

# Técnica de *Shifting*

---

Lucas de Oliveira Silva - 220715

Ricardo Andres Marino Rojas - 175824

MO829 - Algoritmos Parametrizados

1° de Julho de 2025

# Introdução

## Problema Exemplo

CONJUNTO INDEPENDENTE MÁXIMO (CI)

**Instância:** Grafo  $G = (V, E)$ .

## Problema Exemplo

CONJUNTO INDEPENDENTE MÁXIMO (CI)

**Instância:** Grafo  $G = (V, E)$ .

**Objetivo:**  $S \subseteq V$  tal que  $|E(G[S])| = 0$ , com  $|S|$  máximo.

## Teorema ([GJ79])

*O problema CI é NP-difícil, mesmo em grafos cúbicos e planares.*

## Teorema ([CFM<sup>+</sup>15])

O problema *CI* parametrizado pelo tamanho da solução  $k$  é  $W[1]$ -completo.

## Outras Classes de Grafos

- CI em árvores?

## Outras Classes de Grafos

- CI em árvores?

EZ –

## Outras Classes de Grafos

- CI em árvores?
- E quanto a outras classes?

EZ –

## Outras Classes de Grafos

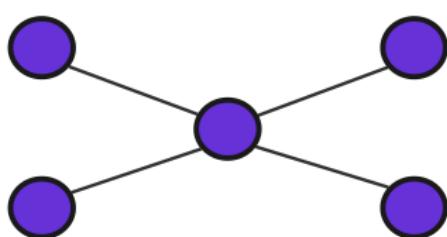
- CI em árvores?  
EZ –
- E quanto a outras classes?  
Você já ouviu falar de grafos planares? –

# Preliminares

Um grafo é ***planar*** se ele pode ser desenhado no plano Euclidiano sem cruzamentos de arestas.

## Grafos Planares

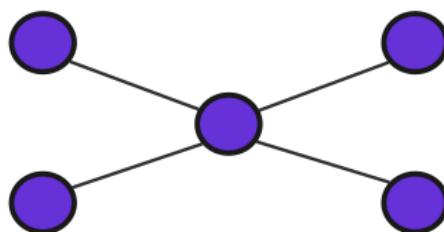
Um grafo é **planar** se ele pode ser desenhado no plano Euclidiano sem cruzamentos de arestas.



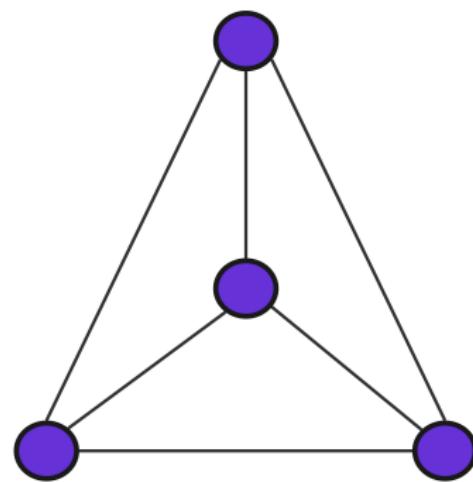
Grafo Borboleta

## Grafos Planares

Um grafo é **planar** se ele pode ser desenhado no plano Euclidiano sem cruzamentos de arestas.

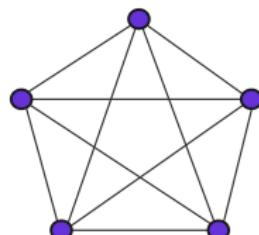
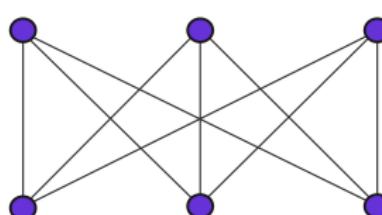
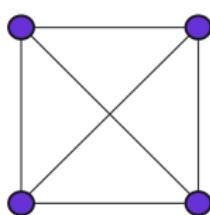


Grafo Borboleta

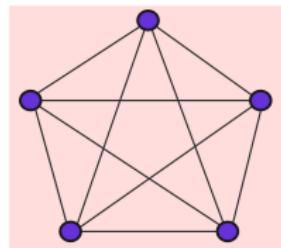
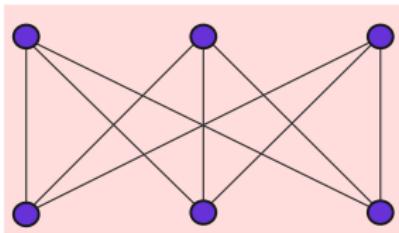
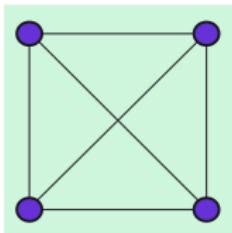


$K_4$

Qual dos três grafos a seguir é planar?

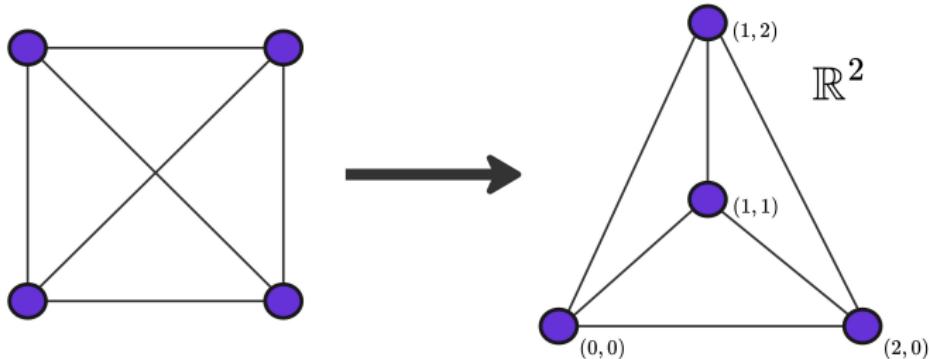


Qual dos três grafos a seguir é planar?

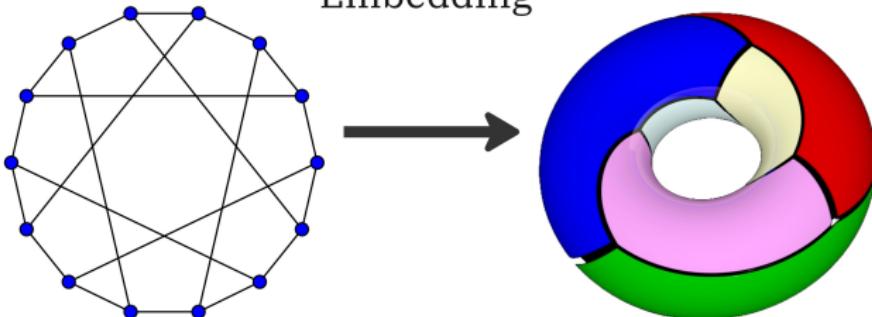


# Mais Definições

Planar embedding

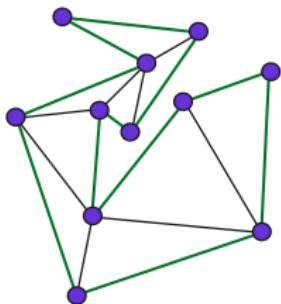


Embedding



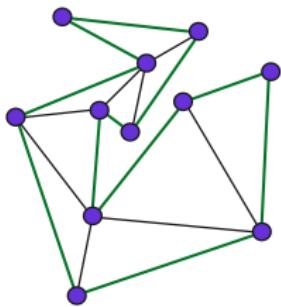
## Muito Mais Definições

Um grafo é ***outerplanar*** se admite uma imersão planar com todos os vértices na face externa.



## Muito Mais Definições

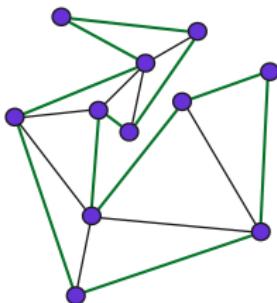
Um grafo é ***outerplanar*** se admite uma imersão planar com todos os vértices na face externa.



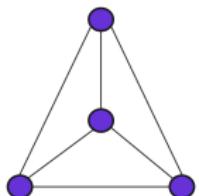
Todo grafo planar é outerplanar?

## Muito Mais Definições

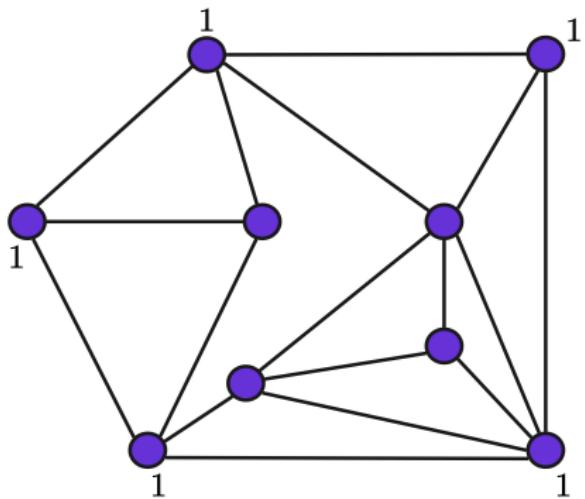
Um grafo é **outerplanar** se admite uma imersão planar com todos os vértices na face externa.



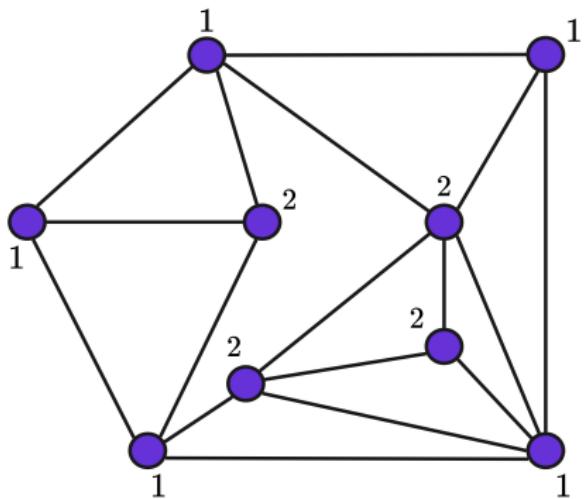
Todo grafo planar é outerplanar?  
Não! Considere o  $K_4$



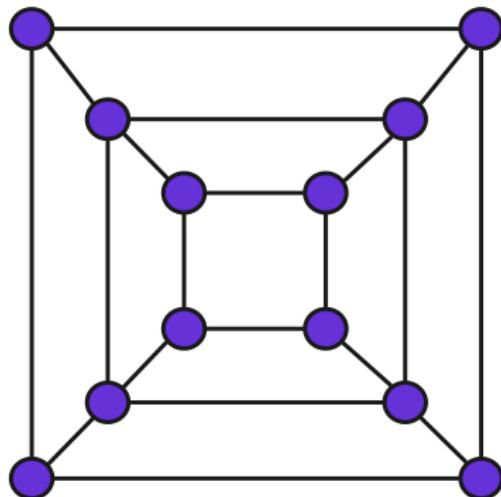
### Nível 1



### Nível 1 e Nível 2



### Nível 1, Nível 2 e Nível 3



# Quando as Definições Vão Acabar!?

## Observação

Os vértices de nível  $i$  de uma imersão planar de  $G$  são aqueles que estão na face externa após a remoção dos vértices de níveis  $1, \dots, i - 1$  e suas arestas incidentes.

# Quando as Definições Vão Acabar!?

## Observação

Os vértices de nível  $i$  de uma imersão planar de  $G$  são aqueles que estão na face externa após a remoção dos vértices de níveis  $1, \dots, i - 1$  e suas arestas incidentes.

## Definição

Um grafo  $G$  é  $k$ -**outerplanar** se admite uma imersão planar na qual todos os seus vértices pertencem a níveis de 1 até  $k$ .

## Largura Arbórea dos $k$ -outerplanar

**Objetivo:** Limitar a largura arbórea de grafos  $k$ -outerplanar.

**Objetivo:** Limitar a largura arbórea de grafos  $k$ -outerplanar.

**Plano:**

**Passo 1:** Considerar apenas grafos  $k$ -outerplanar de grau máximo 3

**Objetivo:** Limitar a largura arbórea de grafos  $k$ -outerplanar.

**Plano:**

**Passo 1:** Considerar apenas grafos  $k$ -outerplanar de grau máximo 3

**Passo 2:** Provar que grafos  $k$ -outerplanar de grau máximo três têm carga máxima limitada

**Objetivo:** Limitar a largura arbórea de grafos  $k$ -outerplanar.

**Plano:**

**Passo 1:** Considerar apenas grafos  $k$ -outerplanar de grau máximo 3

**Passo 2:** Provar que grafos  $k$ -outerplanar de grau máximo três têm carga máxima limitada

**Passo 3:** Provar que carga máxima limitada implica em largura arbórea limitada

# Passo 1

## Lema

*Todo grafo  $k$ -outerplanar  $G$  pode ser transformado em um grafo  $k$ -outerplanar  $G'$  de grau máximo 3, tal que uma decomposição em árvore de  $G'$  corresponde a uma de  $G$  de mesma largura.*

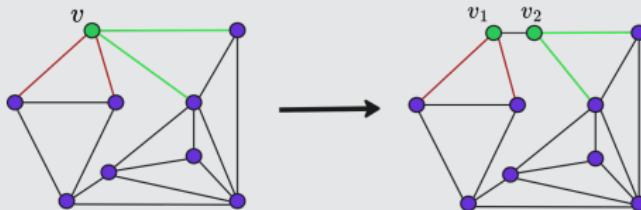
# Passo 1

## Lema

Todo grafo  $k$ -outerplanar  $G$  pode ser transformado em um grafo  $k$ -outerplanar  $G'$  de grau máximo 3, tal que uma decomposição em árvore de  $G'$  corresponde a uma de  $G$  de mesma largura.

## Demonstração.

Divida qualquer vértice  $v$  de grau  $d$  maior que 3 em dois vértices  $v_1, v_2$  de grau 3 e  $d - 1$ .



Se  $v_1$  ou  $v_2$  estão em  $X'_i$ , definimos  $X_i = \{X'_i \setminus \{v_1, v_2\}\} \cup \{v\}$ ; senão  $X_i = X'_i$ .

# Passo 1...

Demonstração.

### Demonstração.

- Crie uma sequência de decomposições  $G_1, G_2, \dots, G_z$  tal que  $G_z$  tem grau máximo 3;

## Demonstração.

- Crie uma sequência de decomposições  $G_1, G_2, \dots, G_z$  tal que  $G_z$  tem grau máximo 3;
- Obtenha uma decomposição em árvore de  $G_z$ ;

### Demonstração.

- Crie uma sequência de decomposições  $G_1, G_2, \dots, G_z$  tal que  $G_z$  tem grau máximo 3;
- Obtenha uma decomposição em árvore de  $G_z$ ;
- Construa decomposições para  $G_{z-1}, G_{z-2}, \dots, G_1, G$ , de tal forma que a decomposição final de  $G$  tenha largura no máximo igual à de  $G_z$ .



Dada uma floresta geradora maximal  $(V, F)$  de um grafo  $G$ , e uma aresta  $e \in E \setminus F$ :

## Definição (Ciclo fundamental)

O **ciclo fundamental** de  $e$  é o ciclo formado ao adicionar  $e$  à  $F$ .

# Definições Que Nunca Acabam

## Definição (Carga de vértice)

A **carga de um vértice**  $v \in V$  é o número de ciclos fundamentais em que  $v$  aparece.

# Definições Que Nunca Acabam

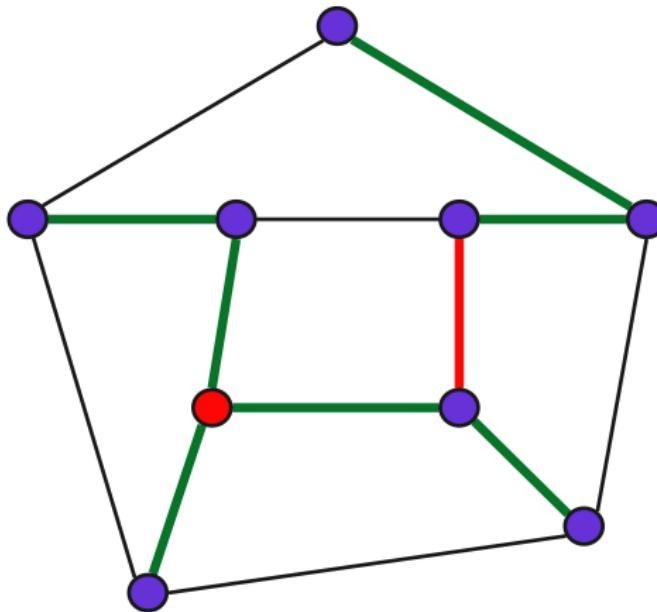
## Definição (Carga de vértice)

A **carga de um vértice**  $v \in V$  é o número de ciclos fundamentais em que  $v$  aparece.

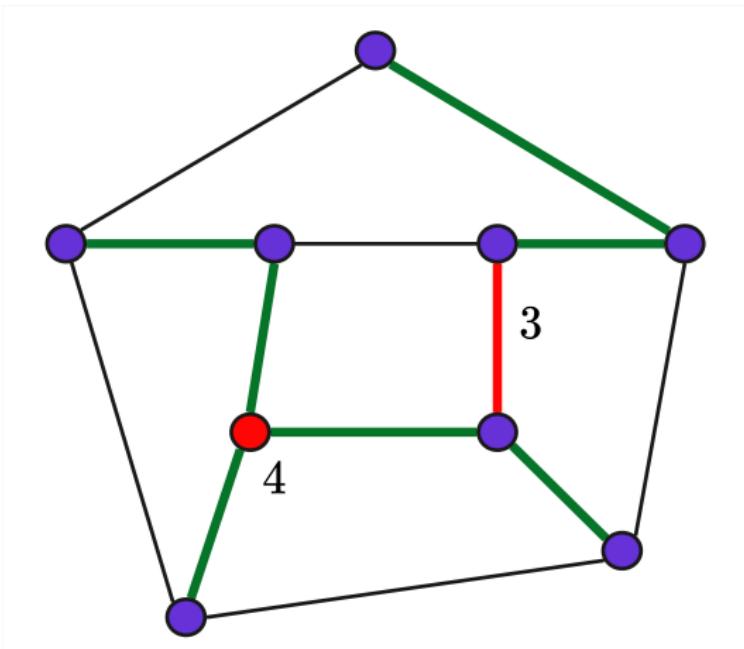
## Definição (Carga aresta)

A **carga de uma aresta**  $e \in F$  é o número de ciclos fundamentais que a contêm.

Encontre a carga do vértice e aresta vermelhos



Encontre a carga do vértice e aresta vermelhos



## Última Definição (desta seção)

### Definição (Carga Máxima)

A **carga máxima** de uma floresta geradora maximal  $(V, F)$  de um grafo  $G$  é o maior valor entre as cargas de seus vértices  $v \in V$  e arestas  $e \in F$ .

## Passo 2

### Lema

*Todo grafo  $k$ -outerplanar com grau máximo 3 admite uma floresta geradora maximal com carga máxima  $\leq 3k$ .*

## Passo 2

### Lema

*Todo grafo  $k$ -outerplanar com grau máximo 3 admite uma floresta geradora maximal com carga máxima  $\leq 3k$ .*

## Passo 2

### Lema

*Todo grafo  $k$ -outerplanar com grau máximo 3 admite uma floresta geradora maximal com carga máxima  $\leq 3k$ .*

Procedemos por indução em  $k$ .

## Passo 2...

### Demonstração.

Suponha  $k = 1$  e seja  $R$  o conjunto de arestas da face externa da imersão de  $G$ :



## Passo 2...

### Demonstração.

Suponha  $k = 1$  e seja  $R$  o conjunto de arestas da face externa da imersão de  $G$ :

- Remova  $R$ ; o grafo resultante é acíclico.



## Passo 2...

### Demonstração.

Suponha  $k = 1$  e seja  $R$  o conjunto de arestas da face externa da imersão de  $G$ :

- Remova  $R$ ; o grafo resultante é acíclico.
- Estenda  $E \setminus R$  para uma floresta geradora maximal  $F$ , adicionando o máximo possível de arestas de  $R$ .



## Passo 2...

### Demonstração.

Suponha  $k = 1$  e seja  $R$  o conjunto de arestas da face externa da imersão de  $G$ :

- Remova  $R$ ; o grafo resultante é acíclico.
- Estenda  $E \setminus R$  para uma floresta geradora maximal  $F$ , adicionando o máximo possível de arestas de  $R$ .
- Considere as arestas em  $E \setminus F$  e as faces internas que elas formam.



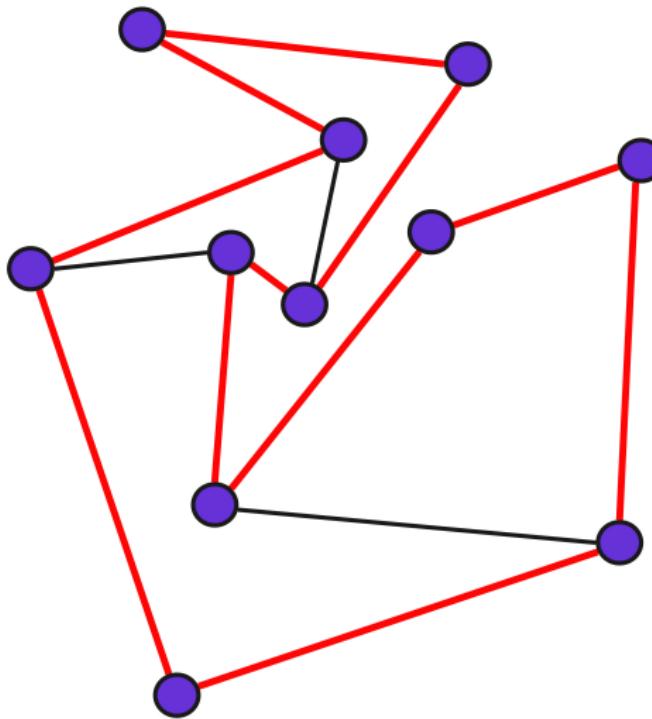
### Demonstração.

Suponha  $k = 1$  e seja  $R$  o conjunto de arestas da face externa da imersão de  $G$ :

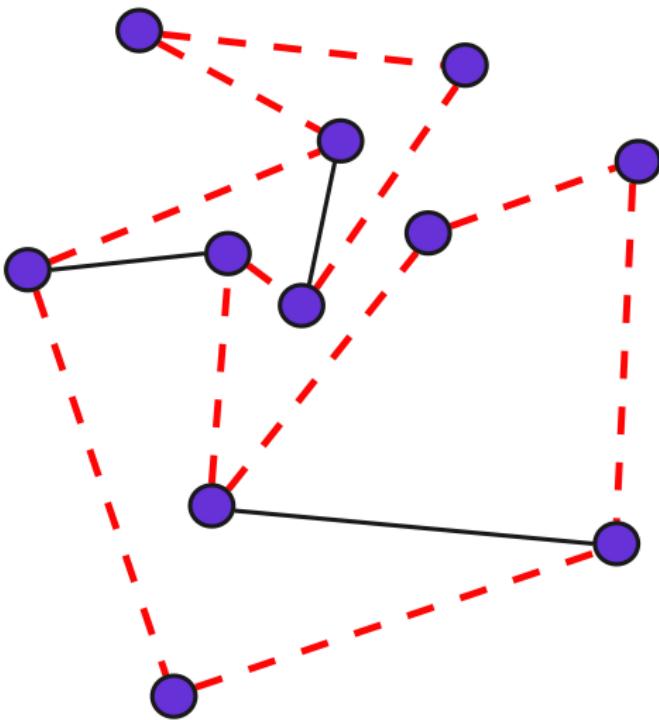
- Remova  $R$ ; o grafo resultante é acíclico.
- Estenda  $E \setminus R$  para uma floresta geradora maximal  $F$ , adicionando o máximo possível de arestas de  $R$ .
- Considere as arestas em  $E \setminus F$  e as faces internas que elas formam.
- Cada aresta de  $G$  delimita no máximo 2 faces internas, e cada vértice  $v \in V$  participa de no máximo  $\deg(v)$  delas.



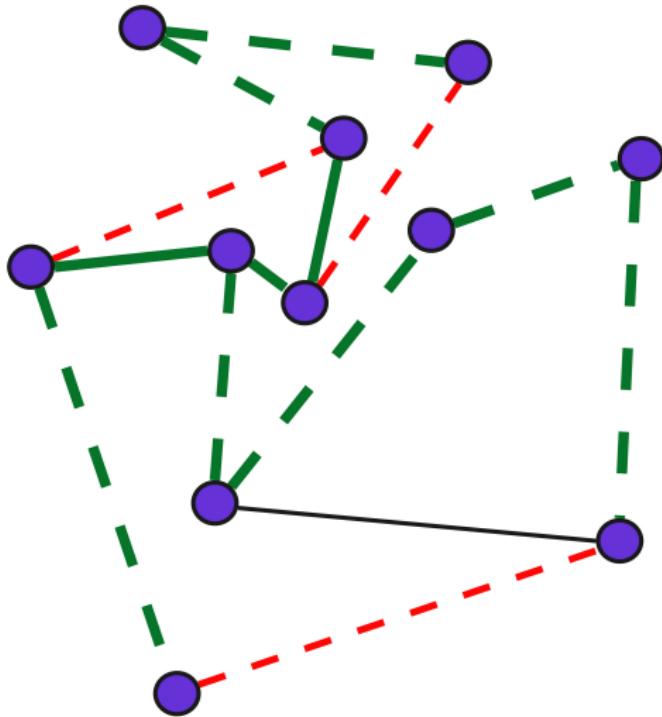
## Exemplo



## Exemplo



## Exemplo



## Passo 2...

### Demonstração.

Para  $k > 1$ , seja  $R$  o conjunto de arestas da face externa da imersão de  $G$ :



## Passo 2...

### Demonstração.

Para  $k > 1$ , seja  $R$  o conjunto de arestas da face externa da imersão de  $G$ :

- Remova  $R$ ; o grafo restante é, no máximo,  $(k - 1)$ -outerplanar.



## Passo 2...

### Demonstração.

Para  $k > 1$ , seja  $R$  o conjunto de arestas da face externa da imersão de  $G$ :

- Remova  $R$ ; o grafo restante é, no máximo,  $(k - 1)$ -outerplanar.
- Por indução, existe uma floresta geradora maximal  $F'$  com carga máxima  $\leq 3(k - 1)$ .



## Passo 2...

### Demonstração.

Para  $k > 1$ , seja  $R$  o conjunto de arestas da face externa da imersão de  $G$ :

- Remova  $R$ ; o grafo restante é, no máximo,  $(k - 1)$ -outerplanar.
- Por indução, existe uma floresta geradora maximal  $F'$  com carga máxima  $\leq 3(k - 1)$ .
- Estenda  $F'$  para uma floresta geradora maximal  $F$  de  $G$ , adicionando arestas de  $R$ .



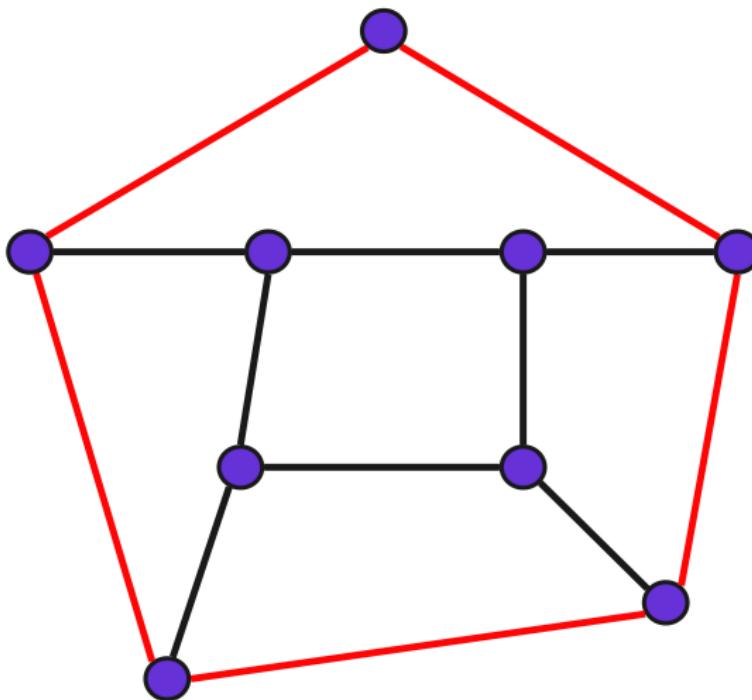
### Demonstração.

Para  $k > 1$ , seja  $R$  o conjunto de arestas da face externa da imersão de  $G$ :

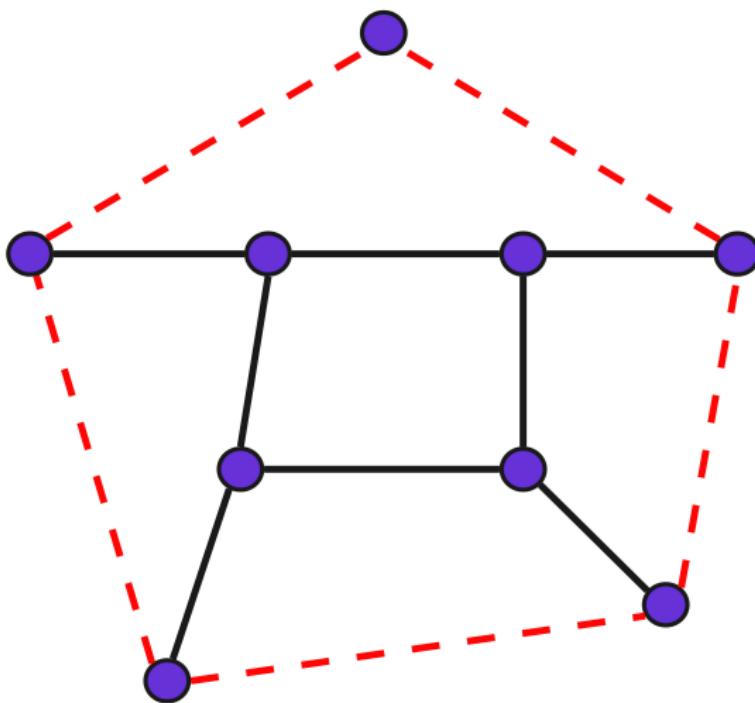
- Remova  $R$ ; o grafo restante é, no máximo,  $(k - 1)$ -outerplanar.
- Por indução, existe uma floresta geradora maximal  $F'$  com carga máxima  $\leq 3(k - 1)$ .
- Estenda  $F'$  para uma floresta geradora maximal  $F$  de  $G$ , adicionando arestas de  $R$ .
- A carga adicional em qualquer aresta de  $F$  é no máximo 2, e em qualquer vértice, no máximo 3.



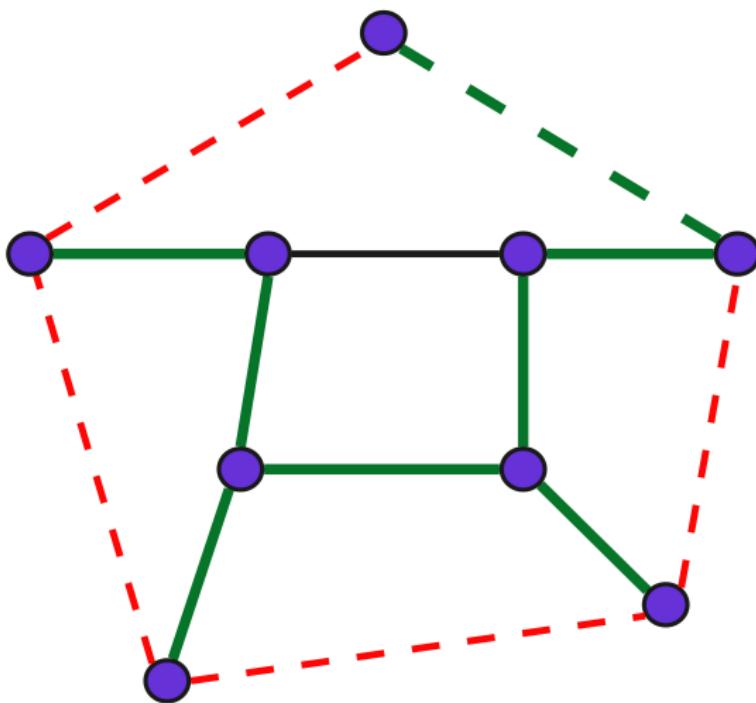
## Exemplo



## Exemplo



## Exemplo



## Passo 3

### Lema

Se um grafo  $G$  de grau máximo 3 admite uma floresta geradora maximal  $(V, F)$  com carga máxima  $\leq \ell$ , então a largura da árvore de  $G$  é no máximo  $\ell + 1$ .

## Passo 3...

### Demonstração.

Construa uma decomposição em árvore para a floresta  $F$  da seguinte forma:

## Passo 3...

### Demonstração.

Construa uma decomposição em árvore para a floresta  $F$  da seguinte forma:

- Para cada vértice  $u \in V$  e cada aresta  $e = (v, w) \in F$ , crie nós  $t_u$  e  $t_e$  em  $T$ .

## Passo 3...

### Demonstração.

Construa uma decomposição em árvore para a floresta  $F$  da seguinte forma:

- Para cada vértice  $u \in V$  e cada aresta  $e = (v, w) \in F$ , crie nós  $t_u$  e  $t_e$  em  $T$ .
- Iniciamos com  $X_u = \{u\}$  e  $X_e = \{v, w\}$ .

## Passo 3...

### Demonstração.

Construa uma decomposição em árvore para a floresta  $F$  da seguinte forma:

- Para cada vértice  $u \in V$  e cada aresta  $e = (v, w) \in F$ , crie nós  $t_u$  e  $t_e$  em  $T$ .
- Iniciamos com  $X_u = \{u\}$  e  $X_e = \{v, w\}$ .
- Para cada aresta  $e = (u, v) \in E \setminus F$ , escolha arbitrariamente um de seus extremos, digamos  $u$ .

## Passo 3...

### Demonstração.

Construa uma decomposição em árvore para a floresta  $F$  da seguinte forma:

- Para cada vértice  $u \in V$  e cada aresta  $e = (v, w) \in F$ , crie nós  $t_u$  e  $t_e$  em  $T$ .
- Iniciamos com  $X_u = \{u\}$  e  $X_e = \{v, w\}$ .
- Para cada aresta  $e = (u, v) \in E \setminus F$ , escolha arbitrariamente um de seus extremos, digamos  $u$ .
- Para cada vértice  $w \neq v$  no ciclo fundamental de  $e$ , adicione  $u$  a  $X_w$ .

## Passo 3...

### Demonstração.

Construa uma decomposição em árvore para a floresta  $F$  da seguinte forma:

- Para cada vértice  $u \in V$  e cada aresta  $e = (v, w) \in F$ , crie nós  $t_u$  e  $t_e$  em  $T$ .
- Iniciamos com  $X_u = \{u\}$  e  $X_e = \{v, w\}$ .
- Para cada aresta  $e = (u, v) \in E \setminus F$ , escolha arbitrariamente um de seus extremos, digamos  $u$ .
- Para cada vértice  $w \neq v$  no ciclo fundamental de  $e$ , adicione  $u$  a  $X_w$ .
- Para cada aresta  $e' \in F$  no ciclo fundamental de  $e$ , adicione  $u$  a  $X_{e'}$ .

## Passo 3...

### Demonstração.

Note que  $(T, \{X_t\}_{t \in V(T)})$  é uma decomposição em árvore de  $G$ :

## Passo 3...

### Demonstração.

Note que  $(T, \{X_t\}_{t \in V(T)})$  é uma decomposição em árvore de  $G$ :

**Propriedade 1:** Satisfeita trivialmente.

## Passo 3...

### Demonstração.

Note que  $(T, \{X_t\}_{t \in V(T)})$  é uma decomposição em árvore de  $G$ :

**Propriedade 1:** Satisfeta trivialmente.

**Propriedade 2:** Para  $e = (u, v) \in E \setminus F$ ,  $u$  foi adicionado a algum  $X_{(w,v)}$  no ciclo fundamental de  $e$ , garantindo que  $u$  e  $v$  coexistem em um mesmo conjunto.

## Passo 3...

### Demonstração.

Note que  $(T, \{X_t\}_{t \in V(T)})$  é uma decomposição em árvore de  $G$ :

**Propriedade 1:** Satisfeta trivialmente.

**Propriedade 2:** Para  $e = (u, v) \in E \setminus F$ ,  $u$  foi adicionado a algum  $X_{(w,v)}$  no ciclo fundamental de  $e$ , garantindo que  $u$  e  $v$  coexistem em um mesmo conjunto.

**Propriedade 3:** Sempre que um vértice  $v$  foi adicionado a subconjuntos, isso ocorreu ao longo de um caminho da árvore, mantendo a propriedade de conectividade.

## Passo 3...

### Demonstração.

Note que  $(T, \{X_t\}_{t \in V(T)})$  é uma decomposição em árvore de  $G$ :

**Propriedade 1:** Satisfeta trivialmente.

**Propriedade 2:** Para  $e = (u, v) \in E \setminus F$ ,  $u$  foi adicionado a algum  $X_{(w,v)}$  no ciclo fundamental de  $e$ , garantindo que  $u$  e  $v$  coexistem em um mesmo conjunto.

**Propriedade 3:** Sempre que um vértice  $v$  foi adicionado a subconjuntos, isso ocorreu ao longo de um caminho da árvore, mantendo a propriedade de conectividade.

## Passo 3...

### Demonstração.

Note que  $(T, \{X_t\}_{t \in V(T)})$  é uma decomposição em árvore de  $G$ :

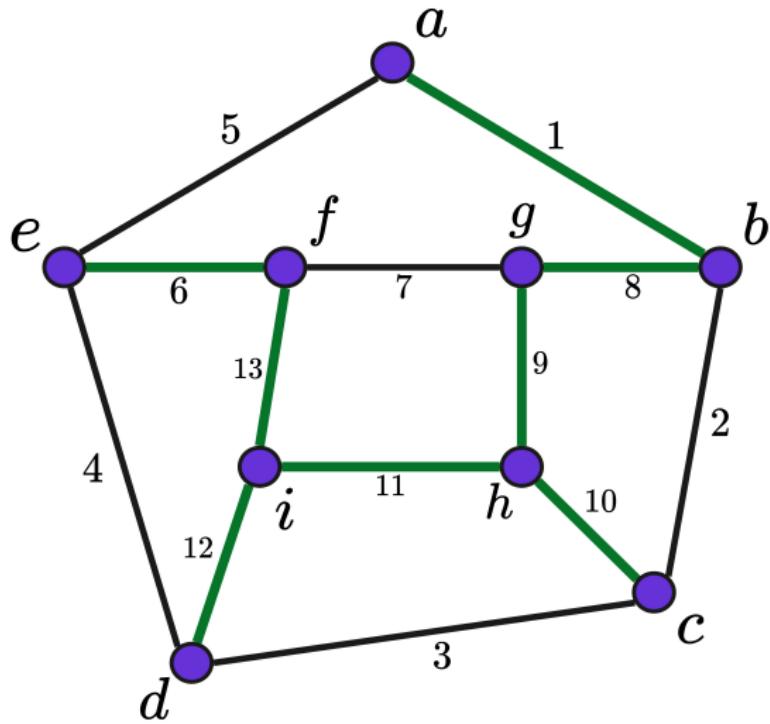
**Propriedade 1:** Satisfeita trivialmente.

**Propriedade 2:** Para  $e = (u, v) \in E \setminus F$ ,  $u$  foi adicionado a algum  $X_{(w,v)}$  no ciclo fundamental de  $e$ , garantindo que  $u$  e  $v$  coexistem em um mesmo conjunto.

**Propriedade 3:** Sempre que um vértice  $v$  foi adicionado a subconjuntos, isso ocorreu ao longo de um caminho da árvore, mantendo a propriedade de conectividade.

Cada conjunto  $X_u$  ou  $X_e$  recebeu no máximo tantos vértices quanto a carga de  $u \in V$  ou  $e \in F$ , respectivamente. Como os conjuntos iniciais tinham no máximo 2 vértices, o tamanho final é no máximo  $\ell + 2$ . □

## Exemplo



# Lema Principal

## Lema (1)

Há um algoritmo para CI em grafos  $k$ -outerplanar que executa em tempo  $2^{O(k)} \cdot n$ .

## Resultado Principal

# Teorema Principal

## Teorema ([CFM<sup>+</sup>15, WS11])

*Existe um algoritmo que, dada uma instância  $I = (G, k)$  do problema CI em que  $G$  é planar, resolve  $I$  em tempo  $2^{O(k)} \cdot n^{O(1)}$ .*

## Observação

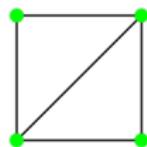
---

## Observação

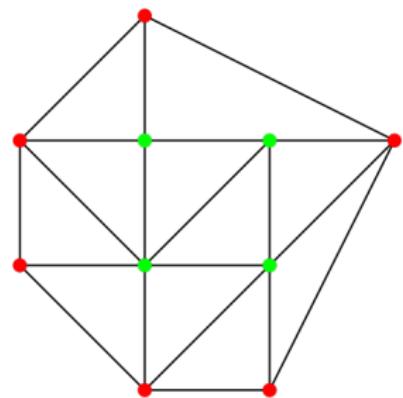
Assim como ogros, grafos planares têm camadas!



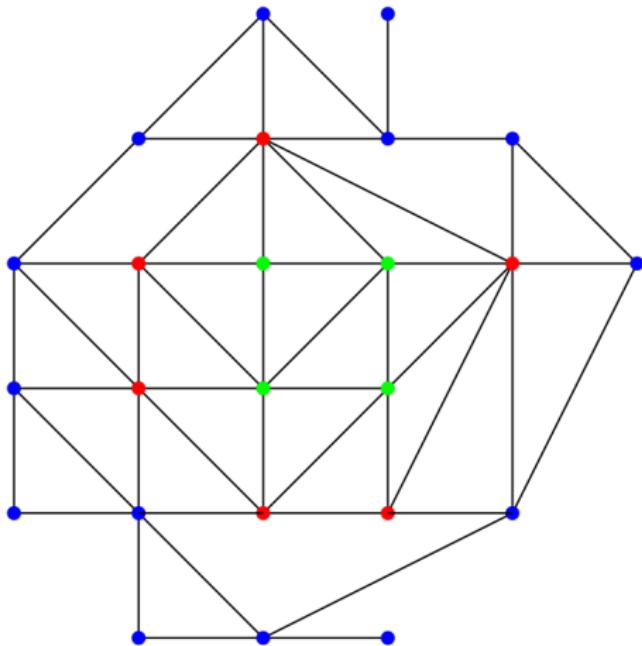
## Relembrando



## Relembrando



## Relembrando



# Preparação

---

Dado  $G = (V, E)$  planar e  $k \in \mathbb{N}^*$ , definimos:

# Preparação

---

Dado  $G = (V, E)$  planar e  $k \in \mathbb{N}^*$ , definimos:

- $\ell = k + 1$ ;

Dado  $G = (V, E)$  planar e  $k \in \mathbb{N}^*$ , definimos:

- $\ell = k + 1$ ;
- Para  $0 \leq i < \ell$ , seja  $S_i = \bigcup\{L_j \mid j \equiv i \pmod{\ell}\}$

Dado  $G = (V, E)$  planar e  $k \in \mathbb{N}^*$ , definimos:

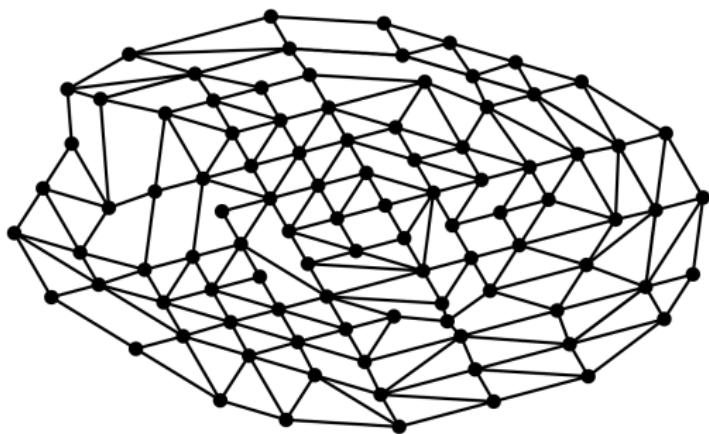
- $\ell = k + 1$ ;
- Para  $0 \leq i < \ell$ , seja  $S_i = \bigcup\{L_j \mid j \equiv i \pmod{\ell}\}$ 
  - Ex.:  $\ell = 4 \Rightarrow S_1 = L_1 \cup L_5 \cup L_9 \cup \dots$

Dado  $G = (V, E)$  planar e  $k \in \mathbb{N}^*$ , definimos:

- $\ell = k + 1$ ;
- Para  $0 \leq i < \ell$ , seja  $S_i = \bigcup\{L_j \mid j \equiv i \pmod{\ell}\}$ 
  - Ex.:  $\ell = 4 \Rightarrow S_1 = L_1 \cup L_5 \cup L_9 \cup \dots$
- $G_i = G[V - S_i]$ .

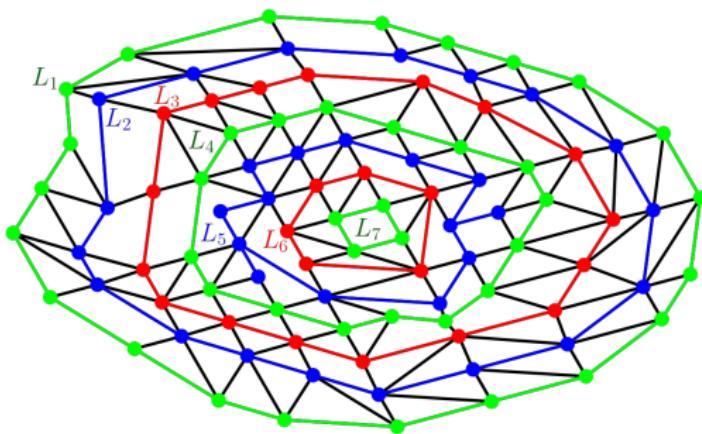


$\ell = 3$ :



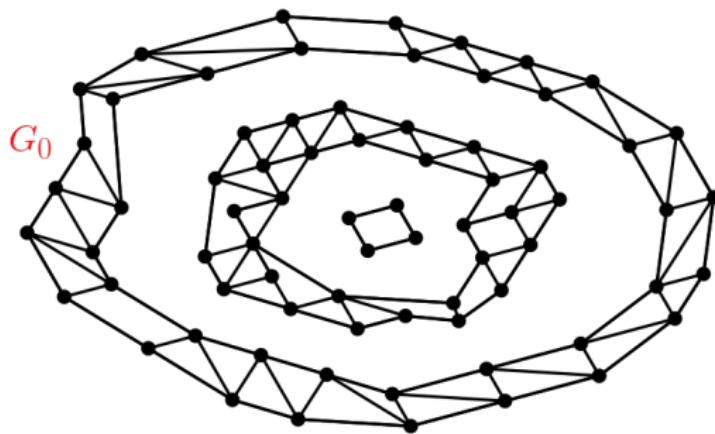


$\ell = 3$ :



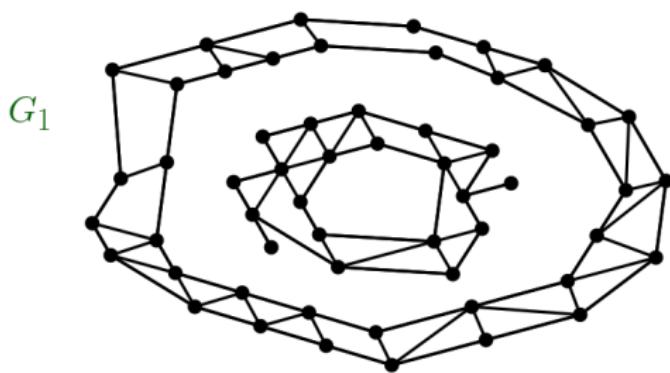


$\ell = 3$ :



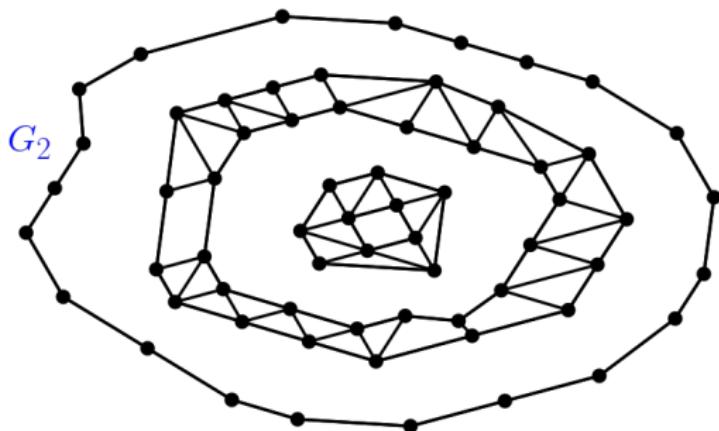


$\ell = 3$ :





$\ell = 3:$



## Solucionando $G_i$

---

Usamos o Lema 1 em cada componente.

## Solução Ótima para $G_i$

---

Para cada  $i$ , geramos uma solução ótima  $X_i$  de  $G_i$ .

O tempo total gasto é  $2^{O(k)} \cdot n$ .

Basta retornar o  $X_\alpha$  de maior cardinalidade!

# Corretude

---

## Demonstração.

1. Seja  $O \subseteq V$  uma solução ótima;
2. Suponha que  $|O| \geq k$ ;
3.  $S_0, \dots, S_{\ell-1}$  particionam  $V$ ;
4. Algum  $i$  satisfaaz  $|O \cap V(G_i)| \geq k$ ;
5.  $O \setminus S_i$  é independente em  $G_i$ ;
6. Então  $|X_\alpha| \geq |O \setminus S_i| = |O \cap V(G_i)| \geq k$ .

## Demonstração.

1. Seja  $O \subseteq V$  uma solução ótima;
2. Suponha que  $|O| \geq k$ ;
3.  $S_0, \dots, S_{\ell-1}$  particionam  $V$ ;
4. Algum  $i$  satisfaz  $|O \cap V(G_i)| \geq k$ ;
5.  $O \setminus S_i$  é independente em  $G_i$ ;
6. Então  $|X_\alpha| \geq |O \setminus S_i| = |O \cap V(G_i)| \geq k$ .

## Demonstração.

1. Seja  $O \subseteq V$  uma solução ótima;
2. Suponha que  $|O| \geq k$ ;
3.  $S_0, \dots, S_{\ell-1}$  particionam  $V$ ;
4. Algum  $i$  satisfaz  $|O \cap V(G_i)| \geq k$ ;
5.  $O \setminus S_i$  é independente em  $G_i$ ;
6. Então  $|X_\alpha| \geq |O \setminus S_i| = |O \cap V(G_i)| \geq k$ .

## Demonstração.

1. Seja  $O \subseteq V$  uma solução ótima;
2. Suponha que  $|O| \geq k$ ;
3.  $S_0, \dots, S_{\ell-1}$  particionam  $V$ ;
4. Algum  $i$  satisfaz  $|O \cap V(G_i)| \geq k$ ;
5.  $O \setminus S_i$  é independente em  $G_i$ ;
6. Então  $|X_\alpha| \geq |O \setminus S_i| = |O \cap V(G_i)| \geq k$ .

## Demonstração.

1. Seja  $O \subseteq V$  uma solução ótima;
2. Suponha que  $|O| \geq k$ ;
3.  $S_0, \dots, S_{\ell-1}$  particionam  $V$ ;
4. Algum  $i$  satisfaz  $|O \cap V(G_i)| \geq k$ ;
5.  $O \setminus S_i$  é independente em  $G_i$ ;
6. Então  $|X_\alpha| \geq |O \setminus S_i| = |O \cap V(G_i)| \geq k$ .

## Demonstração.

1. Seja  $O \subseteq V$  uma solução ótima;
2. Suponha que  $|O| \geq k$ ;
3.  $S_0, \dots, S_{\ell-1}$  particionam  $V$ ;
4. Algum  $i$  satisfaz  $|O \cap V(G_i)| \geq k$ ;
5.  $O \setminus S_i$  é independente em  $G_i$ ;
6. Então  $|X_\alpha| \geq |O \setminus S_i| = |O \cap V(G_i)| \geq k$ .

## Demonstração.

1. Seja  $O \subseteq V$  uma solução ótima;
2. Suponha que  $|O| \geq k$ ;
3.  $S_0, \dots, S_{\ell-1}$  particionam  $V$ ;
4. Algum  $i$  satisfaz  $|O \cap V(G_i)| \geq k$ ;
5.  $O \setminus S_i$  é independente em  $G_i$ ;
6. Então  $|X_\alpha| \geq |O \setminus S_i| = |O \cap V(G_i)| \geq k$ .



## Conclusão

## Outros Resultados

### Teorema ([CFM<sup>+</sup>15])

*Existem algoritmos FPT (parametrização natural) para os seguintes problemas em grafos planares:*

## Outros Resultados

### Teorema ([CFM<sup>+</sup>15])

*Existem algoritmos FPT (parametrização natural) para os seguintes problemas em grafos planares:*

- 1. Isomorfismo de Subgrafos;*

## Outros Resultados

### Teorema ([CFM<sup>+</sup>15])

*Existem algoritmos FPT (parametrização natural) para os seguintes problemas em grafos planares:*

- 1. Isomorfismo de Subgrafos;*
- 2. Bisseção Mínima.*

## Resultado Geral

---

A única propriedade dos grafos planares que utilizamos é serem fechados por menores e terem ***largura arbórea local*** limitada.

## Resultado Geral

---

A única propriedade dos grafos planares que utilizamos é serem fechados por menores e terem ***largura arbórea local*** limitada.

Uma classe de grafos  $\mathcal{G}$  tem ***largura arbórea local*** limitada se, para todo  $G \in \mathcal{G}$  e  $v \in V(G)$ , vale  $tw(G_v^r) \leq f(r)$ , onde  $G_v^r$  é o subgrafo induzido pelos vértices a distância até  $r$  de  $v$ .

## Limitações

---

A técnica de *Shifting/Baker* é eficaz para problemas “locais” ...

## Limitações

---

A técnica de *Shifting/Baker* é eficaz para problemas “locais” ...

... mas falha em casos como TSP ou Árvore de Steiner.

*Decomposição por contração* em grafos livres de  $H$ :

*Decomposição por contração* em grafos livres de  $H$ :

## Teorema ([DHK11])

*Para todo grafo fixo  $H$ , existe uma constante  $c_H$  tal que,*

*Decomposição por contração* em grafos livres de  $H$ :

## Teorema ([DHK11])

*Para todo grafo fixo  $H$ , existe uma constante  $c_H$  tal que,*

*Decomposição por contração* em grafos livres de  $H$ :

## Teorema ([DHK11])

*Para todo grafo fixo  $H$ , existe uma constante  $c_H$  tal que, para todo inteiro  $k \geq 1$ , qualquer grafo livre de  $H$ -menores pode ter suas arestas particionadas em  $k + 1$  classes de cor,*

*Decomposição por contração* em grafos livres de  $H$ :

## Teorema ([DHK11])

Para todo grafo fixo  $H$ , existe uma constante  $c_H$  tal que, para todo inteiro  $k \geq 1$ , qualquer grafo livre de  $H$ -menores pode ter suas arestas particionadas em  $k + 1$  classes de cor, de modo que a contração de qualquer uma dessas classes resulta em um grafo com largura arbórea no máximo  $c_H \cdot k$ .

*Decomposição por contração* em grafos livres de  $H$ :

## Teorema ([DHK11])

*Para todo grafo fixo  $H$ , existe uma constante  $c_H$  tal que, para todo inteiro  $k \geq 1$ , qualquer grafo livre de  $H$ -menores pode ter suas arestas particionadas em  $k + 1$  classes de cor, de modo que a contração de qualquer uma dessas classes resulta em um grafo com largura arbórea no máximo  $c_H \cdot k$ .*

*Além disso, tal partição pode ser encontrada em tempo polinomial.*

**Exercício.** Apresente, usando a técnica de *shifting*, um algoritmo FPT para o problema do Empacotamento Máximo de Triângulos (vértices disjuntos) em grafos planares.

**Exercício.** Apresente, usando a técnica de *shifting*, um algoritmo FPT para o problema do Empacotamento Máximo de Triângulos (vértices disjuntos) em grafos planares.

Obrigado a todos pela atenção...  
Alguma Dúvida?

# Bibliografia

- [CFM<sup>+</sup>15] Marek Cygan, Fedor V Fomin, Daniel Marx, Saket Saurabh, Lukasz Kowalik, Daniel Lokshtanov, and Marcin Pilipczuk.  
*Parameterized Algorithms*.  
Springer International Publishing, Cham, Switzerland, 1 edition, July 2015.
- [DHK11] Erik D. Demaine, Mohammad Taghi Hajiaghayi, and Ken-ichi Kawarabayashi.  
Contraction decomposition in  $H$ -minor-free graphs and algorithmic applications.  
In *Proceedings of the Forty-Third Annual ACM Symposium on Theory of Computing*, STOC '11, page 441–450, New York, NY, USA, 2011. Association for Computing Machinery.
- [GJ79] Michael R Garey and David S Johnson.  
*Computers and Intractability: A Guide to the Theory of NP-Completeness*.  
W.H. Freeman, New York, NY, April 1979.
- [WS11] David P. Williamson and David B. Shmoys.  
*The Design of Approximation Algorithms*.  
Cambridge University Press, 2011.