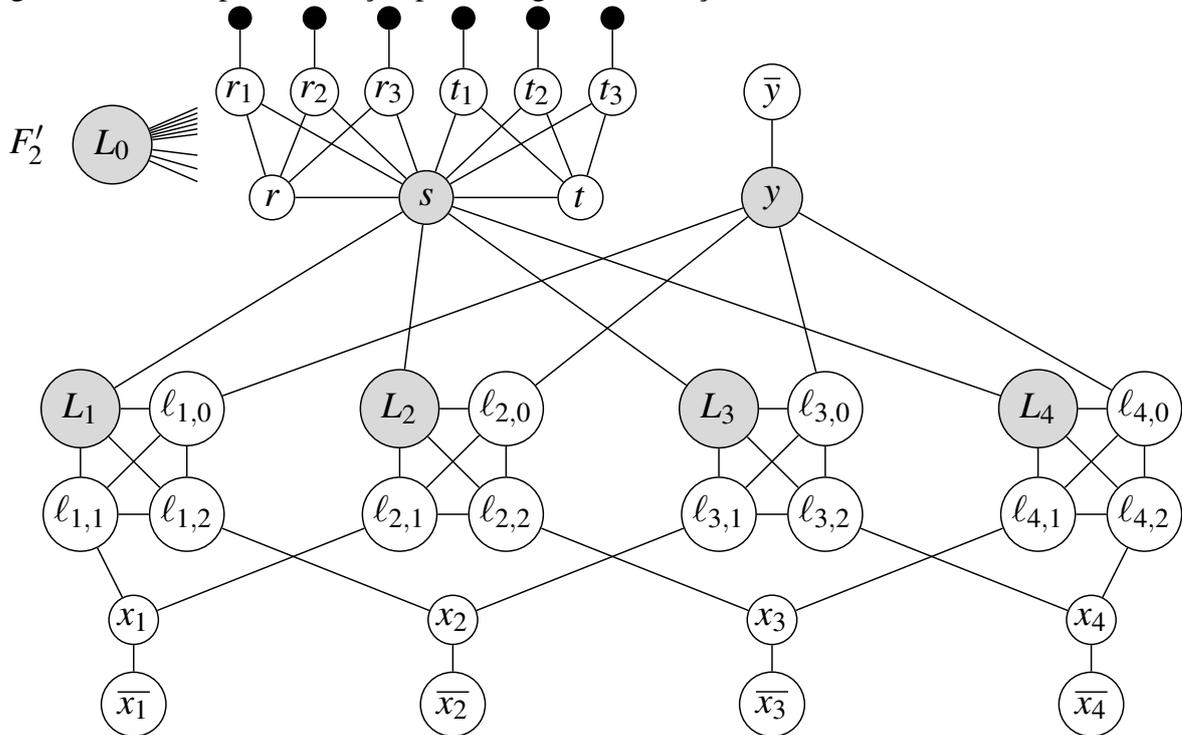
Agora suponha que Alice é a primeira a jogar e colore o vértice s em sua primeira jogada (cor 1). Deste modo, nenhum vértice em $\{r_1, r_2, r_3, t_1, t_2, t_3\}$ pode ser colorido com cor 1 e, assim, todos os vértices pretos serão coloridos com a cor 1 no Jogo da Coloração Gulosa. Sendo assim, todos os vértices em $\{r_1, r_2, r_3, t_1, t_2, t_3\}$ podem ser coloridos com cor 2 ou 3 no jogo. Além disso, Alice pode colorir r e t em suas segunda e terceira jogadas usando as cores $\{2, 3\}$ e ganhar, assim, o jogo. Finalmente suponha que Bob seja o primeiro a jogar. Ele pode ganhar o jogo colorindo o vértice r_2 na primeira jogada (cor 1). Na sua próxima jogada, ele pode colorir o vizinho preto de r_1 ou r_3 (cor 1). Com isso, um vértice em r_1, r_3, r, s será colorido com cor 4, e Bob vence o jogo. Vemos que com um argumento similar temos as mesmas conclusões quando Alice começar primeiro colorindo um vértice preto. \square

Teorema 6. *Dado um grafo G , decidir se $\Gamma_g(G) = \chi(G)$ é PSPACE-completo. Consequentemente, se k é um inteiro, decidir se $\Gamma_g(G) \leq k$ é um problema PSPACE-completo.*

Demonstração. Do Lema 6, ambos os problemas estão em PSPACE. Seguiremos uma ideia similar a do Teorema 5, incluindo um vizinho \bar{x}_i de grau 1 para cada x_i , um vizinho \bar{y} de grau 1 para y e trocando o grafo F_1 pelo grafo F_2' que é o grafo F_2 aplicando a operação de junção com uma nova clique L_0 com $2\beta - 4$ vértices. Lembramos que, dada uma fórmula ϕ com N variáveis X_1, \dots, X_N e M cláusulas C_1, \dots, C_M , onde p_j (para $j = 1, \dots, M$) é o tamanho de C_j , $p = \max_{j=1, \dots, M} \{p_j\}$ e $\beta = 2N + M$. Lembre-se também que cada clique cláusula possui uma subclique L_j com $2(\beta - p_j) - 3$ vértices, para $1 \leq j \leq M$. Neste caso, pelo Lema 7, Alice vence em F_2 , iniciando em s , com 3 cores e como temos $2\beta - 4$ vértices em L_0 contaremos um total de $2\beta - 1$ cores. Em caso contrário, Bob vencerá o jogo.

A Figura 21 mostra a construção do grafo G para a mesma fórmula $\phi = (X_1 \vee X_2) \wedge (X_1 \vee X_3) \wedge (X_2 \vee X_4) \wedge (X_3 \vee X_4)$ do exemplo do Teorema 5. Neste exemplo de redução, temos $N = 4$ variáveis, $M = 4$ cláusulas, $p = 2$, $\beta = 2N + M = 12$, a clique L_0 tem $2\beta - 4 = 20$ vértices e as cliques de L_1 a L_4 possuem $2(\beta - p) - 3 = 17$ vértices cada.

Figura 21 – Exemplo de redução para o Jogo da Coloração Gulosa



Fonte: elaborado pelo autor.

Nota: construção do grafo G a partir da fórmula $\phi = (X_1 \vee X_2) \wedge (X_1 \vee X_3) \wedge (X_2 \vee X_4) \wedge (X_3 \vee X_4)$. Lembramos que cada vértice $\ell_{j,k}$ representa dois vértices gêmeos $\ell'_{j,k}$ e $\ell''_{j,k}$, L_0 é uma clique com 20 vértices e L_1, L_2, L_3 e L_4 são cliques com 17 vértices. Bob tem uma estratégia vencedora para 23 cores no Jogo da Coloração Gulosa.

Como no Teorema 5, $\chi(G) = 2\beta - 1$. Usando um argumento semelhante como na prova do Teorema 5, obtemos o resultado, onde a principal diferença é que, em vez de colorir um vértice x_i (respectivamente y) com uma cor diferente de 1, Alice deve colorir o vértice \bar{x}_i (respectivamente \bar{y}) com a cor 1. \square

3.3 O número guloso de jogo em grafos split

Dizemos que um grafo G é um *grafo split* se $V(G)$ possui uma partição (C, S) ($C \cup S = V(G)$ e $C \cap S = \emptyset$) tal que C induz uma clique e S induz um conjunto independente. Se um vértice v de S é adjacente a todos os vértices em C , então podemos adicionar v a C e removê-lo de S , obtendo outra partição de $V(G)$. Com isso, podemos assumir que cada vértice de S tem um não vizinho em C . Portanto, podemos colorir todos os vértices de C usando as cores

$\{1, \dots, |C|\}$ e colorir cada vértice $s_i \in S$ com uma cor de um vértice $c_i \in C$ não vizinho de s_i . Então $\chi(G) = \omega(G) = |C|$.

Nesta seção, provamos que $\Gamma_g(G) = \chi(G)$ para grafos split, apresentando uma estratégia vencedora para Alice mesmo se Bob começar o jogo e passar alguma de suas jogadas. Para isso, introduzimos o seguinte parâmetro.

Definição 1. *Seja $\Gamma'_g(G)$ o número mínimo de cores tal que Alice tem uma estratégia vencedora no Jogo da Coloração Gulosa em G mesmo se Bob começar o jogo e também puder passar jogadas, inclusive a primeira.*

Considerando a variante em que Bob inicia o jogo e pode passar jogada, temos que ele pode transpor o jogo para a variante normal (passando apenas a primeira jogada e deixando a Alice iniciar o jogo) e logo teremos que $\Gamma_g(G) \leq \Gamma'_g(G)$. Deste modo $\chi(G) \leq \Gamma_g(G) \leq \Gamma'_g(G) \leq \Gamma(G) \leq \Delta(G) + 1$ para um grafo G qualquer. Também note que $\Gamma'_g(G)$ pode ser muito maior que $\Gamma_g(G)$ para grafos em geral. Em (HAVET; ZHU, 2013), mostra-se um exemplo de um grafo G_n obtido de $K_{n,n}$ (grafo bipartido completo) removendo um emparelhamento perfeito e incluindo um vértice isolado (considere o grafo da Figura 6 com um vértice isolado). Se Alice começar o jogo, ela vence usando 2 cores: $\Gamma_g(G_n) = 2$. Caso Bob inicie o jogo mesmo com opção de passar jogadas, ele pode forçar n cores: $\Gamma'_g(G_n) = n$.

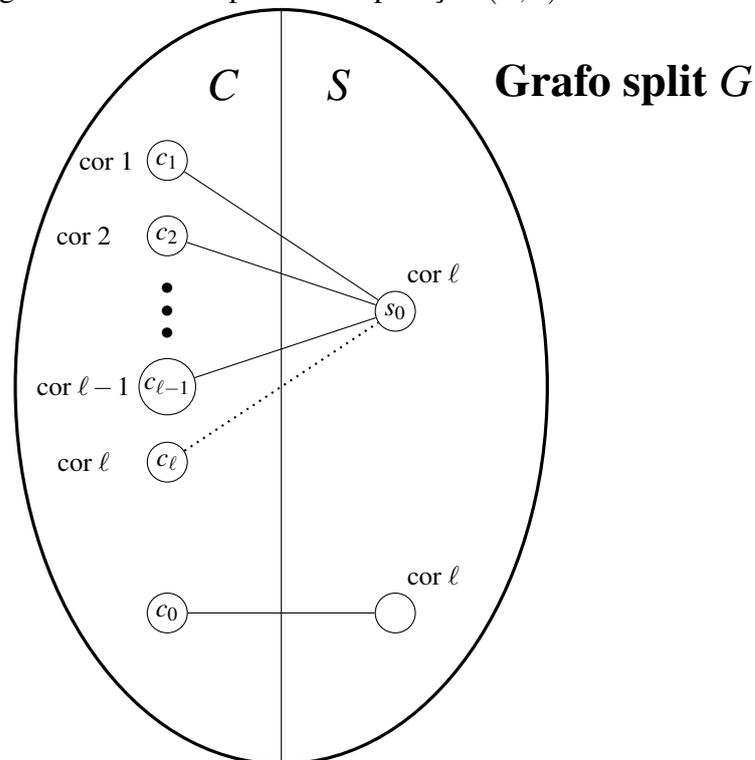
Sabemos que em grafos split $\Gamma(G) \in \{\chi(G), \chi(G) + 1\}$ (ARAÚJO; LINHARES-SALES, 2012). No teorema a seguir mostramos que $\Gamma'_g(G) = \Gamma_g(G) = \chi(G)$ em grafos split. O teorema também é usado na próxima seção para computar $\Gamma_g(G)$ em grafos P_4 -laden estentidos.

Teorema 7. *Seja G um grafo split com partição (C, S) tal que C é uma clique máxima. Então Alice vence usando $|C|$ cores com a seguinte estratégia mesmo se Bob começar o jogo e passar algumas de suas jogadas: se Bob colorir um vértice v de S na sua jogada, Alice colore um vértice não colorido e não vizinho de v que esteja em C , se existir; caso contrário, ela colore qualquer vértice de G dando preferência aos vértices de C . Consequentemente, $\Gamma'_g(G) = \chi(G)$ para grafos split.*

Demonstração. Note que cada vértice de S possui grau de no máximo $|C| - 1$ e, consequentemente, será colorido com uma cor de $\{1, \dots, |C|\}$ no Jogo da Coloração Gulosa. Agora mostraremos que o mesmo acontece para os vértices de C , isto é, os vértices de C serão coloridos com o conjunto de cores $\{1, \dots, |C|\}$ no Jogo da Coloração Gulosa seguindo a estratégia enunciada no teorema.

Seja c_0 um vértice não colorido de C escolhido em algum momento do jogo. Mostraremos agora que, em todos os casos, c_0 será colorido com uma cor de $\{1, \dots, |C|\}$. Se c_0 não possui vizinho colorido em S , então c_0 pode receber qualquer cor de $\{1, \dots, |C|\}$. Então, suponha que c_0 possui um vizinho colorido em S e seja ℓ a cor desse vizinho. Mostraremos agora que a cor ℓ já está também em algum vértice de C . Para isso, seja s_0 o primeiro vértice de S colorido com a cor ℓ no jogo. Então s_0 tem $\ell - 1$ vizinhos $c_1, \dots, c_{\ell-1}$ em C coloridos antes de s_0 com as cores $1, \dots, \ell - 1$, respectivamente. Veja a Figura 22.

Figura 22 – Grafo split G com partição (C, S)



Fonte: elaborado pelo autor.

Nota: C é uma clique, S é um conjunto independente, $c_1, \dots, c_{\ell-1}$ estão colorido com as cores $1, \dots, \ell - 1$, respectivamente, s_0 é o primeiro vértice de S colorido com a cor ℓ no jogo e o vértice vizinho de c_0 possui cor ℓ .

Se Alice coloriu s_0 , então, a partir da estratégia de Alice, todos os vértices de C estavam coloridos antes de s_0 e logo c_0 não existe, uma vez que deveria ser um vértice não colorido de C . Se Bob coloriu s_0 , então, a partir da estratégia de Alice temos duas possibilidades:

- Alice colore um vértice $c_\ell \in C$ com a cor ℓ não vizinho de s_0 imediatamente depois (se existir esse vértice não colorido); ou
- s_0 já possui um vértice não vizinho colorido $c_\ell \in C$ com a cor ℓ (lembre-se que já existem $c_1, \dots, c_{\ell-1}$ com as cores $1, \dots, \ell - 1$).

Em ambos os casos, concluímos que existe $c_\ell \in C$ colorido de ℓ . Com isso, concluímos que C

já possui um vértice com a cor ℓ antes de colorir c_0 . Uma vez que isso é válido para cada cor ℓ na vizinhança de c_0 em S , temos que todas as cores distintas que aparecem nos vizinhos de c_0 em S também aparecem nos vizinhos de c_0 em C . Então, c_0 pode ser colorido com uma cor de $\{1, \dots, |C|\}$.

Deste modo, cada vértices de C , no Jogo da Coloração Gulosa, será colorido com uma das cores de $\{1, \dots, |C|\}$. Portanto, Alice vencerá com $|C| = \chi(G)$ cores. \square

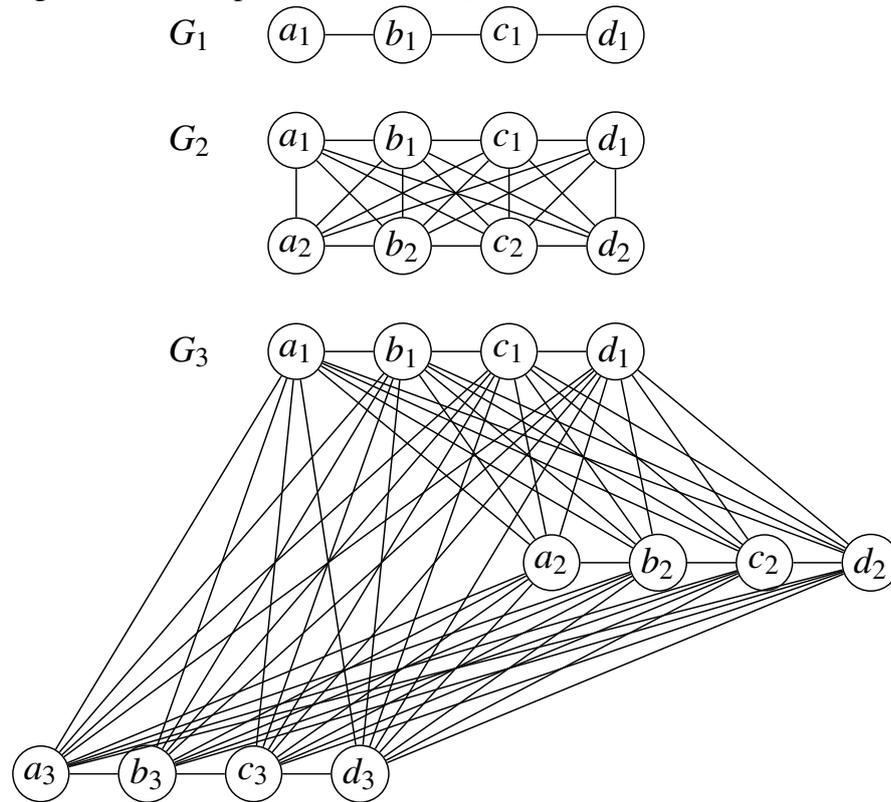
3.4 Número guloso de jogo em grafos com poucos P_4 's

Como mencionado na Introdução, Havet e Zhu mostraram em 2013 (HAVET; ZHU, 2013) que $\Gamma_g(G) = \chi(G)$ para um cografo qualquer. Nesta seção, provamos que essa igualdade também acontece para superclasses de cografos bem conhecidas, como grafos P_4 -esparso, P_4 -tidy e P_4 -laden, tais que $\Gamma(G)$ pode ser maior que $\chi(G)$ o quanto quisermos.

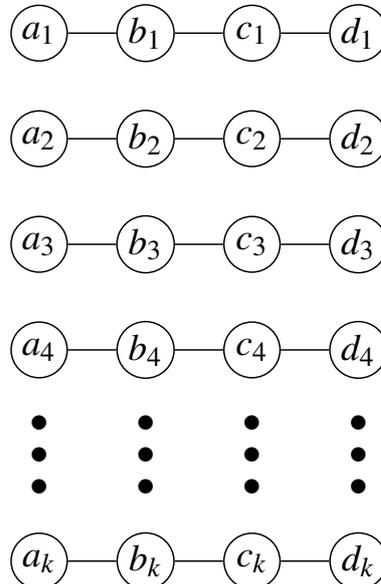
Como mencionado na Seção 2.1.4, uma motivação para desenvolver algoritmos para grafos P_4 -laden estendidos está no fato de que eles estão no topo de uma hierarquia amplamente estudada de muitas classes contendo grafos com poucos P_4 's (veja a Figura 14 na Seção 2.1.4), incluindo cografos, grafos P_4 -esparso, P_4 -lite, P_4 -laden e P_4 -tidy. Com isso, resolver problemas interessantes de um modo eficiente para grafos P_4 -laden estendidos implica imediatamente em algoritmos eficientes para todas essas classes. Outra motivação é que grafos P_4 -laden estendidos não estão na classe de grafos perfeitos (grafos em que, para cada subgrafo induzido H , temos que $\chi(H) = \omega(H)$) e, conseqüentemente, este trabalho obteve resultados não relacionados especificamente à perfeição.

Como exemplo, seja G_1 um P_4 $a_1b_1c_1d_1$. Para $k \geq 2$, seja G_k obtido de uma junção (operação definida na Seção 2.1.4) de G_{k-1} com um P_4 $a_kb_kc_kd_k$. Veja os grafos G_1 , G_2 e G_3 na Figura 23. Para um melhor entendimento, considere a Figura 24 com um G_k , onde as arestas entre os P_4 's são omitidas. Como cada conjunto de cinco vértices de um G_k , para $k \geq 2$, possui no máximo um P_4 induzido, temos que G_k é um P_4 -esparso.

A seqüência (G_1, G_2, G_3, \dots) de grafos P_4 -esparso satisfaz $\Gamma_g(G_k) = \chi(G_k) = 2k$ e $\Gamma(G_k) = 3k$. Para ver que $\chi(G_k) = 2k$, basta colorir a_i e c_i com cor $2i - 1$ e colorir b_i e d_i com cor $2i$ para cada $1 \leq i \leq k$. Além disso, para verificarmos que $\Gamma(G_k) = 3k$ basta colorir a_i e d_i com cor $3i - 2$ e colorir b_i e c_i com as cores $3i - 1$ e $3i$, respectivamente. Finalmente, para notarmos que $\Gamma_g(G_k) = 2k$, Alice pode sempre evitar que ambos os extremos a_i e d_i de um P_4 $a_ib_ic_id_i$ recebam a mesma cor. Se ela iniciar o jogo basta colorir um dos vértices centrais de

Figura 23 – P_4 -esparso: G_1 , G_2 e G_3 

Fonte: elaborado pelo autor.

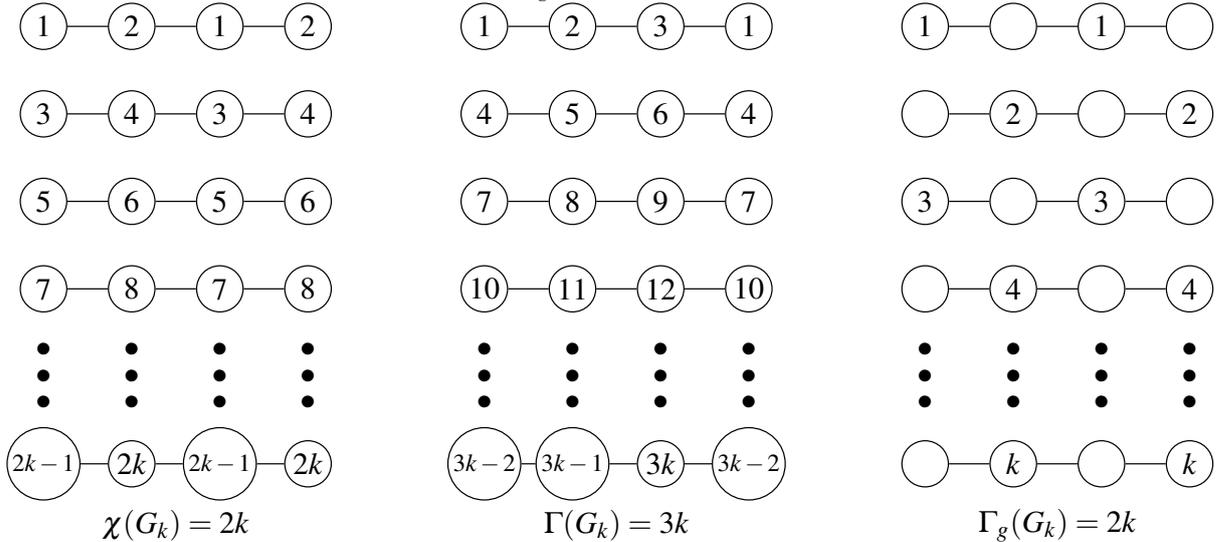
Figura 24 – P_4 -esparso: G_k 

Fonte: elaborado pelo autor.

Nota: para facilitar a visualização omitimos as arestas existentes entre cada P_4 .

algum P_4 e caso o Bob inicie o jogo ela colore um vértice com distância dois no mesmo P_4 que Bob jogou. Veja a Figura 25 para os três casos.

Note que $\Gamma_g(G) \leq \Gamma'_g(G)$ (lembre-se da Definição 1 de $\Gamma'_g(G)$ na seção anterior).

Figura 25 – $\chi(G_k) = 2k$, $\Gamma(G_k) = 3k$ e $\Gamma_g(G_k) = 2k$ 

Fonte: elaborado pelo autor.

Nota: para facilitar a visualização omitimos as arestas existentes entre cada P_4 e consideramos, sem perda de generalidade, que os vértices estão sendo coloridos em cada P_4 de cima para baixo.

Mostramos, a seguir, que $\Gamma'_g(G) = \chi(G)$ para grafos P_4 -laden estendidos, o que implica que $\Gamma_g(G) = \chi(G)$ nesta classe. A seguir, obtemos um limite superior nas operações de união e junção.

Lema 8. *Dados os grafos G_1 e G_2 ,*

- $\Gamma'_g(G_1 \vee G_2) \leq \Gamma'_g(G_1) + \Gamma'_g(G_2)$ e
- $\Gamma'_g(G_1 \cup G_2) \leq \max\{\Gamma'_g(G_1), \Gamma'_g(G_2)\}$.

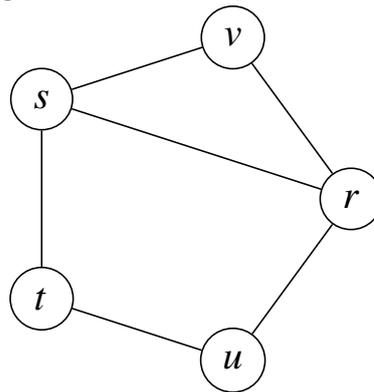
Demonstração. Apresentamos uma estratégia para Alice em $G_1 \cup G_2$ e $G_1 \vee G_2$. Se Bob colorir um vértice, Alice só precisa jogar no mesmo grafo que Bob jogou seguindo sua melhor estratégia naquele grafo, se o grafo possui um vértice não colorido. Se todos os vértices estão coloridos no grafo escolhido por Bob, Alice colore um vértice no outro grafo, considerando que Bob passou a vez nesse outro grafo. Se Bob não coloriu um vértice (ou seja, passou a sua jogada), Alice escolhe G_1 ou G_2 e joga seguindo sua melhor estratégia considerando que Bob passou sua jogada nesse grafo.

Sejam k_1 e k_2 os números de cores usadas em G_1 e G_2 , respectivamente. Então $\Gamma'_g(G_1 \cup G_2) \leq \max\{k_1, k_2\}$, uma vez que qualquer cor de G_1 pode ser usada em G_2 , e vice-versa. Além disso, $\Gamma'_g(G_1 \vee G_2) \leq k_1 + k_2$, uma vez que nenhuma cor de G_1 pode ser usada em G_2 , e vice-versa. Considerando que Alice usou sua melhor estratégia em ambos os grafos, temos que $k_1 \leq \Gamma'_g(G_1)$ e $k_2 \leq \Gamma'_g(G_2)$, uma vez que ela impede Bob de jogar duas vezes seguidas no mesmo grafo. \square

Lema 9. Se G é um grafo pseudo-split ou uma aranha (R, C, S) em que C é clique máxima, então $\Gamma'_g(G) \leq \Gamma'_g(G[R]) + |C|$. Se G é isomorfo a C_5 , P_5 , $\overline{P_5}$ ou K_1 , então $\Gamma'_g(G) = \chi(G)$.

Demonstração. Claramente $\Gamma'_g(K_1) = \chi(K_1) = 1$. Se G é isomorfo a C_5 , temos que $3 = \chi(C_5) \leq \Gamma'_g(C_5) \leq \Delta(C_5) + 1 = 3$ e logo $\Gamma'_g(C_5) = \chi(C_5)$. Se G é isomorfo a P_5 , temos que $\chi(P_5) = 2$, e $\Gamma'_g(P_5) \leq 2$, uma vez que Alice pode impedir Bob de colorir ambas as extremidades de um P_4 induzido (interno ao P_5) com a mesma cor (se Bob começar, Alice colore um vértice a distância dois com a mesma cor; caso contrário Alice colore o vértice do meio de P_5). Se G é isomorfo a $\overline{P_5}$, então G é obtido de um C_4 com vértices r, s, t, u incluindo um vértice v adjacente a r e s (veja a Figura 26). Então $\chi(\overline{P_5}) = 3$ e $\Gamma'_g(\overline{P_5}) \leq 3$, já que Alice pode garantir que os vértices r e s sejam coloridos até a terceira jogada usando as cores em $\{1, 2, 3\}$ e logo os únicos vértices $\{r, s\}$ de $\overline{P_5}$ que poderiam chegar na cor 4 terão cores até 3.

Figura 26 – Grafo $\overline{P_5}$



Fonte: elaborado pelo autor.

Agora assumamos que G seja um grafo pseudo-split ou aranha (R, C, S) . Se Bob colorir um vértice de R e se R ainda continuar com algum vértice não colorido, então Alice segue com sua melhor estratégia em $G[R]$. Se Bob colorir um vértice de $C \cup S$, então Alice segue a estratégia vencedora do Teorema 7 considerando o grafo split (C', S) (construído a partir de (R, C, S) removendo R e aumentando a clique C adicionando a ela $\Gamma'_g(G[R])$ vértices) e seguindo a ordem de prioridade para colorir os vértices: C, R (caso escolha um dos 3 vértices adicionados a C) e S . Essa ordem de prioridade será seguida por Alice, caso Bob passe alguma de suas jogadas. Pelo Teorema 7, Alice vencerá com $|C'| = |C| + \Gamma'_g(G[R])$ cores. \square

O próximo lema trabalha com quase-aranhas que não são aranhas. Isto é, G é obtido de uma aranha (R, C, S) com um vértice $v \in C \cup S$ duplicado por dois vértices v_1 e v_2 . Se $v \in C$ e $v_1 v_2$ é uma aresta ou $v \in S$ e $v_1 v_2$ não é aresta, então temos que $\Gamma'_g(G) = \chi(G)$ usando os mesmos

argumentos da prova do Lema 9. Se $v \in C$ e $v_1 v_2$ não é aresta, então eles sempre possuem a mesma cor, uma vez que eles possuem a mesma vizinhança e o modo de colorir é guloso, e consequentemente o jogo da coloração gulosa (permitindo que Bob comece o jogo e que ele passe algumas de suas jogadas) no grafo quase-aranha é praticamente o mesmo grafo aranha original. Então, no próximo lema, consideramos o caso em que $v_1, v_2 \in S$ e $v_1 v_2$ é uma aresta.

Lema 10. *Seja G um quase-aranha obtida de uma aranha (R, C, S) com um vértice $v \in S$ trocado por dois vértices adjacentes v_1 e v_2 . Então $\Gamma'_g(G) \leq \Gamma'_g(G[R]) + |C|$, se $R \neq \emptyset$ ou v tem pelo menos dois vértices não vizinhos em C . Caso contrário, $\Gamma'_g(G) = \chi(G)$.*

Demonstração. Seja $S' = (S \setminus \{v\}) \cup \{v_1, v_2\}$. A estratégia de Alice é a seguinte. Se Bob colorir um vértice de R , e R tem um vértice não colorido, então Alice segue com sua melhor estratégia em $G[R]$. Se Bob colorir um vértice de S' , então Alice colore um vértice não vizinho dele em C , se existe um vértice não colorido. Caso contrário, Alice colore qualquer vértice de G na ordem de prioridade: vértices de C , R e S' .

Uma vez que Alice segue sua melhor estratégia em $G[R]$ e todos os vértices de R são adjacentes a todos os vértices de C , então qualquer vértice de R pode ser colorido com uma cor de $\{1, \dots, \Gamma'_g(G[R]) + |C|\}$. Também note que cada vértice de $S' - \{v_1, v_2\}$ possui grau máximo $|C| - 1$ e consequentemente será colorido com uma cor de $\{1, \dots, |C|\}$ no Jogo da Coloração Gulosa.

Consideramos agora os vértices v_1 e v_2 . Uma vez que eles têm grau máximo $|C|$, então eles podem ser coloridos com uma cor de $\{1, \dots, |C| + 1\}$ no Jogo da Coloração Gulosa. Se $R \neq \emptyset$, então $\Gamma'_g(G[R]) \geq \chi(G[R]) \geq 1$ e logo Alice vencerá com as cores $\{1, \dots, |C| + 1\} \subseteq \{1, \dots, |C| + \Gamma'_g(G[R])\}$. Se v_1 e v_2 possuem pelo menos dois vértices não vizinhos em C , então eles possuem grau máximo $|C| - 1$ e podem ser coloridos com uma cor de $\{1, \dots, |C|\}$ no Jogo da Coloração Gulosa e logo Alice vencerá com $|C| + \Gamma'_g(G[R])$ cores. Veja que esse último argumento é verdadeiro tanto para $R = \emptyset$ quanto para $R \neq \emptyset$. Finalmente, se $R = \emptyset$ e v_1 e v_2 são vértices não adjacentes de exatamente um vértice c de C , então os vértices de $C \cup \{v_1, v_2\}$ serão colorido com as cores de $\{1, \dots, |C| + 1\}$ seguindo a estratégia de Alice, uma vez que os vértices de C possuem maior prioridade. Assim, também neste caso temos que $\Gamma'_g(G) = |C| + 1 = \chi(G)$. \square

Com isso, obtemos nosso teorema principal para grafos P_4 -laden estendidos.

Teorema 8. *Se G é um grafo P_4 -laden estendido, então $\Gamma_g(G) = \Gamma'_g(G) = \chi(G)$, o qual pode ser computado em tempo linear. Isto é, Alice ganha o Jogo da Coloração Gulosa com $\chi(G)$ cores em grafos P_4 -laden estendidos mesmo se Bob começar o jogo e puder passar qualquer de suas jogadas.*

Demonstração. O resultado segue de uma indução na quantidade de vértices utilizando os Teoremas 1 e 7, e Lemas 8, 9 e 10, uma vez que $\chi(G_1 \vee G_2) = \chi(G_1) + \chi(G_2)$ e $\chi(G_1 \cup G_2) = \max\{\chi(G_1), \chi(G_2)\}$ para quaisquer grafos G_1 e G_2 . \square

4 JOGO DO ESPIÃO

Neste capítulo apresentamos nossos resultados acerca do Jogo do Espião. Primeiro mostramos um limite superior para o número de guarda do produto forte de dois grafos gerais e mostramos grafos grades Rei gerais que chegam nesse limite superior e outros grafos Rei gerais em que o número de guarda são menores que esse limite superior. Também mostramos o valor exato do número de guarda no produto lexicográfico de dois grafos gerais para qualquer distância $d \geq 2$. Provamos que é possível decidir em tempo polinomial $O(n^{3k+2})$ se o espião possui uma estratégia vencedora quando o número de guardas k é fixo, para toda velocidade $s \geq 1$ e distância $d \geq 0$. Em outras palavras, o Jogo do Espião está em XP quando o parâmetro é o número de guardas. Como resultado negativo, provamos que o Jogo do Espião é $W[2]$ -difícil mesmo em grafos bipartidos quando o parâmetro é o número de guarda, para toda velocidade $s \geq 2$ e distância $d \geq 0$, estendendo, deste modo, os resultados de (COHEN *et al.*, 2018).

4.1 Jogo do Espião em grades Rei e produtos de grafos

Sejam G_1 e G_2 dois grafos. O *produto forte* $G_1 \boxtimes G_2$ é um grafo com o conjunto de vértices $V(G_1 \boxtimes G_2) = V(G_1) \times V(G_2)$ (produto cartesiano entre conjuntos) em que vértices distintos (u_1, u_2) e (v_1, v_2) são adjacentes em $G_1 \boxtimes G_2$ se e somente se:

- (a) $u_1 = v_1$ e $u_2v_2 \in E(G_2)$,
- (b) $u_2 = v_2$ e $u_1v_1 \in E(G_1)$ ou
- (c) $u_1v_1 \in E(G_1)$ e $u_2v_2 \in E(G_2)$.

Veja a Figura 27 para os exemplos de $P_3 \boxtimes P_5$ e $P_5 \boxtimes P_3$.

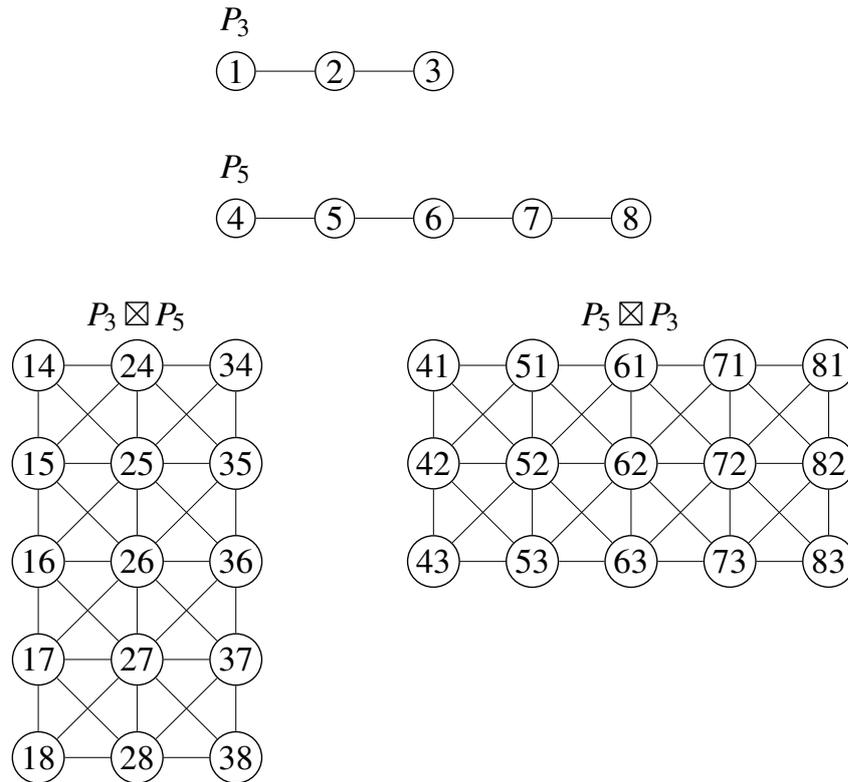
Dizemos que um grafo G é uma *grade Rei* se é um produto forte de dois caminhos, isto é, $G = P_n \boxtimes P_m$ para inteiros positivos m e n . O nome grade Rei foi motivado pelos movimentos do rei no jogo de xadrez, o que é equivalente a fazer movimentos ao longo de uma aresta por vez no grafo Rei $P_8 \boxtimes P_8$.

Inicialmente mostramos um limite superior para o número de guarda para o produto forte de dois grafos gerais.

Teorema 9. *Sejam $s \geq 2$ e $d \geq 0$. Dados dois grafos G_1 e G_2 , temos que*

$$gn_{s,d}(G_1 \boxtimes G_2) \leq gn_{s,d}(G_1) \times gn_{s,d}(G_2).$$

Além disso, temos a igualdade se $gn_{s,d}(G_1) = 1$ ou $gn_{s,d}(G_2) = 1$.

Figura 27 – Exemplos de grades Rei $P_3 \boxtimes P_5$ e $P_5 \boxtimes P_3$ 

Fonte: elaborado pelo autor.

Nota: no lado inferior direito no grafo $P_5 \boxtimes P_3$ as arestas horizontais foram criadas a partir da primeira condição da definição de produto forte (veja que em cada “linha” temos um P_5 induzido), as arestas das verticais foram criadas a partir da segunda condição da definição de produto forte (veja que em cada “coluna” temos um P_3 induzido) e as arestas diagonais foram criadas a partir da última condição da definição de produto forte. Veja também que $P_3 \boxtimes P_5$ e $P_5 \boxtimes P_3$ são isomorfos (você pode girar a Figura de $P_3 \boxtimes P_5$ em 90° e obter a Figura do $P_5 \boxtimes P_3$).

Demonstração. Sejam $k_1 = gn_{s,d}(G_1)$ e $k_2 = gn_{s,d}(G_2)$. A partir das estratégias vencedoras para os guardas em G_1 e G_2 com k_1 guardas em G_1 e k_2 guardas em G_2 , mostraremos uma estratégia vencedora em $G_1 \boxtimes G_2$ para os guardas com um total de $k_1 \cdot k_2$ guardas.

Considere o Jogo do Espião inicialmente com o grafo $G_1 \boxtimes G_2$. Seja o vértice (u_1, u_2) em que o espião é colocado no início do jogo. Seguindo as estratégias vencedoras dos guardas em G_1 e G_2 , considerando que o espião foi colocado em G_1 no vértice u_1 e que o espião foi colocado em G_2 no vértice u_2 , coloque $k_1 \cdot k_2$ guardas em $G_1 \boxtimes G_2$ do seguinte modo: para cada guarda g_1 colocado no vértice v_1 de G_1 e para cada guarda g_2 colocado no vértice v_2 de G_2 , coloque o guarda $g_1 g_2$ no vértice (v_1, v_2) de $G_1 \boxtimes G_2$.

Agora mostraremos que o espião está a uma distância de no máximo d de algum dos guardas colocados em $G_1 \boxtimes G_2$. Pela estratégia em G_1 (respectivamente G_2), temos que existe um vértice x (respectivamente y) com um guarda a uma distância de no máximo d do vértice u_1 em G_1 (respectivamente u_2 em G_2). Seja $x_0 x_1 \cdots x_p$ (respectivamente $y_0 y_1 \cdots y_q$) o menor caminho entre

u_1 e x em G_1 (respectivamente u_2 e y em G_2), onde $x_0 = u_1$ e $x_p = x$ (respectivamente $y_0 = u_2$ e $y_q = y$). Primeiro considere que $p = q$. Por definição, temos que $(x_0, y_0)(x_1, y_1) \cdots (x_p, y_q)$ é um caminho entre (u_1, u_2) e (x, y) em $G_1 \boxtimes G_2$ de tamanho $p = q$. Logo, temos um guarda em (x, y) a uma distância $p = q \leq d$ do espião que está em (u_1, u_2) . Veja que teremos a mesma conclusão para $p < q$, pois temos o caminho $(x_0, y_0)(x_1, y_1) \cdots (x_p, y_p)(x_p, y_{p+1})(x_p, y_{p+2}) \cdots (x_p, y_{q-1})(x_p, y_q)$. De modo análogo, a conclusão segue para $p > q$.

Nas movimentações do espião de um vértice (e_1, e_2) para um vértice (e'_1, e'_2) , o guarda $g_1 g_2$ irá do vértice (f_1, f_2) para (f'_1, f'_2) , onde o guarda g_1 vai do vértice f_1 para f'_1 na estratégia vencedora em G_1 e o guarda g_2 vai do vértice f_2 para f'_2 na estratégia vencedora em G_2 . Por definição, o movimento de cada guarda de (f_1, f_2) para (f'_1, f'_2) é totalmente possível e com os mesmos argumentos do parágrafo anterior, temos que após os movimentos realizados pelos guardas o espião estará a uma distância de no máximo d . Como temos que isto é verdadeiro para cada movimentação no jogo, temos que o jogador dos guardas vence com $k_1 \cdot k_2$ guardas, isto é, $gn_{s,d}(G_1 \boxtimes G_2) \leq gn_{s,d}(G_1) \times gn_{s,d}(G_2)$.

Por fim, se $gn_{s,d}(G_1) = 1$, então $gn_{s,d}(G_1 \boxtimes G_2) \leq gn_{s,d}(G_1) \times gn_{s,d}(G_2) \leq gn_{s,d}(G_2)$. Vemos facilmente pela definição que $gn_{s,d}(G_2) \leq gn_{s,d}(G_1 \boxtimes G_2)$. Assim, pela duas desigualdades podemos concluir que $gn_{s,d}(G_1 \boxtimes G_2) = gn_{s,d}(G_2)$. Analogamente temos que, se $gn_{s,d}(G_2) = 1$, então $gn_{s,d}(G_1 \boxtimes G_2) = gn_{s,d}(G_1)$. \square

Uma questão interessante sobre a desigualdade do Teorema 9 é para quais classes de grafos temos desigualdades estritas e a igualdade quando temos pelo menos dois guardas. Nos lemas a seguir, mostramos exemplos gerais em grafos grades Rei, considerando dois guardas. para desigualdade estrita $gn_{s,d}(P_{2d+3} \boxtimes P_{2d+3}) \leq 2 < 4 = gn_{s,d}(P_{2d+3})^2$ e para a igualdade $gn_{s,d}(P_{2d+4} \boxtimes P_{2d+4}) = 4 = gn_{s,d}(P_{2d+4})^2$ para $s \geq d + 2$.

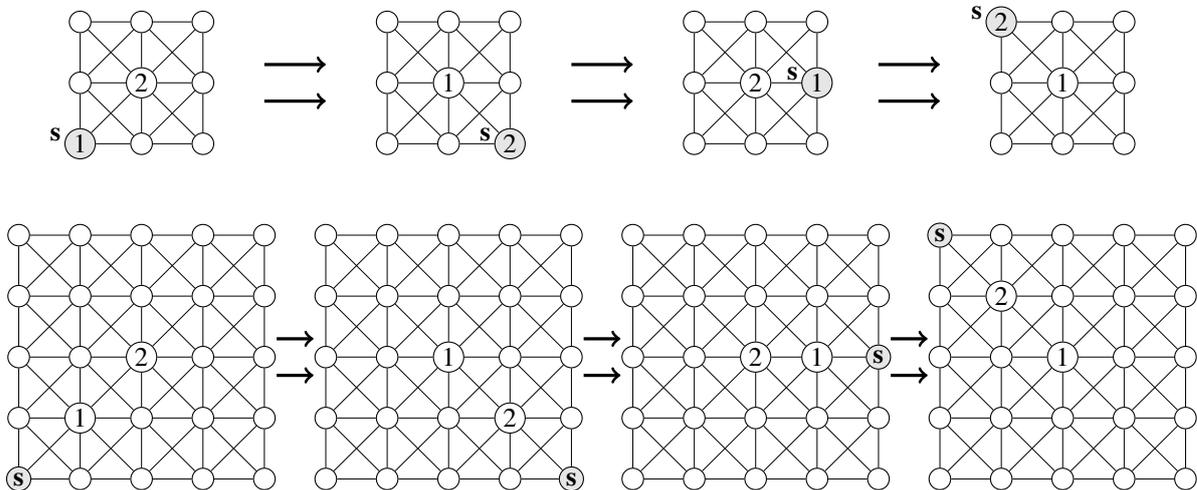
Primeiro mostramos que $gn_{s,d}(P_{2d+3} \boxtimes P_{2d+3}) \leq 2$ para cada $d \geq 0$ e $s \geq 2$. Considere agora que $s \geq 2d + 2$. Note que apenas um guarda com jogadas nos vértices centrais de um P_{2d+2} consegue vigiar o espião. Com isso, dois guardas conseguem vigiar $4d + 4$ vértices num caminho e logo $gn_{s,d}(P_{4d+4}) \leq 2$. Já em um P_{2d+3} com apenas um guarda, o espião pode ir de uma extremidade para a outra no caminho em uma única jogada e vemos que, depois dessa jogada, o espião vence, então $gn_{s,d}(P_{2d+3}) \geq 2$. Assim podemos concluir que $gn_{s,d}(P_{2d+3}) = 2$, uma vez que $2 \leq gn_{s,d}(P_{2d+3}) \leq gn_{s,d}(P_{4d+4}) \leq 2$ para $s \geq 2d + 2$. Teremos então que $gn_{s,d}(P_{2d+3} \boxtimes P_{2d+3}) \leq 2 < 4 = gn_{s,d}(P_{2d+3}) \times gn_{s,d}(P_{2d+3})$. Em outras palavras, os grafos $P_{2d+3} \boxtimes P_{2d+3}$ são exemplos de desigualdade estrita para o Teorema 9.

Lema 11. *Seja $d \geq 0$ e $s \geq 2$ inteiros fixos. Temos que $gn_{s,d}(P_{2d+3} \boxtimes P_{2d+3}) \leq 2$.*

Demonstração. Uma estratégia vencedora com dois guardas é sempre deixar um guarda no vértice central de $P_{2d+3} \boxtimes P_{2d+3}$ e o outro guarda (que fica a uma distância um do vértice central) vigiando o espião quando ele estiver em um vértice extremo de $P_{2d+3} \boxtimes P_{2d+3}$. Para entender melhor veja a Figura 28 que exemplifica a ideia em $P_3 \boxtimes P_3$ e $P_5 \boxtimes P_5$.

Considere que $V(P_{2d+3}) = \{-(d+1), \dots, 0, \dots, (d+1)\}$, onde i é vizinho de $i+1$ para $i = -(d+1), \dots, 0, \dots, d$. Faça $V(P_{2d+3} \boxtimes P_{2d+3}) = V \times V$. Estratégia vencedora com dois guardas: sempre deixe um guarda em $(0,0)$ (se um guarda sair de $(0,0)$ para vigiar o espião o outro move-se para $(0,0)$) e o outro guarda que vigia o espião em um vértice de $\{-1, 0, 1\} \times \{-1, 0, 1\}$ (vértice a distância de no máximo um de $(0,0)$). Veja que na estratégia os guardas sempre estão nos vértices centrais de $P_{2d+3} \boxtimes P_{2d+3}$ e que o guarda em $(0,0)$ vigia os vértices $C = \{-d, \dots, 0, \dots, d\} \times \{-d, \dots, 0, \dots, d\}$, enquanto o outro guarda pode vigiar o espião quando ele não estiver em algum vértice de C , uma vez que temos uma distância d entre um vértice de $\{-1, 1\} \times \{-1, 1\}$ e outro na extremidade de $P_{2d+3} \boxtimes P_{2d+3}$ (vértice fora de C). \square

Figura 28 – Exemplo de estratégia vencedora em $P_3 \boxtimes P_3$ e $P_5 \boxtimes P_5$ com dois guardas



Fonte: elaborado pelo autor.

Nota: o espião é representado por s e os guardas por 1 e 2.

A seguir mostramos que $gn_{s,d}(P_{2d+4} \boxtimes P_{2d+4}) = 4$ para qualquer $d \geq 0$ e $s \geq d+2$. Considere que $d \geq 0$ e $s \geq d+2$. Em um P_{2d+4} o espião consegue ir de uma extremidade para a outra em duas jogadas, enquanto um guarda não consegue vigiar uma extremidade e, em duas jogadas, vigiar a outra extremidade. Com isso, o espião consegue vencer apenas um guarda em P_{2d+4} indo de uma extremidade para outra em duas jogadas. Assim $2 \leq gn_{s,d}(P_{2d+4})$. Como

já visto $gn_{s,d}(P_{4d+4}) \leq 2$ para $d \geq 0$ e $s \geq 2$ e logo temos que $gn_{s,d}(P_{4d+4}) \leq 2$ para $d \geq 0$ e $s \geq d+2$. Portanto, $2 \leq gn_{s,d}(P_{2d+4}) \leq gn_{s,d}(P_{4d+4}) \leq 2$ que implica em $gn_{s,d}(P_{2d+4}) = 2$ para cada $d \geq 0$ e $s \geq d+2$. Então, temos que $gn_{s,d}(P_{2d+4} \boxtimes P_{2d+4}) = 4 = gn_{s,d}(P_{2d+4}) \times gn_{s,d}(P_{2d+4})$. Com isso, os grafos $P_{2d+4} \boxtimes P_{2d+4}$ são exemplos de igualdade para o Teorema 9.

Lema 12. *Seja $d \geq 0$ um inteiro fixo e $s \geq d+2$. Então $gn_{s,d}(P_{2d+4} \boxtimes P_{2d+4}) = 4$.*

Demonstração. Pelo Teorema 9, temos que $gn_{s,d}(P_{2d+4} \boxtimes P_{2d+4}) \leq gn_{s,d}(P_{2d+4}) \times gn_{s,d}(P_{2d+4}) = 2 \times 2 = 4$, isto é, $gn_{s,d}(P_{2d+4} \boxtimes P_{2d+4}) \leq 4$.

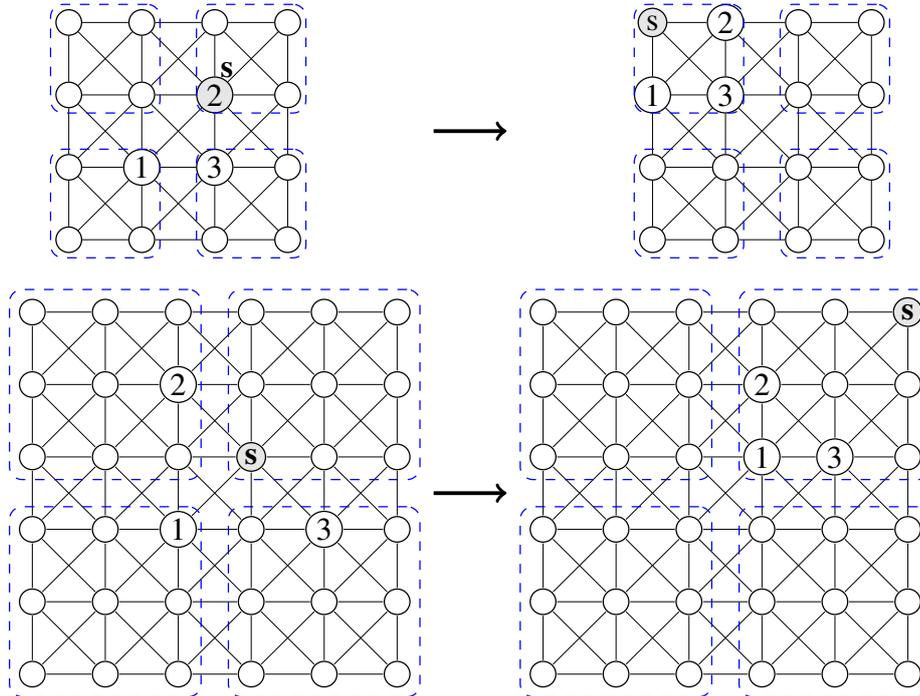
Para concluirmos a igualdade $gn_{s,d}(P_{2d+4} \boxtimes P_{2d+4}) = 4$, basta mostrar que o espião consegue vencer três guardas em $P_{2d+4} \boxtimes P_{2d+4}$ com velocidade $s \geq d+2$. Então considere que $G = P_{2d+4} \boxtimes P_{2d+4}$, onde $V(G) = \{-(d+2), \dots, 0, \dots, (d+1)\} \times \{-(d+2), \dots, 0, \dots, (d+1)\}$. Considere ainda que temos três guardas e que $s \geq d+2$. Diremos que os vértices $\{-(d+2), d+1\} \times \{-(d+2), d+1\}$ são as “pontas” de $P_{2d+4} \boxtimes P_{2d+4}$. Para um guarda vigiar uma “ponta” em até uma jogada tem que estar a uma distância de no máximo $d+1$ da “ponta” (d da vigilância e 1 de uma jogada em direção a “ponta”). Com isso apenas um guarda não consegue vigiar duas “pontas” em até uma jogada, pois a distância entre duas “pontas” é $2d+3$ e o raio de ação de guarda em até uma jogada é $2(d+1)$. Assim, três guardas não conseguem vigiar as quatro “pontas” de $P_{2d+4} \boxtimes P_{2d+4}$ em até uma jogada, uma vez que um guarda teria que conseguir vigiar duas em até uma jogada. Com isso sempre existirá uma “ponta” p que os guardas não conseguem vigiar, mesmo com uma jogada. Uma estratégia vencedora para o espião é colocá-lo no vértice $(0,0)$ e depois que os guardas forem colocados no grafo, o espião irá para p . O espião vencerá, pois os guardas não conseguirão vigiar p mesmo ainda tendo uma jogada. Veja exemplos em $P_4 \boxtimes P_4$ e $P_6 \boxtimes P_6$ na Figura 29. \square

Os lemas anteriores mostram exemplos com poucos guardas. Então mostramos, a seguir, que $gn_{s,d}(P_n \boxtimes P_n)$ pode ficar bem próximo de $gn_{s,d}(P_n)^2$, quando muito guardas são necessários. Especificamente, mostramos que $gn_{s,d}(P_n \boxtimes P_n) \geq (gn_{s,d}(P_n) - 1)^2$ em muitos casos, onde $gn_{s,d}(P_n)$ é um inteiro positivo qualquer.

Lema 13. *Sejam $d \geq 0$ e $2 \leq k \leq 2d+2$ inteiros fixos e $s \geq (k-1)(2d+3)$. Então $gn_{s,d}(P_{k(2d+3)}) = k+1$ e $k^2 \leq gn_{s,d}(P_{k(2d+3)} \boxtimes P_{k(2d+3)}) \leq (k+1)^2$.*

Demonstração. Podemos particionar os vértices de $P_{k(2d+3)} \boxtimes P_{k(2d+3)}$ em k^2 subconjunto de vértices que induzem $P_{2d+3} \boxtimes P_{2d+3}$. Se um desses subconjuntos não possuem um guarda em

Figura 29 – Exemplo de estratégia vencedora para o espião em $P_4 \boxtimes P_4$ e $P_6 \boxtimes P_6$ com três guardas



Fonte: elaborado pelo autor.

Nota: o espião é representado por s e os guardas por 1, 2 e 3. Temos $(s, d) = (2, 0)$ e $(s, d) = (3, 1)$ para $P_4 \boxtimes P_4$ e $P_6 \boxtimes P_6$, respectivamente.

algum momento do jogo, o espião pode ir para o vértice no centro desse subconjunto e nenhum guarda vigiará o espião, que vencerá o jogo, uma vez a distância do centro para a borda é $d + 1$ em $P_{2d+3} \boxtimes P_{2d+3}$. O espião pode realizar essa jogada se sua velocidade é pelo menos $(k - 1)(2d + 3)$. Com isso, note que o diâmetro é $k(2d + 3) - 1$ e a distância máxima entre dois centros dos subconjuntos induzem $P_{2d+3} \boxtimes P_{2d+3}$ é no máximo $k(2d + 3) - 1 - 2(d + 1) = (k - 1)(2d + 3)$.

Por isso, pelo menos k^2 guardas são necessários. Além disso, em (COHEN *et al.*, 2018), o valor exato de $gn_{s,d}(P_n)$ para $s \geq 2$, $d \geq 0$ e $n \geq 2$):

$$gn_{s,d}(P_n) = \left\lceil \frac{n}{2d + 2 + \lfloor \frac{2d}{s-1} \rfloor} \right\rceil$$

Com isso, temos que $gn_{s,d}(P_{k(2d+3)}) = k + 1$ para cada $d \geq 0$, $2 \leq k \leq 2d + 2$ e $s \geq (k - 1)(2d + 3) > 2d + 1$. Então, do Teorema 9, $(k + 1)^2$ guardas são suficientes. \square

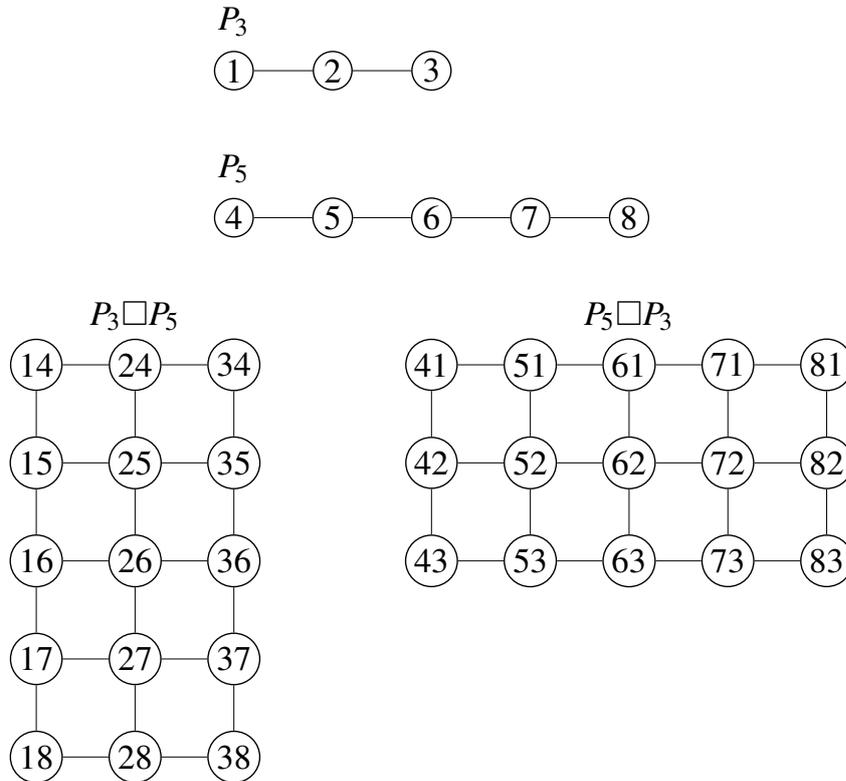
Existem outros produtos de grafos como o produto cartesiano $G_1 \square G_2$ e o produto lexicográfico $G_1 \cdot G_2$ de dois grafos G_1 e G_2 (veja (HAMMACK *et al.*, 2011) como referência) e em ambos os produtos, cartesiano e lexicográfico, temos que o conjunto de vértices é $V(G_1) \times V(G_2)$.

No produto cartesiano $G_1 \square G_2$, (u_1, u_2) e (v_1, v_2) são adjacentes se e somente se

- (a) $u_1 = v_1$ e $u_2v_2 \in E(G_2)$ ou
 (b) $u_2 = v_2$ e $u_1v_1 \in E(G_1)$.

Na Figura 30, temos exemplos para os produtos cartesiano $P_3 \square P_5$ e $P_5 \square P_3$.

Figura 30 – Exemplos de produto cartesiano $P_3 \square P_5$ e $P_5 \square P_3$



Fonte: elaborado pelo autor.

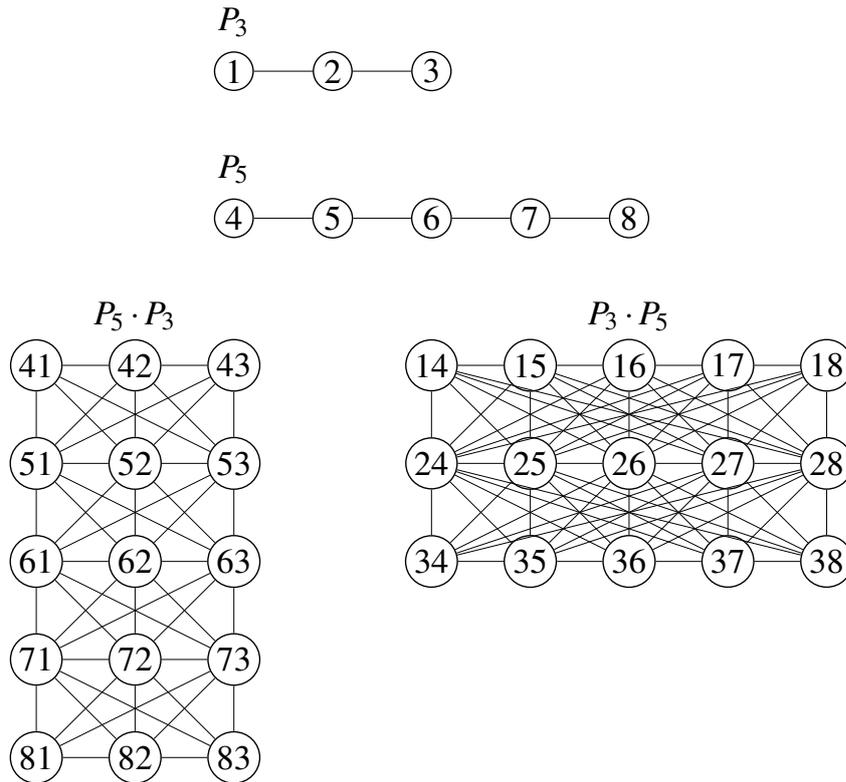
Nota: no lado inferior direito no grafo $P_5 \square P_3$ as arestas horizontais foram criadas a partir da primeira condição da definição de produto cartesiano (veja que em cada “linha” temos um P_5 induzido) e as arestas das verticais foram criadas a partir da segunda condição da definição de produto cartesiano (veja que em cada “coluna” temos um P_3 induzido). Veja também que $P_3 \square P_5$ e $P_5 \square P_3$ são isomorfos (você pode girar a Figura de $P_3 \square P_5$ em 90° e obter a Figura do $P_5 \square P_3$).

No produto lexicográfico $G_1 \cdot G_2$, (u_1, u_2) e (v_1, v_2) são adjacentes se e somente se

- (a) $u_1 = v_1$ e $u_2v_2 \in E(G_2)$ ou
 (b) $u_1v_1 \in E(G_1)$.

Na Figura 31, temos exemplos para os produtos lexicográfico $P_3 \cdot P_5$ e $P_5 \cdot P_3$.

Vemos, pela definição, que o produto cartesiano $G_1 \square G_2$ é um subgrafo de um produto forte $G_1 \boxtimes G_2$, que, por sua vez, é subgrafo de um produto lexicográfico $G_1 \cdot G_2$ e $G_2 \cdot G_1$. Facilmente também vemos exemplos em que o limite superior do Teorema 9 não pode ser aplicado no produto cartesiano. Por exemplo, $gn_{2,0}(P_2 \square P_2) = gn_{2,0}(C_4) = 2 > 1 = gn_{2,0}(P_2)^2$. Já no produto lexicográfico, os exemplos em que a desigualdade do Teorema 9 não é verdadeira não são fáceis de encontrar. No lema a seguir mostra-se um exemplo em que o limite superior do

Figura 31 – Exemplo de grade Rei $P_3 \cdot P_5$ 

Fonte: elaborado pelo autor.

Nota: no lado inferior direito no grafo $P_5 \cdot P_3$ as arestas horizontais foram criadas a partir da primeira condição da definição de produto lexicográfico (veja que em cada “linha” temos um P_5 induzido), as arestas das verticais/diagonais foram criadas a partir da segunda condição da definição de produto lexicográfico (veja que em cada “coluna” com “diagonal” temos um P_3 induzido). Veja que, diferentemente do produto cartesiano e do produto forte, $P_3 \cdot P_5$ e $P_5 \cdot P_3$ não são grafos isomorfos.

Teorema 9 falha no produto cartesiano e no produto lexicográfico.

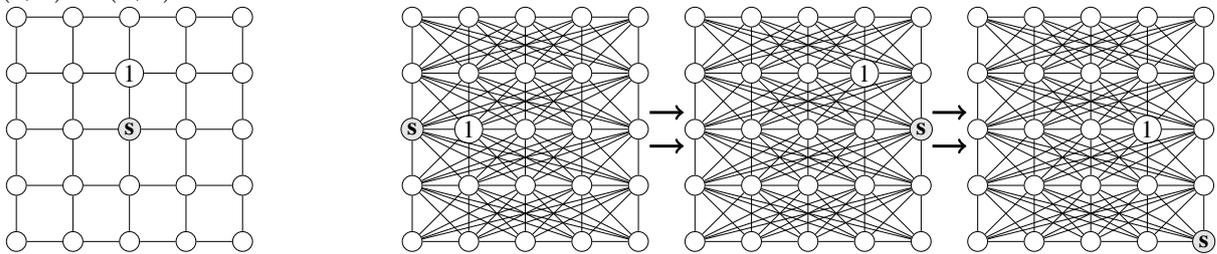
Lema 14. $gn_{2,1}(P_5 \square P_5) > gn_{2,1}(P_5)^2$ e $gn_{2,1}(P_5 \cdot P_5) > gn_{2,1}(P_5)^2$.

Demonstração. Veja $P_5 \square P_5$ e $P_5 \cdot P_5$ na Figura 32. Considere que $V(P_5) = \{-2, -1, 0, 1, 2\}$. Facilmente vemos que $gn_{2,1}(P_5) = 1$: um guarda em 0 movendo-se para -1 e 1 consegue vigiar os vértices -2 e 2 , e o espião não consegue ir de -2 para 2 em apenas uma jogada (e vice-versa). Contudo, em $P_5 \square P_5$, o espião em $(0, 0)$ pode ir para uma “ponta” $(-2, -2)$, $(-2, 2)$, $(2, -2)$ e $(2, 2)$ que estiver mais distante do guarda em duas jogadas e será impossível para apenas um guarda vigiá-lo, pois no melhor caso o guarda estará a uma distância 4 da “ponta” e em duas jogadas conseguirá diminuir a distância para 2.

Agora considere $P_5 \cdot P_5$. Mostraremos uma estratégia vencedora para o espião contra um guarda. Inicialmente, o espião é colocado em $(-2, 0)$. O guarda deve ser colocado em $(-2, 0)$ ou $(-1, 0)$, pois deve estar a distância até 1 do espião e se não estiver na linha 0 (vértices $(x, 0)$, para $x = -2, -1, 0, 1, 2$) o espião irá para $(-2, -2)$ ou $(-2, 2)$ e o guarda não conseguirá vigiá-lo.

Sem perda de generalidade, assuma que o guarda foi colocado em $(-1, 0)$. Veja essa situação do jogo na primeira configuração da Figura 32 em $P_5 \cdot P_5$. No primeiro movimento, o espião irá para $(2, 0)$. Para não perder o jogo, o guarda deve ir para um vértice em $\{-2, -1, 0, 1, 2\} \times \{-1, 1\}$, isto é, ir para um vértice na linha -1 ou na linha 1 . Se o guarda for para linha 1 , supondo sem perda de generalidade, em $(2, 1)$ (veja na segunda configuração da Figura 32 em $P_5 \cdot P_5$), então o espião irá para $(-2, -2)$ e vencerá, pois a distância de $(2, 1)$ para $(-2, -2)$ é três e o guarda consegue diminuir para distância dois com apenas uma jogada. De modo análogo o espião, também vencerá quando o guarda for para linha -1 . \square

Figura 32 – $P_5 \square P_5$ e $P_5 \cdot P_5$ com uma estratégia vencedora para o espião contra um guarda e $(s, d) = (2, 1)$



Fonte: elaborado pelo autor.

Nota: o espião é representado como s e o guarda por 1 .

No entanto, considerando o produto lexicográfico, podemos mostrar um limite superior geral melhor, se a distância de vigilância for $d \geq 2$. Vemos isso no seguinte resultado.

Teorema 10. *Sejam $s \geq 2$, $d \geq 2$ inteiros e G_1 e G_2 dois grafos. Se G_1 não possui vértices isolados, então*

$$gn_{s,d}(G_1 \cdot G_2) = gn_{s,d}(G_1).$$

Em caso contrário,

$$gn_{s,d}(G_1 \cdot G_2) = \max \{ gn_{s,d}(G_1), gn_{s,d}(G_2) \}.$$

Demonstração. Sejam $s \geq 2$, $d \geq 2$ inteiros e G_1 e G_2 dois grafos. Primeiro assuma que G_1 não possui vértice isolado, isto é, cada vértice de G_1 possui um vizinho. Note que cada guarda posicionado em (u_1, u_2) de $G_1 \cdot G_2$ vigia (u_1, v_2) com $v_2 \in V(G_2)$, uma vez que a distância de vigilância é pelo menos 2 e u_1 possui um vizinho v_1 , o que leva a termos o caminho $(u_1, u_2) - (v_1, v_2) - (u_1, v_2)$ de tamanho 2. Portanto, os guardas seguem com sua estratégia vencedora em G_1 com $gn_{s,d}(G_1)$ guardas para vencer o espião em $G_1 \cdot G_2$, movendo-se apenas

pelos vértices em $\{(v_1, b) : v_1 \in V(G_1)\}$, onde b é um vértice arbitrário de G_2 . Em outras palavras, para cada movimentação vencedora em G_1 quando um guarda move-se de x para y , então em $G_1 \cdot G_2$ o guarda move-se de (x, b) para (y, b) . Como temos sempre um guarda a uma distância de no máximo d do espião em G_1 através de um caminho $x_1x_2 \cdots x_{k-1}x_k$, então sempre temos o mesmo em $G_1 \cdot G_2$ uma vez que $(x_1, b)(x_2, b) \cdots (x_{k-1}, b)(x_k, c)$ é um caminho de no máximo d em $G_1 \cdot G_2$, onde (x_k, c) é o vértice em que o espião está.

Caso G_1 tenha um vértice isolado, então $G_1 \cdot G_2$ contém uma componente conexa maximal isomorfa a G_2 , que precisa de pelo menos $gn_{s,d}(G_2)$ guardas para vigiar o espião. Com isso, se o espião escolher um vértice em uma componente isomorfa a G_2 no início do jogo, $gn_{s,d}(G_2)$ guardas são necessários e suficientes para vigiar o espião. Caso contrário, $gn_{s,d}(G_1)$ guardas são necessários e suficientes, como já observado no parágrafo anterior. No pior caso, serão necessários o maior número de guarda entre $gn_{s,d}(G_1)$ e $gn_{s,d}(G_2)$. \square

4.2 O Jogo do Espião está em XP quando o parâmetro é o número de guardas

Nesta seção provamos que, se o número k de guardas é fixo, o Jogo do Espião é solucionável em tempo polinomial $O(n^{3k+2})$ para cada velocidade $s \geq 2$ e distância de vigilância $d \geq 0$. Para isso, seguiremos com as definições abaixo.

Definição 2. *Dados $k \geq 1$, $s \geq 1$, $d \geq 0$ e um grafo G , definimos:*

- configuração de jogo: *um cenário possível do Jogo do Espião depois de colocar todos os guardas e o espião em G , isto é, temos k guardas e um espião nos vértices de G (lembre-se que os guardas com o espião podem ocupar o mesmo vértice).*
- configuração espiã: *configuração de jogo em que o jogador do espião fará a próxima jogada.*
- configuração guarda: *configuração de jogo em que o jogador dos guardas fará a próxima jogada.*

Facilmente vemos que existem no máximo $2n^{k+1}$ configurações de jogo, pois temos n escolhas para colocar o espião e n escolhas para colocar cada guarda, além de que cada configuração pode ser de dois tipos: espiã ou guarda.

De fato, é fácil ver que o número de configurações é igual a $2n$ vezes o número de soluções da equação $x_1 + \dots + x_n = k$ onde $x_1, \dots, x_n \in \mathbb{N}$, considerando que x_v é o número

de guardas no vértice v (combinação com repetição). Portanto, o número de configurações é exatamente $2n \cdot \binom{n+k-1}{k} \leq 2n^{k+1}$.

Dizemos que uma configuração espiã C_1 *leva* a uma configuração guarda C_2 se C_2 pode ser obtida de C_1 por um movimento do espião ao longo de no máximo s arestas. Dizemos que uma configuração guarda C_1 *leva* a uma configuração espiã C_2 se C_2 pode ser obtida de C_1 pelos movimentos dos guardas ao longo de no máximo uma aresta para cada guarda.

Definimos o *dígrafo* D^* do seguinte modo: para cada configuração espiã ou guarda C , existe um vértice associado v_C em D^* . Se a configuração C_1 *leva* a configuração C_2 , adicionamos em D^* um arco direcionado de v_{C_1} para v_{C_2} .

Na demonstração do teorema a seguir mostramos como proceder no dígrafo D^* para decidir se o espião possui uma estratégia vencedora.

Teorema 11. *Seja G um grafo com n vértices. Dados $k \geq 1$, $s \geq 2$ e $d \geq 0$, decidir se o espião possui uma estratégia vencedora no Jogo do Espião é um problema em XP com tempo $O(n^{3k+2})$.*

Demonstração. O algoritmo com tempo $O(n^{3k+2})$ é como se segue. Primeiro, defina como uma *configuração espiã vencedora* uma configuração espiã tal que o espião está a uma distância maior que d de cada guarda. Marque todos os vértices de D^* associados a uma configuração espiã vencedora. A partir de agora nossa ideia é marcar as configurações que chegam através de um caminho forçado pelo espião até alguma configuração espiã vencedora. Repita os dois passos a seguir até que nenhum vértice de D^* possa ser marcado:

1. Para cada configuração guarda C , marque o vértice v_C em D se todos os vizinhos saindo de v_C estão marcados (em outras palavras, qualquer movimento dos guardas nessa configuração *leva* para uma configuração espiã que pode chegar forçadamente pelo espião para uma configuração espiã vencedora); e
2. Para cada configuração espiã C , marque o vértice v_C em D se existe pelo menos um vizinho saindo de v_C que esteja marcado (em outras palavras, existe um movimento do espião que *leva* para uma configuração guarda que pode chegar forçadamente pelo espião para uma configuração espiã vencedora).

Ao final das marcações, se existe um vértice u de G tal que, para cada configuração espiã C com o espião ocupando o vértice u , v_C está marcado em D^* , então o espião possui uma estratégia vencedora (sendo colocado em u na primeira jogada). Caso contrário, os guardas possuem uma estratégia vencedora.

Aplicando busca em largura em cada vértice de G como pré-processamento para checar vértices que podem ser atingidos pelo espião com velocidade s e também vértices que são vigiados por guardas considerando distância d , podemos obter todos os vizinhos saindo de um vértice v_C em D^* .

Criamos $O(n^{k+1})$ vértices (total de configurações). Para cada configuração espiã criaremos no máximo n arestas que saem dela, pois o espião terá no máximo n possibilidades (contamos aqui a possibilidade do espião permanecer no mesmo vértice). Para cada configuração guarda, temos no máximo n possibilidades para cada guarda, o que nos leva a no máximo n^k arestas saindo de cada configuração guarda. Deste modo, teremos um total de $O(n^{k+1}) \cdot n + O(n^{k+1}) \cdot n^k = O(n^{2k+1})$ aresta em D^* . Portanto, a criação de D^* (vértices e arestas) leva tempo $O(n^{k+1}) + O(n^{2k+1}) = O(n^{2k+1})$.

Considere que uma iteração é uma execução dos dois passos que marcam configurações espiãs e configurações guardas. O algoritmo fará no máximo $O(n^{k+1})$ iterações, uma vez que pelo menos um vértice é marcado em cada iteração. Além disso, como cada configuração leva a no máximo n^k configurações, então cada iteração leva tempo $O(n^{k+1}) \cdot n^k = O(n^{2k+1})$.

Portanto, todo algoritmo, considerando o tempo de criação de D^* mais o tempo de todas as iterações, possui tempo $O(n^{2k+1}) + O(n^{k+1}) \cdot O(n^{2k+1}) = O(n^{3k+2})$. \square

4.3 Jogo do Espião está em FPT em grafos com poucos P_4 's

Nesta seção mostramos os valores de $gn_{s,d}(G_1 \cup G_2)$ e $gn_{s,d}(G_1 \vee G_2)$, para dois grafos G_1 e G_2 em que $s \geq 2$ e $d \geq 0$. Também mostramos o número de guarda em grafos aranhas e grafos que possuem p -componentes separáveis. Por fim, mostramos um algoritmo FPT para o Jogo do Espião em grafos com poucos P_4 's, que é um dos resultados mais importantes deste capítulo, uma vez que resolve o jogo em muitas classes de grafos.

Teorema 12. [Decomposição Primeval (BABEL; OLARIU, 1998)] *Se G é um $(q, q-4)$ -grafo, então temos que um dos itens abaixo é verdadeiro:*

- (a) G é uma união disjunta ou uma junção de dois $(q, q-4)$ -grafos;
- (b) G é uma aranha (R, C, S) e $G[R]$ é um $(q, q-4)$ -grafo;
- (c) G contém uma p -componente separável H com $|V(H)| < q$ e partição (H_1, H_2) , tal que $G - H$ é um $(q, q-4)$ -grafo e cada vértice de $G - H$ é adjacente a cada vértice de H_1 e não é adjacente a cada vértice de H_2 ; ou

(d) G possui no máximo q vértices.

Como consequência, um $(q, q - 4)$ -grafo G pode ser decomposto em sucessivas aplicações do Teorema 12 do seguinte modo:

- se (a) é verdadeiro, aplicamos o teorema em cada componente de G ou \bar{G} ;
- se (b) é verdadeiro, aplicamos o teorema em $G[R]$ e
- se (c) é verdadeiro, aplicamos o teorema em $G - H$.

Essa decomposição pode ser obtida em tempo linear (BAUMANN, 1996; JAMISON; OLARIU, 1992), então o problema do Jogo do Espião pode ser resolvido a partir das estratégias em cada um dos seus quatro casos.

Agora mostramos nossos resultados para uniões e junções.

Lema 15. *Sejam G_1 e G_2 dois grafos e $s \geq 2$ e $d \geq 0$ dois inteiros. Então*

$$gn_{s,d}(G_1 \cup G_2) = \max\{gn_{s,d}(G_1), gn_{s,d}(G_2)\}.$$

Além disso,

$$gn_{s,d}(G_1 \vee G_2) = \begin{cases} 1, & \text{se } d \geq 1 \text{ ou } G_1 \text{ e } G_2 \text{ são grafos completos,} \\ 2, & \text{caso contrário.} \end{cases}$$

Demonstração. Em $G_1 \cup G_2$, na primeira jogada o jogador do espião deve escolher um vértice de G_1 ou G_2 e então todos os guardas serão colocado nos vértices do mesmo grafo (G_1 ou G_2) que o espião foi colocado e os movimentos serão apenas nesse grafo. Assim $gn_{s,d}(G_1 \cup G_2) = \max\{gn_{s,d}(G_1), gn_{s,d}(G_2)\}$.

Agora considere a junção $G_1 \vee G_2$.

Se $d \geq 1$, então um guarda é suficiente para vigiar o espião seguindo a estratégia: se o espião está em G_1 (respectivamente G_2), o guarda segue para G_2 (respectivamente G_1). O guarda sempre vigiará, pois cada vértice de G_1 é vizinho de cada vértice de G_2 em $G_1 \vee G_2$ (e vice-versa).

Se $d = 0$ e G_1 e G_2 são grafos completos, então $G_1 \vee G_2$ é também um grafo completo e consequentemente um guarda é suficiente para vigiar o espião com a seguinte estratégia: apenas move-se para o vértice em que o espião está, em cada jogada. Os movimentos do guarda são possíveis, uma vez que os vértices de $G_1 \vee G_2$ estão todos ligados entre si, pois $G_1 \vee G_2$ é um grafo completo.

Agora assumamos que $d = 0$ e sem perda de generalidade G_1 não é completo. Seja v_1 e v_2 dois vértices não adjacentes de G_1 . Como o espião pode mover-se de v_1 para v_2 em uma jogada passando por um vértice de G_2 e apenas um guarda precisa de duas jogadas para ir de v_1 para v_2 (lembre-se que $d = 0$), então mais de um guarda é necessário para vigiar o espião. Então considere dois guardas para vigiar o espião. Se mantermos, em cada rodada, um guarda no mesmo vértice do espião em G_1 (respectivamente G_2) e o segundo guarda no outro grafo G_2 (respectivamente G_1), temos que apenas dois guardas são suficientes para vigiar em $G_1 \vee G_2$. Note também que todas as argumentações são válidas para $d > 0$. \square

O próximo lema determina o número de guarda em grafos aranha.

Lema 16. *Seja G uma aranha (R, C, S) . Se $d \geq 1$, então $gn_{s,d}(G) = 1$. Além disso, se G é uma aranha magra, então*

$$gn_{s,0}(G) = \begin{cases} |C|, & \text{se } R = \emptyset, \\ |C| + 1, & \text{caso contrário,} \end{cases}$$

e, se G é uma aranha gorda, então

$$gn_{s,0}(G) = \begin{cases} 2, & \text{se } R = \emptyset, \\ 3, & \text{caso contrário,} \end{cases}$$

Demonstração. Pela definição de aranha, cada vértice de G possui um vizinho na clique C . Assim, se $d \geq 1$, então um guarda em C é suficiente para vencer: apenas move-se para um vértice também em C a distância 1 do espião. Considere então, que $d = 0$. Caso $s = 1$, então colocamos apenas um guarda no mesmo vértice do espião e o guarda apenas realiza os mesmos movimentos do espião e assim vigiará com $d = 0$. Considere então, que $s > 1$.

Assumamos que G é uma aranha magra. Uma vez que o espião pode mover-se com uma jogada de um vértice de $C \cup R$ para qualquer vértice de S , então cada vértice de S deve ter um guarda distinto em sua vizinhança fechada antes do movimento do espião e logo devemos ter um guarda para cada vértice de S . Assim, devemos ter $|S| = |C|$ guardas para vigiar o espião quando ele estiver em algum vértice de $N[S] = C \cup S$. Caso $R \neq \emptyset$ e o espião mova-se para R , então devemos ter mais um guarda para vigiá-lo em R , uma vez que não podemos ter menos que $|S| = |C|$ guardas para vigiar o espião quando ele estiver em $N[S]$. Portanto, $|S| = |C|$ guardas são necessários se $R = \emptyset$ e $|C| + 1$ guardas são necessários se $R \neq \emptyset$. Além disso, se $R = \emptyset$, $|C|$ guardas são suficientes para vencer apenas mantendo um guarda vigiando cada vértice de C e

seu vértice vizinho em S quando o espião mover-se para ele. Finalmente, se $R \neq \emptyset$, $|C| + 1$ são suficientes para vencer mantendo apenas em todas as rodadas cada vértice de C com um guarda e um guarda no mesmo vértice do espião. Vemos que os movimentos são possíveis, pois todos os vértices de G possuem um vizinho em C .

Agora assumamos que G é uma aranha gorda. Uma vez que o espião pode mover-se com uma jogada de um vértice de $C \cup R$ para um vértice qualquer de S , então cada vértice de S deve ter um guarda em sua vizinhança fechada antes de um movimento do espião. Veja que, em uma aranha gorda, apenas um vértice em C não possui todos os vértices de $N[S]$ como vizinho, mas a vizinhança fechada de dois vértices distintos de C é exatamente $V(G)$. Com isso, dois guardas em vértices distintos de C vigiam o grafo inteiro em até uma jogada (lembre-se que $d = 0$). Se $R \neq \emptyset$, então além dos dois guardas em C temos que ter mais um guarda para vigiar o espião quando ele mover-se para R , uma vez que não podemos tirar um dos 2 guardas em vértices distintos de C necessários para vigiar o espião em S (lembre-se novamente que $d = 0$). Com isso, 2 guardas são necessários se $R = \emptyset$ e 3 guardas são necessários se $R \neq \emptyset$. Além disso, se $R = \emptyset$, 2 guardas são suficientes para vencer: se o espião está em S , um guarda move-se para o mesmo vértice em que o espião está e o outro guarda move-se para o vértice em C que não é vizinho do vértice em que o espião está, uma vez que ele vigia todos os outros vértices de S ; se o espião está em C , ambos os guardas movem-se para vértices distintos de C . Finalmente, se $R \neq \emptyset$, 3 guardas são suficientes para vencer deixando apenas em todas as rodadas dois guardas em vértices distintos de C (qualquer vértice do grafo é vizinho de um deles) e um guarda no mesmo vértice do espião. Isso é possível uma vez que em dois vértices distintos de C é possível vigiar todos os vértices de G em uma única jogada. \square

O próximo lema determina o número de guarda em p -componentes separáveis.

Lema 17. *Seja G um grafo com uma p -componente separável H com $|V(H)| < q$ e bipartição (H_1, H_2) de H tal que cada vértice de $G - H$ é adjacente a cada vértice de H_1 e não é adjacente a cada vértice de H_2 . Então, para cada $s \geq 2$ e $d \geq 0$, $gn_{s,d}(G) = gn_{s,d}(G_R) \leq q$, onde G_R é o grafo "reduzido" obtido de G substituindo $G - H$ por dois vértices adjacentes ou não adjacentes, dependendo se $G - H$ é completo ou não, respectivamente.*

Demonstração. Sejam v_1 e v_2 vértices que substituíram $G - H$ para construção de G_R .

Para cada jogada em G , podemos simulá-la em G_R do seguinte modo:

- se o jogador (do espião ou do guarda) faz movimento para H quando jogado em G , ele vai para o mesmo vértice de H no jogo em G_R ;
- se um jogador faz movimentos para $G - H$ de um vértice que não está em $G - H$ no jogo em G , ele move-se para v_1 ou v_2 no jogo em G_R (dando preferência ao mesmo vértice do espião, se ele estiver em $\{v_1, v_2\}$ no jogo em G_R);
- se o jogador faz movimentos de um vértice de $G - H$ para um vizinho em $G - H$ no jogo em G , ele mantém sua posição no jogo em G_R ;
- se o espião move-se de um vértice de $G - H$ para outro vértice não vizinho também de $G - H$ jogando em G , ele move-se de v_1 para v_2 ou de v_2 para v_1 jogando G_R .

Por outro lado, para cada jogo em G_R , podemos simulá-lo em G , do modo a seguir. Para isso, se $G - H$ não é completo, sejam u_1 e u_2 dois vértices não adjacentes de $G - H$; caso contrário, sejam u_1 e u_2 vértices de $G - H$ (u_1 e u_2 são iguais se $G - H$ possui apenas um vértice).

- Para isso, se $G - H$ não é completo, sejam u_1 e u_2 dois vértices não adjacentes de $G - H$;
- se um jogador (do espião ou do guarda) move-se para H jogando em G_R , ele move-se para o mesmo vértice de H jogando em G .
- se o jogador faz o movimento para v_1 (respectivamente v_2) a partir de um vértice que não está em $\{v_1, v_2\}$ jogando em G_R , então ele move-se para u_1 (respectivamente u_2) jogando em G .
- se o jogador faz um movimento de v_1 para v_2 (respectivamente v_2 para v_1) jogando em G_R , ele move-se de u_1 para u_2 (respectivamente u_2 para u_1) jogando em G .

Com isso, se o espião vence em G , ele também vence em G_R . Além disso, se os guardas vencem em G , facilmente adaptamos a estratégia dos guardas descrita acima em G_R para garantir uma estratégia vencedora em G_R . Logo, existe uma estratégia vencedora em G para cada estratégia vencedora em G_R , e vice-versa. Portanto, $gn_{s,d}(G) = gn_{s,d}(G_R)$. Facilmente vemos que um guarda em cada vértice de H é suficiente para vigiar o espião em G_R , então $gn_{s,d}(G_R) \leq |V(H)| < q$. \square

Teorema 13. *O problema de decisão do Jogo do Espião está em FPT em grafos com poucos P_4 's quando $q(G)$ do grafo G é o parâmetro, com tempo $O(m + q^{3q+2} \cdot n)$, onde $q = q(G)$ e m e n são o número de arestas e vértices de G , respectivamente.*

Demonstração. Da decomposição primeval dos $(q, q-4)$ -grafos em tempo $O(m+n)$, dos Lemas 15, 16 e 17 e do algoritmo XP do Teorema 11 aplicado no grafo G_R do Lema 17 ou em G quando tem no máximo q vértices e no máximo q guardas. \square

4.4 O Jogo do Espião é $W[2]$ -difícil em grafos bipartidos

Nesta seção provamos que o problema de decisão do Jogo do Espião é $W[2]$ -difícil para qualquer velocidade $s \geq 2$ e distância $d \geq 0$, mesmo em grafos bipartidos, quando o parâmetro é o número de guardas. Para isso, mostramos uma redução FPT a partir do PROBLEMA DA COBERTURA DE CONJUNTOS, quando o parâmetro é o tamanho da solução. Estruturalmente, nossa redução FPT é semelhante a redução de (COHEN *et al.*, 2018) para grafos gerais. No entanto, pode-se notar que, em grafos bipartidos, temos uma maior dificuldade técnica e a redução fica significativamente diferente de (COHEN *et al.*, 2018).

Teorema 14. *Sejam $s \geq 2$ e $d \geq 0$ inteiros fixos. O problema de decisão do Jogo do Espião com k guardas é NP-difícil e $W[2]$ -difícil (quando o parâmetro é o número de guardas k), mesmo em grafos bipartidos. Além disso, o problema de minimização do número de guardas é Log-APX-difícil e $(1 - \varepsilon) \ln n$ -inaproximável em tempo polinomial para cada constante $0 < \varepsilon < 1$, a menos que $P = NP$, em grafos bipartidos.*

A prova do Teorema 14 segue dos seis lemas a seguir. A redução é a partir do PROBLEMA DA COBERTURA DE CONJUNTOS e é dividida em seis casos considerando a relação entre s e d , onde $r = r(s, d) = d \bmod (s - 1)$:

- **Caso 1:** $s > 2d + 2$
- **Caso 2:** $s = 2d + 2$
- **Caso 3:** $d + 1 < s < 2d + 2$
- **Caso 4:** $s \leq d + 1$ e $s < 2(r + 1)$
- **Caso 5:** $s \leq d + 1$ e $s = 2(r + 1)$
- **Caso 6:** $s \leq d + 1$ e $s > 2(r + 1)$

Uma instância do PROBLEMA DA COBERTURA DE CONJUNTOS é uma família $\mathcal{S} = \{S_1, \dots, S_m\}$ de conjuntos e um inteiro c , e o objetivo é decidir se existe uma subfamília $\mathcal{C} = \{S_{i_1}, \dots, S_{i_c}\} \subseteq \mathcal{S}$ tal que $|\mathcal{C}| \leq c$ e $S_{i_1} \cup \dots \cup S_{i_c} = U$, onde $U = S_1 \cup \dots \cup S_m$ (dizemos que \mathcal{C} é uma cobertura de conjunto de U). Dada uma instância (\mathcal{S}, c) do PROBLEMA DA COBERTURA DE CONJUNTOS, construímos um grafo $G = G_{s,d}(\mathcal{S}, c)$ e um inteiro $K = K_{s,d}(\mathcal{S}, c)$ tal que existe uma cobertura $\mathcal{C} \subseteq \mathcal{S}$ de U com tamanho no máximo c se e somente se $gn_{s,d}(G) \leq K$. Note que as reduções apresentadas a seguir são de fato reduções FPT e preservam a razão de aproximação. Sendo assim, uma vez que o PROBLEMA DA COBERTURA DE CONJUNTOS é $W[2]$ -difícil (quando o parâmetro é o tamanho c da cobertura) e não possui algoritmo $(1 - \varepsilon) \ln(n)$

aproximativo para alguma constante $0 < \varepsilon < 1$ (a menos que $P = NP$) (ALON *et al.*, 2006; MOSHKOVITZ, 2015), não provamos apenas a NP-dificuldade, mas também o fato do problema ser $W[2]$ -difícil (quando o parâmetro é o número de guardas) e não poder ser aproximado por uma razão logarítmica (a menos que $P = NP$).

A seguir, apresentamos a construção do grafo $G = G_{s,d}(\mathcal{S}, c)$.

Definição 3. Dados $s \geq 2$ e $d \geq 0$, sejam $p = p(s, d) = d + \lceil \frac{d+1}{s-1} \rceil$ e

$$q = q(s, d) = \begin{cases} d+1 & , \text{ se caso 1} \\ d & , \text{ se caso 2} \\ 0 & , \text{ se caso 3} \\ 0 & , \text{ se caso 4} \\ p-1 & , \text{ se caso 5} \\ p & , \text{ se caso 6} \end{cases}$$

Sejam (\mathcal{S}, c) uma instância do PROBLEMA DA COBERTURA DE CONJUNTOS, onde $\mathcal{S} = \{S_1, \dots, S_m\}$ e $U = S_1 \cup \dots \cup S_m = \{u_1, \dots, u_n\}$. Seja o número de guardas $K = K_{s,d}(\mathcal{S}, c)$:

$$K = K_{s,d}(\mathcal{S}, c) = \begin{cases} c+2 & , \text{ se caso 1} \\ c+2 & , \text{ se caso 2} \\ c & , \text{ se caso 3} \\ c & , \text{ se caso 4} \\ c+2 & , \text{ se caso 5} \\ c+2 & , \text{ se caso 6} \end{cases}$$

Seja $G = G_{s,d}(\mathcal{S}, c)$ o grafo definido da seguinte forma:

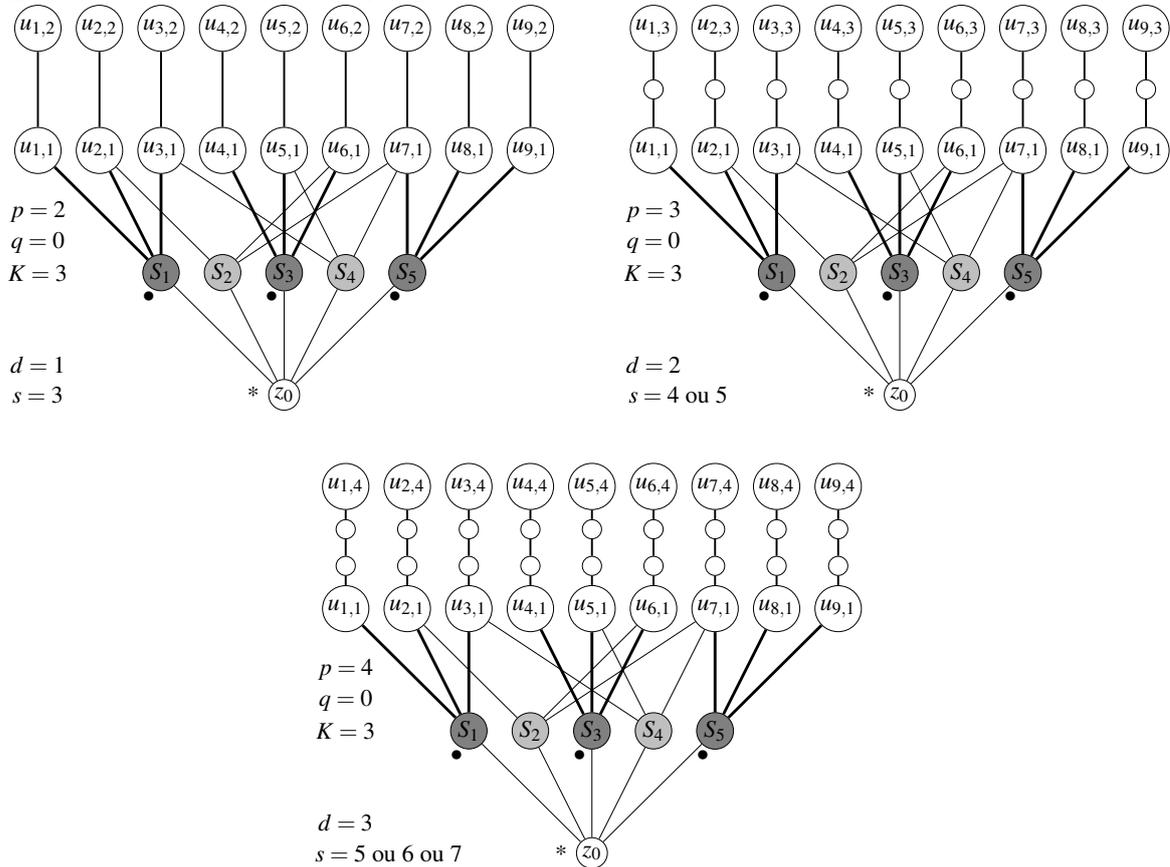
- para cada conjunto $S_j \in \mathcal{S}$, crie um novo vértice S_j em G ;
- para cada elemento $u_i \in U$, crie um caminho U_i com p vértices $u_{i,1}, \dots, u_{i,p}$;
- se $u_i \in S_j$, adicione a aresta $u_{i,1}S_j$ em G ;
- crie um vértice z_0 e adicione todas as arestas possíveis entre z_0 e $\{S_1, \dots, S_m\}$ em G ;
- se não estamos nos casos 3 e 4, crie dois caminhos $Z = (z_1, \dots, z_q)$ e $Z' = (z'_1, \dots, z'_{q'})$ e adicione as arestas z_0z_1 e $z_0z'_1$, onde $q' = p$ se estamos no Caso 1 ou 6 e, em caso contrário, $q' = p + 1$.

As Figuras 33-38 são exemplos para cada caso da redução. Nas figuras, a posição principal do espião é indicada por um asterisco (*) e a posição principal de cada guarda é indicada com um ponto (•). Note que $G = G_{s,d}(\mathcal{S}, c)$ é um grafo bipartido.

A seguir apresentamos um resultado de (COHEN *et al.*, 2018), que usamos nas demonstrações dos lemas apresentados em seguida. Como apontado em (COHEN *et al.*, 2018), a volta da Proposição 2 não é necessariamente verdadeira.

Proposição 2. (COHEN *et al.*, 2018) *Se k guardas vencem (vigilando o espião depois de um número finito de jogadas) na variante do Jogo do Espião em que o espião é colocado depois dos k guardas, então k guardas também vencem no Jogo do Espião normal.*

Figura 33 – Exemplo de redução para o Lema 18



Fonte: elaborado pelo autor.

Nota: Caso 3 ($d+1 < s < 2d+2$). Exemplo de redução da instância (\mathcal{S}, c) do PROBLEMA DA COBERTURA DE CONJUNTOS, onde $c=3$, $S_1 = \{1, 2, 3\}$, $S_2 = \{2, 6, 7\}$, $S_3 = \{4, 5, 6\}$, $S_4 = \{3, 5, 7\}$, $S_5 = \{7, 8, 9\}$ e $U = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$.

Começaremos com o caso mais fácil.

Lema 18. *Dado um grafo bipartido G e um inteiro $K > 0$, decidir se $g_{s,d}(G) \leq K$ é NP-difícil para cada $s \geq 2$ e $d \geq 1$ satisfazendo o Caso 3 ($d+1 < s < 2d+2$).*

Demonstração. Reduzimos a partir do PROBLEMA DA COBERTURA DE CONJUNTOS. Seja (\mathcal{S}, c) uma instância para o PROBLEMA DA COBERTURA DE CONJUNTOS. Lembre-se da Definição 3 e sejam $p = p(s, d) = d + 1$, $q = q(s, d) = 0$, $G = G_{s,d}(\mathcal{S}, c)$ e $K = K_{s,d}(\mathcal{S}, c) = c$.

Primeiramente, suponha que não existe uma cobertura \mathcal{C} de U com no máximo c conjuntos em \mathcal{S} . Provaremos agora que o espião vence o jogo com no máximo $K = c$ guardas, começando em z_0 . De fato, uma vez que existem no máximo K guardas e não tem uma cobertura de U com c conjuntos em \mathcal{S} , então existe algum $1 \leq i \leq n$ tal que não existe guardas em $N[U_i]$. Portanto, o espião caminha para $u_{i,p}$ em uma jogada (note que a distância de z_0 para $u_{i,p}$ é $p + 1 = d + 2 \leq s$). Durante as jogadas dos guardas, nenhum guarda pode chegar em um vértice de U_i , e, desta forma, o espião continua a uma distância de pelo menos d de todos os guardas. Nesta configuração o espião vence.

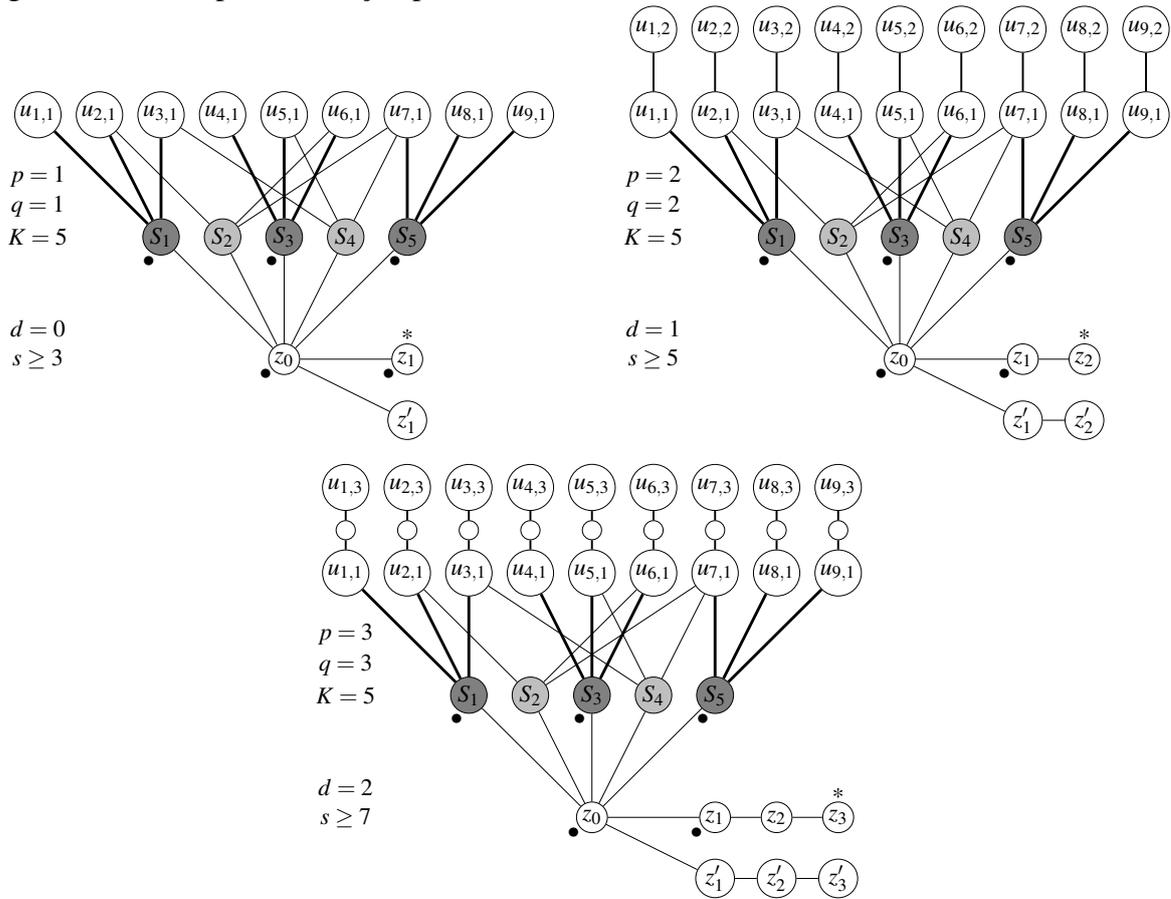
Agora, suponha que existe uma cobertura $\mathcal{C} = \{S_{j_1}, \dots, S_{j_c}\}$ de U com c conjuntos em \mathcal{S} . Para facilitar a demonstração, mostramos que $c = K$ guardas vencem se eles são colocados primeiro. Pela Proposição 2, é suficiente provar que $g_{s,d}(G) \leq K$. A estratégia dos guardas é ocupar inicialmente os vértices S_{j_1}, \dots, S_{j_c} . Uma vez que \mathcal{C} é uma cobertura de U , podemos definir para qualquer elemento $u_i \in U$ um índice $c(i)$ tal que $u_i \in S_{c(i)} \in \mathcal{C}$. Se o espião não está em $\{u_{1,p}, \dots, u_{n,p}\}$, então os guardas ocupam os vértices iniciais e então eles controlam o espião. Se o espião está em um vértice $u_{i,p}$, então o guarda ocupando $S_{c(i)}$ vai para $u_{i,1}$ e controla o espião. Além disso, uma vez que $s < 2d + 2$, o espião não consegue ir em uma jogada de $u_{i,p}$ para outro vértice $u_{j,p}$ com $j \neq i$. Portanto, se o espião sair de $u_{i,p}$, os guardas reocupam os vértices iniciais. Com essa estratégia os guardas vencem o jogo. \square

Para facilitar a compreensão do texto considere as seguintes notações para as jogadas do Jogo do Espião:

- $(u \xrightarrow{e} v)$ o espião move-se do vértice u para o vértice v ;
- $(X \xrightarrow{e} Y)$ o espião move-se de um vértice do conjunto X para um vértice do conjunto Y ;
- $(\bullet \xrightarrow{e} v)$ realiza um movimento possível de um vértice do grafo para o vértice v ;
- $(v_1 \xrightarrow{g} v_2 ; v_3 \xrightarrow{g} v_4 ; \dots ; v_{2k-1} \xrightarrow{g} v_{2k})$ o guarda em v_1 move-se para v_2 , o guarda em v_3 move-se para v_4 , ... , o guarda em v_{2k-1} move-se para v_{2k} . Neste caso, escrevemos $v_i \xrightarrow{g} v_{i+1}$ apenas quando $v_i \neq v_{i+1}$, em caso contrário, o guarda em v_i permanece parado no jogo e não é necessário denotar esse movimento.

Lema 19. *Dado um grafo bipartido G e um inteiro $K > 0$, decidir se $g_{s,d}(G) \leq K$ é NP-difícil para cada $s \geq 2$ e $d \geq 0$ satisfazendo o Caso 1 ($s \geq 2d + 3$).*

Figura 34 – Exemplo de redução para o Lema 19



Fonte: elaborado pelo autor.

Nota: Caso 1 ($s \geq 2d + 3$). Redução para a Figura 33 a partir de uma instância do PROBLEMA DA COBERTURA DE CONJUNTOS.

Demonstração. Reduzimos do PROBLEMA DA COBERTURA DE CONJUNTOS. Seja (\mathcal{S}, c) uma instância do Problema da Cobertura de Conjuntos. Lembre-se da Definição 3 e sejam $p = p(s, d) = d + 1$, $q = q(s, d) = d + 1$, $G = G_{s,d}(\mathcal{S}, c)$ e $K = K_{s,d}(\mathcal{S}, c) = c + 2$.

Primeiro, suponha que não existe uma cobertura \mathcal{C} de U com no máximo c conjuntos em \mathcal{S} . Mostramos agora que o espião vence no máximo $K = c + 2$ guardas. Precisamente, o espião começa em z_q e vence em uma jogada. De fato, se inicialmente nenhum guarda ocupa um vértice em $\{z_1, \dots, z_q\}$, então o espião vence imediatamente. Deste modo, assumimos que existe pelo menos um guarda em $\{z_1, \dots, z_q\}$. Novamente, se não existe nenhum guarda em $\{z_0, z'_1, \dots, z'_q\}$, o espião pode ir para z'_q e vencer o jogo. Uma vez que temos $c + 2$ guardas, então existem no máximo c guardas fora de $\{z_0, z_1, z'_1, \dots, z_q, z'_q\}$. Desde de que não existe uma cobertura de U com c conjuntos em \mathcal{S} , então existe algum $1 \leq i \leq n$ tal que nenhum guarda está em $N[U_i]$. Portanto, o espião caminha para $u_{i,p}$ em uma jogada (note que a distância de z_q para $u_{i,p}$ é $p + q + 1 = 2d + 3 \leq s$) e vence, pois nenhum guarda pode alcançar um vértice em U_i (isto é, nenhum vértice a uma distância de no máximo d de $u_{i,p}$) na próxima jogada.

Agora, suponha que existe uma cobertura $\mathcal{C} = \{S_{j_1}, \dots, S_{j_c}\}$ de U com c conjuntos em \mathcal{S} . Para facilitar a demonstração, provamos que $c + 2 = K$ guardas vencem se eles são colocados primeiro. Pela Proposição 2, é suficiente provar que $g_{s,d}(G) \leq K$. A estratégia dos guardas é a seguinte. Ocupar inicialmente os vértices $z_0, z_0, S_{j_1}, \dots, S_{j_c}$ (dois guardas em z_0). Uma vez que \mathcal{C} é uma cobertura de U , definimos para cada elemento de $u_i \in U$ um índice $c(i)$ tal que $u_i \in S_{c(i)} \in \mathcal{C}$. Com a estratégia a seguir vemos que os guardas vencem o jogo.

- Se o espião ocupar um vértice fora de $\{z_q, z'_q, u_{1,p}, \dots, u_{n,p}\}$, então os guardas mantêm suas posições iniciais e controlam o espião.
- Se $(\bullet \xrightarrow{e} z_q)$, então $(z_0 \xrightarrow{g} z_1)$ (e se existir um guarda em z'_1 , então $(z'_1 \xrightarrow{g} z_0)$), controlando o espião;
- Se $(\bullet \xrightarrow{e} z'_q)$, então $(z_0 \xrightarrow{g} z'_1)$ (e se existir um guarda em z_1 , então $(z_1 \xrightarrow{g} z_0)$), controlando o espião;
- Se $(\bullet \xrightarrow{e} u_{i,p})$, então $(S_{c(i)} \xrightarrow{g} u_{i,1} ; z_0 \xrightarrow{g} S_{c(i)})$, controlando o espião com o guarda de $u_{i,1}$;
- Se $(u_{i,p} \xrightarrow{e} u_{j,p})$ com $c(i) = c(j)$, então $(S_{c(i)} \xrightarrow{g} u_{j,1} ; u_{i,1} \xrightarrow{g} S_{c(i)})$, controlando o espião com o guarda de $u_{j,1}$;
- Se $(u_{i,p} \xrightarrow{e} u_{j,p})$ com $c(i) \neq c(j)$, então $(S_{c(j)} \xrightarrow{g} u_{j,1} ; z_0 \xrightarrow{g} S_{c(j)} ; S_{c(i)} \xrightarrow{g} z_0 ; u_{i,1} \xrightarrow{g} S_{c(i)})$, controlando o espião com o guarda de $u_{j,1}$ e
- Se o espião deixar $u_{i,p}$ para algum vértice fora de $\{z_q, z'_q, u_{1,p}, \dots, u_{n,p}\}$, então os guardas reocupam os vértices iniciais $(S_{c(i)} \xrightarrow{g} z_0 ; u_{i,1} \xrightarrow{g} S_{c(i)})$.

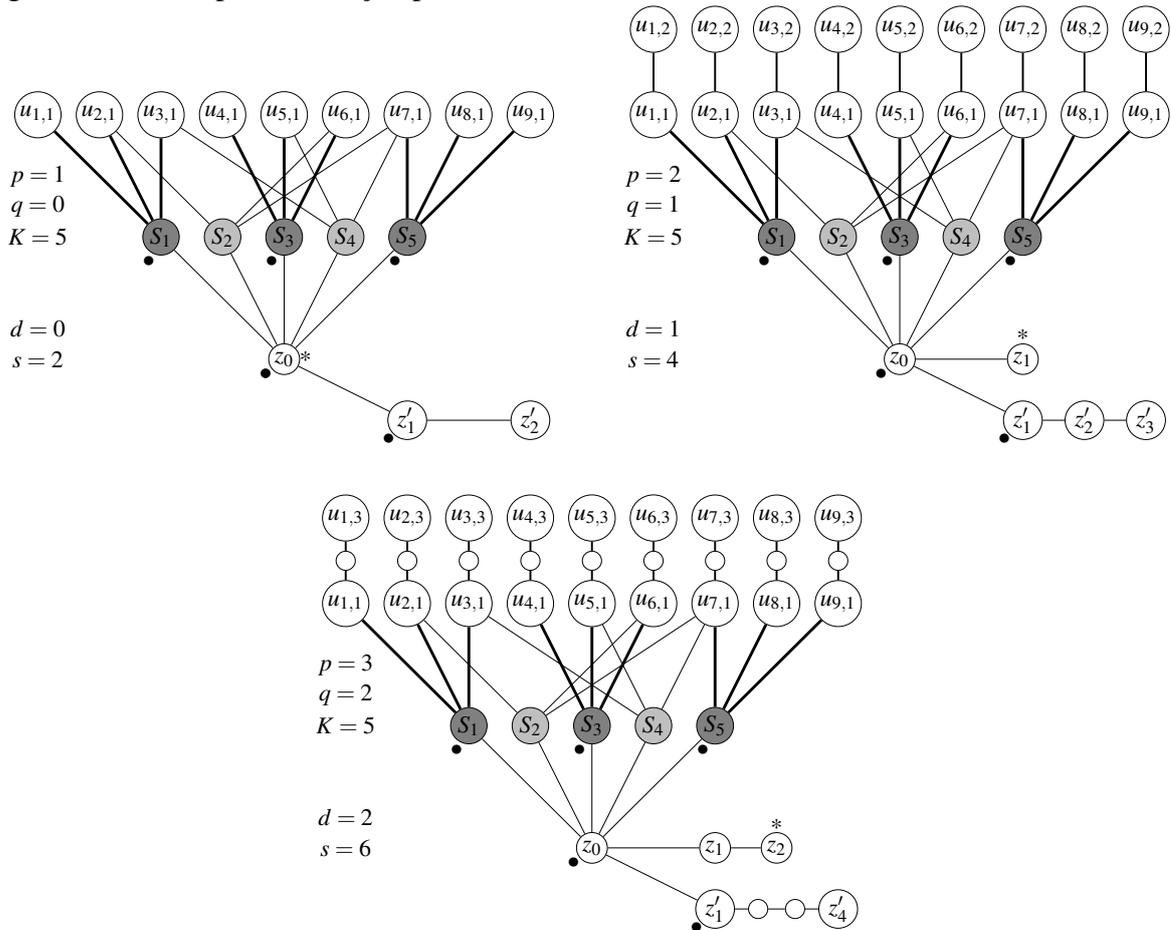
□

Lema 20. *Dado um grafo bipartido G e um inteiro $K > 0$, decidir se $g_{s,d}(G) \leq K$ é NP-difícil para cada $s \geq 2$ e $d \geq 0$ satisfazendo o Caso 2 ($s = 2d + 2$).*

Demonstração. Reduzimos a partir do PROBLEMA DA COBERTURA DE CONJUNTOS. Seja (\mathcal{S}, c) uma instância para o PROBLEMA DA COBERTURA DE CONJUNTOS. Lembre-se da Definição 3 e sejam $p = p(s, d) = d + 1$, $q = q(s, d) = d$, $q' = d + 2$, $G = G_{s,d}(\mathcal{S}, c)$ e $K = K_{s,d}(\mathcal{S}, c) = c + 2$.

Primeiro, suponha que não existe uma cobertura \mathcal{C} de U com no máximo c conjuntos em \mathcal{S} . Mostraremos que o espião vence em uma jogada contra no máximo $c + 2$ guardas, iniciando em z_d . Assuma, sem perda de generalidade, que existe um guarda em Z_0 (caso contrário o espião vence imediatamente). Além disso, uma vez que o espião pode ir em uma jogada para z'_{d+2} (lembre-se que $s = 2d + 2 = p + q$), assumindo, sem perda de generalidade,

Figura 35 – Exemplo de redução para o Lema 20



Fonte: elaborado pelo autor.

Nota: Caso 2 ($s = 2d + 2$). Redução para a Figura 33 a partir de uma instância do PROBLEMA DA COBERTURA DE CONJUNTOS.

que existe outro guarda em z'_1 (caso contrário o espião vence na próxima jogada indo para z'_{d+2}). Como existam $c + 2$ guardas, então existem no máximo c guardas fora de $\{z_0, z_1, z'_1, \dots\}$. Uma vez que não existe uma cobertura de U com C conjuntos em \mathcal{S} , então existe algum $1 \leq i \leq n$ tal que não existe guarda em $N[U_i]$. Portanto, o espião vai de z_d para $u_{i,p}$ em uma jogada (note que a distância de z_d para $u_{i,p}$ é $p + q + 1 = 2d + 2 = s$) e vence, uma vez que nenhum guarda consegue chegar em um vértice de U_i para vigiar o espião, mesmo com uma jogada.

Agora, suponha que existe uma cobertura $\mathcal{C} = \{S_{j_1}, \dots, S_{j_c}\}$ de U com c conjuntos em \mathcal{S} . Usaremos a Proposição 2 para provar que $c + 2$ guardas vencem se eles são colocados primeiro com a seguinte estratégia. Ocupando inicialmente os vértices $z_0, z'_1, S_{j_1}, \dots, S_{j_c}$. Como \mathcal{C} é uma cobertura de U , definimos para cada elemento de $u_i \in U$ um índice $c(i)$ tal que $u_i \in S_{c(i)} \in \mathcal{C}$.

- Se o espião ocupar um vértice fora de $\{u_{1,p}, \dots, u_{n,p}\}$, então os guardas permanecem na posição inicial, controlando o espião;

- Se $(\bullet \xrightarrow{e} u_{i,p})$, então $(z_0 \xrightarrow{g} S_{c(i)} ; z'_1 \xrightarrow{g} z_0)$;
- Se $(u_{i,p} \xrightarrow{e} u_{j,p})$ com $i \neq j$ e $c(i) = c(j)$, então $(S_{c(i)} \xrightarrow{g} u_{j,1} ; u_{i,1} \xrightarrow{g} S_{c(i)})$ com o guarda de $u_{j,1}$ controlando o espião;
- Se $(u_{i,p} \xrightarrow{e} u_{j,p})$ com $c(i) \neq c(j)$, então $(S_{c(j)} \xrightarrow{g} u_{j,1} ; z_0 \xrightarrow{g} S_{c(j)} ; S_{c(i)} \xrightarrow{g} z_0 ; u_{i,1} \xrightarrow{g} S_{c(i)})$ com o guarda de $u_{j,1}$ controlando o espião; e
- Se o espião deixar $u_{j,p}$ para um vértice fora de $\{u_{1,p}, \dots, u_{n,p}\}$, então os guardas reocupam os vértices iniciais $(z_0 \xrightarrow{g} z'_1 ; S_{c(j)} \xrightarrow{g} z_0 ; u_{j,1} \xrightarrow{g} S_{c(j)})$.

Como em todos os casos possíveis temos que os guardas conseguem vigiar o espião, então temos uma estratégia vencedora para os guardas (note que a jogada do espião de $(u_{i,p} \xrightarrow{e} z'_q)$ é impossível). \square

Agora considere o caso em que $2 \leq s \leq d + 1$. Usaremos a proposição abaixo de (COHEN *et al.*, 2018) para as próximas demonstrações.

Proposição 3. (COHEN *et al.*, 2018) *Sejam $s \geq 2$ e $d \geq 1$ satisfazendo $s \leq d + 1$. Sejam $p = d + \lceil \frac{d+1}{s-1} \rceil$ e $r = d \bmod (s-1)$. Seja $P = (u_{-1}, u_0, u_1, \dots, u_p)$ um grafo que é um caminho e considere um guarda jogando em P contra o espião, o qual é o primeiro a mover-se, com velocidade s e distância de vigilância d .*

(a) *se o guarda iniciar em u_{-1} , ele sempre perde;*

(b) *se o guarda iniciar em u_0 , o espião vence se e somente se iniciar em u_i com $i \geq r + 1$.*

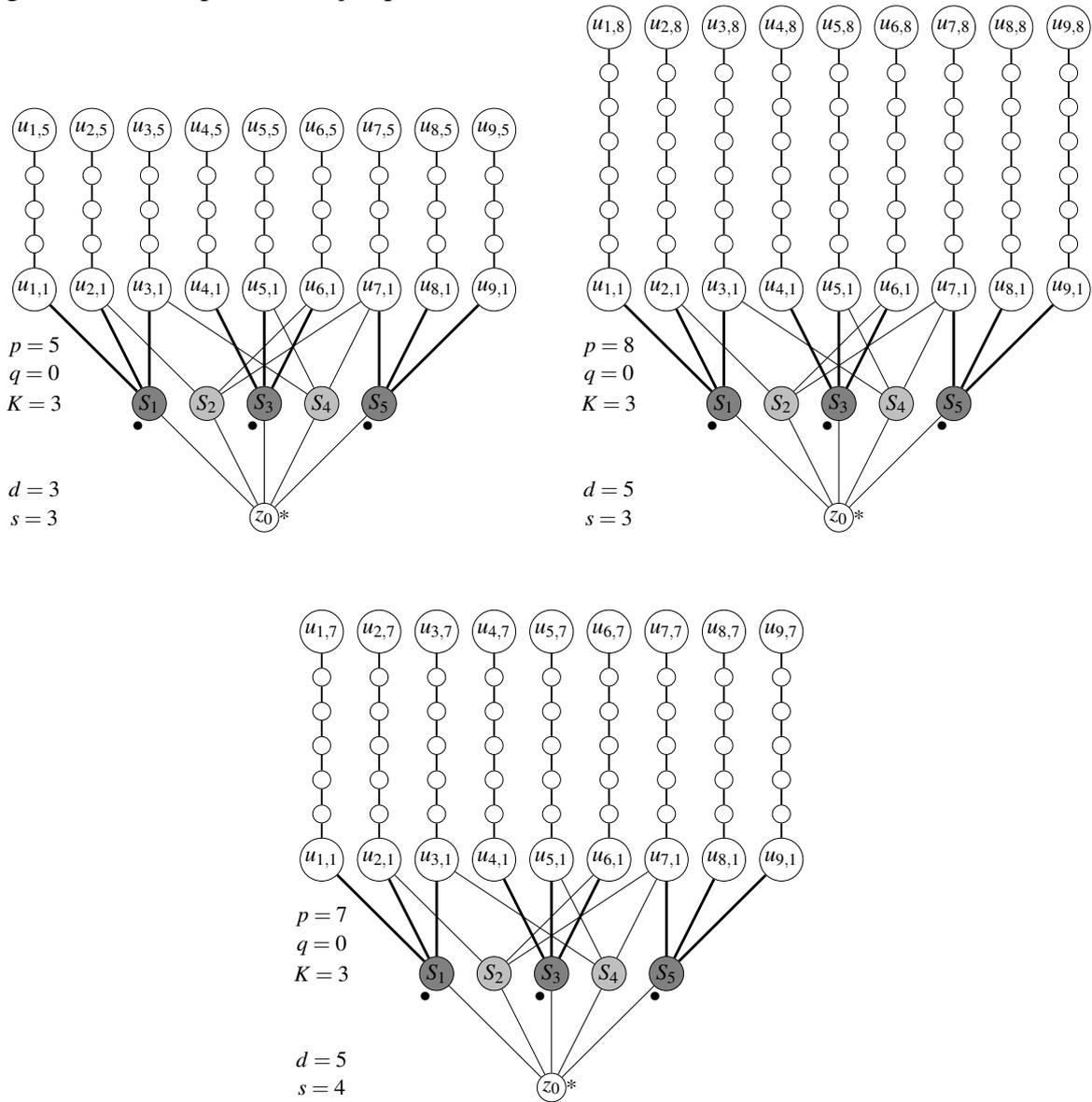
Nos próximos lemas, usamos a notação $U_{i,>j} = \{u_{i,t} \in U_i \mid t > j\}$ e $U_{i,\leq j} = \{u_{i,t} \in U_i \mid t \leq j\}$, dados os inteiros positivos i e j . Analogamente temos as mesmas definições para $Z_{\leq j}, Z_{>j}, Z'_{\leq j}$ e $Z'_{>j}$.

Lema 21. *Dados um grafo bipartido G e um inteiro $K > 0$, decidir se $g_{s,d}(G) \leq K$ é NP-difícil para $s \geq 2$ e $d \geq 1$ satisfazendo o Caso 4 ($s \leq d + 1$ e $s < 2r + 2$, onde $r = d \bmod (s-1)$).*

Demonstração. Reduzimos a partir do PROBLEMA DA COBERTURA DE CONJUNTOS. Seja (\mathcal{S}, c) uma instância para o PROBLEMA DA COBERTURA DE CONJUNTOS. Lembre-se da Definição 3 e sejam $p = p(s, d) = d + \lceil \frac{d+1}{s-1} \rceil$, $q = q(s, d) = 0$, $G = G_{s,d}(\mathcal{S}, c)$ e $K = K_{s,d}(\mathcal{S}, c) = c$.

Primeiro, suponha que não existe uma cobertura \mathcal{C} de U com no máximo c conjuntos de \mathcal{S} . Mostraremos que o espião vence o jogo contra o máximo de c guardas, iniciando o jogo em z_0 . De fato, uma vez que existem no máximo c guardas e não existe uma cobertura de U com c conjuntos em \mathcal{S} , então existe um $1 \leq i \leq n$ tal que não existe um guarda em $N[U_i]$.

Figura 36 – Exemplo de redução para o Lema 21



Fonte: elaborado pelo autor.

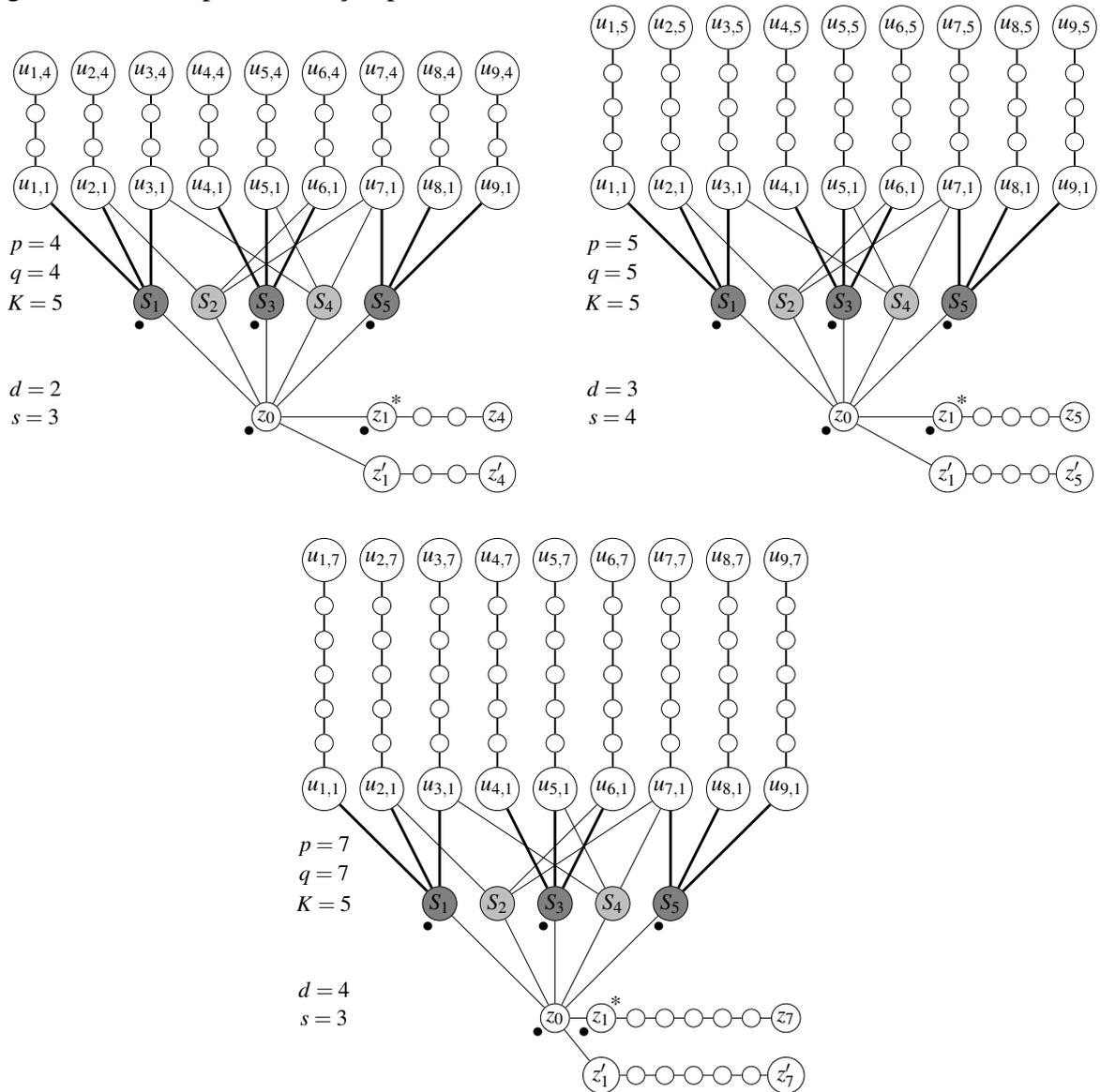
Nota: Caso 4 ($s \leq d + 1$ e $s < 2r + 2$). Redução para a Figura 33 a partir de uma instância do PROBLEMA DA COBERTURA DE CONJUNTOS.

Portanto, pela Proposição 3(a), o espião pode ir para $u_{i,p}$, vencendo o jogo no caminho induzido $z_0 - \dots - u_{i,p}$.

Agora, suponha que existe uma cobertura $\mathcal{C} = \{S_{j_1}, \dots, S_{j_c}\}$ de U com c conjuntos em \mathcal{S} . Usaremos a Proposição 2 para mostrar que c guardas vencem se eles são colocados primeiro. A estratégia dos guardas é a que se segue. Ocupar inicialmente os vértices S_{j_1}, \dots, S_{j_c} . Uma vez que \mathcal{C} é uma cobertura de U , podemos definir para cada elemento de $u_i \in U$ um índice $c(i)$ tal que $u_i \in S_{c(i)} \in \mathcal{C}$. Se o espião está em um vértice de $U_{i, \leq r}$ para algum i , então os guardas ocupam os vértices iniciais, controlando o espião pela Proposição 3(b). Se o espião mover-se para um vértice de $U_{i, > r}$ de um vértice fora de $U_{i, > r}$, então o guarda em $S_{c(i)}$ segue

para $u_{i,1}$, controlando o espião no caminho $u_{i,1} - \dots - u_{i,p}$, pela Proposição 3(b). Além disso, como $s < 2(r + 1)$, o espião não pode em uma jogada ir de um vértice de $U_{i,>r}$ para um vértice de $U_{j,>r}$ com $i \neq j$. Então, se o espião deixar $U_{i,>r}$, os guardas recupam a posição dos vértices iniciais: o guarda em $u_{i,1}$ segue para $S_{c(i)}$. Com essa estratégia os guardas vencem o jogo. \square

Figura 37 – Exemplo de redução para o Lema 22



Fonte: elaborado pelo autor.

Nota: Caso 6 ($s \leq d + 1$ e $s > 2r + 2$). Redução para a Figura 33 a partir de uma instância do PROBLEMA DA COBERTURA DE CONJUNTOS.

Lema 22. Dados um grafo bipartido G e um inteiro $K > 0$, decidir se $g_{s,d}(G) \leq K$ é NP-difícil para cada $s \geq 2$ e $d \geq 1$ satisfazendo o Caso 6 ($s \leq d + 1$ e $s > 2r + 2$, onde $r = d \pmod{s - 1}$).

Demonstração. Reduzimos a partir do PROBLEMA DA COBERTURA DE CONJUNTOS. Seja

(\mathcal{S}, c) uma instância para o PROBLEMA DA COBERTURA DE CONJUNTOS. Lembre-se da Definição 3 e sejam $p = p(s, d) = d + \lceil \frac{d+1}{s-1} \rceil$, $q = q(s, d) = p$, $q' = p$, $G = G_{s,d}(\mathcal{S}, c)$ e $K = K_{s,d}(\mathcal{S}, c) = c + 2$.

Primeiro, suponha que não existe uma cobertura \mathcal{C} de U com no máximo c conjuntos em \mathcal{S} . Mostremos agora que o espião vence o jogo contra no máximo $c + 2$ guardas, iniciando em z_{r+1} . Pela Proposição 3(a) no caminho $z_0 - z_q$ para os guardas não perderem deve existir um guarda em z_1 , supondo sem perda de generalidade. Além disso, uma vez que o espião pode ir para z'_{r+1} em uma jogada, podemos assumir que, pela Proposição 3(a), no caminho $z_0 - z'_q$ deve existir um guarda em z_0 , supondo sem perda de generalidade, para os guardas não perderem o jogo. Uma vez que existem $c + 2$ guardas, então existe no máximo c guardas fora de $\{z_0, z_1, z'_1, \dots, z_q, z'_q\}$. Como não existe uma cobertura de U com c conjuntos em \mathcal{S} , então existe algum $1 \leq i \leq n$ tal que não existe um guarda em $N[U_i]$. Portanto, com uma jogada, o espião pode ir para $u_{i,r+1}$ (lembramos que $s \geq 2r + 3$), vencendo o jogo no caminho induzido $z_0 - \dots - u_{i,p}$ pela Proposição 3(b).

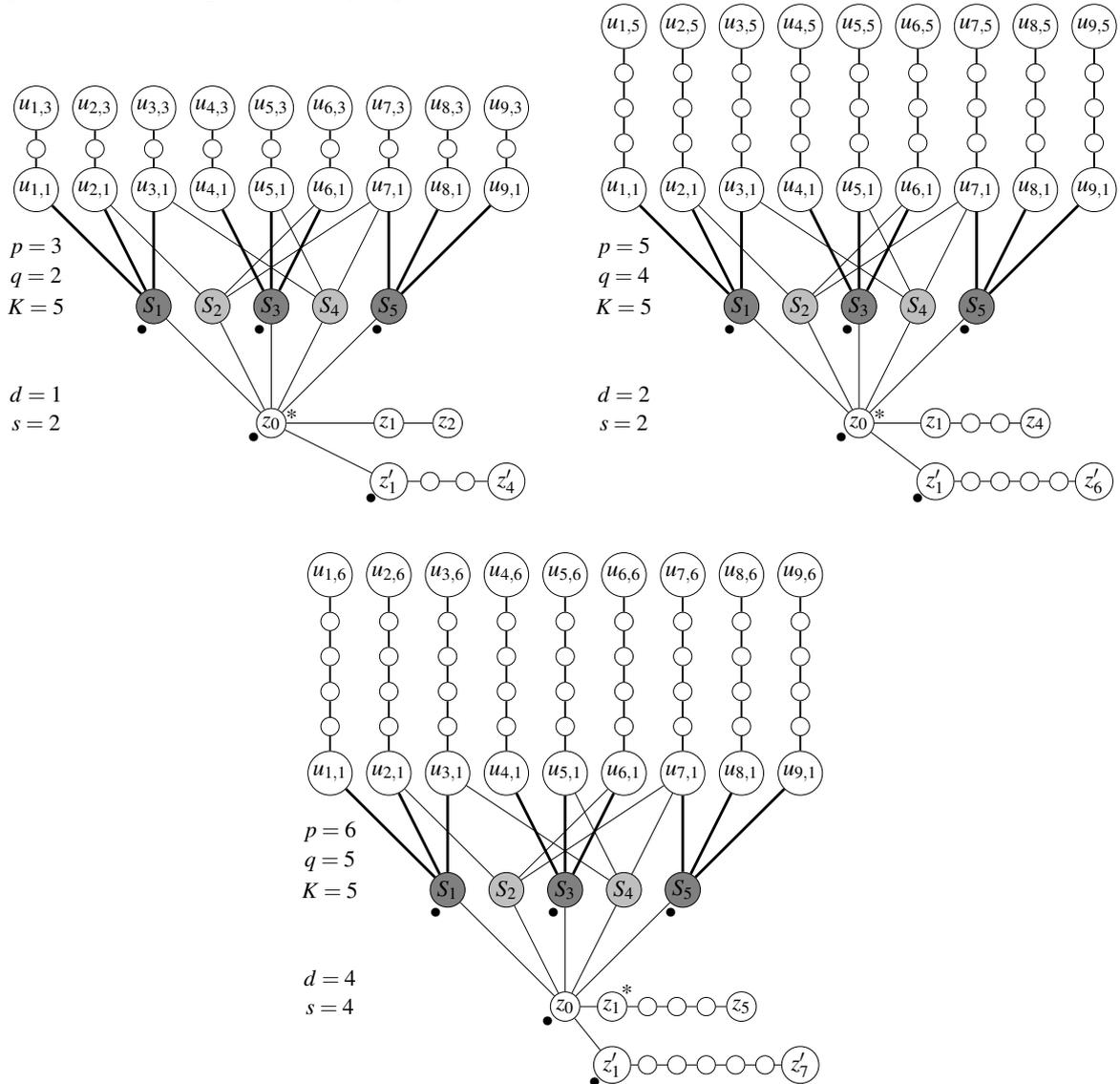
Agora, suponha que existe uma cobertura $\mathcal{C} = \{S_{j_1}, \dots, S_{j_c}\}$ de U com c conjuntos \mathcal{S} . Usaremos a Proposição 2 para mostrar que $c + 2$ guardas vencem se eles são colocados primeiro. Segue a estratégia vencedora dos guardas. Ocupar inicialmente os vértices $z_0, z_0, S_{j_1}, \dots, S_{j_c}$ (dois vértices em z_0) que chamaremos de vértices iniciais. Uma vez que \mathcal{C} é uma cobertura de U , definimos para cada elemento $u_i \in U$ um índice $c(i)$ tal que $u_i \in S_{c(i)} \in \mathcal{C}$.

- se o espião não estiver em $U_{i,>r}$, $Z_{>r}$ ou $Z'_{>r}$, então os guardas ocupam os vértices iniciais, controlando o espião pela Proposição 3(b).
- se $(\overline{U_{i,>r}} \xrightarrow{e} U_{i,>r})$ onde $\overline{U_{i,>r}}$ são vértices fora de $U_{i,>r}$, então $(S_{c(i)} \xrightarrow{g} u_{i,1} ; z_0 \xrightarrow{g} S_{c(i)})$, onde o guarda de $u_{i,1}$ controla o espião no caminho $u_{i,1} - \dots - u_{i,p}$ pela Proposição 3(b);
- se $(U_{i,>r} \xrightarrow{e} U_{j,>r})$, então $(S_{c(i)} \xrightarrow{g} u_{j,1})$ com $c(i) = c(j)$ e em seguida $(S_{c(i)} \xrightarrow{g} u_{j,1} ; u_{i,1} \xrightarrow{g} S_{c(i)})$ com o guarda de $u_{j,1}$ controlando o espião;
- se $(U_{i,>r} \xrightarrow{e} U_{j,>r})$ com $c(i) \neq c(j)$, então $(S_{c(j)} \xrightarrow{g} u_{j,1} ; z_0 \xrightarrow{g} S_{c(j)} ; S_{c(i)} \xrightarrow{g} z_0 ; u_{i,1} \xrightarrow{g} S_{c(i)})$ com o guarda de $u_{j,1}$ controlando o espião;
- se $(\overline{Z_{>r}} \xrightarrow{e} Z_{>r})$ onde $\overline{Z_{>r}}$ são vértices fora de $Z_{>r}$, então $(S_{c(j)} \xrightarrow{g} u_{j,1} ; z_0 \xrightarrow{g} S_{c(j)} ; S_{c(i)} \xrightarrow{g} z_0 ; u_{i,1} \xrightarrow{g} S_{c(i)})$ com o guarda de z_1 controlando o espião;
- análogo ao item anterior para Z' .
- se $(U_{i,>r} \xrightarrow{e} U_{j,\leq r})$ ou $(U_{i,>r} \xrightarrow{e} Z_{<r})$ para algum j , então os guardas ocupam os vértices iniciais $(u_{i,1} \xrightarrow{g} S_{c(i)} ; S_{c(i)} \xrightarrow{g} z_0)$ (temos dois guardas em z_0), controlando o espião pela

Proposição 3(a).

Facilmente vemos que, com essa estratégia, os guardas vencem, uma vez que em cada caso possível temos que o espião é vigiado. \square

Figura 38 – Exemplo de redução para o Lema 23



Fonte: elaborado pelo autor.

Nota: Caso 5 ($s \leq d + 1$ e $s = 2r + 2$). Redução para a Figura 33 a partir de uma instância do PROBLEMA DA COBERTURA DE CONJUNTOS.

Lema 23. Dados um grafo bipartido G e um inteiro $K > 0$, decidir se $g_{s,d}(G) \leq K$ é NP-difícil para cada $s \geq 2$ e $d \geq 1$ satisfazendo o Caso 5 ($s \leq d + 1$ e $s = 2r + 2$, onde $r = d \pmod{s - 1}$).

Demonstração. Reduzimos a partir do PROBLEMA DA COBERTURA DE CONJUNTOS. Seja (\mathcal{S}, c) uma instância para o PROBLEMA DA COBERTURA DE CONJUNTOS. Lembre-se da

Definição 3 e sejam $p = p(s, d) = d + \lceil \frac{d+1}{s-1} \rceil$, $q = q(s, d) = p - 1$, $q' = p + 1$, $G = G_{s,d}(\mathcal{S}, c)$ e $K = K_{s,d}(\mathcal{S}, c) = c + 2$.

Primeiro, suponha que não existe uma cobertura \mathcal{C} de U com no máximo c conjuntos em \mathcal{S} . Mostraremos que o espião vence em uma jogada contra no máximo $c + 2$ guardas, iniciando em z_r . De fato, pela Proposição 3(b) no caminho $z'_2 - z'_1 - z_0 - z_1 - \dots - z_{p-1}$, deve existir um guarda em z_0 , supondo sem perda de generalidade, pois, caso contrário, os guardas perdem. Além disso, uma vez que o espião consegue ir em uma jogada para z'_{r+2} (lembramos que $s = 2r + 2$), supondo sem perda de generalidade que existe outro guarda em z'_1 (caso contrário o espião vence, pela Proposição 3(b) no caminho $z_0 - z'_1 - \dots - z_{p+1}$). Uma vez que existem $c + 2$ guardas, então existem no máximo c guardas fora de $\{z_0, z_1, z'_1, \dots\}$. Como não temos uma cobertura de U com c conjuntos em \mathcal{S} , então existe algum $1 \leq i \leq n$ de modo que não temos um guarda em $N[U_i]$. Portanto, o espião move-se de z_r para $u_{i,r+1}$ em uma jogada (note que a distância é $r + (r + 1) + 1 = 2r + 2 = s$) e vence pela Proposição 3(b) no caminho induzido $z_0 - \dots - u_{i,p}$.

Agora, suponha que existe uma cobertura $\mathcal{C} = \{S_{j_1}, \dots, S_{j_c}\}$ de U com c conjuntos em \mathcal{S} . Uma vez que \mathcal{C} é uma cobertura de U , podemos definir para cada elemento $u_i \in U$ um índice $c(i)$ tal que $u_i \in S_{c(i)} \in \mathcal{C}$. A estratégia dos guardas é como se segue:

- ocupar inicialmente os vértices $z_0, z'_1, S_{j_1}, \dots, S_{j_c}$;
- se o espião mover-se de algum vértice de $U_{i, \leq r}$ para um i , $Z_{\leq r}$ ou $Z'_{\leq r+1}$, então os guardas ocupam os vértices iniciais, controlando o espião pela Proposição 3(b);
- se $(U_{i, > r} \xrightarrow{e} U_{i, > r})$, então $(S_{c(i)} \xrightarrow{g} u_{i,1} ; z_0 \xrightarrow{g} S_{c(i)} ; z'_1 \xrightarrow{g} z_0)$ com o guarda de $u_{i,1}$ controlando o espião no caminho $u_{i,1} - \dots - u_{i,p}$ pela Proposição 3(b);
- se $(u_{i,r+1} \xrightarrow{e} u_{j,r+1})$ com $c(i) = c(j)$ (lembramos que $s = 2r + 2$), então $(S_{c(i)} \xrightarrow{g} u_{j,1} ; u_{i,1} \xrightarrow{g} S_{c(i)})$ com o guarda de $u_{j,1}$ controlando o espião;
- se $(u_{i,r+1} \xrightarrow{e} u_{j,r+1})$ com $c(i) \neq c(j)$, então $(S_{c(j)} \xrightarrow{g} u_{j,1} ; z_0 \xrightarrow{g} S_{c(j)} ; S_{c(i)} \xrightarrow{g} z_0 ; u_{i,1} \xrightarrow{g} S_{c(i)})$ com o guarda de $u_{j,1}$ controlando o espião;
- se o espião está em algum vértice de $U_{i, > r}$, o guarda em z_0 vigia os caminhos $z_0 - \dots - z_{p-1}$ e $z'_1 - \dots - z'_{p+1}$ pela Proposição 3(b), uma vez que $s = 2r + 2$ e o espião não pode ir de $u_{i,r+1}$ para z'_{r+2} ; e
- se o espião mover-se de um vértice de $U_{i, > r}$ para um vértice de $U_{j, \leq r}$, $Z_{\leq r}$ ou $Z'_{\leq r+1}$, então os guardas ocupam os vértices iniciais $(z_0 \xrightarrow{g} z'_1 ; S_{c(i)} \xrightarrow{g} z_0 ; u_{i,1} \xrightarrow{g} S_{c(i)})$.

Facilmente vemos que, com a estratégia acima, os guardas vencem o jogo em todos os casos

possíveis.



5 CONCLUSÕES E TRABALHOS FUTUROS

Neste capítulo apresentamos nossas conclusões e trabalhos futuros acerca dos jogo de coloração e perseguição que são objetos de estudo desta tese.

5.1 Considerações finais para os jogos de coloração em grafos

Nesta tese provamos os Teoremas 5 e 6 que, dados um grafo G e um inteiro k , decidir se $\chi_g(G) \leq k$ ou $\Gamma_g(G) \leq k$ são problemas PSPACE-completos. Respondendo, deste modo, a questão aberta há quase 30 anos de Bodlaender. Nas provas dos Teoremas 5 e 6, o valor de k obtido nas reduções (que é $k = \chi(G)$) pode ser muito alto. Considerando apenas $k = 2$ cores em um grafo conexo G , sabemos que $\chi_g(G) \leq 2$ se e somente se G é uma estrela $K_{1,n}$. Para o Jogo da Coloração Gulosa, temos a seguinte caracterização:

Lema 24. *Um grafo conexo G tem $\Gamma_g(G) \leq 2$ se e somente se G é bipartido com um vértice cuja vizinhança é uma das partes.*

Demonstração. Se $\Gamma_g(G) \leq 2$, então G deve ser bipartido, já que $\chi(G) \leq \Gamma_g(G) \leq 2$. Então, assumamos que G é bipartido com partição dos vértices (A, B) .

Suponha que G tem um vértice v de A cuja vizinhança é B . Se Alice colore v na sua primeira jogada (cor 1), então nenhum vértice de B pode ser colorido com a cor 1 e conseqüentemente, todos os vértices de A serão coloridos com a cor 1. Portanto, todos os vértices de B terão cor 2.

Agora suponha que cada vértice de A tem um não-vizinho em B e que cada vértice de B tem um não-vizinho em A . Seja w o primeiro vértice colorido por Alice. Assumamos sem perda de generalidade que $w \in A$. Uma vez que G é conexo, então existe um vértice z em B que não é vizinho de w a distância 3 (isto é, w e z são os extremos de um P_4). Portanto, colorindo z na sua primeira jogada (cor 1), Bob força 3 cores neste P_4 . \square

Contudo, para $k = 3$ cores, a complexidade do Jogo da Coloração Gulosa ainda não é conhecida. Apesar de nossos esforços, não conseguimos resolver esse problema.

Problema 1. *Decidir se $\Gamma_g(G) \leq 3$ é PSPACE-Completo?*

5.2 Considerações finais para o Jogo do Espião

Nesta tese, mostramos também um limite superior para o número de guarda do produto forte de dois grafos gerais G_1 e G_2 para $s \geq 2$ e $d \geq 0$

$$gn_{s,d}(G_1 \boxtimes G_2) \leq gn_{s,d}(G_1) \times gn_{s,d}(G_2).$$

no Teorema 9. Mostramos, no Lema 11, que essa desigualdade é estrita para os grafos grades Rei gerais $P_{2d+3} \boxtimes P_{2d+3}$, isto é,

$$gn_{s,d}(P_{2d+3} \boxtimes P_{2d+3}) < gn_{s,d}(P_{2d+3}) \times gn_{s,d}(P_{2d+3})$$

quando $d \geq 0$ e $s \geq 2$ são inteiros fixos. No Lema 12, que temos a igualdade para os grafos grades Rei gerais $P_{2d+4} \boxtimes P_{2d+4}$, isto é,

$$gn_{s,d}(P_{2d+4} \boxtimes P_{2d+4}) = gn_{s,d}(P_{2d+4}) \times gn_{s,d}(P_{2d+4})$$

para qualquer $d \geq 0$ e $s \geq d+2$. Contudo, nos Lemas 11 e 12 temos poucas guardas (de dois a quatro), então no Lema 13 mostramos que $gn_{s,d}(P_n \boxtimes P_n) \geq (gn_{s,d}(P_n) - 1)^2$ em muitos casos, onde $gn_{s,d}(P_n)$ é um inteiro positivo qualquer.

No Lema 14, mostra-se $gn_{2,1}(P_5 \square P_5) > gn_{2,1}(P_5)^2$ e $gn_{2,1}(P_5 \cdot P_5) > gn_{2,1}(P_5)^2$, isto é, temos exemplos em que o limite superior do Teorema 9 falha no produto cartesiano e no produto lexicográfico. Assim, no Teorema 10, mostramos o limite para o produto lexicográfico de dois grafos gerais G_1 e G_2

$$gn_{s,d}(G_1 \cdot G_2) = gn_{s,d}(G_1)$$

se G_1 possui vértice isolado e, em caso contrário,

$$gn_{s,d}(G_1 \cdot G_2) = \max \{ gn_{s,d}(G_1), gn_{s,d}(G_2) \}.$$

onde $s \geq 2$, $d \geq 2$ são inteiros.

As questões para determinação do valor exato do número de guarda no produto lexicográfico, produto forte e produto cartesiano de dois grafos gerais onde $d \in \{0, 1\}$ ainda permanecem abertas.

Mostramos também, no Teorema 11, um algoritmo XP com tempo $O(n^{3k+2})$ para o Jogo do Espião quando o número de guardas é fixo. Em seguida, no Teorema 13, usamos esse algoritmo XP para obtermos um algoritmo FPT para grafos com poucos P_4 's quando $q(G)$ é o

parâmetro. Contudo, apesar desse resultado positivo, mostrou-se em (COHEN *et al.*, 2018) que o problema de decisão do Jogo do Espião é $W[2]$ -difícil em grafos gerais quando o parâmetro é o número de guardas. No Teorema 14, estendemos esse resultado, mostrando que também é $W[2]$ -difícil para grafos bipartidos para cada velocidade $s \geq 2$ e distância de vigilância $d \geq 0$.

Como problema em aberto, deixamos a questão de se o Jogo do Espião é PSPACE-Completo ou não. Como motivação, observamos que, somente em 2013, provou-se que o jogo de Polícia e Ladrão é PSPACE-Completo (MAMINO, 2013). Ademais, somente em 2015, provou-se que o Jogo de Polícia e Ladrão é EXPTIME-Completo (KINNERSLEY, 2015). Portanto, outra questão em aberto é se o Jogo do Espião também é EXPTIME-Completo ou não.

REFERÊNCIAS

- AIGNER, M.; FROMME, M. A game of cops and robbers. **Discrete Applied Mathematics**, v. 8, p. 1–12, 1984.
- ALON, N.; MOSHKOVITZ, D.; SAFRA, S. Algorithmic construction of sets for k -restrictions. **ACM Transactions on Algorithms**, v. 2, n. 2, p. 153–177, 2006.
- ARAÚJO, J.; LINHARES-SALES, C. On the Grundy number of graphs with few P_4 's. **Discrete Applied Mathematics**, v. 160, n. 18, p. 2514 – 2522, 2012.
- BABEL, L.; OLARIU, S. On the structure of graphs with few P_4 's. **Discrete Applied Mathematics**, v. 84, p. 1–13, 1998.
- BARTNICKI, T.; BREŠAR, B.; GRZYTCZUK, J.; KOVŠE, M.; MIECHOWICZ, Z.; PETERIN, I. Game chromatic number of cartesian product graphs. **Electronic Journal of Combinatorics**, v. 15, 2008.
- BAUMANN, S. A linear algorithm for the homogeneous decomposition of graphs. **Zentrum für Mathematik, Technische Universität München**, p. Report No. M–9615, 1996.
- BODLAENDER, H. L. On the complexity of some coloring games. **International Journal of Foundations of Computer Science**, v. 02, n. 02, p. 133–147, 1991. Disponível em: <https://doi.org/10.1142/S0129054191000091>. Acesso em: 21 set. 2022.
- BOHMAN, T.; FRIEZE, A.; SUDAKOV, B. The game chromatic number of random graphs. **Random Structures & Algorithms**, v. 32, n. 2, p. 223–235, 2008.
- BONATO, A.; NOWAKOVSKI, R. **The game of Cops and Robber on Graphs**. [S. l.]: American Mathematical Society, 2011.
- BONDY, J. A.; MURTY, U. S. R. **Graph theory**. New York: Springer, 2008. v. 244. (Graduate Texts in Mathematics, v. 244).
- COHEN, N.; HILAIRE, M.; MARTINS, N. A.; NISSE, N.; PÉRENNES, S. Spy-Game on Graphs. In: **8th International Conference on Fun with Algorithms (FUN 2016)**. [S. l.: s. n.], 2016. (Leibniz Intern. Proc. in Informatics (LIPIcs), v. 49), p. 10:1–10:16.
- COHEN, N.; INERNEY, F. M.; NISSE, N.; PÉRENNES, S. Study of a combinatorial game in graphs through linear programming. **Algorithmica**, v. 82, p. 212 – 244, 2020.
- COHEN, N.; MARTINS, N. A.; INERNEY, F. M.; NISSE, N.; PÉRENNES, S.; SAMPAIO, R. Spy-game on graphs: Complexity and simple topologies. **Theoretical Computer Science**, v. 725, p. 1–15, 2018. ISSN 0304-3975. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0304397517308459>. Acesso em: 21 set. 2022.
- CORNEIL, D.; LERCHS, H.; BURLINGHAM, L. Complement reducible graphs. **Discrete Applied Mathematics**, v. 3, n. 3, p. 163 – 174, 1981.
- COSTA, E.; PESSOA, V. L.; SAMPAIO, R.; SOARES, R. Pspace-hardness of two graph coloring games. **Electronic Notes in Theoretical Computer Science**, v. 346, p. 333 – 344, 2019. ISSN 1571-0661. The proceedings of Lagos 2019, the tenth Latin and American Algorithms, Graphs and Optimization Symposium (LAGOS 2019). Disponível em: <http://www.sciencedirect.com/science/article/pii/S1571066119300805>. Acesso em: 21 set. 2022.

COSTA, E.; PESSOA, V. L.; SAMPAIO, R.; SOARES, R. Pspace-completeness of two graph coloring games. **Theoretical Computer Science**, v. 824-825, p. 36–45, 2020. ISSN 0304-3975. Disponível em: <https://www.sciencedirect.com/science/article/pii/S030439752030181X>. Acesso em: 21 set. 2022.

COSTA, E. R.; MARTINS, N. A.; SAMPAIO, R. Spy game: Fpt-algorithm and results on graph products. In: . http://cocoon-conference.org/2021/Accepted_Papers.php: [S. n.], 2022.

COSTA, E. R.; MARTINS, N. A.; SAMPAIO, R. Spy game: FPT-algorithm, hardness and graph products. **Theoretical Computer Science**, 2022. ISSN 0304-3975. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0304397522003061>. Acesso em: 21 set. 2022.

CYGAN, M.; FOMIN, F. V.; KOWALIK, L.; LOKSHTANOV, D.; MARX, D.; PILIPCZUK, M.; PILIPCZUK, M.; SAURABH, S. **Parameterized Algorithms**. 1st. ed. [S. l.]: Springer Publishing Company, Incorporated, 2015. ISBN 3319212745, 9783319212746.

DINSKI, T.; ZHU, X. A bound for the game chromatic number of graphs. **Discrete Mathematics**, v. 196, n. 1, p. 109 – 115, 1999.

DOWNEY; FELLOWS. **Fundamentals of Parameterized Complexity (Texts in Computer Science)**. [S. l.]: Springer-Verlag London, 2013.

DOWNEY, R. G.; FELLOWS, M. R. **Parameterized Complexity**. [S. l.]: Springer Publishing Company, Incorporated, 2012. ISBN 1461267986, 9781461267980.

DUDENEY, H. **Amusements in Mathematics**. [S. l.]: Thomas Nelson and Sons Ltd. Edinburgh, 1917.

DUNN, C.; LARSEN, V.; LINDKE, K.; RETTER, T.; TOCI, D. The game chromatic number of trees and forests. **Discrete Mathematics & Theoretical Computer Science**, Vol. 17 no.2, 2015.

FAIGLE, U.; KERN, U.; KIERSTEAD, H. A.; TROTTER, W. T. On the game chromatic number of some classes of graphs. **Ars Combinatoria**, v. 35, p. 143 – 150, 1993.

FLUM, J.; GROHE, M. **Parameterized Complexity Theory (Texts in Theoretical Computer Science. An EATCS Series)**. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006. ISBN 3540299521.

FORTNOW, L. Beyond NP: The work and legacy of larry stockmeyer. In: **Proceedings of the Thirty-Seventh Annual ACM Symposium on Theory of Computing**. New York, NY, USA: Association for Computing Machinery, 2005. (STOC '05), p. 120–127. ISBN 1581139608. Disponível em: <https://doi.org/10.1145/1060590.1060609>. Acesso em: 21 set. 2022.

GARDNER, M. Mathematical games. **Scientific American**, v. 244, n. 4, p. 18 – 26, 1981.

GAREY, M.; JOHNSON, D. **Computers and Intractability: A Guide to the Theory of NP-Completeness**. New York, USA: W. H. Freeman & Co., 1979.

GIAKOUMAKIS, V. P_4 -laden graphs: A new class of brittle graphs. **Information Processing Letters**, v. 60, n. 1, p. 29 – 36, 1996.

GODDARD, W.; HEDETNIEMI, S.; HEDETNIEMI, S. Eternal security in graphs. **Journal of Combinatorial Mathematics and Combinatorial Computing**, v. 52, 2005.

GOLDWASSER, J. L.; KLOSTERMEYER, W. Tight bounds for eternal dominating sets in graphs. **Discrete Mathematics**, v. 308, n. 12, p. 2589–2593, 2008.

HAMMACK, R.; IMRICH, W.; KLAVŽAR, S. **Handbook of Product Graphs**. [S. l.]: CRC Press, 2011.

HAVET, F.; ZHU, X. The game Grundy number of graphs. **Journal of Combinatorial Optimization**, v. 25, n. 4, p. 752–765, 2013.

HEARN, R. A.; DEMAINE, E. D. **Games, Puzzles, and Computation**. Natick, MA, USA: A. K. Peters, Ltd., 2009. ISBN 1568813228.

JAMISON, B.; OLARIU, S. A tree representation for P_4 -sparse graphs. **Discrete Applied Mathematics**, v. 35, n. 2, p. 115 – 129, 1992.

KIERSTEAD, H. A.; TROTTER, W. T. Planar graph coloring with an uncooperative partner. **Journal of Graph Theory**, v. 18, n. 6, p. 569–584, 1994.

KINNERSLEY, W. B. Cops and robbers is exptime-complete. **Journal of Combinatorial Theory, Series B**, v. 111, p. 201–220, 2015.

KLOSTERMEYER, W.; MACGILLIVRAY, G. Eternal dominating sets in graphs. **Journal of Combinatorial Mathematics and Combinatorial Computing**, v. 68, 2009.

KLOSTERMEYER, W.; MYNHARDT, C. Graphs with equal eternal vertex cover and eternal domination numbers. **Discrete Mathematics**, v. 311, n. 14, p. 1371–1379, 2011.

KRAWCZYK, T.; WALCZAK, B. Asymmetric coloring games on incomparability graphs. **Electronic Notes in Discrete Mathematics**, v. 49, p. 803 – 811, 2015. ISSN 1571-0653. Eurocomb-2015.

LIMA, C. V.; MARCILON, T.; MARTINS, N.; SAMPAIO, R. The connected greedy coloring game. **submitted do Theoretical Computer Science**, 2022.

LIMA, C. V.; MARCILON, T.; MARTINS, N.; SAMPAIO, R. PSPACE-hardness of variants of the graph coloring game. **Theoretical Computer Science**, v. 909, p. 87–96, 2022. ISSN 0304-3975. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0304397522000494>. Acesso em: 21 set. 2022.

MAMINO, M. On the computational complexity of a game of cops and robbers. **Theoretical Computer Science**, v. 477, p. 48–56, 2013.

MOSHKOVITZ, D. The projection games conjecture and the NP-hardness of $\ln n$ -approximating set-cover. **Theory of Computing**, Theory of Computing, v. 11, n. 7, p. 221–235, 2015.

NAKPRASIT, K. M.; NAKPRASIT, K. The game coloring number of planar graphs with a specific girth. **Graphs and Combinatorics**, v. 34, n. 2, p. 349–354, 2018.

PESSOA, V. L. **Estudo de complexidade de jogos de coloração**. Dissertação (Mestrado) – Departamento de Estatística e Matemática Aplicada, Universidade Federal do Ceará, Fortaleza - Ceará, 2019.

SCHAEFER, T. J. On the complexity of some two-person perfect-information games. **Journal of Computer and System Sciences**, v. 16, n. 2, p. 185 – 225, 1978.

SEKIGUCHI, Y. The game coloring number of planar graphs with a given girth. **Discrete Mathematics**, v. 330, p. 11 – 16, 2014.

SIDOROWICZ, E. The game chromatic number and the game colouring number of cactuses. **Information Processing Letters**, v. 102, n. 4, p. 147 – 151, 2007.

STEVEN. **K7 on the torus**. 2008. Disponível em: <http://www.amotlpaa.org/math/k7torus.html>. Acesso em: 14 jun. 2021.

STOCKMEYER, L. J.; MEYER, A. R. Word problems requiring exponential time (preliminary report). In: **Proceedings of the Fifth Annual ACM Symposium on Theory of Computing**. New York, NY, USA: Association for Computing Machinery, 1973. (STOC '73), p. 1–9. ISBN 9781450374309. Disponível em: <https://doi.org/10.1145/800125.804029>. Acesso em: 21 set. 2022.

WIKIPEDIA. **Genus of orientable surfaces**. 2021. Disponível em: [https://en.wikipedia.org/wiki/Genus_\(mathematics\)](https://en.wikipedia.org/wiki/Genus_(mathematics)). Acesso em: 14 jun. 2021.

ZERMELO, E. Über eine Anwendung der Mengenlehre auf die Theorie des Schachspiels. In: **Proc. Fifth International Congr. Math.** [S. l.: s. n.], 1913. p. 501–504.

ZHU, X. The game coloring number of planar graphs. **Journal of Combinatorial Theory, Series B**, v. 75, n. 2, p. 245 – 258, 1999.

ZHU, X. The game coloring number of pseudo partial k-trees. **Discrete Mathematics**, v. 215, n. 1, p. 245 – 262, 2000.

ZHU, X. Refined activation strategy for the marking game. **Journal of Combinatorial Theory, Series B**, v. 98, n. 1, p. 1 – 18, 2008.