

EEL6841 Spring 2002: AI Article

The Challenge of Poker

(Artificial Intelligence #134 2002 pages 201-240)

by

Darse Billings, Aaron Davidson, Jonathan Shaeffer, Duane
Szafron

Presented by

John A. Martiney

TABLE OF CONTENTS

TABLE OF CONTENTS.....	1
I. INTRODUCTION	2
II. TEXAS HOLD'EM	2
III. HOW POKI PLAYS POKER.....	3
1. PRE-FLOP BETTING.....	3
2. POST-FLOP BETTING.....	4
3. IMPROVED BETTING STRATEGY.....	5
4. OPPONENT MODELING	5
a. Statistics-based model.....	5
b. Neural Network-based model	6
IV. PERFORMANCE.....	6
V. CONCLUSION.....	6

I. INTRODUCTION

The artificial intelligence community has seen the success in programs that can defeat the best human opponents in games such as chess, checkers, backgammon, and Othello. All of those games have the common feature that the entire game state is known by every player. Therefore, the program is able to calculate the optimal action for the current state.

In the game of poker, the entire game state is not known by every player. The program must therefore be able to handle a system with imperfect information. The betting behavior of each player is also affected by the number of players, previous betting habits, time of the day, current life events, etc. Therefore, forming appropriate statistical models of the “hidden information” (the other player’s cards) may not be sufficient information to decide on the optimal course of action.

This paper describes the components of the Poki poker playing program. The components include statistical analysis, expert systems, and neural networks.

II. TEXAS HOLD'EM

The Poki program specifically plays the Texas Hold'em version of poker. This is the version of poker used to determine the annual World Series of Poker champion.

The following is a brief description of the rules of play in Texas Hold'em.

1. There are 4 rounds of betting called the *pre-flop*, *flop*, *turn*, and *river*. Bets in the first two rounds are usually \$10. Bets on the second two rounds are usually \$20.
2. Each player receives two private cards face down during the *pre-flop*, 3 community cards face up on the table during the *flop*, and one extra community card face up on the table during the *turn* and *river*. The community cards are shared amongst all of the players.

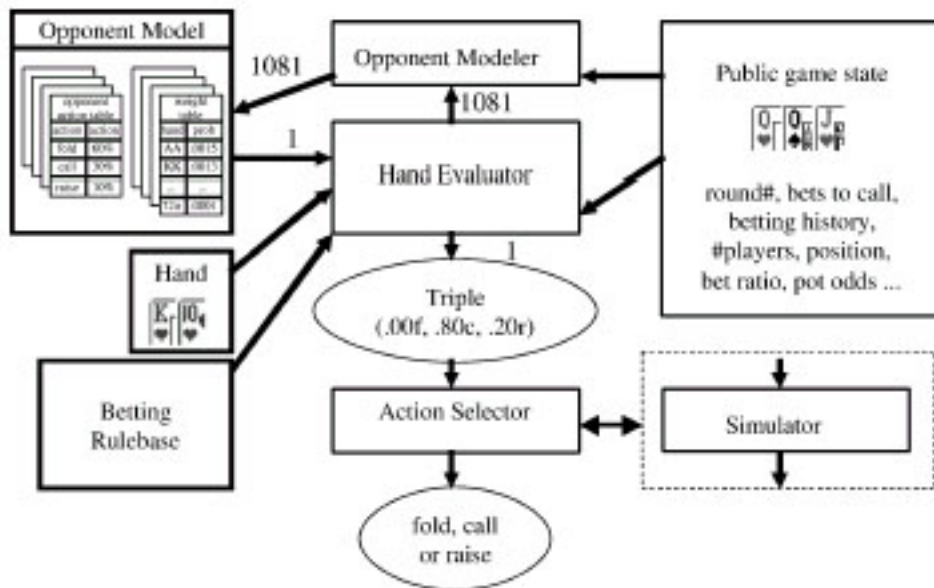
3. Each player can perform one of three actions during their turn in a given round:
fold, check/call, or bet/raise.
4. The player with the strongest hand wins the pot.

III. HOW POKI PLAYS POKER

Poki is written in Java. It is available in three forms: IRC-Dealer, Tournament-Dealer, and TCP/IP-Dealer. The TCP/IP version is available as a Java applet at <http://www.cs.ualberta.ca/~games/poker/>.

The architecture of the Poki program is shown below. The functional components are described in detail when the betting strategy is presented for each round of game-play.

Poki Program Architecture



The architectural components shown above allows Poki to identify the current betting round, number of players left in the game, betting history, community cards currently on the table, current hand strength, etc. It also allows Poki to model the opponents currently playing. It should be noted that each of the squares on the sides shown above (excluding the betting rulebase) can be considered a separate “expert system” dedicated to that specific task. The information from each expert is combined to produce a final bet choice.

1. PRE-FLOP BETTING

The pre-flop betting strategy can be easily modeled using a simple formula-based expert system. There are a total of $52 \text{ choose } 2 = 1326$ possible initial hands. Poki computes the income rate of any given pair off-line. The computation is done by playing all 1326 possible hands to the end; assuming everyone always checks. The amount of micro bets won (1000 micro bets = 1 small bet) by the given hand is then

considered to be the income rate of the given hand. This is a first order approximation of the actual value of a given pair.

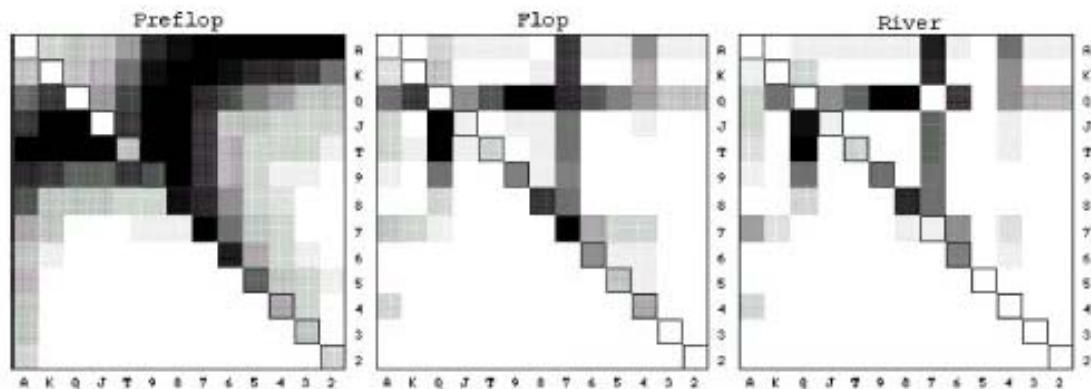
David Sklansky is a poker expert who ranked certain pre-flop pairs in one of his books on Texas Hold'em poker. The pre-flop income rates are validated by comparing them to Sklansky's 8 groupings. A lower group number meant higher hand strength. The income rates calculated by Poki can be used to rank the pre-flop hands in the order shown by Sklansky's 8 groups. Therefore, it is determined that income rate rankings can be considered "expert strategies."

2. POST-FLOP BETTING

After the flop, simple ranking strategies are no longer appropriate. The machine must take into account the current game state and player actions. It must also take action in real-time; therefore, it cannot spend too much time searching a relatively large game tree.

Poki begins forming post-flop betting decisions by first calculating the effective hand strength (EHS). The EHS is based on both the hand strength (HS) and the positive potential (PPot). Hand strength is the probability that the current hand is better than a random hand. The positive potential is the probability that the current hand will benefit from future community cards. The EHS is then used in combination with the output from other game state units to generate a triple. The triple is of the form Fold/Call/Raise and is mapped to values from 0 to 1. It is the probability of performing a fold, call, or raise action. The actual action is chosen using a random number generator between 0 and 1. For example, if the triple is 0.1/0.4/0.5 and the random number generator has an output of 0.3; then, the call action is taken because it would have been mapped to from 0.1 to 0.5 (probability is 0.4).

The calculation of the PPot can be greatly reduced by adding a weight table to the system for each opponent. Initial PPot calculations assumed that all card pairs were possible at every point in the game. It is unlikely that weak card pairs will be present during the turn and river betting stages. A sample set of weight tables is shown below.



All weight probabilities are initialized to 1 in the beginning. A darker color indicates a higher probability of the pair being present. Same suit pairs are shown in the upper right corner. The weight tables help reduce the number of possibilities that Poki must explore before making a decision. It is equivalent to trimming branches in a game tree. The weights are not guaranteed to be correct all of the time. This method sacrifices certain decisions in order to obtain faster response time. The tables are updated based on the betting actions of each opponent. They can be considered to be part of the opponent model.

3. IMPROVED BETTING STRATEGY

Poki attempts to make better betting decisions by using selective sampling. Selective sampling indicates that the assignment of probable hands to each opponent is consistent with some determined distribution. It is used to essentially simulate betting decisions (called full information simulation). The results are then used to make a more educated betting decision. The simulations are performed using the already mentioned methods for post-flop betting. However, the opponents hands are further limited based on known betting habits and game conditions. The simulations result in call/raise/fold expected values (EV). These expected values are then used in the calculation of the betting triple.

The actual implementation of the selective sampling strategy is not covered in very much detail. The authors mention that it is implemented in Poki; but, they give no specific information on the subject. The specific purpose of selective sampling is also unclear. An example of how selective sampling works would make the topic easier to understand.

4. OPPONENT MODELING

The improved betting strategy is highly dependent on the opponent models to be able to execute realistic simulations. Opponents can be modeled using either statistics or neural networks. Expert poker players start the creation of their opponent models before the game even starts. Although neither method is able to successfully capture all of the aspects of an expert poker player, they provide a good base from where to start.

a. Statistics-based model

In this case, opponents are modeled using either generic opponent modeling (GOM) or specific opponent modeling (SOM) in real time. A better opponent modeling scheme would be able to switch between GOM and SOM based on the current game being played.

The generic opponent model assumes that the opponent will make rational betting decisions. Such a model will have some predetermined set of rules to follow. In some cases, the machine may use its own rules to model the opponent. The decisions would be based on statistical information (betting frequencies) for any given betting turn in the game.

The specific opponent model uses the opponents betting history to predict future actions. It essentially assumes that the opponent behaves in a consistent manner. An example would be an opponent which quickly folds each time opponents raise him more than 3 times in 1 turn. The same opponent is expected to fold each time those conditions are met.

b. Neural Network-based model

A given neural network can be trained with real player data. The inputs to the network are different aspects of the game such as: number of players, last betting decision, seating order, losing/winning money, etc. The outputs of the network are the 3 betting decisions of fold/call/raise. With enough training information, the neural network can be used to draw correlations between states of the game and betting decisions. This is particularly useful to explore new conditions that were not previously considered. However, the training of a neural network cannot be performed in real-time (yet). Information from neural networks can be used in the statistics-based opponent model.

IV. PERFORMANCE

The authors show two methods of determining the performance of Poki. Both methods showed positive results.

The first is to have newer versions of Poki play against older versions. It is clear that newer versions of the program should be better because they can perform better selective sampling, faster simulations, etc. Each game could be played in different seating arrangements to reduce “noise.” However, such a method would only test a given strategy. It does not have a wide range of variation between players because all of the players are essentially Poki.

The second benchmark is to have Poki play against real opponents online. Performance is measured in the number of small bets (\$10) Poki wins per hand. For example, one measurement shows that Poki could win 0.13 bets/hand after playing 25,000+ poker hands. This means that Poki was winning approximately \$1.3 per hand over 25,000 hands.

The next step in the development of the program is to have Poki’s betting decisions reviewed by expert poker players.

V. CONCLUSION

This paper presents a good framework for the possible development of a world-class poker playing system. It uses machine intelligence “building blocks” such as expert systems, statistical behavior models, and neural networks. Randomness during game-play is introduced using an action triple of fold/call/raise.

Several poker behavior strategies are presented. However, a technical description of them is lacking. It is evident that Poki does not implement all of the strategies mentioned; but, the ones that it does implement are not properly explained. The generation of the action triple is still left as a black box input/output device. The block diagram shows that information is passed from the various expert systems. The paper then eludes to say that the triple is calculated using a formula-based approach. However, the formulas are never mentioned. It would have been helpful to show an example.

One of the key features of Poki is that it uses the selective sampling strategy. Surprisingly, selective sampling is also poorly explained. The 40 page paper dedicates a 2 ½ page subsection to the topic. However, 1 ½ pages of the subsection are spent explaining why Poki needs a better betting strategy. This is consistent with several other sections in the paper where reasons for improvement are given; but, the actual “fixes” are never mentioned.

Finally, there is much to be said about the user interface available at the website for the program. The creators of Poki have made it very simple for their program to interact with real human players. Experimental results can be gathered with very little effort. Such a system has truly allowed the Poki system to evolve.