

MA4701-1 - Optimización Combinatorial

Profesor: José Soto

Auxiliares: Nicolás Sanhueza - Christian von Borries



Auxiliar N° 15

5 diciembre de 2013

P1. El problema de PARTICIÓN consiste en decidir si, en una lista de números naturales dada $\{c_1, c_2, \dots, c_n\}$, existe un $S \subseteq \{1, \dots, n\}$ tal que $\sum_{i \in S} c_i = \sum_{i \notin S} c_i$. Se sabe que este problema es **NP**-completo.

- (a) El problema de la PARTICIÓNJUSTA tiene como entrada un conjunto S de objetos que tiene asociado una función de precios $w : S \rightarrow \mathbf{Q}^+$, y debe devolver $X \subseteq S$ tal que $\max\{w(X), w(X^c)\}$ sea mínimo. Muestre que PARTICIÓNJUSTA es **NP**-difícil.
- (b) Encuentre un PTAS para PARTICIÓNJUSTA.

Indicación: Si $1 < \rho < 2$ y queremos encontrar una ρ -aproximación, ordene los elementos de mayor a menor precio, ordene de manera óptima (mediante fuerza bruta) los primeros $k(\rho) = \lceil \frac{2-\rho}{\rho-1} \rceil$ elementos y para el resto use una estrategia «glotona».

P2. Un FPTAS (por *Fully Polynomial Time Approximation Scheme*) es un algoritmo de aproximación que, dado un $\varepsilon > 0$ y una instancia de un problema de maximización, encuentra una $(1 - \varepsilon)$ -aproximación en tiempo polinomial en el tamaño de la entrada y polinomial en $\frac{1}{\varepsilon}$.

El objetivo de este problema es encontrar un FPTAS para el problema de la MOCHILA. Recuerde que una instancia de MOCHILA queda dada por un conjunto de objetos $A = \{a_1, \dots, a_n\}$ que tienen asociado un tamaño $s : S \rightarrow \mathbf{N}$ y un valor $v : S \rightarrow \mathbf{N}$, y además una capacidad $B \in \mathbf{N}$ de la mochila, y debemos devolver $S \subseteq A$ tal que todos caben en la mochila y $v(S)$ es máximo.

- (a) Si $P = \max_{a \in A} v(a)$ es el valor del objeto más valioso, use programación dinámica para encontrar un algoritmo que resuelva MOCHILA de forma exacta en orden $O(n^2 P)$. ¿Es un algoritmo polinomial?

Indicación: Resuelva el subproblema $M[i, k] =$ «el tamaño $s(S)$ mínimo de un conjunto $S \subseteq \{a_1, \dots, a_i\}$ que tenga valor exactamente $v(S) = k$ », para $i \in [1, n]$ y $k \in [0, nP]$.

- (b) Usando lo anterior, encuentre un FPTAS para MOCHILA.

P3. El problema del EMPAQUETAMIENTO (o del *bin packing*) consiste en: dado n objetos $A = \{a_1, \dots, a_n\}$ con volúmenes $0 < v(a_i) \leq 1$, queremos guardarlos en cajas que tienen capacidad 1, minimizando el número de cajas usadas. Se sabe que este problema es **NP**-completo.

- (a) Encuentre un algoritmo polinomial que entregue una 2-aproximación de EMPAQUETAMIENTO.

Indicación: Use una estrategia glotona.

- (b) Usando que PARTICIÓN es **NP**-completo, muestre que no existe un algoritmo polinomial que entregue una ρ -aproximación para EMPAQUETAMIENTO con $\rho < \frac{3}{2}$, a menos que $\mathbf{P} = \mathbf{NP}$.