

INTELIGENCIA ARTIFICIAL

Algoritmos de búsqueda

*Búsqueda A^**

J. Marcos Moreno-Vega

OBJETIVO:

Proponer, implementar y evaluar búsquedas A* para encontrar el camino mínimo entre dos vértices de un grafo.

TAREAS:

Además de las tareas descritas en el presente documento, los alumnos tendrán que realizar las modificaciones que se planteen durante la corrección de la práctica. Asimismo, tendrán que responder a un cuestionario de preguntas tipo test sobre los contenidos teóricos de los algoritmos de búsqueda.

CORRECCIÓN:

Semana del 4 al 8 de noviembre.

EVALUACIÓN:

Código fuente y memoria: hasta 4 puntos. Si el día de la corrección falta algún código o este es incorrecto, la práctica se calificará como No apta.

Modificación propuesta el día de la corrección: hasta 4 puntos.

Cuestionario sobre los contenidos teóricos: hasta 2 puntos.

LENGUAJE DE PROGRAMACIÓN:

Java o C++.

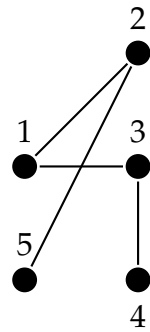
Problema del camino mínimo entre dos vértices de un grafo

Sea dado un grafo $G = (V, E)$, donde V es el conjunto de vértices y E es el conjunto de aristas ($|V| = n, |E| = m$). Cada arista $(i, j) \in E$ tiene asociada una distancia o coste $d(i, j)$. Se desea encontrar el camino mínimo que conecta el vértice origen v_0 con el vértice destino v_d .

Implementación

Las instancias del problema se suministrarán en un fichero de texto con el siguiente formato: en la primera fila se encuentra el número de vértices, n ; a continuación, se enumeran las distancias, $d(i, j)$, entre los pares de vértices. Se asume que las distancias son simétricas, es decir, que $d(i, j) = d(j, i)$, $\forall i, j \in V$. Además, $d(i, i) = 0$, $\forall i \in V$ y $d(i, j) = \infty$ si no hay una arista que conecte al vértice i con el vértice j .

Por ejemplo, si $n = 5$, el fichero de texto para el grafo de la figura 1(a) contendría los datos mostrados en la figura 1(b) (solo la primera columna; la segunda describe qué representa cada dato):



(a) Grafo

```

5      /* número de vértices
1.414  /* d(1,2) = d(2,1)
1.000  /* d(1,3) = d(3,1)
∞      /* d(1,4) = d(4,1)
∞      /* d(1,5) = d(5,1)
∞      /* d(2,3) = d(3,2)
∞      /* d(2,4) = d(4,2)
2.236  /* d(2,5) = d(5,2)
1.000  /* d(3,4) = d(4,1)
∞      /* d(3,5) = d(5,3)
∞      /* d(4,5) = d(5,4)

```

(b) Formato del fichero de datos

Figura 1: Grafo y su representación

La función heurística, $h(\cdot)$, se leerá de un fichero de texto que tiene el siguiente formato: en la primera fila se encuentra el número de vértices, n ; a continuación se enumeran los valores de la función heurística. Para el grafo de la figura 1(a), dicho fichero contendría los valores de la figura 2, si $v_0 = 1$ y $v_d = 5$.

```

5      número de vértices
1.000  /* h(1)
2.236  /* h(2)
1.414  /* h(3)
1.000  /* h(4)
0.000  /* h(5)

```

Figura 2: Formato del fichero con la función heurística

Tareas

- Diseñar e implementar una búsqueda A* para el problema del camino mínimo entre dos vértices de un grafo.

Debe poder indicarse, cómodamente, cuáles son los vértices origen y destino.

- b) Analizar el comportamiento de la búsqueda A* considerando diferentes funciones heurísticas.

Qué debe presentar el alumno

- a) Código fuente, debidamente comentado, y fichero ejecutable.
- b) Una memoria en formato pdf en la que se describan brevemente la búsqueda A* diseñada enumerando las estructuras de datos usadas y cualquier elemento necesario para comprender el diseño propuesto.
- c) La memoria debe incluir también tablas o gráficas de resultados que muestren el comportamiento de la búsqueda sobre diferentes instancias del problema. En la figura 3 se muestra el formato de estas tablas de resultados.

Se han considerado seis grafos (instancias ID_1, ID_2, \dots, ID_6), de diferentes tamaños, con varias combinaciones de vértices origen y destino. En la tabla se mostrará el camino mínimo para ir del vértice v_o al vértice v_d , su longitud y los nodos generados e inspeccionados por la búsqueda A* que usa la función heurística $h_1(\cdot)$.

Búsqueda A*. Función heurística $h_1(\cdot)$								
<i>Instancia</i>	<i>n</i>	<i>m</i>	<i>v_o</i>	<i>v_d</i>	<i>Camino</i>	<i>Distancia</i>	<i>Nodos generados</i>	<i>Nodos inspeccionados</i>
ID_1	10	5	1	10				
ID_1	10	5	2	7				
ID_2	10	10	1	8				
ID_2	10	10	3	9				
ID_3	10	15	1	6				
ID_3	10	15	3	7				
ID_4	20	10	7	10				
ID_4	20	10	1	8				
ID_5	20	20	2	9				
ID_5	20	20	3	9				
ID_6	20	50	1	5				
ID_6	20	50	4	10				

Figura 3: Búsqueda A*. Tabla de resultados