Personality-Based Hex Agent Using Biased Monte Carlo Simulations

Participants:

Your Full Name | Your Student ID | your.email@example.com

1. Problem Description

The goal of this project is to develop an intelligent agent that plays the game of Hex in a way that mimics various human play styles, such as aggressive, defensive, or exploratory. Rather than simply playing optimally, the agent will aim to be engaging and adaptive when playing against human opponents. This requires combining effective move selection with style-driven behavior.

The challenge lies in designing a strategy that balances game performance and personality, without requiring extensive supervised training data or exhaustive search trees.

2. Proposed Solution Approach

We propose treating the problem as a decision-making task under uncertainty, and approaching it through simulation-based evaluation. The core idea is to apply flat Monte Carlo simulations to evaluate the value of each legal move, biased according to the selected play style.

This can be cast as a simplified reinforcement learning setting without explicit value function approximation. Each move is evaluated through playouts that are randomized but strategically biased. The agent chooses the move with the highest empirical win rate. The simulation policy reflects the intended personality type of the agent.

3. System Description

The system consists of four main modules:

1. Game engine: Implements the rules and board logic of Hex.

2. Simulation module: Performs biased playouts from a given board state according to play style.

3. Agent logic: Evaluates legal moves via simulations and selects the most promising one.

4. User interface: Allows a human user to play against the agent and select the play style.

The Hex board and rules will be implemented from scratch. The simulation engine and move selector will be fully custom. Optional external libraries will be used for the UI.

4. Experiment Description

We will evaluate the agent in two main ways:

1. Performance: Win rate against a standard random or greedy agent.

2. Personality consistency: How closely the agent's behavior matches the intended style (measured by move preferences, aggressiveness score, etc.).

Metrics include average playout length, average branching factor explored, and user feedback (if applicable).

5. Technical Details

Programming Language: Python

Environment: Jupyter Notebook / VSCode

Libraries: numpy, matplotlib, possibly tkinter or pygame for GUI

No machine learning frameworks are required for this phase.