

Teoria dos Jogos Combinatórios em Grafos

Slides de Minicurso Introdutório

Colóquio Brasileiro de Matemática

**Samuel Araújo, Nicolas Martins,
Nicolas Nisse, Rudini Sampaio**

CBM-2025, IMPA,
Rio de Janeiro, 28/07 a 01/08, 8h-9h

Foto do último dia (sexta 01/08/2025)



Teoria dos Jogos Combinatórios em Grafos

Aula 01

**Samuel Araújo, Nicolas Martins,
Nicolas Nisse, Rudini Sampaio**

CBM-2025, IMPA,
Rio de Janeiro, segunda, 28-Julho, 8h

Capítulos 1 e 2

Capítulo 1

Introdução a Jogos Combinatórios

Capítulo 2

Teoria de Sprague-Grundy para Jogos Imparciais

Capítulo 1 - Introdução a Jogos Combinatórios

Capítulo 1

Introdução a Jogos Combinatórios

Primórdios dos Jogos Combinatórios

- ▶ Os jogos fazem parte de todas as culturas e são uma das formas mais antigas de interação social.
- ▶ Jogos do tipo **Mancala** (África, 5000 a.C.)
- ▶ Jogo de **Damas** (Mesopotâmia, 3000 a.C.): **Platão** e **Homero**
- ▶ **Jogo da Velha** (Antigo Egito, 1500 a.C.) e **Go** (China, 500 a.C.)
- ▶ Jogo **Chaturanga** (pai do Xadrez, Índia, 600 d.C.)

“**Matemática Recreativa**”: criação e resolução de quebra-cabeças matemáticos ou *puzzles* (jogos de 1 jogador), entre outros jogos.

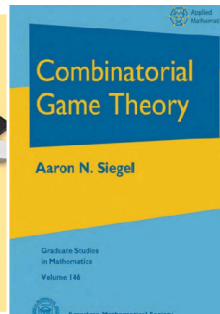
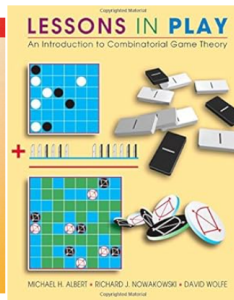
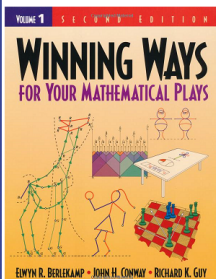
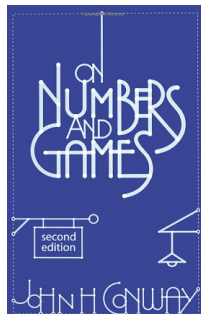
- ▶ **Lewis Carrol**: pseudônimo de Charles Dodgson, autor de *Alice no País das Maravilhas* (1865) e *The Game of Logic*,
- ▶ **Sam Loyd** (*Cyclopedia with 5000 puzzles*, 1898),
- ▶ **Henry Dudeney** (*Amusements in Mathematics*, 1917),
- ▶ **Martin Gardner**: autor da coluna “*Mathematical Games*” da *Scientific American* durante 25 anos, de 1956 a 1981:

“***I just play all the time and I am fortunate enough to get paid for it***”

Nim (1901) + Sprague-Grundy (1936) + Conway (1976)

- ▶ **Charles Bouton (1901)**: primeiro estudo teórico completo de um jogo combinatório não trivial: o jogo **Nim**.
 - ▶ **Sprague (1936)** e **Grundy (1939)** usaram o jogo **Nim** para representar qualquer **jogo normal imparcial** através de números chamados *nimbers*, dando origem a *Teoria de Sprague-Grundy*.
 - ▶ **Conway (1976)** estendeu a Teoria de Sprague-Grundy para qualquer **jogo normal partizan** através de números chamados *surreais*, dando origem a *Teoria Combinatória dos Jogos*.
-
- **Knuth (1969)**: “*Surreal Numbers: How Two Ex-Students Turned on to Pure Mathematics and Found Total Happiness*”.
 - **Conway (1976)**: “*On numbers and games*”.
 - **Berlekamp, Conway e Guy (1982)**: “*Winning ways for your mathematical plays*”.

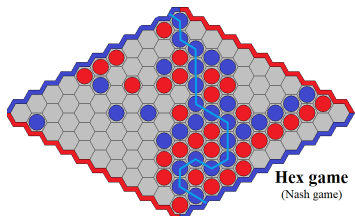
Jogo Normal/Misère: Livros - 1976, 1982, 2007 e 2013



Principalmente para **jogo normal** (imparcial ou partizan), mas também tem um pouco para **jogo misère**.

Jogos Posicionais - Hex - John Nash (1940s)

- ▶ Jogo **Hex**: criado nos anos 1940s independentemente por **Piet Hein** e
- ▶ **John Nash**: Prêmio Nobel de Economia 1994, Equilíbrio de Nash da Teoria Econômica dos Jogos. Filme “*Mente Brilhante*”.
- ▶ Jogo Posicional mais conhecido: **MAKER-BREAKER**. Um jogador deseja obter uma configuração vencedora e o outro deseja impedir.
- ▶ Técnicas principais: *roubo de estratégia (strategy stealing)* e a *estratégia de pareamento (pairing strategy)*.



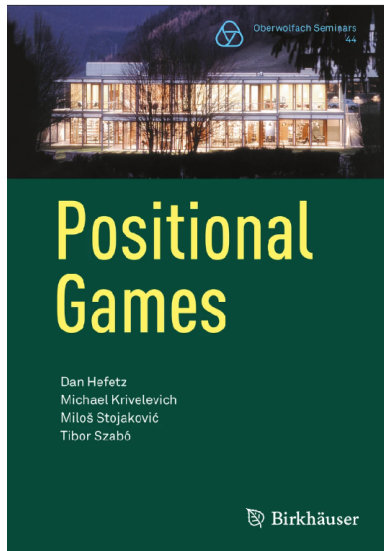
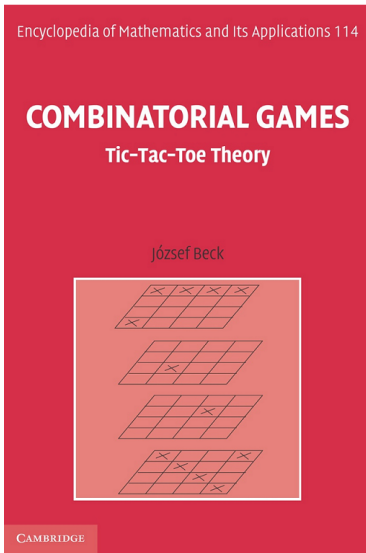
Alice sempre tem estratégia vencedora em Hex, mas não se conhece uma explicitamente para tabuleiro $n \times n$ com $n \geq 10$. Padrão é 11×11

Jogos Posicionais - Combinatória Extremal - 1970s

- ▶ **Paul Erdős**: um dos matemáticos mais prolíficos da história.
- ▶ Combinatória Extremal para Jogos Posicionais: **MAKER-BREAKER**.
- ▶ Método Probabilístico + Técnica do Potencial
- ▶ Teorema de **Erdős-Selfridge (1973)** para MAKER-BREAKER
- ▶ Teorema de **Erdős-Chvátal (1978)** para MAKER-BREAKER tendencioso
- ▶ Teoremas de **Beck (1981)** para o jogo Clique-Ramsey e o jogo de van der Waerden (ou jogo da progressão aritmética).
- ▶ Teoremas de **Beck (2002)** para MAKER-BREAKER tendencioso

Exemplo: O Jogo **Clique** (n, k) é difícil para valores “pequenos” de n . Mas para $n = 2^{(33+10^{10})}$, sabe-se que Alice vence se e só se $k \leq 2 \cdot 10^{10} - 1$. Ou seja, Alice não consegue colorir as arestas de uma clique com 20 bilhões de vértices, mas consegue com qualquer número menor. É surpreendente que esse problema seja tão difícil para instâncias pequenas, mas é possível calcular o valor exato para instâncias tão grandes.

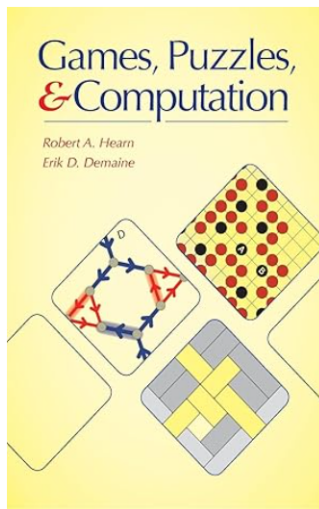
Livros sobre Jogos Posicionais: 2008 e 2014



Complexidade Computacional de Jogos - 1970s

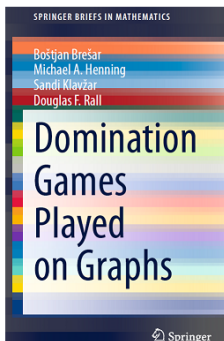
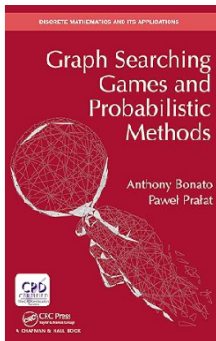
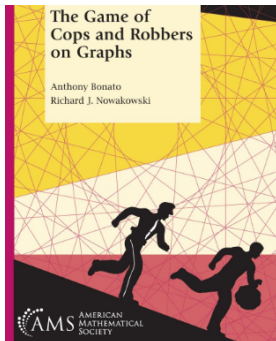
- ▶ Cook-Levin (1971): **SAT** é NP-completo.
- ▶ **SAT**: $(x_1 \vee x_2) \wedge (x_1 \vee \overline{x_2}) \wedge (\overline{x_1} \vee x_2) \wedge (\overline{x_1} \vee \overline{x_2})$
- ▶ Karp (1972): Vários problemas NP-completos em grafos (**Clique**,...)
- ▶ Stockmeyer-Meyer (1973): 1º jogo PSPACE-completo: **QSAT**.
- ▶ **QSAT** (ou **TQBF**): $\exists x_1 \forall x_2 : (x_1 \vee x_2) \wedge (\overline{x_1} \vee \overline{x_2})$
 $\exists x_0 \forall x_1 \exists x_2 : (x_1 \vee x_2) \wedge (\overline{x_1} \vee \overline{x_2})$
- ▶ Even-Tarjan (1976) e Schafer (1976): Vários jogos em grafos são PSPACE-completos, como **Kayles**
- ▶ **PosCNF**: $(x_1 \vee x_3) \wedge (x_1 \vee x_4) \wedge (x_2 \vee x_3) \wedge (x_2 \vee x_4)$
- ▶ Stockmeyer-Chandra (1979): 1º jogo EXPTIME-completo: **ABF** (*Alternating Boolean Formula game*).
- ▶ **ABF**: $(x_{A1} \vee x_{B1} \vee x_{A2} \vee x_{B2}) \wedge$
 $(x_{A1} \vee x_{B1} \vee \overline{x_{A2}} \vee \overline{x_{B2}}) \wedge$
 $(\overline{x_{A1}} \vee \overline{x_{B1}} \vee x_{A2} \vee x_{B2}) \wedge$
 $(\overline{x_{A1}} \vee \overline{x_{B1}} \vee \overline{x_{A2}} \vee \overline{x_{B2}})$

Livros sobre Complexidade de Jogos: 2009



Jogos Combinatórios em Grafos

- ▶ Jogos de Coloração em Grafos
- ▶ Jogos de Convexidade em Grafos
- ▶ Jogos de Perseguição em Grafos: Cops and Robber (2011)
- ▶ Jogos de Busca em Grafos (2017)
- ▶ Jogos de Dominação em Grafos (2021)



Desse resumo, vemos que a área de Jogos não é brincadeira !!

O que é um Jogo Combinatório?

Mas: O que é um Jogo Combinatório?

Poker, Ludo, Gamão, War ou o Dilema dos Prisioneiros
são jogos combinatórios?

Definição de Jogo Combinatório

Posição de jogo: estado do jogo, incluindo quem é o próximo a jogar.

Jogo Combinatório:

- ▶ **2 jogadores:** geralmente **Alice** (1º jogador) e **Bob** (2º jogador);
- ▶ **Sequencial:** jogadores se alternam em turnos;
- ▶ **Finito:** há um número finito de posições e há alguma regra limitando o número de repetições possíveis de cada posição;
- ▶ **Informação completa (ou perfeita):** em cada momento, todos os jogadores conhecem todas as informações referentes ao jogo;
- ▶ **Sem aleatoriedade:** por exemplo, sem lançamento de dados;
- ▶ **Resultado:** vitória, derrota ou empate.
- ▶ **Regras razoáveis:** cada jogador pode determinar rapidamente (tempo polinomial no tamanho da posição) os movimentos possíveis, as posições obtidas com cada movimento e se a posição atingiu as condições de término do jogo e, nesse caso, qual o resultado.

Teoria Econômica dos Jogos permite jogos com informação incompleta e resultados diferentes (como o Dilema dos Prisioneiros).

Árvore de Jogo (game tree)

Grafo direcionado sobre as posições do jogo, finito e sem ciclos.

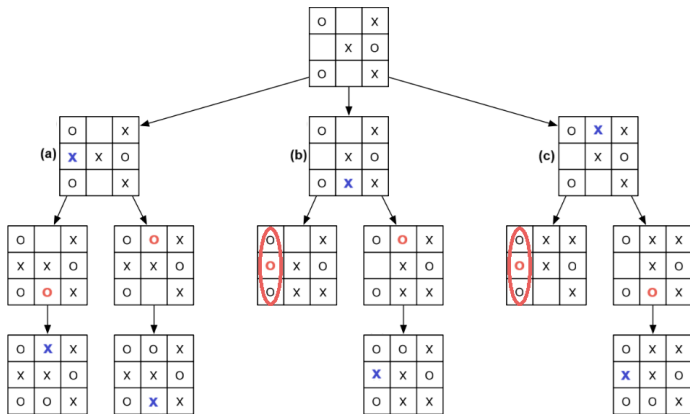
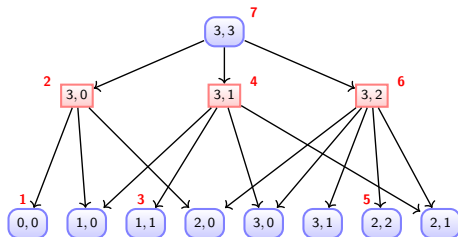
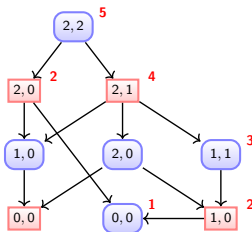


Figure: Árvore de Jogo a partir de uma posição de TIC-TAC-TOE. Note que Alice empata jogando em (a), mas perde jogando em (b) ou (c).

Árvore de Jogo (game tree) - Nim normal

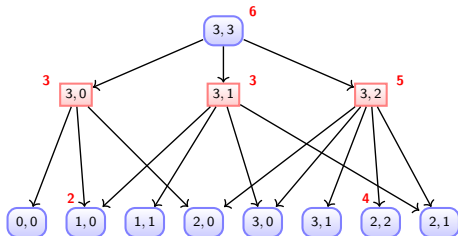
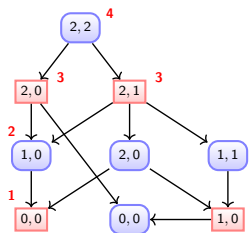
Jogo Nim (normal): k pilhas de objetos em que a pilha i contém n_i objetos. Em cada turno, o jogador escolhe uma pilha e remove qualquer quantidade de objetos de lá. Perde o jogador que não conseguir fazer uma jogada em seu turno, ou seja, todas as pilhas estão vazias. Uma posição de NIM pode ser definida por $(n_1, \dots, n_k)$ e, por simetria, podemos assumir que $n_1 \geq \dots \geq n_k$.



Exercício 1.1: Prove que Bob vence Nim em posições (n, n) .

Árvore de Jogo (game tree) - Nim misère

Jogo Nim-misère: Semelhante a **Nim**, mas vence o jogador que não conseguir fazer uma jogada em seu turno.



Exercício 1.2: Prove que Bob vence **Nim-misère** em posições (n, n) se e só se $n \geq 2$.

Raiz (posição inicial), **Folhas** (posições finais)

Profundidade do jogo: altura da árvore, diâmetro do grafo direcionado (maior distância da raiz para alguma folha).

Estratégia Vencedora e o Teorema de Zermelo

Estratégia vencedora para um jogador: função que determina um movimento para cada posição possível em que o jogador é o próximo a jogar tal que, jogando conforme a função a partir da posição inicial, sempre resultará em uma posição vencedora para o jogador, independente dos movimentos do adversário.

Estratégia de empate: análogo.

Dizemos que um jogador **vence** (resp. **empata**) se possui estratégia vencedora (resp. de empate). Senão, dizemos que o jogador **perde**.

Teorema de Zermelo (1913): Em todo jogo combinatório, um jogador tem estratégia vencedora, ou ambos tem estratégia de empate.

Isso não ocorre no jogo trivialíssimo com 3 jogadores X, Y e Z e apenas uma jogada: X escolhe o vencedor entre Y e Z.

Problema de decisão do jogo: Alice (1º jogador) vence?

Algoritmo para ver se Alice vence

► Algoritmo Vitória-Alice:

1. Marque com rótulo 1 as folhas vencedoras para Alice
2. $T \leftarrow 1$; nova $\leftarrow$ **true**
3. **enquanto** (nova) **faça**
4. $T \leftarrow T + 1$; nova $\leftarrow$ **false**
5. **para cada** posição P_A não marcada de Alice:
6. **se** (P_A leva a uma posição marcada) **então**
7. Marque P_A com rótulo T ; nova $\leftarrow$ **true**
8. $T \leftarrow T + 1$
9. **para cada** posição P_B não marcada de Bob:
10. **se** (P_B leva somente a posições marcadas) **então**
11. Marque P_B com rótulo T ; nova $\leftarrow$ **true**
12. **se** (a raiz está marcada) **então**
13. “Alice tem estratégia vencedora”
14. **senão** “Alice não tem estratégia vencedora”

Algoritmos Vitória-Alice, Vitória-Bob, Empate-Alice, Empate-Bob.

Tempo polinomial no número de posições do jogo.

Exemplo: Jogo Nim com k pilhas: Tempo $O(n^k)$.

Classificação de Jogos Combinatórios

Classificação quanto às Regras:

- ▶ **Jogo Imparcial:** Os movimentos e os resultados possíveis em uma dada posição são os mesmos para ambos os jogadores. Ou seja, para qualquer posição, os movimentos que Alice poderia fazer se fosse ela a próxima a jogar e os resultados obtidos desses movimentos são os mesmos caso Bob fosse o próximo a jogar. **Nim, Nim-misère**
- ▶ **Jogo Partizan:** jogo não imparcial. **Jogo da Velha, Nim-partizan**

Classificação quanto ao Critério de Vitória:

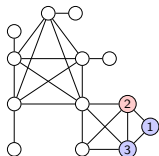
- ▶ **Variante Normal:** O último a jogar vence. **Nim.**
- ▶ **Variante Misère:** O último a jogar perde. **Nim-misère.**
- ▶ **Jogo de Objetivo:** cada jogador possui um objetivo (um tipo de posição a se alcançar ou se evitar). **Jogo da Velha.**
- ▶ **Jogo de Otimização:** acrescenta-se um número k à instância do jogo de modo que um jogador deseja que certo parâmetro do jogo seja limitado por k (superiormente ou inferiormente, dependendo do jogo) **no fim do jogo** e o outro deseja impedir. **Clique**

Exemplo de jogos imparciais: Kayles e Clique-Forming

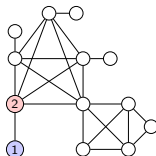
Kayles (normal ou misère): Alice e Bob se alternam selecionando vértices de modo que os vértices selecionados sempre formem um conjunto independente (sem arestas entre os vértices).

Kayles de otimização: além do grafo, é dado um inteiro k e Alice vence se o número de vértices selecionados for $\geq k$. Alice deseja maximizar o número de vértices selecionados e Bob deseja minimizar.

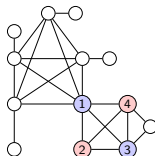
Clique-Forming: similar (clique ao invés de conjunto independente). Kayles no complemento do grafo.



(a) Variante Normal



(b) Variante Misère



(c) Variante Otimização

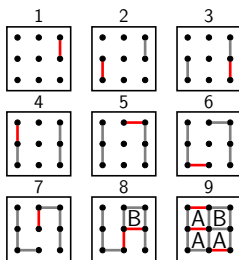
Alice vence em todas as três com estratégias bem diferentes. Alice (azul), Bob (vermelho). Os números indicam a ordem da seleção dos vértices.

Kayles (normal, misère e de otimização) são PSPACE-completos (Schaefer, 1978), (Chandran et al., 2023) e (Brosse et al., 2025).

Dot-and-Boxes: um jogo imparcial de objetivo

Muito popular antigamente: Timbiriche ou jogo dos pontinhos (em português).

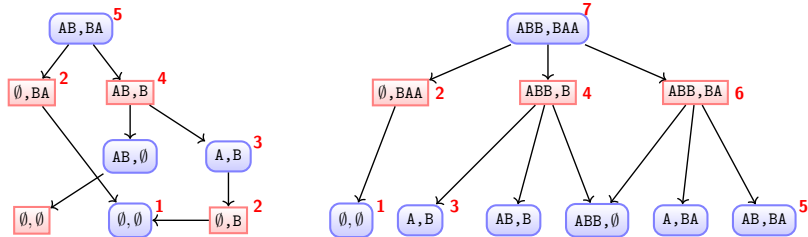
Os jogadores marcam linhas horizontais ou verticais entre pontos vizinhos (dist. 1) numa grade $n \times n$. Quando um jogador fecha um quadrado 1×1 , marca a inicial de seu nome no quadrado e ganha um ponto e mais um turno. No final, vence quem tiver mais pontos. O “objetivo” de cada jogador é atingir uma posição final com mais quadrados marcados com sua inicial.



Note que Alice vence por 3×1 nesse exemplo. As arestas vermelhas são as escolhidas no turno corrente.

Exemplo de jogo partizan: Nim-partizan

Jogo Nim-partizan (normal ou misère): os objetos das pilhas são rotulados com A, B ou C e, em cada turno, Alice (resp. Bob) só pode remover um objeto rotulado com A ou C (resp. B ou C), juntamente com todos os objetos que estão acima dele.



(ABB, BAA): a primeira pilha contém objetos rotulados A, B e B (de baixo para cima) e a segunda pilha B, A e A.

Variantes partizan de jogos imparciais

Kayles-partizan: Os vértices do grafo já vem rotulados com A, B ou C. Alice só pode selecionar vértices A ou C, e Bob só pode selecionar vértices B ou C.

Dot-and-Boxes-partizan: cada aresta selecionada é desenhada com a cor do jogador. Vence o jogador com mais quadrados 1x1 com todas as arestas na sua cor.

Dois tipos comuns de jogos partizan:

(1) os elementos do jogo já vem coloridos e os jogadores só podem jogar em objetos com certo conjunto de cores, que é diferente do conjunto de cores do oponente. **Nim-partizan**

(2) os jogadores colorem com sua cor os elementos jogados e o resultado do jogo depende das cores. **Jogo da Velha.**

Jogos Posicionais

Jogos posicionais são jogos de objetivo em que há um conjunto universo $U = \{u_1, \dots, u_n\}$ (**elementos do tabuleiro**) e uma família $\mathcal{F}$ de subconjuntos de U , chamados de **conjuntos críticos**. Dependendo da variante, são chamados de conjuntos **vencedores** ou **perdedores**.

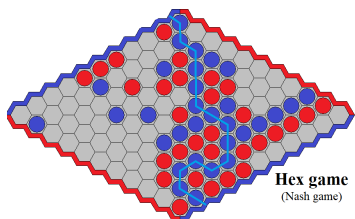
Variantes principais de jogos posicionais:

- ▶ **Maker-Maker**: vence o primeiro a obter um conjunto vencedor.
- ▶ **Maker-Breaker**: Alice vence se conseguir um conjunto vencedor; caso contrário, Bob vence.
- ▶ **Avoider-Forcer**: Alice vence se conseguir evitar um conjunto perdedor; caso contrário, Bob vence.
- ▶ **Avoider-Avoider**: perde o primeiro a obter um conjunto perdedor.
- ▶ **Waiter-Client**: Similar ao **MAKER-MAKER**, mas, em cada turno, Alice pega 2 elementos, Bob escolhe um e Alice o outro.
- ▶ **Scoring Game** (pontuação): no fim do jogo, vence quem tiver o conjunto vencedor com mais pontos, onde a pontuação de cada conjunto vencedor é dada na instância.

Exemplo MAKER-MAKER: Jogo da Velha. Tabuleiro: casas do quadrado 3×3
Conjuntos vencedores: 3 linhas, 3 colunas e 2 diagonais.

Hex: um jogo posicional MAKER-BREAKER

Sabe-se que Alice nunca perde jogos MAKER-MAKER e que não há empate em jogos MAKER-BREAKER (roubo de estratégia).



Exemplo: jogo Hex: É um jogo MAKER-BREAKER com tabuleiro de células hexagonais. Alice de azul tenta conectar os lados azuis, enquanto Bob tenta impedir de vermelho. O único modo de Bob impedir é conectando os lados vermelhos. Por isso, muitos acham que Hex é MAKER-MAKER. Mas não é. Alice vence jogos MAKER-BREAKER simétricos, como Hex. Mas não se conhece estratégia explícita para tabuleiro $n \times n$ com $n \geq 10$. Padrão é 11×11 .

Clique (MAKER-MAKER) e Sim (Avoider-Avoider)

Jogos posicionais de **coloração de arestas**. Dados inteiros $n > k > 0$, cada jogador escolhe uma aresta do grafo completo K_n e a colore com sua cor.

- ▶ Jogo **Clique** é **MAKER-MAKER**: vence quem primeiro obter uma clique de tamanho k com sua cor.
- ▶ Jogo **Sim** é **Avoider-Avoider**: perde quem primeiro obter uma clique de tamanho k com sua cor (Simmons, 1969).

A versão mais comuns desse jogo é com $(k, n) = (3, 6)$, em que **não há empate** pelo **número de Ramsey** $R(3) = 6$.

Outra possibilidade seria $(k, n) = (4, 18)$.

Capítulo 2

Teoria de Sprague-Grundy para Jogos Imparciais

Solução do Jogo Nim por Bouton (1901)

Operação $\oplus$ (ou-exclusivo bit a bit): comutativa e associativa:

$$0 \oplus 0 = 0, \quad 0 \oplus 1 = 1 \oplus 0 = 1 \quad \text{e} \quad 1 \oplus 1 = 0.$$

Exemplos:

$$1 \oplus 2 = 01_2 \oplus 10_2 = 11_2 = 3$$

$$2 \oplus 3 = 10_2 \oplus 11_2 = 01_2 = 1$$

$$1 \oplus 3 = 01_2 \oplus 11_2 = 10_2 = 2$$

$$1 \oplus 2 \oplus 3 = 01_2 \oplus 10_2 \oplus 11_2 = 00_2 = 0$$

Nimber $\text{nim}(n_1, \dots, n_k) = n_1 \oplus \dots \oplus n_k$.

Teorema de Bouton (1901):

Seja $(n_1, \dots, n_k)$ uma instância de **Nim**.

Alice vence **Nim** (variante normal) **se e só se** $\text{nim}(n_1, \dots, n_k) > 0$.

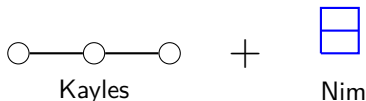
Alice vence **Nim misère** **se e só se** $\text{nim}(n_1, \dots, n_k) > 0$ e existe pelo menos uma pilha com mais de 1 objeto, ou $\text{nim}(n_1, \dots, n_k) = 0$ e toda pilha não vazia tem exatamente 1 objeto.

Teorema de Sprague-Grundy (1936)

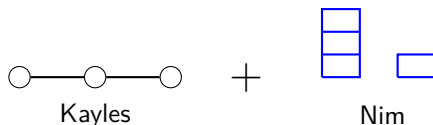
Teoria de Sprague-Grundy é para jogos imparciais na variante normal.

Soma de Jogos: Dados jogos $J_1, \dots, J_k$ (imparciais normais), define-se a soma $J_1 + \dots + J_k$ como o jogo em que cada jogador escolhe uma das k posições (digamos J_i) e faz uma jogada em J_i . É fácil ver que a soma de jogos é comutativa e associativa.

Exemplo: Bob vence no jogo abaixo.



Exemplo: Bob também vence no jogo abaixo.



Teorema de Sprague-Grundy (1936)

Teoria de Sprague-Grundy é para jogos imparciais na variante normal.

Nimber de uma posição: natural associado a um jogo Nim com 1 pilha.

Definição recursiva na profundidade do jogo:

Se não há jogadas possíveis (profundidade 0), o nimber é igual a 0.

Caso contrário, é o **mínimo excludente** $\text{mex}\{n_1, \dots, n_k\}$, onde $\{n_1, \dots, n_k\}$ é o conjunto de todos os nimbers das posições que podem ser obtidas após uma jogada e o **mínimo excludente** mex é o mínimo número natural não pertencente ao conjunto.

Exemplo: Jogo Nim-bolsas: a instância contém pilhas e bolsas com pilhas de objetos sobre uma mesa. Ao escolher uma pilha, deve remover alguns objetos dela. Ao escolher uma bolsa, deve escolher uma pilha da bolsa (sem remover objetos ds pilha), colocar sobre a mesa e descartar o resto da bolsa.

Bolsas $X = \{0, 2, 3, 4\}$, $Y = \{0, 1, 3, 4\}$ e $Z = \{0, 1, 2, 4\}$.

$$\text{nim}(X) = \text{mex}\{0, 2, 3, 4\} = 1$$

$$\text{nim}(Y) = \text{mex}\{0, 1, 3, 4\} = 2$$

$$\text{nim}(Z) = \text{mex}\{0, 1, 2, 4\} = 3$$

Teorema de Sprague-Grundy (1936)

Teoria de Sprague-Grundy é para jogos imparciais na variante normal.

Teorema de Sprague-Grundy:

O 1º jogador vence uma posição de um jogo imparcial na variante normal se e só se ela tem nimber maior que 0. Além disso, uma posição que consiste da soma de k posições disjuntas com nimbers $n_1, \dots, n_k$ tem nimber $n_1 \oplus \dots \oplus n_k$.

Exemplo: Jogo Nim-bolsas: a instância contém pilhas e bolsas com pilhas de objetos sobre uma mesa. Ao escolher uma pilha, deve remover alguns objetos dela. Ao escolher uma bolsa, deve escolher uma pilha da bolsa (sem remover objetos ds pilha), colocar sobre a mesa e descartar o resto da bolsa.

Bolsas $X = \{0, 2, 3, 4\}$, $Y = \{0, 1, 3, 4\}$ e $Z = \{0, 1, 2, 4\}$.

$$\text{nim}(X) = \text{mex}\{0, 2, 3, 4\} = 1$$

$$\text{nim}(Y) = \text{mex}\{0, 1, 3, 4\} = 2$$

$$\text{nim}(Z) = \text{mex}\{0, 1, 2, 4\} = 3$$

$$\text{nim}(X + Y + Z) = 1 \oplus 2 \oplus 3 = 0 \quad \Rightarrow \quad \text{Bob vence}$$

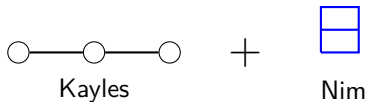
Teorema de Sprague-Grundy (1936)

Teoria de Sprague-Grundy é para jogos imparciais na variante normal.

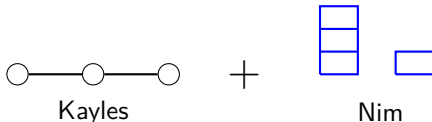
Teorema de Sprague-Grundy:

O 1º jogador vence uma posição de um jogo imparcial na variante normal se e só se ela tem nimber maior que 0. Além disso, uma posição que consiste da soma de k posições disjuntas com nimbers $n_1, \dots, n_k$ tem nimber $n_1 \oplus \dots \oplus n_k$.

Exemplo: Bob vence no jogo abaixo.



Exemplo: Bob também vence no jogo abaixo.



Sprague-Grundy - Kayles em caminhos e ciclos

$$\begin{aligned} \text{nim}(P_0) &= 0; & \text{nim}(P_1) &= \text{nim}(P_2) = \text{mex}\{0\} = 1; \\ \text{nim}(P_3) &= \text{mex}\{0, 1\} = 2; & \text{nim}(P_4) &= \text{mex}\{1\} = 0. \\ \text{nim}(P_5) &= \text{mex}\{0, 1, 2\} = 3. \end{aligned}$$

$$\text{nim}(P_n) = \text{mex} \left\{ \text{nim}(P_{n-2}), \right. \\ \left. \text{nim}(P_{k-2}) \oplus \text{nim}(P_{n-k-1}) : k = 2, \dots, \lceil n/2 \rceil \right\}$$

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33
0	1	1	2	0	3	1	1	0	3	3	2	2	4	0	5	2	2	3	3	0	1	1	3	0	2	1	1	0	4	5	2	7	4
0	1	1	2	0	3	1	1	0	3	3	2	2	4	4	5	5	2	3	3	0	1	1	3	0	2	1	1	0	4	5	3	7	4
8	1	1	2	0	3	1	1	0	3	3	2	2	4	4	5	5	9	3	3	0	1	1	3	0	2	1	1	0	4	5	3	7	4
8	1	1	2	0	3	1	1	0	3	3	2	2	4	4	5	5	9	3	3	0	1	1	3	0	2	1	1	0	4	5	3	7	4

Teorema:

Bob vence **Kayles** no grafo **caminho** P_n **se e só se** $n \in \{0, 14, 34\}$ ou $n \% 34 \in \{4, 8, 20, 24, 28\}$.

Alice vence **Kayles** no grafo **ciclo** C_n **se e só se** $n \in \{3, 17, 37\}$ ou $n \% 34 \in \{7, 11, 23, 27, 31\}$.

Sprague-Grundy - Kayles em grafos boliche

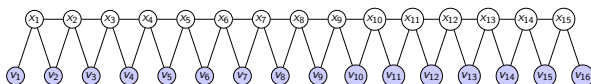


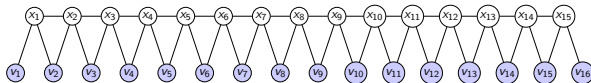
Figure: Grafo boliche B_n com $n = 16$. Os pinos do boliche são os vértices $v_1, v_2, \dots, v_n$ em azul. Alice vence para todo $n > 1$.

$$\text{nim}(B_0) = 0; \quad \text{nim}(B_1) = 1; \quad \text{nim}(B_2) = \text{nim}(P_3) = 2.$$

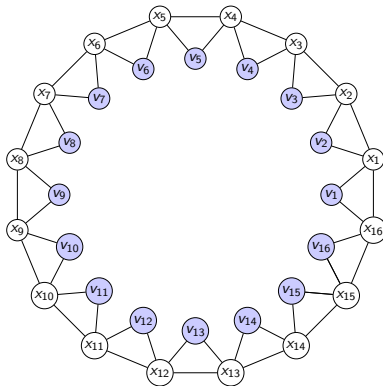
$$\text{nim}(B_n) = \text{mex} \left\{ \begin{array}{l} \text{nim}(B_{n-1}), \text{ nim}(B_{n-2}), \\ \text{nim}(B_{k-1}) \oplus \text{nim}(B_{n-k}), \\ \text{nim}(B_{k-1}) \oplus \text{nim}(B_{n-k-1}) : k = 2, \dots, \lceil n/2 \rceil \end{array} \right\}$$

0	1	2	3	4	5	6	7	8	9	10	11
0	1	2	3	4	5	6	7	8	9	10	11
4	1	2	7	1	4	3	2	1	4	2	6
4	1	2	8	5	4	7	2	1	8	6	7
4	1	2	3	1	4	7	2	1	8	2	7
4	1	2	8	1	4	7	2	1	4	2	7
4	1	2	8	1	4	7	2	1	8	6	7
4	1	2	8	1	4	7	2	1	8	2	7
4	1	2	8	1	4	7	2	1	8	2	7

Sprague-Grundy - Kayles em grafos boliche



Teorema (Guy, Smith, 1956): Seja $n \geq 1$ qualquer. No jogo **Kayles**:
Alice vence no grafo boliche B_n e **Bob** vence no grafo boliche circular B_n^* .



FIM DA AULA 01

FIM DA AULA 01

Perguntas (se houver tempo) ??

Teoria dos Jogos Combinatórios em Grafos

Aula 02

**Samuel Araújo, Nicolas Martins,
Nicolas Nisse, Rudini Sampaio**

CBM-2025, IMPA,
Rio de Janeiro, terça, 29-Julho, 8h

Capítulo 3

Combinatória Extremal para Jogos

Capítulo 4

Jogos Posicionais

Jogos MAKER-MAKER

Instância: $(U, \mathcal{F})$. Conjunto universo $U = \{u_1, \dots, u_n\}$ (**tabuleiro**) e família $\mathcal{F}$ de subconjuntos “**vencedores**” de U .

Vence quem conseguir um conjunto vencedor. Empate se ninguém conseguir.

Roubo de Estratégia: Argumento não-constructivo. Se Bob tivesse estratégia vencedora, Alice poderia escolher um elemento do tabuleiro e, a partir daí, seguir a estratégia de Bob como segundo jogador.

Teorema: Alice nunca perde jogos MAKER-MAKER.
Bob pode no máximo empatar.

Exemplo: Jogo da Velha n^2

Veremos que Alice vence o Jogo da Velha n^2 para $n \in \{1, 2\}$ e empata para $n \geq 3$.

Jogos MAKER-MAKER e MAKER-BREAKER

Instância: $(U, \mathcal{F})$, onde $U = \{u_1, \dots, u_n\}$ é o conjunto universo (**elementos do tabuleiro**) e $\mathcal{F}$ é uma família de subconjuntos de U (**conjuntos críticos, vencedores ou perdedores**).

Estratégia de Pareamento: cobrir o “*tabuleiro*” com pares disjuntos de elementos.

Exemplo: Jogo da Velha n^2 para $n \geq 5$.

Bob empata MAKER-MAKER e vence MAKER-BREAKER com a seguinte estratégia.

2	3	3	10	1
4	2	5	5	12
4	6	7	7	12
9	6	8	9	*
1	11	8	10	11

2	3	3	10	*	1
4	2	5	5	12	14
4	6	7	7	12	14
9	6	8	9	*	*
*	11	8	10	11	*
1	13	13	*	*	*

2	3	3	10	*	*	1
4	2	5	5	12	14	16
4	6	7	7	12	14	16
9	6	8	9	*	*	*
*	11	8	10	11	*	*
*	13	13	*	*	*	*
1	15	15	*	*	*	*

Estratégia de pareamento dos Jogos da Velha 5^2 , 6^2 e 7^2 . Os números indicam os pares. Não é necessário parear os elementos *. Note que toda fila possui dois elementos de algum par.

Jogos MAKER-MAKER e MAKER-BREAKER

Instância: $(U, \mathcal{F})$, onde $U = \{u_1, \dots, u_n\}$ é o conjunto universo (**elementos do tabuleiro**) e $\mathcal{F}$ é uma família de subconjuntos de U (**conjuntos críticos, vencedores ou perdedores**).

Estratégia de Pareamento: cobrir o “*tabuleiro*” com pares disjuntos de elementos.

Exemplo: Jogo da Velha n^2 para $n = 4$.

Bob empata MAKER-MAKER e vence MAKER-BREAKER com a seguinte estratégia.

$$\begin{bmatrix} \mathbf{A} & 1 & 1 & 7 \\ 2 & \mathbf{B} & 4 & 6 \\ 2 & 3 & 3 & 6 \\ 7 & 5 & 4 & 5 \end{bmatrix} \quad \begin{bmatrix} 1 & \mathbf{A} & 1 & 7 \\ 2 & \mathbf{B} & 4 & 6 \\ 2 & 3 & 3 & 6 \\ 7 & 5 & 4 & 5 \end{bmatrix} \quad \begin{bmatrix} 1 & 1 & 4 & 7 \\ 2 & \mathbf{B} & \mathbf{A} & 6 \\ 2 & 3 & 3 & 6 \\ 7 & 5 & 4 & 5 \end{bmatrix}$$

Três estratégias de pareamento do Jogo da Velha 4^2 , as quais dependem da primeira jogada A de Alice. A resposta de Bob a essa primeira jogada é indicada com B. Note que toda fila que não contém B possui dois elementos de algum par.

Jogos Posicionais com instâncias muito grandes

Jogo CLIQUE (n, k) : Alice e Bob coloreм arestas de K_n . Alice vence se obter um grafo completo com k vértices e todas as arestas coloridas com sua cor.

Veremos que, para $n = 2^{(33+10^{10})}$, esse valor é exatamente $k = 2 \cdot 10^{10} - 1$. Ou seja, Alice não consegue colorir as arestas de uma clique com 20 bilhões de vértices, mas consegue com qualquer número menor.

É surpreendente que esse problema seja tão difícil para instâncias pequenas, mas é possível calcular o valor exato para instâncias tão grandes.

Principal técnica usada: a Técnica do Potencial

Técnica do Potencial e os Soldados de Conway

Soldados de Conway (puzzle): jogada consiste num soldado pular sobre outro, que será abatido. Objetivo: atingir o alvo acima da linha inimiga.

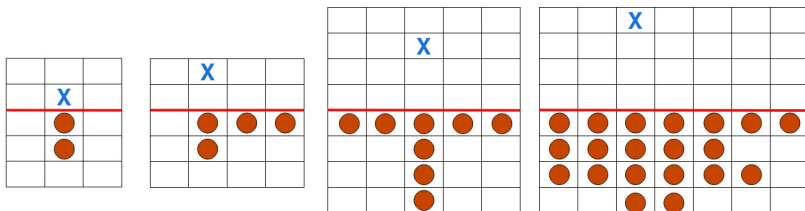


Figure: Solução dos Soldados de Conway para distâncias 1, 2, 3 e 4.

Teorema (Conway, 1961): Não existe sequência finita de movimentos que leve um soldado a uma distância 5 ou mais acima da linha horizontal.

Técnica do Potencial e os Soldados de Conway

Teorema (Conway, 1961): Não existe sequência finita de movimentos que leve um soldado a uma distância 5 ou mais acima da linha horizontal.

PROOF:

Seja X a célula alvo à distância 5 acima da linha. O potencial de uma célula é φ^d , onde $\varphi = (\sqrt{5} - 1)/2$ e d é a distância L^1 para X . Logo o potencial de X é $\varphi^0 = 1$. O potencial de uma posição é a soma dos potenciais de suas células ocupadas.

Como $\varphi^2 + \varphi = 1$, então $p(c_1) + p(c_2) \geq p(c_3)$ para células consecutivas c_1, c_2 e c_3 : **Nenhum movimento aumenta o potencial.** Caso extremo: $\varphi^{d+2} + \varphi^{d+1} = \varphi^d$.

Maior potencial possível: Linha 1

$$\varphi^5 + 2\varphi^6 + 2\varphi^7 + \dots = \varphi^5 + 2\frac{\varphi^6}{1-\varphi} = \varphi^5 + 2\frac{\varphi^6}{\varphi^2} = \varphi^5 + 2\varphi^4 = \varphi^3 + \varphi^4 = \varphi^2$$

Analogamente: Linha i : potencial máximo φ^{i+1} .

Somando os potenciais máximos por linha, temos potencial total máximo:

$$\varphi^2 + \varphi^3 + \varphi^4 + \dots = \frac{\varphi^2}{1-\varphi} = \frac{\varphi^2}{\varphi^2} = 1,$$

mesmo potencial da célula alvo X . Portanto, nenhum número finito de soldados é capaz de colocar um soldado à distância 5 ou mais acima da linha horizontal.

Erdős-Selfridge (1973) - MAKER-BREAKER (Bob vence)

Teorema (Erdős, Selfridge, 1973): Se Bob (Breaker) joga primeiro em MAKER-BREAKER $(U, \mathcal{F})$, ele vence se $\sum_{F \in \mathcal{F}} 2^{-|F|} < 1$.

PROOF: Potencial de conj vencedor $F \in \mathcal{F}$ é $2^{-|F|}$: prob Alice (Maker) pegar todo elemento de F num jogo aleatório (prob $1/2$ de cada elemento ser de Alice ou Bob). Potencial P_0 é a soma dos potenciais de seus conjuntos $P_0 = \sum_{F \in \mathcal{F}} 2^{-|F|}$: número esperado de conjuntos $F \in \mathcal{F}$ conquistados por Alice.

Bob começa selecionando x e Alice seleciona y . Remova de $\mathcal{F}$ todo conjunto F que contém x (conjuntos *mortos* para Alice). Remova y de todo conjunto F que contém y (o que importa para Alice são os elementos ainda não conquistados). Potencial de F zera ou dobra de $2^{-|F|}$ para $2^{-(|F|-1)}$: variação de $2^{-|F|}$.

$$P_1 = P_0 - \sum_{\substack{F \in \mathcal{F}_0: \\ x \in F}} 2^{-|F|} \quad \text{e} \quad P_2 = P_1 + \sum_{\substack{F \in \mathcal{F}_1: \\ y \in F}} 2^{-|F|} = P_0 - \sum_{\substack{F \in \mathcal{F}_0: \\ x \in F}} 2^{-|F|} + \sum_{\substack{F \in \mathcal{F}_1: \\ y \in F}} 2^{-|F|}$$

Tomando x tal que $\sum_{\substack{F \in \mathcal{F}_0: \\ x \in F}} 2^{-|F|}$ é máximo, temos que

$$P_2 \leq P_0 - \sum_{\substack{F \in \mathcal{F}_0: \\ x \in F}} 2^{-|F|} + \sum_{\substack{F \in \mathcal{F}_0: \\ y \in F}} 2^{-|F|} \leq P_0 - \sum_{\substack{F \in \mathcal{F}_0: \\ x \in F}} 2^{-|F|} + \sum_{\substack{F \in \mathcal{F}_0: \\ x \in F}} 2^{-|F|} = P_0.$$

Logo, $P_1 \leq P_0$ e $P_2 \leq P_0$. Se Alice vence, então obtém conjunto vencedor $F_i = \emptyset$ com potencial $P_i \geq 2^{-|F_i|} = 2^0 = 1$. Mas $P_0 < 1$ e potencial nunca aumenta: absurdo.

Erdős-Selfridge (1973) - MAKER-BREAKER (Bob vence)

Teorema (Erdős, Selfridge, 1973): Se Bob (Breaker) **não** joga primeiro em MAKER-BREAKER $(U, \mathcal{F})$, ele vence se $\sum_{F \in \mathcal{F}} 2^{-|F|} < \frac{1}{2}$.

Melhor, se:

$$\sum_{F \in \mathcal{F}} 2^{-|F|} < 1 - \max_{u \in U} \left\{ \sum_{\substack{F \in \mathcal{F}: \\ u \in F}} 2^{-|F|} \right\}$$

PROOF: No máximo, Alice consegue aumentar o potencial da instância no valor

$$\max_{u \in U} \left\{ \sum_{\substack{F \in \mathcal{F}: \\ u \in F}} (2^{-(|F|-1)} - 2^{-|F|}) \right\} = \max_{u \in U} \left\{ \sum_{\substack{F \in \mathcal{F}: \\ u \in F}} 2^{-|F|} \right\}.$$

$$\text{Portanto, se} \quad \sum_{F \in \mathcal{F}} 2^{-|F|} + \max_{u \in U} \left\{ \sum_{\substack{F \in \mathcal{F}: \\ u \in F}} 2^{-|F|} \right\} < 1,$$

então voltamos a situação anterior em que Bob começa. Além disso,

$$\max_{u \in U} \left\{ \sum_{\substack{F \in \mathcal{F}: \\ u \in F}} 2^{-|F|} \right\} \leq \sum_{F \in \mathcal{F}} 2^{-|F|}$$

e portanto Bob vence se $\sum_{F \in \mathcal{F}} 2^{-|F|} < 1/2$.

Erdős-Selfridge (1973) - MAKER-BREAKER (Bob vence)

Teorema (Erdős, Selfridge, 1973): Jogo MAKER-BREAKER $(U, \mathcal{F})$ com $\mathcal{F}$ k -uniforme. Se Bob joga primeiro, ele vence se $|\mathcal{F}| < 2^k$. **Caso contrário**, ele vence se $|\mathcal{F}| < 2^{k-1}$ ou $|\mathcal{F}| + \Delta_1(\mathcal{F}) < 2^k$, onde $d(u)$ é o número de conjuntos $F \in \mathcal{F}$ que contém u e $\Delta_1(\mathcal{F}) = \max_{u \in U} \{d(u)\}$.

PROOF:

$$\sum_{F \in \mathcal{F}} 2^{-|F|} = |\mathcal{F}| \cdot 2^{-k} < 1$$
$$\sum_{F \in \mathcal{F}} 2^{-|F|} + \max_{u \in U} \left\{ \sum_{\substack{F \in \mathcal{F}: \\ u \in F}} 2^{-|F|} \right\} = (|\mathcal{F}| + \Delta_1(\mathcal{F})) \cdot 2^{-k} < 1$$

Valores apertados. Exemplo: $U = \{a_1, b_1, \dots, a_k, b_k\}$ tem $|U| = 2k$ elementos e $\mathcal{F}$ contém todos os conjuntos $\{c_1, \dots, c_k\}$ para $c_i \in \{a_i, b_i\}$. Então $|\mathcal{F}| = 2^k$ e Alice vence mesmo se Bob inicia, usando a seguinte estratégia de pareamento: se Bob escolhe a_i (resp. b_i), Alice escolhe b_i (resp. a_i). No caso em que Alice começa, podemos obter uma instância $\mathcal{F}'$ apertada, adicionando um elemento novo a_0 em cada conjunto $F \in \mathcal{F}$ e fazendo Alice selecionar a_0 em sua primeira jogada (note que $\mathcal{F}'$ é $(k+1)$ -uniforme e $\Delta_1(\mathcal{F}') = |\mathcal{F}'| = |\mathcal{F}| = 2^k$).

Erdős-Selfridge (1973) - MAKER-MAKER (Empate)

Teorema (Erdős, Selfridge, 1973): Bob consegue empate no jogo MAKER-MAKER nas mesmas condições dos teoremas anteriores.

Exemplo: Jogo da Velha n^2 para $n = 4$.

Bob empata MAKER-MAKER e vence MAKER-BREAKER com a seguinte estratégia.

$$\begin{bmatrix} \mathbf{A} & 1 & 1 & 7 \\ 2 & \mathbf{B} & 4 & 6 \\ 2 & 3 & 3 & 6 \\ 7 & 5 & 4 & 5 \end{bmatrix} \quad \begin{bmatrix} 1 & \mathbf{A} & 1 & 7 \\ 2 & \mathbf{B} & 4 & 6 \\ 2 & 3 & 3 & 6 \\ 7 & 5 & 4 & 5 \end{bmatrix} \quad \begin{bmatrix} 1 & 1 & 4 & 7 \\ 2 & \mathbf{B} & \mathbf{A} & 6 \\ 2 & 3 & 3 & 6 \\ 7 & 5 & 4 & 5 \end{bmatrix}$$

Erdős-Selfridge: Bob consegue empate no Jogo da Velha 4^2 .

$\mathcal{F} = 4$ linhas, 4 colunas e 2 diagonais. $\mathcal{F}$ é $k = 4$ -uniforme.

$$|U| = 4^2 = 16. \quad \Delta_1(\mathcal{F}) = 3$$

$$\Rightarrow |\mathcal{F}| + \Delta_1(\mathcal{F}) = 10 + 3 = 13 < 16 = 2^4 = 2^k.$$

Beck (1981) - MAKER-BREAKER (Alice vence)

Teorema (Beck, 1981): Seja $(U, \mathcal{F})$ instância de MAKER-BREAKER. Então Alice (jogando primeiro como Maker) vence se

$$\sum_{F \in \mathcal{F}} 2^{-|F|} > \frac{1}{8} \cdot \Delta_2(\mathcal{F}) \cdot |U|,$$

onde $\Delta_2(\mathcal{F}) = \max\{d_2(u, v)\}$ para $u \neq v \in U$ e $d_2(u, v)$ é o número de conjuntos $F \in \mathcal{F}$ que contém u e v . Portanto, se $\mathcal{F}$ é k -uniforme, então Alice vence se $|\mathcal{F}| > 2^{k-3} \cdot \Delta_2(\mathcal{F}) \cdot |U|$.

Beck (1982) - MAKER-BREAKER tendencioso

Instância $(U, \mathcal{F}, a, b)$: Alice seleciona a elementos de U e Bob seleciona b elementos de U .

Teorema (Beck, 1982): Bob vence. Sejam a e b dois inteiros e $(U, \mathcal{F})$ um jogo MAKER-BREAKER (a, b) -tendencioso. Se Bob (Breaker) joga primeiro, então ele vence se $\sum_{F \in \mathcal{F}} (1+b)^{-|F|/a} < 1$. Caso contrário, ele vence se

$$\sum_{F \in \mathcal{F}} (1+b)^{-|F|/a} < \frac{1}{1+b}$$

Quando $\mathcal{F}$ é k -uniforme, Bob vence sempre que $|\mathcal{F}| < (1+b)^{k/a}$ se Bob joga primeiro ou, caso contrário, sempre que $|\mathcal{F}| < (1+b)^{\frac{k}{a}-1}$.

Teorema (Beck, 1982): Alice vence. Seja $(U, \mathcal{F})$ um jogo MAKER-BREAKER (a, b) -tendencioso. Então Alice (jogando primeiro como Maker) vence se

$$\sum_{F \in \mathcal{F}} \left(1 + \frac{b}{a}\right)^{-|F|} > \frac{a^2 b^2}{(a+b)^3} \cdot \Delta_2(\mathcal{F}) \cdot |U|,$$

onde $\Delta_2(\mathcal{F}) = \max d_2(u, v)$ para $u \neq v \in U$ e $d_2(u, v)$ é o número de conjuntos $F \in \mathcal{F}$ que contém u e v . Consequentemente, se $\mathcal{F}$ é k -uniforme, então Alice vence se $|\mathcal{F}| > a^{2-k} \cdot b^2 \cdot (a+b)^{k-3} \cdot \Delta_2(\mathcal{F}) \cdot |U|$.

Hales-Jewett (1963) - Jogo da Velha n^d

Jogo da Velha n^d : MAKER-BREAKER com tabuleiro no grid cartesiano d -dimensional $[n]^d$ e conjuntos vencedores nas “linhas” de tamanho d .

Corolário (Erdős-Selfridge, 1973):

$\forall d, \exists n$: Bob empata o Jogo da Velha n^d ;

Teorema (Hales, Jewett, 1963):

$\forall n, \exists d$: Alice vence o Jogo da Velha n^d .

Teorema de van der Waerden (1927) e o Jogo da P.A.

Número de van der Waerden $W(k)$: menor n tal que qualquer coloração dos inteiros em $[n]$ com 2 cores gera P.A. monocromática de tamanho k .

Teorema (van der Waerden, 1927): $W(k)$ existe para todo inteiro k .

Teorema (Z. Szabó 1990; Gowers 2001):

$$W(k) < 2^{2^{2^{2^k+9}}}$$

Ademais, para todo $\varepsilon > 0$ e k suficientemente grande,

$$W(k) > \frac{2^k}{k^\varepsilon}$$

Teorema de van der Waerden (1927) e o Jogo da P.A.

Jogo da P.A. (n, k) : MAKER-BREAKER com tabuleiro em $[n]$ e conjuntos vencedores nas P.A.'s de tamanho k em $[n]$.

Número jogo de van der Waerden $W_g(k)$:
menor n tal que Alice vence o Jogo da P.A. (n, k) .

$$W_g(k) \leq W(k).$$

Exemplo: $W_g(3) = 5$ e $W_g(4) = 13$, mas $W(3) = 9$ e $W(4) = 35$.

Teorema (Beck, 1981): $W_g(k) = 2^{k \cdot (1+o(1))}$. Mais especificamente,

$$2^{k \cdot (1-k^{-0.1})} < W_g(k) < 2^k \cdot k^3/8$$

Teorema de van der Waerden (1927) e o Jogo da P.A.

1 2 3 4 5 6 7 8 9 10 11 12 13
 B_1 B_2 A_1 B_2 A_2 A_3 A_4

1 2 3 4 5 6 7 8 9 10 11 12 13
 B_1 B_2 A_4 A_1 A_3 A_2 B_3

1 2 3 4 5 6 7 8 9 10 11 12 13
 B_3 A_3 B_1 A_1 B_2 A_2

1 2 3 4 5 6 7 8 9 10 11 12 13
 B_2 B_1 A_1 B_2 A_2 A_2 A_4 B_4 A_5

1 2 3 4 5 6 7 8 9 10 11 12 13
 B_4 B_1 A_4 A_1 A_3 B_2 A_2 A_4 B_2

1 2 3 4 5 6 7 8 9 10 11 12 13
 A_4 A_3 B_1 A_1 B_2 A_2 B_3

1 2 3 4 5 6 7 8 9 10 11 12 13
 A_5 B_2 A_4 B_1 A_1 B_2 A_3 A_2 B_4

1 2 3 4 5 6 7 8 9 10 11 12 13
 B_3 A_2 A_1 B_1 B_2 A_3

1 2 3 4 5 6 7 8 9 10 11 12 13
 B_4 A_4 A_2 A_3 B_1 A_1 B_1 A_2 B_2

1 2 3 4 5 6 7 8 9 10 11 12 13
 A_4 A_2 A_1 B_2 B_1 A_3 B_3

1 2 3 4 5 6 7 8 9 10 11 12 13
 B_2 B_4 A_2 B_2 A_3 A_1 A_4 B_1 A_2

1 2 3 4 5 6 7 8 9 10 11 12 13
 A_5 B_4 A_4 A_2 A_3 B_2 A_1 B_1 B_2

1 2 3 4 5 6 7 8 9 10 11 12 13
 B_3 A_2 A_3 A_1 A_4 B_2 B_1

1 2 3 4 5 6 7 8 9 10 11 12 13
 B_3 A_1 A_2 B_2 A_1 A_4 B_1

Teorema de Ramsey (1928) e o Jogo da Clique

Número de Ramsey $R(k)$: menor n tal que qualquer coloração das arestas do grafo completo K_n com 2 cores gera uma clique monocromática com k vértices.

Teorema (Ramsey 1928): $R(k)$ existe para todo inteiro k .

Teorema (Erdős 1947; Erdős, Szekeres 1935):

$$2^{k/2} < R(k) < 4^{k-1}$$

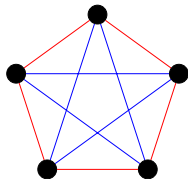
Teorema de Ramsey (1928) e o Jogo da Clique

Jogo da Clique (n, k) : MAKER-BREAKER com tabuleiro nas arestas do grafo completo K_n e conjuntos vencedores nas cliques com k vértices.

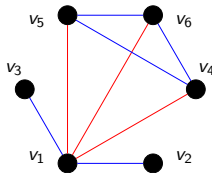
Número jogo de Ramsey $R_g(k)$:

menor n tal que Alice vence o Jogo da Clique (n, k) .

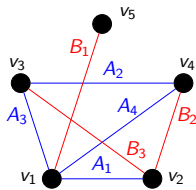
$R_g(k) \leq R(k)$. **Exemplo:** $R_g(3) = 5$, mas $R(3) = 6$.



(a) $R(3) > 5$



(b) $R(3) \leq 6$



(c) $R_g(3) \leq 5$

Número de Ramsey $R(3) = 6$ e Número Jogo de Ramsey $R_g(3) = 5$

Teorema de Ramsey (1928) e o Jogo da Clique

Jogo da Clique (n, k) : MAKER-BREAKER com tabuleiro nas arestas do grafo completo K_n e conjuntos vencedores nas cliques com k vértices.

Número jogo de Ramsey $R_g(k)$:

menor n tal que Alice vence o Jogo da Clique (n, k) .

$R_g(k) \leq R(k)$. **Exemplo:** $R_g(3) = 5$, mas $R(3) = 6$.

Teorema (Beck 2002): No jogo da Clique (n, k) , Alice (Maker) vence se

$$k \leq 2 \log_2 n - 2 \log_2 \log_2 n + 2 \log_2 e - 10/3 + o(1)$$

e Bob (Breaker) vence se

$$k \geq 2 \log_2 n - 2 \log_2 \log_2 n + 2 \log_2 e - 1 + o(1),$$

onde $o(1)$ se torna desprezível para $n \geq 2^{10^{10}}$.

Teorema de Ramsey (1928) e o Jogo da Clique

De acordo com [Beck, 2008] sobre o Jogo da Clique (n, k) , se $n \geq 2^{10^{10}}$ e $k' = 2 \log_2 n - 2 \log_2 \log_2 n + 2 \log_2 e - 3$ não é muito próximo de um inteiro (o que ocorre para grande maioria dos números), então Alice vence com $k = \lfloor k' \rfloor$ e perde com $k = \lceil k' \rceil$. Como exemplo, tomando $n = 2^{(33+10^{10})}$, temos que $k = 2 \cdot 10^{10} - 1$ e, com isso, Alice não consegue obter uma clique com 20 bilhões de vértices, mas consegue com qualquer número menor. Esse nível de precisão é impressionante, considerando que pouco se sabe sobre valores pequenos de n .

Teorema (Beck 2002): No jogo da Clique (n, k) , Alice (Maker) vence se

$$k \leq 2 \log_2 n - 2 \log_2 \log_2 n + 2 \log_2 e - 10/3 + o(1)$$

e Bob (Breaker) vence se

$$k \geq 2 \log_2 n - 2 \log_2 \log_2 n + 2 \log_2 e - 1 + o(1),$$

onde $o(1)$ se torna desprezível para $n \geq 2^{10^{10}}$.

Jogos Posicionais: Complexidade Computacional

MAKER-BREAKER- k : conjuntos vencedores com tamanho no máximo k .

[Schaefer, 1978]: **MAKER-BREAKER-11** é **PSPACE-completo**.

[Reisch, 1981]: Hex com configuração dada é **PSPACE-completo**.

[Bonnet et al., 2017]: Hex generalizado é **$W[1]$ -completo** na profundidade do jogo (Alice quer encontrar um $s - t$ caminho e Bob deseja impedir).

[Rahman e Watson, 2023]:

MAKER-BREAKER-6 e **MAKER-MAKER-7** são **PSPACE-completos**.

[Galliot, 2023]: **MAKER-BREAKER-3** é polinomial.

Problema em aberto: **MAKER-BREAKER-4** e **MAKER-BREAKER-5** ?

Jogos Posicionais sobre arestas do grafo

Jogos sobre as arestas do grafo: Jogo da Clique, Jogo da Conectividade, Jogo Hamiltoniano, Jogo do Emparelhamento Perfeito, Jogo do Grafo H .

Alice e Bob selecionam arestas do grafo. Alice deseja obter uma clique de tamanho k , ou um subgrafo gerador conexo, ou um caminho hamiltoniano, ou um emparelhamento perfeito, ou um subgrafo induzido isomorfo a H . Bob deseja impedir (MAKER-BREAKER) ou deseja o mesmo (MAKER-MAKER).

Jogo da Clique (n, k) : $G = K_n$ e $H = K_k$.

[Beck, 2002] Alice vence se $k \leq \frac{1}{2} \log_2 n$ e perde se $k \geq 2 \log_2 n$.

Jogo da Conectividade: [Lehman, 1964] Alice vence como 2º jogadora se e só se o grafo tem 2 árvores geradoras disjuntas em arestas.

Jogo do Emparelhamento Perfeito: PSPACE-completo [Duchêne et al., 2025]

Jogo do Grafo H : Polinomial se H tem ≤ 3 arestas [Galliot, 2023]
PSPACE-completo para H sendo um grafo com 51 vértices ou uma árvore com 91 vértices [Duchêne et al., 2025].

Mínimo de vértices em H tal que é PSPACE-completo?

Jogos Posicionais sobre vértices do grafo

Jogos sobre os vértices do grafo: Jogo MAKER-BREAKER de Dominação, Jogo do Maior Subgrafo Conexo.

Alice e Bob selecionam vértices do grafo. Alice deseja obter um conjunto dominante ou deseja obter um subgrafo conexo maior que o de Bob.

MAKER-BREAKER de Dominação:

[Duchêne et al., 2020] Alice **vence*** numa árvore T independente se joga primeiro ou não **se e só se** T tem um emparelhamento perfeito.

[Bagan et al., 2024] Alice **vence*** em todo grafo regular. Decidir o vencedor é polinomial em grafo periplanar e grafo de blocos.

* **vence independente se joga primeiro ou não.**

Jogo do Maior Subgrafo Conexo: Pontuação (Scoring Positional Game)
Roubo de estratégia: Alice nunca perde.

[Bensmail et al., 2022] PSPACE-completo em grafos bipartidos com diâmetro 5. Polinomial em caminhos, ciclos e cografos.

Problema em aberto: Complexidade em árvores.

FIM DA AULA 02

FIM DA AULA 02

Perguntas (se houver tempo) ??

Teoria dos Jogos Combinatórios em Grafos

Aula 03

**Samuel Araújo, Nicolas Martins,
Nicolas Nisse, Rudini Sampaio**

CBM-2025, IMPA,
Rio de Janeiro, quarta, 30-Julho, 8h

Capítulo 5

Complexidade Computacional de Jogos

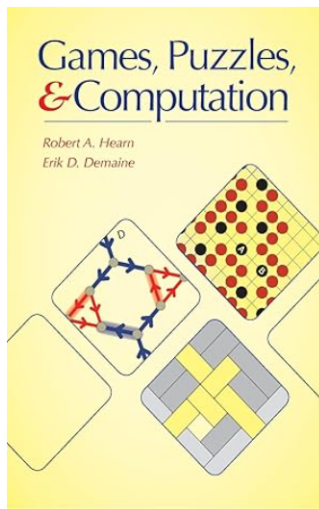
Capítulo 7

Jogos de Coloração em Grafos

Capítulo 5

Complexidade Computacional de Jogos

Livros sobre Complexidade de Jogos: 2009



Principais Classes de Problemas de Decisão

- ▶ Classe P : Problemas que podem ser **decididos** em **tempo polinomial**;
- ▶ Classe NP : Problemas cuja resposta **SIM** pode ser **verificada** em **tempo polinomial** mediante apresentação de um **certificado**;
- ▶ Classe $PSPACE$: Problemas que podem ser **decididos** usando **espaço polinomial**;
- ▶ Classe $EXPTIME$: Problemas que podem ser **decididos** em **tempo exponencial**;

Principais Classes de Problemas de Decisão

- ▶ Classe P : Problemas que podem ser **decididos** em **tempo polinomial**;
- ▶ Classe NP : Problemas cuja resposta **SIM** pode ser **verificada** em **tempo polinomial** mediante apresentação de um **certificado**;
- ▶ Classe $PSPACE$: Problemas que podem ser **decididos** usando **espaço polinomial**;
- ▶ Classe $EXPTIME$: Problemas que podem ser **decididos** em **tempo exponencial**;

Principais Classes de Problemas de Decisão

- ▶ Classe P : Problemas que podem ser **decididos** em **tempo polinomial**;
- ▶ Classe NP : Problemas cuja resposta **SIM** pode ser **verificada** em **tempo polinomial** mediante apresentação de um **certificado**;
- ▶ Classe $PSPACE$: Problemas que podem ser **decididos** usando **espaço polinomial**;
- ▶ Classe $EXPTIME$: Problemas que podem ser **decididos** em **tempo exponencial**;

Principais Classes de Problemas de Decisão

- ▶ Classe P : Problemas que podem ser **decididos** em **tempo polinomial**;
- ▶ Classe NP : Problemas cuja resposta **SIM** pode ser **verificada** em **tempo polinomial** mediante apresentação de um **certificado**;
- ▶ Classe $PSPACE$: Problemas que podem ser **decididos** usando **espaço polinomial**;
- ▶ Classe $EXPTIME$: Problemas que podem ser **decididos** em **tempo exponencial**;

Principais Classes de Problemas de Decisão

- ▶ Classe P : Problemas que podem ser **decididos** em **tempo polinomial**;
- ▶ Classe NP : Problemas cuja resposta **SIM** pode ser **verificada** em **tempo polinomial** mediante apresentação de um **certificado**;
- ▶ Classe $PSPACE$: Problemas que podem ser **decididos** usando **espaço polinomial**;
- ▶ Classe $EXPTIME$: Problemas que podem ser **decididos** em **tempo exponencial**;

Principais Classes de Problemas de Decisão

- ▶ Classe *P*: Problemas que podem ser **decididos** em **tempo polinomial**;
- ▶ Classe *NP*: Problemas cuja resposta **SIM** pode ser **verificada** em **tempo polinomial** mediante apresentação de um **certificado**;
- ▶ Classe *PSPACE*: Problemas que podem ser **decididos** usando **espaço polinomial**;
- ▶ Classe *EXPTIME*: Problemas que podem ser **decididos** em **tempo exponencial**;

$$P \subseteq NP \subseteq PSPACE \subseteq EXPTIME$$

Jogos em PSPACE e EXPTIME

- **Jogo Limitado:** Se o número máximo de jogadas a partir de uma configuração $\mathcal{X}$ é limitado por um polinômio de $|\mathcal{X}|$;

Teorema: Se J é um **jogo combinatório limitado**, então $J \in \text{PSPACE}$.

Teorema: Se J possui um **número exponencial de configurações possíveis**, então $J \in \text{EXPTIME}$.

Jogos em PSPACE e EXPTIME

- **Jogo Limitado:** Se o número máximo de jogadas a partir de uma configuração $\mathcal{X}$ é limitado por um polinômio de $|\mathcal{X}|$;

Teorema: Se J é um **jogo combinatório limitado**, então $J \in \text{PSPACE}$.

Teorema: Se J possui um **número exponencial de configurações possíveis**, então $J \in \text{EXPTIME}$.

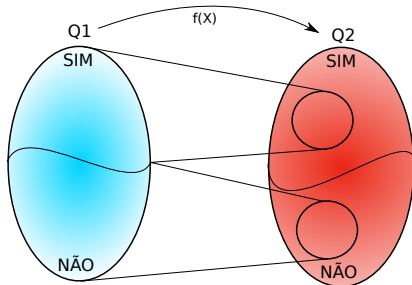
Jogos em PSPACE e EXPTIME

- **Jogo Limitado**: Se o número máximo de jogadas a partir de uma configuração $\mathcal{X}$ é limitado por um polinômio de $|\mathcal{X}|$;

Teorema: Se J é um **jogo combinatório limitado**, então $J \in \text{PSPACE}$.

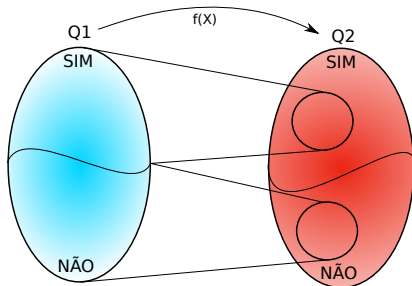
Teorema: Se J possui um **número exponencial de configurações possíveis**, então $J \in \text{EXPTIME}$.

Redução Polinomial e Problemas Completos



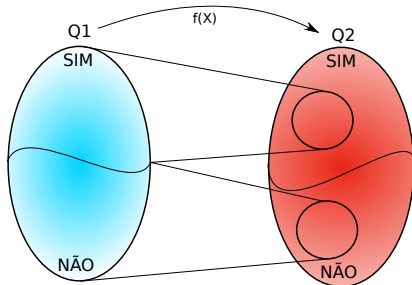
- ▶ $Q1$ é redutível a $Q2$: $Q1 \preceq_p Q2$;
- ▶ Se $Q2 \in P$ então $Q1 \in P$;
- ▶ Um problema Q é dito $\mathcal{C}$ -difícil, se para todo $Q' \in \mathcal{C}$, $Q' \preceq_p Q$. Ademais se $Q \in \mathcal{C}$, Q é dito $\mathcal{C}$ -completo;
- ▶ Se Q é $\mathcal{C}$ -difícil e $Q \in P$ então $\mathcal{C} \subseteq P$;
- ▶ Se $Q1$ é $\mathcal{C}$ -difícil e $Q1 \preceq_p Q2$ então $Q2$ é $\mathcal{C}$ -difícil.

Redução Polinomial e Problemas Completos



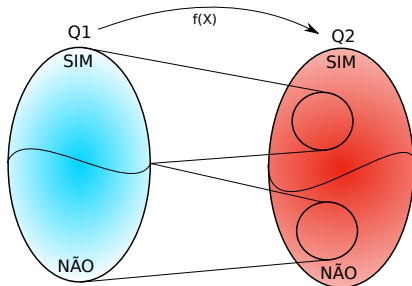
- ▶ $Q1$ é redutível a $Q2$: $Q1 \preceq_p Q2$;
- ▶ Se $Q2 \in P$ então $Q1 \in P$;
- ▶ Um problema Q é dito $\mathcal{C}$ -difícil, se para todo $Q' \in \mathcal{C}$, $Q' \preceq_p Q$. Ademais se $Q \in \mathcal{C}$, Q é dito $\mathcal{C}$ -completo;
- ▶ Se Q é $\mathcal{C}$ -difícil e $Q \in P$ então $\mathcal{C} \subseteq P$;
- ▶ Se $Q1$ é $\mathcal{C}$ -difícil e $Q1 \preceq_p Q2$ então $Q2$ é $\mathcal{C}$ -difícil.

Redução Polinomial e Problemas Completos



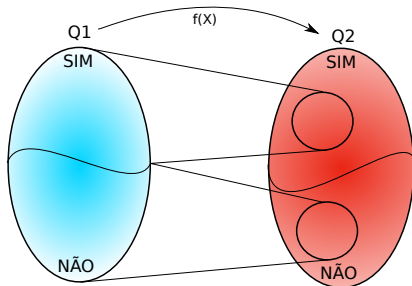
- ▶ $Q1$ é redutível à $Q2$: $Q1 \preceq_p Q2$;
- ▶ Se $Q2 \in P$ então $Q1 \in P$;
- ▶ Um problema Q é dito $\mathcal{C}$ -difícil, se para todo $Q' \in \mathcal{C}$, $Q' \preceq_p Q$. Ademais se $Q \in \mathcal{C}$, Q é dito $\mathcal{C}$ -completo;
- ▶ Se Q é $\mathcal{C}$ -difícil e $Q \in P$ então $\mathcal{C} \subseteq P$;
- ▶ Se $Q1$ é $\mathcal{C}$ -difícil e $Q1 \preceq_p Q2$ então $Q2$ é $\mathcal{C}$ -difícil.

Redução Polinomial e Problemas Completos



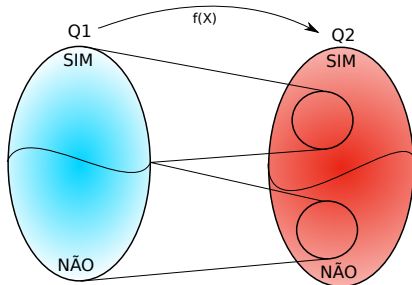
- ▶ $Q1$ é redutível à $Q2$: $Q1 \preceq_p Q2$;
- ▶ Se $Q2 \in P$ então $Q1 \in P$;
- ▶ Um problema Q é dito $\mathcal{C}$ -difícil, se para todo $Q' \in \mathcal{C}$, $Q' \preceq_p Q$. Ademais se $Q \in \mathcal{C}$, Q é dito $\mathcal{C}$ -completo;
- ▶ Se Q é $\mathcal{C}$ -difícil e $Q \in P$ então $\mathcal{C} \subseteq P$;
- ▶ Se $Q1$ é $\mathcal{C}$ -difícil e $Q1 \preceq_p Q2$ então $Q2$ é $\mathcal{C}$ -difícil.

Redução Polinomial e Problemas Completos



- ▶ $Q1$ é redutível à $Q2$: $Q1 \preceq_p Q2$;
- ▶ Se $Q2 \in P$ então $Q1 \in P$;
- ▶ Um problema Q é dito $\mathcal{C}$ -difícil, se para todo $Q' \in \mathcal{C}$, $Q' \preceq_p Q$. Ademais se $Q \in \mathcal{C}$, Q é dito $\mathcal{C}$ -completo;
- ▶ Se Q é $\mathcal{C}$ -difícil e $Q \in P$ então $\mathcal{C} \subseteq P$;
- ▶ Se $Q1$ é $\mathcal{C}$ -difícil e $Q1 \preceq_p Q2$ então $Q2$ é $\mathcal{C}$ -difícil.

Redução Polinomial e Problemas Completos



- ▶ Q_1 é redutível à Q_2 : $Q_1 \preceq_p Q_2$;
- ▶ Se $Q_2 \in P$ então $Q_1 \in P$;
- ▶ Um problema Q é dito $\mathcal{C}$ -difícil, se para todo $Q' \in \mathcal{C}$, $Q' \preceq_p Q$. Ademais se $Q \in \mathcal{C}$, Q é dito $\mathcal{C}$ -completo;
- ▶ Se Q é $\mathcal{C}$ -difícil e $Q \in P$ então $\mathcal{C} \subseteq P$;
- ▶ Se Q_1 é $\mathcal{C}$ -difícil e $Q_1 \preceq_p Q_2$ então Q_2 é $\mathcal{C}$ -difícil.

SAT Quantificado

$$\Psi = Q_1x_1Q_2x_2\ldots Q_nx_n\Phi$$

- ▶ **Entrada:** Fórmula lógica quantificada ($Q_i \in \{\exists, \forall\}$) Ψ com n variáveis;
- ▶ **Pergunta:** Ψ é satisfatível?
- ▶ Podemos assumir que $Q_i = \exists$ se i é ímpar e $Q_i = \forall$ caso contrário;
- ▶ Podemos assumir que Φ está na Forma Normal Conjuntiva;
- ▶ QSAT foi o primeiro problema demonstrado ser PSPACE-completo (Stockmeyer e Meyer, 1973).

SAT Quantificado

$$\Psi = Q_1x_1Q_2x_2\ldots Q_nx_n\Phi$$

- ▶ **Entrada:** Fórmula lógica quantificada ($Q_i \in \{\exists, \forall\}$) Ψ com n variáveis;
- ▶ **Pergunta:** Ψ é satisfatível?
- ▶ Podemos assumir que $Q_i = \exists$ se i é ímpar e $Q_i = \forall$ caso contrário;
- ▶ Podemos assumir que Φ está na Forma Normal Conjuntiva;
- ▶ QSAT foi o primeiro problema demonstrado ser PSPACE-completo (Stockmeyer e Meyer, 1973).

SAT Quantificado

$$\Psi = Q_1x_1 Q_2x_2 \dots Q_nx_n \Phi$$

- ▶ **Entrada:** Fórmula lógica quantificada ($Q_i \in \{\exists, \forall\}$) Ψ com n variáveis;
- ▶ **Pergunta:** Ψ é satisfatível?
- ▶ Podemos assumir que $Q_i = \exists$ se i é ímpar e $Q_i = \forall$ caso contrário;
- ▶ Podemos assumir que Φ está na Forma Normal Conjuntiva;
- ▶ QSAT foi o primeiro problema demonstrado ser PSPACE-completo (Stockmeyer e Meyer, 1973).

SAT Quantificado

$$\Psi = Q_1x_1 Q_2x_2 \dots Q_nx_n \Phi$$

- ▶ **Entrada:** Fórmula lógica quantificada ($Q_i \in \{\exists, \forall\}$) Ψ com n variáveis;
- ▶ **Pergunta:** Ψ é satisfatível?
- ▶ Podemos assumir que $Q_i = \exists$ se i é ímpar e $Q_i = \forall$ caso contrário;
- ▶ Podemos assumir que Φ está na Forma Normal Conjuntiva;
- ▶ QSAT foi o primeiro problema demonstrado ser PSPACE-completo (Stockmeyer e Meyer, 1973).

SAT Quantificado

$$\Psi = Q_1x_1 Q_2x_2 \dots Q_nx_n \Phi$$

- ▶ **Entrada:** Fórmula lógica quantificada ($Q_i \in \{\exists, \forall\}$) Ψ com n variáveis;
- ▶ **Pergunta:** Ψ é satisfatível?
- ▶ Podemos assumir que $Q_i = \exists$ se i é ímpar e $Q_i = \forall$ caso contrário;
- ▶ Podemos assumir que Φ está na Forma Normal Conjuntiva;
- ▶ QSAT foi o primeiro problema demonstrado ser PSPACE-completo (Stockmeyer e Meyer, 1973).

SAT Quantificado

$$\Phi = \exists x_1 \forall x_2 \exists x_3 \forall x_4 : (\overline{x_1} \vee \overline{x_3}) \wedge (\overline{x_2} \vee x_3) \wedge (x_4 \vee \overline{x_3}) \wedge (x_1 \vee x_2)$$

- ▶ **Entrada:** Fórmula lógica quantificada ($Q_i \in \{\exists, \forall\}$) Ψ com n variáveis;
- ▶ **Pergunta:** Ψ é satisfatível?
- ▶ Podemos assumir que $Q_i = \exists$ se i é ímpar e $Q_i = \forall$ caso contrário;
- ▶ Podemos assumir que Φ está na Forma Normal Conjuntiva;
- ▶ QSAT foi o primeiro problema demonstrado ser PSPACE-completo (Stockmeyer e Meyer, 1973).

Jogo GEOGRAFIA

- ▶ Um professor divide os alunos em dois grupos: A e B ;
- ▶ O professor nomeia 1 país;
- ▶ Os grupos se alternam e nomeiam um país cuja inicial é igual a última letra do país nomeado mais recentemente;
- ▶ O primeiro grupo que não puder nomear um país ainda não mencionado perde (variante normal).

Jogo GEOGRAFIA

- ▶ Um professor divide os alunos em dois grupos: A e B ;
- ▶ O professor nomeia 1 país;
- ▶ Os grupos se alternam e nomeiam um país cuja inicial é igual a última letra do país nomeado mais recentemente;
- ▶ O primeiro grupo que não puder nomear um país ainda não mencionado perde (variante normal).

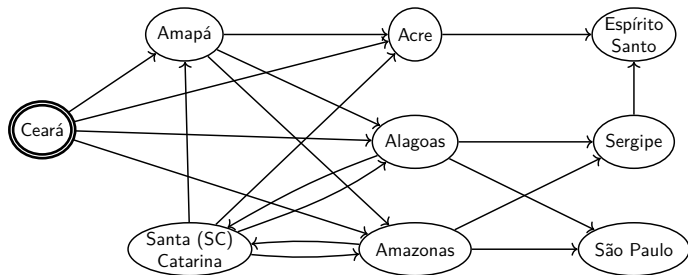
Jogo GEOGRAFIA

- ▶ Um professor divide os alunos em dois grupos: A e B ;
- ▶ O professor nomeia 1 país;
- ▶ Os grupos se alternam e nomeiam um país cuja inicial é igual a última letra do país nomeado mais recentemente;
- ▶ O primeiro grupo que não puder nomear um país ainda não mencionado perde (variante normal).

Jogo GEOGRAFIA

- ▶ Um professor divide os alunos em dois grupos: A e B ;
- ▶ O professor nomeia 1 país;
- ▶ Os grupos se alternam e nomeiam um país cuja inicial é igual a última letra do país nomeado mais recentemente;
- ▶ O primeiro grupo que não puder nomear um país ainda não mencionado perde (variante normal).

Jogo GEOGRAFIA



- ▶ Um professor divide os alunos em dois grupos: A e B ;
- ▶ O professor nomeia 1 país;
- ▶ Os grupos se alternam e nomeiam um país cuja inicial é igual a última letra do país nomeado mais recentemente;
- ▶ O primeiro grupo que não puder nomear um país ainda não mencionado perde (variante normal).

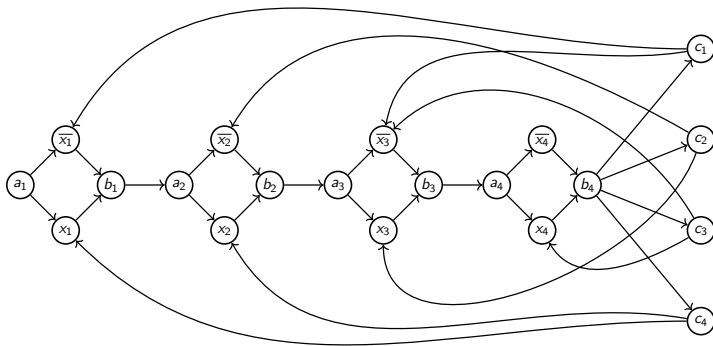
QSAT $\preceq_p$ GEOGRAFIA

- ▶ Assumimos que entrada de QSAT possui uma quantidade par de variáveis;
- ▶ $\Phi = \exists x_1 \forall x_2 \exists x_3 \forall x_4 : (\overline{x_1} \vee \overline{x_3}) \wedge (\overline{x_2} \vee x_3) \wedge (x_4 \vee \overline{x_3}) \wedge (x_1 \vee x_2)$
- ▶ A partir de Φ podemos construir o seguinte digrafo D_Φ ;

QSAT $\preceq_p$ GEOGRAFIA

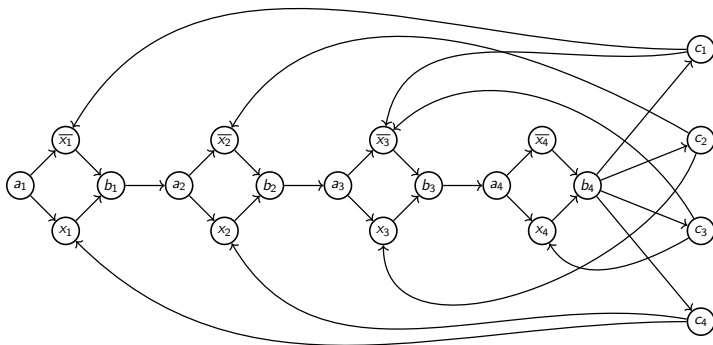
- ▶ Assumimos que entrada de QSAT possui uma quantidade par de variáveis;
- ▶ $\Phi = \exists x_1 \forall x_2 \exists x_3 \forall x_4 : (\overline{x_1} \vee \overline{x_3}) \wedge (\overline{x_2} \vee x_3) \wedge (x_4 \vee \overline{x_3}) \wedge (x_1 \vee x_2)$
- ▶ A partir de Φ podemos construir o seguinte digrafo D_Φ ;

QSAT $\preceq_p$ GEOGRAFIA



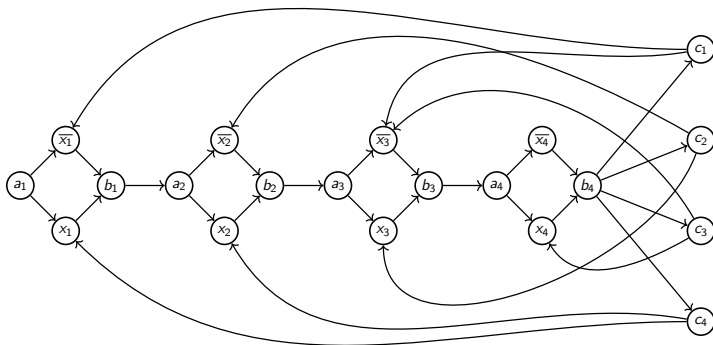
- ▶ Assumimos que entrada de QSAT possui uma quantidade par de variáveis;
- ▶ $\Phi = \exists x_1 \forall x_2 \exists x_3 \forall x_4 : (\overline{x_1} \vee \overline{x_3}) \wedge (\overline{x_2} \vee x_3) \wedge (x_4 \vee \overline{x_3}) \wedge (x_1 \vee x_2)$
- ▶ A partir de Φ podemos construir o seguinte digrafo D_Φ ;

QSAT $\preceq_p$ GEOGRAFIA



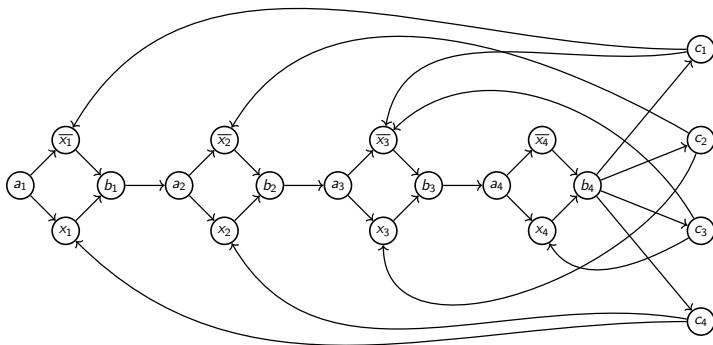
- ▶ Alice escolherá entre x_i e $\bar{x}_i$ se i for ímpar enquanto Bob escolherá se i for par;
- ▶ Se Alice possuir uma estratégia vencedora para a instância de QSAT, então ela joga assumindo que as literais não escolhidas são verdadeiras.

QSAT $\preceq_p$ GEOGRAFIA



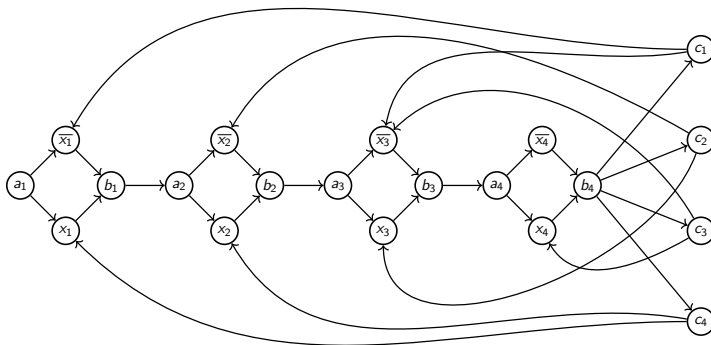
- ▶ Alice escolherá entre x_i e $\overline{x_i}$ se i for ímpar enquanto Bob escolherá se i for par;
- ▶ Se Alice possuir uma estratégia vencedora para a instância de QSAT, então ela joga assumindo que as literais não escolhidas são verdadeiras.

QSAT $\preceq_p$ GEOGRAFIA



- ▶ Ao final Bob escolherá uma das cláusulas. Como Alice tem uma estratégia vencedora em QSAT, todas as cláusulas tem pelo menos uma literal não percorrida (verdadeira).
- ▶ Alice então se move para uma das literais verdadeiras da cláusula e vence o jogo.

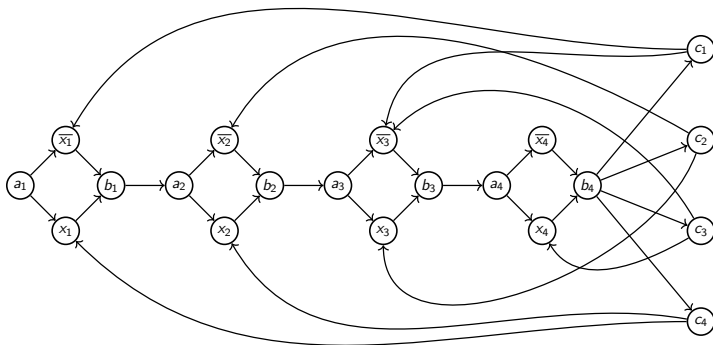
QSAT $\preceq_p$ GEOGRAFIA



- ▶ Ao final Bob escolherá uma das cláusulas. Como Alice tem uma estratégia vencedora em QSAT, todas as cláusulas tem pelo menos uma literal não percorrida (verdadeira).
- ▶ Alice então se move para uma das literais verdadeiras da cláusula e vence o jogo.

QSAT $\preceq_p$ GEOGRAFIA

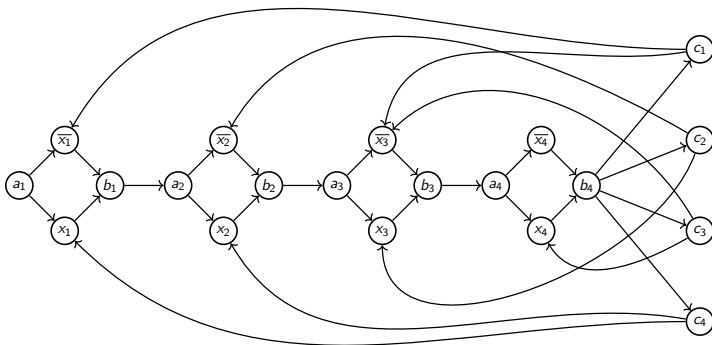
Teorema (Schaefer, 1978): GEOGRAFIA é PSPACE-completo.



- Bob usa sua possível estratégia vencedora em QSAT de modo análogo, com a diferença que haverá pelo menos uma cláusula com todas as literais já percorridas (falsas) e, ao selecionar esta cláusula Bob vence o jogo.

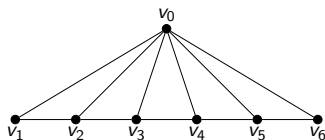
QSAT $\preceq_p$ GEOGRAFIA

Teorema (Schaefer, 1978): GEOGRAFIA é PSPACE-completo.



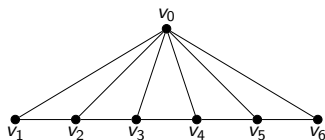
- Bob usa sua possível estratégia vencedora em QSAT de modo análogo, com a diferença que haverá pelo menos uma cláusula com todas as literais já percorridas (falsas) e, ao selecionar esta cláusula Bob vence o jogo.

Jogo de Formação de Conjunto Convexo



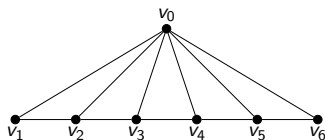
- ▶ Um conjunto de vértices S em um grafo G é dito **convexo** se para todo vértice u pertencente a um caminho mínimo entre dois vértices de S , $u \in S$;
- ▶ No Jogo de Formação de Conjunto Convexo (CSFG) Alice e Bob se alternam selecionando vértices de um grafo G ;
- ▶ A cada turno o conjunto de vértices selecionados sempre devem formar um conjunto convexo;
- ▶ Perde o jogador que não conseguir selecionar um vértice em seu turno (variante normal).

Jogo de Formação de Conjunto Convexo



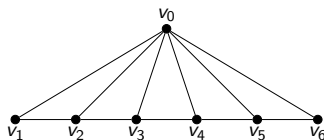
- ▶ Um conjunto de vértices S em um grafo G é dito **convexo** se para todo vértice u pertencente a um caminho mínimo entre dois vértices de S , $u \in S$;
- ▶ No Jogo de Formação de Conjunto Convexo (CSFG) Alice e Bob se alternam selecionando vértices de um grafo G ;
- ▶ A cada turno o conjunto de vértices selecionados sempre devem formar um conjunto convexo;
- ▶ Perde o jogador que não conseguir selecionar um vértice em seu turno (variante normal).

Jogo de Formação de Conjunto Convexo



- ▶ Um conjunto de vértices S em um grafo G é dito **convexo** se para todo vértice u pertencente a um caminho mínimo entre dois vértices de S , $u \in S$;
- ▶ No Jogo de Formação de Conjunto Convexo (CSFG) Alice e Bob se alternam selecionando vértices de um grafo G ;
- ▶ A cada turno o conjunto de vértices selecionados sempre devem formar um conjunto convexo;
- ▶ Perde o jogador que não conseguir selecionar um vértice em seu turno (variante normal).

Jogo de Formação de Conjunto Convexo



- ▶ Um conjunto de vértices S em um grafo G é dito **convexo** se para todo vértice u pertencente a um caminho mínimo entre dois vértices de S , $u \in S$;
- ▶ No Jogo de Formação de Conjunto Convexo (CSFG) Alice e Bob se alternam selecionando vértices de um grafo G ;
- ▶ A cada turno o conjunto de vértices selecionados sempre devem formar um conjunto convexo;
- ▶ Perde o jogador que não conseguir selecionar um vértice em seu turno (variante normal).

CSFG é PSPACE-completo

- ▶ A redução será realizada a partir do Jogo de Formação de Clique (JFC);
- ▶ O JFC é PSPACE-completo (Schaefer, 1978);
- ▶ A partir de uma instância G de JFC, podemos computar a instância G' acima para CSFG em tempo polinomial;
- ▶ Os vértices de G selecionados por Alice e Bob em G' devem formar uma clique;
- ▶ Não é possível que $w_1 \in \{u_1, u'_1\}$ e $w_2 \in \{u_2, u'_2\}$ sejam selecionados em CSFG.

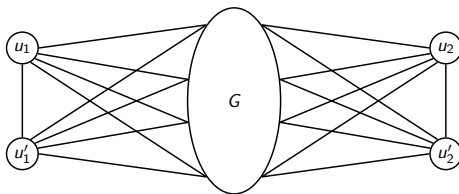
CSFG é PSPACE-completo

- ▶ A redução será realizada a partir do Jogo de Formação de Clique (JFC);
- ▶ O JFC é PSPACE-completo (Schaefer, 1978);
- ▶ A partir de uma instância G de JFC, podemos computar a instância G' acima para CSFG em tempo polinomial;
- ▶ Os vértices de G seleccionados por Alice e Bob em G' devem formar uma clique;
- ▶ Não é possível que $w_1 \in \{u_1, u'_1\}$ e $w_2 \in \{u_2, u'_2\}$ sejam seleccionados em CSFG.

CSFG é PSPACE-completo

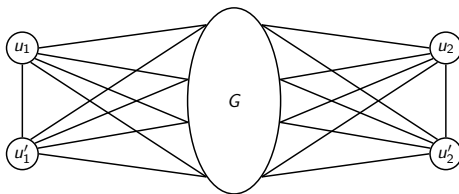
- ▶ A redução será realizada a partir do Jogo de Formação de Clique (JFC);
- ▶ O JFC é PSPACE-completo (Schaefer, 1978);
- ▶ A partir de uma instância G de JFC, podemos computar a instância G' acima para CSFG em tempo polinomial;
- ▶ Os vértices de G selecionados por Alice e Bob em G' devem formar uma clique;
- ▶ Não é possível que $w_1 \in \{u_1, u'_1\}$ e $w_2 \in \{u_2, u'_2\}$ sejam selecionados em CSFG.

CSFG é PSPACE-completo



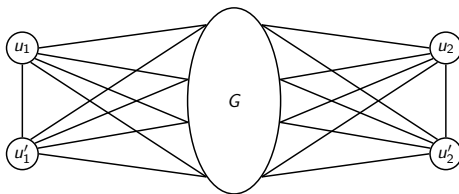
- ▶ A redução será realizada a partir do Jogo de Formação de Clique (JFC);
- ▶ O JFC é PSPACE-completo (Schaefer, 1978);
- ▶ A partir de uma instância G de JFC, podemos computar a instância G' acima para CSFG em tempo polinomial;
- ▶ Os vértices de G selecionados por Alice e Bob em G' devem formar uma clique;
- ▶ Não é possível que $w_1 \in \{u_1, u'_1\}$ e $w_2 \in \{u_2, u'_2\}$ sejam selecionados em CSFG.

CSFG é PSPACE-completo



- ▶ A redução será realizada a partir do Jogo de Formação de Clique (JFC);
- ▶ O JFC é PSPACE-completo (Schaefer, 1978);
- ▶ A partir de uma instância G de JFC, podemos computar a instância G' acima para CSFG em tempo polinomial;
- ▶ Os vértices de G selecionados por Alice e Bob em G' devem formar uma clique;
- ▶ Não é possível que $w_1 \in \{u_1, u'_1\}$ e $w_2 \in \{u_2, u'_2\}$ sejam selecionados em CSFG.

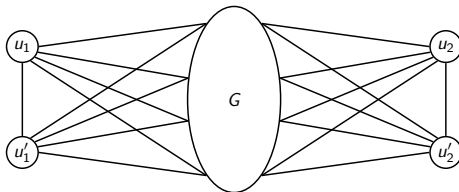
CSFG é PSPACE-completo



- ▶ A redução será realizada a partir do Jogo de Formação de Clique (JFC);
- ▶ O JFC é PSPACE-completo (Schaefer, 1978);
- ▶ A partir de uma instância G de JFC, podemos computar a instância G' acima para CSFG em tempo polinomial;
- ▶ Os vértices de G selecionados por Alice e Bob em G' devem formar uma clique;
- ▶ Não é possível que $w_1 \in \{u_1, u'_1\}$ e $w_2 \in \{u_2, u'_2\}$ sejam selecionados em CSFG.

CSFG é PSPACE-completo

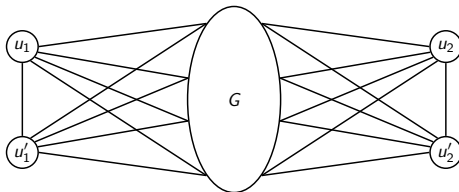
Teorema (Brosse et al., 2025): CSFG é PSPACE-completo.



- ▶ Se Alice possui uma estratégia vencedora para JFC em G , ela seleciona os mesmos vértices de sua estratégia para CSFG em G' ;
- ▶ Quando Bob eventualmente escolher um vértice em $\{u_1, u'_1, u_2, u'_2\}$, Alice escolhe o irmão verdadeiro do vértice escolhido por Bob;
- ▶ Após essa jogada, somente vértices de G poderão ser selecionados;
- ▶ Como Alice possui uma estratégia vencedora em JFC, ela vence CSFG em G' ;
- ▶ Bob age analogamente caso ele possua uma estratégia vencedora para JFC em G ;

CSFG é PSPACE-completo

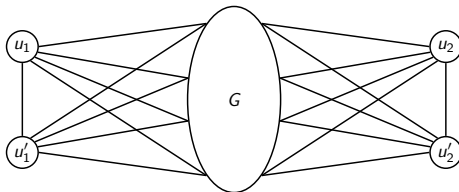
Teorema (Brosse et al., 2025): CSFG é PSPACE-completo.



- ▶ Se Alice possui uma estratégia vencedora para JFC em G , ela seleciona os mesmos vértices de sua estratégia para CSFG em G' ;
- ▶ Quando Bob eventualmente escolher um vértice em $\{u_1, u'_1, u_2, u'_2\}$, Alice escolhe o irmão verdadeiro do vértice escolhido por Bob;
- ▶ Após essa jogada, somente vértices de G poderão ser selecionados;
- ▶ Como Alice possui uma estratégia vencedora em JFC, ela vence CSFG em G' ;
- ▶ Bob age analogamente caso ele possua uma estratégia vencedora para JFC em G ;

CSFG é PSPACE-completo

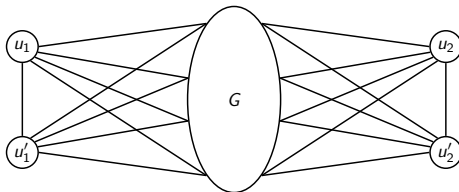
Teorema (Brosse et al., 2025): CSFG é PSPACE-completo.



- ▶ Se Alice possui uma estratégia vencedora para JFC em G , ela seleciona os mesmos vértices de sua estratégia para CSFG em G' ;
- ▶ Quando Bob eventualmente escolher um vértice em $\{u_1, u'_1, u_2, u'_2\}$, Alice escolhe o irmão verdadeiro do vértice escolhido por Bob;
- ▶ Após essa jogada, somente vértices de G poderão ser selecionados;
- ▶ Como Alice possui uma estratégia vencedora em JFC, ela vence CSFG em G' ;
- ▶ Bob age analogamente caso ele possua uma estratégia vencedora para JFC em G ;

CSFG é PSPACE-completo

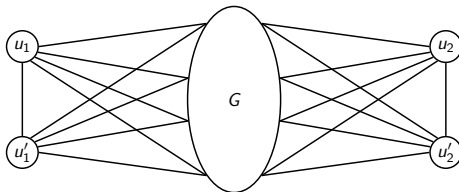
Teorema (Brosse et al., 2025): CSFG é PSPACE-completo.



- ▶ Se Alice possui uma estratégia vencedora para JFC em G , ela seleciona os mesmos vértices de sua estratégia para CSFG em G' ;
- ▶ Quando Bob eventualmente escolher um vértice em $\{u_1, u'_1, u_2, u'_2\}$, Alice escolhe o irmão verdadeiro do vértice escolhido por Bob;
- ▶ Após essa jogada, somente vértices de G poderão ser selecionados;
- ▶ Como Alice possui uma estratégia vencedora em JFC, ela vence CSFG em G' ;
- ▶ Bob age analogamente caso ele possua uma estratégia vencedora para JFC em G ;

CSFG é PSPACE-completo

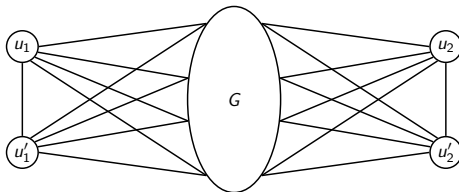
Teorema (Brosse et al., 2025): CSFG é PSPACE-completo.



- ▶ Se Alice possui uma estratégia vencedora para JFC em G , ela seleciona os mesmos vértices de sua estratégia para CSFG em G' ;
- ▶ Quando Bob eventualmente escolher um vértice em $\{u_1, u'_1, u_2, u'_2\}$, Alice escolhe o irmão verdadeiro do vértice escolhido por Bob;
- ▶ Após essa jogada, somente vértices de G poderão ser selecionados;
- ▶ Como Alice possui uma estratégia vencedora em JFC, ela vence CSFG em G' ;
- ▶ Bob age analogamente caso ele possua uma estratégia vencedora para JFC em G ;

CSFG é PSPACE-completo

Teorema (Brosse et al., 2025): CSFG é PSPACE-completo.



- ▶ Se Alice possui uma estratégia vencedora para JFC em G , ela seleciona os mesmos vértices de sua estratégia para CSFG em G' ;
- ▶ Quando Bob eventualmente escolher um vértice em $\{u_1, u'_1, u_2, u'_2\}$, Alice escolhe o irmão verdadeiro do vértice escolhido por Bob;
- ▶ Após essa jogada, somente vértices de G poderão ser selecionados;
- ▶ Como Alice possui uma estratégia vencedora em JFC, ela vence CSFG em G' ;
- ▶ Bob age analogamente caso ele possua uma estratégia vencedora para JFC em G ;

Capítulo 7 - Jogos de Coloração em Grafos

Capítulo 7

Jogos de Coloração em Grafos

Jogo de Coloração Própria

- ▶ **Entrada:** Um grafo G e um conjunto de k cores;
- ▶ Os jogadores se alternam colorindo vértices de modo que os vértices coloridos induzam uma coloração parcial própria de G .
- ▶ Alice vence se todos os vértices de G forem coloridos, caso contrário Bob é o vencedor. (Variante de Otimização)
- ▶ O número cromático de jogo de um grafo G , $\chi_g(G)$, é o menor k para o qual Alice possui uma s.v. com k cores.
- ▶ $\chi(G) \leq \chi_g(G) \leq \Delta(G) + 1$

Jogo de Coloração Própria

- ▶ **Entrada:** Um grafo G e um conjunto de k cores;
- ▶ Os jogadores se alternam colorindo vértices de modo que os vértices coloridos induzam uma coloração parcial própria de G .
- ▶ Alice vence se todos os vértices de G forem coloridos, caso contrário Bob é o vencedor. (Variante de Otimização)
- ▶ O número cromático de jogo de um grafo G , $\chi_g(G)$, é o menor k para o qual Alice possui uma s.v. com k cores.
- ▶ $\chi(G) \leq \chi_g(G) \leq \Delta(G) + 1$

Jogo de Coloração Própria

- ▶ **Entrada:** Um grafo G e um conjunto de k cores;
- ▶ Os jogadores se alternam colorindo vértices de modo que os vértices coloridos induzam uma coloração parcial própria de G .
- ▶ Alice vence se todos os vértices de G forem coloridos, caso contrário Bob é o vencedor. (Variante de Otimização)
- ▶ O número cromático de jogo de um grafo G , $\chi_g(G)$, é o menor k para o qual Alice possui uma s.v. com k cores.
- ▶ $\chi(G) \leq \chi_g(G) \leq \Delta(G) + 1$

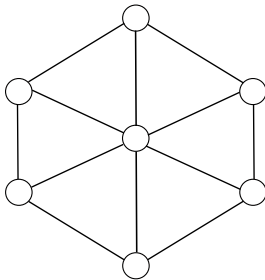
Jogo de Coloração Própria

- ▶ **Entrada:** Um grafo G e um conjunto de k cores;
- ▶ Os jogadores se alternam colorindo vértices de modo que os vértices coloridos induzam uma coloração parcial própria de G .
- ▶ Alice vence se todos os vértices de G forem coloridos, caso contrário Bob é o vencedor. (Variante de Otimização)
- ▶ O número cromático de jogo de um grafo G , $\chi_g(G)$, é o menor k para o qual Alice possui uma s.v. com k cores.
- ▶ $\chi(G) \leq \chi_g(G) \leq \Delta(G) + 1$

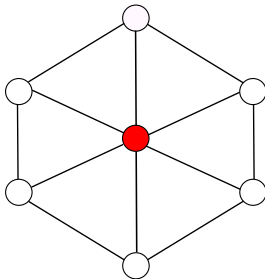
Jogo de Coloração Própria

- ▶ **Entrada:** Um grafo G e um conjunto de k cores;
- ▶ Os jogadores se alternam colorindo vértices de modo que os vértices coloridos induzam uma coloração parcial própria de G .
- ▶ Alice vence se todos os vértices de G forem coloridos, caso contrário Bob é o vencedor. (Variante de Otimização)
- ▶ O número cromático de jogo de um grafo G , $\chi_g(G)$, é o menor k para o qual Alice possui uma s.v. com k cores.
- ▶ $\chi(G) \leq \chi_g(G) \leq \Delta(G) + 1$

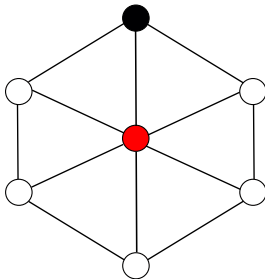
Vamos jogar um pouco



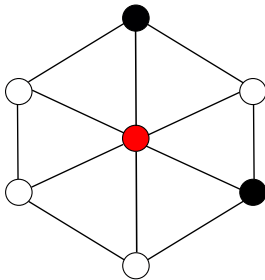
Vamos jogar um pouco



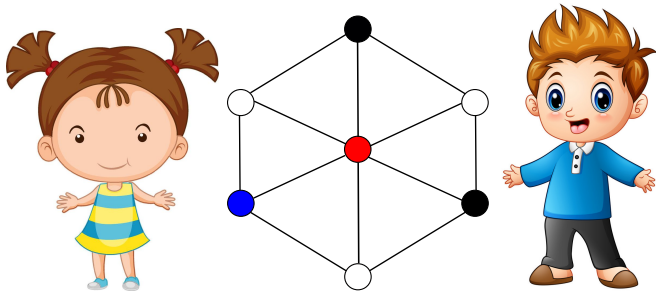
Vamos jogar um pouco



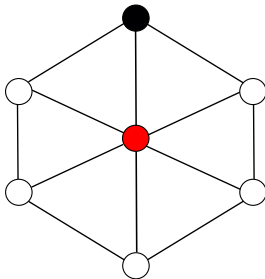
Vamos jogar um pouco



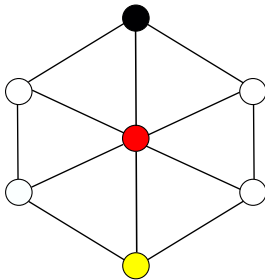
Vamos jogar um pouco



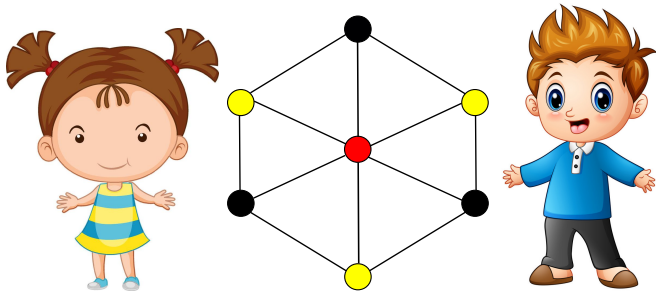
Vamos jogar um pouco



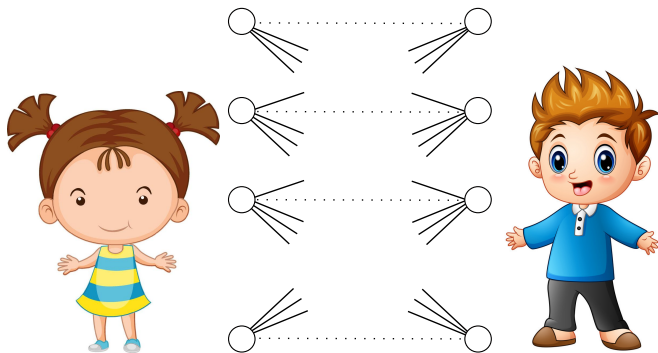
Vamos jogar um pouco



Vamos jogar um pouco



É melhor ser o primeiro a jogar?

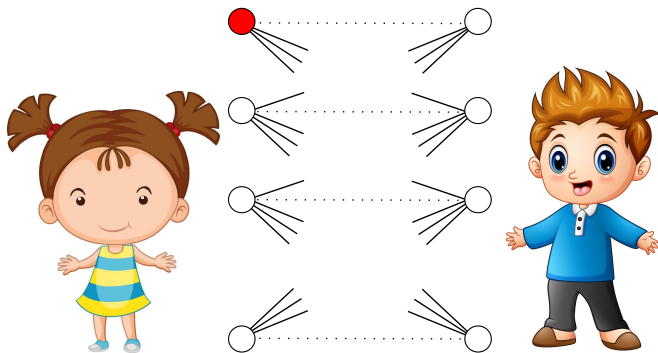


Seja $K_{n,n}^*$ obtido de $K_{n,n}$ removendo um emparelhamento perfeito.

Se não é a primeira a jogar: Alice vence em $K_{n,n}^*$ com 2 cores.

Se é a primeira a jogar: Alice **só** vence em $K_{n,n}^*$ com n cores.

É melhor ser o primeiro a jogar?

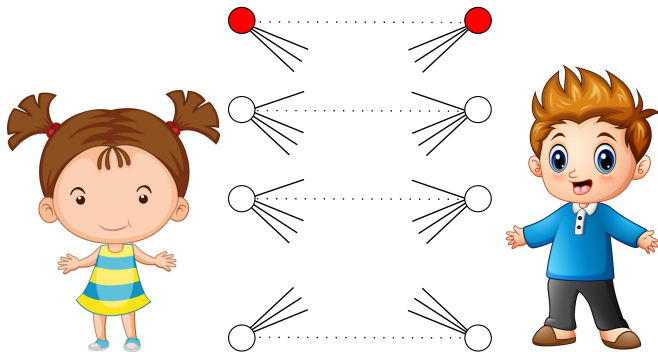


Seja $K_{n,n}^*$ obtido de $K_{n,n}$ removendo um emparelhamento perfeito.

Se não é a primeira a jogar: Alice vence em $K_{n,n}^*$ com 2 cores.

Se é a primeira a jogar: Alice **só** vence em $K_{n,n}^*$ com n cores.

É melhor ser o primeiro a jogar?

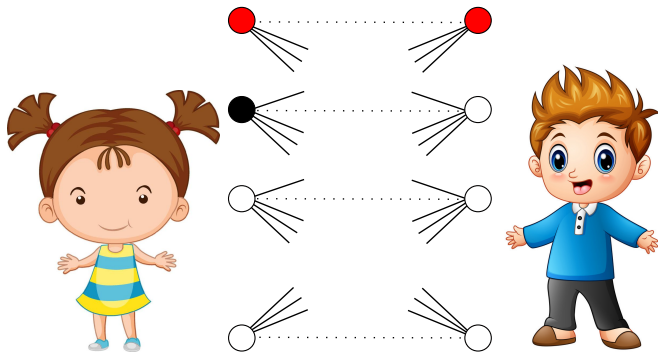


Seja $K_{n,n}^*$ obtido de $K_{n,n}$ removendo um emparelhamento perfeito.

Se não é a primeira a jogar: Alice vence em $K_{n,n}^*$ com 2 cores.

Se é a primeira a jogar: Alice **só** vence em $K_{n,n}^*$ com n cores.

É melhor ser o primeiro a jogar?

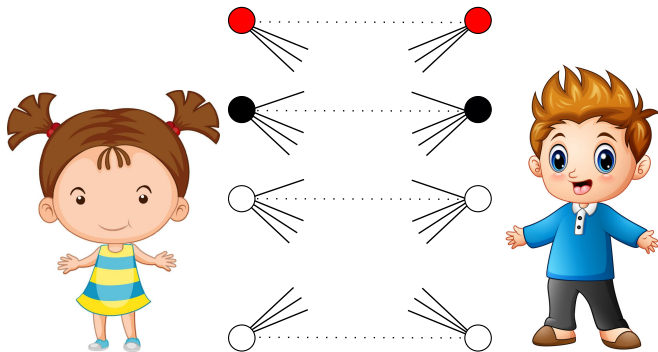


Seja $K_{n,n}^*$ obtido de $K_{n,n}$ removendo um emparelhamento perfeito.

Se não é a primeira a jogar: Alice vence em $K_{n,n}^*$ com 2 cores.

Se é a primeira a jogar: Alice **só** vence em $K_{n,n}^*$ com n cores.

É melhor ser o primeiro a jogar?

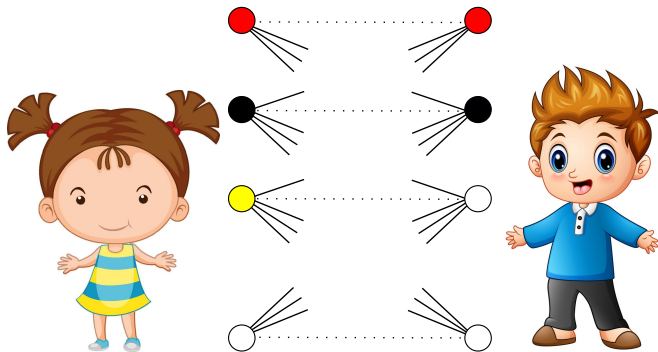


Seja $K_{n,n}^*$ obtido de $K_{n,n}$ removendo um emparelhamento perfeito.

Se não é a primeira a jogar: Alice vence em $K_{n,n}^*$ com 2 cores.

Se é a primeira a jogar: Alice **só** vence em $K_{n,n}^*$ com n cores.

É melhor ser o primeiro a jogar?

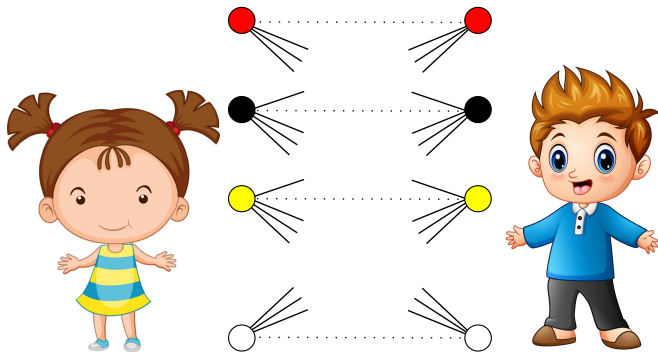


Seja $K_{n,n}^*$ obtido de $K_{n,n}$ removendo um emparelhamento perfeito.

Se não é a primeira a jogar: Alice vence em $K_{n,n}^*$ com 2 cores.

Se é a primeira a jogar: Alice **só** vence em $K_{n,n}^*$ com n cores.

É melhor ser o primeiro a jogar?

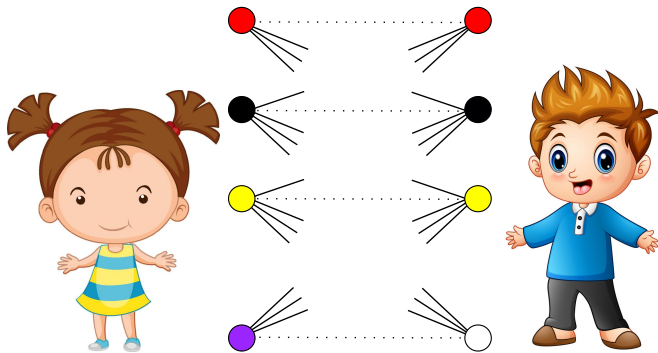


Seja $K_{n,n}^*$ obtido de $K_{n,n}$ removendo um emparelhamento perfeito.

Se não é a primeira a jogar: Alice vence em $K_{n,n}^*$ com 2 cores.

Se é a primeira a jogar: Alice **só** vence em $K_{n,n}^*$ com n cores.

É melhor ser o primeiro a jogar?

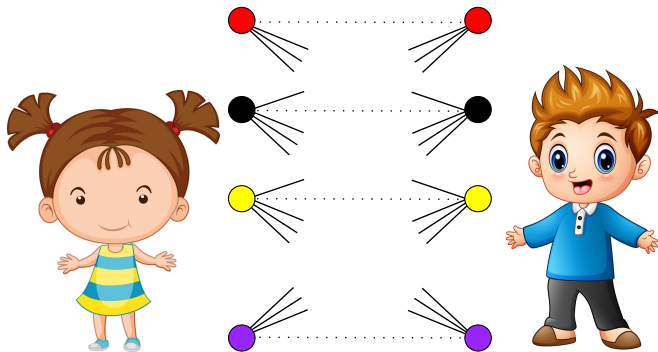


Seja $K_{n,n}^*$ obtido de $K_{n,n}$ removendo um emparelhamento perfeito.

Se não é a primeira a jogar: Alice vence em $K_{n,n}^*$ com 2 cores.

Se é a primeira a jogar: Alice **só** vence em $K_{n,n}^*$ com n cores.

É melhor ser o primeiro a jogar?



Seja $K_{n,n}^*$ obtido de $K_{n,n}$ removendo um emparelhamento perfeito.

Se não é a primeira a jogar: Alice vence em $K_{n,n}^*$ com 2 cores.

Se é a primeira a jogar: Alice **só** vence em $K_{n,n}^*$ com n cores.

Ter mais cores facilita para Alice?

Conjectura (Zhu, 1999): Se Alice vence com k cores, então ela também vence com $k + 1$ cores.

Conjectura (Zhu, 1999): Se Alice vence com k cores, então ela também vence com $f(k)$ cores, para alguma função f , distinta da função identidade.

Ter mais cores facilita para Alice?

Conjectura (Zhu, 1999): Se Alice vence com k cores, então ela também vence com $k + 1$ cores.

Conjectura (Zhu, 1999): Se Alice vence com k cores, então ela também vence com $f(k)$ cores, para alguma função f , distinta da função identidade.

Ter mais cores facilita para Alice?

Conjectura (Zhu, 1999): Se Alice vence com k cores, então ela também vence com $k + 1$ cores.

Conjectura (Zhu, 1999): Se Alice vence com k cores, então ela também vence com $f(k)$ cores, para alguma função f , distinta da função identidade.

O Jogo de Marcação de Vértices

- ▶ **Entrada:** Um grafo G e um inteiro k ;
- ▶ Os jogadores se revezam marcando vértices;
- ▶ Bob vence se a qualquer momento um vértice não marcado tiver k vizinhos já marcados.
- ▶ O menor k para o qual Alice vence em G é chamado de número de jogo de coloração, $col_G(G)$.
- ▶ $\chi_G(G) \leq col_G(G)$

O Jogo de Marcação de Vértices

- ▶ **Entrada:** Um grafo G e um inteiro k ;
- ▶ Os jogadores se revezam marcando vértices;
- ▶ Bob vence se a qualquer momento um vértice não marcado tiver k vizinhos já marcados.
- ▶ O menor k para o qual Alice vence em G é chamado de número de jogo de coloração, $col_g(G)$.
- ▶ $\chi_g(G) \leq col_g(G)$

O Jogo de Marcação de Vértices

- ▶ **Entrada:** Um grafo G e um inteiro k ;
- ▶ Os jogadores se revezam marcando vértices;
- ▶ Bob vence se a qualquer momento um vértice não marcado tiver k vizinhos já marcados.
- ▶ O menor k para o qual Alice vence em G é chamado de número de jogo de coloração, $col_g(G)$.
- ▶ $\chi_g(G) \leq col_g(G)$

O Jogo de Marcação de Vértices

- ▶ **Entrada:** Um grafo G e um inteiro k ;
- ▶ Os jogadores se revezam marcando vértices;
- ▶ Bob vence se a qualquer momento um vértice não marcado tiver k vizinhos já marcados.
- ▶ O menor k para o qual Alice vence em G é chamado de número de jogo de coloração, $col_g(G)$.
- ▶ $\chi_g(G) \leq col_g(G)$

O Jogo de Marcação de Vértices

- ▶ **Entrada:** Um grafo G e um inteiro k ;
- ▶ Os jogadores se revezam marcando vértices;
- ▶ Bob vence se a qualquer momento um vértice não marcado tiver k vizinhos já marcados.
- ▶ O menor k para o qual Alice vence em G é chamado de número de jogo de coloração, $col_g(G)$.
- ▶ $\chi_g(G) \leq col_g(G)$

O Jogo de Marcação de Vértices

- ▶ **Entrada:** Um grafo G e um inteiro k ;
- ▶ Os jogadores se revezam marcando vértices;
- ▶ Bob vence se a qualquer momento um vértice não marcado tiver k vizinhos já marcados.
- ▶ O menor k para o qual Alice vence em G é chamado de número de jogo de coloração, $col_g(G)$.
- ▶ $\chi_g(G) \leq col_g(G)$

Jogo de Marcação em Árvores

- ▶ Seja T uma árvore e T^* a subárvore de minimal de T que contém todos os vértices marcados até o momento;
- ▶ Alice inicia marcando um vértice qualquer r ;
- ▶ Suponha que Bob marca um vértice v e seja P o caminho entre r e v em T e u o vértice mais distante de r em $V(P) \cap T^*$;

Jogo de Marcação em Árvores

- ▶ Seja T uma árvore e T^* a subárvore de minimal de T que contém todos os vértices marcados até o momento;
- ▶ Alice inicia marcando um vértice qualquer r ;
- ▶ Suponha que Bob marca um vértice v e seja P o caminho entre r e v em T e u o vértice mais distante de r em $V(P) \cap T^*$;

Jogo de Marcação em Árvores

- ▶ Seja T uma árvore e T^* a subárvore de minimal de T que contém todos os vértices marcados até o momento;
- ▶ Alice inicia marcando um vértice qualquer r ;
- ▶ Suponha que Bob marca um vértice v e seja P o caminho entre r e v em T e u o vértice mais distante de r em $V(P) \cap T^*$;
 - ▶ R1: Se u não está marcado, Alice marca u ;
 - ▶ R2: Se existe um vértice não marcado w em T^* , Alice marca w ;
 - ▶ R3: Alice marca um vértice arbitrário $x \in V(T^*)$.

Jogo de Marcação em Árvores

- ▶ Seja T uma árvore e T^* a subárvore de minimal de T que contém todos os vértices marcados até o momento;
- ▶ Alice inicia marcando um vértice qualquer r ;
- ▶ Suponha que Bob marca um vértice v e seja P o caminho entre r e v em T e u o vértice mais distante de r em $V(P) \cap T^*$;
 - ▶ R1: Se u não está marcado, Alice marca u ;
 - ▶ R2: Se existe um vértice não marcado w em T^* , Alice marca w ;
 - ▶ R3: Alice marca um vértice adjacente a $V(T^*)$.

Jogo de Marcação em Árvores

- ▶ Seja T uma árvore e T^* a subárvore de minimal de T que contém todos os vértices marcados até o momento;
- ▶ Alice inicia marcando um vértice qualquer r ;
- ▶ Suponha que Bob marca um vértice v e seja P o caminho entre r e v em T e u o vértice mais distante de r em $V(P) \cap T^*$;
 - ▶ R1: Se u não está marcado, Alice marca u ;
 - ▶ R2: Se existe um vértice não marcado w em T^* , Alice marca w ;
 - ▶ R3: Alice marca um vértice adjacente a $V(T^*)$.

Jogo de Marcação em Árvores

- ▶ Seja T uma árvore e T^* a subárvore de minimal de T que contém todos os vértices marcados até o momento;
- ▶ Alice inicia marcando um vértice qualquer r ;
- ▶ Suponha que Bob marca um vértice v e seja P o caminho entre r e v em T e u o vértice mais distante de r em $V(P) \cap T^*$;
 - ▶ R1: Se u não está marcado, Alice marca u ;
 - ▶ R2: Se existe um vértice não marcado w em T^* , Alice marca w ;
 - ▶ R3: Alice marca um vértice adjacente a $V(T^*)$.

Jogo de Marcação em Árvores

- ▶ Seja T uma árvore e T^* a subárvore de minimal de T que contém todos os vértices marcados até o momento;
- ▶ Alice inicia marcando um vértice qualquer r ;
- ▶ Suponha que Bob marca um vértice v e seja P o caminho entre r e v em T e u o vértice mais distante de r em $V(P) \cap T^*$;
 - ▶ R1: Se u não está marcado, Alice marca u ;
 - ▶ R2: Se existe um vértice não marcado w em T^* , Alice marca w ;
 - ▶ R3: Alice marca um vértice adjacente a $V(T^*)$.

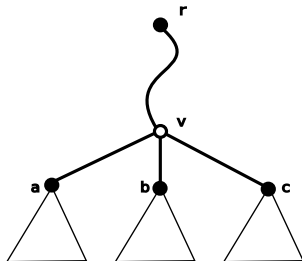
Jogo de Marcação em Árvores

- ▶ Assuma que existe um vértice v não marcado e que três vizinhos de v , a , b e c foram marcados nesta ordem.

- ▶ Assuma que a não está no caminho entre r e v .

Jogo de Marcação em Árvores

- Assuma que existe um vértice v não marcado e que três vizinhos de v , a , b e c foram marcados nesta ordem.



- Assuma que a não está no caminho entre r e v .

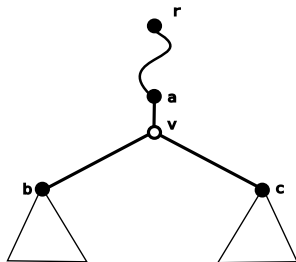
Jogo de Marcação em Árvores

- ▶ Assuma que existe um vértice v não marcado e que três vizinhos de v , a , b e c foram marcados nesta ordem.

- ▶ Assuma que a está no caminho entre r e v .

Jogo de Marcação em Árvores

- Assuma que existe um vértice v não marcado e que três vizinhos de v , a , b e c foram marcados nesta ordem.



- Assuma que a está no caminho entre r e v .

Jogo de Marcação em Árvores

- ▶ Teorema (Faigle et al. 1993): Se T é uma árvore, então $col_g(T) \leq 4$.
- ▶ Corolário: Se T é uma árvore, então $\chi_g(T) \leq 4$.

Jogo de Marcação em Árvores

- ▶ Teorema (Faigle et al. 1993): Se T é uma árvore, então $col_g(T) \leq 4$.
- ▶ Corolário: Se T é uma árvore, então $\chi_g(T) \leq 4$.

Complexidade do Jogo de Coloração

Teorema (Bodlaender, 1991): SEQUENTIAL COLORING GAME é PSPACE-completo.

Teorema (Bodlaender, 1991): SEQUENTIAL COLORING CONSTRUCTION GAME é PSPACE-completo.

- ▶ A complexidade do Jogo de Coloração (COLORING CONSTRUCTION GAME) foi considerada por Bodlaender, *um problema interessante em aberto*.

Teorema (Costa et al., 2020): O Jogo de Coloração é PSPACE-completo.

Complexidade do Jogo de Coloração

Teorema (Bodlaender, 1991): SEQUENTIAL COLORING GAME é PSPACE-completo.

Teorema (Bodlaender, 1991): SEQUENTIAL COLORING CONSTRUCTION GAME é PSPACE-completo.

- ▶ A complexidade do Jogo de Coloração (COLORING CONSTRUCTION GAME) foi considerada por Bodlaender, *um problema interessante em aberto*.

Teorema (Costa et al., 2020): O Jogo de Coloração é PSPACE-completo.

Complexidade do Jogo de Coloração

Teorema (Bodlaender, 1991): SEQUENTIAL COLORING GAME é PSPACE-completo.

Teorema (Bodlaender, 1991): SEQUENTIAL COLORING CONSTRUCTION GAME é PSPACE-completo.

- ▶ A complexidade do Jogo de Coloração (COLORING CONSTRUCTION GAME) foi considerada por Bodlaender, *um problema interessante em aberto*.

Teorema (Costa et al., 2020): O Jogo de Coloração é PSPACE-completo.

Complexidade do Jogo de Coloração

Teorema (Bodlaender, 1991): SEQUENTIAL COLORING GAME é PSPACE-completo.

Teorema (Bodlaender, 1991): SEQUENTIAL COLORING CONSTRUCTION GAME é PSPACE-completo.

- A complexidade do Jogo de Coloração (COLORING CONSTRUCTION GAME) foi considerada por Bodlaender, *um problema interessante em aberto*.

Teorema (Costa et al., 2020): O Jogo de Coloração é PSPACE-completo.

Complexidade do Jogo de Coloração

Teorema (Bodlaender, 1991): SEQUENTIAL COLORING GAME é PSPACE-completo.

Teorema (Bodlaender, 1991): SEQUENTIAL COLORING CONSTRUCTION GAME é PSPACE-completo.

- A complexidade do Jogo de Coloração (COLORING CONSTRUCTION GAME) foi considerada por Bodlaender, *um problema interessante em aberto*.

Teorema (Costa et al., 2020): O Jogo de Coloração é PSPACE-completo.

Jogo Normal de Coloração

Teorema (Bodlaender, 1991): O jogo normal de coloração com $k = 1$ é PSPACE-completo.

- ▶ Equivalente a KAYLES.

Teorema (Bodlaender, 1991) O jogo normal de coloração com $k = 2$ é PSPACE-completo.

- ▶ Redução a partir do jogo com $k = 1$.

Jogo Normal de Coloração

Teorema (Bodlaender, 1991): O jogo normal de coloração com $k = 1$ é PSPACE-completo.

- Equivalente a KAYLES.

Teorema (Bodlaender, 1991) O jogo normal de coloração com $k = 2$ é PSPACE-completo.

- Redução a partir do jogo com $k = 1$.

Jogo Normal de Coloração

Teorema (Bodlaender, 1991): O jogo normal de coloração com $k = 1$ é PSPACE-completo.

- ▶ Equivalente a KAYLES.

Teorema (Bodlaender, 1991) O jogo normal de coloração com $k = 2$ é PSPACE-completo.

- ▶ Redução a partir do jogo com $k = 1$.

Jogo Normal de Coloração

Teorema (Bodlaender, 1991): O jogo normal de coloração com $k = 1$ é PSPACE-completo.

- ▶ Equivalente a KAYLES.

Teorema (Bodlaender, 1991) O jogo normal de coloração com $k = 2$ é PSPACE-completo.

- ▶ Redução a partir do jogo com $k = 1$.

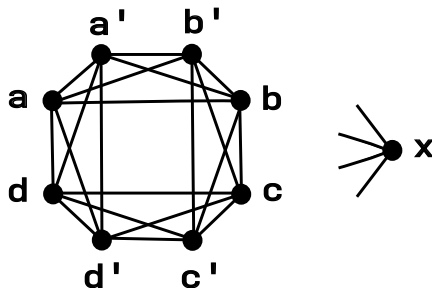
Jogo Normal de Coloração

Teorema (Bodlaender, 1991): O jogo normal de coloração com $k = 1$ é PSPACE-completo.

- Equivalente a KAYLES.

Teorema (Bodlaender, 1991) O jogo normal de coloração com $k = 2$ é PSPACE-completo.

- Redução a partir do jogo com $k = 1$.



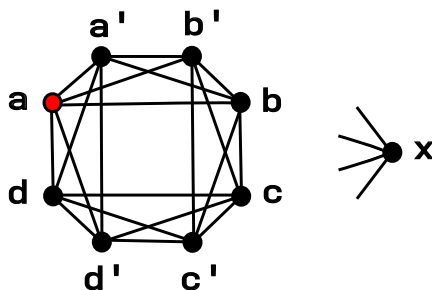
Jogo Normal de Coloração

Teorema (Bodlaender, 1991): O jogo normal de coloração com $k = 1$ é PSPACE-completo.

- ▶ Equivalente a KAYLES.

Teorema (Bodlaender, 1991) O jogo normal de coloração com $k = 2$ é PSPACE-completo.

- ▶ Redução a partir do jogo com $k = 1$.



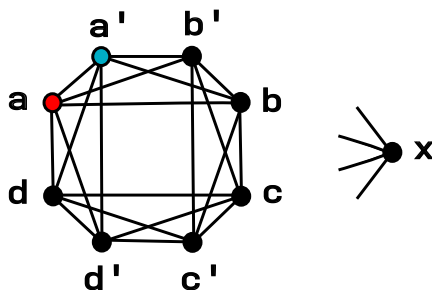
Jogo Normal de Coloração

Teorema (Bodlaender, 1991): O jogo normal de coloração com $k = 1$ é PSPACE-completo.

- Equivalente a KAYLES.

Teorema (Bodlaender, 1991) O jogo normal de coloração com $k = 2$ é PSPACE-completo.

- Redução a partir do jogo com $k = 1$.



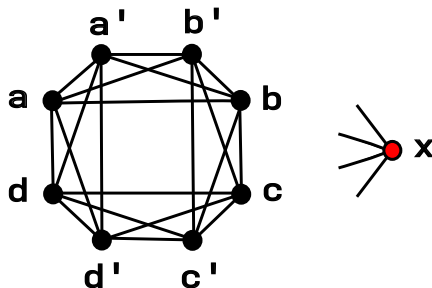
Jogo Normal de Coloração

Teorema (Bodlaender, 1991): O jogo normal de coloração com $k = 1$ é PSPACE-completo.

- ▶ Equivalente a KAYLES.

Teorema (Bodlaender, 1991) O jogo normal de coloração com $k = 2$ é PSPACE-completo.

- ▶ Redução a partir do jogo com $k = 1$.



Jogo de Coloração Conexa

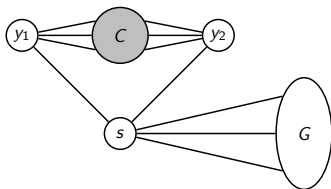
- ▶ No **Jogo de Coloração Conexa** os vértices coloridos devem induzir um grafo conexo.
- ▶ Redução do Jogo de Coloração (G, k) no qual Bob começa, PSPACE-completo (Lima et al., 2022).
- ▶ Assuma que k é par e que $V(G)$ é ímpar.
- ▶ A redução computa o grafo abaixo e permite $k + 1$ cores serem utilizadas. C é uma clique com k vértices.

Jogo de Coloração Conexa

- ▶ No **Jogo de Coloração Conexa** os vértices coloridos devem induzir um grafo conexo.
- ▶ Redução do Jogo de Coloração (G, k) no qual Bob começa, PSPACE-completo (Lima et al., 2022).
- ▶ Assuma que k é par e que $V(G)$ é ímpar.
- ▶ A redução computa o grafo abaixo e permite $k + 1$ cores serem utilizadas. C é uma clique com k vértices.

Jogo de Coloração Conexa

- ▶ No **Jogo de Coloração Conexa** os vértices coloridos devem induzir um grafo conexo.
- ▶ Redução do Jogo de Coloração (G, k) no qual Bob começa, PSPACE-completo (Lima et al., 2022).
- ▶ Assuma que k é par e que $V(G)$ é ímpar.
- ▶ A redução computa o grafo abaixo e permite $k + 1$ cores serem utilizadas. C é uma clique com k vértices.



FIM DA AULA 03

FIM DA AULA 03

Perguntas (se houver tempo) ??

Teoria dos Jogos Combinatórios em Grafos

Aula 04

**Samuel Araújo, Nicolas Martins,
Nicolas Nisse, Rudini Sampaio**

CBM-2025, IMPA,
Rio de Janeiro, quinta, 31-Julho, 8h

Capítulo 6

Jogos de Convexidade em Grafos

Capítulo 8

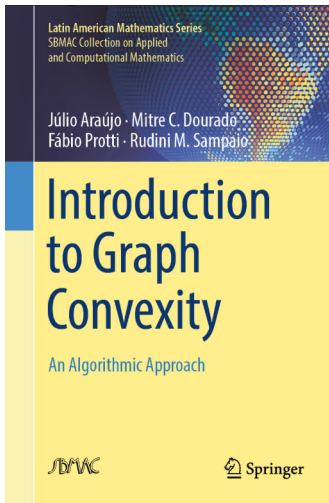
Jogos de Dominação em Grafos

Capítulo 6: Jogos de Convexidade em Grafos

Capítulo 6

Jogos de Convexidade em Grafos

Convexidade em Grafos



Baseado
em livro
de
minicurso
do
CBM-2023

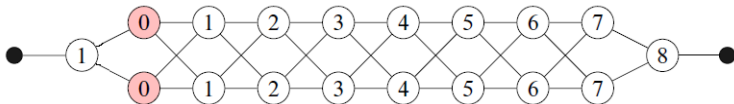
Convexidade Geométrica é uma tema clássico estudado há muito tempo. Convexidade em Grafos começou a ser investigada nos anos 1970. Erdős e outros autores (1972) publicaram um dos 1ºs artigos, voltado a torneios, que são orientações de grafos completos.

Convexidade em Grafos - Conceitos básicos

Intervalo $I(S)$: S e todo vértice em caminho mínimo entre vértices de S .

S é **convexo** se $I(S) = S$.

Fecho convexo $\text{conv}(S)$: menor conjunto convexo que contém S .
(*convex hull*)



S é um **conjunto de intervalo** (*interval set*) se $I(S) = V(G)$.

S é um **conjunto de envoltória** (*hull set*) se $\text{conv}(S) = V(G)$.

$\text{in}(G)$: *interval number*: tamanho do menor *interval set*.

$\text{hn}(G)$: *hull number*: tamanho do menor *hull set*.

Jogos de Convexidade em Grafos

Frank Harary (considerado um dos pais da Teoria Moderna dos Grafos) introduziu em 1984 os primeiros Jogos de Convexidade.

Alice e Bob se alternam selecionando um vértice de G que não está em $f_1(S)$, onde S é o conjunto de vértices selecionados no jogo. O jogo termina quando $f_2(S) = V(G)$.

Quatro primeiros jogos de convexidade:

- | | |
|---|----------------------|
| ▶ Jogo IG: $f_1(S) = S$ e $f_2(S) = I(S)$; | Interval game |
| ▶ Jogo HG: $f_1(S) = S$ e $f_2(S) = \text{conv}(S)$; | Hull game |
| ▶ Jogo CIG: $f_1(S) = f_2(S) = I(S)$. | Closed Interval game |
| ▶ Jogo CHG: $f_1(S) = f_2(S) = \text{conv}(S)$; | Closed Hull game |

Variantes Normal, Misère e de Otimização:

$\text{gin}(G)$, $\text{ghn}(G)$, $\text{gcin}(G)$, $\text{gchn}(G)$

Jogos de Convexidade em Árvore

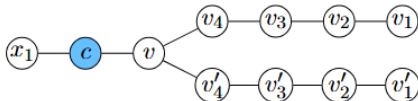
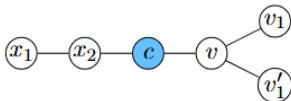
Alice e Bob se alternam selecionando um vértice de G que não está em $f_1(S)$, onde S é o conjunto de vértices selecionados no jogo. O jogo termina quando $f_2(S) = V(G)$.

Quatro primeiros jogos de convexidade:

- ▶ Jogo IG: $f_1(S) = S$ e $f_2(S) = I(S)$; Interval game
- ▶ Jogo HG: $f_1(S) = S$ e $f_2(S) = \text{conv}(S)$; Hull game
- ▶ Jogo CIG: $f_1(S) = f_2(S) = I(S)$. Closed Interval game
- ▶ Jogo CHG: $f_1(S) = f_2(S) = \text{conv}(S)$; Closed Hull game

Variantes Normal, Misère e de Otimização:

$\text{gin}(G)$, $\text{ghn}(G)$, $\text{gcin}(G)$, $\text{gchn}(G)$



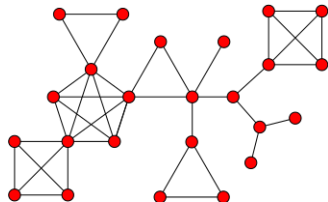
[Buckley, Harary, 1985]: Alice vence IG e HG em **árvores** na variante normal (resp. misère) **se e só se** n é ímpar (resp. par).

[Araújo, Brito, Folz, Freitas, Sampaio, 2024]: CIG e CHG são polinomiais em **árvores** nas variantes normal e misère (**Teoria de Sprague-Grundy**).

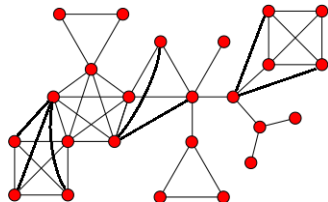
Jogos de Convexidade IG e HG em Árvore

[Buckley, Harary, 1985]: Alice vence IG e HG em **árvores** na variante normal (resp. misère) **se e só se** n é ímpar (resp. par).

[Araújo, Brito, Folz, Freitas, Sampaio, 2024]: Alice vence IG e HG em **grafos ptolemaicos** na variante normal (resp. misère) **se e só se** n é ímpar (resp. par).



Grafo de Blocos



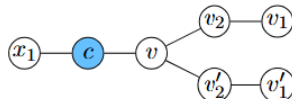
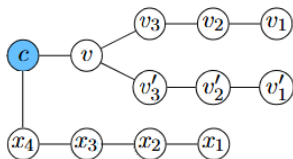
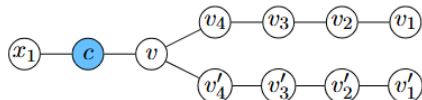
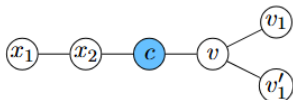
Grafo Ptolemaico

Grafo ptolemaico: grafos cordais (sem ciclos induzidos exceto triângulo) e distância-hereditária (todo caminho induzido é mínimo).

Estratégia: evitar pegar o penúltimo vértice simplicial.

Jogos de Convexidade CIG e CHG em Árvores

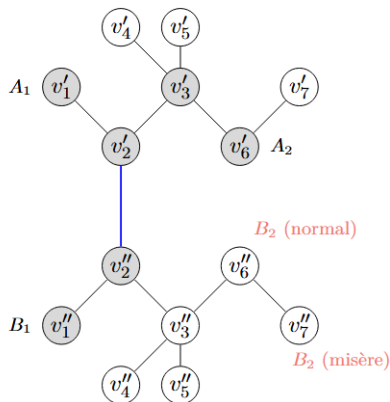
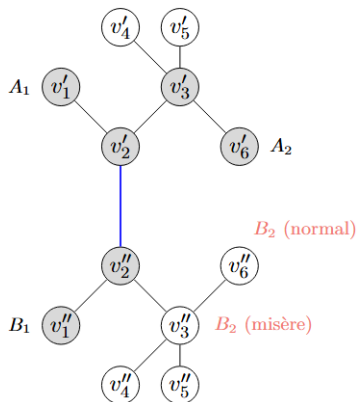
[Araújo, Brito, Folz, Freitas, Sampaio, 2024]: CIG e CHG são polinomiais em árvores nas variantes normal e misère (**Teoria de Sprague-Grundy**).



$$\begin{aligned}
 \text{mex}\{1 \oplus 1, 1 \oplus 0\} &= \text{mex}\{0, 1\} = 2 \\
 \text{mex}\{2 \oplus 2, 2 \oplus 1, 2 \oplus 0\} &= \text{mex}\{0, 3, 2\} = 1 \\
 \text{mex}\{3 \oplus 3, 3 \oplus 2, 3 \oplus 1, 3 \oplus 0\} &= \text{mex}\{0, 1, 2, 3\} = 4 \\
 \text{mex}\{4 \oplus 4, 4 \oplus 3, 4 \oplus 2, 4 \oplus 1, 4 \oplus 0\} &= \text{mex}\{0, 7, 6, 5, 4\} = 1
 \end{aligned}$$

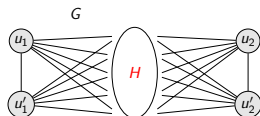
Jogos de Convexidade em Árvore - CIG e CHG

[Araújo, Brito, Folz, Freitas, Sampaio, 2024]: Bob vence CIG (e CHG) nas variantes normal e misère em árvores simétricas.

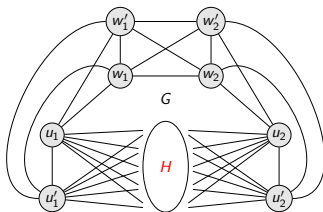


Jogos de Convexidade - HG e CHG são PSPACE-completos

[Araújo, Brito, Folz, Freitas, Sampaio, 2024]: Os jogos HG e CHG são PSPACE-completos nas variantes normal e misère, mesmo em grafos com diâmetro 2.



Grafos G da redução do Jogo CLIQUE-FORMING para HG (ou CHG) na variante misère (acima) e normal (abaixo).



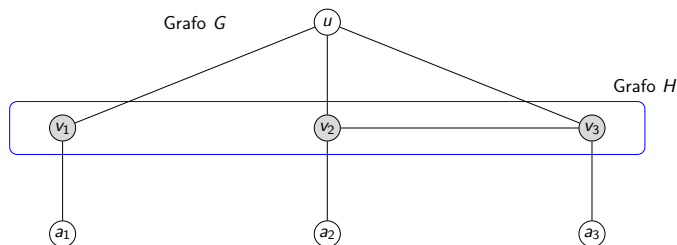
Jogos de Convexidade recentes: Posição Geral GPG

Pontos no plano em posição geral: não há 3 numa mesma linha.

Vértices no grafo em posição geral: não há 3 em um caminho mínimo.

GPG (**General Position Game**): Alice e Bob selecionam vértices que devem sempre estar em posição geral.

[Chandran, Klavžar, Neethu P. K., Sampaio, 2024]: O jogo GPG é PSPACE-completo (normal e misère), mesmo em grafos com diâmetro 4.



Redução de CLIQUE-FORMING em H para GPG-normal em G . Alice vence GPG em G se e só se Bob vence CLIQUE-FORMING em H .

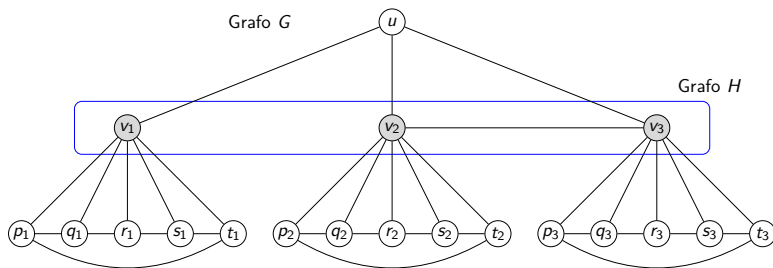
Jogos de Convexidade recentes: Posição Geral GPG

Pontos no plano em posição geral: não há 3 numa mesma linha.

Vértices no grafo em posição geral: não há 3 em um caminho mínimo.

GPG (**General Position Game**): Alice e Bob selecionam vértices que devem sempre estar em posição geral.

[Chandran, Klavžar, Neethu P. K., Sampaio, 2024]: O jogo GPG é PSPACE-completo (normal e misère), mesmo em grafos com diâmetro 4.



Redução de CLIQUE-FORMING-misère p/ GPG-misère. Alice vence GPG-m em G se e só se Bob vence CLIQUE-FORMING-m em H .

Jogos recentes: CSFG *Convex Set Forming Game*

CSFG (*Convex Set Forming Game*): Alice e Bob selecionam vértices que devem sempre formar um conjunto convexo.

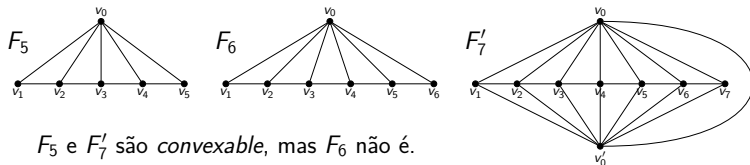
Semelhante ao **CLIQUE-FORMING**, que deve sempre formar uma clique.

[Brosse, Martins, Nisse, Sampaio, 2025]: CSFG é PSPACE-completo, mesmo para diâmetro 2, nas variantes normal, misère e de otimização.

G é **convexable** se $\text{gcn}(G) = |V(G)|$, ou seja, Alice consegue forçar a seleção de todos os vértices durante o jogo.

[Brosse et al., 2025]: Todo grafo *convexable* é cordal.

Exemplos de cordais *convexable* e de cordais não-*convexable*, onde F_k e F'_k são obtidos de P_k com 1 ou 2 vértices universais.



F_5 e F'_7 são *convexable*, mas F_6 não é.

Jogos recentes: CSFG *Convex Set Forming Game*

CSFG (*Convex Set Forming Game*): Alice e Bob selecionam vértices que devem sempre formar um conjunto convexo.

G é *convexable* se $\text{gcn}(G) = |V(G)|$, ou seja, Alice consegue forçar a seleção de todos os vértices durante o jogo.

[Brosse et al., 2025]: Todo grafo *convexable* é cordal.

Todo grafo ptolemaico G é *convexable*., ou seja, $\text{gcn}(G) = n$.

[Brosse et al., 2025]: Seja G um grafo ptolemaico. Alice vence CSFG_g na variante normal (resp. *misère*) se e só se n é ímpar (resp. par).

Caminhos mínimos (convexidade geodésica)

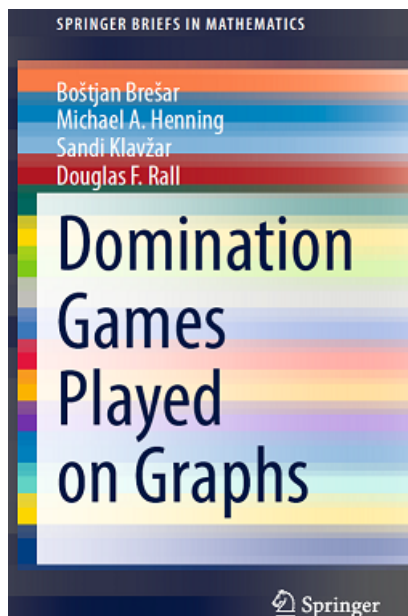
Jogos em outras convexidades: Monofônica (caminhos induzidos), P_3 , ...

Capítulo 8: Jogos de Dominação em Grafos

Capítulo 8

Jogos de Dominação em Grafos

Capítulo 8: Jogos de Dominação em Grafos



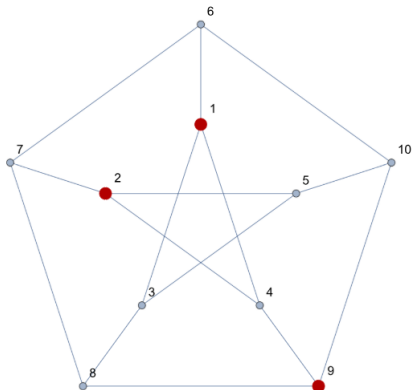
Jogo de Dominação em Grafos: definições

Um vértice x **domina** outro vértice y em um grafo se xy é uma aresta.

Um conjunto S de vértices é **dominante** se todo vértice fora de S é dominado por algum em S .

Domination number $\gamma(G)$: tamanho do menor conjunto dominante de G .

Exemplo: $\gamma(K_n) = 1$. $\gamma(P_n) = \gamma(C_n) = \lceil n/3 \rceil$.



Jogo de Dominação em Grafos: definições

Domination number $\gamma(G)$: tamanho do menor conjunto dominante de G .

Exemplo: $\gamma(K_n) = 1$. $\gamma(P_n) = \gamma(C_n) = \lceil n/3 \rceil$.

Jogo de Dominação: Alice e Bob se alternam escolhendo em cada turno um vértice jogável (domina pelo menos um vértice não dominado pelos vértices escolhidos antes). O jogo termina quando está tudo dominado.

Variante Normal: o último a jogar vence.

Variante Misère: o último a jogar perde.

Variante de Otimização: Alice vence se forem escolhidos $\leq k$ vértices.

Game Domination number $\gamma_g(G)$: menor k tal que Alice vence a variante de otimização do Jogo de Dominação em G . **Exemplo:**

$$\gamma_g(P_n) = \gamma_g(C_n) = \begin{cases} \lceil \frac{n}{2} \rceil - 1, & \text{se } n \% 4 = 3, \\ \lceil \frac{n}{2} \rceil, & \text{caso contrário.} \end{cases}$$

Jogo de Dominação em Grafos: Teorema 2/3

Teo [Kinnersley, West, Zamani, 2013]: Se G não tem vértices isolados,

$$\gamma_g(G) \leq \frac{2}{3} \cdot n$$

Ideia da prova: Técnica do Potencial.

Durante o jogo, um vértice é **P2** se não é dominado, **P1** se é dominado mas tem vizinho não dominado, e **P0** se é dominado e todo vizinho está dominado.

Potencial de G : soma dos potenciais dos vértices (inicial $2n$, todos P2) e termina 0 (todos P0). Como G não tem vértices isolados, nunca haverá um vértice P2 isolado, pois seus vizinhos sempre serão P1 ou P2.

Vértices P0 não são jogáveis. Arestas entre vértices P1 são irrelevantes (pois já estão dominados). Logo, após cada jogada, podemos remover vértices P0 e arestas entre vértices P1. Ou seja, existem apenas vértices P1 ou P2 e só arestas P1-P2 ou P2-P2.

Jogo de Dominação em Grafos: Teorema 2/3

Teo 2/3 de 2013: Se G não tem isolados, $\gamma_g(G) \leq (2/3) \cdot n$.

Prova: P2 (vértice não dominado), P1 (dominado com vizinho não dominado), e P0 (dominado e todo vizinho dominado).

Grafo G : Potencial inicial $2n$, final 0.

Só vértices P1 ou P2. Só arestas P1-P2 ou P2-P2.

Toda jogada reduz o potencial em ≥ 2 , pois a seleção de P2 o torna P0 e a seleção de P1 o torna P0 bem como seu vizinho P2 se torna P0 ou P1.

Fase 1: há jogada que reduz o potencial em ≥ 4 . Senão **Fase 2**.

Fase 2 $\Rightarrow$ (a) P2 tem ≤ 1 vizinho P2 $\Rightarrow$ (b) P2 só tem vizinho P1
 $\Rightarrow$ (c) P1 tem $=1$ vizinho P2 $\Rightarrow$ (d) P2 tem $=1$ vizinho P1

Conclusão: **Fase 2** só tem componentes P1-P2 $\rightarrow$ toda jogada reduz o potencial em 3. **Fase 1:** Alice reduz em ≥ 4 e Bob reduz em ≥ 2 (média 3). Potencial inicial $2n$ e final 0 implica no máximo $n/3$ pares de jogadas. Ou seja, $2n/3$ vértices escolhidos.

Jogo de Dominação em Grafos: Conjectura 3/5

Teo [Versteegen, 2024]: Se G não tem vértices isolados,

$$\gamma_g(G) \leq \frac{3}{5} \cdot n$$

Prova de 33 páginas: Técnica do Potencial.

11 estados, ao invés de 3 estados P_0 , P_1 e P_2 .

Jogo NORMAL de Dominação em Grafos

Variante Normal: vence o último a jogar. Sprague-Grundy.

P_n : caminho com n vértices não selecionados.

P'_n : caminho com $n + 2$ vértices e só 1 extremidade selecionada.

P''_n : caminho com $n + 4$ vértices e as duas extremidades selecionadas.

$$\text{nim}(P_n) = \text{mex} \left\{ \text{nim}(P'_{n-2}), \text{nim}(P'_{n-3}), \right. \\ \left. \text{nim}(P'_{k-2}) \oplus \text{nim}(P'_{n-k-1}) : k = 3, \dots, \lceil n/2 \rceil \right\}$$

$$\text{nim}(P'_n) = \text{mex} \left\{ \text{nim}(P'_{n-1}), \text{nim}(P'_{n-2}), \text{nim}(P'_{n-3}), \text{nim}(P''_{n-2}), \text{nim}(P''_{n-3}), \right. \\ \left. \text{nim}(P''_{k-2}) \oplus \text{nim}(P'_{n-k-1}) : k = 3, \dots, n-2 \right\}$$

$$\text{nim}(P''_n) = \text{mex} \left\{ \text{nim}(P''_{n-1}), \text{nim}(P''_{n-2}), \text{nim}(P''_{n-3}), \right. \\ \left. \text{nim}(P''_{k-2}) \oplus \text{nim}(P''_{n-k-1}) : k = 3, \dots, \lceil n/2 \rceil \right\}$$

Jogo NORMAL de Dominação em Grafos

P_n , P'_n e P''_n tem n vértices não dominados.

$$\text{nim}(P_n) = \text{mex} \left\{ \text{nim}(P'_{n-2}), \text{nim}(P'_{n-3}), \right. \\ \left. \text{nim}(P'_{k-2}) \oplus \text{nim}(P'_{n-k-1}) : k = 3, \dots, \lceil n/2 \rceil \right\}$$

$$\text{nim}(P'_n) = \text{mex} \left\{ \text{nim}(P'_{n-1}), \text{nim}(P'_{n-2}), \text{nim}(P'_{n-3}), \text{nim}(P''_{n-2}), \text{nim}(P''_{n-3}), \right. \\ \left. \text{nim}(P''_{k-2}) \oplus \text{nim}(P'_{n-k-1}) : k = 3, \dots, n-2 \right\}$$

$$\text{nim}(P''_n) = \text{mex} \left\{ \text{nim}(P''_{n-1}), \text{nim}(P''_{n-2}), \text{nim}(P''_{n-3}), \right. \\ \left. \text{nim}(P''_{k-2}) \oplus \text{nim}(P''_{n-k-1}) : k = 3, \dots, \lceil n/2 \rceil \right\}$$

Teorema [Brito, Marcilon, Martins, Sampaio, 2025]:

$\text{nim}(P'_n) = \text{nim}(P''_n) = n \% 4$. $\text{nim}(P_n) = 1, 1, 2, 0, 1, 1, 3, 0, 1, 1, 3, \dots$

Bob vence em P_n se e só se $n \% 4 = 0$.

Bob vence em C_n se e só se $n \% 4 = 3$.

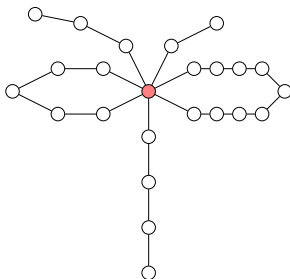
Jogo NORMAL de Dominação em Grafos

Teorema [Brito, Marcilon, Martins, Sampaio, 2025]: $\text{nim}(P'_n) = \text{nim}(P''_n) = n \% 4$.

$\text{nim}(P_n) = 1, 1, 2, 0, 1, 1, 3, 0, 1, 1, 3, \dots$

Bob vence em P_n se e só se $n \% 4 = 0$.

Bob vence em C_n se e só se $n \% 4 = 3$.

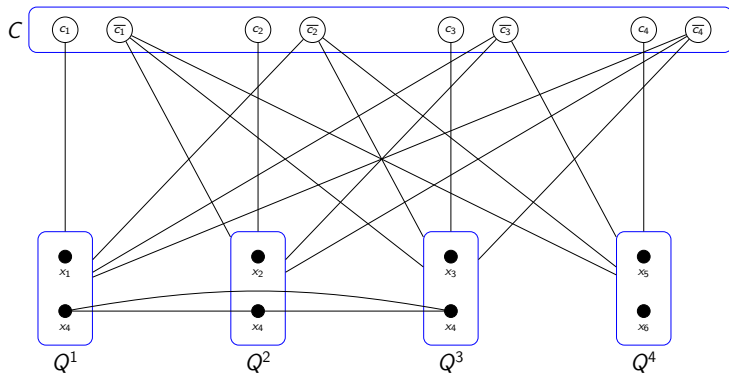


Outra aplicação dos Nimbers: Alice vence selecionando o vértice **vermelho** em sua primeira jogada, pois retorna uma posição equivalente a união disjunta de P'_1 , P'_2 , P'_3 , P'_3 e P''_7 , com nimber igual $1 \oplus 2 \oplus 3 \oplus 3 \oplus 3 = 0$.

Jogo NORMAL de Dominação é PSPACE-completo

Teorema [Brito, Marcilon, Martins, Sampaio, 2025]:

Jogo NORMAL de Dominação é PSPACE-completo.



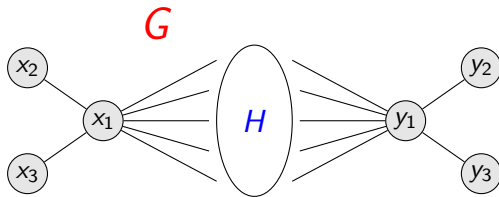
Redução do Jogo Avoid-POSDNF-2 p/ Jogo Normal de Dominação.

Exemplo para a fórmula $(x_1 \wedge x_4) \vee (x_2 \wedge x_4) \vee (x_3 \wedge x_4) \vee (x_5 \wedge x_6)$, onde C é uma clique, mas Q^1, Q^2, Q^3 e Q^4 são conjuntos independentes.

Jogo MISÈRE de Dominação é PSPACE-completo

Teorema [Brito, Marcilon, Martins, Sampaio, 2025]:

Jogo MISÈRE de Dominação é PSPACE-completo.



Redução do Jogo Normal p / Jogo Misère de Dominação.

FIM DA AULA 04

FIM DA AULA 04

Perguntas (se houver tempo) ??

Teoria dos Jogos Combinatórios em Grafos

Aula 05

**Samuel Araújo, Nicolas Martins,
Nicolas Nisse, Rudini Sampaio**

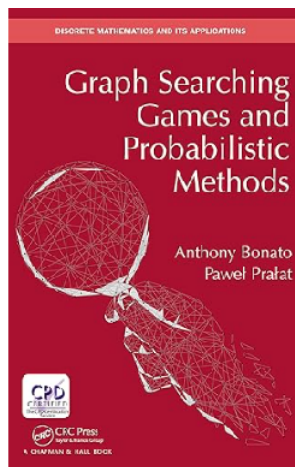
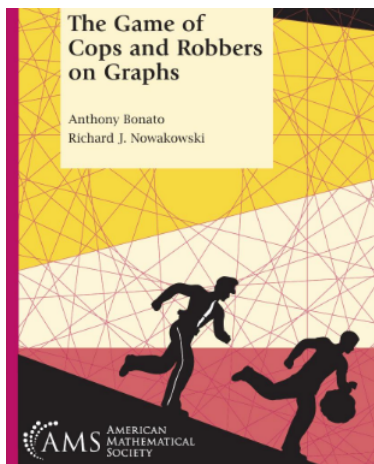
CBM-2025, IMPA,
Rio de Janeiro, sexta, 01-Agosto, 8h

Capítulo 9 - Jogos de Polícia e Ladrão em Grafos

Capítulo 9

Jogos de Polícia e Ladrão em Grafos

Livros sobre Jogos de Perseguição em Grafos (2011 e 2017)



O Jogo de Polícia e Ladrão ($\mathcal{C}\&\mathcal{R}$)

- ▶ **Entrada:** Grafo G , inteiro k ;
- ▶ $\mathcal{C}$ controla os policiais;
- ▶ $\mathcal{R}$ controla o ladrão.

O Jogo de Polícia e Ladrão ($\mathcal{C}\&\mathcal{R}$)

- ▶ **Entrada:** Grafo G , inteiro k ;
- ▶ $\mathcal{C}$ controla os policiais;
- ▶ $\mathcal{R}$ controla o ladrão.

O Jogo de Polícia e Ladrão ($\mathcal{C}$ & $\mathcal{R}$)

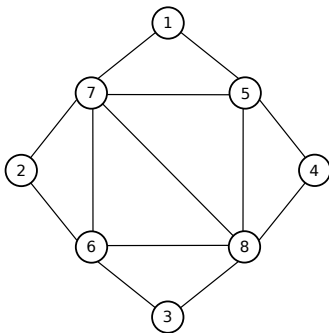
- ▶ **Entrada:** Grafo G , inteiro k ;
- ▶ $\mathcal{C}$ controla os policiais;
- ▶ $\mathcal{R}$ controla o ladrão.

O Jogo de Polícia e Ladrão ($\mathcal{C}$ & $\mathcal{R}$)

- ▶ **Entrada:** Grafo G , inteiro k ;
- ▶ $\mathcal{C}$ controla os policiais;
- ▶ $\mathcal{R}$ controla o ladrão.

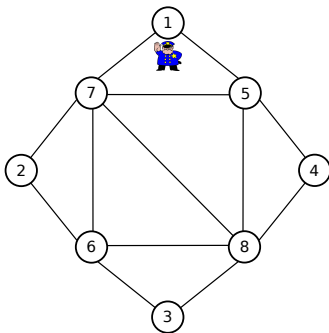
O Jogo de Polícia e Ladrão ($\mathcal{C}$ & $\mathcal{R}$)

- ▶ **Entrada:** Grafo G , inteiro k ;
- ▶ $\mathcal{C}$ controla os policiais;
- ▶ $\mathcal{R}$ controla o ladrão.



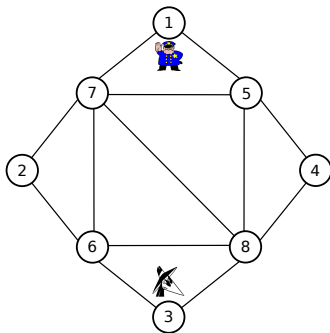
O Jogo de Polícia e Ladrão ($\mathcal{C}$ & $\mathcal{R}$)

- ▶ **Entrada:** Grafo G , inteiro k ;
- ▶ $\mathcal{C}$ controla os policiais;
- ▶ $\mathcal{R}$ controla o ladrão.



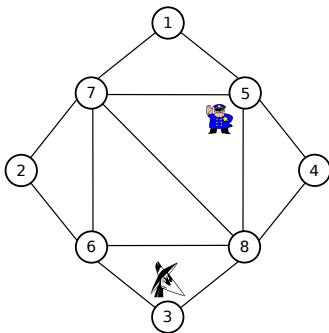
O Jogo de Polícia e Ladrão ($\mathcal{C}$ & $\mathcal{R}$)

- ▶ **Entrada:** Grafo G , inteiro k ;
- ▶ $\mathcal{C}$ controla os policiais;
- ▶ $\mathcal{R}$ controla o ladrão.



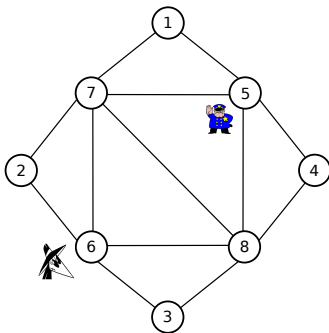
O Jogo de Polícia e Ladrão ($\mathcal{C}$ & $\mathcal{R}$)

- ▶ **Entrada:** Grafo G , inteiro k ;
- ▶ $\mathcal{C}$ controla os policiais;
- ▶ $\mathcal{R}$ controla o ladrão.



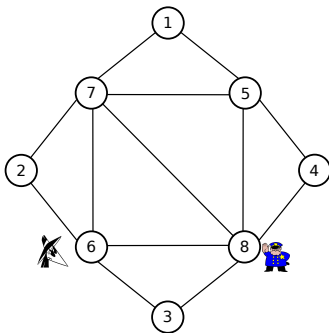
O Jogo de Polícia e Ladrão ($\mathcal{C}$ & $\mathcal{R}$)

- ▶ **Entrada:** Grafo G , inteiro k ;
- ▶ $\mathcal{C}$ controla os policiais;
- ▶ $\mathcal{R}$ controla o ladrão.



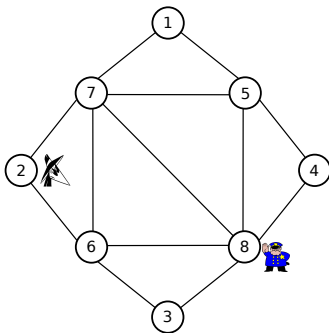
O Jogo de Polícia e Ladrão ($\mathcal{C}$ & $\mathcal{R}$)

- ▶ **Entrada:** Grafo G , inteiro k ;
- ▶ $\mathcal{C}$ controla os policiais;
- ▶ $\mathcal{R}$ controla o ladrão.



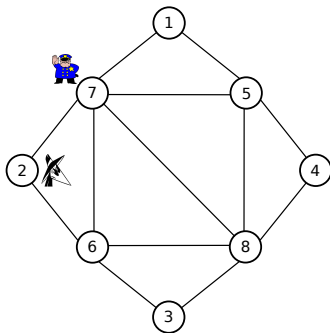
O Jogo de Polícia e Ladrão ($\mathcal{C}$ & $\mathcal{R}$)

- ▶ **Entrada:** Grafo G , inteiro k ;
- ▶ $\mathcal{C}$ controla os policiais;
- ▶ $\mathcal{R}$ controla o ladrão.



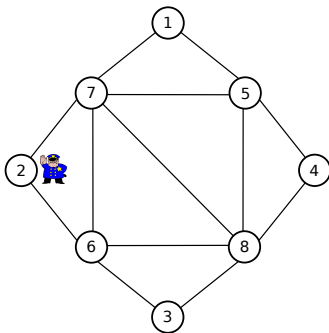
O Jogo de Polícia e Ladrão ($\mathcal{C}$ & $\mathcal{R}$)

- ▶ **Entrada:** Grafo G , inteiro k ;
- ▶ $\mathcal{C}$ controla os policiais;
- ▶ $\mathcal{R}$ controla o ladrão.



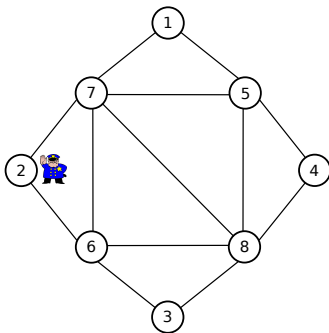
O Jogo de Polícia e Ladrão ($\mathcal{C}$ & $\mathcal{R}$)

- ▶ **Entrada:** Grafo G , inteiro k ;
- ▶ $\mathcal{C}$ controla os policiais;
- ▶ $\mathcal{R}$ controla o ladrão.



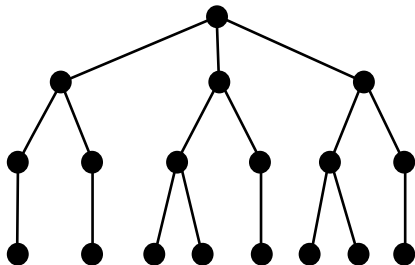
O Jogo de Polícia e Ladrão ($\mathcal{C}$ & $\mathcal{R}$)

- ▶ **Entrada:** Grafo G , inteiro k ;
- ▶ $\mathcal{C}$ controla os policiais;
- ▶ $\mathcal{R}$ controla o ladrão.

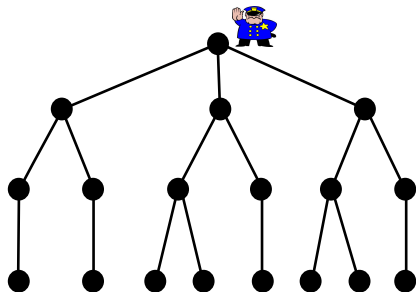


- ▶ $\mathcal{C}$ vence se eventualmente um policial ocupa o mesmo vértice do ladrão. Caso contrário vence $\mathcal{R}$. (Nowakowski e Winkler, 1983) e (Quilliot, 1983)

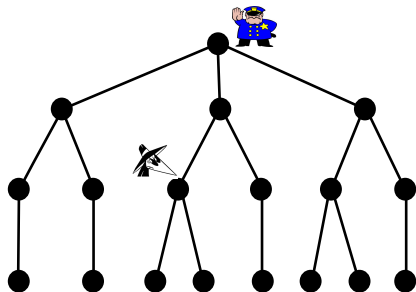
$\mathcal{C}\&\mathcal{R}$ em Árvores



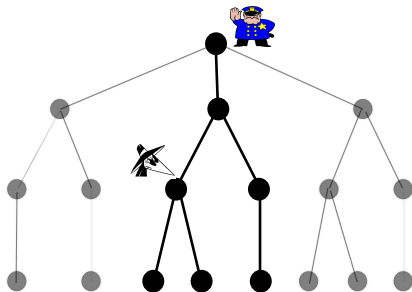
$\mathcal{C}\&\mathcal{R}$ em Árvores



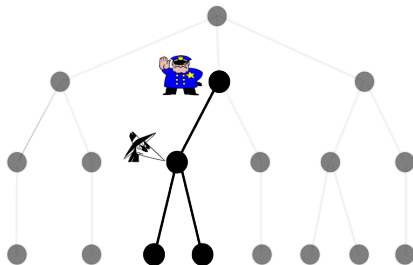
$\mathcal{C}\&\mathcal{R}$ em Árvores



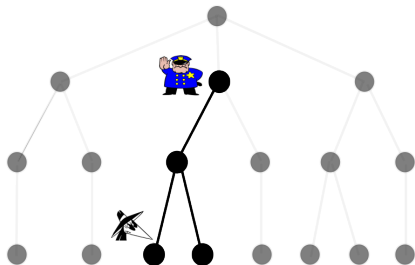
$C\&R$ em Árvores



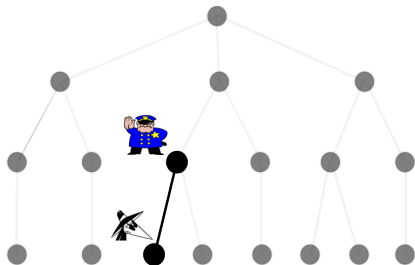
$C\&R$ em Árvores



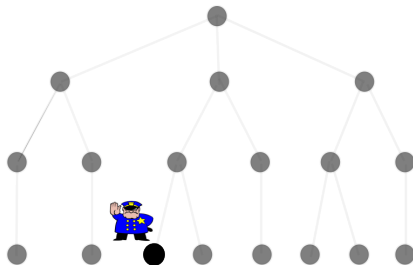
$\mathcal{C}\&\mathcal{R}$ em Árvores



$\mathcal{C}\&\mathcal{R}$ em Árvores



$\mathcal{C}\&\mathcal{R}$ em Árvores



Grafos Cop-Win e o Cop-number

- ▶ Certamente se $k = n$, então $\mathcal{C}$ sempre vence;
- ▶ O cop-number de um grafo G , $cn(G)$, é o menor inteiro k tal que $\mathcal{C}$ vence $\mathcal{C}\&\mathcal{R}$ com k policiais em G ;
- ▶ Um grafo G é cop-win se $cn(G) = 1$.

Lema: Se um grafo G é cop-win, então existem vértices $x, y \in V(G)$ tais que $N[y] \subseteq N[x]$.

- ▶ Considere a posição do policial e do ladrão logo após o penúltimo movimento do policial antes da captura.

Grafos Cop-Win e o Cop-number

- ▶ Certamente se $k = n$, então $\mathcal{C}$ sempre vence;
- ▶ O cop-number de um grafo G , $cn(G)$, é o menor inteiro k tal que $\mathcal{C}$ vence $\mathcal{C}\&\mathcal{R}$ com k policiais em G ;
- ▶ Um grafo G é cop-win se $cn(G) = 1$.

Lema: Se um grafo G é cop-win, então existem vértices $x, y \in V(G)$ tais que $N[y] \subseteq N[x]$.

- ▶ Considere a posição do policial e do ladrão logo após o penúltimo movimento do policial antes da captura.

Grafos Cop-Win e o Cop-number

- ▶ Certamente se $k = n$, então $\mathcal{C}$ sempre vence;
- ▶ O cop-number de um grafo G , $cn(G)$, é o menor inteiro k tal que $\mathcal{C}$ vence $\mathcal{C}\&\mathcal{R}$ com k policiais em G ;
- ▶ Um grafo G é cop-win se $cn(G) = 1$.

Lema: Se um grafo G é cop-win, então existem vértices $x, y \in V(G)$ tais que $N[y] \subseteq N[x]$.

- ▶ Considere a posição do policial e do ladrão logo após o penúltimo movimento do policial antes da captura.

Grafos Cop-Win e o Cop-number

- ▶ Certamente se $k = n$, então $\mathcal{C}$ sempre vence;
- ▶ O cop-number de um grafo G , $cn(G)$, é o menor inteiro k tal que $\mathcal{C}$ vence $\mathcal{C}\&\mathcal{R}$ com k policiais em G ;
- ▶ Um grafo G é cop-win se $cn(G) = 1$.

Lema: Se um grafo G é cop-win, então existem vértices $x, y \in V(G)$ tais que $N[y] \subseteq N[x]$.

- ▶ Considere a posição do policial e do ladrão logo após o penúltimo movimento do policial antes da captura.

Grafos Cop-Win e o Cop-number

- ▶ Certamente se $k = n$, então $\mathcal{C}$ sempre vence;
- ▶ O cop-number de um grafo G , $cn(G)$, é o menor inteiro k tal que $\mathcal{C}$ vence $\mathcal{C}\&\mathcal{R}$ com k policiais em G ;
- ▶ Um grafo G é cop-win se $cn(G) = 1$.

Lema: Se um grafo G é cop-win, então existem vértices $x, y \in V(G)$ tais que $N[y] \subseteq N[x]$.

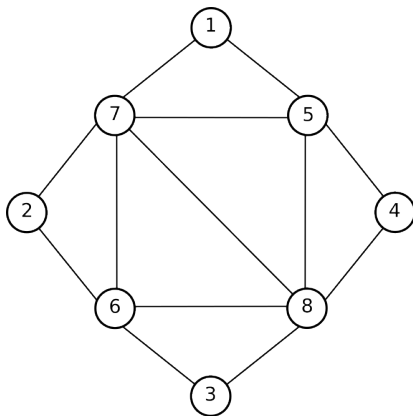
- ▶ Considere a posição do policial e do ladrão logo após o penúltimo movimento do policial antes da captura.

Grafos Desmontáveis

- ▶ Um grafo G é **desmontável** se há uma ordem $(v_1, v_2, \dots, v_n)$ de seus vértices tal que $N[v_i] - \{v_1, \dots, v_{i-1}\}$ está contido em algum $N[v_j]$ com $j > i$ para todo $1 \leq i < n$.

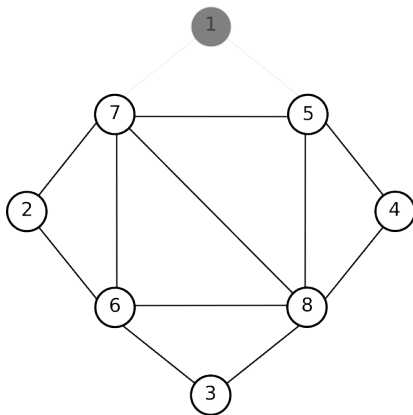
Grafos Desmontáveis

- Um grafo G é **desmontável** se há uma ordem $(v_1, v_2, \dots, v_n)$ de seus vértices tal que $N[v_i] - \{v_1, \dots, v_{i-1}\}$ está contido em algum $N[v_j]$ com $j > i$ para todo $1 \leq i < n$.



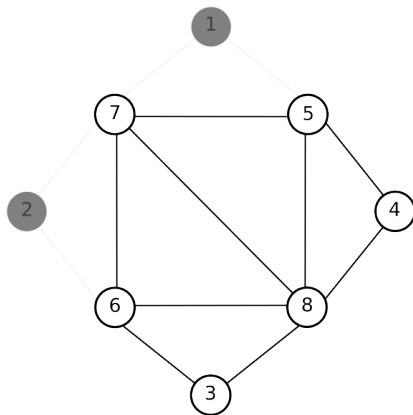
Grafos Desmontáveis

- Um grafo G é **desmontável** se há uma ordem $(v_1, v_2, \dots, v_n)$ de seus vértices tal que $N[v_i] - \{v_1, \dots, v_{i-1}\}$ está contido em algum $N[v_j]$ com $j > i$ para todo $1 \leq i < n$.



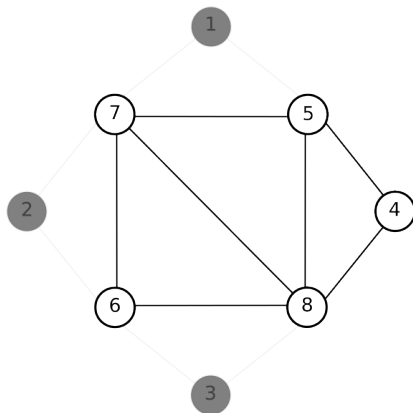
Grafos Desmontáveis

- Um grafo G é **desmontável** se há uma ordem $(v_1, v_2, \dots, v_n)$ de seus vértices tal que $N[v_i] - \{v_1, \dots, v_{i-1}\}$ está contido em algum $N[v_j]$ com $j > i$ para todo $1 \leq i < n$.



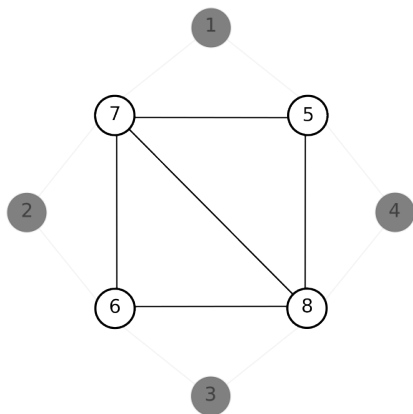
Grafos Desmontáveis

- Um grafo G é **desmontável** se há uma ordem $(v_1, v_2, \dots, v_n)$ de seus vértices tal que $N[v_i] - \{v_1, \dots, v_{i-1}\}$ está contido em algum $N[v_j]$ com $j > i$ para todo $1 \leq i < n$.



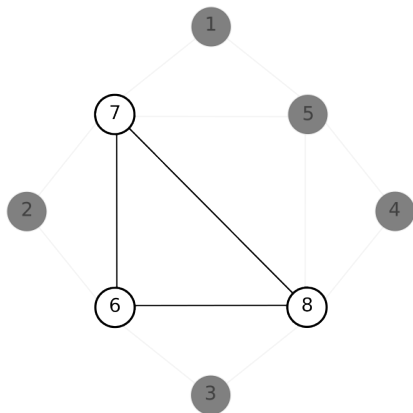
Grafos Desmontáveis

- Um grafo G é **desmontável** se há uma ordem $(v_1, v_2, \dots, v_n)$ de seus vértices tal que $N[v_i] - \{v_1, \dots, v_{i-1}\}$ está contido em algum $N[v_j]$ com $j > i$ para todo $1 \leq i < n$.



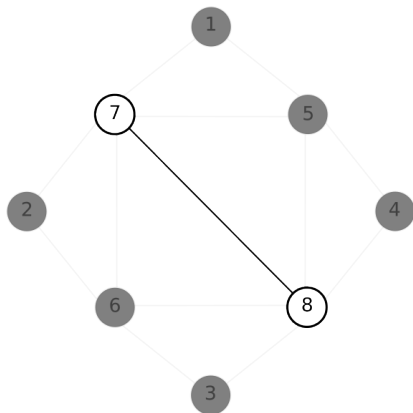
Grafos Desmontáveis

- Um grafo G é **desmontável** se há uma ordem $(v_1, v_2, \dots, v_n)$ de seus vértices tal que $N[v_i] - \{v_1, \dots, v_{i-1}\}$ está contido em algum $N[v_j]$ com $j > i$ para todo $1 \leq i < n$.



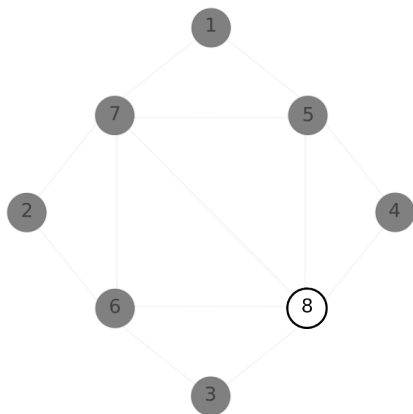
Grafos Desmontáveis

- Um grafo G é **desmontável** se há uma ordem $(v_1, v_2, \dots, v_n)$ de seus vértices tal que $N[v_i] - \{v_1, \dots, v_{i-1}\}$ está contido em algum $N[v_j]$ com $j > i$ para todo $1 \leq i < n$.



Grafos Desmontáveis

- Um grafo G é **desmontável** se há uma ordem $(v_1, v_2, \dots, v_n)$ de seus vértices tal que $N[v_i] - \{v_1, \dots, v_{i-1}\}$ está contido em algum $N[v_j]$ com $j > i$ para todo $1 \leq i < n$.



Grafos Desmontáveis

- ▶ Um grafo G é **desmontável** se há uma ordem $(v_1, v_2, \dots, v_n)$ de seus vértices tal que $N[v_i] - \{v_1, \dots, v_{i-1}\}$ está contido em algum $N[v_j]$ com $j > i$ para todo $1 \leq i < n$.

Lema: Se G é cop-win, então G é desmontável.

- ▶ Por indução no número de vértices de G (Base: $n = 1$).
- ▶ Sejam $x, y \in V(G)$ tais que $N[y] \subseteq N[x]$.
- ▶ Se G é cop-win então $G' = G[V \setminus \{y\}]$ é cop-win. (Por quê?)
- ▶ Por indução G' é desmontável. Adicionando y como primeiro vértice da ordem de G' mostramos que G também é desmontável.

Grafos Desmontáveis

- ▶ Um grafo G é **desmontável** se há uma ordem $(v_1, v_2, \dots, v_n)$ de seus vértices tal que $N[v_i] - \{v_1, \dots, v_{i-1}\}$ está contido em algum $N[v_j]$ com $j > i$ para todo $1 \leq i < n$.

Lema: Se G é cop-win, então G é desmontável.

- ▶ Por indução no número de vértices de G (Base: $n = 1$).
- ▶ Sejam $x, y \in V(G)$ tais que $N[y] \subseteq N[x]$.
- ▶ Se G é cop-win então $G' = G[V \setminus \{y\}]$ é cop-win. (Por quê?)
- ▶ Por indução G' é desmontável. Adicionando y como primeiro vértice da ordem de G' mostramos que G também é desmontável.

Grafos Desmontáveis

- ▶ Um grafo G é **desmontável** se há uma ordem $(v_1, v_2, \dots, v_n)$ de seus vértices tal que $N[v_i] - \{v_1, \dots, v_{i-1}\}$ está contido em algum $N[v_j]$ com $j > i$ para todo $1 \leq i < n$.

Lema: Se G é cop-win, então G é desmontável.

- ▶ Por indução no número de vértices de G (Base: $n = 1$).
- ▶ Sejam $x, y \in V(G)$ tais que $N[y] \subseteq N[x]$.
- ▶ Se G é cop-win então $G' = G[V \setminus \{y\}]$ é cop-win. (Por quê?)
- ▶ Por indução G' é desmontável. Adicionando y como primeiro vértice da ordem de G' mostramos que G também é desmontável.

Grafos Desmontáveis

- ▶ Um grafo G é **desmontável** se há uma ordem $(v_1, v_2, \dots, v_n)$ de seus vértices tal que $N[v_i] - \{v_1, \dots, v_{i-1}\}$ está contido em algum $N[v_j]$ com $j > i$ para todo $1 \leq i < n$.

Lema: Se G é cop-win, então G é desmontável.

- ▶ Por indução no número de vértices de G (Base: $n = 1$).
- ▶ Sejam $x, y \in V(G)$ tais que $N[y] \subseteq N[x]$.
- ▶ Se G é cop-win então $G' = G[V \setminus \{y\}]$ é cop-win. (Por quê?)
- ▶ Por indução G' é desmontável. Adicionando y como primeiro vértice da ordem de G' mostramos que G também é desmontável.

Grafos Desmontáveis

- ▶ Um grafo G é **desmontável** se há uma ordem $(v_1, v_2, \dots, v_n)$ de seus vértices tal que $N[v_i] - \{v_1, \dots, v_{i-1}\}$ está contido em algum $N[v_j]$ com $j > i$ para todo $1 \leq i < n$.

Lema: Se G é cop-win, então G é desmontável.

- ▶ Por indução no número de vértices de G (Base: $n = 1$).
- ▶ Sejam $x, y \in V(G)$ tais que $N[y] \subseteq N[x]$.
- ▶ Se G é cop-win então $G' = G[V \setminus \{y\}]$ é cop-win. (Por quê?)
- ▶ Por indução G' é desmontável. Adicionando y como primeiro vértice da ordem de G' mostramos que G também é desmontável.

Grafos Desmontáveis

- ▶ Um grafo G é **desmontável** se há uma ordem $(v_1, v_2, \dots, v_n)$ de seus vértices tal que $N[v_i] - \{v_1, \dots, v_{i-1}\}$ está contido em algum $N[v_j]$ com $j > i$ para todo $1 \leq i < n$.

Lema: Se G é cop-win, então G é desmontável.

- ▶ Por indução no número de vértices de G (Base: $n = 1$).
- ▶ Sejam $x, y \in V(G)$ tais que $N[y] \subseteq N[x]$.
- ▶ Se G é cop-win então $G' = G[V \setminus \{y\}]$ é cop-win. (Por quê?)
- ▶ Por indução G' é desmontável. Adicionando y como primeiro vértice da ordem de G' mostramos que G também é desmontável.

Grafos Desmontáveis

- ▶ Um grafo G é **desmontável** se há uma ordem $(v_1, v_2, \dots, v_n)$ de seus vértices tal que $N[v_i] - \{v_1, \dots, v_{i-1}\}$ está contido em algum $N[v_j]$ com $j > i$ para todo $1 \leq i < n$.

Lema: Se G é desmontável, então G é cop-win.

- ▶ $\exists y \in V(G)$ tal que $G - \{y\}$ é desmontável e, por indução cop-win.
- ▶ $\mathcal{C}$ joga em G da mesma forma que faria em $G - \{y\}$. Se o ladrão for para y , o policial move-se como se ele estivesse no vértice que domina y .
- ▶ A estratégia vencedora em $G - \{y\}$ garante que em um número finito de turnos, ou o policial captura o ladrão, ou o ladrão está em y e o policial está em um vértice que domina y .

Grafos Desmontáveis

- ▶ Um grafo G é **desmontável** se há uma ordem $(v_1, v_2, \dots, v_n)$ de seus vértices tal que $N[v_i] - \{v_1, \dots, v_{i-1}\}$ está contido em algum $N[v_j]$ com $j > i$ para todo $1 \leq i < n$.

Lema: Se G é desmontável, então G é cop-win.

- ▶ $\exists y \in V(G)$ tal que $G - \{y\}$ é desmontável e, por indução cop-win.
- ▶ $\mathcal{C}$ joga em G da mesma forma que faria em $G - \{y\}$. Se o ladrão for para y , o policial move-se como se ele estivesse no vértice que domina y .
- ▶ A estratégia vencedora em $G - \{y\}$ garante que em um número finito de turnos, ou o policial captura o ladrão, ou o ladrão está em y e o policial está em um vértice que domina y .

Grafos Desmontáveis

- ▶ Um grafo G é **desmontável** se há uma ordem $(v_1, v_2, \dots, v_n)$ de seus vértices tal que $N[v_i] - \{v_1, \dots, v_{i-1}\}$ está contido em algum $N[v_j]$ com $j > i$ para todo $1 \leq i < n$.

Lema: Se G é desmontável, então G é cop-win.

- ▶ $\exists y \in V(G)$ tal que $G - \{y\}$ é desmontável e, por indução cop-win.
- ▶ $\mathcal{C}$ joga em G da mesma forma que faria em $G - \{y\}$. Se o ladrão for para y , o policial move-se como se ele estivesse no vértice que domina y .
- ▶ A estratégia vencedora em $G - \{y\}$ garante que em um número finito de turnos, ou o policial captura o ladrão, ou o ladrão está em y e o policial está em um vértice que domina y .

Grafos Desmontáveis

- ▶ Um grafo G é **desmontável** se há uma ordem $(v_1, v_2, \dots, v_n)$ de seus vértices tal que $N[v_i] - \{v_1, \dots, v_{i-1}\}$ está contido em algum $N[v_j]$ com $j > i$ para todo $1 \leq i < n$.

Lema: Se G é desmontável, então G é cop-win.

- ▶ $\exists y \in V(G)$ tal que $G - \{y\}$ é desmontável e, por indução cop-win.
- ▶ $\mathcal{C}$ joga em G da mesma forma que faria em $G - \{y\}$. Se o ladrão for para y , o policial move-se como se ele estivesse no vértice que domina y .
- ▶ A estratégia vencedora em $G - \{y\}$ garante que em um número finito de turnos, ou o policial captura o ladrão, ou o ladrão está em y e o policial está em um vértice que domina y .

Grafos Desmontáveis

- ▶ Um grafo G é **desmontável** se há uma ordem $(v_1, v_2, \dots, v_n)$ de seus vértices tal que $N[v_i] - \{v_1, \dots, v_{i-1}\}$ está contido em algum $N[v_j]$ com $j > i$ para todo $1 \leq i < n$.

Lema: Se G é desmontável, então G é cop-win.

- ▶ $\exists y \in V(G)$ tal que $G - \{y\}$ é desmontável e, por indução cop-win.
- ▶ $\mathcal{C}$ joga em G da mesma forma que faria em $G - \{y\}$. Se o ladrão for para y , o policial move-se como se ele estivesse no vértice que domina y .
- ▶ A estratégia vencedora em $G - \{y\}$ garante que em um número finito de turnos, ou o policial captura o ladrão, ou o ladrão está em y e o policial está em um vértice que domina y .

Grafos Desmontáveis

- ▶ Um grafo G é **desmontável** se há uma ordem $(v_1, v_2, \dots, v_n)$ de seus vértices tal que $N[v_i] - \{v_1, \dots, v_{i-1}\}$ está contido em algum $N[v_j]$ com $j > i$ para todo $1 \leq i < n$.

Teorema: Um grafo é cop-win se e somente se é desmontável.

- ▶ Árvores, grafos cordais e grafos bridged são exemplos de classes de grafos desmontáveis.
- ▶ Podemos determinar se um grafo é desmontável em tempo polinomial.

Grafos Desmontáveis

- ▶ Um grafo G é **desmontável** se há uma ordem $(v_1, v_2, \dots, v_n)$ de seus vértices tal que $N[v_i] - \{v_1, \dots, v_{i-1}\}$ está contido em algum $N[v_j]$ com $j > i$ para todo $1 \leq i < n$.

Teorema: Um grafo é cop-win se e somente se é desmontável.

- ▶ Árvores, grafos cordais e grafos bridged são exemplos de classes de grafos desmontáveis.
- ▶ Podemos determinar se um grafo é desmontável em tempo polinomial.

Grafos Desmontáveis

- ▶ Um grafo G é **desmontável** se há uma ordem $(v_1, v_2, \dots, v_n)$ de seus vértices tal que $N[v_i] - \{v_1, \dots, v_{i-1}\}$ está contido em algum $N[v_j]$ com $j > i$ para todo $1 \leq i < n$.

Teorema: Um grafo é cop-win se e somente se é desmontável.

- ▶ Árvores, grafos cordais e grafos bridged são exemplos de classes de grafos desmontáveis.
- ▶ Podemos determinar se um grafo é desmontável em tempo polinomial.

Limites para $cn(G)$ e a Conjectura de Meyniel

Teorema(Aigner e Fromme, 1984): Se G é um grafo com cintura (*girth*) pelo menos 5, então $cn(G) \geq \delta(G)$.

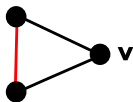
- ▶ Iremos mostrar uma estratégia vencedora para $\mathcal{R}$ contra $\delta(G) - 1$ policiais.
- ▶ Cada policial cobre no máximo um vizinho de um vértice não ocupado v .

Limites para $cn(G)$ e a Conjectura de Meyniel

Teorema(Aigner e Fromme, 1984): Se G é um grafo com cintura (*girth*) pelo menos 5, então $cn(G) \geq \delta(G)$.

- ▶ Iremos mostrar uma estratégia vencedora para $\mathcal{R}$ contra $\delta(G) - 1$ policiais.
- ▶ Cada policial cobre no máximo um vizinho de um vértice não ocupado v .

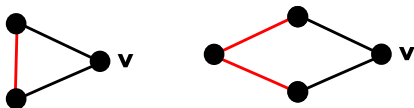
Limites para $cn(G)$ e a Conjectura de Meyniel



Teorema(Aigner e Fromme, 1984): Se G é um grafo com cintura (*girth*) pelo menos 5, então $cn(G) \geq \delta(G)$.

- Iremos mostrar uma estratégia vencedora para $\mathcal{R}$ contra $\delta(G) - 1$ policiais.
- Cada policial cobre no máximo um vizinho de um vértice não ocupado v .

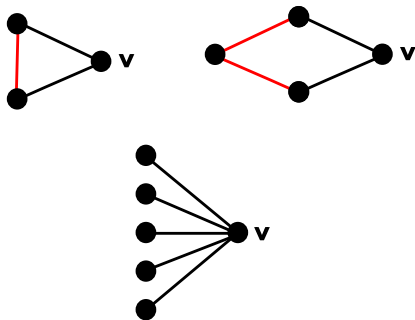
Limites para $cn(G)$ e a Conjectura de Meyniel



Teorema (Aigner e Fromme, 1984): Se G é um grafo com cintura (*girth*) pelo menos 5, então $cn(G) \geq \delta(G)$.

- ▶ Iremos mostrar uma estratégia vencedora para $\mathcal{R}$ contra $\delta(G) - 1$ policiais.
- ▶ Cada policial cobre no máximo um vizinho de um vértice não ocupado v .

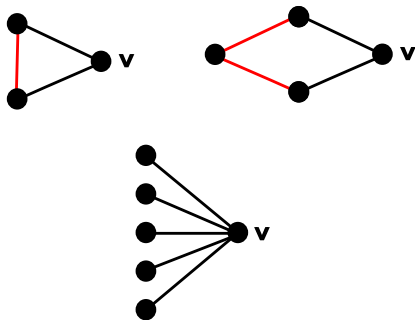
Limites para $cn(G)$ e a Conjectura de Meyniel



Teorema(Aigner e Fromme, 1984): Se G é um grafo com cintura (*girth*) pelo menos 5 então $cn(G) \geq \delta(G)$.

- ▶ Iremos mostrar uma estratégia vencedora para $\mathcal{R}$ contra $\delta(G) - 1$ policiais.
- ▶ Cada policial cobre no máximo um vizinho de um vértice não ocupado v .

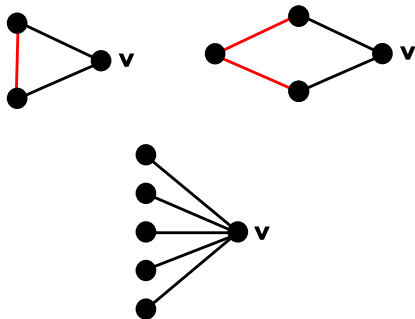
Limites para $cn(G)$ e a Conjectura de Meyniel



Teorema(Aigner e Fromme, 1984): Se G é um grafo com cintura (*girth*) pelo menos 5 então $cn(G) \geq \delta(G)$.

- ▶ Iremos mostrar uma estratégia vencedora para $\mathcal{R}$ contra $\delta(G) - 1$ policiais.
- ▶ Cada policial cobre no máximo um vizinho de um vértice não ocupado v .

Limites para $cn(G)$ e a Conjectura de Meyniel

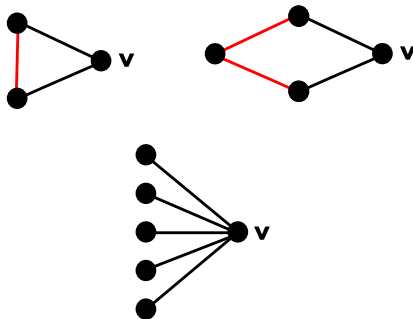


Teorema(Aigner e Fromme, 1984): Se G é um grafo com cintura (*girth*) pelo menos 5 então $cn(G) \geq \delta(G)$.

► **Teorema(Frankl, 1987):** Se G é um grafo com cintura (*girth*) pelo menos $8t - 3$, então $cn(G) > \delta(G)^t$.

Conjectura de Meyniel (1987) Para qualquer grafo G com n vértices, $cn(G) = O(\sqrt{n})$.

Limites para $cn(G)$ e a Conjectura de Meyniel

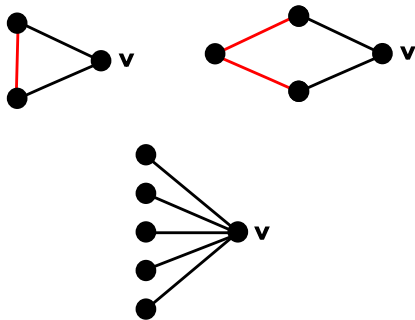


Teorema(Aigner e Fromme, 1984): Se G é um grafo com cintura (*girth*) pelo menos 5 então $cn(G) \geq \delta(G)$.

- **Teorema(Frankl, 1987):** Se G é um grafo com cintura (*girth*) pelo menos $8t - 3$, então $cn(G) > \delta(G)^t$.

Conjectura de Meyniel (1987) Para qualquer grafo G com n vértices, $cn(G) = O(\sqrt{n})$.

Limites para $cn(G)$ e a Conjectura de Meyniel



Teorema(Aigner e Fromme, 1984): Se G é um grafo com cintura (*girth*) pelo menos 5 então $cn(G) \geq \delta(G)$.

- **Teorema(Frankl, 1987):** Se G é um grafo com cintura (*girth*) pelo menos $8t - 3$, então $cn(G) > \delta(G)^t$.

Conjectura de Meyniel (1987) Para qualquer grafo G com n vértices, $cn(G) = O(\sqrt{n})$.

Complexidade Computacional de $\mathcal{C}\&\mathcal{R}$

Teorema(Goldstein e Reingold, 1995): $\mathcal{C}\&\mathcal{R}$ **com as posições iniciais fornecidas** é EXPTIME-completo.

Teorema(Fomin, Golovach e Kratochvíl, 2008): $\mathcal{C}\&\mathcal{R}$ é NP-difícil.

Teorema(Fomin, Golovach e Lokshtanov, 2010): $\mathcal{C}\&\mathcal{R}$ é PSPACE-completo **quando cada policial pode se mover um dado número de vezes.**

Teorema(Mamino, 2013): $\mathcal{C}\&\mathcal{R}$ é PSPACE-completo.

Teorema(Kinnersley, 2015): $\mathcal{C}\&\mathcal{R}$ é EXPTIME-completo.

Complexidade Computacional de $C\&R$

Teorema(Goldstein e Reingold, 1995): $C\&R$ **com as posições iniciais fornecidas** é EXPTIME-completo.

Teorema(Fomin, Golovach e Kratochvíl, 2008): $C\&R$ é NP-difícil.

Teorema(Fomin, Golovach e Lokshtanov, 2010): $C\&R$ é PSPACE-completo **quando cada policial pode se mover um dado número de vezes.**

Teorema(Mamino, 2013): $C\&R$ é PSPACE-completo.

Teorema(Kinnersley, 2015): $C\&R$ é EXPTIME-completo.

Complexidade Computacional de $\mathcal{C}\&\mathcal{R}$

Teorema(Goldstein e Reingold, 1995): $\mathcal{C}\&\mathcal{R}$ **com as posições iniciais fornecidas** é EXPTIME-completo.

Teorema(Fomin, Golovach e Kratochvíl, 2008): $\mathcal{C}\&\mathcal{R}$ é NP-difícil.

Teorema(Fomin, Golovach e Lokshtanov, 2010): $\mathcal{C}\&\mathcal{R}$ é PSPACE-completo **quando cada policial pode se mover um dado número de vezes.**

Teorema(Mamino, 2013): $\mathcal{C}\&\mathcal{R}$ é PSPACE-completo.

Teorema(Kinnersley, 2015): $\mathcal{C}\&\mathcal{R}$ é EXPTIME-completo.

Complexidade Computacional de $\mathcal{C}\&\mathcal{R}$

Teorema(Goldstein e Reingold, 1995): $\mathcal{C}\&\mathcal{R}$ **com as posições iniciais fornecidas** é EXPTIME-completo.

Teorema(Fomin, Golovach e Kratochvíl, 2008): $\mathcal{C}\&\mathcal{R}$ é NP-difícil.

Teorema(Fomin, Golovach e Lokshtanov, 2010): $\mathcal{C}\&\mathcal{R}$ é PSPACE-completo **quando cada policial pode se mover um dado número de vezes.**

Teorema(Mamino, 2013): $\mathcal{C}\&\mathcal{R}$ é PSPACE-completo.

Teorema(Kinnersley, 2015): $\mathcal{C}\&\mathcal{R}$ é EXPTIME-completo.

Complexidade Computacional de $\mathcal{C}\&\mathcal{R}$

Teorema(Goldstein e Reingold, 1995): $\mathcal{C}\&\mathcal{R}$ **com as posições iniciais fornecidas** é EXPTIME-completo.

Teorema(Fomin, Golovach e Kratochvíl, 2008): $\mathcal{C}\&\mathcal{R}$ é NP-difícil.

Teorema(Fomin, Golovach e Lokshtanov, 2010): $\mathcal{C}\&\mathcal{R}$ é PSPACE-completo **quando cada policial pode se mover um dado número de vezes.**

Teorema(Mamino, 2013): $\mathcal{C}\&\mathcal{R}$ é PSPACE-completo.

Teorema(Kinnersley, 2015): $\mathcal{C}\&\mathcal{R}$ é EXPTIME-completo.

Complexidade Computacional de $\mathcal{C}\&\mathcal{R}$

Teorema(Goldstein e Reingold, 1995): $\mathcal{C}\&\mathcal{R}$ com as posições iniciais fornecidas é EXPTIME-completo.

Teorema(Fomin, Golovach e Kratochvíl, 2008): $\mathcal{C}\&\mathcal{R}$ é NP-difícil.

Teorema(Fomin, Golovach e Lokshtanov, 2010): $\mathcal{C}\&\mathcal{R}$ é PSPACE-completo **quando cada policial pode se mover um dado número de vezes.**

Teorema(Mamino, 2013): $\mathcal{C}\&\mathcal{R}$ é PSPACE-completo.

Teorema(Kinnersley, 2015): $\mathcal{C}\&\mathcal{R}$ é EXPTIME-completo.

Jogo da Formula Booleana Alternante (ABF)

- ▶ **Entrada:** Formula Lógica CNF Φ sobre as variáveis $X = X_A \cup X_B$ com valores verdades estabelecidos;
- ▶ Em seu turno Alice (Bob) altera o valor de **no máximo** uma variável de $X_A(X_B)$;
- ▶ Alice vence se conseguir tornar Φ verdadeira. Caso contrário, o vencedor é Bob.

Teorema(Stockmeyer e Chandra, 1979): ABF é EXPTIME-completo.

- ▶ Stockmeyer e Chandra classificaram jogos EXPTIME-completos como Jogos Universais.

Jogo da Formula Booleana Alternante (ABF)

- ▶ **Entrada:** Formula Lógica CNF Φ sobre as variáveis $X = X_A \cup X_B$ com valores verdades estabelecidos;
- ▶ Em seu turno Alice (Bob) altera o valor de **no máximo** uma variável de $X_A(X_B)$;
- ▶ Alice vence se conseguir tornar Φ verdadeira. Caso contrário, o vencedor é Bob.

Teorema(Stockmeyer e Chandra, 1979): ABF é EXPTIME-completo.

- ▶ Stockmeyer e Chandra classificaram jogos EXPTIME-completos como Jogos Universais.

Jogo da Formula Booleana Alternante (ABF)

- ▶ **Entrada:** Formula Lógica CNF Φ sobre as variáveis $X = X_A \cup X_B$ com valores verdades estabelecidos;
- ▶ Em seu turno Alice (Bob) altera o valor de **no máximo** uma variável de $X_A(X_B)$;
- ▶ Alice vence se conseguir tornar Φ verdadeira. Caso contrário, o vencedor é Bob.

Teorema(Stockmeyer e Chandra, 1979): ABF é EXPTIME-completo.

- ▶ Stockmeyer e Chandra classificaram jogos EXPTIME-completos como Jogos Universais.

Jogo da Formula Booleana Alternante (ABF)

- ▶ **Entrada:** Formula Lógica CNF Φ sobre as variáveis $X = X_A \cup X_B$ com valores verdades estabelecidos;
- ▶ Em seu turno Alice (Bob) altera o valor de **no máximo** uma variável de $X_A(X_B)$;
- ▶ Alice vence se conseguir tornar Φ verdadeira. Caso contrário, o vencedor é Bob.

Teorema(Stockmeyer e Chandra, 1979): ABF é EXPTIME-completo.

- ▶ Stockmeyer e Chandra classificaram jogos EXPTIME-completos como Jogos Universais.

Jogo da Formula Booleana Alternante (ABF)

- ▶ **Entrada:** Formula Lógica CNF Φ sobre as variáveis $X = X_A \cup X_B$ com valores verdades estabelecidos;
- ▶ Em seu turno Alice (Bob) altera o valor de **no máximo** uma variável de $X_A(X_B)$;
- ▶ Alice vence se conseguir tornar Φ verdadeira. Caso contrário, o vencedor é Bob.

Teorema(Stockmeyer e Chandra, 1979): ABF é EXPTIME-completo.

- ▶ Stockmeyer e Chandra classificaram jogos EXPTIME-completos como Jogos Universais.

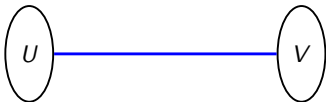
Jogo da Formula Booleana Alternante (ABF)

- ▶ **Entrada:** Formula Lógica CNF Φ sobre as variáveis $X = X_A \cup X_B$ com valores verdades estabelecidos;
- ▶ Em seu turno Alice (Bob) altera o valor de **no máximo** uma variável de $X_A(X_B)$;
- ▶ Alice vence se conseguir tornar Φ verdadeira. Caso contrário, o vencedor é Bob.

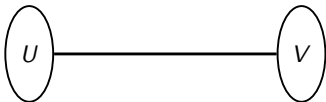
Teorema(Stockmeyer e Chandra, 1979): ABF é EXPTIME-completo.

- ▶ Stockmeyer e Chandra classificaram jogos EXPTIME-completos como Jogos Universais.

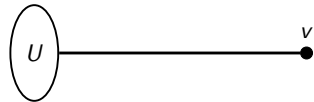
$\mathcal{C} \& \mathcal{R}$ com posições iniciais fornecidas



(a) $u_i v_j$ é aresta se e só se $i = j$.



(c) $u_i v_j$ é aresta para todo i, j .



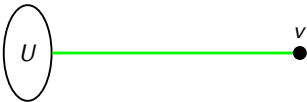
(e) $u_i v$ é aresta para todo i .



(b) $u_i v_j$ é aresta se e só se $i \neq j$.

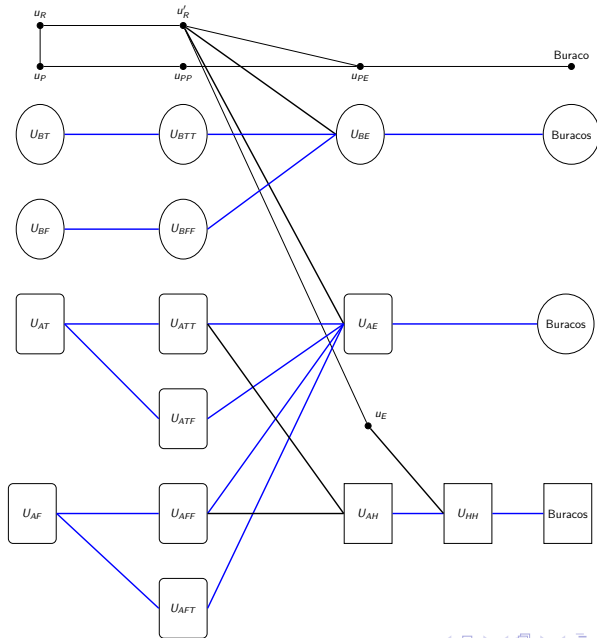


(d) $u_i v_j$ é aresta se e só se $i = j = 1$.

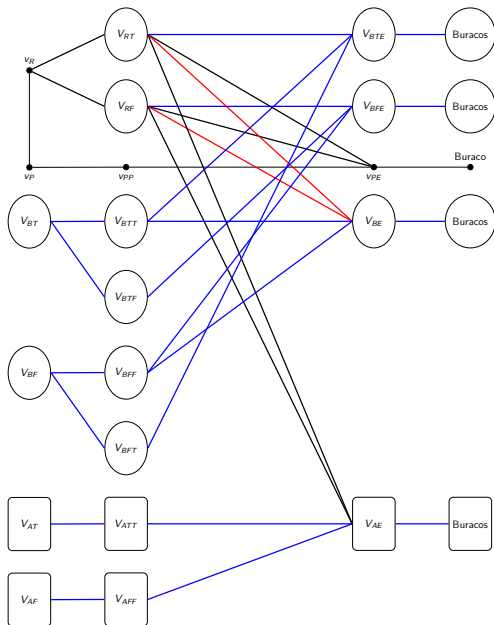


(f) $u_i v$ é aresta se e só se $i = 1$.

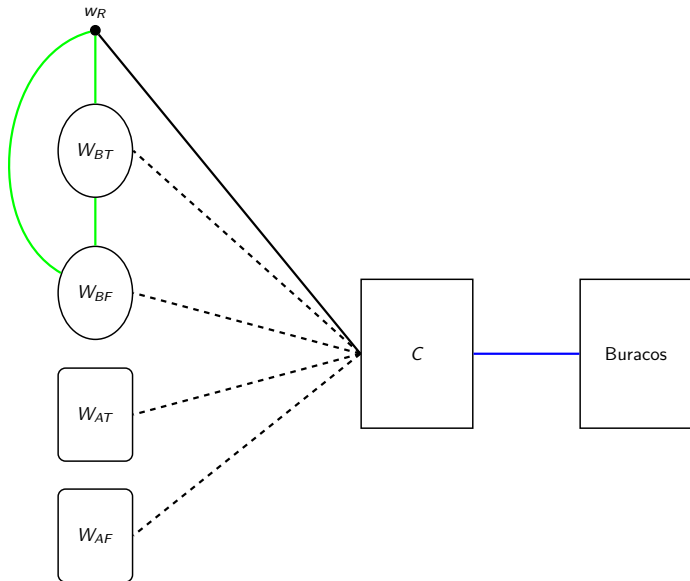
$\mathcal{C}\&\mathcal{R}$ com posições iniciais fornecidas (Turno da Alice)



$\mathcal{C} \& \mathcal{R}$ com posições iniciais fornecidas (Turno do Bob)



$\mathcal{C} \& \mathcal{R}$ com posições iniciais fornecidas (Verificando a Fórmula)



Complexidade Computacional de $\mathcal{C}\&\mathcal{R}$

Teorema(Goldstein e Reingold, 1995): $\mathcal{C}\&\mathcal{R}$ com as posições iniciais fornecidas é EXPTIME-completo.

Teorema(Fomin, Golovach e Kratochvíl, 2008): $\mathcal{C}\&\mathcal{R}$ é NP-difícil.

Teorema(Fomin, Golovach e Lokshtanov, 2010): $\mathcal{C}\&\mathcal{R}$ é PSPACE-completo quando cada policial pode se mover um dado número de vezes.

Teorema(Mamino, 2013): $\mathcal{C}\&\mathcal{R}$ é PSPACE-completo.

Teorema(Kinnersley, 2015): $\mathcal{C}\&\mathcal{R}$ é EXPTIME-completo.

FIM DA AULA 05

FIM DA AULA 05

Perguntas (se houver tempo) ??