Gabarito - Segunda lista de Complexidade 2023/3

Diego Amaro   Guilherme Bridi   Oscar Martins

15 de novembro de 2023

Questão 1. Para o para I e I'.

(a) Por I, conseguimos encontrar a maior clique C de G. Desse modo, basta encontrar a ordem n_C de C. Assim, a resposta de I' é "Sim" para $k \in \{1, \cdots, n_C\}$ e "Não" caso contrário.

(b) Guardaremos o resultado de I' para $k \in \{1, \cdots, n\}$. Assim, conseguimos descobrir a ordem da maior clique de G, digamos que seja n_C. Agora, executamos I' para todo subgrafo de G com n_C vértices até encontrarmos uma resposta "Sim" para $k = n_C$. Por fim, retornamos o subgrafo de G que deu resposta "Sim" para I'.

$\square$

Questão 2. **Problema II'.** Por definição, G possui uma clique C se, e somente se, C é um conjunto independente em $\overline{G}$. Assim, dado um grafo G e um inteiro k, basta aplicar o algoritmo $A_{I'}$ em $\overline{G}$.

Problema III'. G possui uma cobertura de S de tamanho maior ou igual a k se, e somente se, $V(G) \setminus S$ é um conjunto independente de tamanho menor ou igual a $n - k$. Portanto, $V(G) \setminus S$ é independente em G se, e somente se, $V(G) \setminus S$ é clique em $\overline{G}$. Assim, basta aplicar $A_{I'}$ em $\overline{G}$ com o inteiro $n - k$. □

Questão 3. **Primeiro, mostraremos que k-Clique está em $\mathcal{NP}$.** Para um dado grafo $G = (V, E)$ e uma clique C, usa-se $V(C) \subset V(G)$ como certificado. A verificação se $V(C)$ é, realmente, uma clique pode ser realizada em tempo polinomial ao verificar se $uv \in E(G)$ para cada par $u, v \in V(C)$, uma vez que o número máximo de arestas em um grafo é $\binom{n}{2}$.

Formalmente, definimos uma variável booleana denominada Verificador, que para cada par de vértices $x, y \in V(C)$ retornará VERDADEIRO caso haja uma aresta entre x e y e em caso contrário, retornará FALSO.

Algorithm 1 Certificado

```
Verificador = VERDADEIRO
para cada par de vértice {x,y} de V(C) faça
|   Verifique a existência da aresta xy
|   se não existe a aresta xy então
|   |   Verificador = FALSO
|   fim
fim
se Verificador = VERDADEIRO então
|   retorna A solução está correta
fim
se Verificador = FALSO então
|   retorna A solução está incorreta
fim
```

Resta mostrarmos que esse problema é também $\mathcal{NP}$-Difícil.

Solução usando o problema usando a redução a partir do k-StableSet (II') Vamos aplicar o lema visto em sala utilizando o problema II', que é sabidamente $\mathcal{NP}$-completo, e usar o fato de que II' pode ser reduzido a I' para concluir a $\mathcal{NP}$-completude.

Observe que neste problema a instância recebida é um grafo $G = (V, E)$ e um número natural k. Por conveniência, podemos converter cada instância no par formado por $\overline{G}$ e o natural k, onde $\overline{G}$ é o complementar de G.

Nitidamente, transformar G em $\overline{G}$ é um procedimento com tempo polinomial no tamanho da entrada, já que precisamos apenas repetir os vértices de G, apagar todas as arestas originalmente em G e construir arestas para vértices não adjacentes em G.

II' pode ser reduzido ao problema I', pois: Se G possui uma clique de tamanho k, então existem k vértices em G, em que cada um desses k vértices compartilha uma aresta com os demais, portanto, essas arestas não pertencem a $E(\overline{G})$. Como esses k vértices não adjacentes entre si em $\overline{G}$, então, formam um conjunto independente de vértices de tamanho k.

Por outro lado, se $\overline{G}$ possui um conjunto independente de vértices de tamanho k, então não há arestas entre cada par de vértices pertencente a este conjunto.

Como $G = \overline{\overline{G}}$, os vértices que pertencem ao conjunto independente de $\overline{G}$ compartilham arestas entre si em G, e portanto, são vértices adjacentes uns aos outros. De onde concluímos que G tem uma clique de tamanho k.

Como II' é $\mathcal{NP}$-completo, garantimos que o problema I' é também $\mathcal{NP}$-Completo.

Outra solução para provar que o problema é $\mathcal{NP}$-difícil:

Agora, vamos reduzir reduzir 3SAT a k-CLIQUE. O algoritmo de redução começa com uma instância de 3SAT. Seja $\phi = C_1 \wedge \cdots \wedge C_k$ uma expressão booleana na forma normal conjuntiva com três variáveis booleanas em cada cláusula C_r, em que $1 \leq r \leq k$. Para $r = 1, 2, \cdots, k$, cada cláusula C_r, tem exatamente três literais distintos ℓ_1^r, ℓ_2^r e ℓ_3^r. Deve-se construir um grafo G tal que ϕ é satisfazível se, e somente se, G tem uma clique de tamanho k.

O grafo $G = (V, E)$ é construído da seguinte forma: Para cada cláusula $C_r = (\ell_1^r \vee \ell_2^r \vee \ell_3^r)$ em ϕ, coloca-se uma tripla de vértices v_1^r, v_2^r e v_3^r em V. Coloca-se uma aresta entre dois vértices v_i^r e v_j^s se ambos atendem a: i) v_i^r e v_j^s estão em triplas diferentes; e ii) seus literais correspondentes são consistentes, isto é, ℓ_i^r não é a negação de ℓ_i^s.

Deve-se mostrar que esta transformação de ϕ em G e uma redução. Supõe-se que ϕ tem uma atribuição satisfazível. Então, cada cláusula C_r contém, pelo menos, um literal ℓ_i^r que é estabelecido como 1 e cada literal corresponde ao vértice v_i^r. Obtendo-se um desses literais valorados como verdadeiro de cada cláusula, fornece-se um conjunto C de k vértices.

Afirma-se que C e uma clique. Para cada dois vértices $v_i^r v_j^s \in C$, em que $r \neq s$, ambos correspondendo a literais $\ell_i^r \ell_i^s$, são mapeados para 1 pela atribuição de satisfabilidade dada e, assim, os literais não podem ser complementos. Assim, pela construção de G, $v_i^r v_j^s \in E(G)$.

De maneira reversa, suponha que G tem uma clique C de tamanho k. Nenhuma aresta em G conecta vértices na mesma tripla e, então, C contém exatamente um vértice por tripla. Pode-se atribuir 1 para cada literal ℓ_i^r, tal que $v_i^r \in C$ sem risco de atribuir 1 tanto para um literal e seu complemento, ja que G não contém aresta entre literais inconsistentes. Cada clausula é satisfeita e, então, ϕ e satisfeita. Quaisquer variáveis que não correspondem a vértices na clique podem ser estabelecidas arbitrariamente. $\square$