

# **HEURÍSTICA Y OPTIMIZACIÓN**

## **CSP Y HEURÍSTICAS**

Grupo 80

*Fecha de entrega: 15 de diciembre de 2022*

*Jaime Sanz Fraile ([100432147@alumnos.uc3m.es](mailto:100432147@alumnos.uc3m.es))*

*Ignacio Marín de la Bárcena Pérez ([100432039@alumnos.uc3m.es](mailto:100432039@alumnos.uc3m.es))*

*git: [https://github.com/100432039/proyecto2\\_heuristica\\_100432039\\_100432147](https://github.com/100432039/proyecto2_heuristica_100432039_100432147)*

# Índice

<b>INTRODUCCIÓN</b>	<b>3</b>
<b>MODELADO CSP</b>	<b>3</b>
<b>MODELADO BÚSQUEDA</b>	<b>6</b>
<b>ANÁLISIS DE LOS RESULTADOS</b>	<b>7</b>
<b>CONCLUSIÓN</b>	<b>8</b>

## INTRODUCCIÓN

El objetivo de esta práctica es aprender a modelar problemas aplicando diferentes técnicas: problema de satisfacción de restricciones y aplicando algoritmos de búsqueda. Y resolverlos mediante el uso de aplicar la librería python-constraint diferentes: *Python constrain* y el empleo del algoritmo  $A^*$  en cualquier lenguaje (su implementación se llevará a cabo en *Python*).

Para ello, se plantean varios problemas relacionados con la asignación de asientos dentro de un autobús y la asignación de posiciones dentro de una fila. Los cuales habrá que modelar y resolver mediante las herramientas previamente mencionadas respectivamente. En primer lugar, se modela la asignación de los asientos dentro de un autobús cuya salida servirá de entrada para la asignación de puestos dentro de la fila y la correcta resolución por el uso de algoritmos de búsqueda. Para el segundo problema que se plantea, se aplicaran dos heurísticas a la salida del problema anterior por medio del uso de la herramienta anteriormente descrita.

Por último, se analizarán los datos obtenidos, así como el correcto funcionamiento de las restricciones impuestas en el modelo mediante el cambio de los datos para ambos problemas. Además, se discutirán las ventajas y desventajas que existen entre las heurísticas aplicadas utilizadas en esta práctica.

## MODELADO CSP

En el primer problema que se plantea se nos pide asignar a los alumnos un asiento siguiendo una serie de restricciones, estas restricciones están impuestas sobre una serie de características de los alumnos. Los alumnos pueden ser de 1º ciclo o de 2º ciclo, pueden ser conflictivos o no, pueden ser personas o con movilidad de reducida o no y pueden tener hermanos o ser hijos únicos, sobre estas características planteadas se rigen las restricciones del problema:

- Los alumnos con movilidad reducida deben sentarse solos.
- Los alumnos conflictivos que no se pueden sentar cerca de uno de movilidad reducida ni cerca de otro alumno conflictivo.
- Si dos alumnos son hermanos estos se sientan juntos ya sean de distinto ciclo o conflictivos. En caso de ser de distinto ciclo el mayor será el que ocupará el lugar del pasillo y siempre en la zona del bus asignada a los del ciclo 1.
- El autobús cuenta con 32 asientos siendo necesario asignar a cada alumnos un asiento y que sea único para ese alumno.

29	25	21	17	puerta	13	9	5	1	chofer
30	26	22	18		14	10	6	2	
PASILLO									
31	27	23	19		15	11	7	3	
32	28	24	20	puerta	16	12	8	4	puerta

**“Imagen 1:Dibujo de las distintas zonas del autobús”**

Viendo la imagen del autobús vemos los posibles dominios que posteriormente aplicaremos para la resolución del problema:

- Los asientos azules son asientos designados para alumnos con movilidad reducida.
- Los 16 asientos de la parte de adelante son asientos asignados para alumnos del primer ciclo.
- Los 16 asientos de la parte de atrás son asientos asignados para alumnos de segundo ciclo.

A la hora de resolver el problema mediante un problema de satisfacción de restricciones (csp), es necesario definir tanto las variables del problema, como sus dominios y las restricciones que las afectan. Comenzaremos presentando las variables del problema:

- Se disponen de 32 posibles variables como máximo (1 por cada asiento), dónde cada una tiene una serie de características, y que hacen referencia al identificador del alumno que se sube al autobús.

$$A_{M,C,I,H} \in [1, 32]$$

- Las características son las siguientes:
  - M: Hace referencia a si un alumno tiene movilidad reducida o no.  
 $M \in \{X, R\}$ , tal que X si no es de movilidad reducida y R si lo es.
  - C: Referente si a un alumno es conflictivo o no.  
 $N \in \{X, C\}$ , tal que X si no es conflictivo y C si lo es.
  - I: Al ciclo al que pertenece el alumno.  
 $I \in \{1, 2\}$ , tal que 1 si pertenece al primer ciclo y 2 al segundo.
  - H: Identificador del alumno del que es hermano.  
 $H \in [0, 32]$ , dónde 0 es si no tiene hermano y el resto si tiene.

Una vez definidas las variables, estas actúan sobre un dominio que tiene el cual indica los posibles asientos a los que pueden ser asignados. El dominio general con características sin predefinir es el siguiente (Variables sin saber si son conflictivas, de movilidad o su ciclo) :

$$D = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32\}$$

Una vez analizado el dominio general sobre las variables se extraen diferentes restricciones sobre el dominio, es decir dominios que afectan a un determinado tipo de variables siendo estos los siguientes:

- Dominio de alumnos con movilidad reducida y que pertenecen al primer ciclo.  
 $D_1 = \{1, 2, 3, 4, 13, 14, 15, 16\}$
- Dominio de alumnos con movilidad reducida y que pertenecen al segundo ciclo.  
 $D_2 = \{17, 18, 19, 20\}$
- Dominio de alumnos del primer ciclo.  
 $D_3 = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16\}$
- Dominio de alumnos del segundo ciclo.  
 $D_4 = \{17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32\}$
- Dominio de alumnos con un hermano de menor ciclo  
 $D_5 = \{2, 3, 6, 7, 10, 11, 14, 15\}$

Finalmente, para completar el modelado en formato *csp* (problema de satisfacción de restricciones), una vez establecidas las variables y sus dominios. Es necesario definir las diferentes restricciones que afectan a cada una de las variables con el objetivo de llegar a obtener una solución válida del problema. Las restricciones son las siguientes:

- Asiento próximo al de movilidad vacío.

$$R_{A_{R,N,I,H}A_{M,N,I,H}} = \{(a,b): a \in D_i, b \in D_j, a \neq b, a//2 \wedge a - 1 \neq b, \sim a//2 \wedge a + 1 \neq b\}$$

- $i \in \{1, 2\}$
- $j \in \{3, 4\}$
- Si  $i = 1 \Leftrightarrow j = 3$ , si  $i = 2 \Leftrightarrow j = 4$

- Hermanos del mismo ciclo se sienten juntos ( $H = H$ ).

$$R_{A_{M,N,1,H}A_{M,N,1,H}} = \{(a,b): a \in D_i, b \in D_i, a \neq b, a//2 \wedge a - 1 = b, \sim a//2 \wedge a + 1 = b\}$$

- $i \in \{3, 4\}$

- Hermano de movilidad reducida con el resto (conflictivo o de movilidad normal) se sienten en la misma zona ( $H = H$ ).

$$R_{A_{R,N,I,H}A_{M,N,I,H}} = \{(a,b): a \in D_i, b \in D_j, a \neq b\}$$

- $i \in \{1, 2\}$
- $j \in \{3, 4\}$
- Si  $i = 1 \Leftrightarrow j = 3$ , si  $i = 2 \Leftrightarrow j = 4$

- Los dos hermanos tienen movilidad reducida ( $H = H$ )

$$R_{A_{R,N,I,H}A_{R,N,I,H}} = \{(a, b): a \in D_1, b \in D_1, a \neq b\}$$

$$- i \in \{3, 4\}$$

- Hermanos de distinto ciclo ( $H = H$ ) -> En el código realizado a partir del dominio.

$$R_{A_{M,N,1,H}A_{M,N,2,H}} = \{(a, b): a \in D_3, b \in D_5, a \neq b, a//2 \wedge a - 1 = b, \sim a//2 \wedge a + 1 = b\}$$

- Conflictivo lejos de cualquier otro conflicto (a menos que sean hermanos) o de movilidad.

$$R_{A_{M,C,I,H}A_{R,N,I,H}} = \{(a, b): a \in Q, b \in D_i, a \neq b, a \pm 1 \neq b, a \pm 3 \neq b, a \pm 4 \neq b, a \pm 5 \neq b\}$$

$$- Q \in \{2, 3, 6, 7, 10, 11, 14, 15, 18, 19, 22, 23, 26, 27, 30, 31\}$$

$$- i \in \{3, 4\}$$

$$R_{A_{M,C,I,H}A_{R,N,I,H}} = \{(a, b): a \in Q, b \in D_i, a \neq b, a + 1 \neq b, a - 3 \neq b, a \pm 4 \neq b, a + 5 \neq b\}$$

$$- Q \in \{1, 5, 9, 13, 17, 21, 25, 29\}$$

$$- i \in \{3, 4\}$$

$$R_{A_{M,C,I,H}A_{R,N,I,H}} = \{(a, b): a \in Q, b \in D_i, a \neq b, a - 1 \neq b, a + 3 \neq b, a \pm 4 \neq b, a - 5 \neq b\}$$

$$- Q \in \{4, 8, 12, 16, 20, 24, 28, 32\}$$

$$- i \in \{3, 4\}$$

## MODELADO BÚSQUEDA

Para el segundo problema la administración piensa cómo asignar las distintas posiciones a los alumnos dentro de una fila con el objetivo de disminuir el tiempo de subida al autobús. Para ello han calculado el tiempo que tarda en subir cada alumno al autobús así como afectan los alumnos al resto de sus compañeros (conflictivos y de movilidad).

De este modo, un alumnos que fuese conflictivo aumentaría el tiempo tanto a la persona que tiene delante como a la de detrás en dos tiempos. A su vez, aumentará al doble el tiempo de tiempo de todas aquellas personas que se sientan en asientos cuyo número es mayor. De la misma forma, un alumno con movilidad reducida necesitará ayuda para subir al autobús haciendo que tanto él como el siguiente suban a la vez y tarden tres tiempos.

Esto provoca que el problema inicial se represente de la siguiente manera:

- Estado Inicial:
  - Salida del problema anterior: {'3XX': 11, '1CX': 12, '6XX': 15, '5XX': 16, '8XR': 18, '4CR': 20, '2XX': 31, '7CX': 32}.
  - Lista vacía que indica el orden de subida al autobús []
- Estado final: Aquel en el que el orden de lista vacía (la que indica el orden de subida), minimice el tiempo de subirse al autobús.

Una vez declarados el estado inicial y el final procedemos definir los operadores, precondiciones y efectos que resuelven el problema:

Operadores	Precondiciones	Efectos
Insertar (Fx)	$F_x = A_i \wedge i \in [1, 32] \wedge$ $A_i \in \text{Salida pro. anterior}$ $\wedge A_i \notin \text{Lista vacía}$	Lista vacía $[A_i]$ $[\text{Salida pro. anterior}] - [A_i]$

Respecto al coste ( $g(n)$ ) de aplicar el operador será igual al tiempo que tarda en subir ese alumno al autobús, es decir,  $g(n) = t$ . Por otro lado, se deducen dos heurísticas por medio de usar relajación de restricciones. La primera en función al número de alumnos que quedan por insertar en la fila. Y la segunda, el tiempo total de los alumnos que quedan por meter y que son afectados por sus compañeros. Las expresiones matemáticas correspondientes a dichas heurísticas son las siguientes:

$$\rightarrow h(n) = \sum_{i=1}^{\text{len}[\text{fichero entrada}]} [A_i]$$

$$\rightarrow h(n) = \sum_{i=1}^{\text{len}[\text{ficher entrada}]} [t_{A_i}]$$

Finalmente, se hará uso de una tercera heurística con el objetivo de comparar el rendimiento con las dos heurísticas descritas en el párrafo anterior. Esta tercera heurística, será el algoritmo *Dijkstra*.

## ANÁLISIS DE LOS RESULTADOS

Se han ejecutado varias pruebas para comprobar el correcto funcionamiento y aplicación de las distintas restricciones buscando probar un caso de cada una del programa obteniéndose los siguientes resultados para las siguientes pruebas (ficheros denominados alumnosX.txt, donde X es el número de la prueba):

- alumnos1: No hay ningún alumnos. Resultado esperado 0 soluciones. Resultado obtenido 0 soluciones.
- alumnos2: 6 conflictivos en la zona 1. Resultado esperado que no da solución. Resultado obtenido no de solución
- alumnos3: 3 personas de movilidad reducida en el ciclo 2. Resultado esperado que no da solución. Resultado obtenido 0 soluciones.
- alumnos4: Hermanos conflictivos. Resultado esperado de solución. Resultado obtenido solución.
- alumnos5: 5 personas de movilidad reducida en el ciclo 1. Resultado esperado que no da solución. Resultado obtenido 0 soluciones
- alumnos6: Hermanos de distinto ciclo y uno de ellos conflictivo. Resultado esperado que de solución. El resultado obtenido da solución.

Una última prueba sería realizar la ejecución para 33 alumnos, no obstante; esta no se ha llevado a cabo debido a que el tiempo de computación era bastante elevado, invirtiendo una gran cantidad de tiempo sin obtener el resultado esperado de que no existe solución.

## CONCLUSIÓN

A lo largo de este proyecto hemos consolidado todos los conocimientos adquiridos en la asignatura permitiéndonos ver una aplicación en el mundo real. A su vez, hemos aprendido a usar aplicar nuevas herramientas (el algoritmo  $A^*$  y la librería *python-constraint*) con las que resolver ejercicios de restricciones como de búsqueda que esperamos poder utilizar durante nuestra etapa laboral.

Aun así, queremos recalcar, las grandes dificultades que nos hemos encontrado tanto a la hora de modelar ambos ejercicios como en su posterior implementación en las tecnologías previamente nombradas. Finalmente, indicar que no se ha podido implementar la segunda parte en código.



