

Complejidad Computacional

Tarea 2.2

Karla Adriana Esquivel Guzmán
Andrea Itzel González Vargas
Luis Pablo Mayo Vega
Carlos Gerardo Acosta Hernández

Entrega: 25/04/17
Facultad de Ciencias UNAM

Ejercicios

1. Demuestra que si $P = NP$, entonces existe un lenguaje EXP que requiere circuitos de tamaño $2^n/n$.

Demostración: Por el teorema de Meyer (teorema 6.20 del libro) Implica que si $P = NP$ entonces $Exp \not\subseteq P/Poly$.

Entonces como $Exp \notin P/Poly$ por lo tanto debe existir un lenguaje $L \in EXP$ que requiere un circuito de tamaño $2^n/n$.

2. Describe un circuito NC para calcular el producto de dos matrices de $n \times n$.

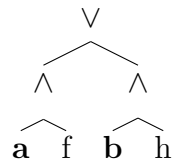
Demostración: Primero veamos el caso para matrices 2×2

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

$$B = \begin{pmatrix} f & g \\ h & i \end{pmatrix}$$

$$AB = \begin{pmatrix} af + bh & ag + bj \\ cf + dh & cg + dj \end{pmatrix}$$

Para cada entrada de la matriz AB la calcularemos con el siguiente circuito



Ahora veamos para una matriz de 3×3

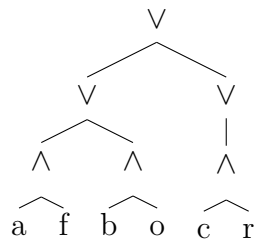
$$A = \begin{pmatrix} a & b & c \\ d & e & f \\ g & h & i \end{pmatrix}$$

$$B = \begin{pmatrix} l & m & n \\ o & p & q \\ r & s & t \end{pmatrix}$$

$$AB = \begin{pmatrix} al + bo + cr & \dots & \dots \\ \vdots & \vdots & \vdots \end{pmatrix}$$

El circuito quedaría

Para la entrada $[1,1]$ el valor de las hojas.



Podemos generalizar un árbol para las matrices de $n \times n$ que tengan profundidad de $\log_2 n$ donde las primeras $(\log_2 n) - 1$ niveles son para ir dividiendo por la mitad los OR (Sumas) que se deben hacer para obtener el valor de la entrada y el último nivel hace el AND (Multiplicación).

3. Demuestra que L es P – completo implica que $L \in NC$ si y sólo si, $P = NC$

Primero, por definición, tenemos que L es P – completo sii $L \in P$ y $\forall L' \in P$

$$L' \leq_{\logspace} L$$

Además. sabemos que $L \in NC^i$ si $\exists c > 0$ tal que L puede ser decidido por una familia de circuitos $\{C_n\}$ *logspace uniformes* con tamaño $\mathcal{O}(n^c)$ y profundidad $\mathcal{O}(\log^i n)$.

La clase de Nick es $NC = \bigcup_{i \leq 1} NC^i$

Como $\{C_n\}$ es *logspace uniforme* cumple las siguientes propiedades.

1. C_i contiene solo circuitos \wedge, \vee
2. tiene i entradas
3. tiene tamaño $\mathcal{O}(n^c)$
4. tiene profundidad $\log^c i$
5. hay una $M \in TM$ que calcula en tiempo polinomial el circuito c_i a partir de la entrada 1^i