# Partition

Sawan J. Kapai Harpalani    Adrián González Martín    Sara Martín Molina    Enrique Tejera González

Universidad de La Laguna

December 17, 2014

# Partition

- Teorema: Partition es NP-Completo.
- Instancia: A, a $\in$ A y S(a) $\in (\mathbb{Z})^{+}$.
- Prueba: Es fácil ver que partition $\in$ NP, puesto que es un algoritmo no determinista necesita sólo encontrar un subconjunto A' de A y comprobar el tiempo polinomial que suma los tamaños de los elementos de A' es igual a la suma de los elementos de A - A'.

# Transformación 3DM a Partition

- Se fijan los conjuntos W, X, Y con tamaño q y M que será una instancia arbitraria del 3DM (M $\subseteq W \times X \times Y$).

$$W = w_1, w_2, w_3, \dots w_q$$

$$X = x_1, x_2, x_3, \dots x_q$$

$$Y = y_1, y_2, y_3, \dots y_q$$

$$M = m_1, m_2, m_3, \dots m_k$$

$$k = |M|$$

- Se debe construir un conjunto A, donde cada elemento tiene tamaño tal que S(a) $\in (\mathbb{Z})^+$ y ese A debe contener un subconjunto A' tal que:

$$\sum_{a \in A'} S(a) = \sum_{a \in A - A'} S(a) \iff si\ M\ contiene\ Matching$$

- El conjunto A contendrá k + 2 elementos.

# Construcción

El conjunto A se construye en dos pasos:

1. Primer paso:
   - Los primeros k elementos de A están asociados con las k tripletas de M

$$a_i \Rightarrow m_i, 1 \leq i \leq k$$

   - El tamaño de cada elemento se obtiene de su representación binaria. Esta representación contendrá q zonas con p bits cada una.

$$|W| = |X| = |Y| = q$$

$$p = [\log_2(k+1)]$$

# Construcción

1. Primer paso:
   - La representación binaria del elemento $a_i$ depende de la tripleta:

$$m_i = (w_{f(i)}, x_{g(i)}, y_{h(i)}) \in M$$

   - Otra forma de obtener el tamaño de $a_i$:

$$S(a_i) = 2^{\mathrm{p(3q\ -\ f(i))}} + 2^{\mathrm{p(2q\ -\ g(i))}} + 2^{\mathrm{p(q\ -\ h(i))}}$$

# Construcción

1. Primer paso:
   - Si se fija:

$$B = \sum_{j=0}^{3q-1} 2^{\mathrm{pj}}$$

   - Entonces:

$$A' \subseteq a_i : 1 \leq i \leq k$$

$$\sum_{a \in A'} = B \iff M' = m_i : a_i \in A' \Rightarrow matching(M)$$

# Construcción

2. Segundo paso:
   - Se especifican los dos últimos elementos de A ($b_1$ y $b_2$) cuyos tamaños son:

$$S(b_1) = 2(\sum_{i=1}^{k} S(a_i)) - B$$

$$S(b_2) = (\sum_{i=1}^{k} S(a_i)) + B$$

   - Ambos pueden ser especificados en binario con no más de $(3pq + 1)$ bits.

# Construcción

2. Segundo paso:
   - Suponiendo que se tiene un conjunto $A' \subseteq A$ se cumple:

$$\sum_{a \in A'} S(a) = \sum_{a \in A - A'} S(a)$$

   - Por lo que las sumas de ambos será:

$$2 \sum_{i=1}^{k} S(a_i)$$

   - Uno de los conjuntos, $A'$ *o* $A-$A', contendrá $b_1$ pero no $b_2$.

# Construcción

2. Segundo paso:
   - El resto de elementos formarán un subconjunto de:

$$a_i : 1 \leq i \leq k$$

   - La suma de los tamaños de esos elementos es igual a B, por lo que ese subconjunto es un matching de M' en M.
   - A la inversa, si tenemos un matching M':

$$b_1 \bigcup a_1 : m_i \in M' \ forma \ A' \ para \ la \ instancia \ de \ Partition$$

   - Por lo tanto 3DM se puede transformar en Partition en tiempo polinomial

# Ejemplo

| | W | | | X | | | Y | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | W1 | W2 | W3 | X1 | X2 | X3 | Y1 | Y2 | Y3 | |
| m1: | 01 | 00 | 00 | 01 | 00 | 00 | 01 | 00 | 00 | = 66576 |
| m2: | 00 | 01 | 00 | 00 | 01 | 00 | 00 | 01 | 00 | = 16644 |
| m3: | 00 | 00 | 01 | 00 | 00 | 01 | 00 | 00 | 01 | = 4161 |
| m4: | 01 | 00 | 00 | 00 | 00 | 01 | 00 | 00 | 01 | = 65601 |
| m5: | 01 | 00 | 00 | 00 | 01 | 00 | 00 | 00 | 01 | = 65793 |
| m6: | 01 | 00 | 00 | 00 | 00 | 01 | 00 | 01 | 00 | = 65604 |
| Matching: | 01 | 01 | 01 | 01 | 01 | 01 | 01 | 01 | 01 | = 87381 = B |

Figure: Ejemplo

FIN