Problemas NP-Completos

Fábio Botler

Programa de Engenharia de Sistemas e Computação
Universidade Federal do Rio de Janeiro

Aula passada

- A classe P
- Problemas aparentemente difíceis
- A classe NP

Aula de hoje

- A classe NP
- A questão P=NP
- Complementos de Problemas

A classe NP

A classe NP

Um **certificado** para um problema Π é uma justificativa para a resposta SIM.

A classe NP

Um **certificado** para um problema Π é uma justificativa para a resposta SIM.

$$NP \quad = \quad \left\{ \begin{array}{l} \text{problemas de decisão para os quais existe} \\ \text{certificado que pode ser reconhecido por} \\ \text{um algoritmo polinomial} \\ \\ \end{array} \right\}$$

A classe NP

Um **certificado** para um problema Π é uma justificativa para a resposta SIM.

$$NP \quad = \quad \left\{ \begin{array}{l} \text{problemas de decisão para os quais existe} \\ \text{certificado que pode ser reconhecido por} \\ \text{um algoritmo polinomial } \textbf{no tamanho} \\ \textbf{da sua entrada} \end{array} \right\}$$

A classe NP

Um **certificado** para um problema Π é uma justificativa para a resposta SIM.

$$NP = \left\{ \begin{array}{l} \text{problemas de decisão para os quais existe} \\ \text{certificado que pode ser reconhecido por} \\ \text{um algoritmo polinomial } \textbf{no tamanho} \\ \textbf{da sua entrada} \end{array} \right\}$$

▷ O certificado também deve ser polinomial no tamanho da entrada!

Como verificar se um problema Π pertence a NP?

Como verificar se um problema Π pertence a NP?

- $\triangleright$ Definir uma justificativa J conveniente para a resposta SIM

Como verificar se um problema Π pertence a NP?

- $\triangleright$ Definir uma justificativa J conveniente para a resposta SIM
- $\triangleright$ Elaborar um algoritmo R para reconhecer se J está correta.

  O conjunto de dados desse algoritmo são os pares (I, J), onde I é uma instância de Π e J é uma justificativa.

Como verificar se um problema Π pertence a NP?

- $\triangleright$ Definir uma justificativa J conveniente para a resposta SIM
- $\triangleright$ Elaborar um algoritmo R para reconhecer se J está correta.

  O conjunto de dados desse algoritmo são os pares (I, J), onde I é uma instância de Π e J é uma justificativa.

Se R for polinomial no tamanho de I, então Π pertence a NP.

Satisfatibilidade

DADOS:	Uma expressão booleana E na Forma Normal Conjuntiva (FNC)
OBJETIVO:	E é satisfatível?

Satisfatibilidade

DADOS:	Uma expressão booleana E na Forma Normal Conjuntiva (FNC)
OBJETIVO:	E é satisfatível?

Certificado: uma atribuição para cada variável de E

Satisfatibilidade

DADOS:	Uma expressão booleana E na Forma Normal Conjuntiva (FNC)
OBJETIVO:	E é satisfatível?

Certificado:	uma atribuição para cada variável de E
Reconhecimento:	substituir em E cada variável. por seu valor atribuído.

Satisfatibilidade

DADOS:	Uma expressão booleana E na Forma Normal Conjuntiva (FNC)
OBJETIVO:	E é satisfatível?

Certificado:	uma atribuição para cada variável de E
Reconhecimento:	substituir em E cada variável. por seu valor atribuído. se cada cláusula possuir pelo menos uma atribuição 1, então E é satisfatível.

Clique

DADOS:	Um grafo G e um inteiro k
OBJETIVO:	G possui uma clique de tamanho pelo menos k?

Clique

DADOS:	Um grafo G e um inteiro k
OBJETIVO:	G possui uma clique de tamanho pelo menos k?

Certificado: um subconjunto de vértices V'

Clique

DADOS:	Um grafo G e um inteiro k
OBJETIVO:	G possui uma clique de tamanho pelo menos k?

Certificado:	um subconjunto de vértices V'
Reconhecimento:	verificar se $\|V'\| \geq k$, e se há par $x, y \in V'$ tal que $xy \notin E(G)$.

Clique

DADOS: Um grafo G e um inteiro k
OBJETIVO: G possui uma clique de tamanho pelo menos k?

Certificado: um subconjunto de vértices V'

Reconhecimento: verificar se $|V'| \geq k$, e
se há par $x, y \in V'$ tal que $xy \notin E(G)$.
se $|V'| \geq k$ e não houver tal par,
então a justificativa está correta.

Cobertura por vértices

DADOS:	Um grafo G e um inteiro k
OBJETIVO:	G possui uma cobertura por vértices de tamanho no máximo k?

Cobertura por vértices

DADOS:	Um grafo G e um inteiro k
OBJETIVO:	G possui uma cobertura por vértices de tamanho no máximo k?

Certificado: um subconjunto de vértices V'

Cobertura por vértices

DADOS: Um grafo G e um inteiro k

OBJETIVO: G possui uma cobertura por vértices de tamanho no máximo k?

Certificado: um subconjunto de vértices V'

Reconhecimento: verificar se $|V'| \leq k$, e
se há aresta $xy \in E(G)$ tal que $x, y \notin V'$.

Cobertura por vértices

DADOS:	Um grafo G e um inteiro k
OBJETIVO:	G possui uma cobertura por vértices de tamanho no máximo k?

Certificado:	um subconjunto de vértices V'
Reconhecimento:	verificar se $\lvert V' \rvert \leq k$, e se há aresta $xy \in E(G)$ tal que $x, y \notin V'$. se $\lvert V' \rvert \leq k$ e não houver tal aresta, então a justificativa está correta.

Clique máxima

DADOS:	Um grafo G e um inteiro k
OBJETIVO:	uma clique máxima de G tem tamanho k?

Clique máxima

DADOS:	Um grafo G e um inteiro k
OBJETIVO:	uma clique máxima de G tem tamanho k?

Certificado: um conjunto S com todas as cliques maximais de G

Clique máxima

DADOS:	Um grafo G e um inteiro k
OBJETIVO:	uma clique máxima de G tem tamanho k?

Certificado:	um conjunto S com todas as cliques maximais de G
Reconhecimento:	comprovar se S tem todas as cliques maximais de G, e verificar se o tamanho da maior clique é k

Clique máxima

DADOS:	Um grafo G e um inteiro k
OBJETIVO:	uma clique máxima de G tem tamanho k?

Certificado:	um conjunto S com todas as cliques maximais de G
Reconhecimento:	comprovar se S tem todas as cliques maximais de G, e verificar se o tamanho da maior clique é k
	se ambas respostas forem afirmativas, então a justificativa está correta.

Caminho mínimo

DADOS:	Um grafo G, dois vértices $x, y \in V(G)$, e um inteiro k
OBJETIVO:	G possui um caminho ligando x a y com comprimento no máximo k?

Caminho mínimo

DADOS: Um grafo G, dois vértices $x, y \in V(G)$, e um inteiro k

OBJETIVO: G possui um caminho ligando x a y com comprimento no máximo k?

Certificado:

Caminho mínimo

DADOS: Um grafo G, dois vértices $x, y \in V(G)$,
e um inteiro k

OBJETIVO: G possui um caminho ligando x a y
com comprimento no máximo k?

Certificado:

Reconhecimento: executar uma busca em largura começando em x
e guardando, para cada vértice z,
sua distância $d(z)$ até x.

Caminho mínimo

DADOS: Um grafo G, dois vértices $x, y \in V(G)$,
e um inteiro k

OBJETIVO: G possui um caminho ligando x a y
com comprimento no máximo k?

Certificado:

Reconhecimento: executar uma busca em largura começando em x
e guardando, para cada vértice z,
sua distância $d(z)$ até x.

se $d(y) \leq k$, então a justificativa está correta.

Wolfram MathWorld™ the web's most extensive mathematics resource
Built with Mathematica Technology

AKS Primality Test

In August 2002, M. Agrawal and colleagues announced a deterministic algorithm for determining if a number is prime that runs in polynomial time (Agrawal *et al.* 2004). While this had long been believed possible (Wagon 1991), no one had previously been able to produce an explicit polynomial time deterministic algorithm (although probabilistic algorithms were known that seem to run in polynomial time). This test is now known as the Agrawal-Kayal-Saxena primality test, cyclotomic AKS test, or AKS primality test.

Annals of Mathematics, **160** (2004), 781–793

PRIMES is in P

By Manindra Agrawal, Neeraj Kayal, and Nitin Saxena*

Abstract

We present an unconditional deterministic polynomial-time algorithm that determines whether an input number is prime or composite.

Primalidade

DADOS:	Um inteiro n
OBJETIVO:	n é primo?

Primalidade

DADOS:	Um inteiro n
OBJETIVO:	n é primo?

Certificado:

Primalidade

DADOS:	Um inteiro n
OBJETIVO:	n é primo?

Certificado:

Reconhecimento: executar o AKS.

Primalidade

DADOS: Um inteiro n
OBJETIVO: n é primo?

Certificado:

Reconhecimento: executar o AKS.

se o AKS retornar SIM,
então a justificativa está correta.

Qual a relação entre as classes P e NP?

Qual a relação entre as classes P e NP?

Proposição

$P \subseteq NP$

Qual a relação entre as classes P e NP?

Proposição

$P \subseteq NP$

Proof.

Seja Π um problema em P, e A um algoritmo polinomial que decide Π.

Qual a relação entre as classes P e NP?

Proposição

$P \subseteq NP$

Proof.

Seja Π um problema em P, e A um algoritmo polinomial que decide Π.

Certificado:

Qual a relação entre as classes P e NP?

Proposição

$P \subseteq NP$

Proof.

Seja Π um problema em P, e A um algoritmo polinomial que decide Π.

Certificado:

Reconhecimento: executar A.

Qual a relação entre as classes P e NP?

Proposição

$P \subseteq NP$

Proof.

Seja Π um problema em P, e A um algoritmo polinomial que decide Π.

Certificado:

Reconhecimento: executar A.

se A retornar SIM,
então a justificativa está correta.

$\square$

$$NP \subseteq P?$$

Classes de Problemas

$$P = \left\{ \begin{array}{l} \text{problemas de decisão que podem ser} \\ \text{decididos por um algoritmo polinomial} \end{array} \right\}$$

$$NP = \left\{ \begin{array}{l} \text{problemas de decisão para os quais existe} \\ \text{certificado que pode ser reconhecido por} \\ \text{um algoritmo polinomial no tamanho} \\ \text{de sua entrada} \end{array} \right\}$$

Proposição

Seja $\Pi \in NP$*. Existe algoritmo exponencial que decide* Π.

Complementos de Problemas

Certificados

Um **certificado** para um problema Π é uma justificativa para a resposta SIM.

Um **co-certificado** para um problema Π é uma justificativa para a resposta NÃO.

Classes de Problemas

$$NP = \left\{ \begin{array}{l} \text{problemas de decisão para os quais existe} \\ \text{certificado que pode ser reconhecido por} \\ \text{um algoritmo polinomial no tamanho} \\ \text{da sua entrada} \end{array} \right\}$$

$$\text{Co-}NP = \left\{ \begin{array}{l} \text{problemas de decisão para os quais existe} \\ \textbf{co}\text{-certificado que pode ser reconhecido por} \\ \text{um algoritmo polinomial no tamanho} \\ \text{da sua entrada} \end{array} \right\}$$

Definição

Dado um problema de decisão Π, *o problema* $\overline{\Pi}$ *é o problema tal que* $\Pi(I) = SIM$ *se e somente se* $\overline{\Pi}(I) = N\tilde{A}O$.

Satisfatibilidade

DADOS:	Uma expressão booleana E na Forma Normal Conjuntiva (FNC)
OBJETIVO:	decidir se E é satisfatível.

Satisfatibilidade

DADOS:	Uma expressão booleana E na Forma Normal Conjuntiva (FNC)
OBJETIVO:	decidir se E é satisfatível.

$\overline{\text{Satisfatibilidade}}$

DADOS:	Uma expressão booleana E na Forma Normal Conjuntiva (FNC)
OBJETIVO:	decidir se E **não** é satisfatível.

Clique

DADOS:	Um grafo G e um inteiro k
OBJETIVO:	decidir se G possui uma clique de tamanho pelo menos k.

`Clique`

DADOS:	Um grafo G e um inteiro k
OBJETIVO:	decidir se G possui uma clique de tamanho pelo menos k.

$\overline{\texttt{Clique}}$

DADOS:	Um grafo G e um inteiro k
OBJETIVO:	decidir se G **não** possui uma clique de tamanho pelo menos k.

Cobertura por vértices

DADOS:	Um grafo G e um inteiro k
OBJETIVO:	decidir se G possui uma cobertura por vértices de tamanho no máximo k.

Cobertura por vértices

DADOS: Um grafo G e um inteiro k

OBJETIVO: decidir se G possui uma cobertura por vértices de tamanho no máximo k.

Cobertura por vértices

DADOS: Um grafo G e um inteiro k

OBJETIVO: decidir se G **não** possui uma cobertura por vértices de tamanho no máximo k.

Proposição

$Co\text{-}NP = \{\overline{\Pi} \colon \Pi \in NP\}$.

Proposição

Co-NP $= \{\overline{\Pi} \colon \Pi \in NP\}$.

Proposição

$\Pi \in NP$ *se e somente se* $\overline{\Pi} \in$ *Co-NP*.

Proposição

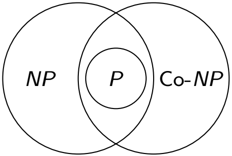
$Co\text{-}NP = \{\overline{\Pi} \colon \Pi \in NP\}$.

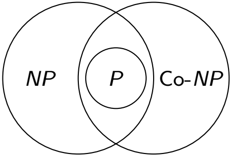
Proposição

$\Pi \in NP$ *se e somente se* $\overline{\Pi} \in Co\text{-}NP$.

Proposição

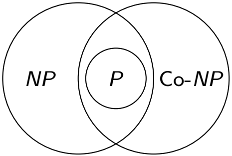
$P \subseteq Co\text{-}NP$.


NP
P
Co-NP



Problemas em aberto:

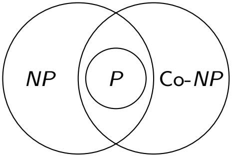
- $NP = \text{Co-}NP$?
- $P = NP \cap \text{Co-}NP$?



Problemas em aberto:

- $NP = \text{Co-}NP$?
- $P = NP \cap \text{Co-}NP$?

Se $P = NP$, então $NP = \text{Co-}NP$ e $P = NP \cap \text{Co-}NP$.



Problemas em aberto:

- $NP = \text{Co-}NP$?
- $P = NP \cap \text{Co-}NP$?

Se $P = NP$, então $NP = \text{Co-}NP$ e $P = NP \cap \text{Co-}NP$.

Se $NP = \text{Co-}NP$, então $P = NP$?

Problemas NP-Completos

Fábio Botler

Programa de Engenharia de Sistemas e Computação
Universidade Federal do Rio de Janeiro