
BattleTech
Jugador Inteligente

María Carrasco Rodríguez
Francisco Manuel Herrero Pérez

Ingeniería del Conocimiento

Curso 09/10

Índice general

1. Introducción	2
2. Algoritmo A*	3
2.1. Implementación	4

1

Introducción

BattleTech es un juego de combates entre enormes máquinas de aspecto humanoide llamados *BattleMechs* (o más brevemente llamados Mechs).

2

Algoritmo A*

El algoritmo de búsqueda A* es un tipo de algoritmo de búsqueda en grafos. Se basa en encontrar, siempre y cuando se cumplan ciertas condiciones, el camino de menor coste entre un nodo origen y uno objetivo.

Nuestro objetivo es encontrar el camino más corto entre dos puntos superando obstáculos (ya que en caso de que no hubiera obstáculos, es la línea recta). Esta técnica muy usada en videojuegos de estrategia y, en general en todos los videojuegos donde se trata la inteligencia artificial. Por ello decidimos incorporarla a nuestra práctica.

La mayor ventaja de este algoritmo con respecto a otros es que tiene en cuenta tanto el valor heurístico de los nodos como el coste real del recorrido. Así, el algoritmo A* utiliza una función de evaluación:

$$f(n) = g(n) + h'(n)$$

Donde:

- **$h'(n)$** Valor heurístico del nodo a evaluar desde el actual n hasta el final.
- **$g(n)$** Coste real del camino recorrido para llegar a dicho nodo, n .

A* mantiene dos estructuras de datos auxiliares:

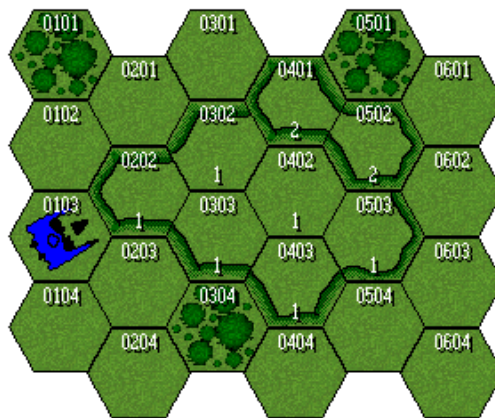
- **Abiertos.** Cola de prioridad, ordenada por el valor $f(n)$ de cada nodo. (Lista de los nodos que necesitan ser comprobados)
- **Cerrados.** Guarda la información de los nodos que ya han sido visitados.

En cada paso del algoritmo se expande el nodo que esté primero en abiertos, y en caso de que no sea un nodo objetivo, calcula la $f(n)$ de todos sus hijos, los inserta en abiertos, y pasa el nodo evaluado a cerrados.

2.1 Implementación

La implementación es genérica y no se basa en ningún juego en concreto. **PathFinder** es una clase genérica que no tiene en cuenta como representes tu grafo ni como representes o como calcules el coste de moverte de un lugar a otro. Deja a gusto del programador especificar la información mediante el paso de funciones al constructor.

¿Cómo sabe *PathFinder* la forma de nuestro grafo? Sólo nos basta con especificarlo en la función **successors**. Esta función especifica los sucesores de un node, que son aquellos nodos a los que se puede llegar desde el nodo inicial en un solo paso.



En nuestro ejemplo, los sucesores del nodo *0103*, donde se encuentra nuestro personaje, son: *0102*, *0202*, *0203*, *0104*

Bibliografía

- [1] González Duque, R., 2003. *PYTHON para todos*. 1sted.
- [2] Gutschmidt, Tom, 2003. *Game Programming with Python, Lua, and Ruby* Premier Press
- [3] Pilgrim, Mark, 2004. *Dive into Python*. Apress.
- [4] <http://theory.stanford.edu/~amitp/GameProgramming/>
- [5] <http://www.policyalmanac.org/games/aStarTutorial.htm>
- [6]
- [7]
- [8]