Intratabilidade

André Luís Mendes Fakhoury
Eduardo Dias Pennone
Gustavo Vinícius Vieira Silva Soares
Matheus Steigenberg Populim
Thiago Preischadt Pinheiro

SCC0205 - Teoria da Computação e Linguagens Formais
Prof. Diego Raphael Amancio

ICMC - USP

Novembro de 2020

Sumário

Sumário

Introdução

Sumário

Computação eficiente

Visão geral

Sumário

Classes P e NP

Visão geral

Sumário

Hipóteses P $\neq$ NP

Visão geral

Sumário

Reduções em tempo polinomial

Visão geral

Sumário

NP-difícil

Visão geral

Sumário

NP-completo
Introdução

- Década de 1970 - Stephen Cook e Leonid Levin
- Alguns problemas em NP possuem complexidade individual relacionada com toda a classe
- Caso um algoritmo em tempo polinomial exista para algum deles, todos os problemas em NP podem ser resolvidos em tempo polinomial

NP-completo

Definição

Definição

Uma linguagem B é dita *NP-completa* caso satisfaça estas duas características:

1. B está em NP;
2. Todo A em NP é redutível em tempo polinomial para B.

NP-completo
Definição

Também pode-se definir a partir da classe NP-difícil:

Definição

Uma linguagem B é dita *NP-completa* caso satisfaça estas duas características:

1. B está em NP;
2. B é NP-difícil.

NP-completo

Definição

Teorema

Se B é *NP-completo* e $B \leq_p C$ (B é redutível em tempo polinomial para C), para C em NP, então C é *NP-completo*.

Com isso, pode-se utilizar um problema NP-completo para encontrar outros problemas NP-completos!

NP-completo
Consequências

- Novos problemas NP-completos podem ser encontrados a partir de problemas NP-completos já conhecidos
- Achar uma solução polinomial para algum NP-completo implica que todos os NP possuem solução polinomial
- Estudos podem ser mais focados diretamente nos NP-completos

NP-completo

Problemas

- **Teorema de Cook-Levin:** SAT
- 21 problemas de Karp - *Reducibility among combinatorial problems* [3]
- SAT, 3SAT, clique, caixeiro viajante (decisão), ...

NP-completo

SAT (Satisfatibilidade booleana)

SAT

Verificar se existe uma valoração para as variáveis que compõe uma fórmula booleana, de forma que esta tenha saída verdadeira.

$$\text{Exemplo: } (x \vee y \vee z) \wedge (\neg x \vee y) \wedge (y \vee \neg z)$$

Neste caso, pode-se atribuir x, y, z como *verdadeiros*, a saída será *verdadeira*.

NP-completo

SAT (Satisfatibilidade booleana)

- Primeiro problema provado NP-completo, pelo **Teorema de Cook-Levin**;
- Auto-redutível: algoritmo que resolve o problema (e verifica se existe tal composição de valores para variáveis) também consegue encontrar a atribuição de variáveis satisfatórias;

NP-completo
3SAT

3SAT

SAT com expressões na forma normal conjuntiva, e cada cláusula contém exatamente três variáveis.

$$(x \vee y \vee z) \wedge (\neg x \vee \neg w \vee z) \wedge (x \vee w \vee \neg z)$$

NP-completo
3SAT

Ideia da demonstração

É **NP** pois dada uma coleção de cláusulas e uma valoração para as variáveis, é possível verificar em tempo polinomial se estas cláusulas são satisfeitas.

SAT $\leq_p$ 3SAT - expressões são denotadas na forma normal conjuntiva, e cláusulas com mais de três termos são substituídas por mais cláusulas com três termos.

NP-completo

Clique

Clique

Dado um grafo $G(V, A)$ e um inteiro k, encontrar se existe um subgrafo de k ou mais vértices, em que todos nós dois a dois são conectados por uma arestas - ou seja, se G possui um conjunto de k nós mutualmente adjacentes.

NP-completo
Clique

Ideia da demonstração

É **NP** pois é possível construir um verificador que recebe um grafo $G(V, A)$, um inteiro k e um conjunto S, e verifica se S é um clique válido em $\mathcal{O}(|S|^2)$.

3SAT $\leq_p$ CLIQUE - constrói-se um grafo a partir da expressão booleana. Será feita a construção de uma função que recebe qualquer instância do 3SAT e retorna uma instância de Clique, que é verdadeira se e somente se a instância do 3SAT for verdadeira.

NP-completo

Cobertura de vértices (decisão)

Cobertura de vértices (decisão)

Dado um grafo $G(V, A)$ e um inteiro k, encontrar se existe um subconjunto de vértices V' de tamanho k tal que toda aresta em A esteja conectada em no mínimo um vértice em V'.

NP-completo

Cobertura de vértices (decisão)

Ideia da demonstração

NP: dado um subconjunto V', pode-se verificar com complexidade polinomial se este é um subconjunto correto (basta percorrer por todas as arestas e verificar se algum de seus vértices pertence ao conjunto V').

CLIQUE $\leq_p$ Vertex Cover: dado um grafo $G(V, A)$, seu complemento $\bar{G}(V, \bar{A})$ possui os mesmos vértices de G, e nenhuma de suas arestas (mas sim todas as arestas que não estão em G). Se existe um clique V' em $\bar{G}$, então $V - V'$ é uma cobertura de vértices em G.

NP-completo

Caminho hamiltoniano

Caminho hamiltoniano (HAMPATH)

O problema do caminho hamiltoniano consiste em, dado um grafo $G(V, A)$ e dois vértices $s, t \in V$, verificar se existe um caminho de s até t que passa por todos os vértices $\in V$ exatamente uma vez.

NP-completo

Caminho hamiltoniano

Ideia da demonstração

NP: Dado um caminho C, pode-se facilmente verificar se este é um caminho válido com complexidade polinomial $\mathcal{O}(|V| + |A|)$.

É possível mostrar que 3SAT $\leq_p$ HAMPATH.

NP-completo

Caixeiro viajante (decisão)

Caixeiro viajante (decisão)

Dados um grafo $G(V, A)$ e uma distância L, retornar se existe um ciclo (que visite todos os vértices exatamente uma vez) com distância total no máximo L.

NP-completo

Caixeiro viajante (decisão)

Ideia da demonstração

Dado um grafo $G(V, A)$, um inteiro L e um ciclo c, pode-se verificar se o ciclo c de fato é um ciclo válido neste grafo e se a soma de suas arestas não ultrapassa L, com complexidade $\mathcal{O}(|c|)$.

É possível mostrar que HamCycle $\leq_p$ TSP.

Sumário

Conclusão

- Área muito importante e interessante.
- Classificação de problemas em classes;
- Provar que determinados problemas são "tão difíceis" quanto vários outros estudados;
- Não "perder tempo" com problemas já considerados difíceis;
- P vs NP é um dos "Millennium Prize Problems";
- Espera-se que $P \neq NP$, mas ainda não foi provado;

Sumário

Referência Bibliográfica I

- Michael R. Garey and David S. Johnson.
  Computers and Intractability - A Guide to the Theory of NP-Completeness.
  1979.
- David Harel and Yishai Feldman.
  Algorithmics - Spirit of Computing.
  Addison Wesley, 2004.
- Richard M. Karp.
  Reducibility among combinatorial problems.
  1972.
- Richard Peng.
  Cs 3510 design and analysis of algorithms - np-complete problems, 2016.

Referência Bibliográfica II

Michael Sipster.
Introduction to the Theory of Computation.
3rd edition, 2013.

Túlio Ferreira Valeri.
Aplicabilidade do problema do caixeiro viajante na roteirização de visitas de representantes de empresas aos clientes., 2019.