DCC206 – Algoritmos 1

Aula 21 – Introdução a Teoria da Complexidade (Parte 03)

Professor Renato Vimieiro

DCC/ICEx/UFMG

Introdução

- Como vimos na aula passada, existe uma subclasse de problemas NP que são pelo menos tão difíceis quanto quaisquer outros da classe
- Essa subclasse é conhecida como a classe dos problemas NP-Completos
- Vimos que o teorema de Cook-Levin estabeleceu o ‘primeiro’ problema NP-Completo
- Vimos também que, pela definição de redução polinomial, a demonstração de pertinência de novos problemas à subclasse pode ser feita reduzindo-se um NP-completo a esses problemas
- Hoje veremos algumas reduções que, além de demonstrar que alguns problemas são NP-Completos, podem servir de base para construção de algoritmos

3SAT é NP-completo

- Uma variante do SAT consiste em restringir que, além de estar na CNF, as funções Booleanas sejam compostas por k cláusulas com 3 literais em cada
- Essa variante é conhecida como 3SAT
- Uma forma de demonstrar que 3SAT é NP-completo é reduzir polinomialmente SAT a ele
- Uma expressão ($a1 \lor a2 \lor a3 \lor a4$) pode ser reescrita como:
  - ($a1 \lor a2 \lor z$) $\land$ ($\neg z \lor a3 \lor a4$)
  - No caso geral, ($a1 \lor a2 \lor ... \lor ak$) -> ($a1 \lor a2 \lor z1$) $\land$ ($\neg z1 \lor a3 \lor z2$) $\land$ ($\neg z2 \lor a4 \lor z3$) $\land$ ... ($\neg zk\text{-}3 \lor ak\text{-}1 \lor ak$)
- Essa transformação introduz uma quantidade linear de novos literais, e, portanto, mantém a complexidade da redução polinomial
- Assim, demonstramos que o 3SAT é NP-difícil. A demonstração de que ele é NP é equivalente ao do SAT.

Conjunto independente é NP-Completo

- Na aula passada, estudamos o problema do conjunto independente máximo e sua variante de decisão
- É fácil perceber que o problema é NP, já que, dado um certificado, podemos verificar se é uma solução válida em tempo polinomial
- Vamos mostrar que o problema também é NP-difícil, reduzindo 3SAT a ele
- No 3SAT, nossa entrada é uma função Booleana com n variáveis e k cláusulas
- No conjunto independente, temos um grafo e um inteiro k
- Logo, para reduzir 3SAT $\leq_p$ CI, precisamos mostrar como construir um grafo a partir de uma função de forma que ela é satisfatível sse existe conjunto independente em G com pelo menos k vértices.

Conjunto independente é NP-Completo

- O conjunto de vértices do nosso grafo será composto pelos literais que aparecem em cada uma das cláusulas da função
  - Teremos 3k vértices no grafo
- Dois vértices são ligados se eles pertencem à mesma cláusula ou se um é o complemento do outro
- O grafo ao lado é construído a partir de $(a \vee b \vee c) \wedge (b \vee \bar{c} \vee \bar{d}) \wedge (\bar{a} \vee c \vee d) \wedge (a \vee \bar{b} \vee \bar{d})$

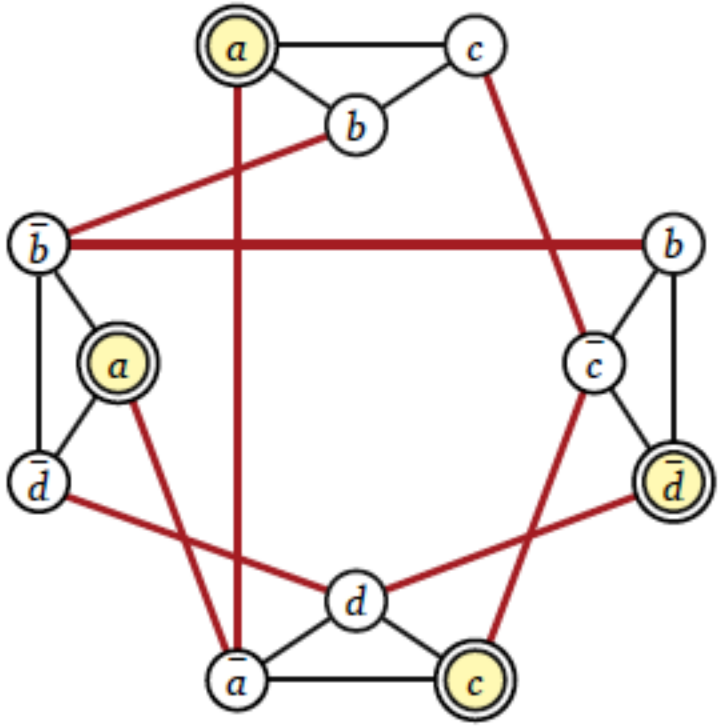


Figura 12.8 - Algorithms, Jeff Erickson

Conjunto independente é NP-Completo

- Note que qualquer conjunto independente em G é formado por vértices de diferentes triângulos
- Logo, um conjunto independente em G tem tamanho, no máximo, k
- Se a função for satisfatível, então, pelo menos, um literal de cada cláusula é verdadeiro. Como essa solução não pode conter uma variável e seu complemento, podemos formar um conjunto independente de tamanho exatamente k
- Por outro lado, se existir um conjunto independente S de tamanho k, então sabemos que cada vértice pertence a um triângulo diferente (cláusula) e que um vértice representando uma variável e seu complemento não podem ambos pertencer a S. Portanto, se atribuirmos 1 a cada literal em S e escolher arbitrariamente valores para variáveis não pertencentes a S, fazemos com que a função avalie para 1.
- Portanto, existe conjunto independente de tamanho k em G sse a função for satisfatível. Ou seja, 3SAT $\leq_p$ Conjunto Independente

Cobertura de Vértices é NP-Completo

- Vimos que CI $\leq_p$ CV, portanto, Cobertura de Vértices é NP-difícil
- Podemos verificar uma solução para o problema em tempo polinomial
- Logo, Cobertura de Vértices também é NP
- Portanto, Cobertura de Vértices é NP-Completo

Subset Sum é NP-Completo

- Durante o estudo de paradigmas, vimos diferentes soluções para variantes de um problema chamado problema da mochila
- Esse problema consistia em escolher itens para compor uma mochila de forma a preservar sua capacidade com o objetivo de maximizar o valor transportado
- Um problema similar ao da mochila é o seguinte:
  - Dado um conjunto S de números naturais e um natural w, existe $X \subseteq S$ tal que $\sum_{x \in X} x = w$?
- Ou seja, queremos determinar um subconjunto de S cuja soma seja exatamente w

Subset sum é NP-Completo

- É fácil demonstrar que esse problema pertence à classe NP
- Vamos demonstrar que ele também é NP-difícil
- Para isso, vamos reduzir cobertura de vértices a ele
- Suponha que numeramos todos os vértices de 1 a n, e todas as arestas de 1 a m
- Agora construímos uma matriz de incidência da seguinte forma H[i,j] = 1 sempre que o vértice i for um dos terminais de j, caso contrário H[i,j] = 0
- Essa matriz é construída em tempo polinomial

Subset sum é NP-Completo

- Agora vamos construir o nosso conjunto S
- Para cada vértice i, geramos o seguinte número $a_i = 4^{m+1} + \sum_{j=1}^{m} H[i,j]4^j$
- Para cada aresta j, geramos o número $b_j = 4^j$
- Nosso conjunto $S = \{a_i | 1 \leq i \leq n\} \cup \{b_j | 1 \leq j \leq m\}$
- Como valor alvo fazemos $w = k4^{m+1} + \sum_{j=1}^{m} 2 \cdot 4^j$
- Onde k é o tamanho da cobertura exigida

Subset sum é NP-Completo

- Note que, para conseguir alcançar w, precisamos escolher k números a_i
- Ou seja, precisamos escolher k números gerados a partir dos vértices
- Como cada vértice contribui com 4^j para cada aresta em que ele é incidente, temos duas escolhas para completar o restante da soma:
  - Escolher os dois terminais de uma aresta
  - Ou escolher um número b_j referente a uma aresta coberta por algum a_i
- Assim, se G tem uma cobertura de vértices de tamanho k, basta escolhermos os números referentes a esses vértices mais o números gerados por cada aresta para alcançarmos a soma

Subset sum é NP-Completo

- Agora suponha que um subconjunto $X \subseteq S$ tem soma igual a w
- Como dito, $k4^{m+1}$ só pode ser alcançado se k a_i's forem escolhidos
- Para completar a segunda parte, cada aresta contribui com, no máximo, $b_j = 4^j$ para a soma
- Como precisamos de duas vezes esse valor para cada aresta, X inclui pelo menos um terminal de cada aresta
- Logo, se escolhermos os vértices referentes aos a_i's temos uma cobertura de vértices em G
- Dessa forma, há uma cobertura de vértices em G de tamanho no máximo k sse existir um subconjunto de S cuja soma é w
- Portanto, Subset sum é NP-completo
- O subset sum é um caso particular do problema da mochila em que os pesos são iguais aos valores. Portanto, mochila também é NP-Completo

Circuito Hamiltoniano é NP-Completo

- Considere o problema de determinar se existe um ciclo (simples) que passa por todos os vértices de um grafo G
- Esse problema é conhecido como circuito Hamiltoniano
- Um certificado para o problema seria uma lista ordenada de vértices
- Assim, podemos verificar em tempo polinomial se cada vértice aparece somente uma vez na lista e se todo para de vértices consecutivos são adjacentes (além de o último ser adjacente ao primeiro)
- Logo, o problema pertence à classe NP

Circuito Hamiltoniano é NP-Completo

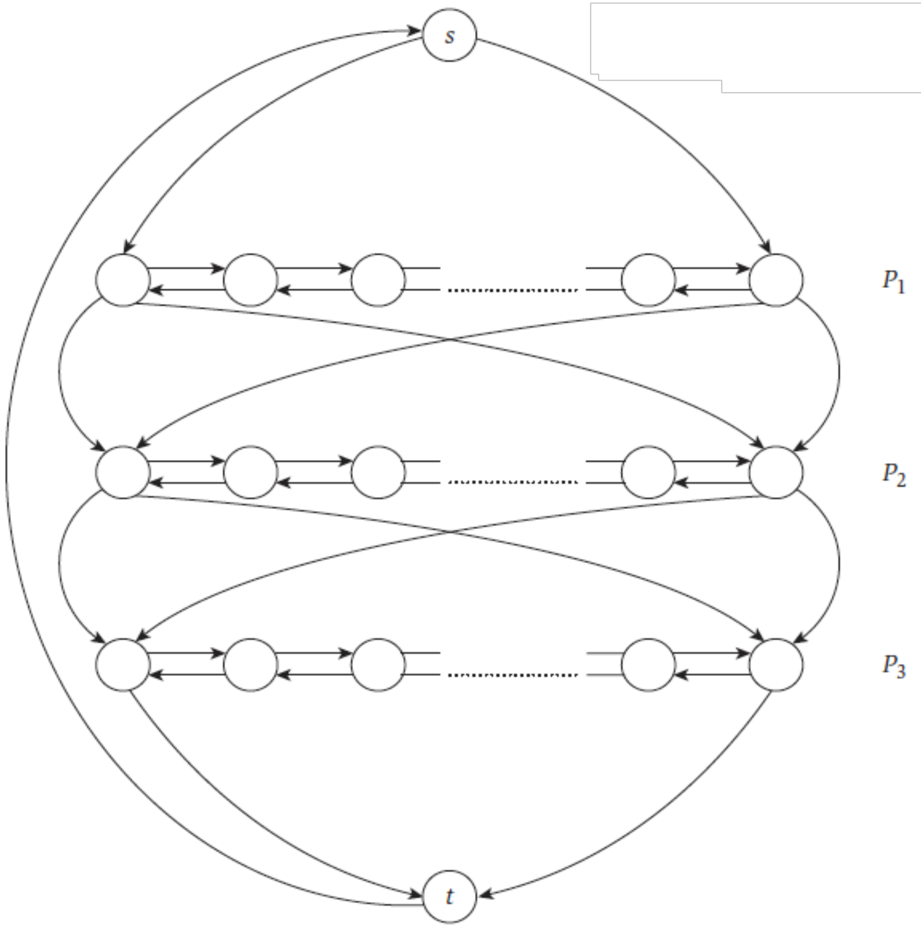
- Para mostrar que o problema é NP-difícil, vamos reduzir 3SAT a ele
- Para cada variável x_i da função, criamos um caminho (direcionado) com b=3k+3 vértices
  - Criamos arestas de ida e volta entre cada par de vértices v_{ij} e v_{ij+1}
- Além disso, criamos arestas ligando os terminais de um caminho aos terminais do próximo
  - Ou seja, criamos arestas de v_{i1} para v_{i+11} e v_{i+1b}, e também de v_{ib} para v_{i+11} e vi+1b
- Logo, podemos percorrer um caminho Pi da esquerda para direita, ou vice-versa, e, depois, escolher independentemente como percorrer o caminho P_{i+1}
- Como cada caminho é referente a uma variável, temos duas possíveis formas de percorrer seu caminho

Circuito Hamiltoniano é NP-Completo

- Podemos entender as duas escolhas esq-dir ou dir-esq como valores Booleanos
- Mas isso não se relaciona diretamente com circuito hamiltoniano
- Assim, adicionamos outros dois vértices s e t ao grafo
- O vértice s é conectado a v_{11} e v_{1b}
- Os vértices v_{1n} e v_{nb} são conectados a t
- O vértice t é conectado a s

Circuito Hamiltoniano é NP-Completo

- O grafo resultante seria algo como ao lado
- Note que existem 2^n possíveis circuitos hamiltonianos nesse grafo
- O circuito pode começar em t e depois seguir para s
- Para cada caminho P_i, podemos escolher ir da esquerda para direita, ou o contrário



Circuito Hamiltoniano é NP-Completo

- Esse grafo por si só não garante que a função é satisfatível sse existe circuito hamiltoniano
- O problema é que não há restrições relativas às cláusulas
- Assim, adicionamos um novo vértice c_j para cada cláusula da função
- Como uma cláusula é satisfeita se pelo menos um de seus literais for 1 (e mapeamos os valores Booleanos para direções dos caminhos da variáveis), criamos uma aresta de v_{i3j} para c_j e outra de c_j para v_{i3j+1} se x_i aparece em C_j
    - Se x_i aparecer negado em C_j, então criamos arestas de v_{i3j+1} para c_j e de c_j para v_{i3j}
- Essa modelagem diz que, para satisfazer Cj através de xi, devemos percorrer o caminho Pi da esquerda para direita e, eventualmente, incluindo cj no percurso ou o contrário se formos satisfazer através de $\overline{x_i}$

Circuito Hamiltoniano é NP-Completo

- Ou seja, se $C_1 = x_1 \vee \overline{x_2} \vee x_3$ um dos três caminhos, P_1, P_2 ou P_3, deve ser percorrido na direção "correta" para que o circuito seja completado
- Logo, se a função for satisfatível, iniciamos o circuito em t e percorremos cada caminho Pi conforme a atribuição de valor à variável x_i
  - Isso resulta em um circuito hamiltoniano
- Analogamente, suponha que exista um circuito hamiltoniano
- Se c_j for visitado passando pela aresta v_{i3j}, então o circuito deve deixar c_j através da aresta v_{i3j+1}, caso contrário, o vértice v_{i3j+2} não seria visitado
- Se c_j for visitado passando pela aresta v_{i3j+1}, então o circuito deve sair de c_j através de v_{i3j}, pela mesma razão anterior
- Isso implica que, para completar o circuito hamiltoniano, pelo menos um dos caminhos referentes aos literais de C_j teve que ser percorrido na direção correta
- Dessa forma, podemos usar essas direções como atribuições de valores às variáveis, garantindo que todas as cláusulas serão satisfeitas
- Portanto, Circuito Hamiltoniano é NP-Completo

Problema do Caixeiro Viajante é NP-Completo

- O problema do caixeiro viajante (TSP) consiste em, dado um grafo completo com n vértices e pesos nas arestas, encontrar um circuito que passe por todos os vértices uma única vez, retornando ao inicial, com o menor custo possível
- O problema de decisão associado consiste em limitar o custo desse circuito a um valor máximo B
- O problema pertence à classe NP, porém não vamos demonstrar
- Podemos reduzir o Circuito Hamiltoniano a TSP para mostrar que ele é NP-difícil
- Para isso, basta criar um novo grafo completo com os mesmos vértices do inicial, atribuir peso 1 às arestas existentes, e peso 2 às não existentes
- Escolhemos $B=|V|$
- A demonstração é direta nesse caso

Leitura

- Seção 12.7 (Jeff Erickson)
- Seção 8.5 (Kleinberg e Tardos)
- Seção 17.5 (Goodrich e Tamassia)