

# Contando Grafos de Forma Algébrica

Emanuel Juliano<sup>1</sup>  
Orientador: Gabriel Coutinho<sup>1</sup>

<sup>1</sup>Universidade Federal de Minas Gerais (UFMG)

{emanuelsilva, gabriel}@dcc.ufmg.br

**Resumo.** *Nesse projeto apresentamos um compilado de resultados da teoria algébrica de grafos, percorrendo diversas áreas da álgebra e utilizando de diferentes técnicas, demonstramos resultados envolvendo contagens de grafos.*

## 1. Introdução

A álgebra é de fundamental importância para a computação nos dias de hoje, seja devido ao cálculo matricial utilizado em algoritmos de inteligência artificial, seja devido à sua aplicação em algoritmos de criptografia como RSA. Porém em todos esses exemplos, a álgebra é utilizada para resolver problemas algébricos. O objetivo desse POC II é demonstrar exemplos de como a álgebra pode ser utilizada para solucionar problemas de contagem de origem combinatória.

No POC I, estudamos se é possível realizar transferência quântica de estado entre dois vértices de um grafo conectados por um caminho simples de comprimento arbitrário. Durante o projeto demonstramos um limite inferior para o comprimento do caminho, comprovando que caso os dois vértices estejam a uma distância maior ou igual a 8, então o fenômeno quântico não ocorre entre os dois. Por fim, ao descobrir limite inferiores, demonstramos que vértices conectados por um caminho simples não admitem transferência de estado entre eles, generalizando assim, os resultados do artigo *Quantum Walks do not Like Bridges*.

Como o problema do POC I foi resolvido, neste projeto abordaremos outros problemas que também estão relacionados à combinatória algébrica. Todos os problemas abordados já foram previamente resolvidos, porém, buscaremos, sempre que possível, apresentar questões em aberto que estão relacionadas, assim como tentativas para solucionar essas questões. Os principais problemas, assim como as técnicas utilizadas para resolvê-los são apresentados a seguir, cada um deles utiliza de uma técnica algébrica diferente em sua solução, portanto, cada problema corresponderá a uma sessão do projeto:

1. Contar o número de grafos distintos com  $n$  vértices. Nesta sessão abordaremos a teoria dos grupos, demonstrando o teorema de Burnside. Utilizaremos como referência o capítulo 3 do livro [Biggs 1993], assim como o livro *The Theory of Finite Groups* [Kurzweil and Stellmacher 2003].
2. Contar o número de subgrafos eulerianos de um grafo. Nesta sessão, utilizaremos de álgebra linear para explorar como o posto de matrizes nos revela propriedades combinatórias do grafo. Utilizaremos como referência o livro *Algebraic Graph Theory* [Biggs 1993] assim como o comentário [Ilfen 2017].
3. Contar o número de árvores geradoras de um grafo. Nesta sessão utilizaremos exploraremos como computar o determinante de uma matriz, nos aproveitando de

estruturas do grafo. Utilizaremos como referência as notas de aula do curso *Algebraic Combinatorics* [Coutinho 2020], assim como o artigo *Matrix Tree Theorems* [Chaiken and Kleitman 1978].

4. Contar o número de árvores binárias com  $n$  vértices. Nesta sessão apresentaremos funções geradoras e suas diversas aplicações e como computar seus coeficientes de modo eficiente. Utilizaremos mais uma vez as notas de aula [Coutinho 2020] e o blog que retrata o uso de funções geradoras *Generating Functions in Competitive Programming*. [(https://codeforces.com/profile/zscoder) 2020].
5. Contar o número de homomorfismos de um grafo para o outro. Nessa sessão estudaremos contagens de homomorfismos, buscando demonstrar a equivalência entre essas contagens e isomorfismo e cospectralidade. Utilizaremos como referência a tese de phd [Roberson 2013] e o artigo [Dell et al. 2018].

## 2. Contagem de grafos

Um grafo simples  $G$  é definido pelo par  $G = (V, E)$ , correspondendo a um conjunto de vértices e de arestas. Nosso objetivo nessa sessão é definir limitantes para o número de grafos com  $n$  vértices, para isso, alguns conceitos precisam ser definidos.

Um **isomorfismo** entre dois grafos  $G, H$  é uma função  $f : V(G) \rightarrow V(H)$  tal que  $uv \in E(G) \iff f(u)f(v) \in E(H)$ , isto é, uma função bijetiva e que preserva adjacência. Se existe um isomorfismo entre dois grafos  $G$  e  $H$  dizemos que eles são *isomorfos*, e denotamos por

$$G \cong H.$$

Nosso objetivo nessa sessão é contar quantos grafos não isomórficos existem com  $n$  vértices, dessa forma estaremos efetivamente contando grafos distintos.

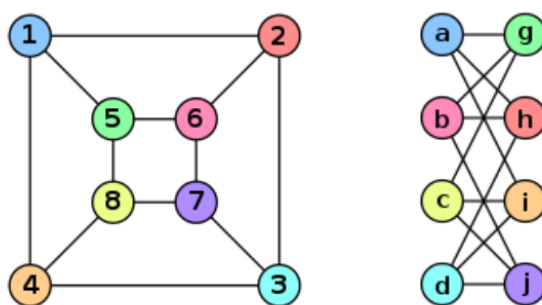


Figura 1. Exemplo de isomorfismo entre grafos.

Um **automorfismo** de um grafo  $G$  é um isomorfismo  $f : G \rightarrow G$ .  $\text{Aut}(G)$  é o conjunto de todos os automorfismos de  $G$ . O conjunto de automorfismos indica as simetrias presentes em um grafo, algumas características interessantes de serem notadas é que todo automorfismo  $f \in \text{Aut}(G)$  permuta sempre vértices de mesmo grau para vértices de mesmo grau:  $\deg_G(v) = \deg_G(f(v))$ , além disso, o automorfismo também sempre preserva distâncias:  $d_G(v, u) = d_G(f(v), f(u))$ .

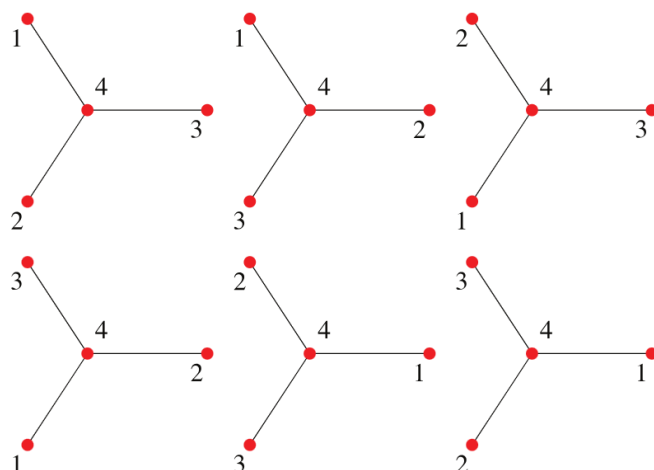


Figura 2. Exemplo de automorfismo do  $S_4$ . Note que o vértice 4 é ponto fixo.

Um grafo  $G$  é dito **assimétrico** se seu grupo de automorfismo  $\text{Aut}(G) = \{\epsilon\}$  é trivial. Um dos resultados mais interessantes relacionando teoria dos grupos e grafos é o teorema 2.1, que diz que “quase” todos os grafos são assimétricos, um dos raros teoremas da teoria algébrica de grafos que diz respeito a estrutura de “quase” todos os grafos.

## 2.1. Teoria dos Grupos

Para demonstrar nosso resultado utilizaremos de resultados da teoria dos grupos. Um **grupo** é definido por um conjunto  $\mathcal{G}$  e uma operação binária  $\cdot$ , tal que:

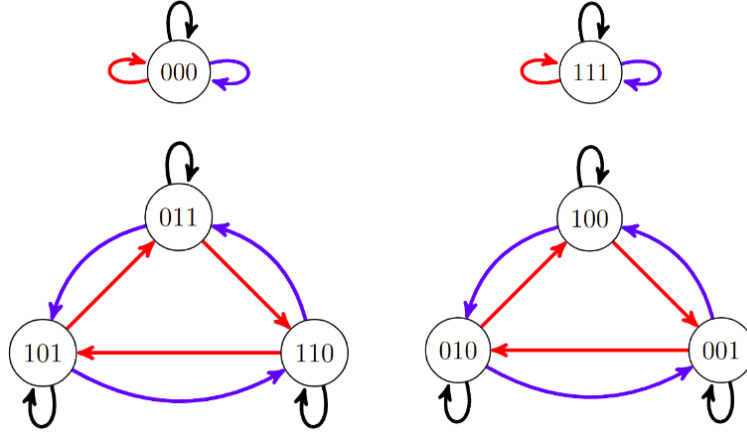
- $\forall a, b, c \in \mathcal{G}, (a \cdot b) \cdot c = a \cdot (b \cdot c)$ , satisfaz associatividade.
- $\epsilon \cdot a = a$ , possui o elemento neutro.
- $\forall a \exists b : a \cdot b = \epsilon$ , todo elemento possui inverso.

O exemplo mais simples de um grupo finito é o grupo de inteiros módulo  $n$ , normalmente denotado por  $(\mathbb{Z}/n\mathbb{Z}, +)$ , o conjunto de elementos  $\mathcal{G} = \{0, 1, \dots, n-1\}$  e a operação binária é a soma módulo  $n$ . Em nosso contexto, o grupo mais importante será o grupo de permutações  $\text{Sym}(n)$ , também conhecido como grupo simétrico, o conjunto de elementos  $\mathcal{G}$  são todas as permutações de tamanho  $n$  e a operação binária é justamente a composição de permutações.

**Lema 2.1.** *Seja  $G$  um grafo com  $n$  vértices,  $\text{Aut}(G)$  é um grupo e seus elementos são um subconjunto de  $\text{Sym}(n)$ .*

Ao utilizarmos grupos teremos um conjunto de objetos  $X$  que estamos trabalhando e um grupo  $\mathcal{G}$  que podemos utilizar para transformar um objeto em outro. Formalmente, temos uma função  $\circ : \mathcal{G} \times X \rightarrow X$  tal que  $x^g$  é o objeto que temos após aplicar a operação  $g$  em  $x$ , chamamos isso de *ação do grupo*. Em nosso caso, se estivermos tratando do grupo de permutações agindo sobre um grafo, a ação é justamente permutar os vértices.

Uma boa forma de pensar a ação de um grupo  $\mathcal{G}$  em um conjunto  $V$  é considerando um grafo em que os vértices são os elementos de  $V$  e as arestas entre vértices  $v$  e  $u$  indicam que um elemento  $g$  mapeia  $v$  para  $u$ . Note que esse grafo é formado por um conjunto de componentes conexas e cada componente possui aresta entre todos seus elementos.



**Figura 3.**  $V = \{0, 1\}^3$ ,  $\mathcal{G} = \{\epsilon, \pi_{312}, \pi_{231}\}$ .

Assuma que  $\mathcal{G}$  age em  $V$ , a **órbita**  $x \in V$  é

$$x^{\mathcal{G}} := \{x^g : g \in \mathcal{G}\},$$

isto é, o conjunto de elementos que se obtém após aplicar qualquer transformação em  $x$ , em nossa representação por grafos, órbitas são as componentes conexas. Nosso objetivo será justamente contar o número de órbitas no conjunto dos grafos com  $n$  vértices.

Seja  $\mathcal{G}$  um grupo agindo em  $V$ . Fixe  $x \in V$ . O **estabilizador** de  $x$  é

$$\mathcal{G}_x := \{g \in \mathcal{G} : x^g = x\},$$

isto é, os elementos de  $g$  que fixam  $x$ , em nossa representação por grafos, o estabilizador de um vértice é o conjunto de self-loops.

**Lema 2.2** (Órbita-Estabilizador). *Seja  $\mathcal{G}$  um grupo de permutação agindo em  $V$  e  $x \in V$ . Então,*

$$|\mathcal{G}_x| \cdot |x^{\mathcal{G}}| = |\mathcal{G}|$$

Não apresentaremos a demonstração do lema acima, porém, utilizaremos com frequência o lema para demonstrar nosso resultado.

## 2.2. Grafos Assimétricos e Lema de Burnside

Denotamos por  $\text{fix}(g) := \{v \in V : v^g = v\}$  o conjunto dos pontos de  $V$  fixos por  $g$ .

**Lema 2.3** (Burnside). *Seja  $\mathcal{G}$  um grupo agindo em  $V$ , então o número de órbitas de  $\mathcal{G}$  é dado por*

$$\frac{1}{|\mathcal{G}|} \sum_{g \in \mathcal{G}} |\text{fix}(g)|.$$

*Demonstração.* Seja  $\Lambda := \{(g, x) : g \in \mathcal{G}, x \in \text{fix}(g)\}$  o par composto por elementos de  $\mathcal{G}$  e seus pontos fixos. Aplicamos um argumento de contagem dupla.

$$\begin{aligned}
\sum_{g \in \mathcal{G}} |\text{fix}(g)| &= |\Lambda| \\
&= \sum_{x \in V} |\mathcal{G}_x| \\
&= \sum_{x \in V} \frac{|\mathcal{G}|}{|x^{\mathcal{G}}|} && \text{Lema Órbita-Estabilizador} \\
&= |\mathcal{G}| \cdot (\text{número de órbitas}).
\end{aligned}$$

□

O lema de Burnside possui diversas aplicações que serão apresentadas nessa sessão, é uma ferramenta poderosa da teoria dos grupos e que é utilizado para contar, desde a quantidade de grafos com  $n$  vértices, até o número de isomerias de uma molécula orgânica e a quantidade de acordes de uma teoria musical [Jin 2018].

Denotamos por  $\Gamma_n$  o conjunto de grafos com  $n$  vértices. Então, escrevemos

$$\text{Iso}(G) := \{H \in \Gamma_n : H \cong G\}$$

para referir à classe de isomorfismos de  $G$ .

**Lema 2.4.** Para todo  $G \in \Gamma_n$ ,

$$|\text{Iso}(G)| = \frac{n!}{|\text{Aut}(G)|}.$$

*Demonstração.* Seja  $\mathcal{G} = \text{Sym}(n)$ . Para  $g \in \mathcal{G}$ , seja  $\tau_g : \Gamma_n \rightarrow \Gamma_n$  o isomorfismo que manda

$$G \rightarrow G^g.$$

O grupo  $\mathcal{H} : \{\tau_g : g \in \mathcal{G}\}$  age em  $\Gamma_n$  e é isomorfo a  $\mathcal{G}$ . Pelo lema Órbita-Estabilizador:

$$\begin{aligned}
n! &= |\mathcal{G}| \\
&= |\mathcal{H}| \\
&= |G^{\mathcal{H}}| \cdot |\mathcal{H}_G| \\
&= |\text{Iso}(G)| \cdot |\text{Aut}(G)|.
\end{aligned}$$

□

O lema a seguir faz uso do lema de Burnside para atingir limitantes no número de classes de isomorfismos e é justamente nele que aparecem limites assintóticos, sua demonstração não será apresentada nesse projeto, porém, no capítulo 2 de [Biggs 1993], no qual essa sessão é baseada, está a demonstração completa.

**Lema 2.5.** O conjunto  $\mathcal{H}$  de todas as classes de isomorfismo de  $\Gamma_n$  satisfaz

$$|\mathcal{H}| = (1 + o(1)) \frac{2^{\binom{n}{2}}}{n!}.$$

**Teorema 2.1.** Assintoticamente, quase todos os grafos são assimétricos. Isto é, se escolhermos um grafo com  $n$  vértices de forma aleatória, a probabilidade dele ser assimétrico tende a 1 conforme  $n \rightarrow \infty$

*Demonstração.* Seja  $\mathcal{H}$  o conjunto de classes de isomorfismo em grafos de tamanho  $n$ . Escolha  $\mathcal{C} \in \mathcal{H}$ . Se os grafos em  $\mathcal{C}$  são assimétricos, então  $|\mathcal{C}| = n!$ . Caso contrário,  $|\mathcal{C}| \leq \frac{n!}{2}$ .

Seja  $\rho$  a proporção de todas as classes em  $\mathcal{H}$  tais que  $|\mathcal{C}| = n!$ . Temos que,

$$\begin{aligned} 2^{\binom{n}{2}} &= |\Gamma_n| \\ &= \sum_{\mathcal{C} \in \mathcal{H}} |\mathcal{C}| \\ &\leq \rho |\mathcal{H}| \cdot n! + (1 - \rho) |\mathcal{H}| \cdot \frac{n!}{2}. \end{aligned}$$

Pelo lema anterior,

$$\begin{aligned} 2^{\binom{n}{2}} &\leq (1 + o(1)) \frac{2^{\binom{n}{2}}}{n!} \left( \rho + \frac{1 - \rho}{2} \right) \\ &= (1 + o(1)) 2^{\binom{n}{2}} \frac{1 + \rho}{2} \end{aligned}$$

E segue-se que  $\rho \rightarrow 1$  conforme  $n \rightarrow \infty$  e a proporção de grafos assimétricos é

$$\frac{\text{tamanho da classe de isomorfismos} \cdot \text{número de classes de isomorfismo assimétricas}}{|\Gamma_n|}$$

$$\begin{aligned} &= \frac{n! \cdot \rho |\mathcal{H}|}{2^{\binom{n}{2}}} \\ &= (1 + o(1)) \rho \\ &\rightarrow 1. \end{aligned}$$

□

### 3. Contagem de subgrafos eulerianos

Um **subgrafo gerador**  $H$  de um grafo  $G$  corresponde a um grafo definido no mesmo conjunto de vértices  $V(H) = V(G)$  e em um subconjunto de arestas  $E(H) \in E(G)$ . Um subgrafo gerador é dito **euleriano** se todos os vértices possuem grau par.

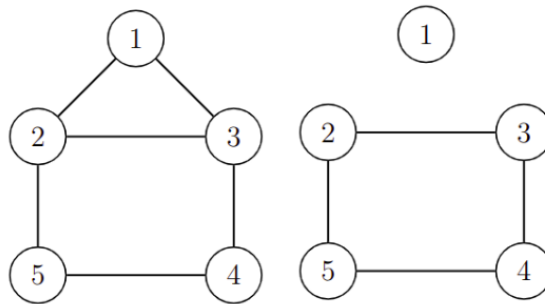
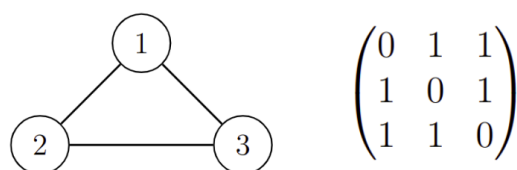


Figura 4. Exemplo de subgrafo euleriano.

A contagem de subgrafos eulerianos de um grafo é o problema motivador desta sessão, porém, também apresentaremos outros resultados com demonstrações semelhantes e que utilizam dos mesmos conceitos algébricos.

Seja  $G$  um grafo não direcionado. Sua **matriz de adjacência**  $A = A(G)$  é uma matriz  $V \times V$  de 0's e 1's tal que

$$A_{u,v} = \begin{cases} 1 & \text{se } u \sim v \\ 0 & \text{c.c.} \end{cases}$$



**Figura 5.**  $K_3$  e  $A(K_3)$

O **polinômio característico** de um grafo é

$$\phi(A, x) := \det(xI - A(G))$$

- Exemplo:  $\phi(A(K_3), x) = x^3 - 3x - 2$

O espectro de um grafo  $G$  é a lista de autovalores de  $A(G)$  junto com suas multiplicidades.

O primeiro resultado que pode ser apresentado no contexto de contagens de grafos e que utiliza de sua matriz de adjacência é uma forma, não muito eficiente, porém surpreendente, de contar arestas e triângulos no grafo:

**Lema 3.1.** *Seja  $A = A(G)$ . Então  $(A^r)_{u,v}$  é o número de passeios fechados de  $u$  para  $v$  em  $G$  de tamanho  $r$ .*

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \quad \begin{pmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix} \quad \begin{pmatrix} 2 & 3 & 3 \\ 3 & 2 & 3 \\ 3 & 3 & 2 \end{pmatrix}$$

**Figura 6.**  $A^0, A^1, A^2, A^3$ .

**Corolário 3.1.** *Assuma que um grafo  $G$  possui  $m$  arestas e  $t$  triângulos. Seja  $A = A(G)$ . Então*

$$\begin{aligned} \text{tr } A &= 0 \\ \text{tr } A^2 &= 2m \\ \text{tr } A^3 &= 6t \end{aligned}$$

---

A **matriz de incidência de um grafo**  $B = B(G)$  é uma matriz de vértices por arestas  $V \times E$  de 0's e 1's tal que

$$B_{u,e} = \begin{cases} 1 & \text{se } u \in e \\ 0 & \text{c.c.} \end{cases}$$

Note que essa matriz possui exatamente dois valores iguais a 1 para cada coluna (as duas pontas da aresta).

Definimos como  $\text{rk}_2(M)$  o rank da matriz  $M$  no espaço do  $\mathbb{Z}_2$ , isto é, o tamanho do espaço gerado pelos vetores  $v$  tais que  $Mv \neq 0$ , lembre-se que no espaço  $\mathbb{Z}_2$ ,  $0 + 0 = 0$ ,  $0 + 1 = 1$  e  $1 + 1 = 0$ . O principal teorema que irá nos ajudar a computar o número de subgrafos eulerianos é o que segue.

**Teorema 3.1.** *Seja  $G$  um grafo com  $n$  vértices e  $c$  componentes conexas. Se  $B$  é a matriz de incidência de  $G$ , então  $\text{rk}_2 B = n - c$ .*

*Demonstração.* Precisamos mostrar que o espaço nulo de  $B$  possui dimensão  $c$ , e portanto que  $\text{rk}_2 B = n - c$ . Suponha que  $z$  é um vetor em  $\mathbb{Z}_2^n$  tal que  $z^T B = 0$ . Se  $uv$  uma aresta de  $G$ , então  $z_u + z_v = z_u - z_v = 0 \implies z_u = z_v$ . Portanto, se observarmos  $z$  como uma função em  $V(G)$ , ele precisa ser constante nos dois vértices de uma mesma aresta e portanto constante nas componentes de  $G$ . O espaço de tais vetores é justamente  $c$ .  $\square$

Na demonstração anterior, ao escolhermos  $z$  tal que  $z^T B = 0$  aquilo que estamos fazendo pode ser visto como uma escolha de vértices de  $G$ , tal que cada aresta é incidente a um número par de vértices escolhidos. Esse caso é muito semelhante ao que desejamos demonstrar sobre subgrafos eulerianos: queremos escolher arestas (subgrafo gerador), tal que todo vértice é incidido em um número par de arestas (euleriano).

Definimos por  $\text{eul}(G)$  o número de subgrafos eulerianos de  $G$ .

**Teorema 3.2.** *Seja  $G$  um grafo com  $n$  vértices,  $m$  arestas e  $c$  componentes conexas. Se  $B$  é a matriz de incidência de  $G$ , então  $\text{eul}(G) = 2^{m-n+c}$ .*

*Demonstração.* Seja  $z \in \mathbb{Z}_2^m$  tal que  $Bz = 0$ , esse vetor define unicamente um subgrafo euleriano e portanto queremos contar quantos desses vetores existem, e para isso basta contar a dimensão do espaço gerado por eles, que é o espaço nulo. Pelo teorema 3.1, sabemos que  $\text{rk}_2(B) = n - c$ , logo,  $\text{null}_2(B^T) = m - (n - c)$ .

Como estamos no espaço do  $\mathbb{Z}_2^m$ , um espaço de dimensão  $k$  possui  $2^k$  vetores distintos, portanto, o número de vetores em nosso espaço é:

$$\text{eul}(G) = 2^{m-(n-c)}.$$

$\square$

Outra maneira de demonstrar o teorema acima é utilizando do lema de Burnside 2.3:



*Demonstração utilizando lema de Burnside.* Assuma  $G$  conexo e escolha um vértice  $v$ . Para todo outro vértice  $w$  escolha um caminho  $P(v, w)$  de  $v$  para  $w$ . Para um subgrafo gerador  $H \subseteq G$ , chamamos de “flipar” uma aresta  $e$  a operação de remover a aresta se ela existe e adicionar a aresta se ela não existe em  $H$ . Essa transformação resulta em um novo subgrafo gerador e ao ser realizada ao longo de um caminho, apenas os vértices das pontas dos caminhos que mudam a paridade de seu grau.

Para aplicar o lema de Burnside, primeiro precisamos definir quem será nosso conjunto  $X$  e nosso grupo  $\mathcal{G}$  que age sobre o conjunto. O conjunto  $X$  será o conjunto de todos os subgrafos geradores de  $G$ , o conjunto gerador do grupo será representado pelos caminhos  $P(v, w)$  escolhidos, a ação de um elemento do grupo em  $G$  é justamente flipar as arestas no caminho de  $v$  para  $w$ . Dessa forma, a órbita de um subgrafo  $H^{\mathcal{G}}$  são todos os subgrafos  $H'$  tal que é possível partindo de  $H$ , aplicar algum conjunto de flips ao longo de caminhos  $P(v, w)$  para qualquer  $w$ . É fácil de perceber dessa forma que cada órbita possui  $2^{n-1}$  subgrafos geradores, já que temos  $n - 1$  caminhos  $P(v, w)$  para flipar e esse também é justamente o tamanho do grupo.

Perceba também que em todas as órbitas existe exatamente um único subgrafo  $H$  tal que todos os vértices possuem grau par, dessa forma, nosso objetivo é contar o número de órbitas e para isso, aplicamos Burnside:

$$\begin{aligned} \# \text{ de subgrafos eulerianos} &= \# \text{ de órbitas} \\ &= \frac{1}{|\mathcal{G}|} \sum_{g \in \mathcal{G}} |\text{fix}(g)|. \\ &= \frac{1}{|\mathcal{G}|} \left( \sum_{g \in \mathcal{G}, g \neq \epsilon} |\text{fix}(g)| + \sum_{g = \epsilon} |\text{fix}(g)| \right) \\ &= \frac{1}{|\mathcal{G}|} \left( 0 + \sum_{g = \epsilon} |\text{fix}(g)| \right) \\ &= \frac{2^m}{2^{n-1}} \end{aligned}$$

Temos assim, que para cada componente conexa o número de subgrafos eulerianos é  $2^{m-n+1}$ , como cada componente é independente, se  $G$  possui  $c$  componentes conexas temos  $2^{m-n+c}$  subgrafos geradores eulerianos.  $\square$

#### 4. Contagem de árvores geradoras

Nosso objetivo nessa sessão é contar o número de árvores geradoras de um grafo, para isso, começamos com um resultado mais simples: contar o número de árvores rotuladas de  $n$  vértices.

**Teorema 4.1** (Fórmula de Cayle). *O número de árvores rotuladas com  $n$  vértices é*

$$n^{n-2}.$$

*Demonstração.* Iremos demonstrar esse resultado criando uma bijeção entre árvores rotuladas e strings de tamanho  $n - 2$  com símbolos que vão de 1 até  $n$ .

Para isso, definimos um algoritmo que mapeia uma árvore  $T$  com  $n \geq 2$  vértices e mapeamos para uma string de tamanho  $n - 2$ . Essa string é chamada de **código de Prüfer** da árvore  $T$ , denotamos por  $P(T)$ . Seja  $v$  a folha com menor rótulo em  $T$  e  $p$  o vizinho de  $v$ , a string  $P(T)$  pode ser definida de forma recursiva:

$$P(T) = \begin{cases} \emptyset & \text{se } n = 2 \\ p \circ P(T - v) & \text{c.c.} \end{cases}$$

Em que  $\circ$  corresponde a concatenação de duas strings.

Para observar a bijeção, basta perceber que o algoritmo de Prüfer é uma função inversível, isto é, dado uma string de  $n - 2$  elementos que vão até  $n$ , conseguimos unicamente determinar qual a árvore rotulada associada revertendo o algoritmo, portanto,  $P(T)$  é uma função bijetiva.  $\square$

**Corolário 4.1.** *O número de árvores geradoras de um grafo completo é  $n^{n-2}$ .*

Agora, para contar a quantidade de árvores geradoras de um grafo qualquer utilizaremos de propriedades de álgebra linear.

Seja  $G$  um grafo,  $D(G)$  é uma matriz diagonal cujas entradas correspondem aos graus dos vértices de  $G$ . A matriz laplaciana de  $G$  é definida por:

$$L = L(G) = D(G) - A(G)$$

O primeiro resultado que será apresentado no contexto de matrizes laplacianas é na verdade uma outra demonstração do teorema 4.1, mas também uma forma de computar de maneira eficiente a quantidade de árvores geradoras de um grafo.

Denotamos por  $\tau(G)$  o número de árvores geradoras de  $G$ . Utilizamos a notação  $G \setminus e$  a remoção da aresta  $e$  e  $G/e$  a contração da aresta  $e$  com seus dois vértices incidentes identificados.

**Lema 4.1.** *Para todo grafo  $G$  e aresta  $e$ , temos que*

$$\tau(G) = \tau(G \setminus e) + \tau(G/e)$$

*Demonstração.* O número de árvores geradoras de  $G$  que não utilizam a aresta  $e$  é precisamente  $\tau(G \setminus e)$  e o número de árvores geradoras que utilizam a aresta é igual a  $\tau(G/e)$ .  $\square$

**Teorema 4.2.** *Seja  $G$  um grafo e  $L$  sua matriz laplaciana. Considere  $a \in V(G)$ , e  $L[a]$  a submatriz de  $L$  obtida após deletar a linha e a coluna correspondente a  $a$ . Então*

$$\tau(G) = \det L[a].$$

*Demonstração.* Essa demonstração segue por indução no número de arestas. Seja  $m$  o número de arestas de  $G$ , seja  $e \in E(G)$  com  $e = \{a, b\}$ . Em  $G/e$ , vértices  $a$  e  $b$  são identificados, digamos por  $c$ . Se mostrarmos que

$$\det L(G)[a] = \det L(G \setminus e)[a] + \det L(G/e)[c],$$

então, por indução no lema acima completamos a demonstração. Ao computar o determinante  $\det L(G)[a]$ , realizamos uma expansão de cofatores na linha correspondendo a  $b$ . Note que todos os termos dessa expansão que são obtidos por posições fora da diagonal também aparecem exatamente iguais em  $\det L(G \setminus e)[a]$ . O único problema está em posições da diagonal - ela é uma unidade maior em  $L(G)[a]$  do que em  $L(G \setminus e)[a]$ , já que o vértice possui uma aresta a menos. Porém, note que a submatriz obtida ao excluir a linha e a coluna de  $b$  de  $L(G \setminus e)[a]$  é precisamente  $L(G/e)[c]$ , portanto

$$\det L(G)[a] = \det L(G \setminus e)[a] + \det L(G \setminus e)[a, b] = \det L(G \setminus e)[a] + \det L(G/e)[c]$$

□

E agora, o número de árvores rotuladas com  $n$  vértices pode ser computado a partir do determinante da matriz  $(n-1) \times (n-1)$

$$L(K_n)[a] = \begin{pmatrix} n-1 & -1 & \dots & -1 \\ -1 & n-1 & & \\ \vdots & & \ddots & \\ -1 & & & n-1 \end{pmatrix}.$$

## 5. Contagem de árvores binárias

O objetivo dessa sessão será de tratar resultados envolvendo funções geradoras, uma das ferramentas mais poderosas no contexto de contagem.

Suponha que temos uma sequência  $a_0, a_1, a_2, \dots$ . Nós queremos associar nossa sequência  $a$  a uma série  $A$  que “encoda” os termos de  $a$  em seus coeficientes. Formalmente, para uma sequência de números  $\{a_i\}_{i=0}^{\infty}$ , nós definimos uma **função geradora ordinária (OGF)** de  $a$  como

$$A(x) = \sum_{i=0}^{\infty} a_i x^i.$$

Por exemplo, considerando a sequência de Fibonacci  $f$ , com termos  $0, 1, 1, 2, 3, 5, 8, \dots$ . Então,

$$F(x) = 0 + x + x^2 + 2x^3 + 3x^4 + 5x^5 + 8x^6 \dots$$

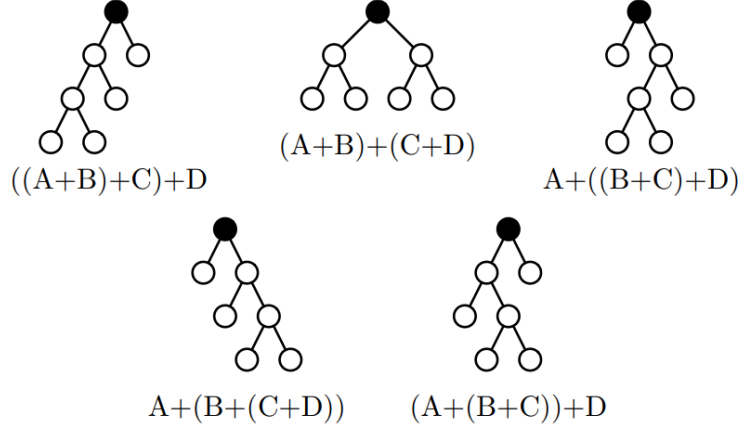
Utilizaremos como notação que para uma série  $A(x) = \sum_{n \geq 0} a_n x^n$ , temos que  $[x^n]A(x) = a_n$ , ou seja, o coeficiente de  $x^n$  em  $A$ .

Um dos exemplos mais simples das propriedades surpreendentes que obtemos ao utilizar funções geradoras é considerar a OGF da sequência  $1, 1, 1, \dots, 1$ . Por definição, temos que  $A(x) = 1 + x + x^2 + x^3 + \dots$ . Porém, isso é uma soma de uma progressão geométrica e portanto  $A(x) = \frac{1}{1-x}$ .

Não iremos nos preocupar com convergência, uma vez que estamos lidando com objetos formais. Nossa única preocupação será descobrir como encontrar os coeficientes da função em tempo finito.

Queremos portanto utilizar de funções geradoras para resolver problemas de contagem. Em nosso caso, queremos contar **quantas árvores binárias enraizadas com  $n$**

**folhas existem.** Perceba também que esse problema é equivalente a contar de quantas formas podemos adicionar parênteses em uma soma de  $n$  elementos.



**Figura 7. Número de árvores binárias. Fonte: [Coutinho 2020]**

Seja  $C_{n-1}$  a quantidade de tais estruturas esse valor é normalmente chamado de **número de catalão**. A figura acima mostra que  $C_3 = 5$ . Podemos escrever a recorrência

$$C_n = C_0C_{n-1} + C_1C_{n-2} + \dots + C_{n-1}C_0 = \sum_{j=0}^{n-1} C_jC_{n-j-1}$$

Assim, conseguimos escrever uma OGF para  $C_n$ . Para isso, vamos escrever  $C(x) - 1$  :

$$C(x) - 1 = \sum_{n=0}^{\infty} C_{n+1}x^{n+1} = \sum_{n=0}^{\infty} \sum_{i=0}^n C_iC_{n-i}x^{n+1} = x \sum_{n=0}^{\infty} \sum_{i=0}^n C_ix^iC_{n-i}x^{n-i}$$

Nós afirmamos que o lado direito da expressão é igual a  $x C(x)^2$ . Considere a expansão de  $C(x)^2$ .

$$C(x)^2 = (C_0x^0 + C_1x^1 + C_2x^2 + \dots)(C_0x^0 + C_1x^1 + C_2x^2 + \dots) = \sum_{n=0}^{\infty} \sum_{i=0}^n C_iC_{n-i}x^n$$

Portanto, temos que  $C(x) - 1 = x C(x)^2$ , utilizando a fórmula de Bhaskara e escolhendo o sinal que garante que  $C_0 = 1$ , concluímos que

$$C(x) = \frac{1 - \sqrt{1 - 4x}}{2x}$$

Porém, ainda temos um problema, como computar os coeficientes dessa função? Para isso, precisamos nos livrar da raiz, assim, iremos utilizar o **teorema binomial generalizado**.

Seja  $r \in \mathbb{C}$  e  $n$  um inteiro não negativo. Então,

$$\binom{r}{n} = \frac{r(r-1)\dots(r-(n-1))}{n!}$$

Essa generalização resulta nos mesmos valores quando  $r \geq n$  e ambos são reais, porém, não estamos mais exigindo que esses valores sejam positivos.

**Lema 5.1.** *Seja  $r$  um número real e  $n$  um inteiro não negativo, então*

$$(1+x)^r = \sum_{n \geq 0} \binom{r}{n} x^n.$$

**Teorema 5.1.** *O número de árvores binárias enraizadas com  $n+1$  folhas é*

$$C_n = \frac{1}{n+1} \binom{2n}{n} x^n$$

*Demonstração.* Sabemos que  $C(x) = \frac{1-\sqrt{1-4x}}{2x}$ , queremos escrever  $\sqrt{1-4x}$  sem utilizar de raízes:

$$\begin{aligned} \sqrt{1-4x} &= (1-4x)^{1/2} \\ &= \sum_{n \geq 0} \binom{1/2}{n} (-4x)^n \\ &= \sum_{n \geq 0} \frac{1}{2} \cdot \frac{-1}{2} \cdots \frac{-(2n-3)}{2} \cdot \frac{1}{n!} \cdot (-4)^n x^n \\ &= 1 + \sum_{n \geq 1} -\frac{2}{n} \cdot \binom{2n-2}{n-1} x^n \end{aligned}$$

Portanto,

$$\begin{aligned} C(x) &= \frac{1}{2x} \left[ 1 - 1 - \sum_{n \geq 1} -\frac{2}{n} \cdot \binom{2n-2}{n-1} x^n \right] \\ &= \sum_{n \geq 1} \frac{1}{n} \cdot \binom{2n-2}{n-1} x^{n-1} \\ &= \sum_{n \geq 0} \frac{1}{n+1} \cdot \binom{2n}{n} x^n \end{aligned}$$

□

Funções geradoras são extremamente poderosas e servem para contar outras propriedades de grafos, o teorema 4.1 por exemplo, pode ser demonstrado uma terceira vez utilizando funções geradoras. Também podemos generalizar o teorema 5.1 e contar o número de árvores com  $n$  folhas tal que todos os vértices possuem grau 0 ou  $m$ .

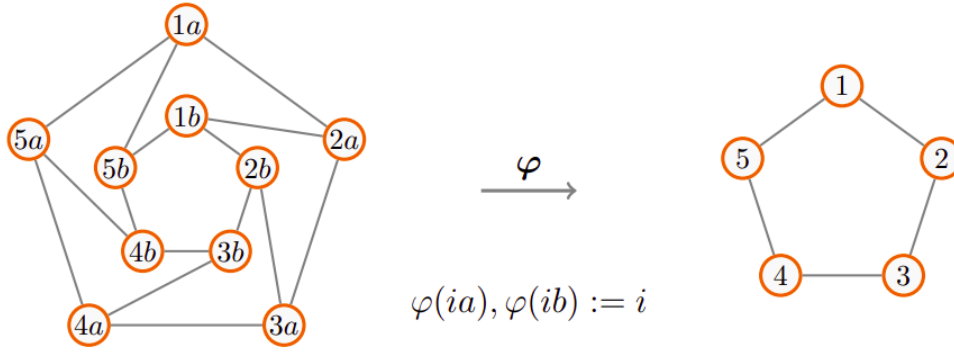
Problemas para grafos gerais também podem ser resolvidos por meios dessas funções, como por exemplo, contar o número de grafos rotulados em que todo vértice possui grau 2, ou o número de grafos bipartidos, ou o número de grafos 2-aresta conexos, isto é, grafos sem pontes.

## 6. Contagem de homomorfismos

Nesta seção, fornecemos uma introdução ao conceito de homomorfismo de grafos. Adicionalmente, apresentamos as provas de resultados relevantes relacionados à contagem de

homomorfismo. Nossa intenção nesta seção é motivar o principal resultado deste artigo, que relaciona contagem de homomorfismo com isomorfismo quântico.

Dados grafos  $G$  e  $H$ , uma função  $\varphi : V(G) \rightarrow V(H)$  é um *homomorfismo* se  $\varphi(x) \sim \varphi(y)$  sempre  $x \sim y$ , onde  $\sim$  denota “é adjacente a”. Usamos  $G \rightarrow H$  para denotar a existência de um homomorfismo de  $G$  a  $H$  [Roberson 2013].



**Figura 8. Homomorfismo do Grafo de Petersen para  $C_5$ . Fonte: [Roberson 2013]**

Algumas propriedades importantes do homomorfismo devem ser esclarecidas:

- Se  $\varphi(x) = \varphi(y)$ , então  $x$  e  $y$  não são adjacentes em  $G$ . Portanto, a pré-imagem  $\varphi^{-1}(y)$  de um vértice em  $H$  é necessariamente um conjunto independente em  $G$ .
- Seja  $\varphi$  um homomorfismo de um grafo  $G$  para um grafo  $H$ , não precisa ser sobrejetivo, ou seja,  $\text{Im}(\varphi)$  pode ser qualquer subgrafo de  $H$ .
- Os homomorfismos são transitivos: se  $G \rightarrow H$  e  $H \rightarrow Z$ , então  $G \rightarrow Z$ . Segue por definição que se  $\varphi_1 : G \rightarrow H$  e  $\varphi_2 : H \rightarrow Z$  são homomorfismo, então  $\varphi_1 \circ \varphi_2$  é um homomorfismo de  $G$  a  $Z$ .
- Se  $G \rightarrow H$  e  $H \rightarrow G$ , então não é verdade que  $G$  é isomórfico a  $H$ . Um exemplo deste fato é considerar que todo grafo bipartido é homomórfico a  $K_2$  e  $K_2$  é homomórfico a qualquer aresta de um grafo.

Para uma família de grafos  $\mathcal{F}$ , seja  $G \cong_{\mathcal{F}} H$  denotando que  $\text{hom}(F, G) = \text{hom}(F, H)$  para todo  $F \in \mathcal{F}$ . Onde  $\text{hom}(F, G)$  denota o número de homomorfismos de  $F$  a  $G$ . Perceba que  $\text{hom}(K_1, G) = |V(G)|$  e  $\text{hom}(K_2, G) = 2 \cdot |E(G)|$ .

A tabela a seguir apresenta alguns exemplos da relação entre equivalência de grafos e contagem de homomorfismo, alguns desses resultados são demonstrados nesta mesma seção.

A família $\mathcal{F}$	A relação $G \cong_{\mathcal{F}} H$
Todos os grafos	Isomorfismo [Lovász 2012]
Ciclos	Cospectralidade
Ciclos e caminhos	Cospectral com complemento cospectral
Árvores	Isomorfismo fracionário [Dvorak 2009], [Dell et al. 2018]
Treewidth $\leq k$	Indistinguível pelo algoritmo $k$ -WL [Dvorak 2009], [Dell et al. 2018]
Grafos planares	Isomorfismo quântico [Mančinska and Roberson 2020]

**Tabela 1. Equivalências na contagem de homomorfismos**

---

**Teorema 6.1.** *Dois grafos são isomórficos se e somente se  $\text{hom}(F, G) = \text{hom}(F, H)$  para qualquer grafo  $F$  [Lovász 2012].*

*Demonstração.* Como o homomorfismo é transitivo, é claro que se  $G$  é isomórfico a  $H$ , então  $\text{hom}(F, G) = \text{hom}(F, H)$ .

Assuma que  $\text{hom}(F, G) = \text{hom}(F, H)$ . Seja  $\text{Inj}(A, B)$  o conjunto de homomorfismos injetivos de um grafo  $A$  para um grafo  $B$ . Nosso objetivo é mostrar que para todo  $F$  temos  $|\text{Inj}(F, G)| = |\text{Inj}(F, H)|$ . Tomando  $F$  igual a  $G$  e depois  $H$ , vemos que existem homomorfismos injetivos de  $G$  a  $H$  e de  $H$  a  $G$ . Como  $G$  e  $H$  devem ter o mesmo número de vértices, um homomorfismo injetivo é sobrejetivo e, portanto,  $G$  é isomórfico a  $H$ .

Provamos que  $|\text{Inj}(F, G)| = |\text{Inj}(F, H)|$  por indução no número de vértices em  $F$ . É claramente verdadeiro se  $F$  tiver um vértice, porque qualquer homomorfismo de um único vértice é injetivo.

Podemos particionar os homomorfismos de  $F$  em qualquer grafo  $W$  de acordo com o kernel, então obtemos

$$|\text{Hom}(F, W)| = \sum_{\pi} |\text{Inj}(F/\pi, W)|,$$

onde  $\pi$  varia em todas as partições. Um homomorfismo é uma injeção se e somente se seu kernel é a partição discreta, que denotaremos por  $\delta$ . Portanto,

$$|\text{Inj}(F, W)| = |\text{Hom}(F, W)| - \sum_{\pi \neq \delta} |\text{Inj}(F/\pi, W)|,$$

Agora, pela hipótese de indução, todos os termos do lado direito desta soma são os mesmos para  $W = G$  e  $W = H$ . Portanto, concluímos que

$$|\text{Inj}(F, G)| = |\text{Inj}(F, H)|,$$

e segue o resultado. □

**Teorema 6.2.** *Seja  $\mathcal{C}$  a classe de todos os ciclos (incluindo o ciclo degenerado de comprimento 0, que é apenas um único vértice). Para todos os grafos  $G$  e  $H$ , temos  $G \cong_{\mathcal{C}} H$  se e somente se  $G$  e  $H$  são coespectrais [Dell et al. 2018].*

Para a prova, revisamos alguns fatos simples da álgebra linear. O traço  $\text{tr}(A)$  é a soma das entradas diagonais. Se os autovalores de  $A$  são  $\lambda_1, \dots, \lambda_n$ , então  $\text{tr}(A) = \sum_{i=1}^n \lambda_i$ . Além disso, para cada  $l \geq 0$ , os autovalores da matriz  $A^l$  são  $\lambda_1^l, \dots, \lambda_n^l$  e, portanto,  $\text{tr}(A^l) = \sum_{i=1}^n \lambda_i^l$ .

O seguinte lema técnico encapsula o fato de que a informação  $\text{tr}(A^l)$  para todo  $l \in \mathbb{N}$  é suficiente para reconstruir o espectro de  $A$  com multiplicidades.

**Lema 6.1.** *Para matrizes  $n \times n$   $A$  e  $B$ , as seguintes são equivalentes [van Dam and Haemers 2003]:*

- (i)  $A$  e  $B$  são coespectrais.
- (ii)  $A$  e  $B$  possuem o mesmo polinômio característico.
- (iii)  $\text{tr}(A^l) = \text{tr}(B^l)$  para todo  $l \in \mathbb{N}$

*Demonstração.* Para verificar a equivalência de (iii) para (ii) basta lembrar que pelas identidades de Newton as raízes  $\lambda_1, \dots, \lambda_n$  de um polinômio de grau  $n$  são determinadas pelas somas das potências  $\sum_{i=1}^n \lambda_i^l$ .  $\square$

*Prova do Teorema 6.2.* Para todo  $l \geq 0$ , o número de homomorfismos do ciclo  $C_l$  para um grafo  $G$  com matriz de adjacência  $A$  é igual ao número de comprimentos  $l$  fechados em  $G$ , que por sua vez é igual ao traço de  $A^l$  3.1. Assim, para grafos  $G, H$  com matrizes de adjacência  $A, B$ , temos  $G \cong_c H$  se e somente se  $\text{tr}(A^l) = \text{tr}(B^l)$  vale para todo  $l \geq 0$ .

Se  $A$  e  $B$  tiverem o mesmo espectro  $\lambda_1 \dots \lambda_n$ , então  $\text{tr}(A^l) = \lambda_1^l + \dots + \lambda_n^l = \text{tr}(B^l)$  vale para todo  $l \in \mathbb{N}$ , assim, pelo lema 6.1,  $G$  e  $H$  são coespectrais.  $\square$

No exemplo a seguir, mostramos que as contagens de homomorfismo de ciclos e árvores são incomparáveis em sua expressividade. Os dois gráficos da figura 9 são coespectrais, e portanto  $G \cong_c H$ , mas é fácil ver que  $\text{hom}(P_3, G) \neq \text{hom}(P_3, H)$ .



**Figura 9. Grafos coespectrais mas não isomorfos.**

## 7. Conclusão

Em conclusão, este estudo ressalta a extrema importância da álgebra no campo da contagem de estruturas combinatórias. Ao explorar diferentes áreas da álgebra e utilizar técnicas diversas, pudemos constatar como um mesmo problema, quando analisado sob diferentes ferramentas, revela propriedades únicas e fascinantes. Essa abordagem multifacetada amplia nossa compreensão das estruturas de grafos e nos permite explorar novos domínios de conhecimento.

Além disso, fica evidente a relevância de não apenas conhecer os resultados existentes, mas também ser capaz de demonstrar os teoremas subjacentes. A habilidade de desenvolver demonstrações sólidas não só valida os resultados encontrados, mas também gera um conjunto de técnicas valiosas que podem ser aplicadas em contextos diversos. Essas técnicas podem revelar-se incrivelmente úteis na resolução de problemas em outras áreas da matemática e até mesmo em outros campos científicos.

Portanto, destacamos a importância de promover a interdisciplinaridade entre a teoria algébrica de grafos e outras áreas do conhecimento. Essa abordagem ampla e colaborativa é fundamental para avançar o campo, desvendar novas perspectivas e abrir caminho para descobertas ainda mais inovadoras.



---

## Referências

- [Biggs 1993] Biggs, N. (1993). *Algebraic Graph Theory*. Cambridge University Press, Cambridge.
- [Chaiken and Kleitman 1978] Chaiken, S. and Kleitman, D. (1978). Matrix tree theorems. *Journal of Combinatorial Theory, Series A*, 24(3):377–381.
- [Coutinho 2020] Coutinho, G. (2020). Algebraic combinatorics an overview - universidade federal de minas gerais.
- [Dell et al. 2018] Dell, H., Grohe, M., and Rattan, G. (2018). Lovász meets weisfeiler and leman. In *International Colloquium on Automata, Languages and Programming*.
- [Dvorak 2009] Dvorak, Z. (2009). On recognizing graphs by numbers of homomorphisms. *Journal of Graph Theory*, 64:330 – 342.
- [(<https://codeforces.com/profile/zscoder>) 2020] (<https://codeforces.com/profile/zscoder>), Z. S. Y. (2020). [tutorial] generating functions in competitive programming (part 1). Codeforces.
- [Ilfen 2017] Ilfen (2017). Number of even spanning subgraphs. Mathematics Stack Exchange. URL:<https://math.stackexchange.com/q/2574962> (version: 2021-01-09).
- [Jin 2018] Jin, J. (2018). Analysis and applications of burnside ’ s lemma. In .
- [Kurzweil and Stellmacher 2003] Kurzweil, H. and Stellmacher, B. (2003). *The Theory of Finite Groups: An Introduction*. Universitext. Springer New York.
- [Lovász 2012] Lovász, L. (2012). *Large Networks and Graph Limits.*, volume 60 of *Colloquium Publications*. American Mathematical Society.
- [Mančinska and Roberson 2020] Mančinska, L. and Roberson, D. E. (2020). Quantum isomorphism is equivalent to equality of homomorphism counts from planar graphs. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 661–672.
- [Roberson 2013] Roberson, D. E. (2013). Variations on a theme: Graph homomorphisms. In *Variations on a Theme: Graph Homomorphisms*.
- [van Dam and Haemers 2003] van Dam, E. R. and Haemers, W. H. (2003). Which graphs are determined by their spectrum? *Linear Algebra and its Applications*, 373:241–272. Combinatorial Matrix Theory Conference (Pohang, 2002).