Using Counterfactual Regret Minimisation to build a Nash Equilibrium Strategy for Pot Limit Poker

James Fennell, Dr Frank Glavin
Information Technology, National University of Ireland, Galway
j.fennell2@nuigalway.ie, frank.glavin@nuigalway.ie
(Electronic and Computer Engineering; Computer Science and Information Technology)

Abstract
Poker is a perfect test bed for developing adversarial agents in an extensive form game of incomplete information. The incomplete information component is characterised by each player knowing their, and only their, private cards. Games of this type are useful in that they mimic many real world environments, whether it be in the field of IT security, economic activities such as auctions, or financial markets. This research investigates whether Pot-Limit versions of Poker can be solved using methods that have recently been successfully applied to Limit versions of Poker.

1. Introduction

1.1 Variants of Poker
There are three main variants of poker we will discuss Limit, Pot-Limit and No-Limit. In Limit Poker if a player wants to raise there is a set amount each player's individual raise is restricted to. In No-Limit each player can raise any amount of chips they have. Pot-Limit poker is similar to No-Limit but restricts the maximum bet to be equivalent to the amount of chips already in the pot. We will always discuss two player versions of the game, sometimes referred to as "heads-up".

1.2 Nash Equilibrium Strategies
Nash Equilibrium strategies are defined as optimal strategies. That is, if all agents in a game are following a Nash strategy they have no incentive to deviate from that strategy as it can only negatively affect their expected pay-off. Take the example of Rock Paper Scissors where both players bet €1. Intuitively, the Nash Equilibrium Strategy is to play each of the three options with a probability of 1/3. Any deviation from this may not result in a negative outcome for the player but it does increase their potential exploitation. If Player 2 chooses to go with the strategy of Rock with probability 1 they will still expect to tie overall with Player 1 using the Nash Equilibrium Strategy. However, they run the risk of losing every time if Player 1 chooses to play Paper with probability 1, thus increasing their potential exploitability.

In zero sum two player games (like rock paper scissors, or poker) if both players are pursuing Nash Equilibrium Strategies they will both have an expected pay-off of 0. In games of imperfect information, the Nash Equilibrium cannot be calculated easily and will often have degrees of exploitability. It is up to us to minimise the exploitability of our strategy.

1.2 The Current State of Poker AI Research
Poker has been a very active area of research in the field of Machine Learning for some time [1]. To encourage interest in the field the Annual Computer Poker Competition (ACPC) was set up. Here, agents are developed with novel poker strategies to compete against each other. Bowling et al. [2] describe their submission to the ACPC, Cephus, and claim to have "*essentially solved heads up limit poker*". They use an algorithm they call CFR+, a development on their previous algorithm, Counterfactual Regret Minimisation (CFR). They calculated a Nash Equilibrium strategy for the game with a potential exploitation of 0.000986 Big Blinds. This meant that over a human lifetime of play it would not be possible to *statistically* beat Cephus. So, their algorithm has "essentially" or "weakly" solved Heads Up Limit Hold'Em.

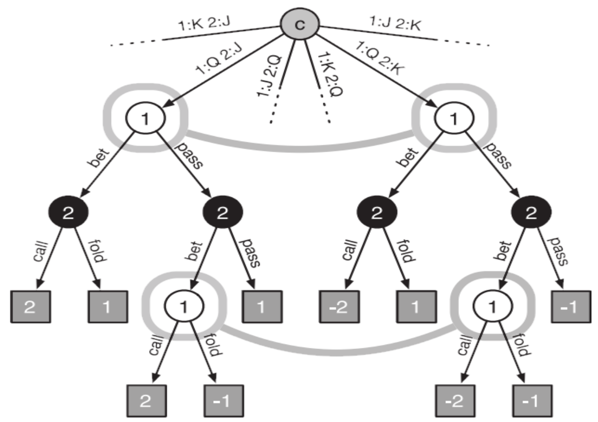
Moravčík et al. [3] had an even greater breakthrough with their No-Limit poker playing agent called *DeepStack*. Here they use CFR+ to solve for the next *x* number of steps. After this they use a Deep Learning Neural Network that has been trained using CFR to provide an "intuition level" that will return the terminal node value from which the CFR+ can calculate the best strategy to pursue. During testing, Deepstack beat 33 professional players with statistical significance. While the effectiveness of this algorithm is very impressive it is not possible to calculate the potential exploitation of the strategy because it does not traverse the entire map at any stage.

1.3 Our Goal
We would like to develop strategies whose performance we can evaluate relative to each other. We would like to use some of the principles from the continual resolving in DeepStack, particularly "forgetting" the strategy deployed up to any given information set. It is our contention that Pot-Limit Poker would be an appropriate stepping stone toward developing No-Limit strategies. By providing an upper bound to the maximum bet that each action node can have, we can build smaller complete game maps which may enable us to capture more granularity than we could in No-Limit games. These games would still have significantly greater complexity than Limit poker. When using abstractions of Pot-Limit Poker we can compare the potential exploitation level of each strategy produced by the algorithms that we will develop.

2. Counterfactual Regret Minimisation

2.1 Counterfactual Regret Minimisation

Counterfactual Regret Minimisation works by using "regret matching" as described by Hart [4]. Here the regret, or opportunity cost of not pursuing the optimum strategy, contributes to the weight of probability that the agent will use this strategy in subsequent visits to the information set. We map the game using information sets, which are identified by the public (the betting sequence and public cards) and private information (private card) available to the player at the time. We do not consider the private information of the opponent.



Information Sets in LeDuc Poker

After each game is completed the regrets are recalculated for each node from the terminal, using the probability assigned to each action by the strategy the node utility is calculated and passed back up through the game map to recalibrate the strategy at each information set. The *average* of all the strategies calculated through the iterations converges to a Nash Equilibrium.

There are two common versions of CFR. Vanilla and Monte Carlo. In Vanilla CFR every possible path through the game is traversed at least once. In Monte Carlo random sampling is undertaken.

2.2 CFR+

CFR+ is a variation of CFR. The most significant change is the use of regret-matching+ where we do not allow negative regrets to accumulate, we convert all negative regrets to 0. If there are no positive regrets, then the strategy is to give every possible action an equal probability. More weight is given to strategies that were calculated on a later iteration rather than averaging all strategies. There is less need for the Monte Carlo method because of the improved speed, so sampling is no longer required.

CFR+ out performs CFR as it converges to a Nash Equilibrium much quicker. The improved performance of CFR+ made it possible for Bowling et al. to compute a complete strategy for Heads-Up Limit Poker using 4800 CPUs for 68 days.

3. Poker Abstractions

We do not have the considerable computing power at our disposal that Tammelin et al. had. As a consequence, we will be investigating CFR+ application to LeDuc Poker, One (High Card), Two (developed by us) and Three Card (Rhode-Island) [5] Pot-Limit versions of poker. This will make it possible to develop a reasonably strong strategy for a stripped back poker game while still incorporating the main facets of the more complex complete Texas Hold'Em game.

Many of the early contestants in the Annual Computer Poker Competition had strategies that were built on abstractions of the game. They typically performed poorly in the competition. However, we are not proposing to develop a strategy for the full game. We will test algorithms that develop strategies for games that retain much of the betting complexity, but have a smaller number of card combinations. We will then analyse the calculated strategies' exploitability.

4. Progress

To date we have developed a test bed where bots, the automated agents with a prescribed strategy, can play the described abstracted poker games. We have also written adaptable CFR code which has developed strategies for the single card versions of poker, both Pot-Limit and No-Limit. The next steps will be to build two card and Rhode-Island Vanilla and Monte Carlo CFR agents. This will be followed by developing CFR+ agents. This research work will also study how long the developed agents take to converge. The relative performance against each other, as well as other basic static strategy bots, will also be evaluated.

5. References

[1] J. Rubin and I. Watson, "Computer poker: A review," *Artificial Intelligence.*, vol. 175, no. 5–6, pp. 958–987, 2011.

[2] O. Tammelin, N. Burch, M. Johanson, and M. Bowling, "Solving Heads-up Limit Texas Hold ' em," 2016.

[3] M. Moravčík, M. Schmid, N. Burch, V. Lisý, D. Morrill, N. Bard, T. Davis, K. Waugh, M. Johanson, M. Bowling, "DeepStack : Expert-Level Artificial Intelligence in Heads-Up No-Limit Poker," pp. 1–37, 2017.

[4] S. Hart and A. Mas-Colell, "A Simple Adaptive Procedure Leading to Correlated Equilibrium," *Econometrica*, vol. 68, no. 5, pp. 1127–1150, 2000.

[5] J. Shi and M. L. Littman, "Abstraction Methods for Game Theoretic Poker," *Computer Games*, no. 333–345, 2001.