

Strategy Report: Team Kerbopots

Rujian Chen, Hongyi Zhang, Shidan Xu

April 14, 2023

1 Overview

Texas hold 'em poker game is a game with simple rules but striking complication. This is due to the game's two characteristics: first, players have incomplete information about the game (including the payoff structure and the strategy of the other players); second, players are able to alter the payoff structure by a series of betting actions. Our team, *kerbopots*, participated in the 2015 MIT Pokerbots Competition, which is a computer poker game using a variant of the Texas hold 'em tournament game called Three-player Pot-limit Sit-and-go Tournament. In the finished Casino and Mini-tournament part, our team got third and fourth place respectively.

In this strategy report, we will briefly describe how we iterated on the design of *kerbopots* during the competition, and discuss about relevant ideas as they appear in the context. In Section 2, we describe the idea of optimal betting when one has perfect information about the other players. In Section 3, we describe how the assumption of optimal betting fails in the incomplete information game, its implication, and a series of remedies we came up. In Section 4, we describe the bold step we took for the final tournament submission, and the rationale behind it. In Section 5, we summarize the lessons we learned and the insights we gained during the competition, which concludes the report.

2 Kelly Bet

Imagine the simplest version of a poker game – you and your opponents know the hole cards of each other, there is no pot limit, and there is only one betting round (the betting is predetermined to happen at one among ‘preflop’, ‘flop’, ‘turn’ or ‘river’), each of you have very large stack sizes, and the goal is to maximize your expected stack size in the long run. Is there an optimal betting strategy in this scenario?

The answer is *yes*. The strategy is called *Kelly Criterion*. The intuition that leads to the formula is simple: in a profitable bet, one wants to bet a large amount to earn more profits,

but one also wants to avoid putting a large portion of his stack size at stake, in case that he could hardly recover in the albeit unlikely event that he loses the bet; the result is a trade-off between the mean and the variance of the outcome. Concretely, in an n -player betting game as described above, if your winning rate is p and your stack size is s , the suggested betting amount is

$$b = \max \left(0, \frac{np - 1}{n - 1} s \right)$$

This betting strategy is ‘optimal’ in the sense that it maximizes the expected return in the long run.

Inspired by this optimality statement, we focused on applying Kelly Criterion to poker playing during the early stage of *kerbopots* development. However, note that in real poker games we do not have perfect information of our winning rate against other players, also the other players do not have perfect information of their winning rates as well. These additional uncertainty makes the naïve Kelly Criterion in disadvantage in various conditions.

3 Zerobot, Onebot and (BE)Twobot

A group of people are asked to each choose an integer in the interval of 0 to 100 to win a prize. The person whose choice is closest to the two-thirds of the mean of the choices of people will be the winner. If you were one of them, what number would you choose?

It turns out that the answer is *it depends*. If people are *all* rational and know the others would also be rational, they would *all* try to choose a number smaller than what the other people would choose *on average*, the result is that all people would choose 0, and 0 would also be your best bet. However, you have to account for the fact that *not all people are rational* and the fact that *not all people assume others would be rational*, so probably 0 would not be that favorable. For example, if one assumes others will only do one level theory of mind reasoning, i.e. others would choose numbers around 33, then his best bet would be $33 \times \frac{2}{3} = 22$. Interestingly, real world experiments find that the winning number is often between 15 and 30, suggesting that people’s recursive reasoning depth is mostly between 0 and 2.

The above discussion leads to our idea of pokerbots of different reasoning depths. Specifically, we define a series of poker players equipped with Kelly betting strategy, but differs in their way of using the criterion in an incomplete information game. Among all design possibilities, we have chosen to implement the following pokerbots:

- **Zerobot.** This is a pokerbot family with zero level theory of mind reasoning, i.e., it completely ignores the information hidden in other players’ actions, and basically bet according to its winning rate against players of random hole cards. In games, this bot tends to lose big stacks when it has a strong hand but another player has a

hand that is even stronger, since it cannot read out this potential risk. In our test, it has a winning rate of about 85% against two randombots (the reference bot provided by the organizer).

- **Onebot.** This is a pokerbot family with one level theory of mind reasoning, i.e., it assumes that the other players are in the Zerobot family, betting approximately according to Kelly Criterion and their winning rates against random hole cards. This means that it can read the strength of the hand of another player by taking into account its betting size. We implement this by designing a heuristic discount function, which adjust its winning rate estimate based on its winning rate against two random hole cards and the betting size of the other players relative to their stack sizes, the pot size and its own stack size. In our test, it has a winning rate of about 85% against two randombots.
- **Twobot.** This is a pokerbot family with two level theory of mind reasoning, i.e., it assumes that the other players are in the Onebot family, betting approximately according to Kelly Criterion and their adjusted winning rates. That is, they will fold a strong hand if they believe another player has a stronger hand, and they will put in more chips when they have a mediocre hand but somehow believe the other players have even weaker hands. Twobot, with the knowledge of this, can exploit this information by bluffing when it has a weak hand but its opponents have some not-so-strong hands, and slow-playing when it has a very strong hand but the other players' hands seem not-so-strong. In our implementation, the bluffing part has not worked very well, but the slow-playing part proved effective in matches against bots in the Onebot family. A specific version, called Bet2bot, which literally just bets 2 when it wants to initialize a larger bet, has about 85% winning rate against our own Onebots.

4 A Bet on the Final Tournament Structure

The design for *kerbopots* in the final tournament follows a rather different rationale from the previous iterations. The new design, which we call ZABot, scales up the aggressiveness of the bot in the pre-flop stage.

As in the previous iterations, ZABot makes the decision to fold, check, call or raise based on the adjusted discounted equity. ZABot raises its opponents (if allowed) when equity is relatively low, calls or checks its opponents when equity is relatively high, and uses the previous pre-flop strategy when equity is moderate. During flop, turn and river, ZABot also performs a semi-bluff if its hand is 1-draw away from straight or flush and the board cards are 1-draw or 2-draw away from straight or flush. Otherwise ZABot behaves as its predecessors. The semi-bluff of ZABot is meant to create the impression that it has a straight or flush and scare its opponents to fold.

The main feature of ZABot is that it will almost never fold to its opponents when its estimated equity is low, and will force its opponents to either fold or call to its raises. This new strategy is designed to force its opponents to fold with relatively low cards. ZABot has an advantage when playing conservative players, but it also fails when facing opponents that resemble ZeroBot, which base does not fold when its opponents continually raise. We will discuss the advantages and disadvantages of this strategy in the Discussion section.

For the final tournament, our bet is that most bots in the competition will be concerned about the implications of repeated raises by an opponent and will often fold under pressure. If this is the case, our new strategy will be effective.

5 Discussion

For the MIT Pokerbot Competition, we developed three main classes of *kerbopots*. ZeroBot, our simplest bot, makes decisions based on the equity of its hands against random other hands, and determines the amount of its bets according to the Kelly Criterion. OneBot, our second iteration bot, estimates its equity against its opponents by discounting its equities against random according to its opponents' actions. ZABot, our final bot, adopts more aggressive pre-flop strategies to force its opponents to fold during pre-flop.

Interestingly, in general, OneBot beats ZeroBot, ZABot beats OneBot, and ZeroBot beats ZABot. We now discuss this phenomenon. OneBot is able to beat ZeroBot because it predicts whether ZeroBot has a good hand and folds when it believes its hands are worse, thereby reducing the amount of money lost when dealt an inferior hand. Conversely, when it has a better hand than ZeroBot, it will put more money into the pot. Thus OneBot often beats Zerobot. On the other hand, when OneBot plays ZABot, ZABot raises its opponents when it believes its chances of winning are low, and calls or checks its opponents when its chances of winning are high. This strategy is more risky and only succeeds when it forces the opponents to fold. However, the aggressiveness often works against OneBot, which believes ZABot has a good hand while in reality it doesn't. This new strategy allows ZABot to use low equity hands to fold OneBot's hands, and keep relatively high equity hands alive. In this way, ZABot would enter post-flop stages with a higher percentage of winning hands than OneBot. In general, ZABot can be expected to be effective when playing against conservative players. On the contrary, if ZABot plays a bot which considers only the equity of his own hands and does not speculate its opponents' hands, such as ZeroBot, ZABot will not be able to fold its opponents' good hands, and will lose much more often due to going to showdown with inferior hands and a large pot.

So, the three generations of *kerbopots* make up an interesting circle of relative dominance. This circle demonstrates how the outcome of one given strategy will vary widely with the opponents' strategies. And to become a truly invincible bot, it is necessary to develop an adaptive strategy to handle different opponents.