

INFORME PROYECTO IA 2023

MATERIA: INTELIGENCIA ARTIFICIAL

Comisión: 8

ALUMNOS: SIMON FREDES HADAD - MATIAS ILZ

DECISIONES DE DISEÑO Y TEMAS ABORDADOS DE LA MATERIA

La representación de las creencias se basa en la información percibida por el agente, incorporando elementos como el tiempo, dirección, nodos, y la ubicación de las entidades en los nodos. Su propósito principal es conservar datos adicionales, como las conexiones y los costos de los nodos, que son fundamentales para determinar la ruta más eficiente hacia un objetivo. En nuestra implementación, esta información se almacena en la estructura ListaNodes. También son esenciales las posiciones de los nodos para que el agente pueda determinar su ubicación actual y planificar los movimientos adecuados. Además, es necesario mantener un registro de los tipos y posiciones de las entidades, las cuales se almacenan en la estructura llamada ListaAts, lo que es crítico para actualizar las creencias del agente. Por ejemplo, si el agente toma un tesoro, debe reflejarse en sus creencias.

En nuestra implementación, se utiliza una estrategia de revisión priorizada. Cuando el agente recibe nueva información, esta tiene prioridad sobre la información existente, ya que si existe una creencia que se contradice con la nueva, se va a eliminar la creencia que existía previamente, agregando la nueva sobre la base de conocimiento del agente. El predicado updateDeletedNodes se encarga de eliminar creencias que ya no son válidas. También se utiliza la revisión priorizada al actualizar el estado interno del agente, incorporando nueva información en su conocimiento.

El algoritmo A* se aplica a los planes de desplazamiento, con una representación de estados basada en nodos. Este algoritmo genera vecinos a partir de las conexiones entre los nodos, y mantiene una frontera de búsqueda representada como una lista de nodos pendiente de explorar. En esta lista, se almacenan los nodos junto con su identificador y costo acumulado. El predicado agregar/6 se utiliza para incluir los nodos vecinos generados en la frontera. Para priorizar la exploración, se utiliza el predicado calcularFenFrontera para calcular el valor de $f(n)$ para cada nodo en la frontera, dando prioridad a los nodos con menor valor de función f . La heurística utilizada se basa en la distancia euclidiana entre un nodo y una meta, utilizando las coordenadas (X,Y) de ambos nodos.

El proceso de toma de decisiones del agente empieza con la recepción de información del entorno (Perc). Luego, actualiza sus creencias en función de la percepción actual, asegurándose de mantener la información de tiempo y dirección actualizada. Posteriormente, evalúa su estado interno y planifica una ruta utilizando la estrategia de A*. El agente sigue este plan, moviéndose por el mapa y realizando acciones. En caso de que no tenga un objetivo específico, opta por explorar aleatoriamente el entorno. Durante la exploración aleatoria, el agente puede tomar decisiones al azar, como girar para obtener más información del entorno. Este ciclo de percepción, actualización de creencias, planificación y acción se repite durante todo el tiempo de vida del agente.

RESPUESTAS A PREGUNTAS:

1. Considerando los tipos de agente vistos en la materia (agente reactivo simple, agente reactivo con planes, agente con estado interno, agente deliberativo, agente con metas, agente social), ¿de qué tipo es el agente que ha implementado? Justifique su respuesta.

1. Es un agente con estado interno. El agente además de reaccionar de forma reactiva al entorno, también mantiene y actualiza un estado interno que le permite tomar decisiones informadas y eficientes.

Es importante el estado interno del agente para su toma de decisiones, dado que almacena y gestiona un conocimiento detallado sobre el mundo. Este conocimiento abarca la información sobre nodos, objetos (como tesoros, pociones y relojes), y la dirección en la que se encuentra. Además, el agente realiza una revisión priorizada de su conocimiento, dándole prioridad a la información recién obtenida. Esto le permite adaptarse mejor a las situaciones cambiantes y tomar decisiones basadas en datos actualizados.

Adicionalmente, el agente puede caracterizarse como un agente con metas, ya que utiliza la estrategia de A^* para calcular rutas eficientes con el fin de tomar objetos específicos, siendo relevante para la gestión de metas, dado que el agente se ajusta dinámicamente a los objetivos cambiantes en función de su conocimiento y las percepciones actuales.

2. Mencione dos algoritmos de búsqueda, distintos de A^* , que podrían utilizarse para que el agente encuentre un camino hacia los tesoros. Justifique su elección mencionando al menos un aspecto positivo de los algoritmos elegidos. Luego, para cada uno, explique un problema que podría presentar en el contexto del juego del proyecto, y si el algoritmo A^* permite solucionar los problemas mencionados.

2.DFS: Un aspecto positivo del algoritmo es que es simple de implementar y puede encontrar una solución rápida (orden lineal) si el camino hacia un premio está cerca del agente en profundidad. DFS no usa una evaluación de distancia a una meta y no evalúa las rutas alternativas antes de embarcarse en una profundización extensa, por lo que el algoritmo de A* soluciona este problema mediante el costo de camino y la función heurística, dado que prioriza los nodos que están más cerca del nodo meta, logrando mejorar los resultados de forma notable.

BFS: Lo positivo es que va a encontrar la meta más cercana al agente en términos de cantidad de movimientos, porque explora todos los vecinos del nodo antes de seguir con los vecinos de los vecinos. Lo negativo es que no lo haría en términos de costo, ya que los costos de los movimientos no son uniformes, debido a obstáculos o condiciones variables del entorno. A* soluciona estos problemas porque explora los nodos de menor valor en la función $f(n)$, que es el costo de sumar la estimación del camino menos costoso $h(n)$ y $g(n)$ que determina el costo de llegar del estado inicial a otro estado "n".

3. ¿Cuál de las siguientes funciones considera que es más apropiada como función heurística en el contexto del proyecto? ¿Son admisibles? Justifique.

| | |
|--|---|
| $d_1 = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$ | Distancia euclídea entre $p_1 = (x_1, y_1)$ y $p_2 = (x_2, y_2)$. |
| $d_2 = \max(x_2 - x_1 , y_2 - y_1)$ | La distancia entre $p_1 = (x_1, y_1)$ y $p_2 = (x_2, y_2)$ es la mayor diferencia en sus dimensiones. |
| $d_3 = \min(x_2 - x_1 , y_2 - y_1)$ | La distancia entre $p_1 = (x_1, y_1)$ y $p_2 = (x_2, y_2)$ es la menor diferencia en sus dimensiones. |

3. d_1 no es admisible porque sobreestima el camino, el agente se mueve en cuadrículas y puede avanzar en ocho

direcciones ortogonales, esta función puede sobreestimar la distancia real debido a su naturaleza diagonal. Por lo tanto, no es la más apropiada en este contexto. La más apropiada sería d_2 , por lo que la d_3 va a ser admisible también, lo que si es que va a ser peor y no nos sirve para el contexto del problema.

4. ¿Sería posible, o adecuado, utilizar un método de búsqueda que no almacene la frontera? ¿Qué pasaría si implementamos Hill Climbing en lugar de A^* como método de búsqueda?

4. No sería adecuado, en el juego el terreno tiene obstáculos que tienen que contemplarse, ya que son muy costosos y el agente debe encontrar planes de movimientos que le permitan evitar estos obstáculos. El objetivo del agente es recolectar tesoros en un tiempo limitado, por lo que es necesario tener una planificación de rutas que permita encontrar los mejores caminos posibles. Si se implementara HC en lugar de A^* , no nos serviría ya que nos devolverá la posición de la meta, y a nosotros lo que nos interesa es que nos devuelva el camino. Además, no cumple con la propiedad de completitud, por lo que no asegura encontrar una solución. Más aún, aunque encontrara una solución, no nos asegura que sea la óptima.