

## INTELIGENCIA ARTIFICIAL

### CURSO 2022-23

#### PRACTICA 2: Repertorio de preguntas para la autoevaluación de la práctica 2.

APELLIDOS Y NOMBRE	QUINTÍN MESA ROMERO		
GRUPO TEORÍA	A	GRUPO PRÁCTICAS	A1D

#### Instrucciones iniciales

En este formulario aparecen preguntas que requieren breves explicaciones relativas a cómo el estudiante ha hecho algunas partes de esa implementación y qué cosas ha tenido en cuenta.

**Enumera los niveles presentados en su práctica (Nivel 0, Nivel 1, Nivel 2, Nivel 3, Nivel 4):**

Nivel 0, Nivel 1, Nivel 2, Nivel 3

#### Nivel 1-Anchura para el agente sonámbulo

- (a) ¿En qué se diferencia desde el punto de vista de la implementación el algoritmo que has usado en este nivel en relación al del nivel 0? (enumera los cambios y describe brevemente cada uno de ellos)

En primer lugar, la diferencia más evidente es que el objetivo ha de ser alcanzado por el agente sonámbulo en lugar de por el jugador como se hace en el nivel 0 y por tanto, se ha implementado una anchura enfocada al sonámbulo (la solución se alcanza (en abiertos) cuando el sonámbulo al avanzar hacia adelante se topa con ella). Es por ello que en primer lugar, el jugador deberá avistar al sonámbulo para poder generar un plan de acción para este y pueda alcanzar el objetivo (se ha implementado una función que determina si el sonámbulo cae en el campo visual del jugador, a partir de las ubicaciones de ambos, dadas por el estado actual) y una vez dentro del campo visual del jugador, se generan los hijos del sonámbulo comprobándose si se ha alcanzado la solución. Si el sonámbulo no cae en el campo visual del jugador, se sigue buscando; se van añadiendo acciones pero del jugador al plan de acción (como en el nivel 0) hasta que se encuentra al sonámbulo en su campo visual.

#### Nivel 2-Dijkstra para el agente jugador

- (a) ¿Qué es propio de este nivel que no tuviste que tener en cuenta en los niveles anteriores? (enumera los cambios y describe brevemente cada uno de ellos y que han implicado en la implementación)

En este nivel se pretende encontrar un camino hacia el objetivo con acciones del jugador que minimice el consumo de batería aplicando el algoritmo de coste uniforme. Es por ello que en este nivel se ha tenido en cuenta el coste de las acciones que constituyen el plan de acción del jugador (se lleva la cuenta del coste acumulado de las acciones que se han realizado hasta el estado

actual). Se ha ampliado el concepto de nodo, añadiéndole un nuevo atributo: “coste” y se han creado unas estructuras en las que se ha implementado un operador de comparación entre estados y otro para comparar nodos. Estos comparadores los he necesitado para la creación de la lista de abiertos (que ahora es una cola con prioridad en la que se establece la prioridad de un elemento mediante el comparador de nodos que lo que hace es comparar los costes acumulados y en caso de ser iguales, compara por estados) y la de cerrados (que al igual que en los niveles 0 y 1 es un set de estados, pero es un conjunto ordenado, en el cual los elementos se ordenan por medio del comparador de estados que simplemente utiliza el operador < implementado en la definición del estado).

Por otra parte, el resto más o menos como antes, salvo que al generar los hijos, se acumula el coste en batería de la acción correspondiente a cada hijo. Para ello se ha creado una función que calcula el coste en batería de una determinada acción del jugador (o del sonámbulo para el nivel 3) en función del tipo de terreno sobre el que la realice, la acción que sea y si lleva los accesorios correspondientes (bikini o zapatillas). Además, como estamos aplicando coste uniforme, la solución la encontraremos en cerrados, en lugar de en abiertos como pasaba con la anchura de los niveles 0 y 1.

- (b) ¿Has incluido dentro del algoritmo de búsqueda usado en este nivel que si pasas por una casilla que da las zapatillas o el bikini, considere en todos los estados descendientes de él el sonámbulo o el jugador tiene las zapatillas y/o el bikini? En caso afirmativo, explicar brevemente cómo.

Sí, para ello, he tenido que actualizar el concepto de estado añadiendo unas variables booleanas para jugador y sonámbulo que toman el valor true cuando estos tienen el bikini/zapatillas puestos y false en caso contrario. He definido una función a la que he llamado “actualizarNodo” que a partir del estado actual, tomando las ubicaciones del jugador y del sonámbulo, en caso de estar sobre agua o bosque, pone la variable correspondiente a true y en caso de estar la variable complementaria también a true, se pone a false (esto es porque tanto jugador como sonámbulo no pueden tener dos objetos puestos a la vez; si se ha puesto el bikini porque está sobre agua y tiene las zapatillas puestas, tiro las zapatillas y lo mismo para el caso en que este en bosque y tuviera el bikini puesto).

### Nivel 3-A\* para el agente sonámbulo

- (a) ¿Qué diferencia este algoritmo del de Dijkstra que tuviste que implementar en el nivel anterior? (enumera los cambios y describe brevemente cada uno de ellos y que han implicado en la implementación)

En este algoritmo, aunque tenemos que encontrar el camino óptimo en consumo de batería para el sonámbulo hacia el objetivo, a diferencia del de Dijkstra, tenemos que utilizar una heurística (admisible). Es por ello que se ha de actualizar el concepto de nodo añadiéndole un atributo para la heurística y para la suma del coste acumulado y la heurística, además, se ha definido un comparador de dicha suma, para discernir qué nodo es más prioritario que otro a la hora de ser explorado en abiertos. Al generar los hijos, a parte de calcular el coste acumulado de batería, se calcula el valor de la heurística y se suma dicho valor con el coste acumulado. Se procede en cuanto a estructura del algoritmo igual que en el nivel 1, con la salvedad de lo antes mencionado y de que al igual que en Dijkstra, la solución se encuentra en cerrados.

- (b) Describe la heurística utilizada para resolver el problema

Como heurística he utilizado la distancia del máximo (o de Chebychev) entre el sonámbulo y el

objetivo. Esta consiste en el máximo entre el valor absoluto de la diferencia entre la fila del sonámbulo y del objetivo y el valor absoluto de la diferencia entre la columna del sonámbulo y la del objetivo.

#### Nivel 4-Reto (Max. Puntuación en misiones)

- (a) Haz una descripción general de tu estrategia general con la que has abordado este nivel. Indica bajo qué criterios es el jugador o el sonámbulo el que va al objetivo. Explica brevemente las razones de esos criterios.

- (b) ¿Qué algoritmo o algoritmos de búsqueda usas en el nivel 4? Explica brevemente la razón de tu elección.

- (c) ¿Bajo qué condiciones replanifica tu agente?

- (d) Explica el valor que le has dado a la casilla desconocida en la construcción de planes cuando el mapa contiene casillas aún sin conocer. Justifica ese valor.

- (e) ¿Has tenido en cuenta la recarga de batería? En caso afirmativo, describe la política usada por tu agente para proceder a recargar.

- (f) ¿Has tenido en cuenta la existencia de aldeanos y lobos para definir el comportamiento del agente? En caso afirmativo, describe en qué sentido los has tenido en cuenta.

- (g) Añade aquí todas los comentarios que desees sobre el trabajo que has desarrollado sobre este nivel, qué consideras que son importantes para evaluar el grado en el que te has implicado en la práctica y que no se puede deducir de la contestación a las preguntas anteriores.

## **Comentario final**

Consigna aquí cualquier tema que creas que es de relevancia para la evaluación de tu práctica o que quieras hacer saber al profesor.

He de confesar que me encuentro mucho más satisfecho con el trabajo realizado en esta práctica en cuanto a comprensión de conceptos y resultados obtenidos que en la práctica anterior. Me siento bastante orgulloso en el sentido de que he podido romper la barrera de la vergüenza y poder pedir ayuda ante cualquier dificultad. Me siento muy agradecido por la atención que he recibido.