ELEKTROTEHNIŠKI VESTNIK 79(1-2): 13–18, 2012
ENGLISH EDITION

A Rembrant Poker Bot Program

Gregor Vohl, Borko Bošković, Janez Brest

University of Maribor, Faculty of Electrical Engineering and Computer Science,
Smetanova ulica 17, 2000 Maribor, Slovenia
E-mail: gregor.vohl@gmail.com

Abstract. At the beginning of the 21th century, texas hold'em poker became very popular around the world. It is played with 52 cards, with each player receiving two closed cards and all of the players seeing additional five community cards. Because poker is a relatively unexplored area of artificial intelligence, we decided to accept the challenge and create a program for playing texas hold'em poker. The program is based on an algorithm divided into two parts. The first part, called preflop, is a game phase with closed cards and one stage of betting. The second part, called postflop, is a game phase with community cards and three stages of betting. Preflop is played using common preflop rules and postflop is played according to the principles of the situation play and experiences of the authors. These are the basics for the develop poker program called, "the Rembrant poker bot".

Keywords: poker, texas hold'em, program, poker bot, card game

1 INTRODUCTION

Poker is a card game where players can bet and where the end-combinations are ranked. Different game variations are separated depending on the cards are dealt, rankings of end-combinations, limitations of the bet, and number of bets. There are a couple of poker games with the same betting patterns and end-combination rankings. The history of the poker game is still a mystery. One of the first known games with betting, combination ranking and bluffing, is from the 15th century and is called "Pochspiel". The first basics of poker were shown in the Persian game called "Âs Nas", but unfortunately the first notes weren't made until 1890. At the end of the last century, game historians created different theories about the beginnings of the poker game. Some of them claimed that poker is a form of the Persian game and some that its roots are from the French game "poque" (the name is probably from the Irish word "poca" - pocket). Modern theories deny previous theories, because the trivial poker game played with cards could have developed from any previous game with cards. The unique properties of poker are based on betting. The betting pattern of poker isn't present in any other card games. Based on this theory poker was first played in the first or middle years of the 18th century and expanded until 1800 over areas of the Mississippi river. They usually played "straight" or "draw" poker with 52 cards.

1.1 The Rembrant poker bot

Poker belongs to games with imperfect information. This is a perfect basis for research on similar problems. These problems can be easily transformed into the real world, where we often find ourselves in situations where we don't know all the information needed, before making decisions (imperfect information). Poker bot competitions help us to develop better strategies and methods for artificial intelligence, which can be used in everyday situations (e-education, stock exchange, finances, medicine, etc.).

The first poker bot ORAC was introduced to the world in 1984. The author of the "poker bot" was a professional poker player and author of many poker books, Mike Caro. Mike named the "poker bot" by turning the surname in the opposite direction. In 1984 Mike presented ORAC at the World Series of Poker and everyone was excited. ORAC also includes the pot odds calculator, for calculating the bet amount and the chips in play.

In 2005 there was a "World Series of Poker Robots" competition, where AI experts using poker bots competed for 100,000 dollars.

The next year, 2006, a research group called the "Computer Poker Research Group" from the University of Alberta (Canada) started to host annual competitions, the "Annual Computer Poker Competition" ACPC for poker bots. This competition is organized by the conference "Association for the Advancement of Artificial Intelligence" or "International Joint Conferences on Artificial Intelligence". Only the best "poker bots" from around the world can compete against each other.

Received November 24, 2011
Accepted December 6, 2011

In 2007, The university of Alberta hosted a match ”Man vs Machine” between their poker bot Polaris and invited poker professionals.

The ACPC 2011 was in San Francisco and the ”Rembrant poker bot” was also played there.

2 ALGORITHM

We implemented a program that has a simple and effective algorithm. In order to reach the set target, we relied on our own experiences and knowledge of the poker game. We transformed collected the information into situation rules and then implemented the program. Our main condition in defining situations was in line with a good knowledge of poker rules. Poker rules are formed according to actions and winning card combinations. Our program also contains a communication protocol, thus enabling interaction between the user or other poker bots with the ”Rembrant poker bot”.

2.1 Actions

The action is a move of a single player having the right to play in each phase of betting.

During a betting phase, the algorithm makes one of the following actions:

- Check – the player doesn't bet and gives the right of the play to the next player,
- Bet or raise – the player puts chips into the play,
- Re-raise – the player puts more chips than the player before him into the play,
- Call – the player equals the number of chips from the bet of the previous player,
- Fold – the player doesn't want to play and gives his cards away,
- All-in – the player puts all his chips into play.

2.2 End combinations

The end combinations are calculated during the second phase of the game, called ”postflop”. A winning combination of five cards is formed with two starting cards and five community cards.

End combinations from the top to the bottom are:

- royal flush: A K Q J T of the same colour,
- straight flush : five sequential cards (e.g. 4 5 6 7 8) of the same colour,
- quads: four cards of the same values (e.g. 7 7 7 7 *),
- full house: trips and a pair (e.g. A A A Q Q),
- flush: five cards of the same colour (e.g. 5 6 7 8 9 in spades),
- straight: five sequential cards (e.g. 5 6 7 8 9) of different colours,
- trips: three cards of the same value (e.g. 8 8 8 * *),
- two pairs: (e.g. A A K K *),
- one pair: two cards of the same value (e.g. 9 9 * * *) and
- high card: (e.g. A * * * *).

Table 1 shows a mathematical probability of forming an end-combination on the flop (1st phase of betting in postflop).

Table 1. Mathematical probability for flop

Card combination	Probability to hit on flop [%]
High card	50
One pair	42
Two pairs	4,75
Trips	2,1
Straight	0,39
Flush	0,20
Full house	0,14
Quads	0,024
Straight flush	0,0015
Royal flush	0,0000375

2.3 Communication protocol

The user interface uses a communication protocol to interact with our program. The rrotocol transfers information about actions.

Position : number of the current game : actions : starting cards / community cards

- Position – small blind or big blind position
- Number of the current game – sequential number of the game
- Actions – poker actions (check, call, raise or fold).
- Betting phases are separated with ”/”.
- Starting and community cards – each card is labeled with a colour and value. Valid card values are: 2, 3, 4, 5, 6, 7, 8, 9, T, J, Q, K, A, where T = ten, J = jack, Q = queen, K = king, A = ace. The valid card colours are: c (club), s (spade), d (diamond) and h (heart). Card example: Ts means ten of spades.

The formal note of the protocol in BNF (Backus-Naur Form) is:

<**command**> ::= <position> ”:” <num. of current game> ”:” <actions> ”:” <starting cards> ”/” <community cards>
<**position**> ::= ”1” | ”0”
<**num. of current game**> ::= <sequential integer>
<**actions**> ::= ”f” | ”c” | ”r” <bet>
<**bet**> ::= <bet amount>
<**starting cards**> ::= <value><colour> <value><colour>
<**community cards**> ::= <value><colour> <value><colour> <value><colour> ”/” <value><colour> ”/” <value><colour>
<**colour**> ::= ”h” | ”s” | ”c” | ”d”
<**value**> ::= ”2” | ”3” | ”4” | ”5” | ”6” | ”7” | ”8” |

”9” | ”T” | ”J” | ”Q” | ”K” | ”A”

Message example:
0:13:b1b2r6r18r30c30/r32r96:Ks2d|/5s2s6h

0:13:b1b2r6r18r30c30/r32**r96**:Ks2d|/5s2s6h
The message contains the last action of the poker bot and indicates that it is the opponent’s move.
0:**13**:b1b2r6r18r30c30/r32r96:Ks2d—/5s2s6h
13th game in progress.
0:13:**b1b2r6r18r30c30/r32r96**:Ks2d—/5s2s6h
Small blind (b1), big blind (b2), preflop actions and actions on flop. Poker bot made a bet of 6 chips and the opponent re-raised to 18. Poker bot re-raised again to 30. The opponent called. After the end of the 1st betting phase, the opponent made a minimum bet of 32 chips. The poker bot re-raised to 96.
0:13:b1b2r6r18r30c30/r32r96:**Ks2d**—/5s2s6h
The poker bot has king of spades and two of diamonds.
0:13:b1b2r6r18r30c30/r32r96:Ks2d**—/5s2s6h**
Three community cards on flop: five of spades, two of spades and six of hearts.
0:13:b1b2r6r18r30c30/**r32r96**:Ks2d—/5s2s6h
The game is currently in the 2nd phase of betting - two actions on flop. The player with position 0, in this case the opponent putted 32 chips into play and the player in position 1, in this case, poker bot putted 96 chips into the play. The opponent must place additional 64 chips into the play to stay in the game.
0:13:b1b2r6r18r30c30/r32r96:**Ks2**d**—**/5**s2s**6h
The poker bot has a bottom pair (pair of twos) and has the possibility for a flush draw (three cards are spades).

3 Description of the algorithm

The main idea of our algorithm was to implement a simple and effective strategy for playing poker. We used methods of the situation play defined with our poker knowledge. The algorithm is divided into two phases, i.e. preflop and postflop.

Preflop is the phase of the game before community cards, when each player receives two closed cards (the player can see only his own cards). Postflop is the phase of the game with community cards. The community cards are divided into three stages: flop, turn, and river. Three community cards are shown on flop and one additional card on turn and river.

3.1 Preflop

The algorithm description for the preflop phase is as follows:
Input: combination of the starting cards and the opponent’s action
Output: poker bot action.
Algorithm:

- calculation of the group for starting cards,
- collecting information about the opponent’s action and
- selection of a proper action.

The preflop phase is the phase of the game before the community cards. Each player gets two closed cards and takes preflop actions. From a packet of 52 cards, it is possible to form 1326 different starting-card combinations. To achieve all the combinations, we multiply the number of all cards (52) with the number of cards when one card is already dealt (51). As some combinations are repeated, they are divided by 2. The best starting cards are two aces (the highest possible pair). The worst starting hand is 27 off-suit. A combination of cards with 2 and 6 can still form a straight. Card 7 is the first card which can’t form a straight with only 3 community cards. In live games, there is often a situation with players trying to bluff with 27 and to have fun with the opponents after winning the pot and showing them starting cards.

Our algorithm for playing preflop divides all the possible starting hands into 9 different groups. Each starting hand belongs to one of the groups. Each group has its own actions, depending on the opponent’s actions. The cards are grouped based on the combinations and colours.

Group 1 represents the strongest starting hands. Actions are shown in Table 2. Group 1 is formed with the two highest pairs AA and KK. This group is also called ”premium cards” and is very hard to beat. To beat it, the opponent needs at least two pairs.

Table 2. Actions for group 1

Opponent	Poker bot
1st position, check or call	Raise 3-5x BB
Raise	Re-raise 3-4x of bet
Re-raise	All-in
All-in	Call

Group 2 is formed with the next pairs QQ and JJ, and a combination of two highest cards AK and AKs. The letter ”s” means that the cards are of the same colour - same suite. The actions are shown in Table 3.

Table 3. Actions for group 2

Opponent	Poker bot
1st position, check or call	Raise 3-5x BB
Raise	Re-raise 3-4x of bet
Re-raise	QQ or JJ: call AK or AKs: All-in
All-in	Call

Group 3 is formed with the next pair TT. In this group there are also the following combinations: AQs, AQ, AJs and KQs. The actions are shown in Table 4.

Group 4 is formed with the next pairs 99, 88 and 77. In this group there are also the following combinations:

Table 4. Actions for group 3

Opponent	Poker bot
1st position, check or call	Raise 3-5x BB
Raise	Call
Re-raise	Call
All-in	Fodl

Table 5. Actions for group 4

Opponent	Poker bot
1st position, check or call	Raise 3-5x BB
Raise	Raise <= 40 call else fold
Re-raise	Raise <= 40 call else fold
All-in	Fodl

ATs, AT, KJs and AJ. The actions are shown in Table 5.

Group 5 is formed with the following combinations: KQ, QJs, KTs and JTs. The actions are shown in Table 6.

Table 6. Actions for group 5

Opponent	Poker bot
1st position, check or call	Raise 3-5x BB
Raise	Raise <= 30 call else fold
Re-raise	Raise <= 30 call else fold
All-in	Fodl

Group 6 is formed with the following combinations: KJ, QTs, A9s, A8s and QJ. The actions are shown in Table 7.

Table 7. Actions for group 6

Opponent	Poker bot
1st position, check or call	Raise 3-5x BB
Raise	Raise <= 20 call else fold
Re-raise	Raise <= 20 call else fold
All-in	Fodl

Group 7 is formed with the following combinations: AXs where X is smaller than 8, KT, JT, K9s, Q9s, T9s and J9s. The actions are shown in Table 8.

Group 8 is formed with the following combinations: AX where X is smaller than T, suited connectors XYs, where X and Y are smaller than T and aren't equal. The actions are shown in Table 9.

Group 9 is formed with the remaining starting hands. The actions are shown in Table 10.

3.2 *Postflop*

The postflop is the phase of the game with community cards, and is divided into flop, turn, and river. Three community cards are shown on flop and, one extra card on turn and river.

Description of the postflop algorithm:
Input : the starting hand, community cards, and opponent's action,
Output : the poker bot's action

Table 8. Actions for group 7

Opponent	Poker bot
1st position, check or call	Raise 3-5x BB
Raise	Raise <= 20 call else fold
Re-raise	Fold
All-in	Fodl

Table 9. Actions for group 8

Opponent	Poker bot
1st position, check or call	Check
Raise	Raise <= 5 call else fold
Re-raise	Fold
All-in	Fodl

Our algorithm calculats the current best end-combination formed from the starting and community cards. The input of the function for calculating the end-combination are the starting cards and the current community cards. The best combination of five cards is the current best end-combination. To follow is the phase of collecting information of the opponent's actions - a function for decoding a message from the opponent is called. At last the function for calculating the poker bot's action is called.

Our algorithm calculating the current end-combination is implemented in three cycles:

- The erase function is called each time before the betting phase. The function deletes all the old information about the end-combinations from the previous betting phase.
- After the erase function is finished, the function to compare between the starting and community cards is called. This function first checks the values of the cards in order to calculate the basic combinations (pair, two pairs, trips, set, full house, quads). After that the function checks if there is a sequence of three or more cards (after the cards are sorted - probability for a straight). Then the function checks the colours of the cards to find three or more cards of the same colour (a probability for a flush).
- In the last cycle of the process, the previously collected information is evaluated to form the best end-combination. The function checks the number of the found pairs in the previous step. Based on the number of pairs, the end-combinations par, two pairs, trips, full house and quads are formed. Some of these combinations are divided into sub-combinations. e.g. one pair has five different sub-combinations: over-pair, top-pair, middle pair, bottom-pair and under-pair. The function checks for the straight and flush draws and re-evaluates the combinations to check if there is a possible stronger combination. If no such combination is found, the combination "high-card" is returned.

The algorithm plays separate phases based on the

Table 10. Actions for group 9

Opponent	Poker bot
1st position, check or call	Check
Raise	Fold
Re-raise	Fold
All-in	Fodl

community cards shown in a separate stage of postflop. This is the basic principle of the situation play. A passive play with a good combination and an aggressive play in the case of draws. In case of draws, a good combination can be beaten by allowing the opponent's free cards to form a better combination.

The algorithm uses a mixed play strategy in order to confuse the opponent. It's very hard for the opponent to determine the playing strategy of the poker bot. The playing strategy is implemented so as to never play the same situation too often in a row. Because of the situation play, there is no mathematical equation for the probability of making decisions about playing an action or not.

The most important phase in postflop is flop. During this phase, the algorithm tries to collect all the possible information from the opponent in order to determine what is the starting hand of the opponent. With this collected information on flop, the playing strategy is adjusted for turn and river.

The algorithm checks what the current combination is at every betting phase, and what could be the best combination if the remaining community cards are good. In each phase, the community cards danger is also calculated to inform poker bot when the opponent may have a flush or straight draw options. If there are at least two community cards of the same colour or have sequence values, poker bot plays very aggressively to get the opponent out of the game if he is on a draw. If there are three community cards of the same colour or have sequence values, poker bot stops the aggressive play and only makes calls to the opponent's bets. If there are four community cards of the same colour or have sequence values, poker bot won't put any additional chips into play.

The algorithm divides the end-combinations into groups. The actions are calculated based on the collected information and the current end-combination.

An end-combination high-card represents an end-combination where poker bot hits nothing. Table 11 shows the actions for an end-combination high-card for all three phases of postflop.

A one-pair end-combination represents an end-combination of the two cards with the same values. Table 12 shows the actions for, an end-combination one-pair for all five sub-combinations of one pair.

Table 11. Postflop, group 1 – High card

Opponent	Flop	Turn	River
Raise	Fold	Fold	Fold
Re-raise	Fold	Fold	Fold
All-in	Fold	Fold	Fold
Check	Check	Check	Check
1st position, check or call	Check/Raise	Check	Check

Table 12. Postflop, group 2 – one pair

Opponent	Top pair	Middle pair	Bottom pair	Over pair	Under pair
Raise	Re-Raise	Call	Fold	Re-Raise	Fold
Re-raise	All-in	Fold	Fold	All-in	Fold
All-in	Call	Fold	Fold	Call	Fold
Check	Check /Raise	Check /Raise	Check	Check /Raise	Check
1st position, check or call	Check /Raise	Check /Raise	Check /Raise	Check /Raise	Check /Raise

A two-pair end-combination represents an end-combination with two different pairs. Table 13 shows the actions for end-combination two-pairs for all five sub-combinations of two pairs.

Table 13. Postflop, group 3 – two pairs

Opponent	2 starting and 2 community	1 started and 1 community + pair community cards	pocket pair and pair community cards	2 pairs community cards
Raise	Re-Raise	Call/Re-Raise	Call/Re-Raise	Call/Fold
Re-raise	All-in	Call/All-in	Re-Raise/Call	Fold
All-in	Call	Call/Fold	Call/Fold	Fold
Check	Raise	Raise	Raise	Raise
1st position, check or call	Raise	Raise	Raise	Raise

A trips end-combination represents an end-combination with three cards of the same value. Table 14 shows the actions for the trips end-combination for two sub-combinations of trips.

Table 14. Postflop, group 4 – trips

Opponent	Trips	Set
Raise	Call/Re-Raise	Re-Raise
Re-raise	Call	Call/All-in
All-in	Call	Call
Check	Check/Raise	Check/Raise
1st position, check or call	Check/Raise	Check/Raise

A ”straight” end-combination represents an end-combination with five cards of sequential values. Table 15 shows the actions for a straight end-combination for all sub-combinations of straight.

Table 15. Postflop, group 5 – straight

Opponent	Up	End	Up-End	Got shot
Raise	Call /Re-Raise	Call	Call /Re-Raise	Call
Re-raise	Call /All-in	Call /Fold	Call /All-in	All-in
All-in	Call	Call /Fold	Call	Call
Check	Check /Raise	Check /Raise	Check /Raise	Check /Raise
1st position, check or call	Check /Raise	Check /Raise	Check /Raise	Check /Raise

A ”flush” end-combination represents an end-combination with five cards of the same colour. Table 16 shows the actions for a flush end-combination flush for two sub-combinations of flush.

Table 16. Postflop, group 6 - flush

Opponent	Nuts - flush with ace	Rest
Raise	Call/Re-Raise	Re-Raise
Re-raise	Call/All-in	All-in
All-in	Call	Call/Fold
Check	Check/Raise	Check/Raise
1st position, check or call	Check/Raise	Check/Raise

End-combinations of ”full house”, ”quads”, ”straight flush” and ”royal flush” are played with the same actions. As these combinations are very rare, they are played together as one group. They are very hard to beat. Table 17 shows the actions for groups 7, 8, 9 and 10.

Table 17. Postflop, group 7,8,9 and 10

Opponent	Poker bot
Raise	Call/Re-Raise
Re-raise	Call/All-in
All-in	Call
Check	Check/Raise
1st position, check or call	Check/Raise

4 Testing

The Rembrant poker bot was played at the ACPC 2011 competition in San Francisco. The competition was divided into three groups: limit and no-limit texas hold’em for two and for three players. The Rembrant poker bot was played no-limit texas hold’dem for two players called, heads up. Each game was played with 6000 hands. At half of the hands, the players played with opposite cards to avoid the luck factor. Inside the no-limit heads up there were two groups: ”total bankroll” and ”instant run-off”. In the category ”total bankroll”, the Rembrant poker bot ended in the 6th place though it beat the champion. In the category ”instant run-off”, the Rembrant poker bot ended in the 4th place.

This result proves the relatively easy algorithm to be a good poker player.

5 Conclusion

A program for playing the texas hold’em poker was implemented based on simple situation rules. They were laid down by using our own experience and knowledge. The algorithm was divided into two phases, e.g. a phases with and a phase without community cards.

The Rembrant poker bot competed at the ACPC competition in 2011, San Francisco, where it achieved the 6th and 4th place in texas hold’em heads up. Based on the achieved results, a conclusion can be drawn that using this simple and easy algorithm makes the program to be a good poker player.

Our future work will be towards improving the Rembrant poker bot by transforming the situation rules into decision trees. Thus making the Rembrant poker bot will faster and more robust. The decision trees will include information about opponents. The algorithm will memorize the patterns of the opponent’s play and will produce better actions by knowing the opponent’s profile.

References

[1] Robert Frederic Foster Foster’s complete Hoyle F.A.Stokec Company, 1909
[2] http://www.legendsofamerica.com/we-poker.html
[3] http://www.pokerbot.si
[4] Mike Caro Caro’s book of poker tells. The psychology and body language of poker. Cardoza publishing New York, 2003.
[5] http://www.aaai.org
[6] http://ijcai.org/
[7] http://webdocs.cs.ualberta.ca/ games/poker/man-machine/2007/
[8] David Sklansky and Ed Miller. No Limit Hold ’em, Theory and practice. Creel Printing, Inc. Las Vegas, Nevada, junij 2006
[9] Sam Braids The intelligent guide to Texas Hold’em poker Intelligent Games Publishing, 2003
[10] Ed Miller, David Sklansky, Mason Malmuth Small stakes Hold ’em – winning big with expert play Creel Printers, Inc. Las Vegas, Nevada, Januar 2005
[11] http://www.articlesbase.com/online-gambling-articles/world-series-of-poker-robots-1230821.html

Gregor Vohl received his B.S. degree in computer science in 2011 from the Faculty of Electrical Engineering and Computer Science, University of Maribor. At the moment he is a Ph.D. student.

Borko Bošković received his B.S. and Ph.D. degrees in computer science from the University of Maribor, Maribor, Slovenia, in 2004 and 2010 respectively. He is currently a teaching assistant at the Faculty of Electrical Engineering and Computer Science, University of Maribor. His research is focused on chess algorithms and evolutionary computing. His areas of expertise also include programming languages, integrative programming and natural language processing.

Janez Brest received his B.S., M.Sc, and Ph.D. degrees in computer science from the University of Maribor, Maribor, Slovenia, in 1995, 1998 and 2000, respectively. He is currently a full professor at the Faculty of Electrical Engineering and Computer Science, University of Maribor.