Universidade de Brasília
Instituto de Ciências Exatas

Departamento de Ciência da Computação

CIC 117536 - Projeto e Análise de Algoritmos

Terceira Prova

Turma: B

NP-completude

Prof. Flávio L. C. de Moura

6 de dezembro de 2018

1. **(2.5 pontos)** O problema 2-SAT tem como instâncias as fórmulas lógicas formadas por conjunções de disjunções de até dois literais, onde um literal é uma variável booleana ou a negação de uma variável booleana. Por exemplo, a expressão a seguir é uma instância de 2-SAT:

$$(x_1 \vee \neg x_2) \wedge (\neg x_1 \vee \neg x_3) \wedge (x_1 \vee x_2) \wedge x_3$$

Prove que 2-SAT $\in$ P.

Solução.Podemos notar que uma disjunção de apenas duas variáveis equivale a dizer que a negação de uma variável implica que a outra seja verdadeira, i.e. $(x_1 \vee x_2) \equiv (\neg x_1 \implies x_2) \wedge (\neg x_2 \implies x_1)$, o que pode ser facilmente percebido da tabela-verdade da disjunção.

Desse modo, podemos reescrever o problema 2-SAT da Forma Normal Conjuntiva para a Formal Normal Implicativa, trocando as conjunções do problema pelas implicações equivalentes. Com isso, podemos notar que caso:

$$x_1 \implies \neg x_1$$

Precisamos que x_1 seja falso, visto que x_1 sendo verdadeiro criaria uma contradição na qual teriamos tanto x_1 quanto $\neg x_1$ sendo verdadeiros. Já no caso:

$$\neg x_1 \implies x_1$$

Precisamos que $\neg x_1$ seja falso, de modo análogo ao caso anterior. Assim, teremos problemas apenas se tivermos:

$$(x_1 \implies \neg x_1) \wedge (\neg x_1 \implies x_1)$$

Caso no qual precisaríamos de x_1 sendo falso e verdadeiro, o que significa que a expressão booleana não pode ser satisfeita.

Se construirmos um grafo de implicações a partir da Forma Normal Implicativa do problema, notando que $(x_1 \implies x_2) \wedge (x_2 \implies x_3)$ significa que $(x_1 \implies x_3)$, podemos chegar a conclusão de que se x_1 e $\neg x_1$ existirem no mesmo componente fortemente conexo do grafo de implicações, a expressão booleana do problema em questão não pode ser satisfeita.

Assim, usando um algoritmo ótimo para encontrar os componentes fortemente conexos, como o algoritmo de Kosaraju ou o algoritmo de Tarjan, podemos resolver o 2-SAT em $O(n+m)$, onde n é o número de váriaveis e m o número de cláusulas da Forma Normal Conjuntiva do problema. Logo, temos que 2-SAT $\in$ P.

2. **(2.5 pontos)** Em aula, assumimos que SAT é um problema NP-completo (Teorema de Cook-Levin), e a partir deste fato mostramos que 3-SAT e CLIQUE também são problemas NP-completos. As reduções foram feitas de acordo com o seguinte diagrama:

$$\begin{tikzcd} SAT \arrow[d] \\ 3\text{-}SAT \arrow[d] \\ CLIQUE \end{tikzcd}$$

Um ciclo Hamiltoniano é um ciclo simple que visita cada vértice de um grafo exatamente uma vez. Considere o problema de decisão HAM-CYCLE que pergunta se um dado grafo (não-dirigido) G possui um ciclo Hamiltoniano. Mostre que HAM-CYCLE é um problema NP-completo. Sua solução deve ser construída a partir de SAT, 3-SAT ou CLIQUE. Caso, você não veja como reduzir diretamente HAM-CYCLE a partir destes, mas sabe como fazê-lo a partir de um certo problema Q então inicialmente mostre que Q é NP-completo a partir de SAT, 3-SAT ou CLIQUE, e assim por diante. Digamos que você não saiba como mostrar que Q é NP-completo diretamente a partir de SAT, 3-SAT ou CLIQUE, mas você sabe como fazê-lo a partir de outro problema Q', e também sabe como mostrar que Q' é NP-completo a partir de 3-SAT, por exemplo. Então o diagrama correspondente à sua solução seria:

$$\begin{tikzcd} SAT \arrow[d] \\ 3\text{-}SAT \arrow[d] \arrow[r] & Q' \arrow[r] & Q \arrow[r] & \text{HAM-CYCLE} \\ CLIQUE \end{tikzcd}$$

E todas as reduções (de 3-SAT para Q', de Q' para Q e de Q para HAM-CYCLE) devem ser detalhadas na sua solução.

Solução.Primeiramente, podemos fácilmente notar que HAM-CYCLE é NP. Dado um caminho no grafo, basta percorre-lo e verificar se o vértice de partida é o mesmo do de destino e se, durante o percurso, algum vértice foi visitado mais de uma vez. Esse algoritmo de verificação pode ser executado em tempo polinominal e, com isso, HAM-CYCLE é NP.

Agora, vamos mostrar que com uma fórmula 3-FNC podemos criar um grafo G que tem um Caminho Hamiltoniano se, e somente se, a fórmula puder ser satisfeita.

Para isso, para cada váriavel da fórmula construímos um gadget de 3k+1 vértices, onde k é o número de cláusula na fórmula. Nesse gadget um vértice acima, uma cadeia de vértices no centro e um vértice abaixo. O vértice de cima leva às duas pontas da cadeia do centro e às duas pontas da cadeia do centro levam ao vértice de baixo. Os vértice da cadeia, por sua vez levam ao anterior e ao próximo, de modo que seja possível percorrer o gadget da esquerda para direita e da direita para a esquerda.

Assim, temos um caminho no qual a váriavel que o gadget representa é verdadeira e uma caminho no qual a váriavel que o gadget representa é falsa. Em um, partindo-se do vértice do topo, vai-se para a esquerda, percorre-se a cadeia da esquerda para a direita e desce-se para o vértice de baixo. No outro, partindo-se do vértice do topo, vai-se para a direita, percorre-se a cadeia da direita para a esquerda e desce-se para o vértice de baixo.

Com isso, concatena-se o vértice de baixo de uma váriavel com o vértice de cima de outra de modo que no final so haja um grafo, no qual temos o vértice de topo de uma váriavel e o de baixo de outra. Além disso, cria-se vértices que representam as cláusulas, que devem ser ligados nos gadgets das variáveis de modo a representar a existência da váriavel ou da sua negação na cláusula.

Assim, tem-se um grafo no qual para que exista um Caminho Hamiltoniano, deve ser possível satisfazer a fórmula 3-FNC que gerou esse grafo, visto que para se atravessar as cadeias de cada váriavel no grafo é necessário que a fórmula possa ser satisfeita.

Portanto, temos que o Caminho Hamiltoniano é NP-Difícil. Com isso, podemos notar que, dado um grafo G para qual temos o problema do Caminho Hamiltoniano, se adicionarmos um vértice e o ligarmos a todos os vértices de G, caímos em um problema do Ciclo Hamiltoniano. Logo, HAM-CYCLE é NP-Díficil e, assim, NP-Completo.

3. **(2.5 pontos)** Considere o seguinte jogo em um grafo (não-dirigido) G, que inicialmente contém 0 ou mais bolas de gude em seus vértices:

um movimento deste jogo consiste em remover duas bolas de gude de um vértice $v \in G$, e adicionar uma bola a algum vértice adjacente de v. Agora, considere o seguinte problema: Dado um grafo G, e uma função $p(v)$ que retorna o número de bolas de gude no vértice v, existe uma sequência de movimentos que remove todas as bolas de G, exceto uma? Mostre que este problema é NP-completo. A mesma observação feita no exercício anterior vale aqui: a prova deve ser feita a partir de problemas que provamos serem NP-completos, e reduções intermediárias, caso existam, devem ser incluídas na solução.

Solução.Primeiro provamos que o problema é NP. Dado que uma sequência de movimentos seja executada, podemos verificar todos os vértices de G com a função $p(v)$ e saber se a sequência executada é uma solução. Sendo $p(v)$ executada em tempo polinomial, temos que esse algoritmo de verificação é executado em tempo polinomial e, logo, o problema é NP.

Em seguida provamos que o problema é NP-Difícil. É fácil notar que ao realizarmos o movimento do jogo estamos fazendo o equivalente de tirar do jogo uma das bolas e mover a outra para um vértice adjacente. Assim, é simples de se perceber que se seguirmos no próximo movimento para o vértice para o qual movemos a bola anterior, podemos tirar mais uma bola do jogo e mover novamente a mesma bola. Note que a bola que está sendo movida jamais precisará ser retirada do jogo.

Assim, podemos notar que dado um Ciclo Hamiltoniano no grafo, basta que se siga esse ciclo um número determinado de vezes que teremos uma solução do problema. Esse número pode ser determinado pela função $p(v)$, visto que a cada ciclo é retirada uma bola de todos os vértices, podemos fazer $p(v)$ em todos os vértices antes de começar a percorrer o ciclo e pegar o maior valor retornado como número de vezes a se percorrer o ciclo.

Com isso, temos que esse problema pode ser reduzido ao HAM-CYCLE do exercício anterior, tornando-o NP-Difícil. Portanto, temos que o problema da questão é NP-Completo.

4. **(2.5 pontos)** Uma fórmula booleana em *forma normal conjuntiva com disjunção exclusiva (FNCX)* é uma conjunção de diversas cláusulas, e cada cláusula é uma disjunção exclusiva (XOR) de diversos literais. Lembre-se que a disjunção exclusiva é dada por:

a	b	$a \oplus b$
V	V	F
V	F	V
F	V	V
F	F	F

O problema FNCX-SAT pergunta se uma dada fórmula em FNCX é satisfatível. Mostre que o problema FNCX-SAT está em P, ou então que FNCX-SAT é NP-completo. No último caso, a mesma observação feita nos dois exercícios anteriores vale aqui: a prova deve ser feita a partir de problemas que provamos serem NP-completos, e reduções intermediárias, caso existam, devem ser incluídas na solução.

Solução.A disjunção exclusiva pode ser vista como uma adição em módulo de dois. Assim, dada uma fórmula em FNCX, podemos transformá-la em um sistema linear de equações, e.g. $(x_1 \oplus \neg x_2) \wedge (x_1 \oplus x_3) \wedge (x_2 \oplus x_3)$ pode ser transformado em:

$$x_1 + x_2 = 0 \mod 2$$

$$x_1 + x_3 = 1 \mod 2$$

$$x_2 + x_3 = 1 \mod 2$$

Visto que $\neg x = (x \oplus 1)$. Assim, podemos resolver o sistema de equações em tempo cúbico por Eliminação Gaussiana e, com isso, temos FNCX-SAT $\in$ P.