Euchre AI

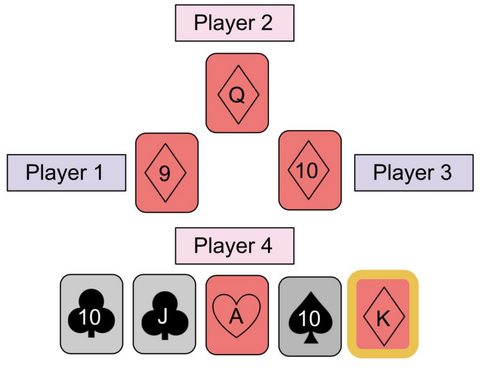
Computer Science Senior Capstone Project
August Nord aanord20@earlham.edu
Earlham College

Intro

Games are often studied in CS as they handle complex problems on a manageable scale. Euchre likewise is a card game with aspects that make it interesting to study:

- Imperfect Information
  - The complete game state is hidden from players
  - As opposed to perfect information games like Chess
- Partner Play
  - It is advantageous to let your partner win sometimes
- The Calling Phase
  - A single decision made first in the round that heavily influences the results of the round

The Game

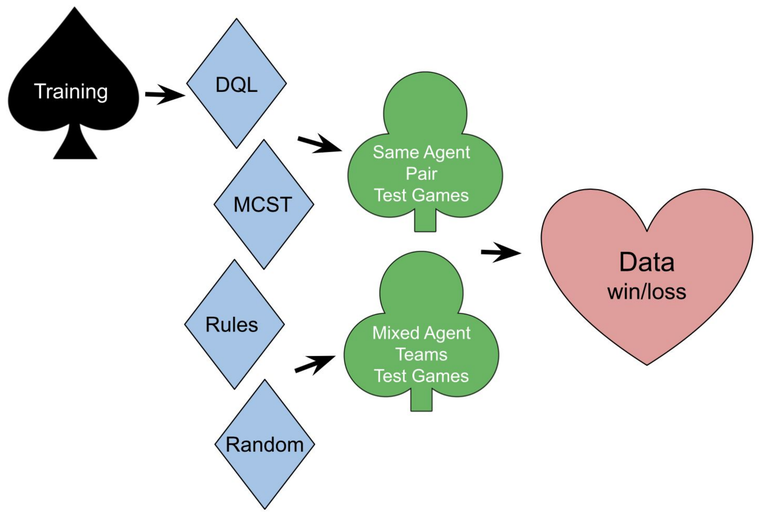


*Player 4 must follow suit and wins the trick with the King

Each game is made of several hands divided into five tricks

- Partners earn points together
  - Players 1 and 3, Players 2 and 4
- Play proceeds clockwise
- Players play the same suit that was led
  - If unable to follow suit, a player can throw a card or 'trump in' with a high-ranking card of that hand's trump suit
- The highest ranking card wins the trick

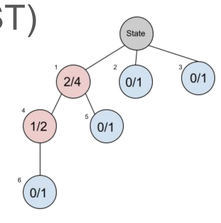
Data Architecture Diagram



Methods

Our work focused on four types of agents:

- Random
  - Selects a playable card at random
- Rules
  - Plays using some basic Euchre strategies
- Deep Q-Learning (DQL)
  - Trains with a neural net
- Monte Carlo Search Tree (MCST)
  - Plays sample games from each decision point and updates probability of winning accordingly



A Monte Carlo Search Tree after 6 simulations. The red nodes are those which, when explored, led to a win. Blue node simulations were losses. Note that the wins and losses of the child nodes affect the parent node's win probability.

Results

In testing, we find that the rules agent performs superior to the Random agent as expected. Equally matched matched pairs perform close to evenly.

Team 1 \ Team 2	Double Rules	Rule/Random	Double Random
Double Rules	34	3522	5314
Rule/Random	-3454	-516	2347
Double Random	-5403	-2412	61

Table showing results of different team pairings playing 10,000 hands. Number reflects the cumulative net score for team 1

Future Work

Ideas for future work include:

- Implement special game cases
  - Add functionality for "going alone"
- Improve the DQL Agent
  - Experiment with more training and fine tuning
- Create an interactive interface
  - Develop an interface that lets the human user play against these computer agents

Acknowledgements

I would like to thank David Barbella, Micah Nord, and Charlie Peck for their support and contributions to this project.