

Tarea 3: Reducciones polinomiales, P vs. NP

CC3102 - Teoría de la Computación
Departamento de Ciencias de la Computación
UNIVERSIDAD DE CHILE

Gabriel De La Parra

15 de diciembre de 2015

Índice

1. Problema 1	3
2. Problema 2	3
3. Problema 3	4
4. Problema 4	4

1. Problema 1

Un lenguaje A es no trivial si $A \neq \emptyset$ y $A \neq \Sigma^*$. Demuestre que si $\mathcal{P} = \mathcal{NP}$ entonces todo lenguaje no trivial en \mathcal{NP} es \mathcal{NP} -Completo.

Por definición, para que un lenguaje sea \mathcal{NP} -Completo, se debe demostrar que:

- A está en \mathcal{NP} , ya en el enunciado.
- Existe una reducción polinomial de A a un lenguaje 'más difícil' B , tal que si $w \in A$, $f(w) \in B$ y si $w' \notin A$, $f(w') \notin B$.

A partir del enunciado:

- Si $\mathcal{P} = \mathcal{NP}$, todo problema en \mathcal{NP} se puede resolver en tiempo polinomial de forma determinista. Es decir, existe una MT que puede resolver el problema de \mathcal{NP} en tiempo polinomial.
- Si $A \neq \emptyset$ y $A \neq \Sigma^*$, entonces existen elementos que $w \in A$ y elementos $w' \notin A$. Estos elementos se puede utilizar para hacer una reducción. En el caso que $A = \emptyset$ o $A = \Sigma^*$, no habrían valores w dentro/fuera de A que sirvan para hacer una reducción.

De los puntos anteriores, se debe construir una reducción polinomial a B . Si B es \mathcal{NP} -Completo, también es \mathcal{NP} , y por la asunción, también es \mathcal{P} , por lo que se puede resolver en tiempo polinomial.

Ahora bien, se puede construir una máquina M que decida B y construir una máquina M' con esta, que decida A .

- Si M acepta $x \in B$, M' acepta $w \in A$.
- Si M rechaza $x' \notin B$, M' rechaza $w' \notin A$.

La reducción se puede construir en tiempo polinomial. Como esta reducción existe, $A \in \mathcal{NP}$ -Completo.

2. Problema 2

Describa el problema INDSET como un problema de decisión de lenguajes y demuestre que es \mathcal{NP} -Completo via una reducción desde CLIQUE.

Se desea demostrar que existe una reducción de INDSET a CLIQUE en tiempo polinomial:

$$\text{CLIQUE} \leq_p \text{INDSET}$$

La definición entregada para INDSET se puede entender como un 'inverso' de CLIQUE. En CLIQUE, hay que encontrar k -nodos estén conectados todos con todos. En INDSET hay que encontrar k -nodos que no estén conectados entre ellos, "ninguno con ninguno", como analogía.

La reducción se hace a partir de un problema en CLIQUE.

A partir de un grafo $G' = (V', E')$ con CLIQUE- k' , se debe construir un grafo $G = (V, E)$ tal que G' tiene un CLIQUE- $k' \iff G$ tiene un INDSET- k . Esta construcción debe correr en tiempo polinomial.

Se toman de esta forma:

- Los nodos son los mismos en CLIQUE e INDSET
- Para cada par de nodos, si hay un arco entre ellos en $E' \in G'$, no debe haber un arco entre ellos en $E \in G$. De la misma forma, si no hay un arco entre ellos en G' , se agrega un arco entre ellos en G
- El tamaño k' del CLIQUE será el mismo tamaño k del INDSET

Como lo anterior se puede realizar proporcionalmente al tamaño de E' , el algoritmo se puede ejecutar en tiempo polinomial. La demostración de que la reducción funciona para ambos lados es intuitiva. Al hacer el cambio, si existían las condiciones para CLIQUE, deben estar para INDSET, de la misma forma, si están para INDSET, deben estar para CLIQUE.

Como existe una reducción en tiempo polinomial, INDSET es \mathcal{NP} -Completo

3. Problema 3

Demuestre que A_{MT} es \mathcal{NP} -duro. Pista: Reduzca SAT a A_{MT} ; ¿Pudiera A_{MT} ser \mathcal{NP} -Completo? Justifique su respuesta.

Un problema es \mathcal{NP} -duro si se puede hacer una reducción polinomial a un problema \mathcal{NP} -Completo. Esto quiere decir que es, por lo menos, tan difícil como un problema \mathcal{NP} -Completo. Esto no implica que el problema deba estar en \mathcal{NP} ni que sea un problema de decisión.

Para este problema, se solicita hacer la reducción polinomial a SAT. Si existe un algoritmo para resolver SAT, este se puede usar para decidir A_{MT} a través de una reducción en tiempo polinomial.

Sea M es la MT que decide SAT. Sea \vec{a} la entrada de esa máquina M , se puede construir un M' con M , tal que si $M(\vec{a})$ acepta, M' se detiene y acepta la entrada. Esto hasta encontrar un \vec{a} que cumpla SAT. Si no existe tal condición, la máquina M' loopeará. Esta conversión de M' a M toma tiempo polinomial.

Lo que no se ha determinado es si A_{MT} es \mathcal{NP} -Completo. Para esto habría que probar que la máquina en la entrada de A_{MT} siempre se detiene y decide y que por lo tanto estaría en \mathcal{NP} , lo cual se ha demostrado anteriormente que no se cumple. De lo anterior, $A_{MT} \notin \mathcal{NP}$ -Completo.

4. Problema 4

Demuestre que CIRCUITOHAM es \mathcal{NP} -Completo; Demuestre que CIRCUITOHAMND es \mathcal{NP} -Completo

Para la primera parte, es necesario demostrar que:

- (1) CIRCUITOHAM es \mathcal{NP} ;
- (2) Existe una reducción polinomial de CIRCUITOHAM a CAMINO HAM.

Sea $G = \{V, E\}$ el grafo que define CIRCUITOHAM. Para (1), bastaría con tener un certificado c . Se puede construir un verificador V . V toma c y prueba si desde el primer nodo en c existen los arcos en E que satisfagan CIRCUITOHAM. Esto ocurre en orden polinomial proporcional al tamaño de E .

Para (2), intuitivamente, a partir de un grafo $G \in \text{CAMINO HAM}$, se podría crear un grafo G' , en el que se agrega un nodo y se conecta hacia/desde todos los otros nodos. Si existe un CAMINO HAM en el grafo G , existe ahora un CIRCUITOHAM en el grafo G' . Análogamente, si no existía un CAMINO HAM en G , no existirá un CIRCUITO en G' . Ahora bien, si existe un CIRCUITOHAM en G' , significa que debe haber un CAMINO HAM en G .

Para la segunda parte del problema, se pide, a partir de un grafo no dirigido H , en el que hay un CIRCUITOHAMND, realizar una reducción a CIRCUITOHAM.

La verificación de \mathcal{NP} se puede hacer igual que en el caso anterior, verificando un certificado, que deberá tener conexiones entre todos los nodos del grafo H sin repeticiones.

Para la reducción se sugiere agregar una tripleta de nodos por cada nodo en $G \in \text{CIRCUITOHAM}$. El nuevo grafo H tiene:

- Tres nodos por cada nodo de G . Sean $v_{i,0}, v_{i,1}, v_{i,2}$ estos nodos.
- Arcos no dirigidos entre $\{v_{i,0}, v_{i,1}\}, \{v_{i,1}, v_{i,2}\}$.
- Arcos no dirigidos según la dirección original, de tal forma que si entran a la tripleta, lo hacen a través de $v_{i,0}$ y si salen, a través de $v_{i,2}$.

Al realizar esta construcción, se debe cumplir que se visiten todos los nodos del nuevo grafo H . Si existe un CIRCUITOHAM G . Esto es valido para los dos lados. Si existe una solución para G , esta debe visitar los 3 nodos en H . De la misma forma, si se visitan todos los nodos en H , quiere decir que en G se estarían visitando todos los nodos según la dirección de los arcos.

Se puede construir un CIRCUITOHAMND H con orden polinomial, considerando los 3 nodos y los nuevos arcos que se trazan como se describió anteriormente.

Como se puede verificar que es posible realizar la reducción y que adicionalmente CIRCUITOHAMND es en \mathcal{NP} , se concluye que CIRCUITOHAMND es \mathcal{NP} -Completo.