

Trabajo Práctico N° 1: Búsqueda

Grupo 32

Emiliano Gioria – emigioria@hotmail.com
Esteban Javier Rebechi – estebanrebechi_5@gmail.com
Andrés Leonel Rico – andres.rico94@gmail.com.

Inteligencia Artificial, 2017, ISI, UTN - FRSF

Resumen. En este trabajo se presentan los resultados de aplicar algoritmos de búsqueda no informados e informados a un problema en el que un patrullero debe moverse de un lugar del mapa de una ciudad en el que se encuentra inicialmente, a otro en el que se sucede un incidente que debe resolver. En general la aplicación de los algoritmos fue exitosa, salvo en la aplicación del método de búsqueda en profundidad que, en la mayoría de las veces simuladas, entró en un bucle infinito debido a la presencia de calles doble mano y de cómo se definieron las acciones del patrullero.

1 Introducción

Este trabajo está dedicado a la aplicación de una estrategia de búsqueda de inteligencia artificial para la resolución de un problema en dónde un patrullero (el agente) debe trasladarse por las calles de una ciudad (el ambiente) para llegar de un punto inicial en el que se encuentra a otro en el que surgió un incidente. Recorrer la ciudad sin ningún tipo de percance o imprevisto conlleva un tiempo mínimo. Sin embargo, en la ciudad se presentan obstáculos en las intersecciones y calles que al ser atravesados incrementan la duración de trasladarse por estos lugares. Esos obstáculos aparecen por períodos específicos de tiempo. La duración total del trayecto se utiliza como medida para comparar las diferentes estrategias de búsqueda.

En la sección 2, se explica la solución conceptual del problema. En la sección 3, se muestran las pruebas realizadas a la solución planteada y los resultados de las mismas. Finalmente, en la sección 4 se presentan las conclusiones de este trabajo.

2 Solución

Para representar el agente patrullero y el ambiente, con sus respectivos estados, y la información que fluye entre ellos, se utilizó la herramienta Idem-IA, mientras que la codificación de la solución se basó en la extensión del framework FAIA dado por la cátedra de Inteligencia Artificial. En la figura 1 se muestra el diagrama Idem-IA de la solución. En el anexo A se puede ver el modelo utilizado para la implementación del

escenario. Y en el anexo B se representan las extensiones realizadas a FAIA para el desarrollo de la simulación.

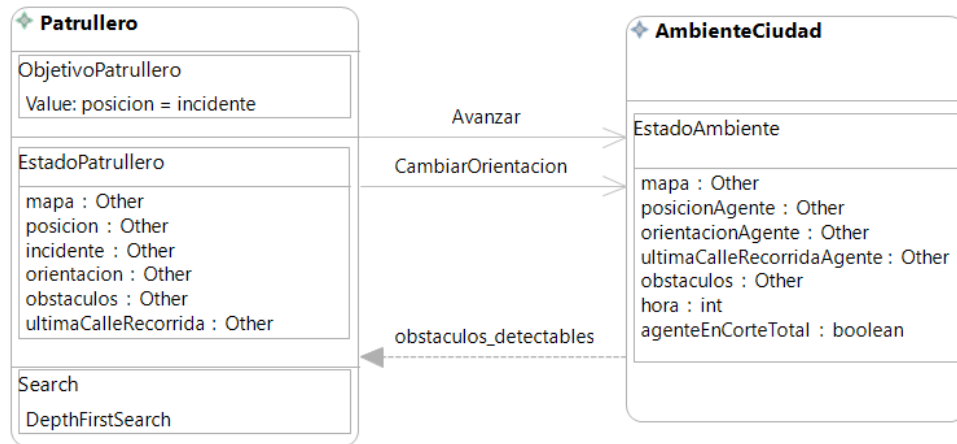


Figura 1. Diagrama Idem-IA de la solución planteada.

Estados

Como se muestra en la figura 1, el estado del agente patrullero incluye como componentes: el mapa de la ciudad por la cual debe transitar, su posición dentro del mismo, su orientación, que representa la siguiente calle por la que planea moverse, la posición del incidente al que necesita llegar, los obstáculos que conoce y la última calle recorrida por él. Asimismo, el estado del ambiente tiene: el mapa de la ciudad, la posición, orientación y última calle recorrida por el agente, todos los obstáculos en todos los momentos, la hora actual (relacionada con el tiempo que tarda en moverse el agente de un lugar a otro) y una bandera que indica si el agente se encuentra en un corte total y no puede moverse.

El mapa se implementó como un grafo dirigido, con pesos asignados a sus nodos y aristas que representan el costo de atravesarlos. Los nodos del grafo representan las intersecciones de la ciudad y las aristas, los segmentos de calle entre dos intersecciones contiguas. Este grafo es igual para el agente y el ambiente, aunque ambos tienen una copia independiente del mismo.

Los obstáculos se pueden encontrar en intersecciones o en aristas. Estos tienen asociado un nombre o categoría y una visibilidad que puede ser informada, visible o invisible. El estado del agente contiene los obstáculos que el agente conoce al percibir, mientras que el estado del ambiente contiene todos los obstáculos. Los obstáculos están presentes en la simulación durante un período de tiempo, el cual se refiere al campo "hora" del estado del ambiente. Existen dos tipos: obstáculos parciales y obstáculos totales.

La visibilidad de un obstáculo con respecto al agente puede ser: Informada, el obstáculo es percibido por el patrullero tan pronto como aparece sin importar dónde se encuentre; Visible, el obstáculo es percibido por el patrullero sólo si se encuentra en

una intersección o arista adyacente a la posición de este; e Invisible, el obstáculo no es percibido por el patrullero hasta que éste lo atraviesa.

Los obstáculos parciales tienen un retardo multiplicativo, es decir, un valor que multiplica el costo de atravesar la intersección o arista donde se encuentran. Sin embargo, el agente no puede escapar de los obstáculos totales una vez que los atraviesa, lo que significa el fallo de la misión y hace que la bandera "AgenteEnCorteTotal" se levante (sea verdadera) en el estado del ambiente. Por esto, el agente los evita siempre que los percibe, limitando la ocurrencia de una falla a obstáculos totales invisibles.

La posición del agente, tanto en el estado del agente como en el del ambiente, es la intersección sobre la cual el agente se encuentra parado en un momento dado. El agente sólo se puede posicionar en intersecciones, y saltar de una a la otra por medio de las aristas siendo, el atributo UltimaCalleRecorrida, la arista por la cual el agente llegó a su posición actual y, la orientación del agente, la arista por la cual el agente va a salir de la intersección donde se encuentra, si decide avanzar.

El incidente en el estado del patrullero es la intersección a la cual éste debe ir y, como el patrullero, sólo puede ubicarse en una intersección y no en una arista.

La hora del estado del ambiente es el costo (o tiempo) del camino que el patrullero tomó para llegar a su posición actual desde la posición de inicio. Inicialmente es 0 y se va incrementando con los pesos de las aristas e intersecciones que el agente atraviesa multiplicados por los retardos de sus obstáculos. Es lo que se usará como medida de performance de las diferentes estrategias de búsqueda que se prueben.

Percepciones

En cada ciclo de simulación, el patrullero percibe los obstáculos que son detectables por él, según la posición del patrullero y la visibilidad de los obstáculos, y la hora en que los mismos aparecen y desaparecen. Si el campo hora del estado del ambiente se encuentra fuera del período de aparición de un obstáculo, el mismo no se percibirá. Luego el patrullero actualiza su estado, agregando obstáculos nuevos y eliminando obstáculos que han desaparecido.

Acciones

El patrullero puede elegir entre dos acciones: Avanzar y CambiarOrientacion.

Avanzar implica moverse de su posición actual a la intersección al final de la arista apuntada en su orientación, pasando por la misma. Esta acción se puede llevar a cabo si el patrullero no conoce ningún obstáculo total en la arista por la que quiere salir ni en la intersección a la que desea arribar. Luego de moverse, el campo Orientación toma el valor de la primera arista saliente de la nueva intersección (Las aristas salientes de una intersección deben estar ordenadas alfabéticamente según el nombre de la calle de la que forman parte). Realizar esta acción adiciona tiempo a la hora del ambiente (Ver Función de Costo).

CambiarOrientacion implica cambiar el valor de "orientación" a la próxima arista saliente de la intersección actual del patrullero. Esto no cambia la posición del patrullero ni la hora en el ambiente. Para poder elegir esta acción, debe existir alguna

arista saliente que no haya sido seleccionada como el valor de Orientación desde que el patrullero llegó a su posición actual.

Estado inicial del patrullero

Al comienzo de la simulación, el patrullero no tiene ningún obstáculo en su estado, ni una UltimaCalleRecorrida. Su posición es la asignada por el caso de prueba específico, así como la posición del incidente. Su orientación toma el valor de alguna de las aristas de salida de la intersección donde se encuentra al principio.

Prueba de meta

El patrullero ha logrado su objetivo si se encuentra en la misma posición que el incidente.

Fallo del Patrullero

El agente falla en su objetivo en dos circunstancias: si pasa por un obstáculo total invisible o si durante la búsqueda no encuentra ningún camino al incidente.

Función de Costo

Como se mencionó bajo el título de Acciones, CambiarOrientacion tiene un costo de 0. Esto representa que no ha transcurrido ningún tiempo luego de tomar la acción.

Mientras tanto, el costo de la acción Avanzar se calculó como el peso de la arista saliente multiplicado por el producto de los retardos multiplicativos de sus obstáculos más el peso de la intersección destino multiplicado por el producto de los retardos multiplicativos de sus obstáculos. Este cálculo representa el tiempo que le lleva al patrullero ir de una intersección a la siguiente.

Heurística

Como heurística para las estrategias de búsqueda informadas se utilizó la distancia Euclídea¹ entre la posición del patrullero y la posición del incidente. Se eligió como unidad de medida la distancia en píxeles entre el patrullero y el incidente en la representación gráfica del mapa que se construyó como parte de la resolución del trabajo.

Búsqueda

El algoritmo de búsqueda evita bucles cuando desarrolla su árbol de búsqueda mediante la comparación de los estados del patrullero que va calculando. Para esto definimos que dos estados del patrullero son iguales si tienen la misma posición y orientación.

Visualización del comportamiento del patrullero

Para analizar visualmente el camino elegido por el patrullero, se creó una interfaz gráfica donde se puede comparar paso por paso el estado del patrullero y el del ambiente. La figura 2 muestra una captura de la interfaz. Del lado izquierdo, se ve el

¹ La distancia Euclídea entre dos puntos se calcula como el módulo del vector entre ambos puntos.

mapa del ambiente, y todos los obstáculos que están presentes en el mapa en el tiempo en que se tomó la captura. Del lado derecho, se ve el mapa del patrullero, con los obstáculos que él conoce. También en los dos mapas se puede ver la posición del patrullero (imagen de auto azul) y en el mapa del patrullero se ve la posición del incidente (imagen de silueta negra). El mapa completo de la zona del problema se puede ver en el anexo D.



Figura 2. Captura de la Interfaz Gráfica. El mapa de la izquierda corresponde al estado del ambiente y el de la derecha al estado del patrullero. Los colores naranja y rojo representan obstáculos parciales y totales respectivamente en la arista o intersección. Se puede apreciar cómo el patrullero sólo conoce una parte de los obstáculos presentes.

3 Resultados

Estrategias no informadas

- Profundidad

En primer lugar, se realizaron pruebas de la efectividad del patrullero cuando usa la estrategia de búsqueda en profundidad. La mayoría de las pruebas resultaron en el agente desplazándose por el mapa sin dirigirse directamente al objetivo y eventualmente entrando en un bucle interminable.

La aparente falta de dirección en el movimiento del patrullero se debe a la forma en que esta estrategia trabaja, ya que, por el orden arbitrario en que se definieron las acciones del patrullero, éste siempre expande el árbol de búsqueda por la acción de avanzar. Esta falta de dirección hace que se necesiten muchos movimientos para llegar al incidente, haciendo esta estrategia muy ineficiente para la resolución de este problema.

Por otro lado, los bucles interminables son el resultado de no haber implementado ningún mecanismo que los prevenga en primer lugar, por ejemplo, mantener en memoria los estados previos del agente. Esto produce que, por la forma particular en que se implementaron el mapa y las acciones, en dos posiciones contiguas se generen estrategias opuestas.

Estas dos anomalías se continúan presentando aunque se cambie el orden de los operadores. La diferencia es que el patrullero primero intenta avanzar por la última arista de la lista de salientes del nodo en vez de por la primera, lo que sigue produciendo falta de dirección y bucles.

- Costo Uniforme

Con el propósito de probar la efectividad del patrullero utilizando la estrategia de Costo Uniforme, se diseñó un caso de prueba donde el camino más corto desde la posición inicial del patrullero a la posición del incidente contiene un obstáculo que hace que seguir ese camino sea muy costoso.

Se presentan los resultados de la simulación en la parte 1 del anexo C. Como se esperaba, el patrullero decidió tomar un camino que fuera más largo y que rodeara el obstáculo, llegando al incidente en un tiempo menor que si hubiera ido por el camino corto.

Cuando se utiliza Costo Uniforme, cambiar el orden de definición de los operadores del agente no afecta al resultado porque, con esta estrategia, siempre se expande el árbol de búsqueda por el nodo que lleva a un costo total menor sin importar el orden en que se encuentra.

- Amplitud

Se probó la estrategia de amplitud con el mismo caso de prueba que se usó para Costo Uniforme. Los resultados se encuentran en la parte 2 del anexo C. Debido a que Amplitud toma siempre el camino que conlleva la menor cantidad de acciones, el patrullero no esquivó el obstáculo y fue por el camino corto, haciendo que su tiempo final sea mayor que con la estrategia previa.

Si se cambiara el orden de definición de los operadores con esta estrategia, se puede alterar el camino elegido por el patrullero ya que, si dos caminos involucran realizar la misma cantidad de acciones pero uno parte de avanzar y el otro, de CambiarOrientacion; el orden en que estos operadores se definieron determinaría cuál de los dos nodos se expande primero y cuál de los dos caminos se descubre primero. Esto puede verse en la parte 3 del anexo C.

Estrategias informadas

- A*

Para comparar, se sometió al agente con estrategia de búsqueda A* al mismo caso de prueba con el que se probaron Costo Uniforme y Amplitud. Los resultados se pueden ver en la parte 4 del anexo C. En resumen, la estrategia logró encontrar la solución óptima para este caso y esa solución es la misma que encontró Costo Uniforme. Sin embargo, comparando los árboles de búsqueda generados por ambas, es notable que los

árboles creados por A* son de menor tamaño que los creados por Costo Uniforme, como se ve en la figura 3. Esto se debe a que la estrategia informada toma en cuenta su estimación de costo del camino restante cuando expande los nodos del árbol y eso actúa como una guía para expandir primero los nodos de menor costo que llevan más rápidamente al objetivo.

Se compararon los resultados de usar las anteriores estrategias por medio de numerosos casos de prueba y, en todos ellos, A* condujo al patrullero por el mismo camino que lo hizo Costo Uniforme. Aunque en la mayoría de los casos de prueba esas dos estrategias obtuvieron mejores resultados que las demás, hubo casos en que estrategias como Amplitud produjeron mejores tiempos, como puede verse en la parte 5 del anexo C.

Lo que nos dice esto es que, incluso con estrategias informadas, no es posible lograr siempre la solución óptima al problema. El ambiente es dinámico, es decir, los obstáculos aparecen y desaparecen a medida que el agente se mueve por el ambiente, lo cual conlleva a variaciones en los cálculos de costos de los distintos caminos y, por lo tanto, también varía el camino óptimo. Otro factor que influye en esto es el hecho de que existen obstáculos que sólo son visibles una vez que el patrullero se encuentra con o cerca de ellos, lo que puede ocasionar que el camino óptimo calculado por el agente no sea el correcto debido a la falta de información que posee, es decir, los obstáculos que no ve.

Cuando se utiliza A*, cambiar el orden de definición de los operadores del agente no afecta al resultado porque, con esta estrategia, siempre se expande el árbol de búsqueda por el nodo que lleva a un costo total más costo estimado del camino restante menor sin importar el orden en que se encuentra.

 0.xml	13 KB	 0.xml	5 KB
 1.xml	13 KB	 1.xml	5 KB
 2.xml	13 KB	 2.xml	5 KB
 3.xml	14 KB	 3.xml	5 KB
 4.xml	11 KB	 4.xml	5 KB
 5.xml	11 KB	 5.xml	5 KB
 6.xml	9 KB	 6.xml	6 KB
 7.xml	6 KB	 7.xml	3 KB
 8.xml	1 KB	 8.xml	1 KB

Figura 3. Comparación de los tamaños de archivo de los árboles generados por Costo Uniforme (a la izquierda) y A* (a la derecha). A* los produce más pequeños.

- Avara

Por último, se sometió al agente con estrategia avara al mismo caso de prueba que las estrategias anteriores. Los resultados se pueden ver en la parte 6 del anexo C. Se puede observar como esta estrategia busca dirigirse de la forma más directa posible al incidente ignorando los obstáculos que la retrasan, tal como hace amplitud. Sin embargo, a diferencia de amplitud, esta estrategia genera un árbol más guiado y

acotado, además, como CambiarOrientacion deja al agente en el mismo lugar, esta estrategia no dudará en girar si le conviene, es decir, si avanzar lo aleja del objetivo.

Cuando se utiliza la estrategia de búsqueda Avara, cambiar el orden de definición de los operadores del agente no afecta al resultado porque, con esta estrategia, siempre se expande el árbol de búsqueda por el nodo que lleva a un costo estimado del camino restante menor sin importar el orden en que se encuentra.

Para finalizar, corrimos una serie de casos de prueba sobre cada estrategia y obtuvimos los resultados mostrados en la tabla 1.

Inicio	Fin	Costo Uniforme	Amplitud	A*	Avara
1	148	129	192	129	191
15	130	96	97	96	168
130	23	97	97	97	127
15	45	24	24	24	24
78	100	56	56	56	56
12	120	124	124	124	124
45	93	95	87	95	95
Cantidad de veces el mejor		6	5	6	3

Tabla 1. Resultados de aplicar 7 casos de prueba distintos. Costo uniforme y A* destacan por sobre el resto.

4 Conclusiones

Se puede concluir que en la resolución del problema de ir de un lugar de una ciudad a otro sorteando obstáculos e intentando obtener el menor tiempo posible no es posible garantizar optimalidad con ninguna de las estrategias utilizadas ya que la presencia de obstáculos tanto parciales como totales cambia de forma dinámica y la visibilidad del agente no es total. Sin embargo, en la mayoría de los casos las estrategias de costo uniforme y A* fueron las que llegaron al destino más rápido, siendo la primera la que menos árboles generó de las dos. Por todo esto se puede decir que la mejor estrategia, en promedio, fue la de A*.

El modelo propuesto inicialmente funcionó como se esperaba al realizar las simulaciones y fue fácil de adaptar a las clases de FAIA que generó Idem-IA sin necesidad de realizar cambios mayores. Sin embargo, a medida que se fue avanzando se agregaron detalles de implementación que hicieron posible programar ciertas cosas más complejas que no se contemplaron, como la heurística.

El hecho de basar nuestro programa en FAIA nos permitió ahorrar tiempo de implementación y aprender sobre cómo simular estrategias de búsqueda, también nos permitió enfocarnos en la implementación propia del problema y no en la del simulador. Todo esto ayudó a que podamos crear una mejor interfaz gráfica que nos permitió hacer un mejor informe.

Como se previó, la decisión de programar las acciones de la forma que en que se hizo tuvo varias repercusiones en la performance de las simulaciones. Por un lado, llevó a que los árboles fueran más profundos que si se hubiera optado por otras alternativas, como una acción IrA parametrizada. Por otro lado, aumentó la cantidad de ciclos de percepción, búsqueda y acción, cada uno de ellos produciendo un árbol de búsqueda completo. Aun así, el tamaño de los árboles se mantuvo en un rango manejable en la mayoría de los casos. Esta decisión se basó en respetar el modelo de agente basado en objetivos dado por la cátedra de Inteligencia Artificial en clases.

El plantear el mapa de la ciudad como un grafo dirigido basado en listas de adyacencia propició un desarrollo más ágil y fácil de implementar y entender. A pesar de ser más costoso de crear que otras representaciones de grafos, el hecho de necesitar sólo dos copias (una para el ambiente y otra para el agente) y de que esta última se pudo compartir entre los nodos del árbol generado (ya que durante el procesamiento del árbol de búsqueda no cambian los obstáculos) hizo que esta desventaja no sea apreciable en la ejecución del algoritmo.