Group Members: Vraj Patel, Daishak Patel
March 19, 2024
CAP 4621

CAP 4621 - Final Project Proposal

Project Proposal: AI Adversarial Search Program for Playing Ultimate Tic-Tac-Toe

Problem Statement:

Ultimate Tic-Tac-Toe presents a complex game environment where traditional AI techniques need to be employed to create a challenging opponent for human players. The task involves developing an AI program capable of playing Ultimate Tic-Tac-Toe against a human player with sufficient expertise to entertain a novice player.

Approach:

The primary approach to developing an AI for Ultimate Tic-Tac-Toe will revolve around adversarial search algorithms, particularly on implementing the minimax algorithm. Minimax is a fundamental technique in game theory, enabling the AI to explore possible moves by simulating alternate game states and selecting the move that minimizes the maximum possible loss. However, due to the complexity and branching factor of Ultimate Tic-Tac-Toe, simply using minimax without optimization techniques would result in long computation times hindering game play. Therefore, alpha-beta pruning will be incorporated into the algorithm to mitigate this issue. Alpha-beta pruning helps prune branches of the search tree that are determined to be irrelevant, thereby significantly reducing the number of nodes that need to be evaluated while still guaranteeing the selection of the optimal move.

At the initial stages of the game, the depth of the search tree will be limited to a moderate level. This limitation is based on the assumption that in the early stages of the game, there are fewer "bad" moves to consider, allowing for more efficient exploration of the search space. By limiting the depth initially, the AI can focus on just making a move without being overwhelmed by unnecessary search at the expense of computational resources. An evaluation function will be used at the bottom of the game tree, and these can be based on board control, positional advantage, winning configurations, blocking the opponent, or adaptability to ensure strategic decision-making. As the game progresses and the complexity of the decision-making increases, the depth of the search tree can be dynamically adjusted to adapt to the evolving strategic landscape. This adaptive approach ensures that the AI strikes a balance between computational efficiency and strategic depth throughout the course of the game.

Evaluation Methods:

1. *Win Rate:* We will measure the AI's success by its win rate against human opponents, aiming for 70% or more against various novice players out of a total of 50 games (or as time permits).
2. *Efficiency:* Evaluate the AI's responsiveness by tracking the average time taken to make a move, aiming for quick decisions without sacrificing strategic depth. Average move time less than 10 seconds will be considered a success.
3. *User Interface Easy to Use:* The user interface should be easy to work with for human players. Success criteria – survey of users averages greater than 3 out of a scale of 1-5.
4. *Exploration Depth:* Assess the AI's ability to analyze complex game states by measuring the average depth reached during the search process, aiming for depth exploration of at least 5 moves (large branching factor).
5. *Consistency:* Assess the AI's reliability across multiple game sessions, aiming for reasonable game playing (also measured by a survey of various novice players).