Heurística y Optimización

GRADO EN INGENIERÍA INFORMÁTICA

Práctica 1

Curso 2020/2021

Jorge Rodríguez Fraile, 100405951, Grupo 81, 100405951@alumnos.uc3m.es
Carlos Rubio Olivares, 100405834, Grupo 81, 100405834@alumnos.uc3m.es

Índice

INTRODUCCIÓN A LOS CONTENIDOS

En esta práctica se nos presentan dos problemas de programación lineal que resolver. Ambos se basan en problemas de programación lineal y están relacionados con diferentes aviones y sus vuelos. En el primero se nos presenta un problema más sencillo donde tenemos que ver que venta de billetes es más rentable, pero en el segundo problema se nos da un grid que representan las pistas de aterrizaje y sus franjas horarias de uso. Estas franjas de uso están delimitadas y tienen restricciones, por lo que supone una vuelta de tuerca visto anteriormente y supondrá un problema algo más desafiante. Se nos pide por último fusionar los dos problemas para que sean un solo modelo de MathProg, seguramente esta sea la parte más 'fácil' de realizar ya que si podemos llegar a comprender de manera correcta los dos problemas anteriores, juntarlos en uno solo no supondrá gran complicación.

PARTE 1

DESCRIPCIÓN DE LOS MODELOS

El primero de ellos se basa en averiguar cúal es la mejor combinación de billetes que podemos vender en 5 aviones teniendo en cuenta diferentes restricciones. Este problema lo debemos resolver en Calc, por lo que tendremos que presentar unos datos, seguramente calcular otros intermedios y establecer una función objetivo. Una vez hecho eso debemos representar nuestras restricciones de forma algebraica para poder adaptarlas a una función. Al terminar, aplicaremos dichas restricciones a la función objetivo y así obtener el resultado deseado. El desarrollo obtenido en esta parte es el siguiente:

Variables:

$Tarifa = \{Estándar,\ Leisure,\ Business\};\ Avión = \{A1,\ A2,\ A3,\ A4,\ A5\}$

x_{ij} : Número de unidades de la tarifa i que se venden en el avión j,

$\forall i \in Tarifa;\ \forall j \in Avión$.

La variable de decisión es el número de billetes de cada tarifa para cada avión, se hace de esta manera para poder manejar las restricciones que tiene cada uno de los aviones por separado.

Datos:

c_i : Coste de la tarifa i, $\forall i \in Tarifa$.

e_i : Equipaje permitido en la tarifa i, $\forall i \in Tarifa$.

a_j : Número de asientos del avión j, $\forall j \in Avión$.

b_j : Capacidad del avión j (en kg), $\forall j \in Avión$.

Función objetivo:

$$z = \sum_{j=Avión} \sum_{i=Tarifa} x_{ij} \cdot c_i$$

La función objetivo calcula el beneficio total, y se calcula multiplicando el número de billetes de un tipo por el precio del mismo para todos los aviones y tarifas.

Restricciones:

$$\sum_{i=Tarifa} x_{ij} \leq a_j;\ \ \forall j \in Avión$$

Esta restricción explica que no se deben vender más billetes que los asientos que hay en el avión.

$$\sum_{i=Tarifa} x_{ij} \cdot e_i \leq b_j;\ \ \forall j \in Avión$$

No podemos superar la capacidad máxima de cada avión.

$$x_{leisure\, j} \geq\ 20;\ \ \forall j \in Avión$$

El número de billetes leisure debe ser mayor o igual a 20 en cada avión.

$$x_{business\, j} \geq\ 10;\ \ \forall j \in Avión$$

El número de billetes business debe ser mayor o igual de 10 en cada avión.

$$\frac{\sum_{j=Avión} x_{estándar j}}{\sum_{j=Avión} b_j} \geq 0,6$$

Esta restricción supone que tiene que haber al menos un 60% de billetes tipo estándar vendidos.

$$x_{ij} \geq 0;\ \ \forall i \in Tarifa;\ \ \forall j \in Avión\,.$$
$$x_{ij} \in\ Z^{+}{}_{0};\ \ \forall i \in Tarifa;\ \ \forall j \in Avión\,.$$

PARTE 2

DESCRIPCIÓN DE LOS MODELOS

En la segunda parte de la práctica, se nos pregunta cómo acomodar 5 aviones en 4 pistas, con diferentes franjas horarias, teniendo en cuenta ciertas restricciones. Esta parte aumenta bastante más la dificultad a la hora de incluir una relación entre 3 elementos, aunque el proceso será muy parecido al anterior. Estableceremos nuestras variables y función objetivo en MathProg, nuestras restricciones también y lo ejecutaremos en una terminal con glpk; el resultado será volcado en otro fichero. El trabajo sobre esta parte ha sido el siguiente:

Variables:

$Avion\_Franja\_Pista$

Es una matriz tridimensional binaria de dimensiones $\#Avion\ x\ \#Franja\ x\ \#Pista$, que representa la posición de cada avión en su franja y pista asignada. 1 representa asignado y 0 sin asignar.

$Avion\_Franja\_Pista\_i$

Es la matriz opuesta a $Avion\_Franja\_Pista$, lo que quiere decir que la franja y pista asignada se representa con un 0 y el resto son 1.

Datos:

$Ocupado$: Matriz binaria 4x6 ($\#Pista\ x\ \#Franja$) que representa los slots del problema, se rellenan con 1 los slots que aparecen en negro en el enunciado, y con 0 los verdes, es decir los slots libres.

p_i : hora programada del avión i, representada en minuto.

l_i : hora límite del avión i, representada en minuto.

c_i : coste por minuto de retraso del avión i.

i_i : hora de inicio de la franja i, representada en minuto.

$Pista\ =\ \{P1,\ P2,\ P3,\ P4\}$

$Franja\ =\ \{F1,\ F2,\ F3,\ F4,\ F5,\ F6\}$

$Avión\ =\ \{A1,\ A2,\ A3,\ A4,\ A5\}$

Función objetivo:

$$z = \sum_{i=Avión} \sum_{j=Franja} \sum_{k=pista} Avión\_Franja\_Pista_{ijk} * c_i * (i_j - p_i)$$

Restricciones:

$$Avión\_Franja\_Pista_{ijk} + Ocupado_{kj} \leq\ 1;\ \ \forall\, i\ \in\ Avión,\ \ \forall\, j\ \in\ Franja,\ \ \forall\, k\ \in\ Pista$$

Esta restricción asegura que no se asigna a un avión una franja y pista restringida por la aerolínea. Para lograrlo como como ambas son matrices binarias, si no se puede usar en Ocupado habrá un 1 y en Avion_Franja_Pista un 0, por lo que si sumamos esa franja y pista con la Avion_Franja_Pista como máximo podrá valer 1, ya que ambas pueden valer 0.

$$\sum_{i=avión} Avión\_Franja\_Pista_{ijk} \leq 1 \ \forall j \in Franja, \ \forall k \in Pista$$

Esto restringe que cada slot de tiempo puede estar asignado como máximo a un avión, para esto obligamos a que nuestra matriz tridimensional pueda tener o un solo avión o ninguno en esa franja y pista para todos los aviones.

$$\sum_{j=franja} \sum_{k=pista} Avión\_Franja\_Pista_{ijk} = 1 \ \forall i \in Avión$$

Mediante esta fórmula restringimos que en cada slot puede estar como máximo un avión, para esto fijamos cada avión y comprobamos que sólo aparece una vez en la matriz resultado.

$$(1440 * Avion\_Franja\_Pista\_i_{ijk}) * i_j * Avión\_Franja\_Pista_{ijk} \geq p_i; \ \forall i \in Avión, \ \forall j \in Franja, \ \forall k \in Pista$$

En este caso estamos controlando que la franja asignada al avión deba ser mayor que su hora de aterrizaje programada, lo que quiere decir que sea posterior temporalmente. Para controlar que la condición se cumpla siempre, aunque la franja que comprobamos no se le haya asignado al avión y además asegurarnos de que sea un 0 en la matriz, sumamos un valor suficientemente grande cuando esto ocurra gracias a la inversa de la matriz.

$$(1440 * Avion\_Franja\_Pista\_i_{ijk}) * l_i * Avión\_Franja\_Pista_{ijk} \geq i_j; \ \forall i \in Avión, \ \forall j \in Franja, \ \forall k \in Pista$$

De la misma manera que en la restricción anterior vamos a controlar la hora de pista asignada, pero ahora con respecto a que la hora asignada no supere la hora límite de aterrizaje del avión. Cuando la franja no está asignada la desigualdad se cumplirá gracias al valor suficientemente grande, y cuando esté asignada ese valor no influirá y la condición se deberá cumplir con normalidad.

$$(1455 * Avion\_Franja\_Pista\_i_{ijk}) + i_j * Avion\_Franja\_Pista_{ijk} - inicio_g * Avion\_Franja\_Pista_{bgp} >= 16;$$
$$\forall i, b \in Avión, \ \forall j, g \in Franja, \ \forall k, p \in Pista, \ i \neq b, \ i_j > i_g$$

Esta restricción nos asegura que dos aviones no estarán ocupando slots consecutivos. Para esto, utilizamos la matriz opuesta de Avión_Franja_Pista para que cuando nuestra matriz valga 1, la opuesta valga 0 y viceversa. Empezamos multiplicando el valor de las casillas de esta matriz por un número grande para que podamos satisfacer la restricción en caso de que no esté ocupado el slot. En las operaciones siguientes, simplemente restamos las horas de inicio de las franjas de dos aviones y comprobamos que dicho valor es mayor de 15. Si es menos, significa que ambas franjas son consecutivas y por tanto no se debe pasar la restricción.

$$Avion\_Franja\_Pista_{ijk} + Avion\_Franja\_Pista\_i_{ijk} \ = \ 1; \ \forall i \in Avión, \ \forall j \in Franja, \ \forall k \in Pista$$

La última de las restricciones es la que nos asegura que Avión_Franja_Pista_i es la matriz opuesta de Avión_Franja_Pista, ya que la suma de sus casillas correspondientes es 1.

$$x_{ijk} \geq 0; \ \forall i \in Avión, \ \forall j \in Franja, \ \forall k \in Pista \, .$$

UNIFICACIÓN DE LOS DOS MODELOS

Cabe recalcar que para unificar ambas partes, sólo debemos modificar la función objetivo, que acaba resultando en la resta de ambas funciones anteriores, para obtener el beneficio total de ambas partes:

$$z = (\sum_{j=Avión} \sum_{i=Tarifa} x_{ij} \cdot c_i) \ - \ (\sum_{i=Avión} \sum_{j=Franja} \sum_{k=pista} Avión\_Franja\_Pista_{ijk} * c_i * (i_j - p_i))$$

PARTE 3

ANÁLISIS DE LOS RESULTADOS PARTE 1

Los resultados obtenidos con el modelo descrito arriba han sido:

	Estándar	Leisure	Business
Nº billetes Avion 1	37	23	30
Nº billetes Avión 2	20	66	34
Nº billetes Avión 3	160	23	17
Nº billetes Avión 4	100	20	30
Nº billetes Avión 5	133	21	36

La venta de todos estos billetes proporcionará unos ingresos de 26190 € a la aerolínea, respetando todas las restricciones de peso de los aviones, completando todos los asientos disponibles y ofertando tantos asientos de leisure, business y estándar como están indicados.
La solución al problema es factible y única, hay una sola solución óptima.
Teniendo en cuenta los resultados anteriores, vamos a evaluar las restricciones para ver cúal es más restrictiva:

- Si no limitamos que el nº de billetes vendidos sea en un 60% billetes estándar, obtenemos 1040€ más de beneficios.
- Si eliminamos la restricción de carga de los aviones, obtenemos 20€ más que en el anterior caso, 1060€.
- Al quitar la restricción de límite de asientos obtenemos un beneficio de 84760€ más.
- Si no restringimos el mínimo de billetes leisure & business obtenemos un beneficio mucho mayor que en los anteriores casos, por lo que asumimos que esta restricción es la más restrictiva de las anteriores.

En cuanto al estudio de los valores de los datos:

- Si aumentamos el número de asientos de A1 a 900, obtenemos 15140€ más comparado con la solución inicial.
- Si disminuimos la capacidad de A2 a 799 obtenemos una solución infactible, en cambio, si lo ponemos a 800 sí que obtenemos una solución factible, por lo que este es su valor mínimo.

Estas han sido las valoraciones que hemos tenido en cuenta, ya que sobre todo hemos deseado ver los límites que puede tener el problema modificando algunos de sus parámetros y restricciones.

ANÁLISIS DE LOS RESULTADOS PARTE 2

Pista x Franjas

				A4	
			A3		
	A1				A5
A2					

El mínimo de dinero que podemos perder con esta combinación inicial de slots es de 4500€, teniendo en cuenta todas las restricciones anteriores, tales como la unicidad de aviones, aterrizajes en slots libres evitando los consecutivos, y que el aterrizaje esté entre valores factibles de las franjas. En definitiva, hemos obtenido una solución factible y única, por lo que este es el mínimo más efectivo que podemos obtener.

ANÁLISIS DE LOS RESULTADOS PARTE 1 y 2 UNIFICADAS

El resultado de combinar ambos modelos son los esperados, 21690 €, la resta del beneficio obtenido con el primer modelo uno, 26190 €, menos los gastos por retraso del segundo modelo, 4500 €.

Para ver cómo se comporta el problema con otros parámetros y datos, hemos realizado las siguientes pruebas:

- Si añadimos un sexto avión, con capacidad 100 asientos, 1500 de capacidad de equipaje, que desea aterrizar a las 9:40, con límite 10:15 y su coste/min 120€, se añade en la Pista 4, a slot 5, con lo que se cumplen todas las restricciones correctamente. El beneficio en este caso baja a 23030€.
- Si eliminamos la franja 3, que es la única que no está ocupada en ninguna de las pistas, el resultado es el mismo que el inicial, que es lo esperado.
- Si borramos la pista 1, el avión pasa a la pista 5, en la misma franja horaria, por lo que el beneficio tampoco cambia.
- Teniendo en cuenta las restricciones anteriores, eliminando la franja horaria 5, el avión cuatro se mueve a la siguiente franja. El beneficio baja a 17940€.

Por último vamos a deducir qué restricción es la más restrictiva de nuestro problema:

- Al quitar la restricción de que un avión se asigne a un slot libre, el beneficio es el mismo, aunque los aviones siguen asignados a la franja más eficiente.
- Si eliminamos la restricción que relaciona 1 solo avión con un solo slot, el beneficio sigue siendo el mismo, era de esperar, ya las franjas horarias óptimas restringen las posiciones de los aviones y ningún avión comparte hora de aterrizaje.

- Al eliminar la restricción de tener asignado un slot de tiempo a cada aterrizaje, la solución óptima es no aterrizar, por lo que no hay pérdidas por retraso.
- Eliminamos que dos aviones no puedan asignarse en slots consecutivos, el valor de la función objetivo sigue sin cambiar, de nuevo, podemos afirmar que lo que de verdad limita nuestro problema son las franjas horarias.
- Por último vamos a eliminar que las franjas están acotadas con las horas límite y de llegada de los aviones, con esto, vemos que los resultados difieren mucho de los originales, con lo que podemos llegar a la conclusión de que estas restricciones son las que más condicionan a este problema.

Respondiendo al caso de que el avión 1 llevase 20 minutos de retraso, es decir que pasa de aterrizar a las 9:10 y hora límite 10:15, a 9:30 y 10:35 respectivamente. En este caso el avión 1 pasa a aterrizar en la siguiente franja, manteniéndose el resto de aviones franja igual, pero en el avión 2 ahora podrá pasar a la pista 3 al estar a un slot de distancia.
Para modelar este retraso, deberíamos añadir una nueva variable con el retraso de cada avión antes de llegar al aeropuerto, y restar dicho coste por retraso en vuelo a la función objetivo, igual que hacemos con el coste de retraso en pista.

				A4	
			A3		
A2		A1			A5

VENTAJAS Y DESVENTAJAS DE LAS HERRAMIENTAS USADAS.

Las herramientas utilizadas en esta práctica han sido LibreOffice y glpk.
En cuanto a LibreOffice, opinamos que, aunque sea un software que pueda servir para resolver ciertos problemas de programación lineal, no es la más óptima, o, al menos, no la más pensada para ello, ya que está centrado en más funcionalidades que sólo esa, y puede ocasionar ciertas restricciones. Por otro lado, es un programa muy accesible y fácil de usar, ya que tiene un formato de hoja de cálculo, con la que la mayoría de personas ya ha trabajado, por lo que es muy fácil familiarizarse con su uso.
Respecto a glpk, es una herramienta más 'profesional' o al menos pensada para este tipo de problemas, por lo que da la sensación de que le estamos dando el uso correcto. En cuanto a sus desventajas, opinamos que es un sistema de resolución algo tosco, a base de comandos, y se echa en falta algún recurso gráfico que ayude a vislumbrar el problema de una mejor manera.

CONCLUSIONES

En esta práctica hemos podido ver cómo podemos codificar un problema de programación lineal con diferentes tipos de software, además de poder modificar sus parámetros para entender mejor cómo funciona el problema y sus iteraciones. Por otro lado, ha sido bastante complicado poder recrear las restricciones acertadas en MathProg para la segunda parte del problema, aunque el resto de la práctica ha sido más llevadera en cuanto a dificultad. Una de las partes más interesantes ha sido la del análisis de los resultados, ya que ahí es donde de verdad hemos podido ‘trastear’ el problema y entenderlo mejor.