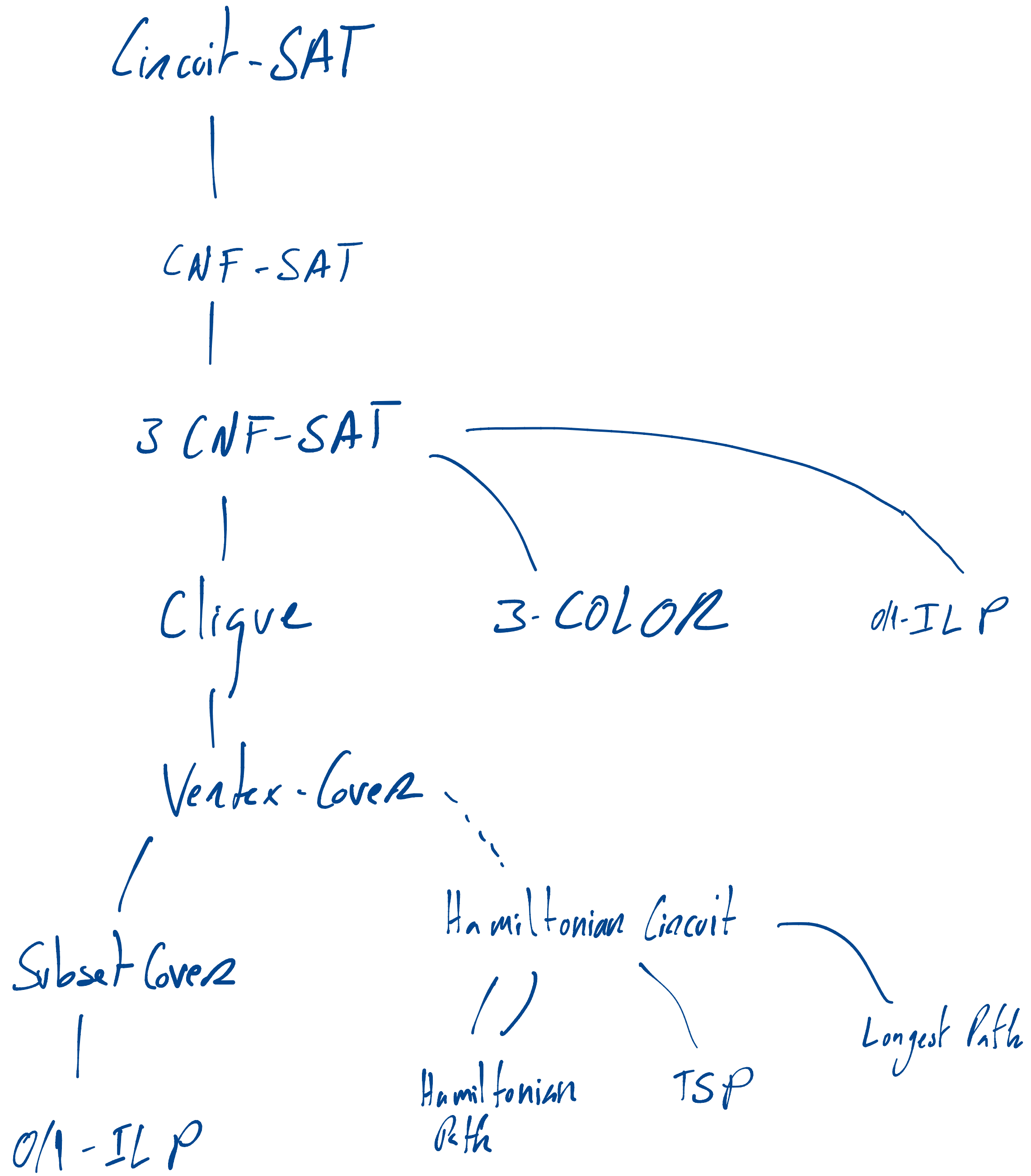
Aula 25

Road Map


Circuit-SAT
CNF-SAT
3 CNF-SAT
Clique
3-COLOR
0/1-ILP
Vertex-Cover
Subset Cover
Hamiltonian Circuit
0/1-ILP
Hamiltonian Path
TSP
Longest Path

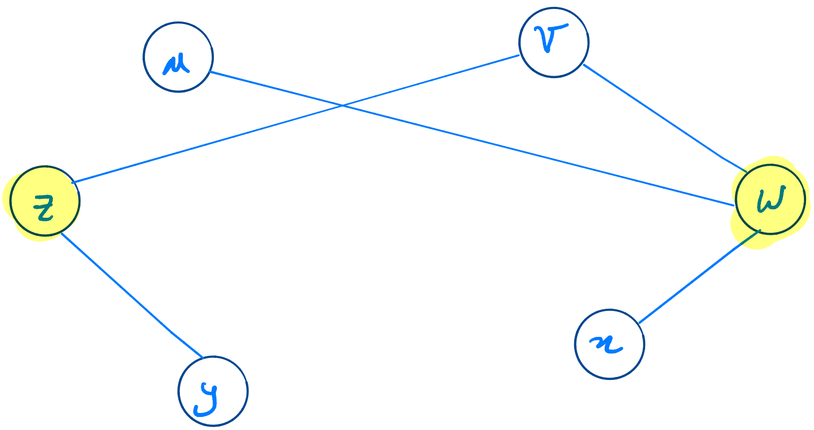
$\boxed{\text{Clique} \leq_P \text{Vertex-Cover}}$

Vertex-Cover: V' é uma cobertura de vértices do grafo não dirigido $G=(V,E)$ sse
$\forall (u,v) \in E.\ u \in V'$ ou $v \in V'$

Vertex-Cover Problem: Encontrar a cobertura de vértices de cardinalidade mínima

$\Downarrow$ Problema de Decisão

$$\text{VertexCover} = \{ \langle G, k \rangle \mid G \text{ é um grafo não dirigido com uma cobertura de vértices de tamanho } k \}$$



$\boxed{\text{Clique} \leq_p \text{Vertex-Cover}}$

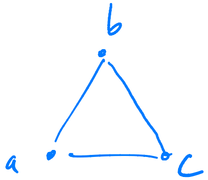
Proposição: $G=(V,E)$ tem um clique de tamanho k <u>sse</u> $\bar{G}=(V,\bar{E})$ tem uma cobertura de vértices de tamanho $|V|-k$.

Onde:

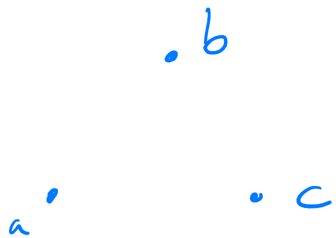
$$\bar{G}=(V,\bar{E}) \qquad \bar{E}=\{(u,v)\in V^2 \mid (u,v)\notin E \wedge u\neq v\}$$

Exemplo:

G :



$\bar{G}$:



Clique $=\{a,b,c\}$

$VC=\emptyset$

$\boxed{\text{Clique} \leq_p \text{Vertex-Cover}}$

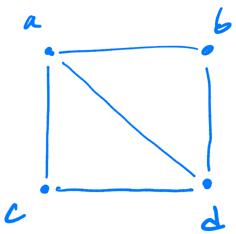
Proposição: $G=(V,E)$ tem um clique de tamanho k <u>sse</u> $\bar{G}=(V,\bar{E})$ tem uma cobertura de vértices de tamanho $|V|-k$.

Onde:

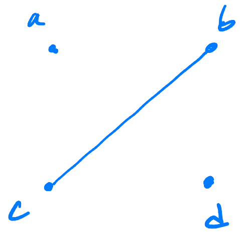
$$\bar{G}=(V,\bar{E}) \qquad \bar{E}=\{(u,v)\in V^2 \mid (u,v)\notin E \wedge u\neq v\}$$

Exemplo 2:

G:



$\bar{G}$:



Clique: $\{a,b,d\}$

$VC=\{c\}$

$\boxed{\text{Clique} \leq_p \text{Vertex-Cover}}$

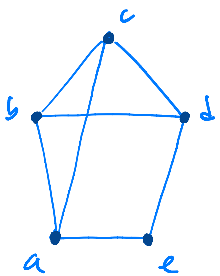
Proposição: $G=(V,E)$ tem um clique de tamanho k <u>sse</u> $\overline{G}=(V,\overline{E})$ tem uma cobertura de vértices de tamanho $|V|-k$.

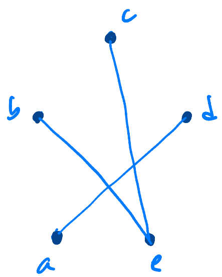
Onde:

$$\overline{G}=(V,\overline{E}) \qquad \overline{E}=\{(u,v)\in V^2 \mid (u,v)\notin E \wedge u\neq v\}$$

Exemplo 3

G:





Clique: $\{a,b,d\}$ $\qquad$ $VC=\{a,e\}$

Clique $\leq_p$ Vertex-Cover

Proposição: $G=(V,E)$ tem um Clique de tamanho k <u>sse</u> $\bar{G}=(V,\bar{E})$ tem uma cobertura de vértices de tamanho $|V|-k$.

Onde:

$\bar{G}=(V,\bar{E}) \quad \bar{E}=\{(u,v)\in V^2 \mid (u,v)\notin E \wedge u\neq v\}$

Basta provar que:

$\hat{V}$ é um Clique em G sse $V\setminus\hat{V}$ é um VC em $\bar{G}$

$\hat{V}$ é um Clique em G

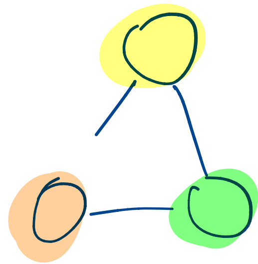
$\Leftrightarrow \forall u,v.\ u\in\hat{V} \wedge v\in\hat{V} \Rightarrow (u,v)\in E$

$\Leftrightarrow \forall u,v.\ (u,v)\notin E \Rightarrow u\notin\hat{V} \vee v\notin\hat{V}$

$\Leftrightarrow \forall u,v.\ (u,v)\in\bar{E} \Rightarrow u\in V\setminus\hat{V} \vee v\in V\setminus\hat{V}$

$\Leftrightarrow V\setminus\hat{V}$ é uma VC em $\bar{G}$

$$\boxed{\text{3-CNF-SAT} \leq_p \text{3-Color}}$$



3-Color:

- $G = (V, E)$, grafo não dirigido
- Coloração válida para G – atribuição de cores aos vértices de G tal que vértices adjacentes têm cores diferentes

- Problema 3-Color ⇒ Decidir se G pode ser colorido com 3 cores.

$$\text{3-Color} = \{ \langle G \rangle \mid G \text{ pode ser colorido c/ 3 cores} \}$$

Proposição: 3-Color $\in$ NP

Proposição: 3-Color é NP-difícil

$$\boxed{3\text{-CNF-SAT} \le_p 3\text{-Color}}$$

Proposição: 3-Color é NP-difícil

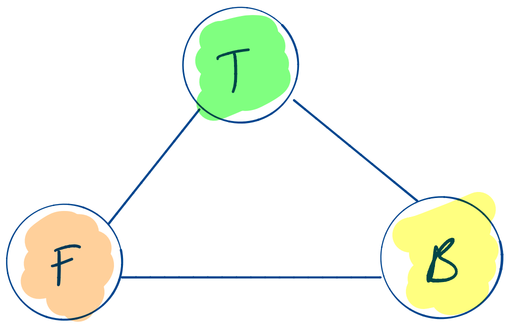
Prova: 3-CNF-SAT $\le_p$ 3-Color

$$\underbrace{\phi = C_1 \wedge C_2 \wedge \dots \wedge C_n}_{\Downarrow}$$

Instância do problema 3-Color

$$l ::= x \mid \neg x$$

$$C = l_1 \vee l_2 \vee l_3$$



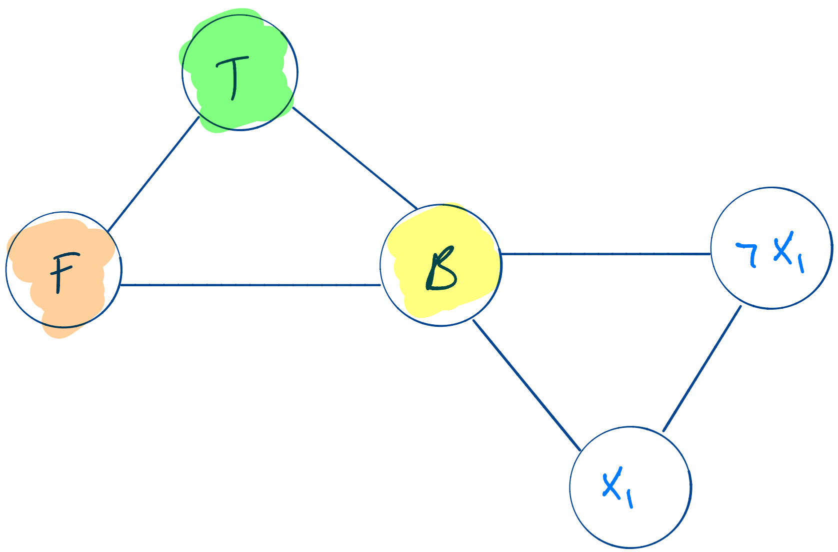
T - True
F - False
B - Base

3-CNF-SAT $\leq_p$ 3-Color

Proposição: 3-Color é NP-difícil

Prova: 3-CNF-SAT $\leq_p$ 3-Color

$\phi = C_1 \wedge C_2 \wedge \dots \wedge C_m$



- Por cada variável x adicionamos dois vértices: x e $\neg x$
  - ligamos os dois vértices entre si e ligamos os dois vértices à base

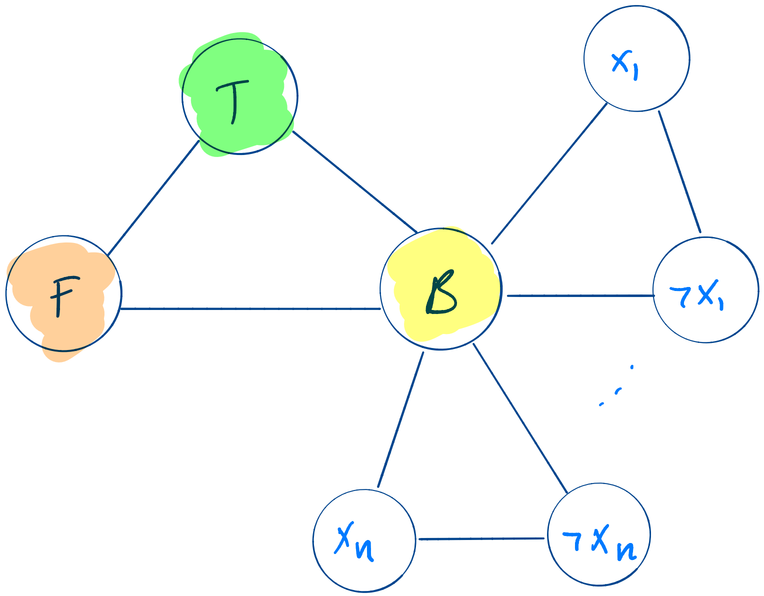
- x_1 tem de ser "verde" ou "vermelho", verdadeiro ou falso e $\neg x_1$ vai ter a cor oposta à cor de x_1

$\boxed{\text{3-CNF-SAT} \leq_p \text{3-Color}}$

Proposição: 3-Color é NP-difícil

Prova: 3-CNF-SAT $\leq_p$ 3-Color

$$\phi = C_1 \wedge C_2 \wedge \dots \wedge C_m$$



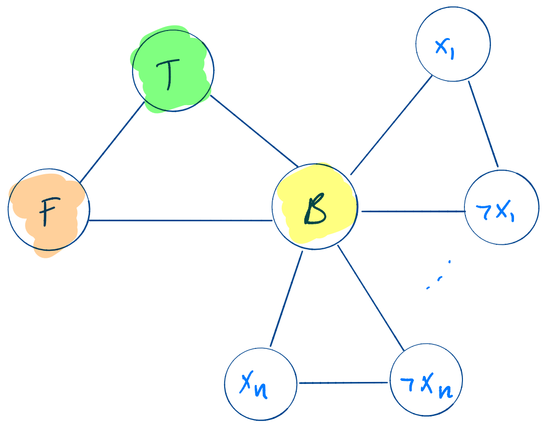
· O que é que nos falta?

$$\boxed{3\text{-CNF-SAT} \leq_p 3\text{-Color}}$$

Proposição: 3-Color é NP-difícil

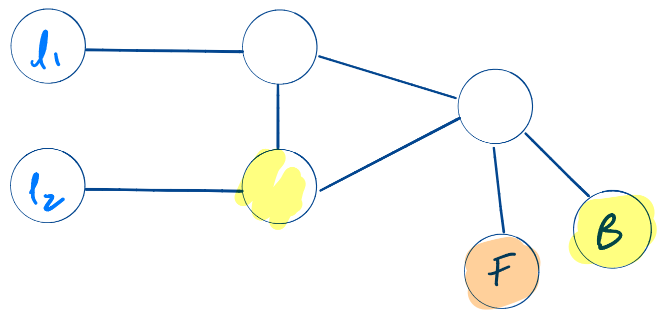
Prova: 3-CNF-SAT $\leq_p$ 3-Color

$$\phi = C_1 \wedge C_2 \wedge \dots \wedge C_m$$



* Clause Satisfiability Gadget

$l_1 \vee l_2$

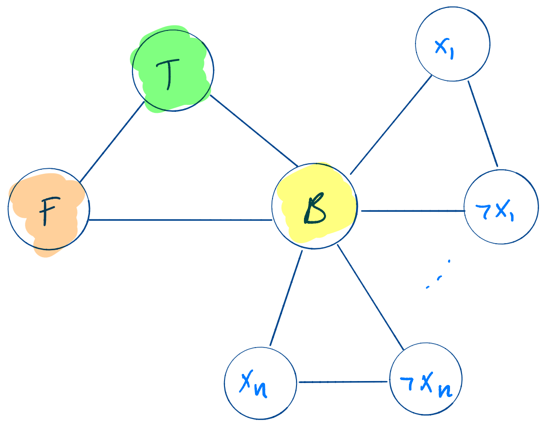


$$\boxed{\text{3-CNF-SAT} \leq_p \text{3-Color}}$$

Proposição: 3-Color é NP-difícil

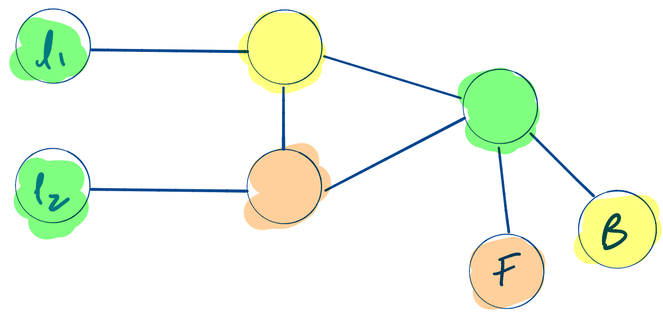
Prova: 3-CNF-SAT $\leq_p$ 3-Color

$\phi = C_1 \wedge C_2 \wedge \dots \wedge C_n$



* Clause Satisfiability Gadget

$l_1 \vee l_2$

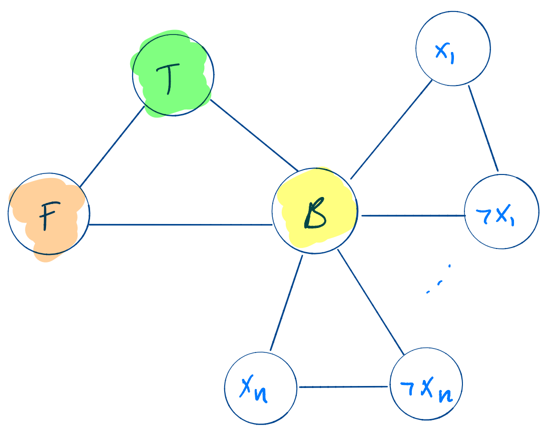


$$\boxed{\text{3-CNF-SAT} \le_p \text{3-Color}}$$

Proposição: 3-Color é NP-difícil

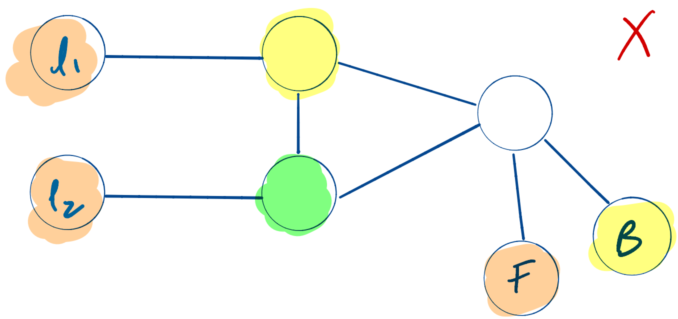
Prova: 3-CNF-SAT $\le_p$ 3-Color

$$\phi = C_1 \wedge C_2 \wedge \dots \wedge C_n$$



* Clause Satisfiability Gadget

$$l_1 \vee l_2$$



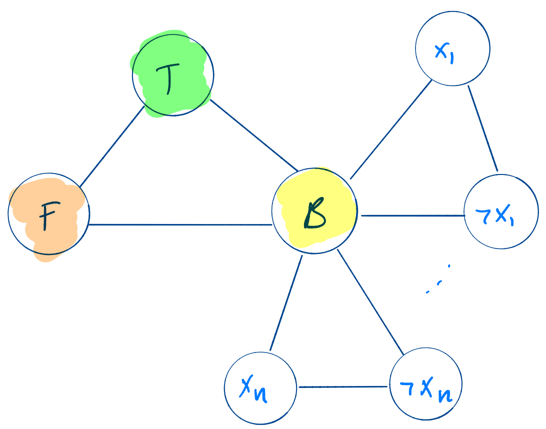
Proposição: ρ satisfaz a disjunção $l_1 \vee l_2$ sse existe uma extensão de ρ, ρ', que é uma coloração válida do 2Or-gadget.

$\boxed{\text{3-CNF-SAT} \leq_p \text{3-Color}}$

Proposição: 3-Color é NP-difícil

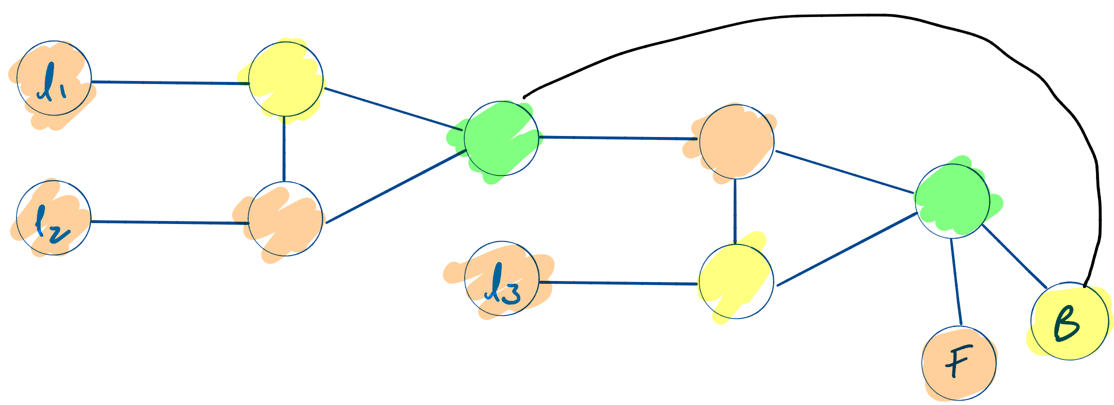
Prova: 3-CNF-SAT $\leq_p$ 3-Color

$\phi = C_1 \wedge C_2 \wedge \dots \wedge C_n$



* Clause Satisfiability Gadget

$l_1 \vee l_2 \vee l_3$



Proposição: ρ satisfaz a disjunção $l_1 \vee l_2 \vee l_3$ sse existe uma extensão de ρ, ρ', que é uma coloração válida do 3Or-gadget

$\boxed{\text{3-CNF-SAT} \leq_P \text{0/1-ILP}}$

- 0/1-ILP: Determinar o vector x tal que: $Ax = b$

  onde:

  - A: matriz $n \times m$ c/ valores inteiros
  - b: vector de inteiros de dimensão m
  - $x \in \{0, 1\}^n$

Proposição: 0/1-ILP $\in$ NP

Proposição: 0/1-ILP é NP-difícil

$\boxed{3\text{-CNF-SAT} \leq_p 0/1\text{-ILP}}$

- 0/1-ILP: Determinar o vector x tal que: $Ax = b$
  - A: matriz $n \times m$ c/ valores inteiros
  - b: vector de inteiros de dimensão m
  - $x \in \{0,1\}^n$

Proposição: 0/1-ILP é NP-difícil

Prova: Seja $\phi = C_1 \wedge \ldots \wedge C_n$ uma fórmula no formato 3-CNF

Ideia:

- Duas variáveis do programa 0/1-ILP por cada variável da fórmula: $x_j \rightsquigarrow x_j^1, x_j^0$
  $$x_j^1 \downarrow T \qquad x_j^0 \downarrow F$$

- Transformar cada cláusula C_i numa desigualdade
  $$C_i \rightsquigarrow a_i^T \cdot x \leq b_i$$

- Duas desigualdades por cada variável da fórmula original
  $$x_j \rightsquigarrow \left| \begin{array}{l} \hat{a}_j^T \cdot x \leq \hat{b}_j \\ \hat{\hat{a}}_j^T \cdot x \leq \hat{\hat{b}}_j \end{array} \right.$$

3-CNF-SAT $\leq_p$ 0/1-ILP

Ideia:

- Duas variáveis do programa 0/1-ILP por cada variável da fórmula: $x_j \leadsto x_j^1, x_j^0$ ($x_j^1 \to T$, $x_j^0 \to F$)

- Transformar cada cláusula C_i numa desigualdade
  $C_i \leadsto a_i^T . x \leq b_i$

$$Ax \leq b$$

Exemplo: $C_1 : x_0 \lor \neg x_2 \lor x_4$

$$x_0^1 + x_2^0 + x_4^1 \geq 1$$

$$\Leftrightarrow -x_0^1 - x_2^0 - x_4^1 \leq -1$$

$$a_1^T = \begin{array}{c} \\ \end{array}\begin{array}{cccccccccc} x_0^0 & x_0^1 & x_1^0 & x_1^1 & x_2^0 & x_2^1 & x_3^0 & x_3^1 & x_4^0 & x_4^1 \\ \end{array}$$
$$a_1^T = \begin{bmatrix} 0 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}$$

$b_1 = -1$

$$\boxed{\text{3-CNF-SAT} \le_P \text{0/1-ILP}}$$

Ideia :

- Duas variáveis do programa 0/1-ILP por cada variável da fórmula: $x_j \rightsquigarrow \underset{T}{x_j^1}, \underset{F}{x_j^0}$

- Transformar cada cláusula C_i numa desigualdade

$$C_i \rightsquigarrow a_i^T \cdot x \le b_i$$

Exemplo: $C_1 : x_0 \vee \neg x_2 \vee x_4$

$$\Downarrow$$

$$x_0^1 + x_2^0 + x_4^1 \ge 1$$

$$\Downarrow$$

$$-x_0^1 - x_2^0 - x_4^1 \le -1$$

$$a_1^T = \begin{array}{cccccccccc} x_0^0 & x_0^1 & x_1^0 & x_1^1 & x_2^0 & x_2^1 & x_3^0 & x_3^1 & x_4^0 & x_4^1 \\ \end{array}$$
$$\qquad \begin{bmatrix} 0 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}$$

$$b_1 = -1$$

$$\boxed{\text{3-CNF-SAT} \le_p \text{0/1-ILP}}$$

Ideia:

- Duas variáveis do programa 0/1-ILP por cada variável da fórmula: $x_j \rightsquigarrow \underset{\substack{\downarrow \\ T}}{x_j^1}, \underset{\substack{\downarrow \\ F}}{x_j^0}$

- Duas desigualdades por cada variável da fórmula original

$$x_j \rightsquigarrow \left| \begin{array}{l} \hat{a}_j^T \cdot x \le \hat{b}_j \\ \hat{\hat{a}}_j^T \cdot x \le \hat{\hat{b}}_j \end{array} \right.$$

Exemplo: x_1 $\qquad x_1^1 \quad x_1^0 \qquad x_1^1 + x_1^0 = 1 \iff \begin{cases} x_1^1 + x_1^0 \le 1 \\ x_1^1 + x_1^0 \ge 1 \end{cases} \iff \begin{cases} x_1^1 + x_1^0 \le 1 \\ -x_1^1 - x_1^0 \le -1 \end{cases}$

$$\begin{array}{ll} & \begin{array}{cccccccccc} x_0^0 & x_0^1 & x_1^0 & x_1^1 & x_2^0 & x_2^1 & x_3^0 & x_3^1 & x_4^0 & x_4^1 \end{array} \\ \hat{a}_1^T = & \left[\begin{array}{cccccccccc} 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{array} \right] \qquad \hat{b}_1 = 1 \\ \hat{\hat{a}}_1^T = & \left[\begin{array}{cccccccccc} 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \end{array} \right] \qquad \hat{\hat{b}}_1 = -1 \end{array}$$

$$\boxed{\text{3-CNF-SAT} \leq_p \text{0/1-ILP}}$$

Ideia:

- Duas variáveis do programa 0/1-ILP por cada variável da fórmula: $x_j \rightsquigarrow \underset{\substack{\downarrow \\ T}}{x_j^1},\ \underset{\substack{\downarrow \\ F}}{x_j^0}$

- Duas desigualdades por cada variável da fórmula original

$$x_j \rightsquigarrow \left| \begin{array}{l} \hat{a}_j^T \cdot x \leq \hat{b}_j \\ \hat{\hat{a}}_j^T \cdot x \leq \hat{\hat{b}}_j \end{array} \right.$$

Exemplo: x_1

$$x_1^1 + x_1^0 = 1 \iff \begin{array}{l} x_1^1 + x_1^0 \leq 1 \\ x_1^1 + x_1^0 \geq 1 \end{array} \iff \left\{ \begin{array}{l} x_1^1 + x_1^0 \leq 1 \\ -x_1^1 - x_1^0 \leq -1 \end{array} \right.$$

$$\left| \begin{array}{ll} \hat{a}_1^T = \begin{array}{c} \begin{array}{cccccccccc} x_0^0 & x_0^1 & x_1^0 & x_1^1 & x_2^0 & x_2^1 & x_3^0 & x_3^1 & x_4^0 & x_4^1 \end{array} \\ \begin{bmatrix} 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array} & \hat{b}_1 = 1 \\ \hat{\hat{a}}_1^T = \begin{bmatrix} 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} & \hat{\hat{b}}_1 = -1 \end{array} \right.$$

Vertex-Cover $\leq_p$ Subset-Cover

- Subset Cover:
  - F é uma família de conjuntos de elementos de um dado conjunto base Ω

    $F = \{X_1, \dots, X_n\} \quad \forall_{1 \leq i \leq n}.\ X_i \subseteq \Omega$

  - $Y \subseteq \Omega$

  - $\text{SubsetCover} = \{ \langle F, Y, k \rangle \mid \exists\, X_1, \dots, X_k \in F.\ X_1 \cup X_2 \cup \dots \cup X_k = Y \}$

Proposição: Subset Cover está em NP

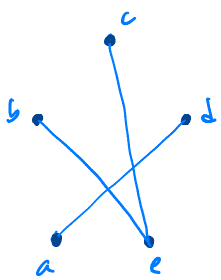
Proposição: Subset Cover é NP-difícil

Vertex-Cover $\leq_p$ Subset-Cover

- Subset Cover:

  - $\text{SubsetCover} = \{ \langle F, Y, k \rangle \mid \exists\, X_1, \ldots, X_k \in F.\ X_1 \cup X_2 \cup \cdots \cup X_k = Y \}$

Proposição: Subset Cover é NP-difícil



$X_a = \{(a,d)\}$ $\qquad$ $X_d = \{(a,d)\}$

$X_b = \{(b,e)\}$ $\qquad$ $X_e = \{(b,e), (c,e)\}$

$X_c = \{(c,e)\}$

$VC = \{a, e\}$

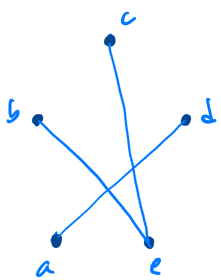
$Y = \{(b,e), (c,e), (a,d)\}$

Vertex-Cover $\leq_p$ Subset-Cover

- Subset Cover:

  - $\text{SubsetCover} = \{ \langle F, Y, k \rangle \mid \exists\, X_1, \dots, X_k \in F.\ X_1 \cup X_2 \cup \dots \cup X_k = Y \}$

Proposição: Subset Cover é NP-difícil



$X_a = \{(a,d)\}$ $\quad$ $X_d = \{(a,d)\}$

$X_b = \{(b,e)\}$ $\quad$ $X_e = \{(b,e), (c,e)\}$

$X_c = \{(c,e)\}$

$VC = \{a, e\}$

$Y = \{(b,e), (c,e), (a,d)\}$

Vertex-Cover $\leq_p$ Subset-Cover

- Subset Cover:

  - $\text{Subset Cover} = \{ \langle F, Y, k \rangle \mid \exists\, X_1, \ldots, X_k \in F.\ X_1 \cup X_2 \cup \cdots \cup X_k = Y \}$

Proposição: Subset Cover é NP-difícil

$$\langle G, k \rangle \in VC \iff \langle F_G, E, k \rangle \in \text{Subset Cover}$$

onde
- $G = (V, E)$
- $F_G = \{ E_v \mid v \in V \}$
- $E_v = \{ (u, v) \mid (u, v) \in E \} \cup \{ (v, u) \mid (v, u) \in E \}$

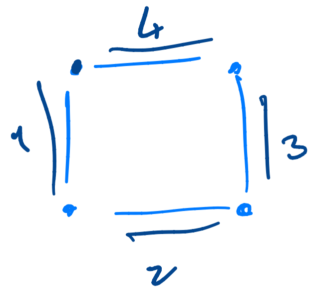
- O resultado segue directamente, observando que:

$$\hat{V} \text{ é uma VC de } G \text{ sse } \bigcup_{u \in \hat{V}} E_u = E$$

Ciclo Hamiltoniano

Ciclo Hamiltoniano: Dado um grafo não-dirigido $G=(V,E)$, um ciclo Hamiltoniano de G é um ciclo simples que contém todos os vértices de V.

$$\text{Ham-Cycle} = \{\langle G\rangle \mid G \text{ é um grafo não dirigido com um ciclo hamiltoniano}\}$$



Proposição: Ham-Cycle $\in NP$

Proposição: Ham-Cycle é NP-difícil.

↳ Prova:

Redução: $\underbrace{\text{Vertex-Cover} \le_p \text{Ham-Cycle}}_{34.5.3\ \text{(CLRS)}}$

Caminho Hamiltoniano

Ciclo Hamiltoniano: Dado um grafo não-dirigido $G=(V,E)$, um caminho Hamiltoniano de G é um caminho que contém todos os vertices de V exactamente uma vez.

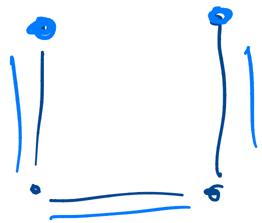
$$\text{Ham-Path} = \{ \langle G \rangle \mid G \text{ é um grafo não dirigido com um caminho hamiltoniano} \}$$

Proposição: Ham-Path $\in NP$

Proposição: Ham-Path é NP-difícil.

↳ Prova:

Redução: $\text{Ham-Cycle} \leq_p \text{Ham-Path}$

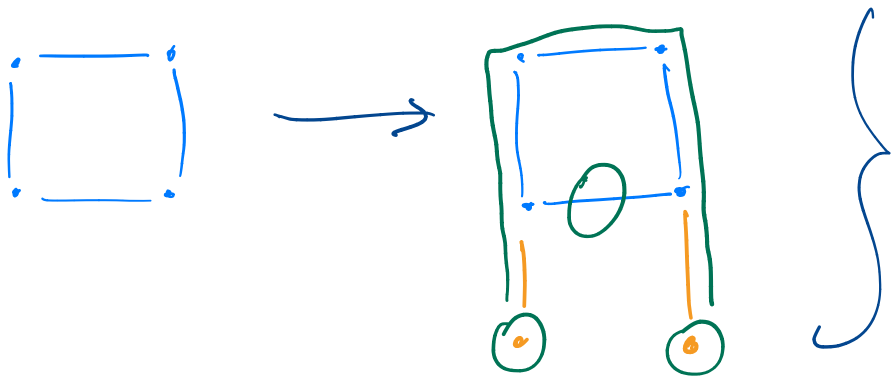


Caminho Hamiltoniano

Proposição: Ham-Path é NP-difícil.

↳ **Prova:**

Redução: Ham-Cycle $\leq_p$ Ham-Path



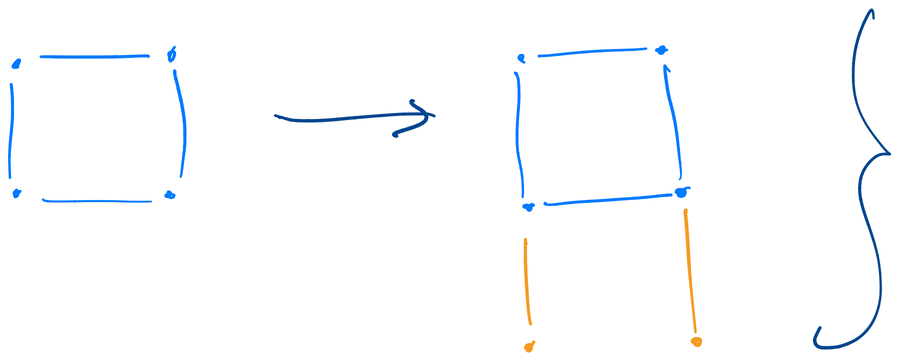
Ideia

Caminho Hamiltoniano

Proposição: Ham-Path é NP-difícil.

↳ Prova:

Redução: Ham-Cycle $\leq_p$ Ham-Path



Ideia

Proposição:

$G \in$ Ham-Cycle sse $G' \in$ Ham-Path

onde:

$G = (V, E) \wedge (u, v) \in E$

$G' = (V \cup \{u', v'\}, E \cup \{(u', u), (v', v)\})$

Observação: É a redução: Ham-Path $\leq_p$ Ham-Cycle ?

(TPC)

Coping with NP-Completeness

1. Branch-and-Bound

2. Procura Heurística

3. Algoritmos de Aproximação

4. Algoritmos Probabilísticos

Coping with NP-Completeness

1. Branch-and-Bound
2. Procura Heurística
3. Algoritmos de Aproximação
4. Algoritmos Probabilísticos

(1)–(2): IA, Procura & Planeamento (Mestrado)

(3)–(4): Algoritmos Avançados (Mestrado)

Investigação @ Técnico: Grupo SAT (INESC-ID)
→ Agora: ARSR

Coping with NP-Completeness

1. Branch-and-Bound
2. Procura Heurística

} IA, Procura & Planeamento (Mestrado)

3. Algoritmos de Aproximação
4. Algoritmos Probabilísticos

} Algoritmos Avançados (Mestrado)

Investigação @ Técnico : Grupo SAT (INESC-ID)

↳ Agora: ARSR

Algoritmos de Aproximação

Exemplo: Vertex - Cover

Approx-Vertex-Cover (G)

$C = \emptyset$

$E' = G.E$

while $E' \neq \emptyset$

Pick (u, v) in E'

if $u \notin C \lor v \notin C$

$C = C \cup \{u, v\}$

Remove from E' all edges incident on either u or v

return C

Algoritmos de Aproximação

Exemplo: Vertex - Cover

```
Approx-Vertex-Cover (G)
    C = ∅
    E' = G.E
    while E' ≠ ∅
        let (u,v) be an arbitrary edge of E'
        C = C ∪ {u,v}
        remove from E' every edge incident on either u or v
    return C
```

Proposição: $V' = $ Approx-Vertex-Cover (G)

$V^* \rightarrow$ cobertura de vértices de tamanho mínimo

$$\frac{|V'|}{|V^*|} \leq 2 \rightarrow \text{razão de aproximação}$$

$$|V'| \leq 2 \times |V^*|$$