Algoritmia y Complejidad

Jean-Paul Beaudry[1] Salvador Carrillo Fuentes[1]

[1]Grado Ingeniería Informática
Universidad de Málaga

SCHEDULE $\in$ NP-Complete

Tabla de contenidos

- Lista de exámenes finales $F_1, ..., F_k$.
- Lista de estudiantes $S_1, ..., S_l$.
- Cada estudiante se presentará a un subconjunto de esos exámenes.
- Objetivo: planificar los exámenes en intervalos de tiempo de tal forma que ningún estudiante tenga que hacer dos exámenes en el mismo intervalo.
- Determinar si existe una planificacón válida que use h intervalos de tiempo.

Definición

Sea $SCHEDULE = \{\langle F_1, ..., F_k, S_1, ..., S_l, T_1, ..., T_l, h\rangle \mid T_i \subseteq \{F_1, ..., F_k\}$ y existe una planificación válida que no use más de h intervalos de tiempo$\}$, donde cada T_i es el conjunto de exámenes que realizará el estudiante S_i.

Ejemplo de instancia de *SCHEDULE*

Dado $h = 2$, $T_1 = \{F_1, F_2\}$, $T_2 = \{F_1, F_3\}$, $T_3 = \{F_2\}$, $T_4 = \{F_2, F_4\}$, podemos programar F_1 y F_4 para el primer intervalo y F_2 y F_3 para el segundo, y ningún estudiante tiene dos exámenes en el mismo intervalo.

SCHEDULE $\in$ NP

Teorema

SCHEDULE $\in$ NP

Demostración: Sea N una máquina de Turing no determinista que decide a *SCHEDULE* en tiempo polinómico.

$N =$"Con la entrada $\langle F, S, T, h \rangle$:

1. De forma no determinista, programar los exámenes finales en alguno de los h intervalos de tiempo.
2. Comprobar que ningún T_i tenga exámenes pertenecientes al mismo intervalo.
3. Si ningún S_i tiene que hacer más de un examen en el mismo intervalo, *aceptar*; en otro caso, *rechazar*."

$SCHEDULE \in$ NP

Teorema

$SCHEDULE \in$ NP

Demostración: Sea V un verificador poli-tiempo para $SCHEDULE$. El certificado c es la planificación válida de los F_i.

$V =$"Con la entrada $\langle\langle F, S, T, h\rangle, c\rangle$:

1. Comprobar si c es una programación tal que en ningún T_i haya exámenes en el mismo intervalo h_i.
2. Comprobar que F contiene todos los exámenes de c y h es el número de intervalos en c.
3. Si se dan ambas comprobaciones, *aceptar*; en otro caso, *rechazar*."

SCHEDULE $\in$ NP-Complete

Teorema

SCHEDULE $\in$ NP-Complete

- $COLOR = \{\langle G, h\rangle \mid G$ es un grafo donde sus vértices pueden ser coloreados con h colores donde no hay dos vértices adyacentes con el mismo color$\}$
- En nuestro caso, $3COLOR$ sería una variante de $COLOR$ donde $h = 3$.
- Sabemos que $3COLOR \in$ NP-Complete, vamos a demostrar que $3COLOR \leq_P SCHEDULE$.

$SCHEDULE \in$ NP-complete

- Debemos encontrar una función computable en tiempo polinomial $f : \Sigma^* \to \Sigma^*$, tal que para toda w,

$$w \in 3COLOR \Leftrightarrow f(w) \in SCHEDULE$$

  donde w es un grafo $\langle V, E \rangle$ y $f(w)$ es $\langle F, S, T, h \rangle$.

- Dado un grafo $G \in 3COLOR$, cada examen F_i es representado por un vértice v_i y supongamos que cada estudiante S_i se va a presentar a dos exámenes.
- Cada arista e_i que une dos vértices corresponde con un estudiante S_i y los dos exámenes que tiene que hacer. El número de franjas horarias h será 3, que es la cantidad de colores que podemos usar.

$SCHEDULE \in$ NP-complete

- El siguiente algoritmo computa la redcción $3COLOR \leq_P SCHEDULE$

$M_f =$ "Con entrada $\langle G \rangle$ donde G es un grafo coloreado:

1. Para cada vértice v_i de G crear un examen F_i.
2. Para cada arista e_i de G crear un estudiante S_i.
3. Con cada par de vértices unidos mediante la arista e_i, establecemos el conjunto de exámenes T_i que realizará el estudiante S_i.
4. Para cada color, establecemos una franja horaria y programamos el examen F_i en la franja horaria indicada por el color del vértice v_i.

$SCHEDULE \in$ NP-complete

Demostración:
($\Rightarrow$) Si $G \in 3COLOR$, podemos hacer una planificación válida. El color del vértice v_i se corresponde con la franja horaria del examen F_i .

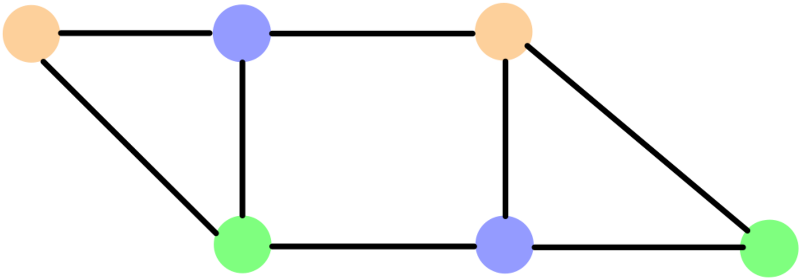
Como no puede haber dos vértices adyacentes del mismo color, los dos exámenes que cada alumno debe realizar se planificará en franjas horarias diferentes.

($\Leftarrow$) Si existe una planificación válida con $h = 3$ franjas horarias, podemos generar una 3-coloración para G. Seleccionamos un color para cada franja horaria.

Para todos los exámenes en la misma franja, coloreamos los vértices correspondientes con el mismo color. Elegimos un color distinto para cada franja. Obtenemos una 3-coloracion porque cada arista corresponde con un estudiante y une los dos exámenes que tiene que realizar, los cuales no pueden estar planificados en la misma franja horaria (mismo color), si la planificación es válida. Concluimos que $\underline{SCHEDULE \in NP - Complete}$.

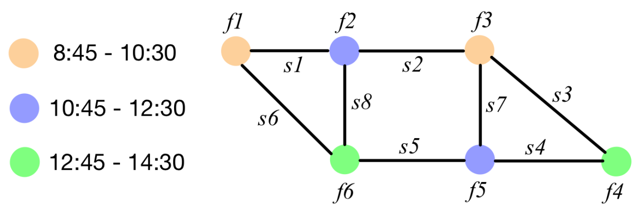
Ejemplo

Grafo 3-coloreable



Ejemplo

Planificación válida usando 3 franjas



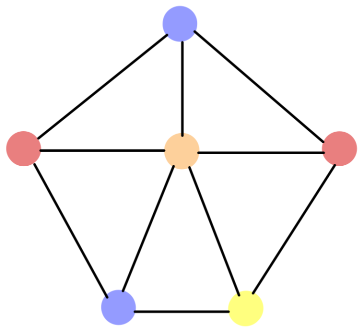
$$F = \{f_1, f_2, f_3, f_4, f_5, f_6\}$$

$$S = \{s_1, s_2, s_3, s_4, s_5, s_6\}$$

$$T_1 = \{f_1, f_2\},\ T_2 = \{f_2, f_3\},\ T_3 = \{f_3, f_4\},\ T_4 = \{f_4, f_5\}$$
$$T_5 = \{f_5, f_6\},\ T_6 = \{f_1, f_6\},\ T_7 = \{f_3, f_5\},\ T_8 = \{f_2, f_6\}$$

Ejemplo

Grafo NO 3-coloreable



El grafo no pertenece a $3COLOR$ y por tanto no podemos obtener una planificación válida usando 3 franjas horarias, ya que algún estudiante tendría dos exámenes en la misma franja.