MAC 414
Autômatos, Computabilidade e Complexidade
aula 21 — 2/12/2020

Como lidar com NP-completude?

Como lidar com NP-completude?

L&P, 7.4.

- Casos especiais

Como lidar com NP-completude?

L&P, 7.4.

- Casos especiais
- Algoritmos de aproximação

Como lidar com NP-completude?

L&P, 7.4.

- Casos especiais
- Algoritmos de aproximação
- Backtracking e Branch-and-bound

Casos especiais

Casos especiais

2-SAT está em P

Casos especiais

$(x_1+\bar{x}_2)(\bar{x}_2+x_3)(x_2+x_4)(x_4+\bar{x}_1)$

$x_1 \leftarrow 0$ $\quad \bar{x}_2 \, (\bar{x}_2+x_3)(x_2+x_4) \to x_4 \quad x_3 \; \bar{x}_3$

$x_2 \leftarrow 0$

2-SAT está em **P**

Processo de limpeza da fórmula.

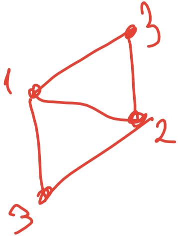
Casos especiais

2-SAT está em P

Processo de limpeza da fórmula.

k-COLORAÇÃO de grafos

Casos especiais



2-SAT está em P

Processo de limpeza da fórmula.

k-COLORAÇÃO de grafos

Dado G e um inteiro k existe coloração própria de G com k cores?

Casos especiais

2-SAT está em **P**

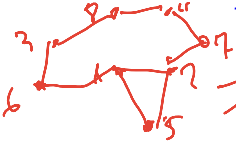
Processo de limpeza da fórmula.

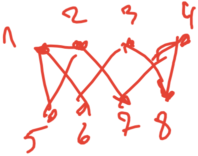
k-COLORAÇÃO de grafos

Dado G e um inteiro k existe coloração própria de G com k cores?

Em **P** para $k \leq 2$.

Casos especiais





2-SAT está em **P**

Processo de limpeza da fórmula.

k-COLORAÇÃO de grafos

Dado G e um inteiro k existe coloração própria de G com k cores?

Em **P** para $k \leq 2$.

NP-completo se $k = 3$, *mesmo se o grafo for planar*.

Casos especiais

2-SAT está em **P**

Processo de limpeza da fórmula.

k-COLORAÇÃO de grafos

Dado G e um inteiro k existe coloração própria de G com k cores?

Em **P** para $k \leq 2$.

NP-completo se $k = 3$, *mesmo se o grafo for planar*.

Em **P** se $k = 4$ e G é planar. Sempre "Sim"

Grafos especiais

- Grafos bipartidos

Grafos especiais

- Grafos bipartidos

Grafos especiais

- Grafos bipartidos

  Estão em P: NÚMERO CROMÁTICO, CLIQUE, CONJUNTO INDEPENDENTE

  Fluxos em redes
  FMCM

- Grafos planares

Grafos especiais

- Grafos bipartidos
  Estão em P: NÚMERO CROMÁTICO, CLIQUE, CONJUNTO INDEPENDENTE
- Grafos planares

Grafos especiais

$m \leq 3n - 6$

K_5 → $n = 5$, $m = 10$

- Grafos bipartidos

  Estão em P: NÚMERO CROMÁTICO, CLIQUE, CONJUNTO INDEPENDENTE

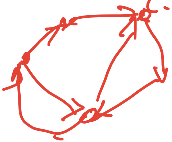
$O(n^4)$

- Grafos planares

  Estão em P: CLIQUE, FEEDBACK ARC SET.

- Vários grafos interseção:

C. Lucchesi

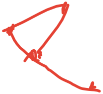


Grafos especiais

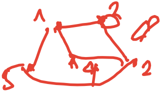
- Grafos bipartidos
  Estão em P: NÚMERO CROMÁTICO, CLIQUE, CONJUNTO INDEPENDENTE
- Grafos planares
  Estão em P: CLIQUE, FEEDBACK ARC SET.
- Vários grafos interseção:

Grafos especiais

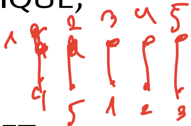
- Grafos bipartidos
  Estão em P: NÚMERO CROMÁTICO, CLIQUE, CONJUNTO INDEPENDENTE
- Grafos planares
  Estão em P: CLIQUE, FEEDBACK ARC SET.
- Vários grafos interseção:
  Vértice correspondem a objetos de uma estrutura externa, ligados se os objetos se intersectam



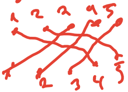
Grafos especiais



- Grafos bipartidos
  Estão em P: NÚMERO CROMÁTICO, CLIQUE, CONJUNTO INDEPENDENTE



- Grafos planares
  Estão em P: CLIQUE, FEEDBACK ARC SET.



- Vários grafos interseção:
  Vértice correspondem a objetos de uma estrutura externa, ligados se os objetos se intersectam

  Ex: coleção de intervalos, coleção de sub-árvores de uma árvore, inversões em uma permutação...

Grafos especiais

- Grafos bipartidos
  Estão em P: NÚMERO CROMÁTICO, CLIQUE, CONJUNTO INDEPENDENTE
- Grafos planares
  Estão em P: CLIQUE, FEEDBACK ARC SET.
- Vários grafos interseção:
  Vértice correspondem a objetos de uma estrutura externa, ligados se os objetos se intersectam
  Ex: coleção de intervalos, coleção de sub-árvores de uma árvore, inversões em uma permutação... Ver: Golumbic, Algorithmic Graph Theory and Perfect Graphs. 1980 → 2004

Em branco

Tava 

Algoritmos de aproximação

Algoritmos de aproximação

Problema de otimização, valor ótimo $\text{opt}(x)$, algoritmo computando $A(x)$.
Qualidade da aproximação: garantia de que $A(x)$ e $\text{opt}(x)$ estão próximos.

Algoritmos de aproximação

Problema de otimização, valor ótimo opt(x), algoritmo computando $A(x)$.
Qualidade da aproximação: grantia de que $A(x)$ e opt(x) estão próximos.

Prá agora: A é um algoritmo de $f(n)$-aproximação se

$$\frac{\max(\text{opt}(x), A(x))}{\min(\text{opt}(x), A(x))} \leq f(n), \text{ para toda instância } x \text{ de tamanho } n.$$

opt = max $\quad \frac{\text{opt}(x)}{A(x)}$ $\qquad$ opt = min $\quad \frac{A(x)}{\text{opt}}$

Algoritmos de aproximação

Problema de otimização, valor ótimo $\mathrm{opt}(x)$, algoritmo computando $A(x)$.
Qualidade da aproximação: grantia de que $A(x)$ e $\mathrm{opt} x$ estão próximos.

Prá agora: A é um algoritmo de $f(n)$-aproximação se

$$\frac{\max(\mathrm{opt}(x), A(x))}{\min(\mathrm{opt}(x), A(x))} \leq f(n), \text{ para toda instância } x \text{ de tamanho } n.$$

Mais comum: $f(n)$ é constante.

Algoritmos de aproximação

Problema de otimização, valor ótimo $\text{opt}(x)$, algoritmo computando $A(x)$.
Qualidade da aproximação: grantia de que $A(x)$ e $\text{opt}x$ estão próximos.

Prá agora: A é um algoritmo de $f(n)$-aproximação se

$$\frac{\max(\text{opt}(x), A(x))}{\min(\text{opt}(x), A(x))} \leq f(n), \text{ para toda instância } x \text{ de tamanho } n.$$

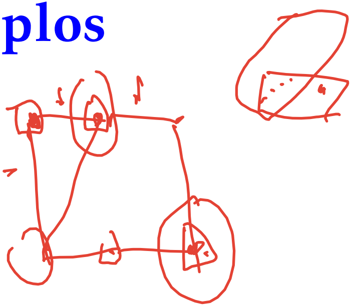
Cris

Yô

Mais comum: $f(n)$ é constante.

MAC0450

Exemplos



- NODE COVER
  2-aproximação

Exemplos

- NODE COVER
  2-aproximação
- TRAVELING SALESMAN
  sem aproximação constante.

$G \longrightarrow K_n$

$$c_{ij} = \begin{cases} 1 & \text{se } ij \in EG \\ 2 & \text{se } ij \notin EG \end{cases}$$

$1+\varepsilon$

$1+n\varepsilon$

G Ham n

n Ham $n(1+\varepsilon)$

Backtracking

Backtracking

Aplicável para problemas em **NP** em que é possível ter "soluções parciais".

Backtracking

Aplicável para problemas em **NP** em que é possível ter "soluções parciais".

Dada uma instância, ramifica em subproblemas conforme a solução parcial escolhida.

Backtracking

Aplicável para problemas em **NP** em que é possível ter "soluções parciais".

Dada uma instância, ramifica em subproblemas conforme a solução parcial escolhida.

Se um ramo da árvore chega a uma solução, ôba!

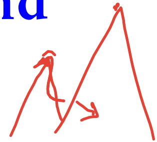
Branch-and-bound

Branch-and-bound

Backtracking em problemas de otimização.

Estimativas da função objetivo ajudam a podar a árvore.

Branch-and-bound



New-age

Backtracking em problemas de otimização.
Estimativas da função objetivo ajudam a podar a árvore.

Xô
Carlinhos

Simmulated
annealing

MAC0473

Polinomial, pero no mucho

Polinomial, pero no mucho

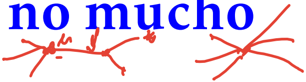
Dado um grafo G e uma aresta $\alpha = uv$, definimos:

Polinomial, pero no mucho

Dado um grafo G e uma aresta $\alpha = uv$, definimos:

$G \backslash \alpha$ (remoção), subgrafo gerador sem α

Polinomial, pero no mucho



Dado um grafo G e uma aresta $\alpha = uv$, definimos:

$G \backslash \alpha$ (remoção), subgrafo gerador sem α

G / α (contração), obtido de $G \backslash \alpha$ identificando u e v

Polinomial, pero no mucho

Dado um grafo G e uma aresta $\alpha = uv$, definimos:

$G \backslash \alpha$ (remoção), subgrafo gerador sem α

G/α (contração), obtido de $G \backslash \alpha$ identificando u e v

Um menor de G: grafo obtido de G por uma sequência de remoções e contrações.

Teorema (Robertson-Seymour, 1983–2004)

Em todo conjunto infinito de grafos, existe um que é menor que o outro.

do

Polinomial, pero no mucho

Dado um grafo G e uma aresta $\alpha = uv$, definimos:

$G \backslash \alpha$ (remoção), subgrafo gerador sem α

G/α (contração), obtido de $G \backslash \alpha$ identificando u e v

Um menor de G: grafo obtido de G por uma sequência de remoções e contrações.

Teorema (Robertson-Seymour, 1983–2004)

Em todo conjunto infinito de grafos, existe um que é menor que o outro.

Polinomial, pero no mucho

Dado um grafo G e uma aresta $\alpha = uv$, definimos:

$G \backslash \alpha$ (remoção), subgrafo gerador sem α

G/α (contração), obtido de $G \backslash \alpha$ identificando u e v

Um menor de G: grafo obtido de G por uma sequência de remoções e contrações.

Teorema (Robertson-Seymour, 1983–2004)

Em todo conjunto infinito de grafos, existe um que é menor que o outro.

Uma propriedade de grafos é hereditária se ela é preservada para menores.

A B G não tem menor em A

Ex: Fixada uma superfície: grafo desenhável nela.

Kuratowski
Wagner

Ex: Fixada uma superfície: grafo desenhável nela.
Um desenho pode ser "descrito" por uma estrutura combinatória em cima do grafo: problema em **NP**.

Ex: Fixada uma superfície: grafo desenhável nela.
Um desenho pode ser "descrito" por uma estrutura combinatória em cima do grafo: problema em **NP**.

RS também deram algoritmo polinomial para cada grafo H: dado G, ele tem um menor isomorfo a H?

Ex: Fixada uma superfície: grafo desenhável nela.
Um desenho pode ser "descrito" por uma estrutura combinatória em cima do grafo: problema em **NP**.

RS também deram algoritmo polinomial para cada grafo H: dado G, ele tem um menor isomorfo a H?

Corolário

Para cada propriedade hereditária, existe um algoritmo polinomial que decide se um dado grafo a satisfaz.

Ex: Fixada uma superfície: grafo desenhável nela.
Um desenho pode ser "descrito" por uma estrutura combinatória em cima do grafo: problema em **NP**.

RS também deram algoritmo polinomial para cada grafo H: dado G, ele tem um menor isomorfo a H?

Corolário

Para cada propriedade hereditária, existe um algoritmo polinomial que decide se um dado grafo a satisfaz.

Ex: Fixada uma superfície: grafo desenhável nela.
Um desenho pode ser "descrito" por uma estrutura combinatória em cima do grafo: problema em **NP**.

RS também deram algoritmo polinomial para cada grafo H: dado G, ele tem um menor isomorfo a H?

Corolário

Para cada propriedade hereditária, existe um algoritmo polinomial que decide se um dado grafo a satisfaz.

Mas a demonstração não dá a mínima idéia de como obter esse algoritmo!