Proposal: Applications of Computational Intelligence to Pineapple Open Face Chinese Poker

Peter Williamson & James Glenn

CPSC 490

1 Introduction

The game of Pineapple Open Face Chinese Poker (Pineapple OFC/Pineapple) is a card game played by one to three players. In general, games are hard for computers to solve when either 1) the number of moves for a computer to make is large, 2) there are hidden information and unknowns, or 3) there are multiple players in a game. While all three apply to Pineapple OFC, it is largely the first two that pose a challenge to its solution (as each game has around $3.8 \cdot 10^{38}$ potential move choices, with each move determined by random card draws).

2 Gameplay

Players compete against one another for “points” on each round of the game (in the solitaire version of the game, players mainly track the number of “royalty points” earned). Pineapple is played on a board of three rows, in which the bottom two rows (Middle/Back) consist of five card hands, and the top row (Front) consists of a three card hand.

Each player’s goal is to create three poker hands as strong as possible (determined by poker hand rankings), with each row having an absolute strength less than or equal to the row below it (i.e. the top row may not be stronger than the middle row which may not be stronger than the back row). Should a player’s board have a row whose strength is greater than the one below it, the player fouls and the hand is forfeited.

Players are initially dealt five cards and must place them all in any arrangement on any of the three rows. Players are then dealt four sets of three cards sequentially, of which two cards from each set are played in any of the three rows (with the third card discarded). Once a card is played on the board, its location may not be changed. Once all players have set 13 cards, scoring is conducted.
2.1 Points and Scoring

Players win points by either winning a row against an opponent, or earning royalty points from high-value hands. A player wins a row by having a hand with a strength greater than that of the opponents corresponding row. E.g. a player whose middle row is a flush will win the row against an opponent whose middle row is a straight. “Royalty points” are earned by placing high-value hands on each of the three rows without fouling. For instance, playing a flush on the back row would earn 4 points, and playing three of a kind in the middle row would earn 2 points.

To score the hand, each player earns the value of the difference of their total points and their opponent’s total points (with solitaire solely concerned with royalty points earned per hand). For instance, if player one wins the back and front rows, and loses the middle row to player two (with no royalties earned by either player), then player one would win one point, and player two would lose one point.

A player might also “scoop” the hand by winning all of the rows against an opponent. In this case, the player will earn an additional three points. Should a player foul and forfeit every row, her opponent will automatically scoop the hand with any hand that does not foul.

2.2 Fantasyland

Should a player score a pair of queens or stronger in the front row and does not foul, the player will enter “Fantasyland” in the next hand. During Fantasyland, a player is dealt fourteen cards at once (while other players must play out the hand as normal). To remain in Fantasyland, the player must achieve four of a kind or stronger in the back row, or three of a kind in the front row.

3 Project Goals

The goal of this work will be to use techniques from computational intelligence to create agents that will play Pineapple OFC at an advanced level. As the computational space for Pineapple is quite large, simple brute force algorithms such as minimax would are too computationally expensive to use (save for the last move or two). In lieu of this, this project will aim to develop a more intelligent agent using other techniques.

Once the game structure for Pineapple is created, the first agents made will use simple heuristics, and more standard techniques, to set a baseline performance that other agents will be compared with. A more intelligent agent will then be created using Q-Learning.
on a reduced state space using a linear function approximator. Should this go well (or go poorly), an agent might be made using a neural net to learn these features independently of human input. Data from previous agents, or from human play, may be used to kick start the neural network with some supervised learning.

Initially, agents will be tasked with solving a solitaire version of the game, with performance measured in “royalty points per hand” (RPH). As royalty points tend to be the largest factor in who will win a match in the two- and three- person variants of the game (as opposed to points won by winning a row), this will be a good metric to track performance of agents.

3.1 Stretch Goals

Should development of the agents for a solitaire game go well, agents will be trained to play in both two- and three- person games. While decisions should not be changed greatly between the two variants, agents must consider more information when playing a additional players. Moreover, this will allow for testing the performance of different agents against one another to calculate win percentages on games to a set number of points.

4 Deliverables

- **02/07/2019**: Submit project description
- **02/08/2019 - 02/28/2019**: Implementation of Pineapple OFC Game Structure
- **Heuristic development for simple agents**
  - Highly naive agent (e.g. greedy agent)
  - Intelligent heuristic based agent (possibly combined with Monte Carlo search trees or some other computational technique)
- **Development of Q-learning agents**
  - Agent with linear function approximator
  - Agent with neural network model (ala AlphaZero)
- **04/24/2019**: Submission of Project