

# EASY IA

EasyAI es un marco de inteligencia artificial para juegos abstractos de dos jugadores como Tic Tac Toe, Connect 4, Reversi, etc.

Está escrito en Python y hace que sea fácil definir los mecanismos de un juego y jugar contra la computadora o resolver el juego (ver un ejemplo rápido ).

Bajo el capó, la IA es un algoritmo Negamax con tablas de poda y transposición alfa-beta como se describe en Wikipedia . Se ha escrito teniendo en cuenta la claridad / simplicidad, en lugar de la velocidad, por lo que puede ser lento, pero hay correcciones (consulte Cómo hacer que la IA sea más rápida ).

## Minimax

Minimax es un método de decisión para minimizar la pérdida máxima esperada en juegos con adversario y con información perfecta. Minimax es un algoritmo recursivo. John von Neumann es el creador del teorema minimax Un procedimiento recursivo y el corte de la recursión está dado por alguna de las siguientes condiciones: Gana algún jugador Se han explorado N capas, siendo N el límite establecido Se ha agotado el tiempo de exploración Se ha llegado a una situación estática donde no hay grandes cambios de un nivel a otro. En simples palabras el algoritmo de minimax consiste en la elección del mejor movimiento para el computador, suponiendo que el contrincante escogerá uno que lo pueda perjudicar, para escoger la mejor opción este algoritmo realiza un árbol de búsqueda con todos los posibles movimientos, luego recorre todo el árbol de soluciones del juego a partir de un estado dado, es decir, según las casillas que ya han sido rellenas. Por tanto, minimax se ejecutará cada vez que le toque mover a la IA.

Espacio de búsqueda queda definido por:

Estado inicial: es decir estado en el que el juego inicia Operadores: Corresponde a las jugadas legales que se pueden hacer en el juego. Condición Terminal: Determina cuando el juego se acabó, es decir cuando uno de los dos gana. Función de Utilidad: Da un valor numérico a una configuración final de un juego. En un juego en donde se puede ganar, perder o empatar, los valores pueden ser 1, 0, o -1.

## Alpha-Beta pruning

La poda alfa beta es una técnica de búsqueda que reduce el número de nodos evaluados en un árbol de juego por el algoritmo Minimax. Se trata de una técnica muy utilizada en programas de juegos entre adversarios como el ajedrez, el tres en raya o el Go.

Entre los pioneros en el uso de esta técnica encontramos a Arthur Samuel, D.J Edwards y T.P. Hart,<sup>1</sup> Alan Kotok,<sup>2</sup> Alexander Brudno,<sup>3</sup> Donald Knuth y Ronald W. Moore<sup>4</sup>

El problema de la búsqueda Minimax es que el número de estados a explorar es exponencial al número de movimientos. Partiendo de este hecho, la técnica de poda alfa-beta trata de eliminar partes grandes del árbol, aplicándolo a un árbol Minimax estándar, de forma que se devuelva el mismo movimiento que devolvería este, gracias a que la poda de dichas ramas no influye en la decisión final.

La poda alfa-beta toma dicho nombre de la utilización de dos parámetros que describen los límites sobre los valores hacia atrás que aparecen a lo largo de cada camino.

$\alpha$  es el valor de la mejor opción hasta el momento a lo largo del camino para MAX, esto implicará por lo tanto la elección del valor más alto\*  $\beta$  es el valor de la mejor opción hasta el momento a lo largo del camino para MIN, esto implicará por lo tanto la elección del valor más bajo.\*

Esta búsqueda alfa-beta va actualizando el valor de los parámetros según se recorre el árbol. El método realizará la poda de las ramas restantes cuando el valor actual que se está examinando sea peor que el valor actual de  $\alpha$  o  $\beta$  para MAX o MIN, respectivamente.