Análise e Síntese de Algoritmos

Complexidade Computacional (Reduções)

Prof. Pedro T. Monteiro

IST - Universidade de Lisboa

2024/2025

Contexto

- Revisão [CLRS, Cap.1-13]
  - Fundamentos; notação; exemplos
- Técnicas de Síntese de Algoritmos [CLRS, Cap.15-16]
  - Programação dinâmica [CLRS, Cap.15]
  - Algoritmos greedy [CLRS, Cap.16]
- Algoritmos em Grafos [CLRS, Cap.21-26]
  - Algoritmos elementares
  - Caminhos mais curtos [CLRS, Cap.22,24-25]
  - Árvores abrangentes [CLRS, Cap.23]
  - Fluxos máximos [CLRS, Cap.26]
- Programação Linear [CLRS, Cap.29]
  - Algoritmos e modelação de problemas com restrições lineares
- Tópicos Adicionais
  - Complexidade Computacional [CLRS, Cap.34]

Resumo

CNFSAT $\leq_p$ 3CNFSAT

CNFSAT $\leq_p$ CLIQUE

3CNFSAT $\leq_p$ ISET

ISET $\leq_p$ CLIQUE

ISET $\leq_p$ VCOVER

VCOVER $\leq_p$ SETCOVER

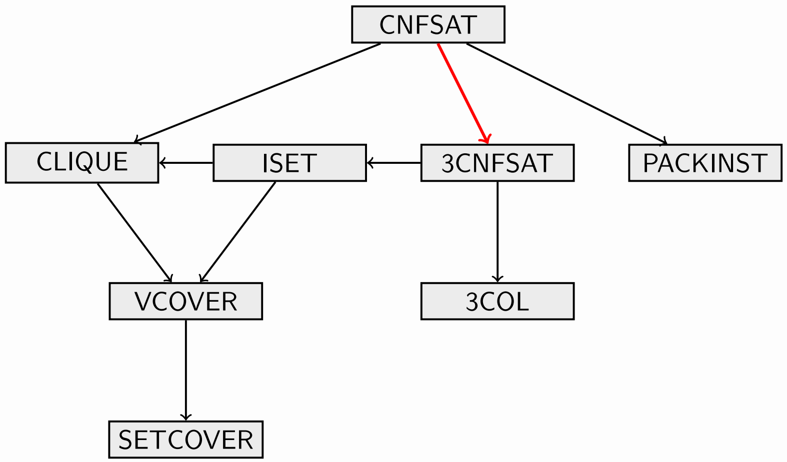
3CNFSAT $\leq_p$ 3COL

Reduções

Provar que ProblemaX é NP-completo

- **Passo 1:** Provar que ProblemaX $\in$ NP
- **Passo 2:** Provar que ProblemaX $\in$ NP-difícil
  ProblemaY $\leq_p$ ProblemaX
  (por redução a partir de um ProblemaY já conhecido como sendo NP-difícil)

Reduções



CNFSAT $\leq_p$ 3CNFSAT

Problema 3CNFSAT
Dado um conjunto de n variáveis booleanas $x_1, x_2, \ldots, x_n$ e uma fórmula φ na forma normal conjuntiva (CNF) com exactamente 3 literais em cada uma das m cláusulas, existe uma atribuição de valores para as variáveis de φ, tal que φ é verdadeira?

Vemos que é um problema combinatório dado que:

- teríamos de testar todas as 2^n combinações de variáveis

Teorema: 3CNFSAT é NP-completo

CNFSAT $\leq_p$ 3CNFSAT

Provar que 3CNFSAT $\in$ NP

1. Identificar:
   - Instância do problema 3CNFSAT: $\langle \varphi \rangle$
   - Certificado: $C = \{(x_1, v_1), (x_2, v_2), \ldots, (x_n, v_n)\}$ (atribuição de valores às n variáveis)
2. Dada uma instância $\langle \varphi \rangle$ e certificado C, o algoritmo de verificação tem de:
   - Verificar que o certificado tem tamanho polinomial
     $|C| = n$, logo $O(n)$
   - Verificar o certificado na instância em tempo polinomial
     Para cada uma das m cláusulas de φ verificar que:
     pelo menos um dos 3 literais é verdadeiro em C, logo $O(|\varphi|) = O(m)$

$\therefore$ 3CNFSAT $\in$ NP
(dado que o certificado tem tamanho polinomial e pode ser verificado em tempo polinomial)

CNFSAT $\leq_p$ 3CNFSAT

Provar que 3CNFSAT $\in$ NP-difícil: CNFSAT $\leq_p$ 3CNFSAT
Dada uma instância de CNFSAT $\langle \varphi \rangle$, derivar instância de 3CNFSAT $\langle \varphi_3 \rangle$

A redução é definida como $f(\varphi) = \langle \varphi_3 \rangle$:

- Por cada cláusula unitária $w = (l_1)$:
  - Criar cláusulas: $(l_1 \vee y_1 \vee y_2) \wedge (l_1 \vee \neg y_1 \vee y_2) \wedge (l_1 \vee y_1 \vee \neg y_2) \wedge (l_1 \vee \neg y_1 \vee \neg y_2)$, com variáveis adicionais y_1 e y_2
- Por cada cláusula binária $w = (l_1 \vee l_2)$:
  - Criar cláusulas: $(l_1 \vee l_2 \vee y_1) \wedge (l_1 \vee l_2 \vee \neg y_1)$, com variável adicional y_1
- Por cada cláusula com mais que 3 literais $w = (l_1 \vee l_2 \vee l_3 \vee \ldots \vee l_k)$:
  - Criar cláusulas: $(l_1 \vee l_2 \vee y_1) \wedge (\neg y_1 \vee l_3 \vee y_2) \wedge \ldots \wedge (\neg y_{k-4} \vee l_{k-2} \vee y_{k-3})$ $\wedge(\neg y_{k-3} \vee l_{k-1} \vee l_k)$, com variáveis adicionais $y_1, y_2, \ldots, y_{k-3}$

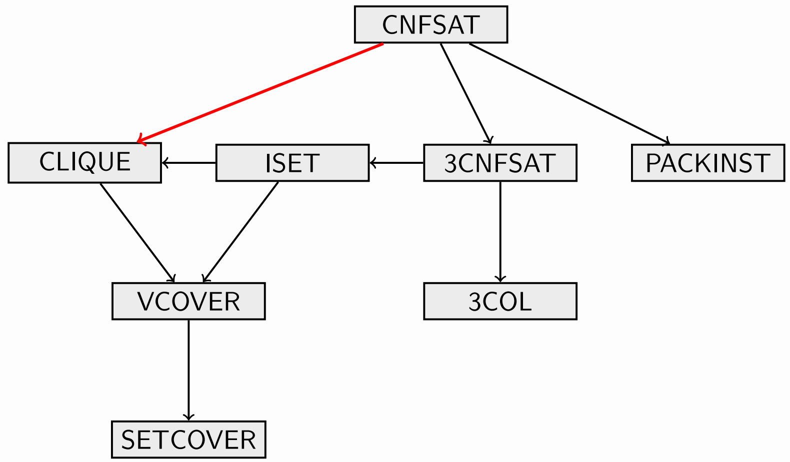
Complexidade do algoritmo de redução: $O(|\varphi|)$ (# variáveis/cláusulas adicionais é $O(|\varphi|)$)

Provar que 3CNFSAT $\in$ NP-difícil: CNFSAT $\leq_p$ 3CNFSAT

$\langle\varphi\rangle$ é satisfazível $\iff$ $\langle\varphi_3\rangle$ é satisfazível

- Cláusulas unitárias e binárias: prova é simples
- Cláusulas com mais que 3 literais:
  - Se φ é satisfazível, então φ_3 é satisfazível
    - Para cada cláusula w, existe $l_r = 1, 1 \leq r \leq k$
    - Atribuir valor 1 às variáveis y_s, $1 \leq s \leq \min(r-1, k-3)$
    - Atribuir valor 0 às variáveis y_t, $\min(r-1, k-3) + 1 \leq t \leq k-3$
    - Todas as cláusulas satisfeitas, pelo que φ_3 é satisfazível
  - Se φ_3 é satisfazível, então φ é satisfazível
    - Um dos literais de $(l_1 \vee l_2 \vee l_3 \vee \ldots \vee l_k)$ tem de ter valor 1
    - Caso contrário $(y_1) \wedge (\neg y_1 \vee y_2) \wedge \ldots - \wedge(\neg y_{k-4} \vee y_{k-3}) \wedge (\neg y_{k-3})$ teria que ser satisfeita
    - Impossível. Logo, φ é satisfazível

$\therefore$ 3CNFSAT $\in$ NP-difícil



Problema CLIQUE

Dado um grafo não-dirigido $G = (V, E)$ e um inteiro k,
existe um subconjunto de vértices $V' \subseteq V$ de tamanho $|V'| = k$,
tal que todos os vértices em V' formam uma clique?

Vemos que é um problema combinatório dado que:

- teríamos de testar todas as $C_k^{|V|}$ combinações de vértices

Provar que CLIQUE $\in$ NP

1. Identificar:
   - Instância do problema CLIQUE: $\langle G = (V, E), k\rangle$
   - Certificado: subconjunto de vértices $V' \subseteq V$, $|V'| = k$
2. Dada uma instância $\langle G = (V, E), k\rangle$ e um certificado V', descrever um algoritmo de verificação:
   - Verificar que o certificado tem tamanho polinomial
     $|V'| = k \leq |V|$, logo $O(|V|)$
   - Verificar o certificado na instância em tempo polinomial
     Para todo $u, v \in V'$, verificar se $(u, v) \in E$, logo $O(|V'|^2)$

$\therefore$ CLIQUE $\in$ NP
(dado que o certificado tem tamanho polinomial e pode ser verificado em tempo polinomial)

CNFSAT $\leq_p$ CLIQUE

Provar que CLIQUE $\in$ NP-difícil: *CNFSAT $\leq_p$ CLIQUE*
Dada uma instância de CNFSAT $\langle \varphi \rangle$, derivar instância de CLIQUE $\langle G = (V, E), k \rangle$

A redução é definida como $f(\langle \varphi \rangle = \omega_1 \wedge \ldots \wedge \omega_m) = \langle G = (V, E), k \rangle$:

- Criar coluna por cada cláusula ω_i, com vértices correspondentes aos literais de ω_i $(|V| = O(n.m))$
- Criar arco (v_a, v_b) entre todos os pares de vértices, excepto se:
  - v_a e v_b pertencem à mesma coluna w_i
  - v_a e v_b correspondem à variável x e a sua negação $\neg x$

  (cada vértice pode ter $O(n.m)$ arcos)

Complexidade do algoritmo de redução: $O(n.m \times n.m)$

CNFSAT $\leq_p$ CLIQUE

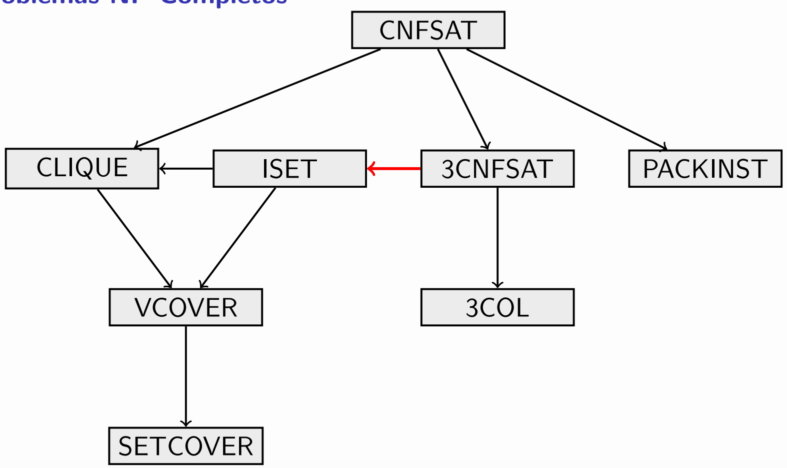
Provar que CLIQUE $\in$ NP-difícil: CNFSAT $\leq_p$ CLIQUE
$\langle \varphi = \omega_1 \wedge \ldots \wedge \omega_m \rangle$ é satisfazível $\iff$ $\langle G = (V, E), k \rangle$ é satisfazível

- G tem sub-grafo completo de dimensão $k = m$ se e só se fórmula φ é satisfeita
- Se φ é satisfeita:
  - Seleccionar vértice, de cada coluna i, cujo literal respectivo tem atribuído o valor 1
  - Para este vértice u existe arco para vértice v em qualquer outra coluna, tal que v associado com literal com valor 1 atribuído
  - Logo, existe sub-grafo completo com dimensão $k = m$
- Se G tem sub-grafo completo de dimensão $k = m$:
  - Seleccionar um vértice de cada coluna
  - Atribuir valor 1 a cada vértice seleccionado (tal que x e $\neg x$ não seleccionados simultaneamente)
  - Cada cláusula tem 1 literal atribuído valor 1. Logo, satisfaz φ

$\therefore$ CLIQUE $\in$ NP-difícil

Reduções

Provar Problemas NP-Completos



3CNFSAT $\leq_p$ ISET

Problema ISET (Independent Set)
Dado um grafo não-dirigido $G = (V, E)$ e um inteiro k,
existe um subconjunto de vértices $I \subseteq V$ de tamanho $|I| = k$,
tal que todos os vértices em I são independentes?

Vemos que é um problema combinatório dado que:

- teríamos de testar todas as $C_k^{|V|}$ combinações de vértices

3CNFSAT $\leq_p$ ISET

Provar que ISET $\in$ NP

1. Identificar:
   - Instância do problema ISET: $\langle G = (V, E), k \rangle$
   - Certificado: $I \subseteq V$, $|I| = k$
2. Dada uma instância $\langle G, k \rangle$ e certificado I, descrever um algoritmo de verificação:
   - Verificar que o certificado tem tamanho polinomial
     $|I| = k \leq |V|$, logo $O(|V|)$
   - Verificar o certificado na instância em tempo polinomial
     Para todo $u, v \in I$, verificar se $(u, v) \notin E$, logo $O(|I|^2)$

$\therefore$ ISET $\in$ NP
(dado que o certificado tem tamanho polinomial e pode ser verificado em tempo polinomial)

3CNFSAT $\leq_p$ ISET

Provar que ISET $\in$ NP-difícil: 3CNFSAT $\leq_p$ ISET
Dada uma instância de 3CNFSAT $\langle \varphi \rangle$, construir uma instância de ISET $\langle G, k \rangle$:

- Criar um vértice para cada literal de φ
  ($|V| = 3 \times k$, onde $k = |\varphi|$ é o # cláusulas)
- Criar arestas entre literais que aparecem na mesma cláusula
  (impede soluções com literais da mesma cláusula)
- Criar arestas entre cada literal e sua negação
  (impede soluções com literais e suas negações)

Complexidade do algoritmo de redução: $O(|\varphi|)$

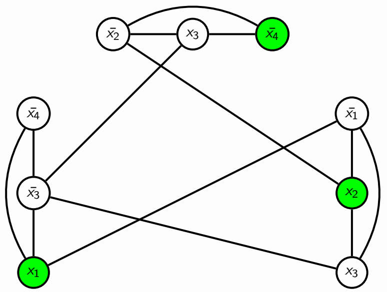
A redução é definida como $f(\varphi) = \langle G = (V, E), |\varphi| \rangle$
$\langle \varphi \rangle$ é satisfazível $\iff$ $\langle G, k \rangle$ possui um ISET de tamanho $k = |\varphi|$

$\therefore$ ISET $\in$ NP-difícil

3CNFSAT $\leq_p$ ISET

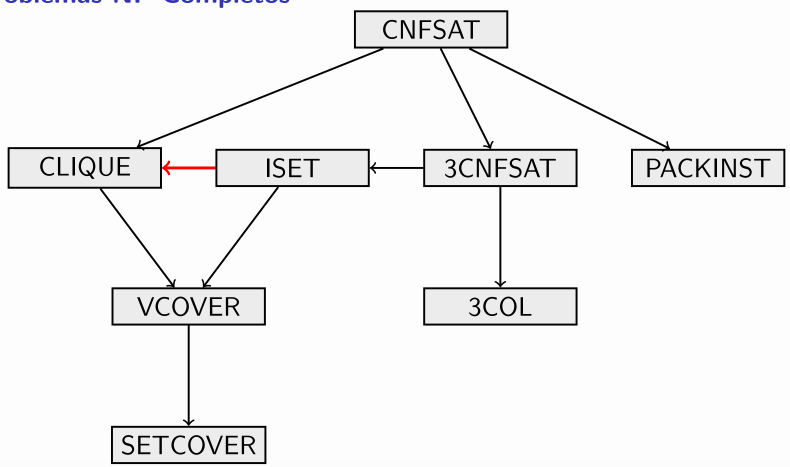
Exemplo de redução:
Instância de 3CNFSAT:
$\varphi = (x_1 \vee \bar{x_3} \vee \bar{x_4}) \wedge (\bar{x_2} \vee x_3 \vee \bar{x_4}) \wedge (\bar{x_1} \vee x_2 \vee x_3)$

Instância de ISET:



Reduções

Provar Problemas NP-Completos



ISET $\leq_p$ CLIQUE

Problema CLIQUE
Dado um grafo não-dirigido $G = (V, E)$ e um inteiro k, existe um subconjunto de vértices $I \subseteq V$ de tamanho $|I| = k$, tal que todos os vértices em I formam uma clique?

Vemos que é um problema combinatório dado que:

- teríamos de testar todas as $C_k^{|V|}$ combinações de vértices

ISET $\leq_p$ CLIQUE

Provar que CLIQUE $\in$ NP

1. Identificar:
   - Instância do problema CLIQUE: $\langle G = (V, E), k\rangle$
   - Certificado: subconjunto de vértices $I \subseteq V$, $|I| = k$
2. Dada uma instância $\langle G = (V, E), k\rangle$ e um certificado I, descrever um algoritmo de verificaçao:
   - Verificar que o certificado tem tamanho polinomial
     $|I| = k \leq |V|$, logo $O(|V|)$
   - Verificar o certificado na instância em tempo polinomial
     Para todo $u, v \in I$, verificar se $(u, v) \in E$, logo $O(|I|^2)$

$\therefore$ CLIQUE $\in$ NP
(dado que o certificado tem tamanho polinomial e pode ser verificado em tempo polinomial)

ISET $\leq_p$ CLIQUE

Provar que CLIQUE $\in$ NP-difícil: *ISET $\leq_p$ CLIQUE*
Dada uma instância de ISET $\langle G = (V, E), k\rangle$, construir uma instância de CLIQUE $\langle G' = (V', E'), k\rangle$:

- Criar um vértice v' para cada $v \in V$
  ($|V'| = |V|$)
- Para todo $u, v \in V$, adicionar uma aresta $(u', v') \in E'$, se $(u, v) \notin E$
  (G' complementa as arestas de G)

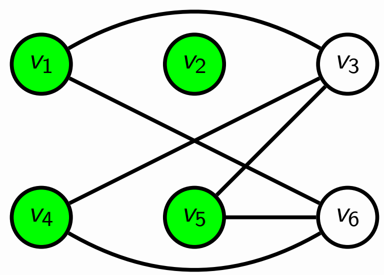
Complexidade do algoritmo de redução: $O(|V|^2)$

A redução é definida como $f(\langle G = (V, E), k\rangle) = \langle G' = (V', E'), k\rangle$
$\langle G = (V, E), k\rangle \in$ ISET $\iff \langle G' = (V', E'), k\rangle \in$ CLIQUE

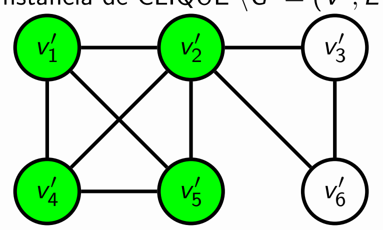
$\therefore$ CLIQUE $\in$ NP-difícil

ISET $\leq_p$ CLIQUE

Exemplo de redução:
Instância de ISET $\langle G = (V, E), k\rangle$:

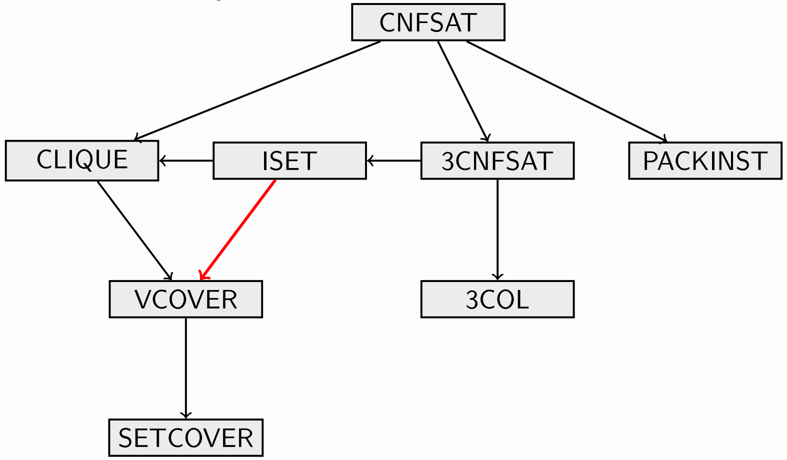


Instância de CLIQUE $\langle G' = (V', E'), k\rangle$:



Reduções

Provar Problemas NP-Completos



ISET $\leq_p$ VCOVER

Problema VCOVER (Vertex Cover)
Dado um grafo não-dirigido $G = (V, E)$ e um inteiro k,
existe um subconjunto de vértices $C \subseteq V$ de tamanho $|C| = k$,
tal que qualquer arco em E é incidente em pelo menos um dos vértices de C?

Vemos que é um problema combinatório dado que:

- teríamos de testar todas as $C_k^{|V|}$ combinações de vértices

ISET $\leq_p$ VCOVER

Provar que VCOVER $\in$ NP

1. Identificar:
   - Instância do problema VCOVER: $\langle G = (V, E), k\rangle$
   - Certificado: subconjunto de vértices $C \subseteq V$, $|C| = k$
2. Dada uma instância $\langle G = (V, E), k\rangle$ e um certificado C, descrever um algoritmo de verificação:
   - Verificar que o certificado tem tamanho polinomial
     $|C| = k \leq |V|$, logo $O(V)$
   - Verificar o certificado na instância em tempo polinomial
     Para cada (u,v) $\in$ E, verificar se u $\in$ C ou v $\in$ C, logo $O(E.V)$

$\therefore$ VCOVER $\in$ NP
(dado que o certificado tem tamanho polinomial e pode ser verificado em tempo polinomial)

ISET $\leq_p$ VCOVER

Provar que VCOVER $\in$ NP-difícil: ISET $\leq_p$ VCOVER
Dada uma instância de ISET $\langle G = (V, E), k\rangle$, construir uma instância de VCOVER $\langle G' = (V', E'), k'\rangle$:

- Mantemos o grafo $G' = G$
- Mas $k' = |V| - k$
  (dado um certificado I, $C = V - I$ é um certificado para VCOVER)

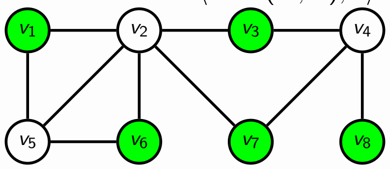
Complexidade do algoritmo de redução: O(V)

A redução é definida como f($\langle G = (V, E), k\rangle$) = $\langle G = (V, E), |V| - k\rangle$
$\langle G = (V, E), k\rangle \in$ ISET $\iff$ $\langle G = (V, E), |V| - k\rangle \in$ VCOVER

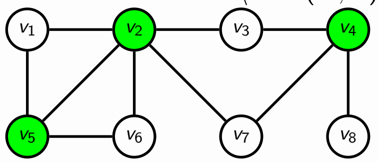
$\therefore$ VCOVER $\in$ NP-difícil

ISET $\leq_p$ VCOVER

Exemplo de redução:
Instância de ISET $\langle G = (V, E), k \rangle$:

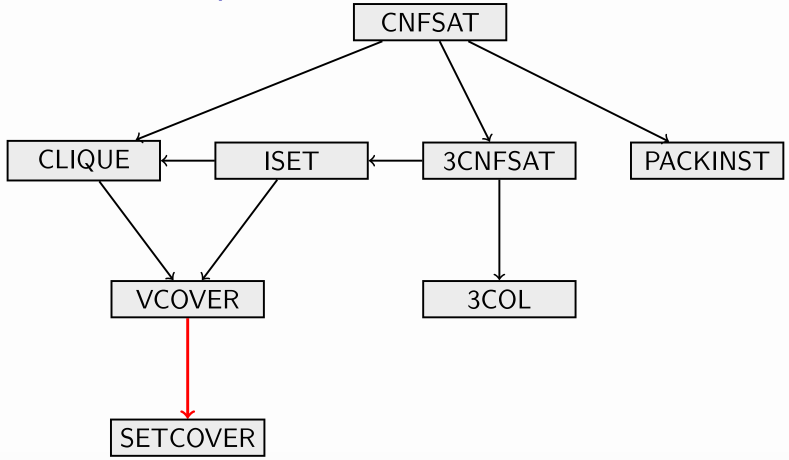


Instância de VCOVER $\langle G = (V, E), |V| - k \rangle$:



Reduções

Provar Problemas NP-Completos



VCOVER $\leq_p$ SETCOVER

Problema SETCOVER (Set Cover)
Dado um conjunto $E = \{e_1, e_2, \ldots, e_n\}$ com n elementos,
e uma colecção $\mathcal{C} = \{C_1, C_2, \ldots, C_m\}$ de m subconjuntos de E,
existe uma subcolecção $\mathcal{C}' \subseteq \mathcal{C}$ de tamanho $|\mathcal{C}'| = k$, tal que $\bigcup_{C_i \in \mathcal{C}'} C_i = E$?
(ou seja, em que $\mathcal{C}'$ cobre todo o conjunto E)

Vemos que é um problema combinatório dado que:

- teríamos de testar todas as C_k^m combinações de subconjuntos

VCOVER $\leq_p$ SETCOVER

Provar que SETCOVER $\in$ NP

1. Identificar:
   - Instância do problema SETCOVER: $\langle E, \mathcal{C}, k \rangle$
   - Certificado: $\mathcal{C}' \subseteq \mathcal{C}$, $|\mathcal{C}'| = k$
2. Dada uma instância $\langle E, \mathcal{C}, k \rangle$ e certificado $\mathcal{C}'$, o algoritmo de verificação tem de:
   - Verificar que o certificado tem tamanho polinomial
     $|\mathcal{C}'| = k \leq |\mathcal{C}| = m$, logo $O(m)$
   - Verificar o certificado na instância em tempo polinomial
     Verificar que os k subconjuntos com $O(n)$ elementos cobrem todo o conjunto E
     $\bigcup_{C_i \in \mathcal{C}'} C_i = E \in O(m.n)$

$\therefore$ ISET $\in$ NP
(dado que o certificado tem tamanho polinomial e pode ser verificado em tempo polinomial)

VCOVER $\leq_p$ SETCOVER

Provar que SETCOVER $\in$ NP-difícil: VCOVER $\leq_p$ SETCOVER
Dada instância de VCOVER $\langle G = (V, E), k \rangle$, construir instância de SETCOVER $\langle E, \mathcal{C}, k \rangle$:

- Cada vértice $v_i \in V$ corresponde a um subconjunto $C_i \in \mathcal{C}$ $\quad O(V)$
- Cada aresta $(v_i, v_j) \in E$ corresponde a um elemento $e_{ij} \in E$, que está presente nos subconjuntos C_i e C_j $\quad O(E)$

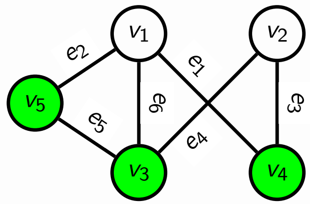
Complexidade do algoritmo de redução: O(V+E)

A redução é definida como f($\langle G = (V, E), k \rangle$) = $\langle E, \mathcal{C}, k \rangle$
$\langle G = (V, E), k \rangle \in$ VCOVER $\iff$ $\langle E, \mathcal{C}, k \rangle \in$ SETCOVER

$\therefore$ SETCOVER $\in$ NP-difícil

VCOVER $\leq_p$ SETCOVER

Exemplo de redução:

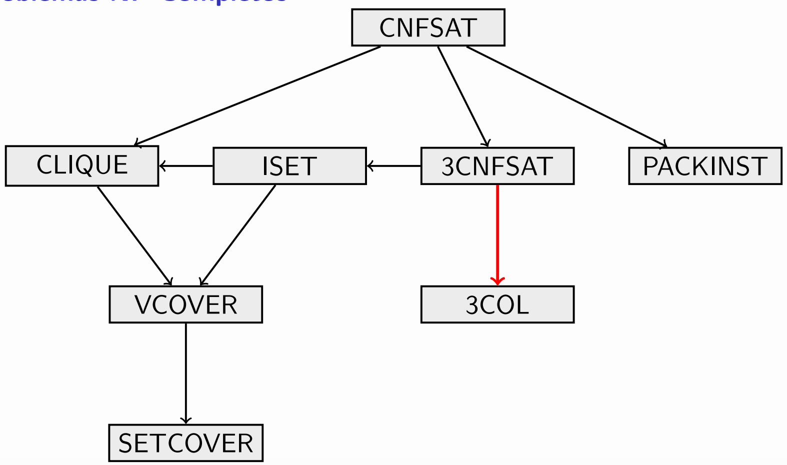


$E = \{e_1, e_2, e_3, e_4, e_5\}$
$S_1 = \{e_1, e_2, e_6\}$
$S_2 = \{e_3, e_4\}$
$S_3 = \{e_4, e_5, e_6\}$
$S_4 = \{e_1, e_3\}$
$S_5 = \{e_2, e_5\}$

$\mathcal{C}' = \{S_3, S_4, S_5\}$

Reduções

Provar Problemas NP-Completos



3CNFSAT $\leq_p$ 3COL

Problema 3COL (3-Coloring)
Dado um grafo $G = (V, E)$, existe uma coloração dos vértices V com 3 cores, tal que todos os vértices adjacentes têm cores diferentes?

Vemos que é um problema combinatório dado que:

- teríamos de testar todas as $3^{|V|}$ combinações de cores

Teorema: 3COL é NP-completo

3CNFSAT $\leq_p$ 3COL

Provar que 3COL $\in$ NP

1. Identificar:
   - Instância do problema 3COL: $\langle G = (V, E) \rangle$
   - Certificado: vector $cor[]$ com a coloração dos vértices V com 3 cores
2. Dada uma instância $\langle G = (V, E) \rangle$ e um certificado, descrever um algoritmo de verificação:
   - Verificar que o certificado tem tamanho polinomial
     $|cor[]| = |V|$, logo $O(V)$
   - Verificar o certificado na instância em tempo polinomial
     Para cada $(u, v) \in E$, verificar que $cor[u] \neq cor[v]$, logo $O(E)$

3CNFSAT $\leq_p$ 3COL

Provar que 3COL $\in$ NP-difícil: 3CNFSAT $\leq_p$ 3COL

Dada uma instância de 3CNFSAT φ, derivar instância de 3COL $\langle G = (V, E) \rangle$:

A redução é definida como $f(\varphi) = \langle G = (V, E) \rangle$

- Criar um primeiro triângulo com vértices: T (True), F (False), B (Base)
- Para variável x_j em φ, criar um triângulo com vértices: v_j, $\bar{v}_j$, B $O(n)$
  (cada variável é obrigatoriamente verdadeira ou falsa)
- Para cada cláusula $\omega_i \in \varphi$, criar grafo com 4 vértices: $O(|\varphi|)$
  - um triângulo com nós internos: l_1, l_2, l_3
  - nó interno l_1 liga ao 1º vértice da cláusula
  - nó interno l_2 liga ao 2º vértice da cláusula
  - nó interno l_3 liga a nó externo X
  - nó externo X liga com T e ao 3º vértice da cláusula

Complexidade do algoritmo de redução: $O(n + |\varphi|)$

3CNFSAT $\leq_p$ 3COL

Provar que 3COL $\in$ NP-difícil: 3CNFSAT $\leq_p$ 3COL

A redução é definida como $f(\varphi) = \langle G = (V, E) \rangle$

$\langle \varphi \rangle$ é satisfazível $\iff$ $\langle G = (V, E) \rangle$ é satisfazível

- φ é satisfazível, então $G = (V, E)$ é 3-colorível
  - Para cada cláusula ω_i existe um literal l com valor T
  - Atribuir a vértice O associado a l o valor F
  - Aos restantes vértices O atribuir cor B
  - Triangulo interno pode ser colorido com 3 cores
  - G é 3-colorível
- G é 3-colorível, então φ é satisfazível
  - Admitir φ não satisfeita
  - Existe ω_i em que todos os literais são *False*
  - Vértices em G com cor F
  - Vértices O todos com cor B
  - Triângulo interno não pode ser colorido com 3 cores $\Rightarrow$ Contradição

$\therefore$ 3COL $\in$ NP-difícil

Questões?

Dúvidas?