

O Jogo da Cobertura de Vértices em Grafos

J. Bastos¹, F. Benevides¹, T. Marcilon², N. Martins³, N. Nisse⁴, R. Sampaio¹

¹ Universidade Federal do Ceará (UFC), Brazil

Universidade Federal do Cariri (UFCA), Brazil

Inst. de Eng. e Desenvolvimento Sustentável (UNILAB), Brazil

Université Côte d’Azur, Inria, CNRS, I3S, France

{josefran,fabricio,rudini}@ufc.br, thiago.marcilon@ufca.edu.br, nicolas.nisse@inria.fr

Abstract. *In this paper, we introduce the Vertex Cover Game in graphs. Regarding the main (optimization) variant, we determine the winner in star forests and complete multipartite graphs, as well as obtaining a 2-approximation polynomial algorithm and an FPT algorithm parameterized by the number of selected vertices. Regarding the normal variant, we determine the winner for any complete multipartite graph.*

Resumo. *Nesse artigo, nós introduzimos o Jogo da Cobertura de Vértices em grafos. Com relação à variante principal (de otimização), nós determinamos o vencedor em florestas de estrelas e grafos multipartido completos, além de obter um algoritmo polinomial 2-aproximativo e um algoritmo FPT parametrizado pelo número de vértices selecionados. Com relação à variante normal, determinamos o vencedor de qualquer grafo multipartido completo.*

1. Introdução

Dado um grafo G , dizemos que um vértice $v \in V(G)$ cobre uma aresta $e \in E(G)$ se v é extremidade de e . Dizemos que $C \subseteq V(G)$ é uma *cobertura de vértices* se toda aresta tem uma extremidade em C . Nesse artigo, nós introduzimos o Jogo da Cobertura de Vértices em grafos. Nesse jogo sobre um grafo G , Alice e Bob se alternam escolhendo um vértice jogável, onde um vértice é *jogável* se cobre pelo menos uma aresta não coberta pelos vértices escolhidos anteriormente no jogo. O jogo termina quando o conjunto de vértices escolhidos é uma cobertura de vértices de G . Na variante normal, vence quem jogar por último. Na variante *misère*, perde quem jogar por último. Na variante de otimização, é dado um inteiro k e Alice vence se forem selecionados no máximo k vértices; senão, Bob vence. O menor inteiro k tal que Alice vence essa variante é o *número jogo de cobertura* $vc_g(G)$. Note que $vc_g(G) \geq vc(G)$, onde o *número de cobertura* $vc(G)$ é o tamanho da menor cobertura de vértices de G . O Jogo de Cobertura de Vértices se assemelha ao Jogo de Dominação, introduzido por [Brešar et al. 2010] e provado PSPACE-completo na variante de otimização por [Brešar et al. 2016] e na variante normal por [Brito et al. 2026]. Neste artigo, determinamos o vencedor, tanto da variante de otimização como da variante normal, em algumas classes de grafos. Omitimos as demonstrações por falta de espaço.

2. Variante de Otimização em classes de grafos

Seja P_n o caminho com n vertices. Note que $vc_g(P_0) = vc_g(P_1) = 0$, já que não há arestas a cobrir. Em florestas de caminhos, provamos que tanto Alice como Bob preferem jogar em caminhos com tamanho múltiplo de 3: Bob prefere jogar em extremidades e Alice prefere jogar em vizinhos de extremidades.

Teorema 1. Para qualquer inteiro $k \geq 1$, temos que $vc_g(P_{3k-1}) = vc_g(P_{3k}) = 2k - 1$ e $vc_g(P_{3k+1}) = 2k$. Além disso, dada uma floresta de caminhos G , é possível calcular o Número Jogo de Cobertura $vc_g(G)$ e determinar o vencedor da variante de otimização do Jogo da Cobertura de Vértices em tempo linear.

Teorema 2. Dada uma floresta de estrelas G , é possível calcular o Número Jogo de Cobertura $vc_g(G)$ e determinar o vencedor da variante de otimização do Jogo da Cobertura de Vértices em tempo polinomial.

O teorema abaixo lida com grafos multipartido completos. A intuição do jogo em um grafo multipartido completo é que Bob tenta alcançar uma configuração em que pelo menos duas partes tem o mesmo tamanho (sempre reduzindo a maior parte se $n_k > n_{k-1}$), enquanto Alice tenta evitar isso sempre reduzindo a segunda maior parte se $n_k > n_{k-1}$. Se em algum momento as duas maiores partes tem o mesmo tamanho, Bob sempre consegue manter as duas maiores partes de mesmo tamanho (se n é par) ou com 1 ou 2 vértices de diferença (se n é ímpar).

Teorema 3. Seja G um grafo multipartido completo com $k \geq 2$ partes de tamanhos $1 \leq n_1 \leq \dots \leq n_k$ e seja $n = \sum_{i=1}^k n_i$. Dado $x \in \mathbb{N}$, seja j_x o menor inteiro j tal que $n_j \geq x$. Se existe $x \in \mathbb{N}$ tal que $\sum_{i=j_x}^{k-1} (n_i - x) \geq n_k - x$, então $vc_g(G) = n - 1$ se n é par, e $vc_g(G) = n - 2$ se n é ímpar. Senão, $vc_g(G) = 2 \cdot \sum_{i=1}^{k-1} n_i - 1$.

3. Variante de Otimização: Algoritmo Aproximativo

O algoritmo guloso para Alice descrito na prova do teorema abaixo é um algoritmo aproximativo. A ideia é que os vértices selecionados induzem um emparelhamento maximal e portanto de tamanho no máximo 2 vezes o menor tamanho $vc(G)$ de uma cobertura de vértices de G .

Teorema 4. Existe um algoritmo guloso que garante a vitória de Alice em no máximo $2vc_g(G) + 1$ jogadas em qualquer grafo G .

Demonstração. Alice seleciona uma vértice qualquer não isolado v . Seja $G_0 = G - v$. Mostramos que existe um emparelhamento M de G_0 tal que o jogo com Bob começando em G_0 terminará após no máximo $2 \cdot |M|$ vértices selecionados. O resultado segue daí, pois $|F| \leq vc(G_0) \leq vc_g(G_0) \leq vc_g(G)$ para qualquer emparelhamento F de G_0 .

Para $i \geq 0$, seja S_i o conjunto de vértices selecionados após Alice e Bob terem selecionados i vértices cada. $S_0 = \emptyset$. Seja G_i o grafo obtido de $G - S_i$ após a remoção de vértices isolados. Seja $M_0 = \emptyset$. Seja $i \geq 1$ e suponha que o jogo ainda não terminou, ou seja, $G_{i-1} \neq \emptyset$. Seja $b_i \in V(G_{i-1})$ o i -ésimo vértice selecionado por Bob (note que b_i deve ter algum vizinho em G_{i-1}). Se algum vizinho de b_i em G_{i-1} tem grau 1 em G_{i-1} , seja w_i tal vértice em $N_{G_{i-1}}(b_i)$. Seja $M_i = M_{i-1} \cup \{\{b_i, w_i\}\}$. Note que, como por indução M_{i-1} é um emparelhamento de tamanho $i - 1$ de G_0 , então M_i é um emparelhamento de tamanho i de G_0 . Ou o jogo termina (se $G_{i-1} - b_i$ não tiver arestas), ou tome $a_i \in V(G_{i-1}) \setminus \{b_i\}$ tal que a_i tenha pelo menos um vizinho em $G_{i-1} - b_i$. Seja Alice quem seleciona a_i , isto é, $S_i = S_{i-1} \cup \{a_i, b_i\}$. Caso contrário, seja $a_i \in N_{G_{i-1}}(b_i)$

(portanto, a_i tem em G_{i-1} algum vizinho distinto de b_i). Seja Alice quem seleciona a_i (isto é, $S_i = S_{i-1} \cup \{a_i, b_i\}$) e seja $M_i = M_{i-1} \cup \{\{b_i, a_i\}\}$. Observe que, como por indução M_{i-1} é um emparelhamento de tamanho $i - 1$ de G_0 , então M_i é um emparelhamento de tamanho i de G_0 . Seja ℓ a última jogada do jogo (isto é, Bob jogou ℓ vezes em G_0 e Alice jogou ℓ ou $\ell - 1$ vezes). Então, M_ℓ é um emparelhamento de tamanho ℓ de G_0 e o resultado segue. $\square$

4. Variante de Otimização: Algoritmo FPT

Seja (G, k) uma instância do Jogo da Cobertura de Vértices na variante de otimização. Seja $H = \{v \in V \mid d(v) > k\}$, $A = \{v \in V \setminus H \mid N(v) \subseteq H\}$, e $B = V \setminus (H \cup A)$.

Redução 1. Se $|H| > k$ ou $|B| > 2k^2$, então (G, k) é uma instância NÃO.

Esboço. Para todo $v \in H$, se v não pertence a uma cobertura de vértices S de G , todos os seus $d(v) \geq k + 1$ vizinhos devem pertencer a S . Portanto, o conjunto H é subconjunto de toda cobertura de vértices de G de tamanho no máximo k .

Além disso, $\delta(G[B]) \geq 1$ e $\Delta(G[B]) \leq k$. Logo, como todo vértice de B cobre no máximo k arestas, se existem mais de k^2 arestas em $G[B]$, então $G[B]$ não pode ter uma cobertura de tamanho no máximo k . Portanto, como $\delta(G[B]) \geq 1$, se $|B| > 2k^2$, $G[B]$ não tem uma cobertura de tamanho no máximo k . $\square$

Dado $D \subseteq H$, seja $A_D = \{v \in A \mid N(v) = D\}$.

Redução 2. Para todo $D \subseteq H$, se $|A_D| > k + 1$, remova um vértice arbitrário $v \in A_D$ de G . A instância resultante é $(G - v, k)$.

Esboço. Como o jogo dura no máximo $k + 1$ turnos, todo vértice de H sempre será uma opção de jogada para ambos os jogadores até que alguém jogue em H . Além disso, quaisquer dois vértices v e w em algum A_D tem a mesma vizinhança durante todo o jogo. Portanto, se algum vértice de algum A_D se torna isolado em algum momento do jogo, todo vértice de A_D se torna isolado.

Assim, quando um jogador deseja escolher um vértice de A_D , ele pode escolher qualquer vértice de A_D atingindo o mesmo objetivo. $\square$

Teorema 5. Dado um grafo G e um inteiro k , existe um algoritmo que determina se $vc_g(G) \leq k$ em tempo $O(|V(G)| + O(2^{\Theta(k^2)}))$.

Demonstração. Após aplicar exaustivamente as Reduções 1 e 2, se (G, k) não for previamente conhecido como uma instância NÃO, temos que $|V(G)| = |B| + |H| + |A| \leq 2k^2 + k + (k + 1) \cdot 2^k$. Observe que qualquer vértice em H antes da aplicação da Redução 2 ainda estará em H depois.

Ambas as reduções podem ser trivialmente aplicadas em tempo linear. Além disso, a aplicação do algoritmo Minimax para determinar se Alice possui a estratégia vencedora leva tempo $O(|V(G)|^{k+1})$. $\square$

5. Variante Normal do Jogo da Cobertura de Vértices

A Teoria de Sprague-Grundy [Sprague 1936, Grundy 1939] dos anos 1930 mostra como associar um número natural, chamado *nimber*, a cada posição de um jogo imparcial na variante normal de modo que o primeiro jogador sempre perde se e só se o nimber é 0. Prova-se que o nimber de um jogo J (normal imparcial) é exatamente o mínimo valor excluído $\text{mex}(N) = \min\{k \in \mathbb{N} \mid k \notin N\}$, onde N é o conjunto de nimbers de todas as posições que podem ser obtidas após uma jogada em J . Além disso, define-se a união de dois jogos J_1 e J_2 como o jogo $J_1 + J_2$ em que o jogador atual sempre deve escolher um dos jogos e fazer uma jogada nele. Prova-se também que o nimber $\text{nim}(J_1 + J_2)$ é $\text{nim}(J_1) \oplus \text{nim}(J_2)$, onde $\oplus$ é a operação de ou-exclusivo bit-a-bit.

Teorema 6. *O nimber do Jogo da Cobertura de Vértices na variante normal em caminhos pode ser obtido em tempo polinomial pela recorrência: $\text{nim}(P_0) = \text{nim}(P_1) = 0$ e*

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

Teorema 7. *O nimber de uma floresta de estrelas pode ser computado em tempo polinomial.*

Demonstração. Seja S_p a estrela com $p + 1$ vértices. Logo $\text{nim}(S_0) = 0$, $\text{nim}(S_1) = 1$, e $\text{nim}(S_p) = 2$ se $p \geq 2$ é par e $\text{nim}(S_p) = 1$ caso contrário. O nimber de uma floresta de estrelas é a operação $\oplus$ aplicado ao nimber de cada estrela. $\square$

Teorema 8. *Seja G um grafo multipartido completo com $k \geq 2$ partes de tamanhos $1 \leq n_1 \leq \dots \leq n_k$. Seja $n = \sum_{i=1}^k n_i$ e $n' = n - n_k$. Logo, sobre o nimber do Jogo da Cobertura de Vértices na variante normal sobre G :*

- (a) *se $n' \leq 2k - 4$, então $\text{nim}(G)$ é 1 ou 0 dependendo se n é par ou ímpar;*
- (b) *se $n' = 2k - 3$, então $\text{nim}(G)$ é 1 ou 2 dependendo se n é par ou ímpar;*
- (c) *se $n' = 2k - 2$, então $\text{nim}(G)$ é 0 ou 2 dependendo se n é par ou ímpar;*
- (d) *se $n' \geq 2k - 1$, então $\text{nim}(G)$ é 0 ou 1 dependendo se n é par ou ímpar.*

Esboço da Prova. Seja G um grafo multipartido completo com $n \geq 2$ vértices e $k \geq 2$ partes. Prova por indução em $n + k$. O caso base da indução é o caso das estrelas, visto no Teorema 7, em que $k = 2$ e $n' = 1$, sendo sempre caso (b): nimber igual a 1 ou 2 dependendo se n é par ou ímpar.

Por indução, assumo valer para grafos multipartido completos em que o número de vértices mais o número de partes é menor que $n + k$. Toda jogada para um número n par (resp. ímpar) de vértices leva a uma posição com número ímpar (resp. par) de vértices. Uma jogada no caso (a) pode levar ao caso (a) ou possivelmente ao caso (b) e, de fato, $1 = \text{mex}\{0, 2\}$ e $0 = \text{mex}\{1, 1\}$. Uma jogada no caso (b) pode levar aos casos (a), (b) e (c) e, de fato, $1 = \text{mex}\{0, 2, 2\}$ e $2 = \text{mex}\{1, 1, 0\}$. Uma jogada no caso (c) pode levar aos casos (b), (c) e (d) e, de fato, $0 = \text{mex}\{2, 2, 1\}$ e $2 = \text{mex}\{1, 0, 0\}$. Uma jogada no caso (d) pode levar ao caso (c) ou possivelmente ao caso (d) e, de fato, $0 = \text{mex}\{2, 1\}$ e $1 = \text{mex}\{0, 0\}$. $\square$

Referências

- Brešar, B., Klavžar, S., and Rall, D. F. (2010). Domination game and an imagination strategy. *SIAM J. Discrete Math.*, 24(3):979–991.
- Brešar, B., Dorbec, P., Klavžar, S., Košmrlj, G., and Renault, G. (2016). Complexity of the game domination problem. *Theoretical Computer Science*, 648:1–7.
- Brito, J. M., Marcilon, T., Martins, N. A., and Sampaio, R. M. (2026). The normal domination game in graphs. *J. Comput. Syst. Sci.*, 157:103751.
- Grundy, P. M. (1939). Mathematics and games. *Eureka (The Archimedeans' Journal)*, 2:6–8.
- Sprague, R. (1936). Über mathematische Kampfspiele. *Tôhoku Math J*, 41:438–444.