Problemas NP-completos: 3SAT e CLIQUE

Teoria da Computação

Professor: Rennan Dantas

Universidade Federal do Ceará
Campus de Crateús

15 de junho de 2023

Introdução

- Antes de exibir uma redução de tempo polinomial, introduzimos 3SAT, um caso especial do problema da satisfazibilidade no qual todas as fórmulas estão em uma forma especial
- Um literal é uma variável booleana ou uma variável booleana negada, como x ou $\overline{x}$
- Uma cláusula é uma fórmula composta de vários literais conectados por $\vee$s, como em

$$(x_1 \vee \overline{x_2} \vee \overline{x_3} \vee x_4) \wedge (x_3 \vee \overline{x_5} \vee x_6) \wedge (x_3 \vee \overline{x_6})$$

NP-completude: 3-SAT e CLIQUE

Introdução

- Uma fnc-fórmula é uma 3fnc-fórmula se todas as cláusulas tiverem três literais, como em

$$(x_1 \vee \overline{x_2} \vee \overline{x_3}) \wedge (x_3 \vee \overline{x_5} \vee x_6) \wedge (x_3 \vee \overline{x_6} \vee x_4) \wedge (x_4 \vee x_5 \vee x_6)$$

NP-completude: 3-SAT e CLIQUE

Introdução

- Seja 3-SAT $= \{< \phi > | \phi$ é uma 3fnc-fórmula satisfazível$\}$
- Em uma fnc-fórmula satisfazível, cada cláusula deve conter pelo menos um literal a que é atribuído o valor 1
- O teorema a seguir apresenta uma redução de tempo polinomial do problema 3SAT para o problema CLIQUE

Teorema

3-SAT é redutível em tempo polinomial a CLIQUE

Prova

- Seja ϕ uma fórmula com k cláusulas tal como

$$\phi = (a_1 \vee b_1 \vee c_1) \wedge (a_2 \vee b_2 \vee c_2) \wedge \ldots \wedge (a_k \vee b_k \vee c_k)$$

- A redução f gera a cadeia $< G, k >$, onde G é um grafo não-direcionado definido da seguinte forma
- Os nós em G são organizados em k grupos de três nós cada um chamados de triplas, $t_1, \ldots, t_k$
- Cada tripla corresponde a uma das cláusulas em ϕ, e cada nó em uma tripla corresponde a um literal na cláusula associada
- Rotule cada nó de G com seu literal correspondente em ϕ

NP-completude: 3-SAT e CLIQUE

Prova

- As arestas de G conectam todos os pares de nós em G, exceto dois tipos de pares
- Nenhuma arestas está presente entre nós na mesma tripla e nenhuma aresta está presente entre dois nós com rótulos contraditórios, como x_2 e $\overline{x_2}$
- A figura no quadro ilustra essa construção quando $\phi = (x_1 \vee x_1 \vee x_2) \wedge (\overline{x_1} \vee \overline{x_2} \vee \overline{x_2}) \wedge (\overline{x_1} \vee x_2 \vee x_2)$

Prova

- Agora demonstramos porquê essa construção funciona
- Mostramos que ϕ é satisfazível sse G tem um k-clique
- Suponha que ϕ tenha uma atribuição que a satisfazibilidade
- Nessa atribuição, pelo menos um literal é verdadeiro em cada cláusula
- Em cada tripla de G, selecionamos um nó correspondendo a um literal verdadeiro na atribuição que satisfaz a fórmula
- Se mais que um literal for verdadeiro em uma cláusula específica, escolhemos um deles arbitrariamente
- Os nós que acabam de ser selecionados formam uma k-clique

NP-completude: 3-SAT e CLIQUE

Prova

- O número de nós selecionado é k, porque escolhemos um para cada uma das k triplas
- Cada par de nós selecionados é ligado por uma aresta porque nenhum par se encaixa em uma das exceções descritas anteriormente
- Eles não poderiam ser da mesma tripla, porque selecionamos somente um nó por tripla
- Eles não poderiam ter rótulos contraditórios porque os literais associados eram ambos verdadeiros na atribuição que satisfaz a fórmula
- Por conseguinte, G contém um k-clique

Prova

- Suponha que G tenha um k-clique
- Nenhum par de nós do clique ocorre na mesma tripla, porque nós na mesma tripla não são conectados por arestas
- Consequentemente, cada uma das k triplas contém exatamente um dos k nós do clique
- Atribuímos valores-verdade às variáveis de ϕ de modo que cada literal que rotula um nó do clique torne-se verdadeiro
- Fazer isso é sempre possível, pois dois nós rotulados de maneira contraditória não são conectados por uma aresta e, portanto, não podem estar ambos no clique
- Essa atribuição às variáveis satisfaz ϕ porque cada tripla contém um nó do clique e assim, cada cláusula contém um literal ao qual é atribuído VERDADEIRO
- Logo, ϕ é satisfazível

Conexão entre os problemas

- O último teorema que vimos na aula passada e o teorema que acabamos de ver nos dizem que, se CLIQUE for solúvel em tempo polinomial, o mesmo acontece com 3SAT
- À primeira vista, essa conexão entre esses dois problemas parece um tanto impressionante porque, superficialmente, eles são bastante diferentes
- Mas a redutibilidade de tempo polinomial nos permite relacionar suas complexidades
- Agora nos voltamos para uma definição que nos permitirá similarmente relacionar as complexidades de uma classe inteira de problemas

Definição de NP-completude

Uma linguagem B é NP-completa se satisfaz duas condições

1. B está em NP, e
2. toda A em NP é redutível em tempo polinomial a B

NP-completude: 3-SAT e CLIQUE

Teorema

Se B for NP-completa e $B \in P$, então P=NP

Prova

Esse teorema segue diretamente da definição de redutibilidade de tempo polinomial

NP-completude: 3-SAT e CLIQUE

Teorema

Se B for NP-completa e B $\leq_P$ C para C em NP, então C é NP-completa

Prova

- Já sabemos que C está em NP, portanto devemos mostrar que toda A em NP é redutível em tempo polinomial a C
- Como B é NP-Completa, toda linguagem em NP é redutível em tempo polinomial a B e B, por sua vez, é redutível em tempo polinomial a C
- Reduções em tempo polinomial se compõe; ou seja, se A for redutível em tempo polinomial a B e B for redutível em tempo polinomial a C, então A é redutível em tempo polinomial a C
- Logo, toda linguagem em NP é redutível em tempo polinomial a C

Corolário

3SAT é NP-completa

NP-completude: 3-SAT e CLIQUE

Prova

- Obviamente 3SAT está em NP, portanto somente precisamos provar que todas as linguagens em NP se reduzem a 3SAT em tempo polinomial
- Uma maneira de fazê-lo é mostrar que SAT se reduz em tempo polinomial a 3SAT
- Vamos converter a fórmula do problema SAT em uma fórmula com três literais por cláusula
- Em cada cláusula que correntemente tem um ou dois literais, replicamos um dos literais até que o número total seja três
- Em cada cláusula que tem mais de três literais, a dividimos em várias cláusulas e acrescentamos variáveis extras para preservar a satisfazibilidade ou não-satisfazibilidade da original

Prova

- Por exemplo, substituímos a cláusula $(a_1 \vee a_2 \vee a_3 \vee a_4)$, na qual cada a_i é um literal, pela expressão de duas cláusulas $(a_1 \vee a_2 \vee z) \wedge (\overline{z} \vee a_3 \vee a_4)$, na qual z é uma nova variável
- Se alguma valoração de a_is satisfaz a cláusula original, podemos encontrar alguma valoração de z de modo que as duas novas cláusulas sejam satisfeitas
- Em geral, se a cláusula contém l literais,

$$(a_1 \vee a_2 \vee ... \vee a_l)$$

- podemos substituí-la pelas l-2 cláusulas

$$(a_1 \vee a_2 \vee z_1) \wedge (\overline{z_1} \vee a_3 \vee z_2) \wedge (\overline{z_2} \vee a_4 \vee z_3) \wedge ... \wedge (\overline{z_{l-3}} \vee a_{l-1} \vee a_l)$$

- Podemos facilmente verificar que a nova fórmula é satisfazível sse a fórmula original o era, portanto, a prova está completa

Próxima Aula

O que vem por aí?

- Cobertura de vértices
- Caminho Hamiltoniano
- Soma de subconjuntos

Problemas NP-completos: 3SAT e CLIQUE

Teoria da Computação

Professor: Rennan Dantas

Universidade Federal do Ceará
Campus de Crateús

15 de junho de 2023