



DEPARTAMENTO DE INFORMÁTICA
UNIVERSIDAD CARLOS III DE MADRID

Ingeniería en Informática

Inteligencia Artificial

Noviembre 2007

Hoja de Ejercicios 4: Búsqueda heurística y de dos agentes

Comentarios generales sobre los ejercicios

- Asumiendo que se conocen los contenidos teóricos, el tiempo estimado para realizar los ejercicios es de **2 horas**
- Describir las soluciones a los ejercicios de una manera lo más formal posible

Ejercicio 1

Aplicar el algoritmo de búsqueda A* sobre la representación del problema de *los misioneros y los caníbales* extraída de la anterior hoja de problemas. Especifica cuándo un nodo del árbol de búsqueda es expandido y cuándo es sólo generado. A modo de resumen, se describen los siguientes elementos:

1. Un estado es una terna (M_i, C_i, B, M_f, C_f) en la que:
 - $B \in [i, f]$ indica la posición de la barca, por lo que toma el valor i si está en el extremo inicial, o f si está en el final
 - $M_i, C_i, M_f, C_f \in [0, \dots, 3]$ indican el número de misioneros y caníbales que quedan en el extremo inicial y final del río, respectivamente
2. El estado inicial se representa como $(3, 3, i, 0, 0)$ y el final como $(0, 0, f, 3, 3)$
3. Se dispone de cinco operadores: $\text{Mover1C}(x,y)$, $\text{Mover2C}(x,y)$, $\text{Mover1M}(x,y)$, $\text{Mover2M}(x,y)$, y $\text{Mover1M1C}(x,y)$.
4. La heurística a utilizar es:

$$h_2(n) = 2 \times (C_i + M_i) - \text{orilla}(n) \quad (1)$$

donde $\text{orilla}(n)=1$ si en el estado n la barca está en la orilla i ($B = i$), y $\text{orilla}(n)=0$ si la barca está en la orilla final ($B = f$).

Ejercicio 2

Supongamos que tenemos un algoritmo de búsqueda de tipo *el mejor primero* que utiliza la siguiente función de evaluación: $f(n) = (2 - w) * g(n) + wh(n)$

- ¿Qué tipo de búsqueda hace cuando $w=0$? ¿y cuando $w=1$? ¿y para $w=2$?
- ¿Para qué valores de w el algoritmo es admisible?

Ejercicio 3

HARE & HOUNDS (Figura ??) es un juego de estrategia para 2 jugadores: *la presa* y *el cazador*. *La presa* decide los movimientos de la liebre mientras que *el cazador*, decide los movimiento de los perros. *La presa* gana el juego si consigue llegar de un extremo al otro del tablero. En cambio, *el cazador* gana si logra atrapar a la liebre, es decir si se alcanza una situación del juego en la que la liebre no puede realizar ningún movimiento. En cada turno cada jugador puede mover sólo una ficha y siempre a una de las casillas adyacentes. Las fichas *del cazador* no pueden retroceder. El primer turno corresponde siempre *al cazador*. Se pide:

- Indica posibles representaciones de los estados.
- Describe los operadores del juego.
- ¿Qué funciones de evaluación se te ocurren para programar un jugador automático inteligente de HARE & HOUNDS.

En <http://www.appletonline.com/JavaApplets/HoundsAndHare/> puede encontrarse un *applet* para jugar a HARE & HOUNDS.

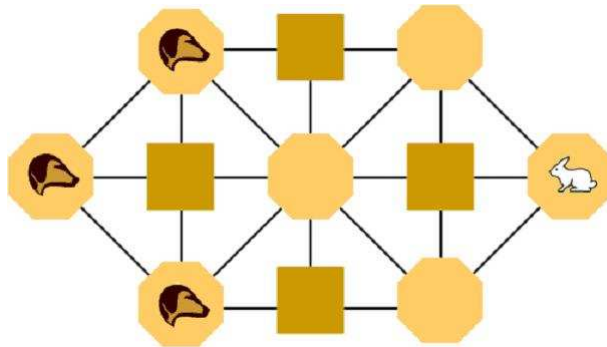


Figura 1: Posición inicial de la liebre y los perros en el juego de HARE & HOUNDS

Ejercicio 4

Se tiene un robot autónomo en una habitación cuadrada de 6x6 casillas (Figura ??). El robot es capaz de realizar desplazamientos verticales y horizontales y reconoce un obstáculo si esta en una de las casillas adyacentes. Suponiendo que el robot utiliza algoritmos de búsqueda para planificar sus movimientos contesta a las siguientes preguntas:

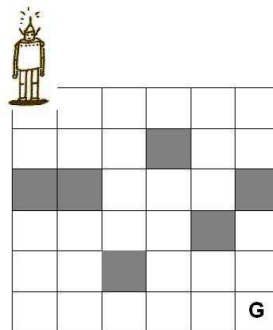


Figura 2: El Robot tiene que ir de una esquina de la habitación a la otra

- ¿Qué heurística podría utilizar el robot para guiarse desde su posición actual hasta el otro extremo de la habitación?

- ¿Qué camino tomaría el robot si su movimiento estuviese determinado por un algoritmo de búsqueda en escalada con la heurística del apartado anterior? Suponer que los estados sucesores se generan aplicando los operadores de desplazamiento en este orden: [derecha, abajo, izquierda, arriba]?
- Y siguiesen el orden:[arriba, izquierda, abajo, derecha] ¿Qué camino tomaría?
- Siguiendo el primer orden de operadores ¿Qué camino tomaría el robot si sus movimientos estuviesen guiados por un algoritmo de búsqueda en haz con $k=3$?

Ejercicio 5

Recorrer el grafo de la figura ?? utilizando los algoritmos de búsqueda heurística: *Escalada*, A^* e IDA^* , teniendo en cuenta los costes indicados en cada arco.

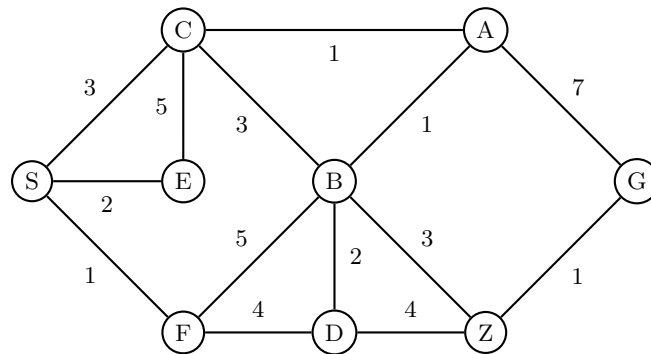


Figura 3: Espacio de estados

En todos los casos, indicar en qué orden se visitan los nodos, distinguiendo nodos generados de nodos expandidos. Tomar como estado inicial el nodo S y como único estado meta el nodo G.

Cada nodo del grafo tiene el valor heurístico descrito en la tabla ??

n	S	A	B	C	D	E	F	G	Z
$h(n)$	8	2	3	5	4	6	6	0	1

Cuadro 1: Tabla de valores heurísticos

Ejercicio 6

En algunas aplicaciones reales, el espacio de problemas se suele formular con un único estado inicial, s , y un conjunto arbitrariamente grande de estados finales o meta $\Gamma : \{t_1, t_2, \dots, t_n\}$. Considerando los espacios de problemas formulados de esta manera, y considerando únicamente el caso de espacios de estados no dirigidos, se pide:

- Sin modificar la función heurística que se esté empleando, ¿qué modificaciones deben hacerse al algoritmo de búsqueda A^* para que encuentre la secuencia óptima de operadores que transforma un único estado inicial, s , en uno cualquiera de varios estados meta $\Gamma : \{t_1, t_2, \dots, t_n\}$?