An Experimental Approach to Online Opponent Modeling in Texas Hold'em Poker

Dinis Felix[1] and Luís Paulo Reis[1,2]

[1] FEUP - Faculty of Engineering of the University of Porto
felixdinis@gmail.com
[2] LIACC – Artificial Intelligence and Computer Science Lab., University of Porto, Portugal
lpreis@fe.up.pt

Abstract. The game of Poker is an excellent test bed for studying opponent modeling methodologies applied to non-deterministic games with incomplete information. The most known Poker variant, Texas Hold'em Poker, combines simple rules with a huge amount of possible playing strategies. This paper is focused on developing algorithms for performing simple online opponent modeling in Texas Hold'em. The opponent modeling approach developed enables to select the best strategy to play against each given opponent. Several autonomous agents were developed in order to simulate typical Poker player's behavior and one other agent, was developed capable of using simple opponent modeling techniques in order to select the best playing strategy against each of the other opponents. Results achieved in realistic experiments using eight distinct poker playing agents showed the usefulness of the approach. The observer agent developed is clearly capable of outperforming all its counterparts in all the experiments performed.

Keywords: Opponent Modeling, Texas Hold’em, Poker, Autonomous Agents.

1 Introduction

Incomplete knowledge, risk management, opponent modeling and dealing with unreliable information are topics that identify Poker as an important research area in Artificial Intelligence (AI). Unlike in games of perfect information, in the game of Poker, players face hidden information resulting from the opponents’ cards and future actions. In such a domain, to be successful, players face the need to use opponent modeling techniques in order to understand and adapt themselves to the opponents playing style [1] [2].

In a multi-player game with imperfect knowledge, where multiple competing agents must deal with risk management, unreliable information and deception, agent modeling is an essential element in successful agent play. In this kind of environment, agents act under uncertainty, and a crucial issue is to have a good opponent modeling (OM) system, learning and problem solving capabilities.

Opponent modeling allows determining a likely probability distribution for the opponent’s hidden cards. However, the huge amount of possible playing strategies in Poker makes opponent modeling a very hard task in this domain.

G. Zaverucha and A. Loureiro da Costa (Eds.): SBIA 2008, LNAI 5249, pp. 83–92, 2008.

The main goal of this work was to prove that a poker agent that considers the opponent behaviour has better results, against players that use typical poker playing strategies, than an agent that doesn't, even when playing the same global betting strategy.

The rest of the paper is organized as follows. Section 2 describes games with incomplete information, Texas Hold'em Poker and some related work. Section 3 describes the opponent modeling strategies developed. Section 4 describes the Poker playing autonomous agents developed and section 5 the results achieved in controlled experiments. Section 6 contains the conclusions of the paper and pointers to future work.

2 Games with Incomplete Information

Games have proven to be both interesting and rewarding for research in Artificial Intelligence (AI). Many success stories like Chinook (checkers) [3], Logistello (Othello) [4], Deep Blue [5] and Hydra [6] (chess), TD Gammon (backgammon) [7], and Maven (Scrabble) [8] have demonstrated that computer programs can surpass all human players in skill. Games such as Poker are difficult because of the elements of imperfect information and partial observability [9].

Games with incomplete information are games where the player does not have complete knowledge of the entire game state. In Poker, the players only have access to the information of their own cards. Predicting opponent cards, probabilities for possible future card combinations and future opponent moves is a challenge in the Artificial Intelligence domain. Poker is also a stochastic game because the shuffling of the deck introduces the chance element into the game state.

Von Neumann introduced game theory [10] in 1940s and has since become one of the foundations of modern economics [11]. He used the game of poker as a basic model for 2-player zero-sum adversarial games, and proved the first fundamental result, the famous Minimax Theorem. However, all reasoning in poker must be probabilistic, as things are rarely ever certain. Also, the cumulative sum of a series of games matter more than any individual game [12][13]. Poker is also a non-cooperative multi-player game. Although multi-player games are inherently unstable, due in part to the possibility of coalitions (i.e., teams), those complexities are minimized in a non-cooperative game such as Poker [14].

Poker is a popular type of card game in which players bet on the value of the card combination ("hand") in their possession, by placing a bet into a central pot. The winner is the one who holds the hand with the highest value according to an established hand rankings hierarchy, or otherwise the player who remains "in the hand" after all others have folded. The game has many variations, all following a similar pattern of play. Depending on the variant, hands may be formed using cards, which are concealed from others, or from a combination of concealed cards and community cards. The hand ranking hierarchy starts whith Royal Flush, the highest of all poker hands (10, J, Q, K, A of the same suit), then Straight Flush (five cards in consecutive numerical order, all of the same suit), Four of a Kind (four cards of the same value and any other card), Full House (three cards of the same value and another two cards that form a pair), Flush (five non-consecutive cards of the same suit), Straight (five consecutive cards, but not of the same suit.), Three of a Kind (three cards of the same value, and two supporting cards that are not a pair.), Two Pair (two sets of pairs, and another random card.), One Pair (two cards of the same value and three random supporting cards.)

Texas Hold'em is the most popular Poker game in the world, and is thus the variant of Poker considered for this project. Hold'em is a community card game where each player may use any combination of the five community cards and the player's own two hole cards to make a poker hand, in contrast to Poker variants like Stud or Draw where each player holds a separate individual hand.

This project is based on previous work from the University of Alberta [15] and Billings [16, 17, 18] and a previously developed poker simulator multi-agent system [19].

There are 1326 possible hands prior to the flop. The value of one of these hands is called an income rate and is based on an off-line computation that consists of playing several million games where all players call the first bet [16, 17]. The basic betting strategy after the flop is based on computing the hand strength (HS), positive potential (PPot), negative potential (NPot), and effective hand strength (EHS) of agent's hand relative to the board. EHS is a measure of how well the agent's hand stands in relationship to the remaining active opponents in the game. The hand strength (HS) is the probability that a given hand is better than that of an active opponent. Suppose an opponent is equally likely to have any possible two hole card combination. Thus it is possible to calculate the hand strength as:

```
HandStrength(ourcards, boardcards) {
  ahead = tied = behind = 0
  ourrank = Rank(ourcards, boardcards)
  /*Consider all two-card combinations of remaining cards*/
  for each case(oppcards)
  {
       opprank = Rank(oppcards, boardcards)
       if (ourrank>opprank) ahead += 1
         else if (ourrank==opprank) tied += 1
              else behind += 1
  }
  handstrength = (ahead+tied/2) / (ahead+tied+behind)
  return(handstrength)
}
```

After the flop, there are still two more board cards to be revealed. On the turn, there is one and it's essential to determine the potential impact of these cards. The positive potential (PPot) is the chance that a hand that is not currently the best improves to win at the showdown. The negative potential (NPot) is the chance that a currently leading hand ends up losing.

PPot and NPot are calculated by enumerating over all possible hole cards for the opponent, like the hand strength calculation, and also over all possible board cards.

```
HandPotential(ourcards,boardcards,player_classification){
  int array HP[3][3], HPTotal[3]      /* initialize to 0 */
  ourrank = Rank(ourcards, boardcards)
  /*Consider all two-card combinations of remaining cards*/
  for each case(oppcards) {
       opprank = Rank(oppcards,boardcards)
       if(ourrank>opprank) index = ahead
```

```
        else if(ourrank=opprank) index = tied
             else index = behind
      HPTotal[index] += 1
      /* All possible board cards to come. */
      for each case(turn)       {
              for each case(river) {
                 board = [boardcards,turn,river]
                 ourbest = Rank(ourcards,board)
                 oppbest = Rank(oppcards,board)
                 if(ourbest>oppbest) HP[index][ahead]+=1
                   else if(ourbest==oppbest)HP[index][tied]+=1
                        else HP[index][behind]+=1
              }
      }
  }
  /* PPot: were behind but moved ahead. */
  PPot = (HP[behind][ahead] + HP[behind][tied]/2
        + HP[tied][ahead]/2) / (HPTotal[behind]+HPTotal[tied]/2)
  /* NPot: were ahead but fell behind. */
  NPot = (HP[ahead][behind] + HP[tied][behind]/2
        + HP[ahead][tied]/2) / (HPTotal[ahead]+HPTotal[tied]/2)
   return(PPot,NPot)
}
```

The effective hand strength (EHS) combines hand strength and potential to give a single measure of the relative strength hand against an active opponent. One simple formula for computing the probability of winning at the showdown is:

$$\Pr(\text{win}) = HS \times (1 - NPot) + (1 - HS) \times PPot \quad (1)$$

Since the interest is the probability of the hand is either currently the best, or will improve to become the best, one possible formula for EHS sets NPot = 0, giving:

$$EHS = HS + (1 - HS) \times PPot \quad (2)$$

These betting strategies, divided in betting strategy before and after the flop [16], were developed at University of Alberta [18] and are enough to develop a basic agent capable of playing poker.

3 Opponent Modelling

No poker strategy is complete without a good opponent modeling system [20]. A strong Poker player must develop a dynamically changing (adaptive) model of each opponent, to identify potential weaknesses. In Poker, two opponents can make opposite kinds of errors and both can be exploited, but it requires a different response for each [16]. The Intelligent Agent developed in this project observes the moves of the other players at the table. There are many possible approaches to opponent modeling [2,13,21], but in this work the observation model is based on basic observations of the starting moves of the players, so it could be created a fast, online estimated guess of their starting hands in future rounds.

3.1 Loose/Tight and Passive/Aggressive

Players could be classified generally in four models that depend of two variables: loose/tight and passive/aggressive. Knowing the types of hole cards various players tend to play, and in what position, is probably the start point of opponent modeling. Players are classified as loose or tight according to the percentage of hands they play. These two concepts are obtained analyzing the percentage of the time a player puts money into a pot to see a flop in Hold'em - VP$IP. The players are also classified as passive or aggressive. These concepts are obtained analyzing the Aggression Factor (AF) which describes the nature of a player. Figure 1 shows the target playing area for the agents developed as a factor of the number of starting hands played and the bet/raise size and frequency.

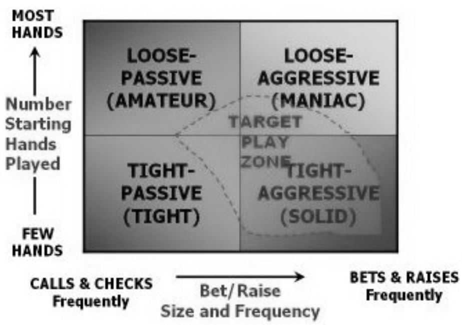

Fig. 1. Player Classification based on the number of starting hands played (VP$IP) and bet/raise size and frequency (AF)

3.2 Sklansky Groups

One of the most difficult and yet crucial decisions when playing Texas Hold'em is whether to even play or not the starting hand. David Sklansky and Mason Malmuth, co-authors of "Hold'em Poker and Advanced Hold'em Poker", were the first to apply rankings to the starting 2-card hands, and place them in groupings with advice on how to play those groups [22,23].

There are some computer simulations developed to test Sklansky's hand rankings that suggests some alterations. But in general, the classification is very similar. Considering a player loose/tight behavior and the Sklansky groups, it is easy to conclude what starting hands a tight player usually plays. If the VP$IP of the player is bellow 28%, he is probably playing most of the hands from higher groups and rarely from the other groups. On the other hand, if a player is a loose player, he's probably playing more hands from lower groups than a tight player. With this simple analysis, it is easy to exclude some of the hands that our opponents probably don't play. The percentage defined here, 28%, is an estimated approach that classifies players in loose or thight style. When using different player classification, with sub-levels for loose and tight (i.e. slightly loose or very loose), this percentage should be adapted.

4 Poker Playing Autonomous Agents

Based on the player classification developed, 8 intelligent agents were created, two for each player style:

- Two Loose Aggressive Agents (Maniac and Gambler);
- Two Loose Passive Agents (Fish and Calling Station);
- Two Tight Aggressive Agents (Fox and Ace);
- Two Tight Passive Agents (Rock and Weak Tight).

A general observer agent was also created capable of keeping the information of every move made from the opponents and calculating playing information like the VP$IP and AF of each opponent in every moment of the game. The opponents are classified into 4 types of players. So, an opponent with VP$IP above 28% is considered loose, otherwise, the player is considered tight. With an AF above 1, the player is considered aggressive and less than 1 is considered passive (table 1).

When the observer's turn comes, he knows which of the opponents are in the game and predicts, based on the available information, what kind of player they are.

Table 1. Player Classification considered by the observer agent

	AF<=1	AF>1
VP$IP>=28%	Loose Passive (classification1)	Loose Aggressive (classification2)
VP%IP<28%	Tight Passive (classification3)	Tight Aggressive (classification4)

After player classification the agent consider a different range of possible hands for different opponents. These considerations are based in the study of each kind of poker player. A general consideration is that tight players have a small range of possible hands than loose agents.

In order to pass this information to Hand Strength calculation, for each player is determined a parameter that was called "sklansky". This parameter is a float number that represents the lowest value of a hand that belongs to the most probable range of hands that the player plays with that specific movement (call or raise).

With conscience that many times the correct hand of the opponent is wrongly ignored, the better approach of Effective Hand Strength calculation given with this technique should give a better result that compensates this.

```
Sklansky(player_classification, player_move) {
  random = (rand() % 10) + 1;
  switch(player_classification) {
    case(1): /*loose passive*/
       if(player_move==raise) {
           if(random<=3){return 26.2;} /*last hand from group3*/
                   else {return 44.2;} /*last hand from group1*/
       }
```

```
      else if(player_move==call)
      {return -100.0;} /* all the possible hands */
    case(2):
      ...
```

The calculation of the "sklansky" parameter considers two variables for each opponent analyzed. The first is the "player classification", each one of the opponents are classified as one of the four kind of players described in table 1. The second parameter is "player move", that is the action made by the opponent in the pre-flop round. Based on those variables it's possible to exclude some hands that the opponent probably doesn't play. The random function is used in order to get a more flexible and correct result of the hands to exclude. For example, a loose passive player usually raises in the pre-flop hands from the group 1 of the Sklansky groups, meanwhile a small percentage of the times, these players also raise hands from other groups.

The Hand Strength and Potential Hand Strength could now be calculated with a better approach. They are calculated only for active players and only considering the hands with a rank better than the "sklansky" parameter. The Hand Strength Formula presented in chapter 2 is reformulated as follows:

```
HandStrength(ourcards, boardcards) {
  ahead = tied = behind = 0
  ourrank = Rank(ourcards, boardcards)
  /*Consider all two-card combinations of remaining cards*/
  for each case(oppcards) {
       if(oppcards belong to player_starting_hands_range) {
              opprank = Rank(oppcards, boardcards)
              if (ourrank>opprank) ahead += 1
                else if (ourrank==opprank) tied += 1
                     else behind += 1
       }
  }
  handstrength = (ahead+tied/2) / (ahead+tied+behind)
  return(handstrength)
}
```

A similar reformulation is performed for Hand Potential Strength. The Effective Hand Strength for each one of the opponents is given by the equation 3.

$$EHSi = HSi + (1 - HSi) \times PPoti \tag{3}$$

The observer developed with the strategy presented is an agent capable of observe the opponents and take decisions based on this observation. This new strategy only considers the possible cards of the opponent to calculate the Effective Hand Strength.

5 Results

The methodology used to test the approach was based on performing game simulations with poker agents playing different strategies. This was similar to a simulation of a real game with the objective to analyze the differences between the performance of an observer agent and a non observer agent. Both agents were set to play at the same table using the same strategy of pre-flop hand selection.

In order to obtain some results, several simulations were made with the agents created. There are 8 normal agents and 1 observer, so the simulations were performed with 9 players at each table.

The intention is to give the Observer Agent the possibility to play in a table with different kind of players. The Observer had the chance to test his new strategy against different players several times along a complete simulation. The observer was programmed to act like a Loose Aggressive in the first round of simulations, Loose Passive in the second, Tight Aggressive in the fourth and Tight Passive in the final round of simulations.

Figures 2, 3, 4 and 5 show the results achieved. Each of the figures compares the evolution of the observer and non-observers agent's bankroll during the simulations using a distinct behavior: Loose Aggressive (Gambler), Loose Passive (Calling Station), Tight Aggressive (Fox) and Tight Passive (Rock).

In the 12 tests done (more than 10 000 hands played) with observer agents, the Observer has better results than the non observer agent that uses the same hand selection in pre-flop. Even with no significant advantage in some of the simulations, the global result of Opponent Modeling reveals to be positive.

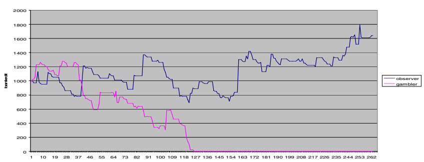

Fig. 2. Bankroll of the Loose Aggressive (Gambler) Observer and Non-Observer Agents

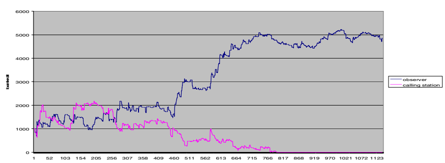

Fig. 3. Bankroll of the Loose Passive (Calling Station) Observer and Non-Observer Agents

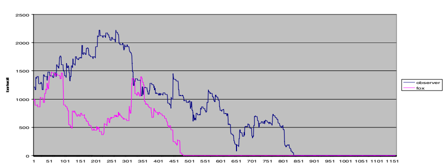

Fig. 4. Bankroll of the Tight Aggressive (Fox) Observer and Non-Observer Agents

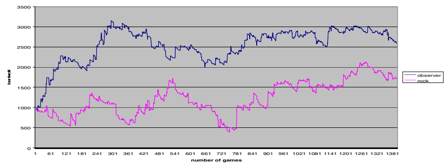

Fig. 5. Bankroll of the Tight Passive (Rock) Observer and Non-Observer agents

The most conclusive results are with passive agents, Observer besides having always a big advantage from non observer, the results are also very good, reaching a good level of bankroll. With aggressive agents, the simulations seem to be a bit inconclusive due to big variations of bankroll that sometimes causes the end of the game too soon for an agent. Although, we can conclude that Opponent Modeling could help these kinds of agents to keep in game for a long time.

6 Conclusions and Future Work

The results achieved with all the agents developed showed the usefulness of the opponent modeling techniques developed. In most of the tests it is possible to verify that the Observer agent has clearly better results than a non observer agent, even when the strategy of hand selection is not very good.

In this project several other techniques were not considered. So, the agent developed is not, globally, a great poker player if compared with good human poker players. However, the main objective was reached and the agent is capable of modeling opponents and effectively using the models to improve its playing style which is an added value to future work in this area.

Future work done in Artificial Intelligence applied to poker may use the work done in this project and the conclusions achieved. The agent developed till the moment must be explored in several other topics, like learning to play in function of the position at the table and bluffing. These topics could be better explored considering Opponent Modeling.

In the domain of player classification, future projects could tune the approach done in this work. The Opponent Modeling described intended to be very simple and basic like a first approach. Future work may include: to consider more than the 4 type of players; analyze other player style variables; and retrieve information from the cards shown at showdown.

References

[1] Billings, D., Papp, D., Schaeffer, J., Szafron, D.: Opponent modeling in poker. In: American Association of AI National Conference, AAAI 1998, pp. 493–499 (1998)

[2] Davidson, A.: Opponent modeling in poker. Master's thesis, Department of Computing Science, University of Alberta (2002)

[3] Schaeffer, J., Culberson, J.C., Treloar, N., Knight, B., Lu, P., Szafron, D.: A world championship caliber checkers program. Artificial Intelligence 53(2-3), 273–289 (1992)

[4] Buro, M.: The Othello match of the year: Takeshi Murakami vs. Logistello. International Computer Chess Association Journal 20(3), 189–193 (1997)
[5] Newborn, M.: Kasparov versus deep blue: computer chess comes of age. Springer, New York (1996)
[6] Donninger, C., Lorenz, U.: Hydra chess webpage (consulted) (March 2008), http://hydrachess.com
[7] Tesauro, G.: Temporal difference learning and TD-gammon. Communications of the ACM 38(3), 58–68 (1995)
[8] Sheppard, B.: World-championship-caliber scrabble. Artificial Intelligence 134(1-2), 241–275 (2002)
[9] Kan, M.: Post-game analysis of poker decisions. Master's thesis, Department of Computing Science, University of Alberta (2006)
[10] Wikipedia. Game theory. Wikipedia: The Free Online Encyclopedia (consulted) (February 2008), http://en.wikipedia.org/wiki/Game_theory
[11] von Neumann, J., Morgenstern, O.: The Theory of Games and Economic Behavior. Princeton University Press, Princeton (1944)
[12] Billings, D., Papp, D., Schaeffer, J., Szafron, D.: Poker as a testbed for machine intelligence research. In: Mercer, R., Neufeld, E. (eds.) Advances in Artificial Intelligence, AI 1998, pp. 228–238. Springer, Heidelberg (1998)
[13] Davidson, A., Billings, D., Schaeffer, J., Szafron, D.: Improved opponent modeling in poker. In: International Conference on Artificial Intelligence, ICAI 2000, pp. 1467–1473 (2000)
[14] Sturtevant, N.: A comparison of algorithms for multi-player games. In: Schaeffer, J., Müller, M., Björnsson, Y. (eds.) CG 2002. LNCS, vol. 2883, pp. 108–122. Springer, Heidelberg (2003)
[15] UA GAMES Group. The University of Alberta GAMES Group (consulted) (March 2008), http://www.cs.ualberta.ca/~games
[16] Billings, D., Davidson, A., Schaeffer, J., Szafron, D.: The challenge of poker. Artificial Intelligence 134(1-2), 201–240 (2002)
[17] Billings, D., Burch, N., Davidson, A., Holte, R.C., Schaeffer, J., Schauenberg, T., Szafron, D.: Approximating Game-Theoretic Optimal Strategies for Full-scale Poker. In: IJCAI 2003, pp. 661–668 (2003)
[18] Billings, D.: Ph.D. dissertation. Algorithms and Assessment in Computer Poker. Department of Computing Science, University of Alberta, Canada (2006)
[19] Afonso, D., Silva, H.: Aplicação para jogar Texas Hold'em Poker (2007)
[20] Southey, F., Bowling, M., Larson, B., Piccione, C., Burch, N., Billings, D., Rayner, C.: Bayes' bluff: Opponent modelling in poker. In: 21st Conference on Uncertainty in Artificial Intelligence, UAI 2005, pp. 550–558 (July 2005)
[21] Carmel, D., Markovitch, S.: Incorporating opponent models into adversary search. In: American Association of AI National Conference, AAAI 1996, pp. 120–125 (1996)
[22] Sklansky, D., Malmuth, M.: Hold'em Poker for Advanced Players, 2nd edn. Two Plus Two Publishing (1994)
[23] Sklansky, D.: The Theory of Poker. Two Plus Two Publishing (1992)