**Heurística y Optimización**

**Práctica 2**



Arturo Cardenal López-Santos

100432160

# Introducción

En esta memoria se recoge la modelización del problema propuesto en la primera parte como un problema de satisfacción de restricciones (CSP). Además de dicha modelización se incluyen algunas explicaciones sobre la implementación en código utilizando la librería *python-constraint*. A continuación, se realiza un análisis sobre las pruebas propuestas.

Puesto que no iba a poder realizar la segunda parte completa con su respectiva modelización e implementación, he decidido centrarme exclusivamente en la primera parte de la práctica para intentar sacarla de la mejor manera posible.

Por último, se incluye una sección en la que se valora la dificultad y la utilidad de la práctica

# Parte 1

## Descripción Formal del Modelo

A continuación, se modela el primer problema planteado como un problema de programación lineal entera.

* **Datos del problema:**
* LOCALIZACIONES = {L1, L2, L3}
* i ∈ {1, 2, 3}
* DISTRITOS = {D1, D2, D3, D4, D5}
* j ∈ {1, 2, 3, 4, 5}
* Tij : tiempo que tarda una ambulancia la localización *Li* en atender una llamada del distrito *Dj*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tiempo de atención (en minutos) de Li a Dj** | **D1** | **D2** | **D3** | **D4** | **D5** |
| **L1** | 20 | 30 | 50 | 15 | 32 |
| **L2** | 15 | 18 | 30 | 12 | 15 |
| **L3** | 34 | 20 | 14 | 30 | 18 |

* Llamadas\_D1 = 5.000
* Llamadas\_D2 = 6.500
* Llamadas\_D3 = 5.400
* Llamadas\_D4 = 7.500
* Llamadas\_D5 = 5.500
* Máximo\_llamadas\_Li = 10.000 ∀ i ∈ I
* Tiempo\_máximo (minutos) = 35
* Porcentaje\_máximo = 150% = 1,5
* **Variables de decisión:**
* Xij : Número de llamadas del distrito Dj atendidas por una ambulancia del distrito Li
* **Función Objetivo:**
* Minimizar Tiempo\_total :

El tiempo total que se desea minimizar, es la suma de todos los productos de llamadas asignadas a una localización, por el tiempo que tarda en atender a esa localización.

* **Restricciones:**
* Se deben atender todas las llamadas de todos los distritos

* Cada localización no puede atender a más de un máximo de llamadas (10.000 llamadas para todas las localizaciones)
* Garantizar que una ambulancia no tardará más de un tiempo máximo (35 minutos) en atender una llamada

Si el tiempo en atender una llamada es superior al máximo establecido, la diferencia será negativa. Por tanto, la única manera de satisfacer la restricción es que la variable de decisión valga 0, es decir, que no se atienda.

* Las llamadas asignadas a una localización *Li* no pueden superar más del porcentaje máximo el número de llamadas asignadas a cualquier otra localización

Esto significa que, por ejemplo, las llamadas asignadas a L1, se compararían con las llamadas asignadas a L2 y a L3, pero no se compararían consigo mismas. Aunque comparando L1 con L1, la condición se cumpliría, no tiene ningún sentido añadir dicha condición, puesto que en ningún caso va a limitar el problema.

* El número de llamadas asignadas a las localizaciones es un número natural (asumiendo que el 0 es natural).

## Implementación del modelo

Para resolver este modelo con la herramienta *Solver*, he introducido las siguientes celdas en los campos correspondientes:

* *Set objective:*
  + $B$23
* *By Changing Variable Cells:*
  + $B$2:$F$4
* *Subject to the Constraints:*
  + $B$45:$B$59 >= $C$45:$C$59
  + $B$27:$B$31 = $C$27:$C$31
  + $B$34:$B$42 <= $C$34:$C$42
  + $B$2:$F$4 = integer

## Resultados

La **función objetivo** minimizada da como resultado: **483.800 minutos**.

Y la asignación de llamadas a las localizaciones es:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Li atiende … llamadas de Dj** | **D1** | **D2** | **D3** | **D4** | **D5** |
| **L1** | 2400 | 0 | 0 | 7500 | 0 |
| **L2** | 2600 | 1900 | 0 | 0 | 5500 |
| **L3** | 0 | 4600 | 5400 | 0 | 0 |

Todas las llamadas de todos los distritos son atendidas, puesto que:

* Para D1 🡪 2.400 + 2.600 = 5.000
* Para D2 🡪 1.900 + 4.600 = 6.500
* Para D3 🡪 5.400
* Para D4 🡪 7.500
* Para D5 🡪 5.000

Ninguna localización se pasa del máximo de llamadas (10.000):

* L1 🡪 2.400 + 7.500 = 9.900
* L2 🡪 2.600 + 1.900 + 5.500 = 10.000
* L3 🡪 4.600 + 5.400 = 10.000

Ninguna de las combinaciones asignadas tiene un tiempo de atención superior al máximo propuesto (35 minutos):

|  |  |
| --- | --- |
| L1 🡪 D1 | 20 minutos |
| L2 🡪 D1 | 15 minutos |
| L2 🡪 D2 | 18 minutos |
| L3 🡪 D2 | 20 minutos |
| L3 🡪 D3 | 14 minutos |
| L1 🡪 D4 | 15 minutos |
| L2 🡪 D5 | 15 minutos |

Ninguna de las localizaciones tiene asignadas más del 50% de todas las llamadas asignadas a cualquier otra localización:

* Para L1:
  + 9.900 < 1,5 · (10.000)
  + 9.900 < 1,5 · (10.000)
* Para L2 y L3:
  + 10.000 < 1,5 · (9.900)
  + 10.000 < 1,5 · (10.000)

Además, como todas las asignaciones de llamadas son números enteros, podemos concluir que se cumplen todas las restricciones propuestas en este modelo.

Las restricciones que están limitando el problema son:

* Las restricciones que determinan que todas las llamadas deben ser atendidas.
* Las restricciones que determinan que L2 y L3 no pueden atender más de 10.000 llamadas.
* Algunas de las restricciones que determinan que no se puede exceder el tiempo máximo de atención de 35 minutos.

Las restricciones que tratan el tiempo de atención máximo para las variables de decisión, cuyo tiempo de atención es inferior al límite, no están limitando el problema.

De igual manera, las restricciones que determinan que el total de llamadas asignadas a una localización no deben sobrepasar por más del 50% las llamadas asignadas a cualquier otra localización tampoco están limitando el problema.

En este problema se han definido 15 variables de decisión y hasta 30 restricciones (contando con la que dictamina que las variables de decisión han de ser números naturales).

A continuación, se plantean 2 casos distintos basados en el mismo problema:

1. En el primer caso se utilizan las mismas variables de decisión y se modifican algunas de las restricciones que limitaban el problema anterior. Ahora el tiempo máximo de atención de llamada es de 28 minutos y el máximo de llamadas que puede atender una localización es de 12.000. Los resultados obtenidos son:

Función objetivo = 474.100 minutos

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Li atiende … llamadas de Dj** | **D1** | **D2** | **D3** | **D4** | **D5** |
| **L1** | 0 | 0 | 0 | 7500 | 0 |
| **L2** | 5000 | 750 | 0 | 0 | 5500 |
| **L3** | 0 | 5750 | 5400 | 0 | 0 |

En este caso, el número máximo de llamadas ya no limita el problema. Sin embargo, ahora la restricción que determina que el número máximo de llamadas asignadas a L2 no puede ser mayor que el 150% de las llamadas asignadas a L1 sí que pasa a limitar el problema (11.250 = 1,5 · [7500]).

La dificultad de resolución del problema permanece igual, puesto que tanto el número de variables como de restricciones son iguales que en la versión original del problema.

1. En este segundo caso, se añade la localización L4 incluida en la Parte 2 aplicándole las mismas restricciones que a las primeras, para aumentar la complejidad del problema. Los resultados obtenidos son:

Función objetivo = 432.290 minutos

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Li atiende … llamadas de Dj** | **D1** | **D2** | **D3** | **D4** | **D5** |
| **L1** | 0 | 0 | 0 | 5980 | 0 |
| **L2** | 0 | 1950 | 0 | 1520 | 5500 |
| **L3** | 0 | 580 | 5400 | 0 | 0 |
| **L4** | 5000 | 3970 | 0 | 0 | 0 |

Aquí la restricción que limita el número máximo de llamadas atendidas por localización reduce su importancia y deja de limitar el problema. Por otra parte la restricción que determina que el número de llamadas atendidas por una localización no puede superar el 150% de llamadas asignadas a cualquier otra, gana importancia y pasa a limitar el problema para varias combinaciones de Li y Dj.

En este caso, la dificultad de resolución si que aumenta, puesto que pasamos a tener 20 variables de decisión y hasta 42 restricciones.

# Parte 2

## Descripción Formal del Modelo

## Resultados

# Conclusión de la práctica

En cuanto a la resolución del problema mediante el uso de hojas de cálculo, me parece que ha sido un proceso bastante intuitivo. En el programa de Excel no he encontrado el problema que mencionaste en clase acerca de tener que volver a aplicar la función *Solver* tras cerrar el programa. En mi caso, las celdas introducidas en la herramienta *Solver* se quedaban guardadas. No obstante, como inconveniente le veo que, al trabajar con una matriz de variables, es decir, trabajando a lo largo de los dos ejes, y teniendo que establecer las restricciones únicamente en sentido vertical, es mucho más complicado hacer uso de las referencias relativas y absolutas. Esto ralentiza en gran medida el fujo de trabajo. Por otra parte, añadir variables es bastante tedioso. Al añadir la localización L4 para comparar los resultados obtenidos originalmente, he tenido que volver a introducir manualmente unas cuantas restricciones. Esto ha provocado que solo haya añadido una localización más, en lugar añadir también L5 y L6.

El MathProg no lo he llegado a utilizar demasiado. No obstante, me parecía bastante menos intuitivo de primeras. Me resulta un poco confuso cómo funcionan los *sets* y sus posibles interacciones con otros sets. Pero por otra parte, creo que añadir restricciones y variables de decisión debe de ser mucho más cómodo y rápido que con las hojas de cálculo.

La práctica en sí me parece una gran forma de visualizar cuáles son las aplicaciones de la optimización en el mundo real. De hecho, creo que es algo bastante más interesante que aprender a realizar todos los ejercicios en papel o a desarrollar el método SIMPLEX paso por paso. Aprender a manejar estas herramientas con soltura en problemas aplicables a la vida real me parece algo bastante más valioso que saber hacer los ejercicios en papel, sobre todo hoy en día.

En cuanto a la complejidad de la práctica, creo que es acertada, aunque choca bastante el contraste entre la primera y la segunda parte. Sin embargo, creo que algunas restricciones eran un poco confusas y daban pie a interpretaciones ligeramente distintas. Esto levanta bastantes dudas, no sobre si se ha implementado bien, sino sobre si se ha interpretado bien, lo que no me parece ideal para con el objetivo de esta práctica.

Lo que creo que no es acertado es que, si no se acaban la parte 1 o la 2 completas, no cuente nada de estas. Por ejemplo en mi caso, en el que me he visto sin tiempo suficiente a completar la segunda parte, no tengo ningún incentivo para seguir intentando sacar algún apartado y rascar algún punto. Pero no es sólo el caso de quedarse sin tiempo, si uno cree que no va a ser capaz de comprender el lenguaje de MathProg, o si uno no es capaz de sacar todas las restricciones del segundo modelo, no tiene ningún incentivo a seguir trabajando más en el proyecto.