

INTELIGENCIA ARTIFICIAL

2 Grado en Ingeniería Informática

RELACION DE PROBLEMAS 2. Métodos de Búsqueda Heurística

1. Se han descubierto A fuentes de contaminación en un parque natural y se quieren colocar B (donde $B < A$) aparatos de descontaminación para mejorar la situación. Para ello se dispone de un mapa del parque que indica la posición de la estación de trenes donde se han almacenado todos los aparatos y de los A lugares donde se necesita colocar los aparatos de descontaminación. Además también se dispone del nivel de contaminación que hay alrededor de cada fuente, de un mapa de los desplazamientos (dirigidos) posibles de los aparatos en el territorio y del coste de cada desplazamiento. Cada aparato puede eliminar por completo la contaminación de una fuente, independientemente de su nivel. El objetivo es colocar los aparatos de manera que se minimice la contaminación total en el parque y el coste del recorrido (suma de desplazamientos) que harán los aparatos en el sentido “estación \rightarrow fuente de contaminación”. En los siguientes apartados se proponen diferentes alternativas para algunos de los elementos necesarios para plantear la búsqueda (solución inicial, operadores, función heurística). Hay que comentar la solución que se propone respecto a si es correcta, es eficiente, y es mejor o peor en comparación con otras alternativas. Y hay que justificar todas las respuestas.
 - a) Se plantea solucionar el problema mediante un método de escala, partiendo de una solución inicial sin ningún aparato y con un operador que coloca un aparato en una fuente de contaminación determinada, controlando que el número de los aparatos colocados sea como máximo B .
 - b) Se plantea solucionarlo mediante un método de escalada, partiendo de una solución inicial con B aparatos colocados aleatoriamente, y utilizando como función heurística la suma de los costes de desplazamiento de la estación a cada una de las B fuentes.
 - c) Se plantea solucionarlo mediante un método de escalada, partiendo de una solución inicial con B aparatos colocados aleatoriamente, y utilizando como función heurística la suma de los costes mínimos de los recorridos “estación \rightarrow fuente de contaminación” multiplicada por la suma de los niveles de contaminación correspondientes a los B aparatos.
 - d) Se plantea solucionarlo mediante un método de escalada, partiendo de una solución inicial alcanzada colocando los B aparatos ordenadamente según el coste mínimo “estación \rightarrow fuente de contaminación” y empezando con el que tiene coste menor. Se plantea como operador mover un aparato a cualquier fuente cuyo producto de “coste mínimo estación \rightarrow fuente” por “nivel de contaminación” sea menor que el actual.

e) Se plantea solucionarlo mediante algoritmos genéticos: se usan individuos de A bits y como población inicial se generan n individuos donde en cada uno hay exactamente B bits a 1. La función de idoneidad es la suma de los costes mínimos “estación → fuente de contaminación” más la contaminación total residual del parque multiplicada por una constante. Como operadores se usan los habituales de cruce y mutación.

2. Tenemos una pequeña flota de C camiones que utilizamos para repartir mercancías y cada día tenemos que determinar qué ruta ha de seguir cada camión para abastecer un conjunto de ciudades por todo el país. El objetivo es que todos los camiones acaben la jornada aproximadamente a la misma hora, por lo que el número de kilómetros que ha de recorrer cada camión ha de ser muy parecido, y que recorran en total el mínimo número de kilómetros. Disponemos de un mapa de carreteras que nos dice la distancia en kilómetros entre cada ciudad donde hemos de dejar nuestra mercancía, suponemos que todos los camiones parten de la misma ciudad y han de volver a ella al final del día, cargan al principio de la jornada todo lo que han de repartir, han de pasar sólo una vez por cada ciudad. Una posible solución a éste problema se puede obtener mediante el uso de algoritmos de búsqueda local. En los siguientes apartados se proponen diferentes alternativas para algunos de los elementos necesarios para plantear la búsqueda (solución inicial, operadores, función heurística,...). Comenta muy brevemente la solución que se propone respecto a si es correcta y si es mejor/peor respecto a otras alternativas posibles. Justifica tus respuestas.

a) Usar un método de escalada. Como solución inicial asignamos al azar a cada camión un número aproximadamente igual de ciudades, recorriéndolas en orden también al azar. Como operadores usamos el intercambiar dos ciudades entre los recorridos de dos camiones e intercambiar las posiciones de dos ciudades en el recorrido de un camión. La función heurística es la siguiente:

$$h(n) = \sum_{i=1}^C \left(\frac{LR_i}{\sum_{j=1}^C LR_j} - \frac{1}{C} \right)$$

donde LR_i es la longitud del recorrido del camión i.

b) Usar un método de escalada. Como solución inicial asignamos todas las ciudades a un camión, estableciendo el recorrido inicial mediante una estrategia que determine la ciudad siguiente con un criterio de menor coste. Como operadores usamos el mover una ciudad del recorrido de un camión a otro e intercambiar las posiciones de dos ciudades en el recorrido de un camión. La función heurística es la siguiente:

$$h(n) = \prod_{i=1}^C LR_i$$

- c) Usar un método de escalada. Como solución inicial escogemos las C ciudades más cercanas a la ciudad origen como la primera ciudad a visitar por cada camión, como segunda ciudad en el recorrido de cada camión escogemos la más cercana a la primera que no esté ya asignada, y así sucesivamente hasta asignar todas las ciudades. Como operadores usamos el mover una ciudad del recorrido de un camión a otro, intercambiar las posiciones de dos ciudades en el recorrido de un camión e intercambiar dos ciudades entre los recorridos de dos camiones. La función heurística es la siguiente:

$$h(n) = \frac{C(C-1)}{2} - \sum_{i=1}^C \sum_{j=i+1}^C \frac{LR_i}{LR_j}$$

- d) Usar algoritmos genéticos. Donde cada ciudad está representada por tantos bits como sean necesarios para codificar el valor C , la tira de bits contiene concatenados los bits de todas las ciudades, es decir, representamos en la tira de bits el número del camión que ha de recorrerla. Como operadores utilizamos los operadores habituales de cruce y mutación.

3. El canal musical de televisión XL Recordings desea realizar una planificación del contenido a transmitir durante un día. Se ha decidido que durante un día se pueden emitir 200 videos y quieren poder determinar qué videos han de emitirse y en qué orden. La cadena dispone de una videoteca con V videos de entre los que elegir. Para cada video conocemos los ingresos por publicidad que genera su emisión y la popularidad del grupo musical del video (valor de 1 a 5). El objetivo de la cadena es maximizar los ingresos por publicidad, evitando que el número de videos con cierto valor de popularidad sea superior al 40 %. Para no aburrir a la gente, no debemos emitir ningún video mas de una vez, tampoco debemos poner demasiado juntos los videos de la misma popularidad. En los siguientes apartados se proponen diferentes alternativas para algunos de los elementos necesarios para plantear la búsqueda (solución inicial, operadores, función heurística). Hay que comentar la solución que se propone respecto a si es correcta, es eficiente, y es mejor o peor en comparación con otras alternativas. Y hay que justificar todas las respuestas.

- a) Usar un método de escalada. Como solución inicial elegimos 40 videos de cada popularidad y los colocamos aleatoriamente en la secuencia de emisión. Como operador intercambiar el orden de emisión de dos videos. Como función heurística usamos

$$h(n) = \sum_{i=1}^{200} IngresosPublicidad_i$$

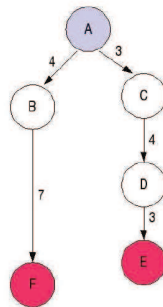
Donde $IngresosPublicidad_i$ es la cantidad que se ingresa por publicidad con el video i .

- b) Usar método de escalada. Como solución inicial elegimos los 200 videos que generan mayores ingresos por publicidad. Como operadores usamos cambiar un video de la solución por otro que no este emitido e intercambiar el orden de emisión de dos videos.
- c) Usar un método de escalada. Usamos como operador cambiar un video de la solución por otro que no este emitido, como función heurística:

$$h(n) = \sum_{i=1}^{200} f(IngresosPublicidad_i)$$

donde $f(IngresosPublicidad_i)$ vale 0 si el porcentaje de videos de la misma popularidad que el video i supera el 40 % o $IngresosPublicidad_i$ en caso contrario.

- d) Usar algoritmos genéticos. Supondremos que hacen falta b bits para codificar el identificador de un video. Representamos cada individuo como una tira de $200 \cdot b$ bits. Como población inicial, generar n individuos escogiendo 200 videos aleatoriamente. Como operadores utilizamos solamente el operador de cruce.
4. El grafo que se muestra en la figura describe un problema de búsqueda. Suponga que A es el estado inicial y que F y E son estados meta. Los arcos están etiquetados con el coste real de los operadores.

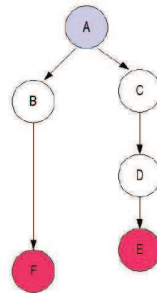


h es una función heurística, cuyos valores son

	h
A	8
B	6
C	6
D	5
E	0
F	0

- a) Desarrolle el árbol de búsqueda que genera el algoritmo A*. Indique el orden en que se expanden los nodos. ¿Cuál de los nodos meta se encuentra primero?

- b) ¿La función heurística es admisible? Argumente su respuesta.
5. Considere el problema de búsqueda del ejercicio precedente. Suponga que A es el estado inicial y que F y E son estados meta.



- a) Asigne los valores del coste real de los operadores y de la función heurística h , de modo que ésta resulte ser admisible.
- b) Desarrolle el árbol de búsqueda que genera el algoritmo A^* . ¿Se ha obtenido la solución óptima?
6. Considérese el problema del 3-puzzle, versión reducida del problema del 8-puzzle, en el que en un cuadrado 2×2 se disponen tres bloques (y por tanto un hueco) numerados del 1 al 3. Los estados inicial y final son, respectivamente:

	1
3	2

Estado inicial

1	2
	3

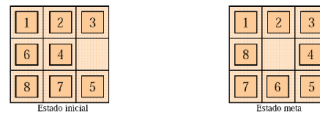
Estado final

Tomamos como operadores del problema el movimiento del hueco arriba, abajo, a la izquierda y a la derecha (en ese orden) y como coste de aplicar un operador el valor 1. Representar gráficamente los árboles correspondientes a los siguientes algoritmos de búsqueda.

- a) Búsqueda en profundidad.
- b) Búsqueda A^* con la heurística h_1 que cuenta la distancia Manhattan desde la posición del hueco a la posición del hueco en el estado final.
- c) Búsqueda por primero el mejor con la heurística h_2 que cuenta el número de bloques que no están en la posición que deben ocupar en el estado final.

En cada caso, anotar junto a cada nodo el orden en el que se analiza y su heurística o coste más heurística, cuando sea relevante. A igualdad de valoración, resolver los conflictos escogiendo el nodo que más tiempo lleve en la cola de espera. ¿Es h_1 admisible? ¿Es h_2 admisible?

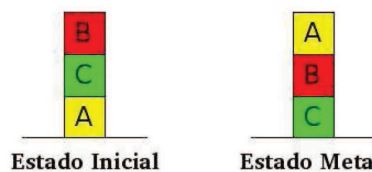
7. Considere el 8-puzzle cuyo estado inicial y estado meta se muestran en la siguiente figura:



Desarrolle el árbol de búsqueda que expande el algoritmo A*, utilizando las siguientes heurísticas. Evite ciclos generales, indique el orden de expansión de los estados y muestre en cada paso los valores de f, g y h. Suponga que el coste de cada operador es 1.

- Heurística 1: $h_1(n)$ = número de piezas mal colocadas
- Heurística 2: $h_2(n)$ = suma de distancia Manhattan
La distancia Manhattan de una pieza es la suma de las distancias vertical y horizontal a su posición final.
- ¿Cuál de las heurísticas expande menos nodos? ¿Por qué? ¿Puede sacar una conclusión general con respecto a la calidad de las funciones heurísticas?

8. Considere el problema de los bloques cuyo estado inicial y estado meta se muestran en la siguiente figura:

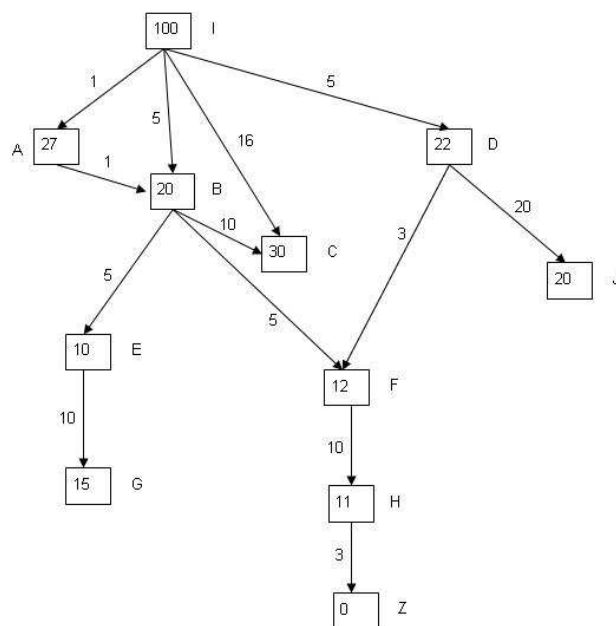


Desarrolle el árbol de búsqueda que expande el algoritmo A*, utilizando la siguiente heurística:

$h(n)$ = número de bloques mal colocados

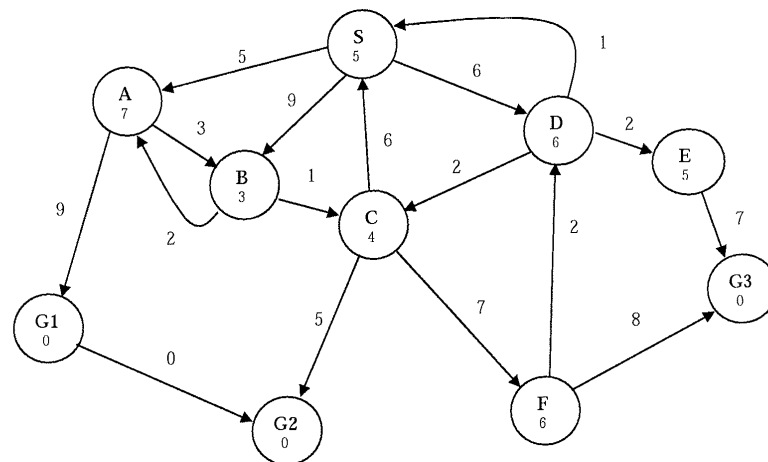
Filtre los ciclos simples, indique el orden de expansión de los estados y muestre en cada paso los valores de f , g y h . Suponga que el coste de cada operador es 1.

9. Dado el siguiente grafo, representando un espacio de estados, utilizar el algoritmo A^* para obtener la solución al problema. En dicho grafo considerar como función h la que se indica dentro de cada nodo, y para el cálculo de la función g considerar el coste que aparece en el arco. El nodo inicial es I y hay un solo nodo meta en este caso es Z .



- Detallar la secuencia de nodos de ABIERTOS que ha sido regenerados, hacerlo en el mismo orden en que el algoritmo A^* los haya vuelto a generar y no cuenta la primera vez que se insertan en ABIERTOS.
- Detallar la secuencia de nodos de CERRADOS que ha sido regenerados, hacerlo en el mismo orden en que el algoritmo A^* los haya vuelto a generar y no cuenta la primera vez que se insertan en CERRADOS.
- Mostrar el gráfico final con la solución obtenida, los valores g y h finales para cada nodo, junto con los enlaces a los hijos y enlace al mejor padre.
- ¿Es la función heurística utilizada en el problema admisible?

10. Considerar el espacio de búsqueda siguiente



en donde S es el nodo de inicio y G1, G2 y G3 son nodos que satisfacen la condición objetivo. Los arcos están etiquetados con el costo asociado al operador correspondiente. El costo estimado desde un nodo a la solución se muestra dentro del mismo nodo. Utilizar el algoritmo A* para resolver el problema. Indicar con claridad los siguientes datos:

- Nodo objetivo alcanzado.
- Listar la secuencia de nodos en el orden concreto en el que salen de ABIERTOS.

Importante: cuando se produzca un empate entre los nodos de ABIERTOS siempre seleccionarlos en orden alfabético.

11. Aplicar el algoritmo A* para hallar el camino con menor distancia que una las ciudades 1 y 8. Las distancias por carretera entre las distintas ciudades vienen especificada por la siguiente tabla:

	1	2	3	4	5	6	7	8
1		200						
2	200		150	350	450			
3		150			400	225		
4		350			300			
5		450	400	300			250	
6			225				450	
7					250	450		125
8							125	

Además, se dispone de la distancia aérea en línea recta que existe entre todas las ciudades con la ciudad de destino:

1	2	3	4	5	6	7
800	650	500	650	325	375	125

12. Considérese una modificación del algoritmo A^* en el que en lugar de usar $f(n) = g(n) + h(n)$ para ordenar la cola de abiertos se usa $f_w(n) = (1 - w)g(n) + wh(n)$, siendo w un número real constante entre 0 y 1. ¿Qué algoritmo de búsqueda estaríamos aplicando si tomamos $w = 0$? ¿Y $w = 0.5$? ¿Y $w = 1$?