

Busca Complexa - Jogo da Velha

1. Descrição

O projeto aplica o algoritmo de busca adversária **Minimax** para criar uma Inteligência Artificial que joga **Jogo da Velha (Tic-Tac-Toe)** de maneira ótima. O objetivo é desenvolver um agente que não pode ser derrotado, ou seja, que sempre irá vencer ou forçar um empate, assumindo que o oponente também joga da melhor forma possível.

2. Modelagem

- **Estado:** Uma lista de 9 elementos representando o tabuleiro, onde 'X' é a IA, '0' é o humano e ' ' é um espaço vazio.
 - **Jogador Inicial:** A IA ('X') ou o humano ('0').
 - **Teste de Terminal:** O jogo termina se um jogador vence (três marcas em linha, coluna ou diagonal) ou se o tabuleiro está cheio (empate).
 - **Ações:** Colocar a marca do jogador atual em uma das posições vazias do tabuleiro.
 - **Função de Utilidade:** Atribui um valor a um estado terminal do jogo:
 - +1 : Se a IA (MAX) vence.
 - -1 : Se o Humano (MIN) vence.
 - 0 : Em caso de empate.
-

3. Algoritmo

- **Minimax:** Um algoritmo recursivo que explora a árvore de todos os movimentos possíveis do jogo.
 - O jogador **MAX (IA)** tenta escolher jogadas que levem a um estado com a máxima pontuação de utilidade.
 - O jogador **MIN (Oponente)** tenta escolher jogadas que levem a um estado com a mínima pontuação de utilidade.
 - O algoritmo "propaga" os valores de utilidade dos estados terminais de volta pela árvore, permitindo que o MAX escolha a jogada inicial que garante o melhor resultado possível contra um oponente perfeito.
-

4. Resultados

Diferente dos projetos de busca de caminho, o resultado aqui é o comportamento do agente, que é determinístico e ótimo.

- **Invencibilidade:** A IA que utiliza o Minimax **não pode ser derrotada**.
 - **Jogo Ótimo:** O melhor resultado que um jogador humano pode alcançar contra a IA é um **empate**.
 - **Capitalização de Erros:** Se o jogador humano cometer qualquer erro (uma jogada não ótima), a IA irá capitalizar sobre esse erro para garantir sua vitória.
 - **Desempenho:** Para o Jogo da Velha, o Minimax é extremamente rápido, pois a árvore de jogo é relativamente pequena e pode ser explorada completamente em uma fração de segundo.
-

5. Conclusão

O algoritmo **Minimax** se mostrou perfeitamente eficaz para criar um agente invencível em um jogo determinístico de informação perfeita como o Jogo da Velha. O projeto demonstra com sucesso como um agente pode tomar decisões ótimas em um ambiente competitivo ao simular recursivamente as jogadas futuras e assumir que o oponente também jogará de forma otimizada. A implementação evidencia a base da IA para jogos e a lógica por trás da tomada de decisão estratégica.