

# 3-DIMENSIONAL MATCHING

Problema básico  $\mathcal{NP}$ -completo

Universidad de La Laguna

Curso 2016/2017

# Índice

1. NP-Problem
2. Clasificación NP-Problems
3. Clasificación NP-Compleitud
4. Definición 3 - Dimensional Matching
  - a) Caso general
  - b) Problema decisión
  - c) Problema optimización
  - d) Ejemplo
5. Demostración 3 - Dimensional Matching es NP-Problem
6. Demostración 3 - Dimensional Matching es NP-Completo
7. Comparación grafo bipartito

# NP-Problems

- Un problema se asigna a la clase **NP** (nondeterministic polynomial time - tiempo polinómico no determinista) si es resoluble en tiempo polinómico por una **Máquina de Turing** no determinista.
- Gran cantidad de programas no (necesariamente) se ejecutan en tiempo polinómico en un PC, pero se ejecuta en tiempo polinómico en una máquina de Turing no determinista.
- Estos programas resuelven **problemas en NP**.

Todos los problemas en **P** **están también en NP**, es decir, una máquina de Turing no determinista puede hacer todo lo que un ordenador normal puede y más.

# NP-Problems I

## Definición equivalente NP-Problem

Una forma equivalente de definir NP es señalando los problemas que se pueden verificar en tiempo polinómico. Esto significa que **no** se requiere necesariamente **tiempo polinómico** para encontrar una **solución**, pero una vez que haya una solución sólo se necesita **tiempo polinómico para verificar que es correcta**.

Si  $P = NP$ , cualquier problema que se pueda verificar en tiempo polinómico también puede ser resuelto en tiempo polinómico y viceversa.  
**Si pudieran probar  $\Rightarrow$  algoritmos rápidos para problemas importantes  $\Rightarrow$  revolucionar la informática.**

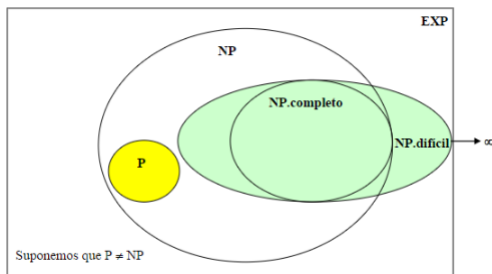
# Tipos NP-Problems

- **NP-Hard**. Problemas como mínimo tan difícil como NP, pero no necesariamente en NP.
- **NP-Completo**. Problemas que son completos en NP, es decir, está a la vez en NP y es un NP-Hard, los más difíciles de resolver en NP.
- **NP-Easy**. Problemas como mucho tan difícil como NP, pero no necesariamente en NP.
- **NP-Equivalente**. Problema igualmente difícil que NP, pero no necesariamente en NP.

# NP-Compleitud

Un problema es NP-completo si el problema es a la vez

- NP-hard
- está en NP



# 3-DM

# Definición 3- Dimensional Matching I

## Caso general

$X, Y, Z$  sean conjuntos finitos, disjuntos, y sea  $T$  un subconjunto de tripletas:

$(x, y, z) / x \in X, y \in Y, z \in Z.$

Ahora  $M \subseteq T$  es un conjunto, emparejamiento de 3 dimensiones, si se cumple lo siguiente: para cualquier par de tripletas distintas:

$$(x_1, y_1, z_1) \in M$$

$$(x_2, y_2, z_2) \in M$$

tenemos  $x_1 \neq x_2, y_1 \neq y_2, z_1 \neq z_2.$



# Definición 3- Dimensional Matching II

## Con otras palabras..

Trata emparejar elementos de los 3 conjuntos(dimensiones x-y-z) de 3 en 3 sin que se repitan en ningún caso los elementos escogidos de cada conjunto.

# Definición 3- Dimensional Matching III

## Problema de decisión

Dado un conjunto  $T$  y un entero  $k$ , decidir si existe un 3-DM  $M \subseteq T$  con  $|M| \geq k$ .

Sean  $X$ ,  $Y$  y  $Z$  conjuntos disjuntos finitos y sea  $T$  un subconjunto de tripletas:

$(x, y, z) / x \in X, y \in Y, z \in Z$ .

$M \subseteq T$  es un emparejamiento 3-Dimensional si cumple que para cualquier par de tripletas:

$$(x_1, y_1, z_1) \in M$$

$$(x_2, y_2, z_2) \in M$$

tenemos  $x_1 \neq x_2, y_1 \neq y_2, z_1 \neq z_2$ .

# Definición 3- Dimensional Matching IV

## Con otras palabras..

Trata de responder verdadero o falso, existe o no existe, que la cardinalidad de un conjunto de selecciones sea mayor o igual que un número dado ( $K$ ).

# Definición 3- Dimensional Matching V

## Problema de optimización

Dado un conjunto  $T$ , encontrar un juego de 3 dimensiones,  $M \subseteq T$  que maximiza  $|M|$ .

- Dado que el problema de decisión descrito anteriormente NP-completo, este problema de optimización es NP-hard, y por lo tanto parece que no hay algoritmo de tiempo polinómico para la búsqueda de una coincidencia máxima 3-dimensional.
- Sin embargo, existen algoritmos de tiempo polinómico eficientes para la búsqueda de un máximo coincidente bipartito (coincidencia máxima de 2 dimensiones), por ejemplo, el algoritmo de Hopcroft-Karp.

# Definición 3- Dimensional Matching VI

## Con otras palabras..

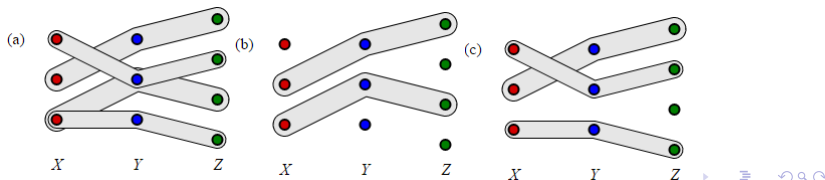
Trata de emparejar elementos de los 3 conjuntos (dimensiones x-y-z) de 3 en 3 sin que se repitan en ningún caso los elementos escogidos de cada conjunto, cumpliendo que la cardinalidad de los conjuntos seleccionados sea máxima.

## Ejemplo 3- Dimensional Matching VII

### Ejemplo

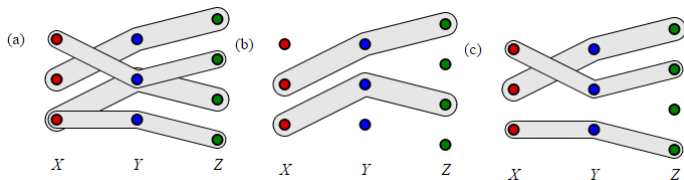
Abajo se muestran emparejamientos de 3 dimensiones o elementos.

- El conjunto X está marcado con puntos rojos
- Y está marcada con puntos azules
- Z está marcado con puntos verdes
- La figura (a) muestra el conjunto T (zonas grises).
- La figura (b) muestra un juego de 3 dimensiones M con  $|M| = 2$ .
- La Figura (c) muestra un juego de 3 dimensiones M con  $|M| = 3$ .



## Ejemplo 3- Dimensional Matching

La coincidencia de  $M$  se ilustra en la figura (c) es un emparejamiento de 3 dimensiones máximo, es decir, se maximiza  $|M|$ . Las agrupaciones de las figuras (b) y (c) son máximos 3-DM, es decir, no pueden ser extendidos por la adición de más elementos de  $T$ .



# Demostración 3DM es NP

## 3DM es NP

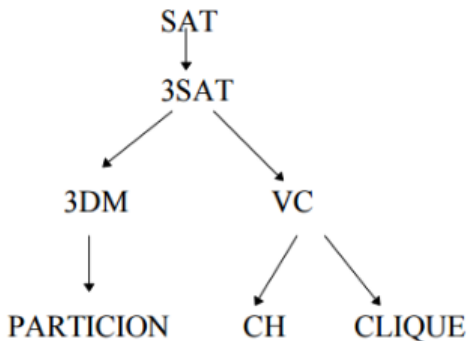
Suponga que hay algún algoritmo no determinista que de alguna manera te devuelve una lista de  $n$  tripletas. Todo lo que debe suceder ahora es que hay que verificar en tiempo polinómico que las tripletas dadas son válidas, que es simple, ya que sólo se asegura que ninguna de las tripletas tienen el mismo valor en todas coordenada. Por lo tanto podemos concluir que 3DM está en NP.



# Demostración 3DM es NP-completo I

## $3SAT \preceq 3DM$

Para probar que 3DM es NP-completo, tenemos que demostrar que 3SAT es reducible a 3DM.



# Demostración 3DM es NP-completo II

## Presentación del problema

Sea  $\alpha = C_1 \wedge C_2 \wedge \dots \wedge C_{k-1}$  una instancia del 3SAT con variables  $x_1, \dots, x_n$ . Construiremos una instancia  $I$  de 3DM de manera que  $\alpha$  sea satisfactible si y sólo si  $I$  tiene solución. Especificamos las tripletas de  $U$  en tres grupos:

- El primer grupo se usa para **seleccionar una asignación verdadera**.
- El segundo grupo se usa para **comprobar la satisfacción**.
- El tercer grupo se utiliza de **recolección de basura**.

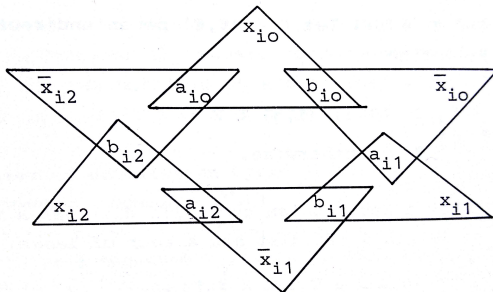
# Demostración 3DM es NP-completo III

## Grupo 1: Seleccionar asignación verdadera

Para cada variable  $x_i, 1 \leq i \leq n$ , tenemos  $2k$  tripletas:

$$G_{i1} = (a_{ij}, x_{ij}, b_{ij}), (b_{ij}, \bar{x}_{ij}, a_{i,(j+1) \bmod k}), 0 \leq j < k$$

No va a haber otra tripleta conteniendo los puntos  $a_{ij}$  ni  $b_{ij}, (1 \leq i \leq n, 0 \leq j < k)$ . Las tripletas del grupo uno se pueden visualizar de la siguiente forma ( $k = 3$ ):



# Demostración 3DM es NP-completo IV

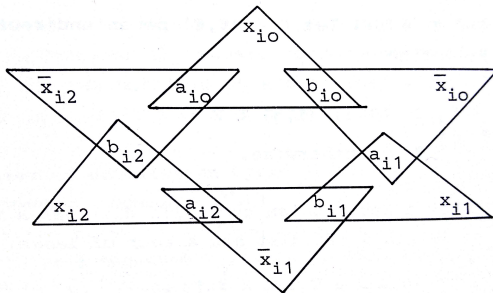
## Grupo 1: Seleccionar asignación verdadera

Hay exactamente dos formas de cubrir los puntos  $a_{ij}, b_{ij}, 0 \leq j < k$ , usando las tripletas en  $G_{i1}$ :

Dejar todos los  $x_{ij}$  descubiertos y cubrir los  $\bar{x}_{ij}, 0 \leq j < k$  ( $x_i$  es true)

Dejar los puntos  $\bar{x}_{ij}$  descubiertos y cubrir los  $x_{ij}, 0 \leq j < k$  ( $x_i$  es falso).

De esta forma, las tripletas de  $G_{i1}$  arreglan una asignación verdadera.



# Demostración 3DM es NP-completo V

## Grupo 2: Comprobación de satisfacción

Para toda cláusula  $C_j, 0 \leq j < k$ , tenemos las tripletas  $(S_1, \lambda, S_2)$  para cada literal  $\lambda$  que aparezca en la cláusula. No hay más tripletas que contengan los puntos  $S_2$  y  $S_2$ .

Supongamos ahora que los puntos del grupo 1 ya están seleccionados. Los puntos  $S_1$  y  $S_2$  pueden cubrirse si y sólo si hay una  $i$  tal que  $x_{ij}$  (ó  $\bar{x}_{ij}$ ) aparece en  $C_j$  y se queda descubierta por los puntos del grupo 1. Así, los puntos  $S_1, S_2, 0 \leq j < k$ , **pueden ser cubiertos** si y sólo si la asignación verdadera especificada en las tripletas del grupo 1 **satisfacen**  
 $\alpha$

# Demostración 3DM es NP-completo VI

## Grupo 3: Recolección de basura

En este punto, hemos construido una instancia de  $I$  de 3DM a partir de  $\alpha$  con la siguiente propiedad:

Si  $\alpha$  es **insatisfactible**, **no puede haber matching completo** en  $I$ .

Si  $\alpha$  es **satisfactible**, hay una manera de seleccionar las tripletas del grupo 1 y 2 de manera que los puntos  $a_{ij}, b_{ij}, S_1$  y  $S_2$  **están todos cubiertos** y exactamente  $nk - k = (n - 1)k$  de los puntos  $x_{ij}, x_{ij}$  están descubiertos.

Para completar el cubrimiento, añadimos en el grupo 3 las tripletas:

$$(h_r, x_{ij}, g_r) / 1 \leq r \leq (n - 1)k, 1 \leq i \leq n, 0 \leq j \leq k$$

# Demostración 3DM es NP-completo VI

$$Z = \{x_{ij}, \bar{x}_{ij} / 1 \leq i \leq n, 1 \leq j \leq k\}$$

$$X = \{A \cup S_1 \cup H\}$$

$$Y = \{B \cup S_2 \cup G\}$$

## Conclusión

Ahora es fácil ver que las instancias  $I(\alpha)$  de 3DM definidas anteriormente pueden ser construidas en tiempo polinomial dado  $\alpha$  y sabiendo que ese  $\alpha$  es satisfactible si y sólo si  $I(\alpha)$  permite una coincidencia completa.

Esto demuestra que  $SAT(3) \leq 3DM$

# Comparación con matching bipartito

## 2 Dimensional Matching

Un 2 Dimensional Matching se puede definir de una manera completamente análoga. Sean  $X$  e  $Y$  conjuntos finitos, disjuntos, y  $T$  subconjunto de formado por parejas  $(x,y)$ .

Ahora  $M \subseteq T$  es un 2 Dimensional Matching, si se cumple lo siguiente:  
Para dos pares distintos:

$$(x_1, y_1) \in M$$

$$(x_2, y_2) \in M$$

tenemos  $x_1 \neq x_2$  y  $y_1 \neq y_2$ .



# Comparación con matching bipartito

En el caso de coincidencia de 2-dimensional, el conjunto  $T$  se puede interpretar como el conjunto de bordes en un grafo bipartito  $G = (X, Y, T)$ , donde cada borde en  $T$  se conecta un vértice en  $X$  a un vértice en  $Y$ .

2-DM es entonces un emparejamiento a partir del grafo  $G$ , es decir, un conjunto de aristas no adyacentes por parejas.

Por lo tanto 3-DM se puede interpretar como una generalización de matchings a hypergrafos (una generalización de un grafo en el que un borde puede unirse a cualquier número de vértices):

los conjuntos de  $X$ ,  $Y$  y  $Z$  contienen los vértices, cada elemento de  $T$  es un hyperedge, y el conjunto de  $M$  consta de bordes no adyacentes por pares (bordes que no tienen un vértice común). En caso de coincidencia de 2 dimensiones, tenemos  $Y = Z$ .

# Referencias

- <http://www.math.cmu.edu/~af1p/Textfiles/PLANAR3DM.pdf>
- <https://sites.google.com/site/theuniverseofproblems/problems/3-dimensional-matching>
- [https://en.wikipedia.org/wiki/3-dimensional\\_matching](https://en.wikipedia.org/wiki/3-dimensional_matching)
- <http://users.dsic.upv.es/~esanchis/clases/ALT/tema3.PDF>
- <http://studylib.es/doc/535768/01-medidas-de-complejidad>
- <https://www.cs.cmu.edu/~ckingsf/bioinfo-lectures/3dm.pdf>
- <http://npcomplete.owu.edu/tag/3dm/>
- <http://courses.cs.vt.edu/cs5114/spring2009/lectures/lecture24-np-complete-problems.pdf>

**Diego Luis Afonso** - alu0100825657@ull.edu.es

**Daniel Daher Pérez** - alu0100811933@ull.edu.es

**Cristina Tosco González** - alu0100821338@ull.edu.es

**Alberto Antonio Sarabia Suárez** - alu0100734676@ull.edu.es

**Daniel Antonio Fernández Pérez** - alu0100812534@ull.edu.es