

- Trabajo Práctico 6 -

Parte 2. Inteligencia Artificial.

Resultado obtenido de Parte2_TP6.java

Problem	Depth	Gener.	Expand	Q.Size	MaxQS	tiempo ms
dfs.MC	11	25	11	3	4	6
A*.MC	11	28	13	1	3	1
BFS.MC	11	28	13	1	3	1
RBFS.MC	11	106	50	0	0	2
IDA*.MC	11	100	47	0	0	2
A*-15p	22	526	167	178	179	10
RBFS-15p	22	479	155	0	0	2
IDA*-15p	22	1386	449	0	0	4

Para el problema de Misioneros y Caníbales, se ha utilizado la heurística número 2.

Fichero Heuristic2.java:

$(h2 = 2 * (\text{misioneros a la izquierda} + \text{caníbales a la izquierda}) - f_{\text{orilla}})$ siendo $f_{\text{orilla}} = 0$ si la barca está a la izquierda y 1, si está a la derecha.

Por otra parte, para el problema del 15 puzzle, se ha implementado la heurística ManhattanHeuristicFunction15p, basada en la heurística Manhattan para la versión del 8 puzzle y adaptándola a este nuevo problema.

Con el fin de comprobar los algoritmos para el 15puzzle, se generan problemas aleatoriamente a profundidad alrededor de 20, como establece el enunciado. En este caso, al invocar a la función `generarTablero15p(22)`, se crea un tablero inicial del que partir en los algoritmos para el 15puzzle.

Algoritmo IDA*

A la hora de implementar el algoritmo de Iterative Deepening A Star Search (IDA*), me he basado en el código del algoritmo DLS (Depth Limited Search) y he ido adaptándolo, de forma que se pareciese al IDS (Iterative Deepening Search) en versión recursiva.

En lugar de realizar una búsqueda limitada por valor, he cambiado el límite de búsqueda por la función de evaluación del nodo inicial para priorizar los nodos.

Y en cada iteración de la búsqueda recursiva, se almacena el valor asociado al nodo descartado cuando ocurre un CutOff, es decir, incrementa el límite de profundidad basándose en el nodo con el menor valor de la función de evaluación que excede el límite actual. De esta forma, finalmente, tras ejecutar recursiveSearch, se obtiene el valor mínimo al que llega el algoritmo IDA*.

Este algoritmo combina las ventajas de dos algoritmos: la completitud de la búsqueda en profundidad y la eficiencia de la búsqueda A*.