Facultad de Ciencias Exactas, Ingeniería y Agrimensura

Tópicos Avanzados en Optimización Combinatoria y Teoría de Grafos

PRÁCTICA 0.2

Alumno: Rodríguez Jeremías

4 de julio de 2017

1. Ejercicio 3A

Dados los parámetros $A \in \mathbb{R}^{m\times n}$; $b, c \in \mathbb{R}^n$ y $x^* \in \{0,1\}^n$ del problema, realizamos dos formulaciones en programación lineal entera:

Primer formulación:

$$\begin{aligned}
& min\ cx \\
s/a & \\
& Ax \leqslant b \\
& x_i \in \mathbb{Z}, \qquad \forall\ i \in \{1,\dots,n\} \\
& 0 \leqslant x_i \leqslant q \quad \forall\ i \in \{1,\dots,n\} \\
& \sum_{i=1}^{n}((x_i.x_i^*) + ((1-x_i).(1-x_i^*))) < n
\end{aligned}$$

En esta última desigualdad, cada término de la sumatoria vale 1 si la componente i-ésima de x es igual a la de x^*; y 0 en caso contrario. De este modo, la última restricción impone que no todas las componentes sean iguales; es decir, que $x \neq x^*$.

Segunda formulación:

2. Ejercicio 7

Claramente:

$$dom(\Pi) = \{(0,0,0,0),(0,0,0,1),(0,0,1,0),(0,0,1,1),(0,1,0,0),(0,1,0,1),(1,0,0,0)\} = dom(\Pi')$$

Además, Π y Π' tienen la misma función objetivo. Por lo tanto, son problemas equivalentes.

Sea P el poliedro dominio de la relajación lineal de Π, y P' el de Π'. Para probar que Π' es una mejor formulación que Π, veamos que $P' \subset P$.

Sea $(x_1, x_2, x_3, x_4) \in P'$. Luego, $4x_1 + 3x_2 + 2x_3 + x_4 \leqslant 4$. Equivalentemente, $100x_1 + 75x_2 + 50x_3 + 25x_4 \leqslant 100$, de donde se deduce $83x_1 + 61x_2 + 49x_3 + 20x_4 \leqslant 100$.

Además, $(\frac{6}{5}, 0, 0, 0) \in (P - P')$.

$$\therefore P' \subset P$$

3. Ejercicio 8.a

Realizamos la siguiente formulación del problema en formato lp:

```
/* Variables */

// xj : cantidad de jamones no ahumados producidos
// xl : cantidad de lomos no ahumados producidos
// xs : cantidad de salchichas no ahumadas producidas

// xja: cantidad de jamones ahumados producidos (sin emplear horas extra)
// xla: cantidad de lomos ahumados producidos (sin emplear horas extra)
// xsa: cantidad de salchichas ahumadas producidas (sin emplear horas extra)

// xje: cantidad de jamones ahumados en horas extra
// xle: cantidad de lomos ahumados en horas extra
// xse: cantidad de salchichas ahumadas en horas extra

// j : se enciende la máquina J (variable de decisión, 0-1)
// l : se enciende la máquina L (variable de decisión, 0-1)
// s : se enciende la máquina S (variable de decisión, 0-1)

/* Objective function */
MAX: 8 xj + 4 xl + 4 xs + 14 xja + 12 xla + 13 xsa + 11 xje + 7 xle + 9 xse - 50j - 50l - 50s;

/* Variable bounds */

xj + xja + xje - 10 j <= 480;  // Producción de jamones
xl + xla + xle - 40 l <= 400;  // producción de lomos
xs + xsa + xse - 20 s <= 230;  // producción de salchichas

xja + xla + xsa <= 420;       // capacidad de ahumado
xje + xle + xse <= 250;       // capacidad de ahumado extra

xj  >=0;
xja >=0;
xje >=0;
xl  >=0;
xla >=0;
xle >=0;
xs  >=0;
xsa >=0;
xse >=0;

0 <= j <= 1;
0 <= s <= 1;
0 <= l <= 1;
                     //         J L S
j + l + s <= 2;      // evita   1 1 1
j <= l + s;          // evita   1 0 0
l <= j + s;          // evita   0 1 0
```

```
int j;
int s;
int l;
```

Usando LPSolve, obtuvimos que el valor máximo de la función objetivo es 11110 para la siguiente asignación de valores:

```
xj = 480
xl = 20
xs = 0
xja = 0
xla = 420
xsa = 0
xje = 0
xle = 0
xse = 250
j = 0
l = 1
s = 1
```