



VNIVERSITAT
DE VALÈNCIA

FACULTAD DE MATEMÁTICAS - Curso 2018/ 2019

Formulación del problema de los festivales

Realizado por:
Juan Francisco Gómez González

 Facultat de
Ciències Matemàtiques

Índice general

1. Introducción	2
1.1. Historia	2
2. Formulación del problema	3
2.1. Con un escenario	3
2.2. Con varios escenarios	4
3. Resolución del problema con LPSolve	5

Capítulo 1

Introducción

Este problema parte de la asignación de los grupos en los distintos escenarios haciendo que los grupos con distinta duración de concierto cuadren en tiempo con los distintos días y escenarios que están disponibles, teniendo en cuenta el tiempo de limpiado del escenario y el tiempo de set-up.

En los distintos capítulos veremos la formulación más sencilla con un solo escenario hasta la formulación para varios escenarios.

1.1. Historia

Todo parte de un problema surgido en la semana cultural de la Universitat de Valencia, donde teníamos que asignar a 9 grupos en 3 días. Estos grupos tenían duraciones diferentes e instrumentos diferentes, por lo tanto necesitaban un tiempo de preparación antes de empezar el concierto.

Capítulo 2

Formulación del problema

Formularemos primeramente el problema con un escenario para que el lector note el cambio entre poner varios escenarios.

2.1. Con un escenario

Tenemos n grupos para tocar en d días. El tiempo máximo en el día j es de t_{max_j} y el tiempo de limpiar incluyendo set-up del grupo j es de c_j .

Nuestro objetivo es distribuir equitativamente entre los días disponibles los grupos que tocan en el escenario, entonces nuestra función objetivo será la siguiente:

$$\text{Min} \sum_{i=1}^d (t_{max_i} - \sum_{j=1}^n x_{i,j}) \quad (2.1)$$

Nuestra variable $x_{i,j} \in \{0, 1\}$, será 1 si el grupo j toca el día i y 0 si no toca. Este problema está sujeto a la restricción

$$\sum_{j=1}^n t_j x_{i,j} \leq t_{max_i} - \sum_{j=1}^n c_j x_{i,j} \quad \forall i \in \{1, \dots, d\} \quad (2.2)$$

Esta expresión nos dice que los grupos que toquen el día i -ésimo con la duración estimada t_j no podrán exceder el tiempo máximo menos el tiempo que tarden en limpiar el escenario y montar las cosas necesarias ($\sum_{j=1}^n c_j x_{i,j}$).

La siguiente restricción es la que nos dice que un grupo solo podrá tocar una única vez.

$$\sum_{i=1}^d x_{i,j} \leq 1 \quad \forall j \in \{1, \dots, n\} \quad (2.3)$$

Nos dice que en el grupo j -ésimo solo podrá tocar un único día. Le ponemos menor o igual puesto que por restricciones de tiempo es posible que no todos los grupos puedan tocar. Con esta desigualdad aseguramos que el problema tenga solución.

Por último, y es lo que hace que el problema sea difícil de resolver, es las condiciones que le daremos a nuestras variables.

$$x_{i,j} \in \{0, 1\} \quad \forall i \in \{1, \dots, n\} \quad \forall j \in \{1, \dots, d\} \quad (2.4)$$

Con esto tenemos un problema lineal entero.

2.2. Con varios escenarios

La mayor diferencia con este es el super índice que le pondremos a nuestra variable, x , y al tiempo máximo que tenemos en el escenario k del día i . La formulación quedaría del siguiente modo.

La función objetivo sería:

$$\text{Min} \sum_{k=1}^s \sum_{i=1}^d (t_{max_i}^k - \sum_{j=1}^n x_{i,j}^k) \quad (2.5)$$

Sujeto a las restricciones siguientes.

$$\sum_{j=1}^n t_j x_{i,j}^k \leq t_{max_i}^k - \sum_{j=1}^n c_j x_{i,j}^k \quad \forall i \in \{1, \dots, d\}, \quad \forall k \in \{1, \dots, s\} \quad (2.6)$$

$$\sum_{k=1}^s \sum_{i=1}^d x_{i,j}^k \leq 1 \quad \forall j \in \{1, \dots, n\} \quad (2.7)$$

$$x_{i,j}^k \in \{0, 1\} \quad \forall i \in \{1, \dots, n\} \quad \forall j \in \{1, \dots, d\} \quad \forall k \in \{1, \dots, s\} \quad (2.8)$$

Capítulo 3

Resolución del problema con LPSolve

El código de LPSolve para un único escenario es el siguiente:

```
/* LPSolver */
param n;
param d;
param s;
set Dias:=1..d;
set Grupos:=1..n;
param tmax{Dias};
param t{Grupos};
param c{Grupos};
/* Variable definitions */
var x{Dias,Grupos} binary;

/* Objective function */
minimize obj: sum{i in Dias} (tmax[i]-sum{j in Grupos}t[j]*x[i,j]);
subject to Maxtime{i in Dias}: sum{j in Grupos}t[j]*x[i,j]<=tmax[i]-sum{j in Grupos}c[j]*x[i,j];
subject to SoloTocanUnaVez{j in Grupos}: sum{i in Dias}x[i,j]<=1;
```