Entrega6.pdf

patriciacorhid

Modelos Avanzados de Computacion

4º Doble Grado en Ingeniería Informática y Matemáticas

Facultad de Ciencias
Universidad de Granada

Problema de la Contratación Eficiente

Problema de la Contratación Eficiente:
Tenemos un conjunto de n deportes $D = \{d_1, ..., d_n\}$ y un conjunto de m especialistas $E = \{e_1, ..., e_m\}$ tal que cada especialista e_j conoce un conjunto de deportes $D_j \subseteq D$. Si tenemos un límite de contratos $K \leq m$, determinar si existe un subcojunto $E' \subseteq E$ de tamaño menor o igual a K y tal que para cada deporte d_i exista un especialista $e_j \in E'$ con $d_i \in D_j$.

Voy a demostrar que el *Problema de la Contratación Eficiente* (CE) es NP-Completo.
Primero vamos a comprobar que efectivamente CE $\in$ NP. Para ello diseñaremos un algoritmo no determinista con complejidad polinómica en tiempo.

Observamos primero que si existe un número de especialistas menor o igual que K que satisface el problema, en particular existirán K especialistas que lo satisfagan, ya que solo hay que contratar más especialistas de los necesarios hasta llegar a K. Es por esto que nuestro algoritmo buscará a K especialistas que satisfagan las condiciones de EC.

El algoritmo es el siguiente:

Sea una entrada del problema EC: $D = \{d_1, ..., d_n\}, E = \{e_1, ..., e_m\}, \{D_1, ..., D_m\}, K$.
Elegimos de forma no determinista K elementos de E, $\{\tilde{e}_1, ..., \tilde{e}_k\}$.
Para cada elemento elegido $\tilde{e}_i$:
    Borramos de D los deportes contenidos en $\tilde{D}_i$.
Si $D = \emptyset$ aceptamos.

Veamos que esto se hace de forma polinómica en tiempo. Sea p la longitud de la entrada, elegir K especialistas supone como mucho recorrer la entrada, así que eso tiene complejidad en tiempo $O(p)$. Para cada especialista elegido hay que ver en qué deportes es especialista y borrarlos de D, luego por cada uno de ellos hay que recorrer la entrada tantas veces como deportes esté especializado. Como el máximo de deportes que puede estar especializado es n, esto supone una complejidad en tiempo de $O(n * p) \subseteq O(p * p) = O(p^2)$ ($n \leq p$ porque necesito al menos un símbolo para codificar cada deporte). Como esto ha de hacerse $K \leq m$ veces, obtenemos una complejidad de $O(K * n * p) \subseteq O(p^3)$ (tenemos que $K \leq m$ y necesito al menos un símbolo por especialista, luego $K \leq m \leq p$).

Por tanto el problema CE pertenece a NP por tener complejidad en tiempo $O(p^3)$.

Ahora tenemos que comprobar que cualquier otro problema NP se reduce a éste. Para ello, reduciremos 3-SAT al problema CE. Como 3-SAT es NP-Completo, si se reduce a CE, éste también será NP-Completo.

Sea $U = \{p_1, ..., p_m\}$ un conjunto de símbolos preposicionales y $C = \{c_1, ..., c_n\}$ una colección de claúsulas sobre los símbolos. Construimos a partir de ésta una entrada del problema CE ($D = \{d_1, ..., d_n\}, E = \{e_1, ..., e_m\}, \{D_1, ..., D_m\}, K$) de manera que exista un subcojunto $E' \subseteq E$ de tamaño menor o igual a K tal que para cada deporte d_i exista un especialista $e_j \in E'$ con $d_i \in D_j$ si y solo si son satisfacibles todas las cláusulas de C.

Cada deporte representará una cláusula y tomaremos como conjunto E al conjunto donde cada especialista e_i representa a un símbolo preposicional o a su negación. Por tando $E = \{e_1, ..., e_{2m}\}$, donde e_i representa a p_i y e_{i+m} representa a $\neg p_i \quad \forall i \in \{1, ..., m\}$. D_i será el conjunto de índices de las claúsulas donde aparece el símbolo al que representa e_i. Contratar a un especialista será equivalente a darle el valor "verdadero" al símbolo que representa. Tomamos $K = m$, ya que en el problema 3-SAT no podemos hacer verdaderos más de m literaales.

Como en el problema CE existe la posibilidad de contratar a $e_i \equiv p_i$ con $i \leq m$ y a $e_{i+m} \equiv \neg p_i$ pero en el problema 3-SAT no pueden ser ambos valores verdaderos, añadiremos a D deportes correspondientes a claúsulas de la forma $e_i \vee e_{i+m} \quad \forall i \in 1, ..., m$. Al hacer esto, tengo m nuevos deportes para los que necesito un especialista y cuento solo con 2 especialistas en E que son expertos en uno de esos m deportes y que no lo son en los otros $m-1$ deportes. De esta manera, para tener un especialista en cada uno de ellos tengo que elegir como mínimo $m = K$ diferentes especialistas, y como no puedo contratar a más de K, no contrataré a e_i y a e_{i+m} a la vez. De esta manera, siempre contrataré m especialistas, que corresponde a dar siempre un valor de verdad a p_i o a $\neg p_i$ (Sabemos que el conjunto $\{p_1, ..., p_m, \neg p_1 \neg p_m\}$ siempre tiene m literales verdaderos).

Si el problema 3-SAT tiene solución, hay una asignación de valores de verdad tal que en cada claúsula siempre hay un literal verdadero. Por tanto, hay m literales verdaderos en el conjunto $\{p_1, ..., p_m, \neg p_1 \neg p_m\}$ de manera que siempre encontramos uno de ellos en cada claúsula. Esto se traduce en que podemos contratar m especialistas de E que cubran los deportes de D.

De igual manera, si existe solución para el problema CE con la entrada definida, hay m elementos de $E = \{e_1, ..., e_{2m}\}$ que cubren todos los deportes de D, luego el asignar el valor "verdadero" a los literales correspondientes a los especialistas contratados resuelve el problema 3-SAT inicial.

Sólo queda comprobar que esta reducción se hace en espacio logarítmico.
Construimos una MT que transforme la entrada de 3-SAT en la entrada del problema CE, de manera que la entrada de la MT sea $U = \{p_1, ..., p_m\}$ $C = \{c_1, ..., c_n\}$ y su salida sea $D = \{d_1, ..., d_{n+m}\}$, $E = \{e_1, ..., e_{2m}\}$, $D_i = \{$índices de las claúsulas donde aparece el literal al que representa $e_i\}$ $\forall i \in \{1, ..., 2m\}$, $K = m$.

La MT funciona:
Por cada símbolo en U y en C añadimos un deporte a D, en la cinta de salida (un total de $n + m$ deportes).
Por cada símbolo en U añadimos dos especialistas a E, $e_i(\equiv p_i)$ y $e_{i+m}(\equiv \neg p_i)$, colocandolos en la cinta de salida tras D.
Para cada especialista e_i con $i \in \{1, ..., m\}$, recorremos C, usando un contador en una cinta auxiliar que indique el índice de la claúsula que estamos comprobando. Si aparece el literal p_i en esa claúsula, añadimos ese índice a D_i en la cinta de salida. Al final de D_i añadimos el valor $n + i$, que corresponde con la claúsula $e_i \vee e_{i+m}$.
Para cada especialista e_i con $i \in \{m+1, ..., 2m\}$, recorremos C, usando un contador en una cinta auxiliar que indique el índice de la claúsula que estamos comprobando. Si aparece el literal $\neg p_i$ en esa claúsula, añadimos ese índice a D_i en la cinta de salida. Al final de D_i añadimos el valor $n+i-m$, que corresponde con la claúsula $e_i \vee e_{i+m}$.
Para calcular $K = m$ llevamos un contador en una cinta auxiliar que cuente los elementos de U, y escribimos el valor resultante en la cinta de salida.

No consideramos la complejidad de recorrer la cinta de entrada o escribir de izquierda a derecha en la de salida. Por tanto, lo único que añade complejidad son los contadores en cintas auxiliares. Cada contador puede valer como mucho $n+m$. Para representarlo necesitamos a lo sumo $log(n+m)$ casillas y si llamamos s a la longitud de la entrada $n + m \leq s$, ya que necesitámos al menos una casilla para codificar cada símbolo en U y cada claúsula en C, luego $log(n + m) \leq log(s)$. Por tanto, la complejidad en espacio del algoritmo es $O(log(s))$.