University of Alberta

Library Release Form

Name of Author: Morgan Hugh Kan

Title of Thesis: Postgame Analysis of Poker Decisions

Degree: Master of Science

Year this Degree Granted: 2007

Morgan Hugh Kan

Date: ____________________

University of Alberta

POSTGAME ANALYSIS OF POKER DECISIONS

by

Morgan Hugh Kan

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of **Master of Science**.

Department of Computing Science

Edmonton, Alberta
Spring 2007

University of Alberta

Faculty of Graduate Studies and Research

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research for acceptance, a thesis entitled **Postgame Analysis of Poker Decisions** submitted by Morgan Hugh Kan in partial fulfillment of the requirements for the degree of **Master of Science**.

Jonathan Schaeffer
Supervisor

Michael Bowling

Michael Carbonaro
External Examiner

Date: ____________________

To my parents, Janet and Chay Kan,
and my sister, Megan Kan,
I would never have made it this far without them.

Abstract

In Artificial Intelligence research, evaluation is a recurring theme. A newly crafted game-playing program is interesting if it can be shown to be better by some measure. Usually, this is done by running a series of matches against other programs to generate a statistically significant result. In some games, particularly ones with imperfect information or stochastic elements, it may be infeasible to play enough games to achieve statistical significance.

A new tool called DIVAT is presented for analyzing the quality of decisions in the game of poker. Instead of using the net monetary outcome of each game to evaluate the player, the tool focuses on the skill elements of the game. The expected value of the player's actions are compared to a reasonable baseline policy. The effect of stochasticity and imperfect information is greatly reduced resulting in an unbiased low-variance estimator for the game of poker.

Acknowledgements

There are far too many people to thank, and not enough space for all of them!

First, thanks to **Darse Billings**. Somehow, he managed to not get too annoyed at my constant questions for clarity of what he wanted me to do. Our collaboration on the DIVAT project was a pleasure, and I am excited to continue working with him on future projects. In addition, he helped me an incredible amount with proofreading of this thesis. This thesis is far better due to his help.

A big thank-you goes to my supervisor, **Jonathan Schaeffer**. Despite being an extremely busy man with his department chair responsibilities, he was still consistently available for questions from an ignorant graduate student. He is an amazing person, and has been a tremendous positive influence on my academic career.

Michael Bowling is gifted with the amazing ability to make the most complicated topics easy to understand. It is thanks to him that I understand the DIVAT unbiased proof, and am able to present it in this thesis with clarity.

Other professors that have had a tremendous impact on my academic career include **Ryan Hayward** and **Paul Lu**. Ryan was willing to take me on as an undergraduate student to work on the Mongoose hex project. This project introduced me me to academic research after just my second year of undergraduate studies. Since then, I've returned to the hex project many times. Paul was also my supervisor for two summers, during which I helped work on the Trellis project. Both these professors helped me develop an interest in research and convinced me to continue towards a master's program.

I am extremely grateful for the many people I've had the pleasure of calling my friends over the past several years of university. Among them, in no particular order, are **Jeff Ryan**, **Nolan Bard**, **Michael Johanson**, **Mark Lee**, **Curtis Badke**, **Curtis Onuczko**, **Jeff Siegel**, **Dirk Henkemans**, **Leah Hackman**, **Andrew Butcher**, and many many others. I would never have been able to get where I am today and still be relatively sane without their constant harassment, group gatherings, and other events to keep me pleasantly distracted from my work!

Lastly, I have been incredibly fortunate to have the love and affection of **Diane Barker**. There are not enough words I can say to properly thank her for her incredible patience and tender loving care.

This research was generously supported by NSERC and iCore. I am grateful for their support.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Games have proven to be both interesting and rewarding for research in Artificial Intelligence (AI). Many success stories like Chinook (checkers) [19], Logistello (Othello) [8], Deep Blue [17] and Hydra [11] (chess), TD_Gammon (backgammon) [23], and Maven (Scrabble) [21] have demonstrated that computer programs can surpass all human players in skill. However, there remain many challenges in computer game-playing. Computer Go is a very difficult domain to study due to the size of the search space, and the difficulty of heuristic evaluation. Games such as Poker are difficult because of the elements of imperfect information and partial observability.

As research in AI continues to expand to handle the intricacies of complex and challenging games, particularly ones where stochasticity, imperfect information, and partial observability are pervasive, another problem arises: accurate evaluation. Evaluating the performance of a program in such a domain may require tens of thousands of games to be played to attain statistical significance. With stochasticity providing high variance, a statistically significant result can be difficult to achieve if two players are closely matched in skill. If one of the competitors is human, the number of games may be prohibitive.

Two closely-matched poker players who differ in skill by 0.01 small bets per game (sb/g)[1] might require a match of a million hands to say with 95% confidence that the better player prevails (assuming a standard deviation of 5 sb/g which is

[1] A player's performance in Texas Hold'em is measured using the number of ***small bets*** won per game played. A typical poker professional might win at a rate of 0.05 sb/g, which means that a 0.01 sb/g skill difference between bots is not inconsequential.

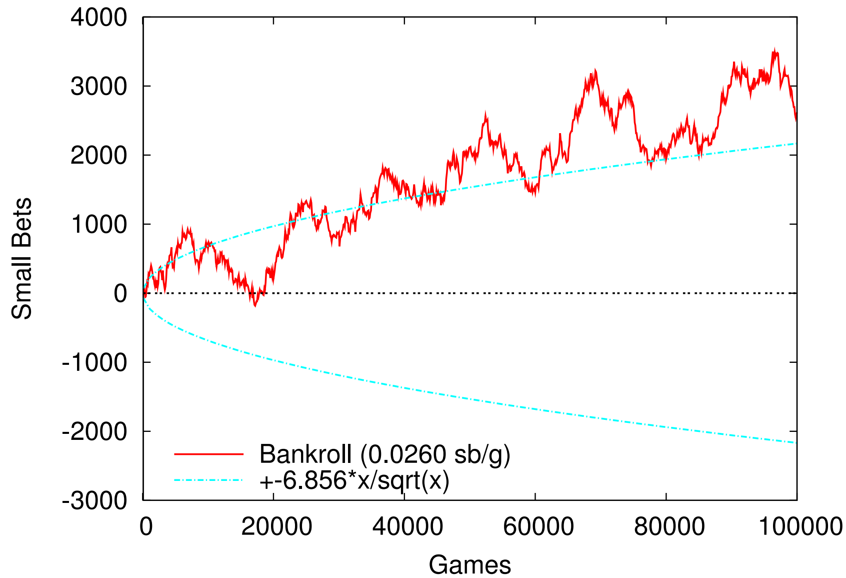


Figure 1.1: An example Texas Hold'em poker match between two players.

a reasonable assumption for a game of heads-up limit Texas Hold'em poker). As poker playing programs improve, the matches between top programs will be decided by increasingly smaller win rates.

An example match between two competitors is presented in Figure 1.1. The bankroll line shows how many small bets one of the players has won over the course of the match. Lasting 100,000 games, the match shows that one of the players has a solid winning rate of 0.026 sb/g. The measured standard deviation for this match was ±6.856 sb/g. The one standard deviation line is plotted on the graph, indicating that the player is winning with statisical confidence of about 70%.

The surprising thing about this match is that the competitors are very simple, and the theoretical result of the match is exactly zero. The "winning" player is playing an always-call strategy, and the "losing" player is playing an always-raise strategy. When these strategies play against each other the pot size is always the same (seven small bets) and the game always progresses to a showdown. The game resembles coin flipping with occasional ties. Obviously, neither strategy truly holds an advantage in this match, and the bankroll graph is very misleading.[2] This exam-

[2] This match was not selected to show an anomaly – it was the outcome of the first match played under these conditions

ple shows why another method for evaluation is needed.

In this thesis, a statisically unbiased estimator, called the Ignorant Value Assessment Tool (DIVAT[3]), is described. The tool provides post-game analysis of heads-up Limit Texas Hold'em decisions, while filtering out most of the luck elements of the game. The components of the tool are simple enough that one could apply the same concepts to many other domains. The work presented in this thesis has also been presented in a number of publications including a technical report with more detailed analysis of several matches [6], the International Computer Games Association Journal (special issue on poker) [7], and Darse Billing's doctoral thesis [3]. The remainder of this introduction includes an introduction to Texas Hold'em poker (Section 1.1), a motivating example for the DIVAT tool (Section 1.2), a description of previous and related work (Sections 1.3 and 1.4), and a description of the layout of this thesis (Section 1.5).

1.1 Texas Hold'em

Texas Hold'em is currently the most popular variant of poker. It is played by millions of people, and is the game used to determine the world champion of poker every year at the World Series of Poker.

Technically, the game can be played with any number of players from 2 through 23, but in practice the number is usually between 2 and 10 players. A white disk called the ***dealer button*** moves around the table in a clockwise direction after each hand and signifies the player who acts last on all rounds after the first. The player to the left of the dealer posts the ***small blind*** and the player to the left of that posts the ***big blind***. These are forced bets which encourage the players to bet (much like antes in other poker variants). After the blinds are posted, each player is dealt two ***hole-cards*** and the game begins.

The first round of betting occurs, starting with the player to the left of the big blind. Each player can choose to ***call*** the amount of the big blind, ***raise*** a particular

[3] The first letter in the acronym is the first initial of the inventor, Darse Billings. This author implemented the original concept, provided experimentation and feedback for refining the metric, and co-developed the final useable DIVAT system.

amount, or ***fold***. The betting round ends when all players have called the latest raise or folded. The initial ***community cards*** are then dealt. Three cards are dealt face up in the center of the table, called the ***flop***. A second round of betting occurs, always starting with the player to the left of the dealer. The next community card, called the ***turn***, is dealt followed by a third round of betting. The last community card, called the ***river***, is dealt with a final round of betting. Those remaining in the hand reveal their hole cards and the winner is the player holding the best five-card poker hand.

The amount a player can raise at a given time is determined by the variant of poker and the stakes being played. In Limit Poker, the amount a player can raise is pre-determined. For instance, in a $10-20 game, a player can raise exactly $10 in the pre-flop and flop betting rounds, and exactly $20 in the turn and river betting rounds. In general, the smaller betting amount is usually half that of the larger bet, and these amounts are known as ***small bets*** and ***big bets*** respectively. In most Limit games, players cannot raise more than four times in any particular round. This is called ***capping the betting***. In some games, however, there may be no limit on the number of raises when there are only two players remaining.

In contrast to Limit poker, two other common variants of Texas Hold'em are Pot Limit and No Limit. In Pot Limit, a player can raise any amount that is less than or equal to the amount currently in the pot. In No Limit, a player can raise any amount up to and including all of the chips they have in play. The research in this thesis focuses on Limit Texas Hold'em, though adaptations to Pot Limit and No Limit Texas Hold'em are certainly possible.

When playing with just two players, or ***heads-up***, a common variant is to use ***reversed blinds***. In this case, the player with the dealer button gets the small blind and the other player gets the big blind. The dealer acts first in the preflop round, and then last on every other round. While both variants are fair, acting last is a large advantage; therefore the reversed blinds format slightly balances out this advantage, and both variants are played in practice.

It is often useful to describe the betting in a game of poker in a compact form. For this purpose, we use a one-line betting sequence with the following conventions:

's' stands for small blind, 'l' stands for the big blind, 'k' for check, 'c' for call, 'b' for bet, 'r' for raise, 'f' for fold, and '/' is used to separate the four betting rounds.[4] To more easily distinguish players, the actions of the second player are capitalized. An example one-line betting sequence for a two-player heads-up reversed-blinds Limit Texas Hold'em game is SlCrC/kK/bC/bRc (producing a final pot size of 16 small bets).

1.2 Motivating Example Game

The work presented in this thesis is applied to the specific domain of two-player heads-up Limit Texas Hold'em poker. The following example should help motivate the need for an analysis tool more powerful than its predecessors. This example will be revisited later on to show how the tool deals with some of the issues raised.

Suppose Alice and Bob are playing a game of Texas Hold'em with reversed-blinds. Bob is the dealer, so he posts a small blind and Alice posts the big blind. Alice is dealt **T♠-6♠** and Bob is dealt **J♢-5♢**. Bob is first to act, and holds a starting hand that is not particularly strong. However, it certainly has potential to improve and folding here would be a mistake. He decides to take the straight-forward action and call. Alice has a similar hand in strength and elects check.

The flop dealt is the **T♢-5♡-T♡** giving Alice an extremely strong three of a kind (***trips***). She decides to play deceptively and checks to Bob with her strong hand hoping he will bet. Bob has hit a five for two pair, and is quite likely to have the best hand. Since his pair is small, and there are two hearts on the board, Bob would like to protect his hand by betting rather than letting Alice draw a ***free card*** that could make her a better two pair or a ***flush***. Therefore, Bob bets. Now that Bob has shown strength, Alice believes it will be more profitable to "charge full-speed ahead", and executes a ***check-raise***. Caught off guard, Bob simply calls.

The turn is the **2♢** which gives Bob a ***flush draw***, but Alice's hand continues to be very strong. Alice simply bets out with her very strong hand. Bob feels he still might have the best hand, perhaps because Alice is an aggressive player who

[4] A check is equivalent to calling when the bet to be called is zero for the current round. Similarly, bet is equivalent to raise when the bet to be called is zero for the current round.

often check-raises with marginal hands. He decides to raise applying maximum pressure if his two pair is the best hand, and having a flush draw to fall back on if it is not. Alice gladly re-raises, and now Bob calls, suspecting that he is beat, but with correct odds to draw to a flush.

The river is the 2♡, which completes a possible heart flush, but also promotes a Ten or Deuce to a full house. Alice can only lose to one possible hand (the **2♠-2♣** for four of a kind), and thus makes a ***bet for value***. Bob calls with his pair, primarily because there is so much money in the pot that making a folding error here would be a very large mistake. Alice wins the 22 small bet pot with her full house.

A one-line summary of the game between Alice and Bob looks like this:

T♠-6♠,J♢-5♢,T♢-5♡-T♡/2♢/2♡ SlCk/kBrC/bRrC/bC (1.1)

After the game is over, how do you evaluate each player's play? Alice certainly looks like the better player, but she was also lucky to make a such big hand.[5] Did Alice make as much money as possible from Bob, or could she have played the hand more profitably? Expert analysis is one possible way to try to discern the better player. However, it is likely that different experts would choose to play the hand differently in either Alice or Bob's seat. Even if experts agree on a particular assessment, there is still no quantifiable amount by which one player outplayed the other. Moreover, this is merely one game between the two players. The goal is to reveal which player will win in the long run, and by how much. These questions are difficult to answer without an objective metric that quantifies the skill differential on each game played.

One way to analyze this game is to consider what would happen had Alice and Bob's seats been reversed. This is known as playing the match in duplicate, and is discussed in more detail in Section 1.3.

For the purposes of illustration, let's suppose the players swapped seats, and the game played out in the following way. Alice is the dealer and posts the small blind with Bob posting the big blind. Alice is dealt **J♢-5♢**, and Bob is dealt **T♠-6♠**.

[5] In general, a strong hand is easier to play than a mediocre one. The decision to fold is a much more drastic decision than choosing to call or raise because it sacrifices all of the pot equity the hand may have. Mediocre hands face more folding decisions than strong hands do.

Alice is first to act before the flop and simply calls with the mediocre hand. Bob also just checks.

On the flop of **T♢-5♡-T♡**, Bob now has the very strong three of a kind. He chooses to play it deceptively and checks, with the intention of waiting for the next betting round when the size of a raise doubles. He may prefer this strategy because Alice is aggressive, and he wants to take advantage of her tendency to bet. Alice has hit two-pair and chooses to bet with what is likely the best hand. Bob simply calls, continuing his ***slow-play***.

With the turn being the **2♢**, Bob wants to try to get the most he can out of this opportunity. He decides to try to check-raise, knowing that Alice is likely to bet. Indeed, Alice does bet. Although she is suspicious of Bob's quiet call, she feels it is more dangerous to not bet her good but vulnerable hand. Bob executes his check-raise, which makes Alice believe her hand is beat. However, because she has a flush draw, there are many cards that will make her a very strong hand, so she chooses to call.

The river is the **2♡**, which doesn't help either player directly, but almost guarantees a win for Bob. Bob bets out knowing that another check-raise will likely fail. Alice believes she is likely beat. Perhaps Bob does not check-raise very often, and only with strong hands; or perhaps she viewed a physical tell which led her to believe that Bob held a strong hand. Since the **2♡** also completes the possible ***semi-bluffs*** Bob may have had, she chooses to fold her hand even though it is still a respectable holding in most cases. Bob wins the 12 small bet pot (his river bet was not called so is not considered part of the pot).

A one-line summary of the duplicate game between Bob and Alice looks like this:

T♠-6♠,J♢-5♢,T♢-5♡-T♡/2♢/2♡ SlCk/kBc/kBrC/bF (1.2)

After comparing the two different games, it is now fairly obvious that Alice played better than Bob in this pair of cases. The difference in the amount won can be viewed as a lower variance estimator for the skill involved in the game. By comparing the two halves of the duplicate game, much of the luck in the game is balanced out, leaving the skill element. However, duplicate matches are not always possible. If Alice and Bob are both computer programs, then it is fairly simple to erase the program's memories before playing the duplicate match. Unfortunately, if either Alice or Bob is human, this no longer works. A desirable analysis tool would not need to consider duplicate games, but could make use of them if they are available, to make the analysis more accurate.

This example only examined one particular game, but poker is much more than one-game instances. The history of preceding games is critically important for proper poker play. A common strategy is to deliberately misplay a number of games so the opponent builds false beliefs. These beliefs can then be exploited later in the match. The existence of this facet of poker play means that building an analysis tool that is perfect for all poker decisions is essentially impossible.

Fortunately, it is possible to build a tool that ignores the previous history, and still reports useful numbers. DIVAT offers a sound baseline for comparison, and reports a quantifiable value which can be used to estimate skill differential between two players playing heads-up Limit Texas Hold'em. DIVAT will be shown to be statistically unbiased, and it has a much lower variance than previous metrics. The tool also offers several features, including round-by-round analysis which can be used used to yield additional insights.

1.3 Previous Work

The poker research group at the University of Alberta previously used two metrics, called EVAT and LFAT, for analyzing poker matches. These metrics have some significant shortcomings that make them less useful for practical analysis.

The Expected Value Assessment Tool (EVAT) was developed by Darse Billings in his early work on poker, and is a hindsight analysis tool. A player's actions are judged by whether each action was "correct" if the opponent's hole cards are revealed. The EVAT score a player receives is the expected value difference between the action the player chooses and the ideal action. In short, the EVAT tool is a quantifiable application of David Sklansky's Fundamental Theorem of Poker (FToP) which states:

> Every time you play a hand differently from the way you would have played it if you could see all your opponents' cards, they gain; and every time you play your hand the same way you would have played it if you could see all their cards, they lose. Conversely, every time opponents play their hands differently from the way they would have if they could see all your cards, you gain; and every time they play their hands the same way they would have played if they could see all your cards, you lose. [22]

Poker, however, cannot simply be mapped to a game of perfect information. To put this in perspective, imagine a player who calls a bet with the second ***nut flush*** only to lose to the player with the nut flush. For example, **K♡-Q♡** versus **A♡-2♡** with a board of **K♠-J♡-3♡-4♣-8♡**. EVAT penalizes a river call by the player holding the **K♡-Q♡**. From the player's point of view though, calling with the second nut flush could be considered a brilliant play, since such a strong hand could easily justify re-raising.

The FToP is not a theorem. It acts as a method of judging the results of a hand that was played, but it is still very much affected by the natural variance of the game. A player can make a fully justifiable play against the distribution of possible opponent cards, but if the player happens to hold a better hand, then both EVAT and the FToP would penalize them. Moreover, supposing a different policy based on hidden information is irrational, and in violation of the fundamental principles of imperfect information games (where the same policy must be used for all nodes in a given information set). A much better heuristic would take into account some form of rational and consistent play when the opponent's cards are unknown.

The Luck Filtering Analysis Tool (LFAT) was developed to compliment EVAT's shortcomings. It is essentially the difference between a player's ***pot equity*** before and after each of the chance events. For example, a player might have 60% pot equity before the flop, but only a 35% pot equity after the flop. The LFAT score would be $-0.25 * potsize$, which is a quantity describing just how much equity the player lost due to the chance event. Note that LFAT makes no attempt to directly measure the quality of the decisions the players make.

The difficulty with using a metric like LFAT is that it only has definite meaning for each of the chance events (the dealing of cards to players or the community cards). There is no obvious way to combine the preflop, flop, turn and river values of LFAT into a score for the player for the whole game. How does one combine the "good luck" of earlier rounds with "bad luck" if the player loses to an unlucky card the river? One could argue that the good luck from earlier rounds was actually bad luck since it lead to a bigger loss in the end. There is no obvious way of working with these scores that removes the interdependence of chance events from each other in a reasonable and helpful manner.

Aaron Davidson invented the "luck-o-meter" metric as a heuristic measure of luck [10]. While this metric is correlated with luck, it is not expressed in the same units of money won or lost. As a result, a "fudge factor" has to be applied to scale each game's score, which makes the tool imprecise and less useful.

With computer programs, another method of comparing the relative strength is by putting them in the same situations. In other words, the effect of luck can be dampened by re-playing the hand with reversed seats so that player A gets the cards that player B received in the corresponding game. The motivating example earlier in this chapter is an example of this approach. Darse Billings first proposed running duplicate matches for variance reduction in his master's thesis [2]. This approach requires that the random events for the match be either predetermined or seeded in such a way that the same cards are dealt in two matches with the seats reversed. As well, the programs must be restarted for the second half of the duplicate match with no memory of the first half. A duplicate poker match is similar in nature to duplicate bridge tournaments.

In experiments performed by the author, duplicate matches were shown to greatly reduce the variance in the bankroll metric, so they are certainly a powerful way of conducting matches between programs. However, running a duplicate match is not always plausible. In particular, when humans are involved there is no way to "restart" the human with no memory of the recent half of a duplicate match.[6] An

[6] Even delaying the repetition for several thousand hands has shown to be detectable by an experienced player.

analysis tool is needed precisely for the situations where humans are involved in the match. Two computer players can play a hundred thousand hands in a few days of constant play to achieve statistical confidence, however human testers and opponents will never play matches this long. Although duplicate matches are very powerful, they do not fulfill all of the needs that an analysis tool like DIVAT does.

1.4 Related Work

David Wolfe has applied a method to blackjack that is very similar to the ideas presented in this thesis for poker [26]. The purpose of his analysis is to identify the skilled players from the gambling ones at the blackjack table. The method presented in his paper is almost identical to the motivational example on blackjack in Chapter 2. Blackjack differs from poker in that it is essentially a one-player perfect information game, where the notion of a “best move” is well-defined. One simply needs to choose the highest expected value action at every decision point. In poker, however, multiple actions can carry the same maximum expected value, which means there is no notion of a single “best move” in general.

Also similar in spirit is the work of Matej Guid and Ivan Bratko on analyzing the skill levels of the world chess champions through time [13]. The historical games of chess that the grandmasters played were analyzed, with each move compared to a baseline move chosen by an expert computer program. Some interesting results were achieved with their method, but it is not statistically unbiased, and has been severely criticized by other computer scientists. However, the notion of comparing actual actions to a rational baseline is shared with the DIVAT approach.

The work on the DIVAT tool depended a great deal on the work of other researchers at the University of Alberta, past and present. Work on poker at the University of Alberta began in 1992 with Darse Billings, and grew to a full fledged Computer Poker Research Group (CPRG) formed in 1997. The group has worked on poker bots and associated tools with many successful projects to its name. This work depended on the expertise of several people in the area, and many of the experiments that were run were made possible due to the proliferation of different programs to test with.

1.5 Thesis Outline

In Chapter 2, further motivation for the discussion of low-variance estimators is presented for the example domain of blackjack, with emphasis on the intuition used behind the DIVAT metric. In Chapter 3, the DIVAT baseline strategy is introduced, and is used to calculate a much more accurate expected value assessment of a particular game. The DIVAT analysis tool in its full form is then presented. In Chapter 4, the proof that DIVAT is statistically unbiased is presented, along with empirical evidence that it is not only unbiased but also robust. In Chapter 5, the insightful analysis the DIVAT tool provides is applied to several noteworthy matches the University of Alberta programs have played over the past several years. The thesis is concluded in Chapter 6 with a discussion of future work.

Chapter 2

Motivational Example: Blackjack

Before the DIVAT tool is presented, some motivation can first be built by examining an analogous approach for the game of blackjack. Blackjack is a nice domain to examine for this purpose because it is well understood. The exact expected value of all potential player actions can be calculated which means a perfect strategy is simple: choose the available action that maximizes expected value. The ideas presented in this chapter are very similar to Wolfe's techniques [26], but are presented differently to motivate the application to the poker domain.

2.1 Blackjack Rules

Blackjack is a game played against a dealer. The player starts by making a wager, and then is dealt two cards face up. The dealer exposes one of his two cards, at which point the player has the choice to either take another card off the top of the deck (*hit*), or stop taking cards (*stand*). If the total of the player's cards exceeds 21, then the player *busts* and their wager is lost. Numbered cards are worth their face value, face cards are worth ten points and aces are worth either one or eleven points.

Once the player has chosen to stand, the dealer then tries to make a better score than the player. His strategy to do so is predefined: he will hit until he has a score of 17 or higher. If the dealer busts, or the player's score beats the dealer's then the player gets paid double their wager. In the event of a tie, the player gets their wager back.

In addition to the basic actions, a player is sometimes allowed to *double down* or *split*. In a double down, the player doubles the amount of the wager to receive exactly one more card from the deck. This action cannot be used once the player has already hit. In a split, the player must have been dealt two cards with the same value (eg., a Jack and a Ten, or two Eights). The player can choose to split the cards into two separate hands which each receive another card from the deck. The player assigns the original wager to one of these hands and pays the same wager to the other. From this point on, the two hands are played separately.

Blackjack is significantly different from all other casino table games in that there is a *memory*. Blackjack is a house-edge game when no cards have been dealt from the shoe.[1] However, once cards have been dealt, the distribution of cards left in the shoe changes. A player who is keeping track of the distribution can determine when favourable situations arise, and make larger bets to gain an advantage over the house. Therefore, there is an element of skill in blackjack known as ”card counting”. This knowledge was brought to the public's attention by Edward Thorpe in his 1962 book *Beat the Dealer* [24]. Several other books have been written on the subject, including *Bringing Down the House* [16] – the story of the MIT blackjack team.

2.2 Variance and Assessment Problems

Suppose Alice and Bob wanted to decide who was a better blackjack player. A simple way to answer this question would be to have them play a statistically significant number of games of blackjack, and compare the bankroll of both players at the end. How many games would they have to play to be able to conclude that Alice was better than Bob? Suppose Alice is capable of winning 0.01 bets/game more than Bob on average. The average amount won or lost on a given game is probably a little more than 1 bet given the other options the player has (splitting and doubling down). In practice, because of the memory of the deck, good or bad streaks between shuffles can add to the variance of the game and the standard de-

[1] The shoe in blackjack holds the remaining undealt cards.

viation is 1.148 bets [20].[2] To determine that Alice is better than Bob we require $0.01 = 2 * (1.148)/\sqrt{n}$ games. Hence $n = 52,716$ games would be required to say that Alice is better than Bob with 95% certainty. If the games were played at a rate of 500 per hour, this means roughly 100 hours of continuous play!

By the definition of statistical variance, as the skill differential between Alice and Bob decreases, quadratically more games are required to achieve a statistically significant conclusion. Computer players can easily play enough games to do this, but the same cannot be said if Alice and Bob are human because they are prone to losing concentraction or making other human errors. Once n gets into the thousands of games, this assessment starts to become difficult, time consuming and inaccurate.

The main problem with this bankroll assessment technique is that players may make objectively good decisions, but lose due to the stochastic element of the game. For instance, the player could be dealt a six and a five for a total of 11 and correctly choose to double down when the dealer shows a six. The player receives a four for 15, while the dealer flips over a 10 and hits a four to make 20. There was nothing wrong with the player's play, but the dealer got lucky and won two bets from the player. A better assessment tool would factor out the luck elements of a game which the player has no control over, and instead focus on the *decisions* the player makes during the game.

2.3 Decision Based Assessment

Consider this alternative method of assessing a player's skill. Suppose we define a *baseline strategy* that defines a typical rational strategy. This baseline strategy need not be perfect, but is used as a "measuring stick" to compare the player's actions to. Each action carries with it an expected value for that particular situation. When the expected value of both the baseline strategy's recommendation and the player's actual action are compared, one of three things can happen:

- The player chooses an action which has a lower expected value than the baseline. In this case, the player is penalized with the difference between the

[2] The variance a player experiences is determined by the strategy they use. This is the variance if the player employs the basic strategy

expected values of the baseline and the player's actual action.

- The player chooses an action which has a higher expected value than the baseline. In this case, the player is rewarded with the difference between the expected values of the baseline and the player's actual action. If the baseline strategy is perfect, then this case cannot happen, because the perfect strategy would not include decisions that have a lower expected value than another option (although different choices could have the same expected value).

- The player chooses the same action as the baseline, or another action that has the same expected value. In this case the player is neither rewarded nor penalized since the difference in expected values is zero.

Blackjack is a well analyzed game, and it carries with it two desirable characteristics for our example. The first is that it is fairly simple to calculate the expected value for a decision given the player's current cards, the dealer's upcard and the distribution of cards left in the deck. Second, blackjack has a simple *basic strategy* which can be used as a baseline strategy. A sample basic strategy is presented in Table 2.1. For each state, the maximum expected value action is taken with the assumption that the deck is always complete (no cards are missing).

The formal definition of our proposed assessment technique is as follows, with EV meaning expected value:

$$score = EV(actualaction) - EV(baselineaction). \tag{2.1}$$

Over the course of a few hundred games of blackjack, much of the stochastic *luck* elements of the game have been removed because a player's score is determined by expected values and not by the random outcomes for that particular game. Note that not all the stochastic luck is removed. For example, a player may unluckily be forced into situations where he plays badly.

Note that the baseline actions can potentially be the maximum expected value action for that particular situation. In this case, the player's score would only be negative or zero to reflect the number and size of errors that the player made. While this is a perfectly good tool, the baseline need not be perfect. Using an imperfect

	Dealer's Card									
	2	3	4	5	6	7	8	9	T	A
8	H	H	H	H	H	H	H	H	H	H
9	H	D	D	D	D	H	H	H	H	H
T	H	D	D	D	D	H	H	H	H	H
11	D	D	D	D	D	D	D	D	D	D
12	H	H	S	S	S	H	H	H	H	H
13	S	S	S	S	S	H	H	H	H	H
14	S	S	S	S	S	H	H	H	H	H
15	S	S	S	S	S	H	H	H	H	H
16	S	S	S	S	S	H	H	H	H	H
17	S	S	S	S	S	S	S	S	S	S
A,2	H	H	H	D	D	H	H	H	H	H
A,3	H	H	H	D	D	H	H	H	H	H
A,4	H	H	D	D	D	H	H	H	H	H
A,5	H	H	D	D	D	H	H	H	H	H
A,6	H	D	D	D	D	H	H	H	H	H
A,7	D	D	D	D	D	H	H	H	H	H
A,8	S	S	S	S	D	S	S	S	S	S
2,2	P	P	P	P	P	P	H	H	H	H
3,3	P	P	P	P	P	P	H	H	H	H
4,4	H	H	H	P	P	H	H	H	H	H
5,5	D	D	D	D	D	D	D	D	H	H
6,6	P	P	P	P	P	H	H	H	H	H
7,7	P	P	P	P	P	P	H	H	H	H
8,8	P	P	P	P	P	P	P	P	P	P
9,9	P	P	P	P	P	S	P	P	S	S
T,T	S	S	S	S	S	S	S	S	S	S
A,A	P	P	P	P	P	P	P	P	P	P

Table 2.1: A Typical Blackjack Basic Strategy (from Wizard of Odds [20]). H Stands for Hit, S for Stand, D for Double Down and P for Split

baseline helps motivate the DIVAT tool since there is no single perfect poker baseline. Since this technique works with an imperfect baseline, the application of this idea to poker is justified.

2.4 Examples

In this section, some example blackjack games are described. While the situations are not likely to be realistic,[3] they demonstrate the fundamental ideas of this assessment technique. The expected value calculations are provided by a blackjack calculator from gamblingtools.net [15].[4]

Suppose that the player is dealt a four and a five for a total of nine. The dealer's upcard is a seven. The baseline action in the basic strategy chart is to hit. Regardless of the distribution of cards in the deck, the player will receive neither a bonus nor a penalty for choosing to hit. Suppose that the game started from a fresh shoe, and the player decides to deviate from the basic strategy by doubling down instead of hitting. Note that starting from a fresh shoe means the baseline strategy is perfect, and any deviation is a negative EV play. The EV of hitting is 0.1786 bets, and the EV of doubling down is 0.1252 bets. So the score assigned the player for this game is $score = EV(actualaction) - EV(baselineaction) = 0.1252 - 0.1786 = -0.0534$ bets.

Instead of starting with a fresh shoe, suppose the distribution contains: ten of each card aces through nines, and 55 ten point cards before the hand is dealt. A shoe like this is much more favourable to the player because it is rich in tens.[5] If the player chooses to double down with the same four-five against the dealer's seven, then they are making the best move. Now the equation looks like this: $score =$

[3] The author has no experience playing blackjack for profit, so these situations are crafted to show the features of the assessment tool rather than to show likely game scenarios.

[4] The blackjack variant used for the calculation of these values has the following rules: four decks, dealer hits soft 17 (soft totals are when an ace is being used to count for eleven points), full peeking (the dealer reveals when he is dealt a blackjack before the player takes any actions), double down on any two cards, no surrender (in some blackjack games, a player may surrender half of the wager and retrieve the other half and move onto the next game), double after splitting allowed, maximum of one split, no re-split aces, and no draw to split aces.

[5] Tens are good for the player for most double down scenarios, and are also good for busting the dealer.

$EV(actualaction) - EV(baselineaction) = 0.2911 - 0.2388 = 0.0523$ bets.

For the purposes of blackjack, it is worth noting that the player has control over how much they wish to bet *before* the game begins. Thus, another element of a player's skill is choosing a proper betsize when conditions are favourable. To combine scores from a series of blackjack games, the score should be calculated in the way presented and then multiplied by the amount wagered. Relating this to poker, the player must choose what stakes to play at based on how favourable they believe the game to be.

2.5 Extending to Poker

Poker is a very different game than blackjack. The two largest components of the analysis tool presented for blackjack are a baseline strategy and an expected value calculation. Both of these concepts are difficult to define for the game of poker. Depending on the opponents in a particular game, all three actions (fold, call or raise) could be likely candidates for the correct action. Often, the correct decision relies on partial observations of previous games played with these opponents. As a result, a baseline strategy that uses all available information to make the correct action appears to be poker-complete.[6] Similarly, calculating the expected value of a given decision is not straight-forward, and is largely dependent on how the players will play the rest of the game.

The reason blackjack does not have these problems is because the game is not played against another player's policies. The dealer's strategy is fixed, and the dealer's second card (which is hidden to the players) does not cause issues for the expected value calculation in blackjack. In poker, however, the opponent's hidden information affects their strategy, and the expected return for that particular game.

Nevertheless, these challenges can be overcome using heuristics to provide a useful tool. These two important components, a basic strategy and an expected value calculation, will be explained in detail in the coming chapter.

[6] This means that properly taking into account all previous knowledge about a player to make the best decision is at least as hard as anything else in poker.

Chapter 3

DIVAT

The DIVAT tool consists of two important components. The first is a baseline strategy to compare the player's actual decisions to. This baseline acts as a reasonable strategy to expect both players to play. The second component is an expected value calculation used to estimate the difference between the player's actual decisions and the actions recommended by the baseline strategy. After these components are defined, the DIVAT tool is formally presented.

3.1 The DIVAT Baseline

Texas Hold'em is a complex game. Standard strategies for the game involve context sensitive tactics such as ***bluffs***, ***check-raising***, ***slowplaying*** to make it harder to discern the private information the player holds. At its core, however, poker is about ***betting for value***. In general, a strong hand should be bet to win money. Conversely, a weak hand should not be bet because more often than not the player will lose money.

Playing a strictly bet-for-value strategy is a terrible policy in principle. If a player notices that your bets and raises are only made when you have a strong hand, they will know how to react to your bets. For instance, if you bet they will know to fold more often with stronger hands than they would if they did not know your strategy. Conversely, if you don't bet they will know you do not have a strong hand and can bet and raise with more hands than they would normally. In other words, by being predictable, you can be exploited. However, against unobservant opponents, a pure bet-for-value strategy can be good. It also forms the core of most good poker player's strategy. For this reason, a reasonable baseline strategy is a simple bet-for-value one.

The primary purpose of the DIVAT baseline is a way of judging how much money each player should be willing to invest with their hand when not knowing the opponent's cards. The players will then be compared to this baseline to determine which player played better than the other.

3.1.1 Hand Rank: IHR, 7cHR, and EHR

The tool depends on several metrics which will be defined as they are needed. The first of these metrics is ***hand rank***. Immediate Hand Rank (IHR) is the simplest of several metrics that can be used to determine how strong a hand is against a general opponent. IHR represents the relative strength of a hand at the current moment of time, on a scale from zero to one. The intuition is that the likelihood that the hand wins is relative to what the opponent can hold. An enumeration over possible hands the opponent can hold is taken and the number of times the player's hand is best, tied, or beaten by the opponent are counted. The formula for IHR is:

$$IHR = (ahead + tied/2) \;/\; (ahead + tied + behind). \qquad (3.1)$$

This calculation is fast and can be done in real time since there are only 1,225 possible two-card combinations for the opponent before the flop, 1,081 on the flop, 1,035 on the turn and 990 on the river.

For example, suppose a player holds **A♣-K♣** and the flop is **K♢-9♠-2♡**. The cases where the player is behind the opponent are KK, 99, 22, K9, K2, 92, and AA. The opponent can hold KK one way, 99 three ways, 22 three ways, K9 six ways, K2 six ways, 92 nine ways, and AA three ways. The total number of ways the opponent is ahead is 28. The opponent can also tie by holding AK, which can be held six ways. That leaves 1,044 ways the player is ahead of his opponent. Therefore: IHR $= (1,044 + 6/2)/1,081 = 0.968548$.

By extension, the 7-card Hand Rank (7cHR) metric enumerates the remaining board cards before enumerating the remaining possible opponent holdings, and computes the average outcome. The calculation after the board is rolled out is identical to the IHR, with counts maintained for wins, ties, and losses. On the river, 7cHR is equivalent to the IHR since there are no board cards remaining to be dealt. In the same example where the player holds **A♣-K♣**, the 7cHR is 0.884692. There are several possible ways for the opponent's hand to improve to be ahead when future cards are considered, which is why this number is somewhat lower than the IHR. The presence of ***runner-runner*** flush or straight draws plus the multitude of ways that the opponent can make two pair without the player improving makes the 7cHR lower than the IHR.

7cHR is more computationally intensive to calculate on the fly, so many of these values were precalculated and stored in large tables for use by DIVAT. The tables used by DIVAT were computed for all pre-flop, flop, and turn scenarios. They use suit isomorphisms to reduce storage requirements and speed up table generation.

Note that both 7cHR and IHR assume a uniform distribution of opponent hands. Of course, it is possible for a player to have a belief about the opponent's holdings which would skew the distribution. For example, a simple assumption would be that the opponent holds stronger hands after the flop betting round (because the opponent will have folded his weakest hands). This better informed metric is called Immediate Hand Strength (IHS) and 7-card Hand Strength (7cHS). For the purposes of simplicity this alternative is ignored, but the idea may be of interest for future versions of DIVAT.

7cHR is essentially an average over all 2-card future outcomes. Two of the major issues with the metric are that it tends to undervalue hands that are strong at the moment but susceptible to opponent draws, as well as overvalue hands that are weak but can improve to a slightly better hand. The effect is more pronounced for the latter of the two cases. Consider the example, **P1: 3♣-2♣ Board: K♠-T♡-7♢**, with P1 having a 7cHR = 0.1507. In this case, the player has the lowest possible hand at the current moment with no possible flush or straight potential, but has a 7cHR much higher than zero because of the possibility of making a pair by the river. However, the opponent may very well hold a pair already (the 32 is said to be ***drawing dead***, and the opponent may force P1 to fold before the river if he doesn't make his pair on the turn).

To overcome this significant shortcoming of 7cHR, Effective Hand Rank (EHR) is used instead. To counter the underestimate effect of 7cHR, the maximum of the IHR and 7cHR is used if either can be described as a betable hand (see the following section). This allows hands that are currently very strong to be raised and reraised appropriately to the betting cap. For instance, a hand such as **A♠-A♢** is the strongest hand that can be held preflop, but only has a 7cHR of 0.8520 (the probability of winning against a random hand). Nevertheless it should be willing to raise to the maximum.

If the hand is not betable on the flop, the EHR is defined as EHR = (IHR + 7cHR)/2. This counters the overestimating tendency of 7cHR, but also contains some additional knowledge of the cases where a really strong hand (such as a straight or a flush) can be made with two future community cards. In the example posed earlier where the player holds **3♣-2♣**, the player has an IHR = 0.0042 but a 7cHR = 0.1507 for an EHR = 0.0775.

To summarize, EHR is calculated as follows:

$$EHR = \begin{cases} \max\{\text{7cHR, IHR}\} & : \text{When 7cHR or IHR} > \text{M1 threshold} \\ (\text{IHR} + \text{7cHR})/2 & : \text{When} < \text{M1 and current round is the flop} \\ \text{7cHR} & : \text{All other times} \end{cases}$$

where M1 means “a betable hand”.

The EHR metric, while much improved over both 7cHR and IHR, still has several limitations. However, the EHR has several advantages, the most important one being that it is simple to compute. Most of its shortcomings can be dealt with using empirical sampling, which will be explored in more detail when the DIVAT thresholds are discussed in the upcoming section.

3.1.2 DIVAT Baseline Thresholds

A player’s baseline strategy for a particular round is defined by being in one of 6 different classes: Fold (F), Call (C), Make 1 (M1), Make 2 (M2), Make 3 (M3) and Make 4 (M4). The M1 through M4 thresholds define how many bets a player is willing to put in to the pot before simply calling to finish the betting round. For

1st P \ 2nd P	F	C	M1	M2	M3	M4
F	kK	kK	kBf	kBf	kBf	kBf
C	kK	kK	kBc	kBc	kBc	kBc
M1	bF	bC	bC	bRc	bRc	bRc
M2	bF	bC	bC	bRc	bRc	bRc
M3	bF	bC	bC	bRrC	bRrC	bRrRc
M4	bF	bC	bC	bRrC	bRrC	bRrRc

Table 3.1: DIVAT Baseline Strategies for Post-flop Betting Rounds

instance, if a player is in the M2 class, he will bet or raise until each player has contributed 2 bets, after which he will just call. Similarly, a hand in the M4 is perfectly willing to re-re-raise until the betting is ***capped*** at 4 bets.[1] By looking from each player's perspective and placing them in one of these six classes, a static baseline strategy is obtained. This baseline can be compared against the actual player actions. In Table 3.1, all post-flop[2] baseline sequences are shown for each combination of possible player classes.

For the DIVAT tool, the EHR metric is used with a set of thresholds to put each player in their class. These thresholds are theoretically derived and are described in detail in this section. Note that the baseline need not be perfect, as demonstrated by the blackjack example described previously. DIVAT is unbiased regardless of the policy that is actually selected, as will soon be shown (see Section 4.1). From this perspective, the DIVAT thresholds are not particularly important. However, the better the thresholds are, the stronger the resulting tool will be for variance reduction.

[1] Some limit games allow more than four bets (and sometimes unlimited betting) when playing heads-up (two players). Situations where players are willing to re-raise are increasingly rare as more raises are made. These cases can be handled, but for simplicity, we assume the four bet cap.

[2] The preflop baseline sequences differ slightly because of the blinds which are posted before the cards are dealt.

DIVAT Fold Thresholds

A DIVAT fold threshold is calculated based on a game-theoretic equilibrium principles. The idea is to avoid exploitation by playing a Nash-equilibrium[3] style optimal fold rate. The calculation for this is simple:

$$\text{fold rate} = (\text{betsize})/(\text{potsize} + \text{betsize}) \tag{3.2}$$

When the potsize is large, the fold rate is comparatively small. Conversely, a small pot size causes the fold rate to be a great deal higher. With more at stake, it is obviously a mistake to fold too much because any error will cost a great deal of value.

The problem then is how to map this folding policy to our hand rank metric so that hands that should be folded are easily classified. Obviously, the weakest possible holdings will be chosen for folding, since they have the smallest chance of winning. A simple mapping works well for river situations. The fold rate is calculated and used as the actual threshold for the DIVAT folding policy. The EHR metric is roughly uniform across the 0 to 1 interval, so a threshold of 0.2 represents cutting off the lowest 20% of all possible hands. For example, suppose the pot size is $50 when the river card is dealt and the bet size is $10. The DIVAT fold threshold would be $10/(10 + 50) = 0.167$ so the policy would dictate that all hands below an EHR of 0.167 would be folded in this situation (and represents folding the weakest 16.7% of all possible hands).

For pre-river rounds, however, there are a number of other issues to deal with. First, because there are cards still to come, there are many hands that should not be folded because they still have ***pot odds*** or ***implied odds*** to continue with the hand. Secondly, the distribution of hand ranks on pre-river rounds are not uniform. There is an absence of hands with very weak effective hand ranks because all hands carry with them some chance to improve with cards still to come. Conversely, very strong hands tend to lose some value when there are still future board cards. Therefore,

[3] A nash-equilibrium strategy is one that will not lose. In rock paper scissors, the nash equilibrium strategy is to play randomly which guarantees the same win rate against every opponent. A Nash-equilibirum strategy cannot be exploited due to this feature. In a similar fashion, the Nash-equilibrium fold policy should ensure that the player does not fold too often nor fold too rarely since either of these cases are exploitable.

the notion that a threshold of 0.2 represents removing the bottom 20% of hands no longer applies. Third, the EHR metric is not well-suited for balancing the effects of ***reverse implied odds***. As mentioned previously, the 7cHR and EHR metrics are overly optimistic when it comes to improving a mediocre hand's value in future rounds.

To deal with the above issues, several experiments were run to find a simple way of calculating a suitable threshold modifier to apply. It was discovered that an offset was all that was needed to apply to the flop and turn betting rounds as follows:[4]

$$\text{fold threshold} = (\text{betsize})/(\text{potsize} + \text{betsize}) + \text{offset}. \quad (3.3)$$

The actual offsets used were first estimated theoretically, and then verified and tuned empirically. The offsets found were +0.075 for the flop and +0.1 for the turn. The pre-flop was considered, but in most situations it is a mistake to fold pre-flop in heads-up Limit, since any hand has a good chance of winning if it connects with the flop. As an example, the worst heads-up limit hand is 3-2 offsuit with a 32.3% chance of winning against a random opponent hand after all five board cards have been dealt.[5] Using the game-theoretic optimal policy without an offset results in a maximum fold threshold of 0.333 which means just two hands are recommended folds: 3-2 offsuit (32.3%) and 4-3 offsuit (33.2%).

Responding to a pre-flop raise is a different situation altogether. The opponent hand distribution is normally skewed towards better hands making it correct to fold some weaker hands. Such decisions depend greatly on the opponent's style of play.

Future rounds of betting are the key to tuning the offsets. When a player faces a folding decision on the flop or turn, it is important to consider the repercussions of future opponent bets. Presumably, in many situations where the opponent bets, they

[4] Originally, the possibility of a fold threshold multiplier was considered, so the fold threshold could be expressed in the linear equation form of mx+b. Although the software is capable of accepting parameters in this way, the fold multiplier was found to be unnecessary.

[5] While it is commonly known that 7-2 offsuit is the worst Texas Hold'em hand, this is only true for multiplayer ring game play. The added chance of making a straight with 3-2 offsuit is not enough to conteract the high card power that the 7 has over the 3 when playing heads-up. 7-2 offsuit has a 34.58% chance to win over a random opponent hand. With more opponents in a ring game, the high card power is no longer the issue. To win the pot, a stronger hand than 7-high needs to be made making 3-2 the stronger hand due to its straight-making ability. Even when the 7-2 hand makes a pair, another player in a multihand situation will likely make a higher pair.

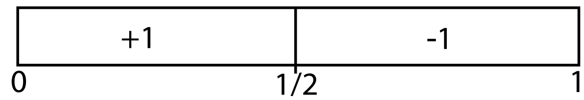
will be willing to bet again in future rounds. Therefore deciding to call profitably at the current point depends on having a favourable situation when the opponent bets in future rounds, something which is not guaranteed when using just the EHR. A simple heuristic was used to handle future round folds. There is a set of hands that are safe folds even against a player who always raises. These hands are safe to be marked as folds no matter what strategy the player is employing, since this is a minimal set of folding hands. If these hands are assumed to fold to future opponent bets, the current folding decision changes to include more hands. The fold offsets were determined using millions of simulations of different flop and turn scenarios to find the break-even point.

DIVAT Betting Thresholds

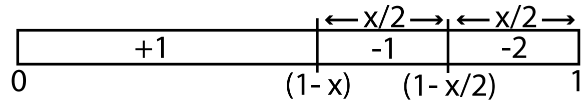
The DIVAT bet thresholds are used to determine how many bets a given hand rank is worth. It is a strict bet-for-value policy, and is based on game-theoretic equilibrium principles.

The thresholds are derived recursively based on the number of possible raises remaining in a betting round. A stronger hand is required to raise if the opponent is allowed to re-raise. However, if a re-re-raise is possible for the player, then the threshold is slightly lower again. This pattern will emerge from the derivation. In all cases, the derivation deals with the opponent's *calling* distribution. Because the DIVAT baseline is aiming for a bet-for-value policy, the player only makes money when the opponent calls with a worse hand rather than folding. For simplicity sake, we assume that the opponent's calling distribution is uniform across a range from 0 to 1 and compute the expected values based on continuous intervals.

The base case of the recursive derivation is when the player can choose to cap the betting against the opponent. This case is fairly simple since the opponent cannot re-raise. The aim then is to find the minimum bet threshold that ensures that the player will break even when the opponent calls. This point is precisely at 50% of the opponent's calling distribution. If the player bets a hand at 0.5 of the opponent's calling distribution, then half the time the opponent wins the extra bet, and half the time the player wins the extra bet for a net result of zero. This is illustrated in the following diagram of the opponent's calling distribution.



The next step in the recursion is the case where the opponent can choose to cap the betting if the player re-raises. The goal is to find the break-even point if the player bets with the top $x\%$ of his opponent's distribution. For this purpose, suppose the player bets with a hand precisely at that threshold. If the opponent holds a calling hand which is worse than x, then the player wins an extra bet. On the other hand, if the opponent holds a hand that is better, they will either call or re-raise depending on how strong it is. From the base case, the opponent knows to re-raise with the top half of the opponent's distribution which is the top $x/2$. If the opponent just calls in this situation, the player will lose one bet and if the opponent re-raises then the player loses two bets. The following diagram shows the pertinent numbers on the opponent's calling distribution.



Solving for x, we now have:

$$0 = (1 - x)(1) + (\frac{x}{2})(-1) + (\frac{x}{2})(-2) \tag{3.4}$$

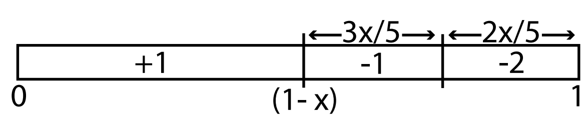
$$0 = 1 - x - x/2 - x \tag{3.5}$$

$$\frac{5}{2}x = 1 \tag{3.6}$$

$$x = \frac{2}{5}. \tag{3.7}$$

Thus the player needs to bet the top 40% of his hands to break-even.

The next step of the derivation is very similar. Again, the task is to solve for x to be the break even point for raising. This time, the opponent will re-raise with the top 40% of the player's distribution (from the previous recursive case). From there, the situation looks like this:

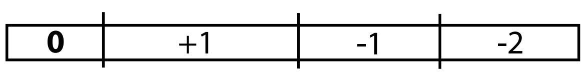


$$
\begin{aligned}
0 &= (1-x)(1) + (\frac{3}{5}x)(-1) + (\frac{2}{5}x)(-2) && (3.8)\\
0 &= 1 - x - \frac{3}{5}x - \frac{4}{5}x && (3.9)\\
\frac{12}{5}x &= 1 && (3.10)\\
x &= \frac{5}{12} && (3.11)
\end{aligned}
$$

The next recursive step results in a betting threshold of the top $x = \frac{12}{29}$. In the infinite case, this recurrance generalizes to the pattern $t(n)/t(n+1)$ where $t(n) = 1, 2, 5, 12, 29 \cdots$. An equivalent form of this sequence is $t(0) = 0, t(1) = 1, t(n) = t(n-1) + t(n-2)$ and is known as the Pell sequence.

Now that the game-theoretic equilibrium betting ratios are defined, they need to be applied to the full distribution of opponent hands including situations where the opponent may wish to fold.

In the river case, the opponent can no longer improve the strength of their hand since there are no more cards to come. There is no equity lost when a poor hand is folded. To ensure a profitable bet, the player must make a profit *when called*.[6] This is best illustrated by the following diagram presented in the same form as the threshold derivations earlier.



The leftmost portion of the opponent's hand distribution represents the profit made when the opponent folds to the player's bet. If the opponent folds, the player reaps no profit with their stronger hand. For the purposes of keeping the DIVAT betting policy as simple as possible, we will assume that the fold threshold is 1/7.[7] There are two reasons this can be done. First, the actual fold policy is the deciding factor in whether or not DIVAT recommends a fold. This means that the choice of 1/7 here is only a heuristic to make things easier. Second, unless the pot size is very small,

[6] The only situations that are being considered are betting-for-value situations. There are some scenarios where bluffing can show a profit from a combination of the chance the opponent will fold plus the chance the player will win the pot at a showdown. These situations are far outweighed by betting-for-value situations which is what the tool focuses on.

[7] The folding fraction here was chosen to make the thresholds convenient numbers to deal with for the river round. Previous derivations allowed this value to be a variable rather than a constant, but it was found to make only a small difference in the vast majority of practical cases. Thus, a constant is used for greater simplicity.

	Fold Offset	Make1	Make2	Make3	Make4
Pre-flop	0.000	0.580	0.825	0.930	0.965
Flop	0.075	0.580	0.825	0.930	0.965
Turn	0.100	0.580	0.825	0.930	0.965
River	0.000	0.640	0.850	0.940	0.970

Table 3.2: Standard DIVAT Settings

the opponent will not fold any more often than this if they are playing a game-theoretic strategy. Therefore a large number of hands are removed that will likely be folded, but keep the DIVAT metric as simple as possible. The DIVAT thresholds derived using this method are: 0.64(M1)[8], 0.85(M2), 0.94(M3) and 0.97(M4).

In pre-river rounds, if the opponent folds then the player *does* gain. In most cases with community cards still to come, the opponent will have potential to improve. If they fold, they sacrifice whatever equity they may still have in the pot. In poker terminology, bets for value in these situations also ***protect your hand***. For this reason, the betting thresholds are set over the entire opponent EHR distribution. This choice is somewhat arbitrary. However, the final choice of thresholds is not critical as will be explained later. The pre-river thresholds are: 0.58(M1), 0.825(M2), 0.93(M3), 0.965(M4).

To summarize, the standard DIVAT betting thresholds used are presented in Table 3.2. They represent a bet-for-value strategy against an equilibrium opponent. Keep in mind that there are a couple of simplifications used in the process. These simplifications are for convenience and do not affect the unbiasedness of the tool. This will be proven in the next chapter. Better thresholds might increase the variance reduction capabilities of the tool slightly, but such enhancements add complications, and are considered future work.

3.2 Expected Value

The notion of an expected value in poker is not easily defined. For a particular set of player hands, board cards, and pot size, the amount that one player will eventually

[8] 0.64 is chosen for convenience, since $1 - 0.64 = 0.36$ is divisible by 6 for non-repeating decimals.

expect to win strongly depends on the policies of the two players. These policies normally depend in turn on the past history of games between the players. For example, playing against a habitual bluffer is very different than playing against a ***tight*** and ***passive*** player. As a result, the amount a player expects to win or lose in a given situation is dependent on the opponent and is likely impossible to calculate exactly.

However, as will be seen in Section 4.1, it is possible to use a heurisitic to estimate the expected value of the situation. As long as the heuristic meets certain criteria, the resulting estimator will be unbiased from a statistical perspective.

This section describes the development of the expected value calculation used in the resulting DIVAT estimator.

3.2.1 All-in Equity

During early development of the DIVAT tool, we turned to a well-known metric to estimate how much each player will win in a particular situation. This measure is called All-in Equity (AIE). The AIE is essentially the percentage of the time that a player wins the pot after all the remaining board cards have been dealt. It is calculated by enumerating the remaining board cards and counting the number of times a player wins, loses and ties against the opponent's cards. Note that this is similar to the 7cHR calculation, except that the opponent's cards are known so the only cards requiring rollout are the remaining board cards (if any). The AIE percentage can then be multiplied by the current pot size to give the "fair-share" of the current pot that that player is "entitled to", sometimes referred to as ***pot equity***. From this point on, the term AIE will refer to pot equity.

AIE gets its name from real poker situations. If a player is *all-in* at a particular point in a game, then his AIE is an accurate view of the amount he should expect to win on average. Any further betting for players who are still in the hand and also have chips go into a side-pot and cannot be won by the player who is all-in.

AIE is a convenient metric to use because it is fast to compute. On the river, there is no rollout to compute (the winner is known).[9] On the turn, there are only 44

[9] A rollout is an enumeration of remaining states. In this case, both player cards are known so a

river cards to rollout. On the flop, there are 990 turn and river card combinations to rollout. The preflop is the most expensive calculation at 1,712,304 cases, however there are relatively few preflop situations so they are easily pre-calculated and stored in a table for future use.

The main disadvantage for the AIE metric is that it is not an accurate reflection of the true equity a player will have in a particular situation when they still have chips. The reason for this inaccuracy is because the AIE metric, like 7cHR, does not properly take implied odds and reverse implied odds into consideration. For example, suppose Alice has **2♡-3♡** on a **6♡K♡A♢9♠** board, with Bob holding **A♠J♢**. In this situation, Alice will not call a bet on the river if she does not make her flush since she has the worst possible hand. However, if she makes her flush, she stands to win the whole pot plus one or two additional big bets on the river. This characteristic of "win a big pot or lose a small one" is a much better situation than one with an equivalent AIE but without this feature. For example, consider the case where Alice holds **8♠8♣** against Bob with the same hand and same board. In this case, Alice will likely need to call another bet on the river with a mediocre hand, even though she may very well be beat already.

It turns out that AIE is not a very good estimator for the amount each player expects to win in a given scenario. It is also not sufficient to make DIVAT unbiased (Section 4.1). For these reasons, we consider another approach which counters both of these flaws.

3.2.2 DIVAT Rollout Equity

The main issue with AIE is that it does not attribute any betting after the current game situation. The amount each player can win is just a share of the current pot size. This is unrealistic for reasons pointed out in the last section. What is desired is a metric that can account for a reasonable amount wagered for the strength of the hands the two players are holding. Fortunately, we've already formulated a baseline which accounts for a reasonable betting strategy: the DIVAT baseline.

After a particular betting round, the DIVAT rollout is defined to be a complete

rollout would mean enumerating all remaining community cards.

enumeration of remaining board cards where for each set of cards the DIVAT baseline sequence is applied resulting in a larger pot size and a winner of each pot. The average amount won or lost for each player is then averaged over the enumerated set of outcomes, resulting in the DIVAT rollout equity (ROE) score. Because the DIVAT baseline provides a rational strategy that involves folding poor hands, betting strong hands, and re-raising with stronger hands, the ROE score gives a much more accurate view of the situation, including the elusive components of implied and reverse implied odds.

Algorithm 1 DIVAT rollout.

```
Require: gs is a valid game state at the conclusion of a round of betting
Ensure: gs is unchanged from entry state
  {Base case: game ends in showdown}
  if gs.round() == SHOWDOWN then
    return moneyWon
  end if
  {Base case: game ends with a player folding}
  if gs.numActivePlayers() == 1 then
    return moneyWon
  end if
  {Recursive case: average winnings over all possible chance outcomes}
  sum ← 0
  iterations ← 0
  for all chanceOutcomes do
    gs.dealCards(newCommunityCards)
    baseSequence ← gs.getBaselineSequence()
    gs.applyActionSequence(baseSequence)
    sum = sum+ DIVAT_rollout(gs)
    gs.undoRoundActions()
    gs.undoCardsDealt()
    iterations = iterations + 1
  end for
  return sum/iterations
```

The pseudo-code used for general DIVAT rollouts is shown in Algorithm 1. It is a recursive routine that ends when either the end of the river betting round has been reached (the showdown), or all players except one have folded. If there are still betting rounds left to play, all possible chance outcomes are iterated over with the DIVAT rollout of each being determined. The average of these outcomes is then returned. The code presentation relies on a gamestate object which contains pertinent information such as the current betting round, what cards each player holds, the actions each player chose, and the community cards that are dealt so far. Functions like round(), numPlayersFolded(), numPlayers(), dealCards(), applyActionSequence(), undoRoundActions(), and undoCardsDealt() either query or modify the gamestate in the appropriate way. The getBaselineSequence() function returns the DIVAT baseline actions. Using the EHR of both players hands, the baseline sequence calculation is simply a table lookup.

Unfortunately, ROE is also a lot more computationally intensive than AIE. The preflop calculation cannot be computed at runtime. Thanks to the help of Martin Zinkevich and Nolan Bard, look-up tables for the preflop were computed using dynamic programming, a large number of processors and a week of computational time.[10] These tables provide ROE entries for each pair of player cards as well as the pot size after the preflop betting round. This table completely enumerates all possible situations making it practical to run DIVAT using ROE.

ROE not only gives us a more accurate picture of the actual equity each player has in a given situation, it also makes DIVAT statistically unbiased as will soon be seen.

3.3 The DIVAT Difference

All the tools are now available that are needed to create a metric that can be used to measure the difference in skill between two players. The metric is defined in precisely the same way that was alluded to in Chapter 2. Recall that the primary goal of the DIVAT metric is to reduce the effects of stochastic luck as much as possible, revealing the elements of skill. The resulting metric takes the difference between the actions that were chosen in a particular game and the baseline action sequence that would have been recommended.

The metric is applied to each round of betting completely independent of other

[10] The tables that were computed and used for the experiments in this thesis were slightly inaccurate in that they used 7cHR instead of EHR as the metric to determine the classes for the rolled out hands. Strictly speaking, this does introduce a small bias to the metric, however we expect this bias to be miniscule given that the magnitude of errors in the preflop are very small.

rounds. For a particular past game history g, the DIVAT Difference scores are calculated for each round played, each using the same formula:

$$\text{DIVAT Difference} = ROE(\text{Actual Actions}) - ROE(\text{Baseline Actions}) \quad (3.12)$$

The resulting DIVAT Difference scores are summed, and the result is an unbiased estimator for the actual outcome of the match. It has much lower variance and therefore is more useful for post-game analysis.

Algorithm 2 DIVAT calculation for one round.

Require: gs is a valid gamestate at the start of a betting round
Require: $actions$ is a valid action sequence representing what actually happened
Ensure: gs is unchanged from entry state

$baseActions \leftarrow gs$.getBaselineSequence()
gs.applyActionSequence($baseActions$)
$baselineEV \leftarrow gs$.DIVAT_rollout()
gs.undoRoundActions()
gs.applyActionSequence($actions$)
$actionEV \leftarrow gs$.DIVAT_rollout()
gs.undoRoundActions()
return $actionEV - baselineEV$

The pseudo-code for how the calculation is executed is shown in Algorithm 2. The functions used in this pseudo-code section are as discussed in the previous sections.

One of the nice things about this metric is that the baseline sequence is fairly easy to understand on an intuitive level because it is a bet-for-value strategy. If the players play in such a way as to contribute the same number of bets to the pot that the baseline sequence does, then neither player is penalized or rewarded for their play. However, when one or both players deviate from the baseline and produce a different sized pot than the baseline sequence, the magnitude of the difference in equity will be rewarded to the better player. These deviations can represent mistakes or insightful plays, and it is the job of the ROE to determine how much the player should lose or gain. Insightful plays might include successfully check-raising, a successful bluff, or a timely fold. Mistakes might include incorrect folds, unsuccessful bluffs, or continuing with a hand that should be folded. Note that all of these examples are not bet-for-value actions, which means they will appear as deviations in our analysis.

Another nice property of the DIVAT Difference analysis is that it is a round-by-round analysis technique. It is possible to keep these values separate and analyze trends for individual rounds of play. As we will see later, there are some surprising insights that can be discovered due to round-by-round analysis. However, this analysis must be tempered with some caution because each round is not independent of previous rounds. It is common for a player to intentionally make a misleading play in an earlier round of the game to exploit mistakes from the opponent in later rounds (particularly when the bet size doubles from the flop betting round to the turn). In addition, later betting rounds like the turn and the river are reached less often than the preflop and flop because of player folds earlier in the game.

In addition to the DIVAT Difference metric, and the obvious division of looking at each round individually, there are a number of other metrics we can examine. Though DIVAT Difference is the most useful metric, these others fall out naturally from the DIVAT computations almost for free. Some metrics we will discuss are: Dcash, Money minus Dcash, and red green points.

Dcash, or DIVAT cash line, is a metric that ignores the actions the players chose and simply computes how much money each player would have won had they played the baseline sequence. You could say it represents the amount two sane players would win or lose if they were dealt the particular cards that occurred in the match. There is one small caveat which causes some issues with this metric. If a player folds partway through a hand, the remaining cards are not revealed so we do not know what the remaining DIVAT baselines would be. The policy we use is to compute the DIVAT ROE calculation and use that as the average amount each player wins. However, this is not an unbiased situation since the instances when players choose to fold are style dependent, and influence the times when the rollout is computed. Nevertheless, this metric is interesting to consider because it gives a notion of what "should have happened" as the cards lay (from a bet-for-value perspective).

Money minus Dcash is another possible estimator for skill. Intuitively, if Dcash represents how much players should have won based on luck, then it stands to reason that the difference of the actual money won in a game and the appropriate Dcash for that game should result in a decent estimator. One problem is that Dcash is biased, which means that Money minus Dcash is likely biased. Experiments will show that the DIVAT Difference is a better estimator in that it yields lower variance as well as being provably unbiased, so the existence of this metric is more academic than useful.

RedGreen points is another interesting metric. The idea is to assign the "credit" or the "blame" to the player who first deviated from the baseline. The metric is called RedGreen points with the idea that green points are positive deviations and red points are negative deviations. The implementation of RedGreen points was a simple first approximation in that it only considers the first player to deviate and assigns the credit or blame accordingly. A better metric would take into account the possibility that both players may deviate from the baseline. The best way to handle this from a credit/blame point of view is unclear, which is why this is left for future work.

DIVAT relies on knowledge of all hole cards to calculate a relatively accurate EV for a particular hand. However, this constraint means that DIVAT cannot be used in this form to evaluate the performance of a player *during* a match. One possible way to help reduce the variance of an in-game analyzer is to use DIVAT for all hands where the opponent reveals their hole cards, and use the bankroll for all others. This method was used to provide some feedback for the winning poker bot at the American Advances in Artificial Intelligence conference (AAAI) 2006 in Boston. The University of Alberta bot, Hyperborean, was a hybrid program utilizing two different pseudo-optimal poker solutions. A "Coach" program was used to select between the bots based on performance, with a runtime version of DIVAT providing lower variance feedback than the bankroll metric.

A runtime DIVAT version of performance analysis has not been proven to be unbiased, and it very well may have a small bias. However, the unbiased distinction is less important here for a number of reasons. Because the tool is part of the program, the bias only affects the bot's performance rather than making incorrect con-

Hole Cards		Alice T♠-6♠	Bob J♢-5♢		
Round		Preflop	Flop	Turn	River
Board Cards		<none>	T♢-5♡-T♡	2♢	2♡
IHR:	Alice	0.3857	0.9672	0.9599	0.9980
	Bob	0.4771	0.8483	0.8406	0.7091
7cHR:	Alice	0.4894	0.9273	0.9378	0.9980
	Bob	0.4999	0.6767	0.7827	0.7091
EHR:	Alice	0.4376	0.9672	0.9599	0.9980
	Bob	0.4884	0.8483	0.8406	0.7091
Actual Actions		SlCk	kRrC	bRrC	bC
DIVAT Baseline		SlCk	bRrC	bRrC	bC
DIVAT AIE analysis for Player 1 (Alice):					
Actual Equity		-0.1487	+2.7576	+5.7273	+11.0000
Baseline Equity		-0.1487	+3.6768	+5.7273	+11.0000
DIVAT Difference		0.0000	-0.9192	0.0000	0.0000

Table 3.3: DIVAT AIE analysis for the Alice versus Bob half of the example game.

clusions about a match's result. Since bankroll observations are usually too noisy to gain any useful information, the reduction in variance is likely far more important than any minor bias. Even if a bias exists, gaining useful information hidden by all the stochastic noise might be almost impossible for the opponent. Finally, runtime DIVAT was used as a way of selecting between two perfectly reasonable programs. An error in choice would only result in a small EV penalty in the worst case.

3.4 Revisiting the Example Game

In the introduction, an example duplicate game was used to motivate DIVAT. In this section, the example is revisited to show the actual DIVAT analysis. The calculations for the first half of the duplicate match are shown in Table 3.3. AIE was used for the equity calculations shown in the tables. While AIE is not statistically unbiased, it is easier to calculate and verify by hand, and thus is better for illustrative purposes.

In the pre-flop betting round Bob calls in the small blind and Alice checks.

Hole Cards		Bob T♠-6♠	Alice J♢-5♢		
Round		Preflop	Flop	Turn	River
Board Cards		<none>	T♢-5♡-T♡	2♢	2♡
IHR:	Bob	0.3857	0.9672	0.9599	0.9980
	Alice	0.4771	0.8483	0.8406	0.7091
7cHR:	Bob	0.4894	0.9273	0.9378	0.9980
	Alice	0.4999	0.6767	0.7827	0.7091
EHR:	Bob	0.4376	0.9672	0.9599	0.9980
	Alice	0.4884	0.8483	0.8406	0.7091
Actual Actions		SlCk	kBc	kBrC	bF
DIVAT Baseline		SlCk	bRrC	bRrC	bC
DIVAT AIE analysis for Player 1 (Bob):					
Actual Equity		-0.1487	+1.8384	+3.8182	+6.0000
Baseline Equity		-0.1487	+3.6768	+5.0909	+8.0000
DIVAT Difference		0.0000	-1.8384	-1.2727	-2.0000

Table 3.4: DIVAT AIE analysis for the Bob versus Alice half of the example game.

These actions are exactly the same as the DIVAT baseline, so the DIVAT Difference is zero, meaning that neither player gains or loses.

Both players are helped by the flop, but Alice has the stronger hand. Alice's check turns out to be a slight mistake since Bob has a strong enough hand to raise. So instead of the baseline actions of bet-Raise-reraise-Call (bRrC), the actual actions played are check-Bet-raise-Call (kBrC). This means that less money went into the pot, and therefore Alice gets penalized almost a full small bet.

On the turn, Alice now knows Bob has some strength so she bets out this time and the players follow the baseline sequence. Like the preflop, neither player gains or loses DIVAT score.

Finally, on the river, both players again follow the DIVAT baseline with Bob unable to lay down his hand even though he suspects he may be beat. Again, neither player gains or loses DIVAT score since the bet-call actions are the same as the baseline. In total, Alice loses a small amount of DIVAT score on the flop totalling -0.9192 small bets worth of equity.

The analysis for the reverse half of the duplicate match is shown in Figure 3.4.

Both players again execute the same actions as the baseline in the pre-flop betting round. In each of the remaining rounds, however, the players stop following the baseline.

Bob's slowplay on the flop gives him a DIVAT penalty twice the size that Alice received in the same situation. In general, slowplaying for a particular round will cost the player an immediate penalty. A player executing a slowplay hopes that their play will pay greater dividends in later betting rounds. In this case, Bob's slowplay did allow him to check-raise Alice on the turn. However, Alice's hand was stronger than Bob realized, so he was not able to capitalize on this opportunity as much as he perhaps could have. Not only does he receive another DIVAT penalty for not getting as much in the pot as the DIVAT baseline, he has also revealed the strength of his hand. When Bob slowplays on the flop and then check-raises the turn, Alice can be fairly sure that Bob has a very strong hand.

On the river, Alice is now convinced that Bob has the best hand. The **2**♡ completes a possible flush draw semi-bluff, and a possible (though unlikely) check-raise with a deuce, thereby eliminating any remaining doubt. Alice's fold gains her a DIVAT bonus, since her hand was certainly strong enough to call given the large pot size. In the end, Alice gains a total of 5.1111 small bets in DIVAT Difference due to Bob's mistakes and her correct fold on the river.

For the duplicate match then, DIVAT assigns Alice a total of 4.1919 small bets for this pair of hands. If Alice continues to outplay Bob in this fashion, she will win the match with ease. However, this is merely a one-game instance. Though it appears that Bob may have misplayed this particular game, he might recover when confronted with different and more common game situations.

Chapter 4

Theoretical and Empirical Results

4.1 Unbiased Estimator

DIVAT began as an intuitive attempt at removing the element of luck as much as possible from the analysis of a poker game. What was left was the elements of skill that we were truly interested in. After the initial implementation with the simple but incorrect AIE DIVAT, other members of the poker group were interested in what the properties of the tool were. Not long afterwards, Martin Zinkevich proved that ROE DIVAT was unbiased from a statistical perspective[1] [27]. The proof presented in the AAAI paper [27] is a general version of the proof. In this section, we will present a more domain-specific version of the proof with the aim of building an intuitive understanding for why the estimator is unbiased.

First, a number of conventions used throughout the proof are defined. A game's history will be written as a number of chance-action pairs.

$$h_i = c_1 a_1 c_2 a_2 \cdots c_i a_i. \tag{4.1}$$

We also define a *value function* which maps histories to real numbers: $V : H \to \mathbb{R}$. For purposes of this proof, V is the DIVAT rollout equity calculation. In this notation the DIVAT Difference can be expressed in the following way:

$$\text{DIVAT Diff} = \sum_{i=1}^{4} V(h_i c_i a_i) - V(h_i c_i). \tag{4.2}$$

[1] In this context, we take unbiased to mean that the DIVAT metric measures the same expected value as the bankroll measure.

Proof that the DIVAT Difference metric is unbiased: First, let us consider the following sum:

$$\underbrace{\begin{matrix} V(c_1) \\ -V(\phi) \end{matrix}}_{luck} + \underbrace{\begin{matrix} V(c_1a_1) \\ -V(c_1) \end{matrix}}_{skill} + \underbrace{\begin{matrix} V(c_1a_1c_2) \\ -V(c_1a_1) \end{matrix}}_{luck} + \cdots + \underbrace{\begin{matrix} V(c_1a_1c_2a_2c_3a_3c_4a_4) \\ -V(c_1a_1c_2a_2c_3a_3c_4) \end{matrix}}_{skill} \quad (4.3)$$

This sum represents the changes in the value function after every chance event and every action (betting) round. Colloquially, we can label each column of the equation as *luck* or *skill*, depending on whether the node in question is a chance node or an action node. For example, the column $V(c_1) - V(\phi)$ is the difference between the value function before and after the initial cards are dealt. This is labeled a luck node because the chance node is beyond the player's control and is part of the stochastic part of the game. Conversely, skill nodes consist of the difference in the value function before and after an action node has occurred. Here, the players have control of their actions. Equation 4.3 can intuitively be thought of as $skill + luck$.

Notice that when line 4.3 is summed, most of the terms cancel out leaving just two:

$$V(c_1a_1c_2a_2c_3a_3c_4a_4) - V(\phi) \quad (4.4)$$

$V(c_1a_1c_2a_2c_3a_3c_4a_4)$ is the value function at the end of the hand. This value is equal to the net amount of money won in the game. $V(\phi)$ is a constant for the value each player has with no cards dealt. In the poker domain this represents the advantage of position that the second player has over the first player. Therefore, we can intuitively think of the resulting equation as:

$$luck + skill = money - position \quad (4.5)$$

$$\text{or } money = luck + skill + position \quad (4.6)$$

Furthermore, we are dealing with mathematical expectation rather than the results of any one game, so we would like to see how these values relate in terms of expectation. Fortunately, expectation is easy to introduce due to linearity of expectation:

$$E(money) = E(skill) + E(position) + E(luck) \quad (4.7)$$

In standard heads-up Limit Texas Hold'em, the positional advantage is swapped every game. Therefore, every two games the advantage of position cancels out and is zero. Even in a match with an odd number of games, the effect of the one player having a slight positional advantage diminishes after a few hands. Therefore, $E(position) \approx 0$.

If the game is fair, we should expect that $E(luck) = 0$.

We have established that

$$\begin{aligned} luck = \; & (V(c_1) - V(\phi)) + (V(c_1 a_1 c_2) - V(c_1 a_1)) + \cdots + \\ & (V(c_1 a_1 c_2 a_2 c_3 a_3 c_4) - V(c_1 a_1 c_2 a_2 c_3 a_3)) \end{aligned}$$

In general, there are a number of terms in this sum equal to the number of chance nodes in the game. Consider any such term $V(h_i c_{i+1}) - V(h_i)$. In expectation:

$$E(V(h_i c_{i+1}) - V(h_i)) \tag{4.8}$$

$$= E(V(h_i c_{i+1})) - E(V(h_i)) \tag{4.9}$$

$$= E(V(h_i c_{i+1})) - E(E(V(h_i c_{i+1})|h_i)) \quad \text{property of poker} \tag{4.10}$$

$$= E(V(h_i c_{i+1})) - E(V(h_i c_{i+1})) \quad \text{law of iterated expectations} \tag{4.11}$$

$$= 0 \tag{4.12}$$

The crux of the argument is line 4.10 where the claim that $V(h_i) = E(V(h_i c_{i+1})|h_i)$ is made. The claim means that the value function is **consistent** across each chance node. In other words, the value function correctly returns the average of all value function evaluations for each possible chance node. ROE DIVAT has this property. In considering all future outcomes, each situation is assessed in exactly the same manner as it would be if it actually arose. For example, there is no difference in the value of a river situation whether it currently exists, or is only one of 990 possible futures from a flop situation. This is what is meant by **consistency**. The law of iterated expectations is a well known law of statistics [25]. Therefore the $luck$ term has an expectation of zero, as expected.

Returning to equation 4.7:

$$E(money) = E(skill) + 0 + 0 \tag{4.13}$$

$$E(money) = E(skill) \tag{4.14}$$

$$E(money) = \sum_{i=1}^{4} V(h_i c_i a_i) - V(h_i c_i) \tag{4.15}$$

Money is an existing unbiased estimator for the game, and since ROE DIVAT is equal to money in expectation, ROE DIVAT must also be unbiased. □

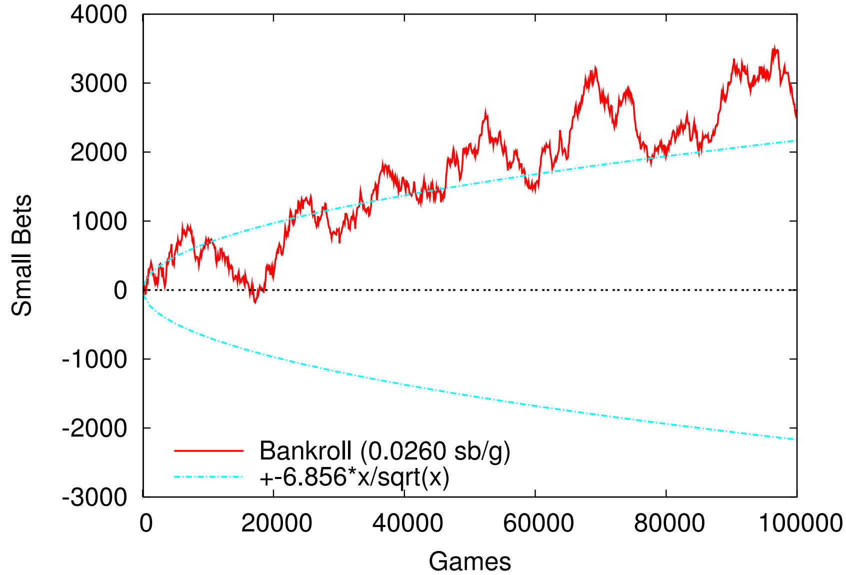
4.2 Empirical Results

The proof that the DIVAT Difference is unbiased is very nice from a theoretical perspective. However, the transition from theory to practice is often not a smooth path. Therefore, there is a great deal of value in testing the tool from an empirical standpoint to verify that the tool works as advertised. In this section we present the results of experiments designed to show empirically that the tool is unbiased.

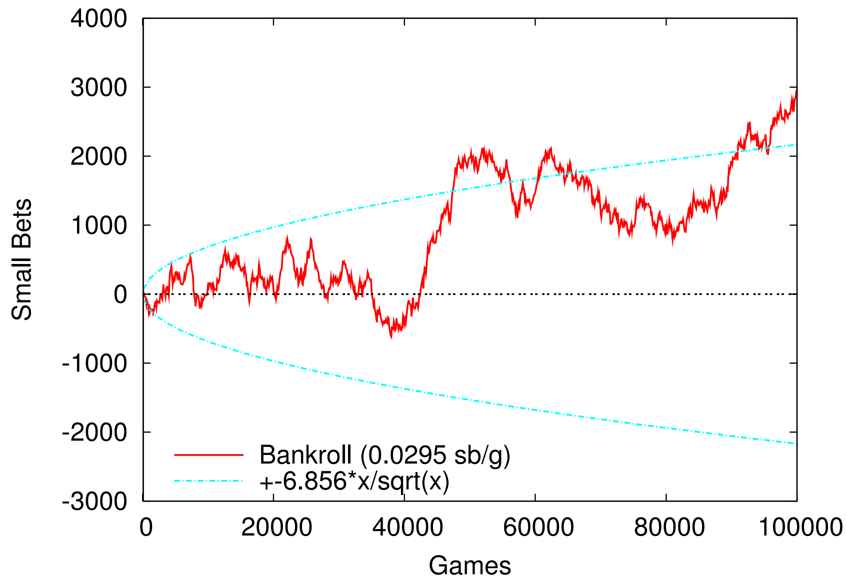
4.2.1 Always-Call versus Always-Raise

The first match examined is the one discussed in the introduction: Always-Call versus Always-Raise. While this match is not realistic for many reasons, it has some useful properties. In particular, neither player exhibits superior skill, so the long-term expected result of the match is that neither player wins money from the other. The match closely resembles coin-flipping, with the exception that the players are sometimes allowed to tie. All games result in a pot of seven small bets, which is either won by a single player or split in the event of a tie. 25 million games were simulated achieving an empirically calculated variance of 47.010, making the standard deviation $\pm$ 6.856 small bets per game (sb/g).[2]

[2] This variance tends to be higher than a typical match. Since players can fold on any round, there are many games which end before the river round. However, those games are countered by some where several bets go into the pot on each round resulting in a very large pot size. These large outcomes carry a disproportionately heavy quadratic weight towards overall variance. A baseline standard deviation of about 6 sb/g is fairly normal, though a lower standard deviation is certainly possible.



(a)



(b)

Figure 4.1: Two Always-Call versus Always-Raise matches.

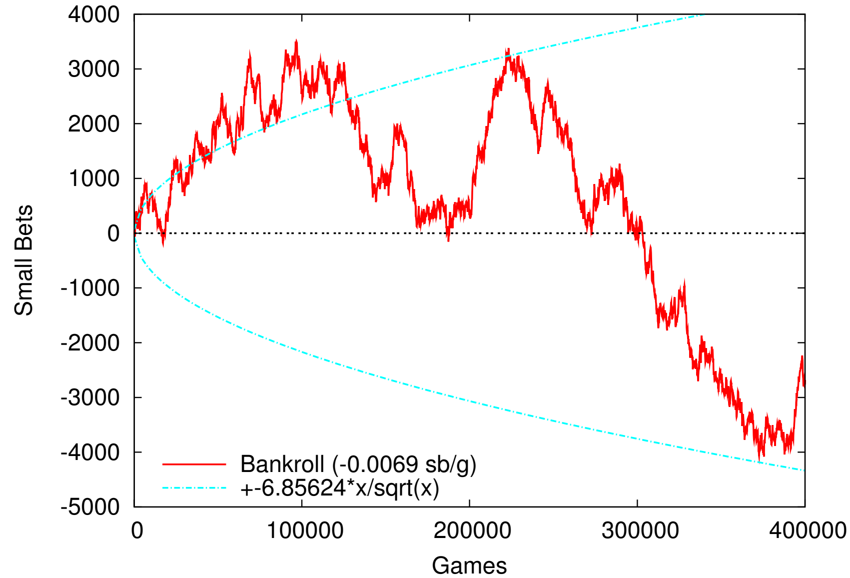


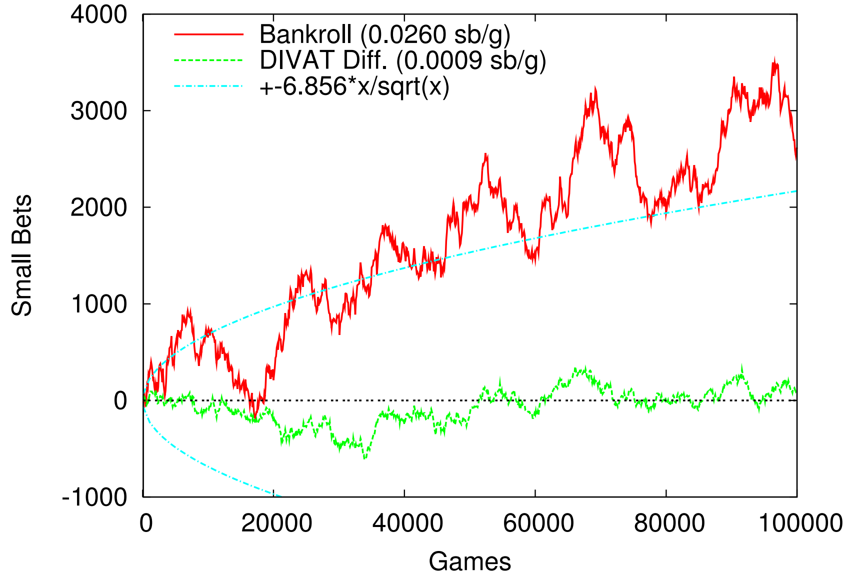
Figure 4.2: A longer Always-Call versus Always-Raise match.

Figure 4.1 shows two Always-Call versus Always-Raise matches that clearly show how difficult a problem variance presents for analyzing matches. If one were to examine the matches presented in Figure 4.1 without knowing the properties of the players, one might easily conclude that Always-Call was exhibiting superior skill. In reality, both matches show Always-Call winning at a rate just outside the one standard deviation curve, meaning that we only have about 70% confidendence that Always-Call beats Always-Raise from a statistical standpoint. Conversely, there is about a 30% chance that that conclusion is faulty.[3]

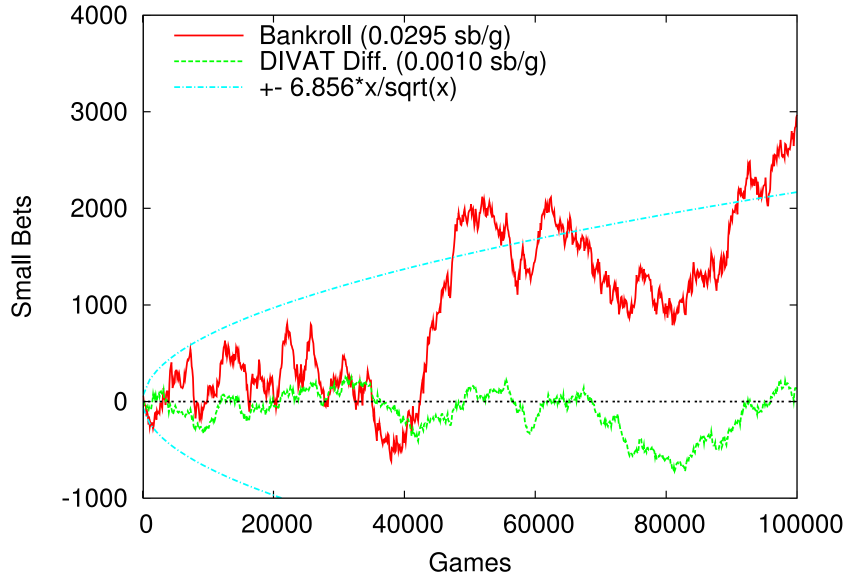
In Figure 4.2, the match in Figure 4.1(a) is extended to 400,000 games. Over this long match, the bankroll varies a great deal between the one standard deviation curves. The match is a long 400,000 games, and it is still hard to tell that the two players are of equal skill (though the net overall win value of -0.0069 is suggestive).

The DIVAT Difference metric tells a much different story. Figure 4.3 shows the two 100,000 length matches from Figure 4.1 when analyzed with DIVAT. The DIVAT Difference hugs the zero line showing that it believes there is little difference in skill between the two players. Notice that the DIVAT line is not completely seperate

[3] These matches were not selected from a variety of runs. They were the first two experiments executed (with different random number generators due to our own suspicions).



(a)



(b)

Figure 4.3: DIVAT analysis of the Always-Call versus Always-Raise matches.

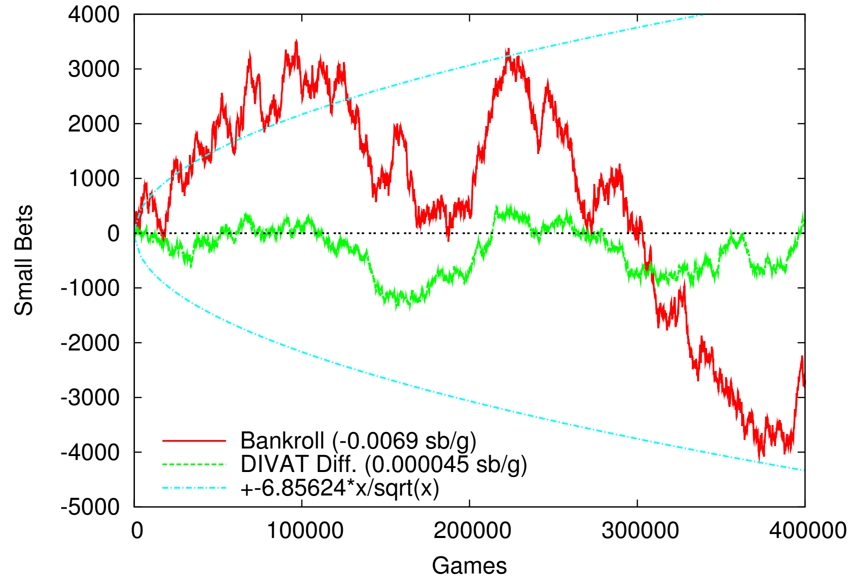


Figure 4.4: DIVAT analysis of the longer Always-Call versus Always-Raise match.

from the trends that the money line shows. For instance, in Figure 4.3(b), the dip in the money line around hand 80,000 is also evident in the DIVAT Difference line. This reinforces the idea that while the DIVAT Difference does a very good job of removing the 'luck' of the game, it cannot account for all circumstances that might arise (opportunities are based on luck too, so there is still some correlation). The standard deviation for the DIVAT Difference in the two matches is 2.934 and 2.917 respectively. The variance is the standard deviation squared, so the variance reduction factor is calculated by dividing the squares of the standard deviations. The variance reduction in this case is a factor of 5.46 and 5.52 respectively, which is a large improvement. The large reduction in variance means that many fewer hands are required to attain the same statistically significant result.

In Figure 4.4, the DIVAT analysis of the extended 400,000 hand match is shown. Again, the DIVAT Difference line is much closer to the zero line than the bankroll. The standard deviation for the DIVAT Difference is 2.913; a factor of 5.54 reduction in variance.

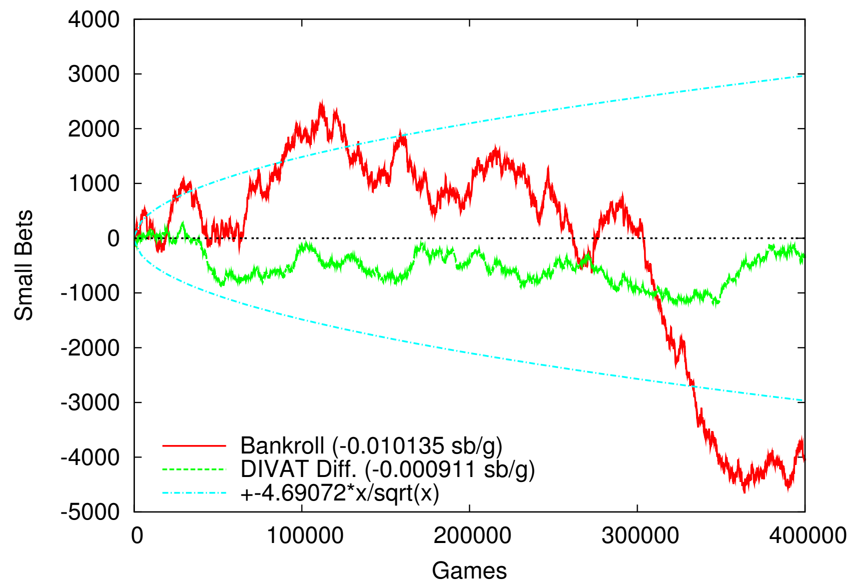


Figure 4.5: DIVAT analysis of a PsOpti4 selfplay match.

4.2.2 Selfplay Match

The next obvious step in our empirical testing of the DIVAT tool is a selfplay match for a bot that actually attempts to play good poker. For this, we turn to a pseudo-optimal bot which the University of Alberta has had for a number of years: PsOpti4 [4]. PsOpti4 plays using a static strategy and does not learn over time. Once again, the theoretical result of this match is zero since both players are exactly the same and do not change. The graph with both the bankroll and the DIVAT analysis is presented in Figure 4.5. For consistency, this match uses the same series of cards that were dealt in the 400,000 hand match between Always-Call and Always-Raise.

Once again, the bankroll leads us to some potentially misleading conclusions. At the end of the match, one of the Opti4 bots seems to have an advantage over the other by a win rate of 0.010 sb/g. However, one is better than the other with only 68% statistical confidence, and we can plainly see that the general trends of the curve matches those witnessed in Figure 4.4. This illustrates how the cards dealt influence the bankroll. The slope of the DIVAT line, on the other hand, is much more level. In this case, it does not align with the zero line. However, the average win rate is the main concern, and this value is very close to zero.

The bankroll in this match has a standard deviation of 4.691, which includes a fairly low variance. This is because the style of the PsOpti4 player is fairly ***tight*** so there are more folds early in the game compared to matches where the variance is a great deal higher. In comparison, the DIVAT Difference standard deviation is just 1.863 – a factor of 6.34 reduction in variance.

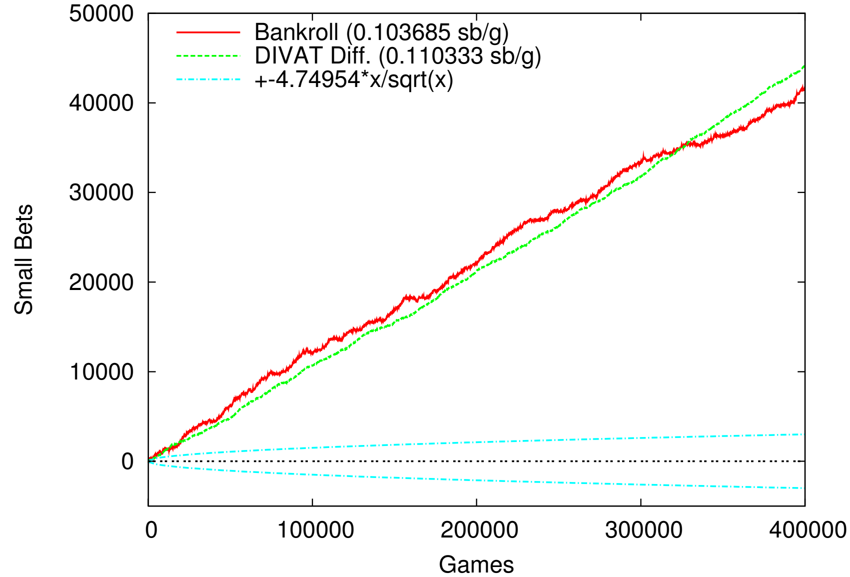
4.2.3 Two Different Static Programs

Our final empirical test of correctness for the DIVAT Difference score is a match between two different versions of the pseudo-optimal family of bots. In this experiment, the versions are PsOpti4 and PsOpti6. Presumably, one of these bots beats the other at a certain rate. Ideally, the DIVAT Difference line will match this rate sooner and with lower variance. This match can be viewed in a number of different ways. As before, the cards used in these matches were the same as the Always-Call versus Always-Raise match. Obviously one could simply run the match once. However, because they are bots it is a good idea to run the match with PsOpti4 in seat 1 and PsOpti6 in seat 2 and then reverse the seats so that both players are presented with precisely the same cards, and similar opportunities.[4] This is called a duplicate match, and is one way to reduce variance when two programs play against each other. Additionally, the variance reduction of DIVAT can be compared to running the match in duplicate.

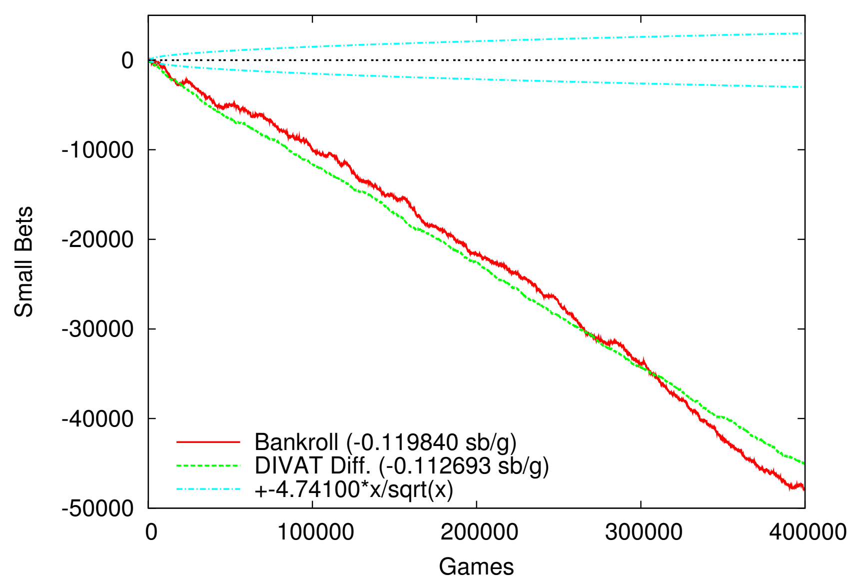
In Figure 4.6, the DIVAT Difference analysis for both sides of the duplicate match is shown. In the first match, PsOpti4 wins at a rate of about 0.10 sb/g. In the second match, PsOpti6 sits in the same seat PsOpti4 held in the first match and proceeds to lose at a rate of about 0.12 sb/g.[5] The DIVAT Difference line closely matches the money line in both graphs, at a rate of 0.11 sb/g in each case, showing empirical evidence that it is unbiased. In addition, the DIVAT Difference standard deviations are 1.929 and 1.932 in the two matches, whereas the bankroll standard deviations are 4.750 and 4.747 respectively. The overall reduction in variance is 6.05.

[4] Note that the opportunities will not be identical since the other player may not play the hand the same way. This is one source of variance for a duplicate match.

[5] The fact that PsOpti4 wins this match does not mean that PsOpti4 is the better player overall. Poker results are highly non-transitive. Scenarios where player A beats player B who beats player C who in turn beats player A are common.



(a) PsOpti4 vs PsOpti6



(b) PsOpti6 vs PsOpti4

Figure 4.6: DIVAT analysis of PsOpti4 vs PsOpti6 duplicate match.

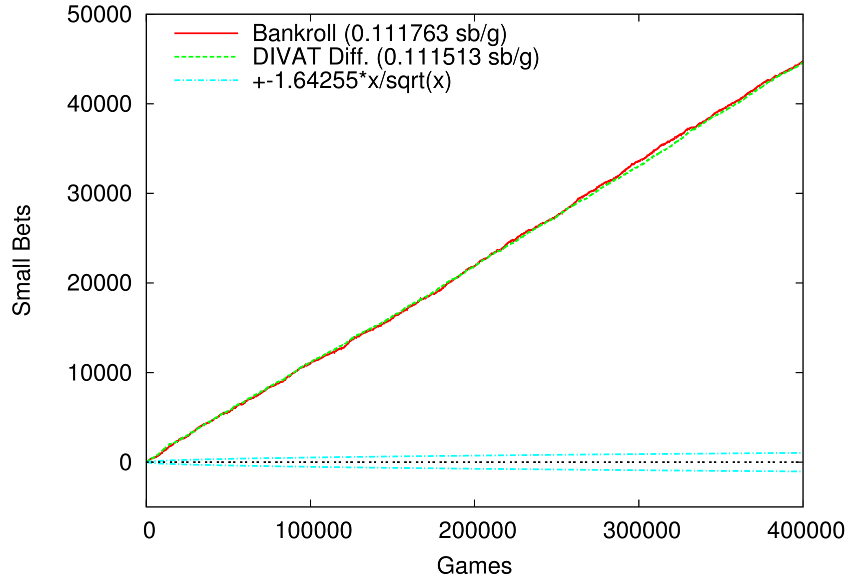


Figure 4.7: The PsOpti4 vs PsOpti6 duplicate match.

After analyzing the separate parts of the duplicate match, the two matches are combined into a single graph by averaging the results for the same games in the two matches. For instance if PsOpti4 wins 6 sb in the first game, and loses 4 sb in the corresponding hand reverse game of the duplicate match, then the average won is 1 sb/g. This graph is shown in Figure 4.7. In this graph, the average duplicate bankroll exhibits a great deal less variance than one-sided bankroll. In fact, both the money line and the DIVAT Difference line are accurate measures of long-term expected value, and are almost overlayed. For comparison, the measured standard deviation was 1.643 and 1.178 for the averaged duplicate bankroll and average duplicate DIVAT respectively.

This serves as a good example to show how DIVAT does a very good job of reducing variance by itself – almost as good as running the match entirely in duplicate. The further variance reduction for duplicate DIVAT happens when two players play a hand differently. In the duplicate version of a hand, if one player continued with the hand but the other didn't, then the duplicate bankroll has no way to counter the effect of the continuing player "getting lucky". In these cases, DIVAT is able to filter more "signal from the noise" which results in a lower variance estimator. From this experiment, the DIVAT analysis shows that it provides more

	Make1	Make2	Make3	Make4
Aggressive High	0.4600	0.7750	0.9100	0.9550
Aggressive Medium	0.4900	0.7875	0.9150	0.9575
Aggressive Low	0.5200	0.8000	0.9200	0.9600
Moderate Aggressive	0.5500	0.8125	0.9250	0.9625
Moderate (Normal)	0.5800	0.8250	0.9300	0.9650
Moderate Conservative	0.6100	0.8375	0.9350	0.9650
Conservative Low	0.6400	0.8500	0.9400	0.9700
Conservative Medium	0.6700	0.8625	0.9450	0.9725
Conservative High	0.7000	0.8750	0.9450	0.9750

Table 4.1: Range of Values for the DIVAT Betting Thresholds

insight than even matches run in duplicate. For the strongest possible estimator to this date, matches should be run in duplicate with DIVAT analysis on both sides of the duplicate match.

4.3 Robustness of DIVAT

In this section the effects of changing the DIVAT thresholds are analyzed to determine how robust the metric is. The main question is how far the DIVAT thresholds can be perturbed before vastly different analysis is obtained. The theoretical result proved earlier in this chapter guarantees that as long as the thresholds are consistent throughout the DIVAT calculations, the estimator is unbiased. To show this empirically, DIVAT is run with several different thresholds on the PsOpti4 vs PsOpti6 match discussed in the previous section.

Due to the number of experiments to be run, only the first 100,000 games of the PsOpti4 versus PsOpti6 match were analyzed with varying thresholds for the flop and turn. Both folding and betting thresholds were perturbed by a significant margin to fully test the robustness of the DIVAT metric. The betting thresholds were perturbed across a total of nine different variants, as shown in Table 4.1. The fold offsets were modified in a similar way for nine different cases, as shown in Table 4.2. Note that the changes from most aggressive to most conservative and from loosest to tightest are very large. For instance, the change from 0.46 to 0.70 for an M1 threshold represents betting almost twice as often (almost 54% compared

	Flop	Turn
Loose High	-0.025	0.000
Loose Medium	0.000	0.025
Loose Low	0.025	0.050
Moderate Loose	0.050	0.075
Normal	0.075	0.100
Moderate Tight	0.100	0.125
Tight Low	0.125	0.150
Tight Medium	0.150	0.175
Tight High	0.175	0.200

Table 4.2: Range of Values for the DIVAT Fold Offsets

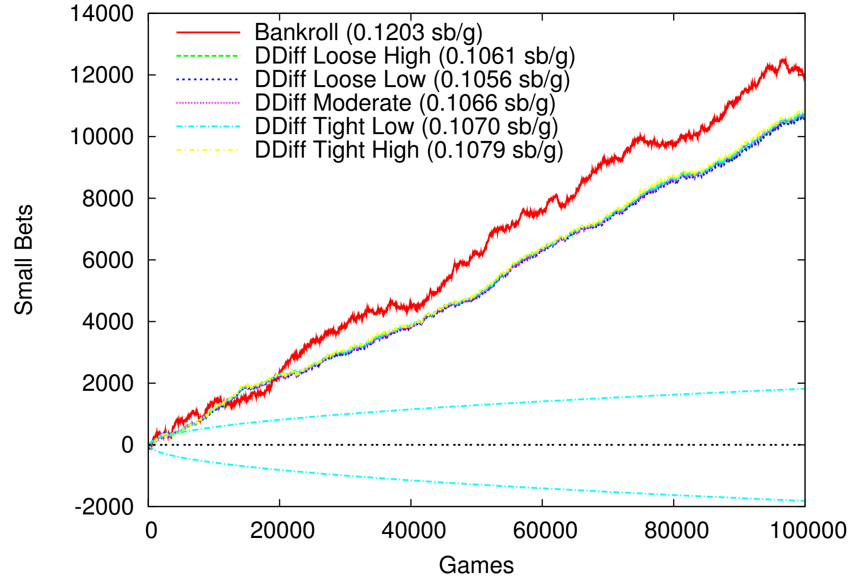
to about 30%). Therefore this should be a good test for DIVAT unbiasedness.

In Figures 4.8 and 4.9 the results of tweaking the bet thresholds and fold offsets are presented. In each case, even though the changes to the thresholds are significant, the DIVAT Difference line remains almost exactly the same. This provides another piece of empirical evidence in support of the DIVAT metric being unbiased.

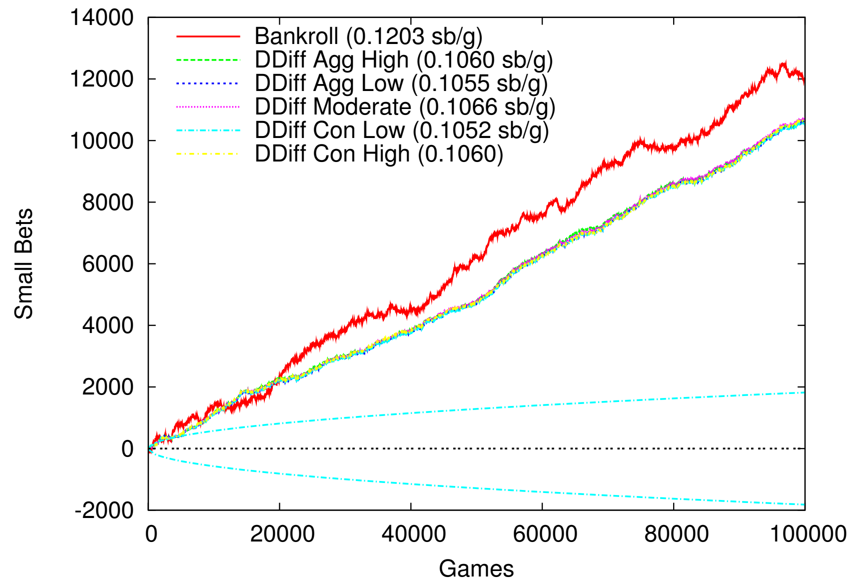
Another experiment that can be run is examining how different DIVAT thresholds affect the short-term DIVAT analysis. If DIVAT analysis of a short match does not change much even with wildly different thresholds, then the choice of thresholds is not critical for the performance of our metric.

In Figures 4.10 and 4.11, the same PsOpti4 versus PsOpti6 match is shown, pruned to the first 500 games. All 5 DIVAT lines closely coincide with each other despite the short length of the match and the vast differences in thresholds. Keep in mind that this is just changing the bet thresholds or fold offsets for one particular round. If the thresholds for all rounds were changed at the same time, the separating effect would be a little stronger. Nevertheless, this experiment gives us a great deal of confidence in the robustness of the DIVAT measure even for shorter matches.

It is possible that the metric will differ more when the style of the players contrasts more than the differences between PsOpti4 and PsOpti6. One possible way to make DIVAT a little less judgemental about particular styles of play is to run with several sets of thresholds and average the result. For example, if the baseline suggested folding for four different offsets and calling with the fifth, the average

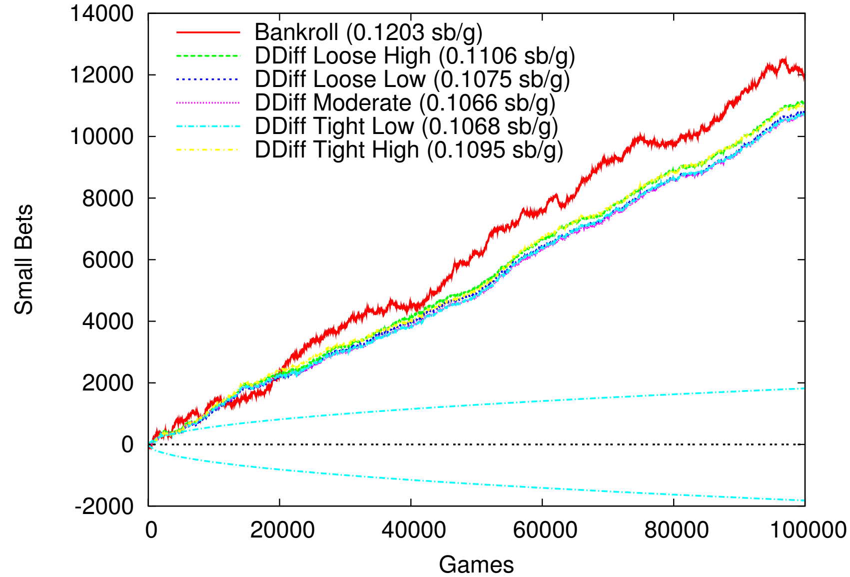


(a) Varying flop folding thresholds

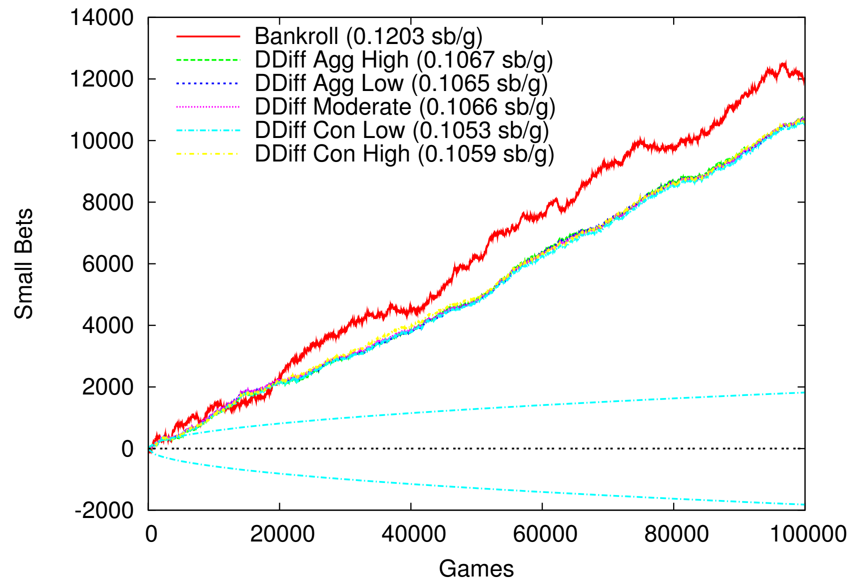


(b) Varying flop betting thresholds

Figure 4.8: Varying bet thresholds and fold offsets for the flop.

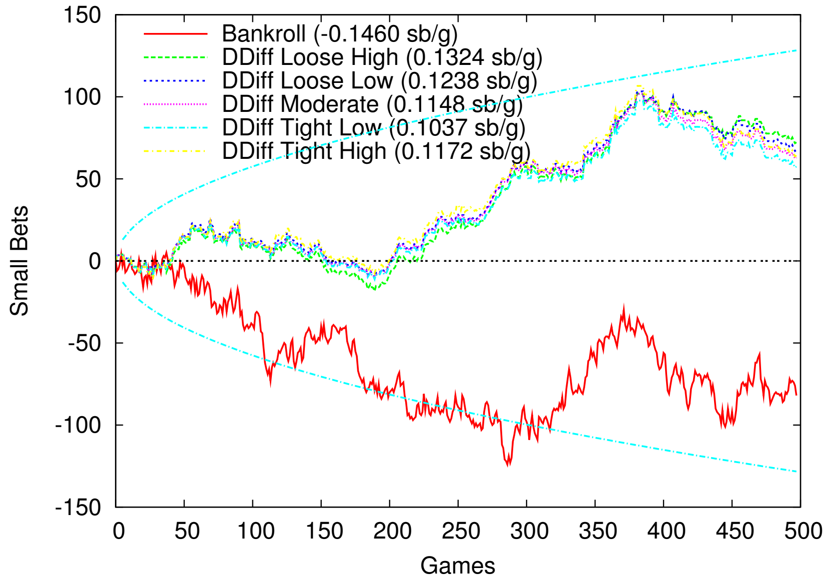


(a) Varying turn folding thresholds

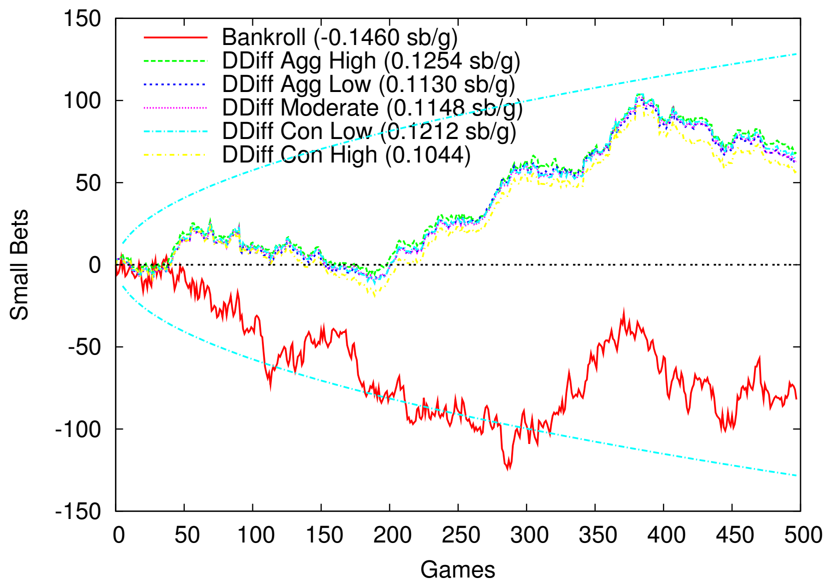


(b) Varying turn betting thresholds

Figure 4.9: Varying bet thresholds and fold offsets for the turn.

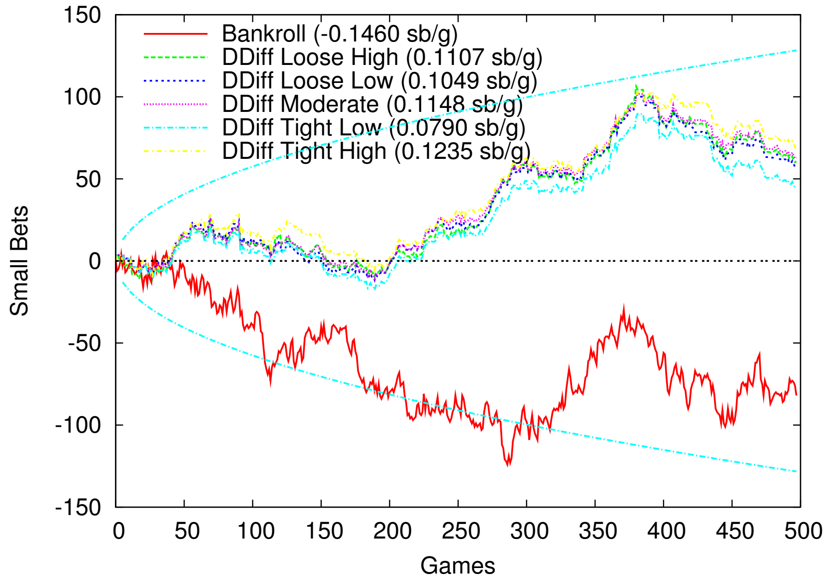


(a) Varying flop folding thresholds

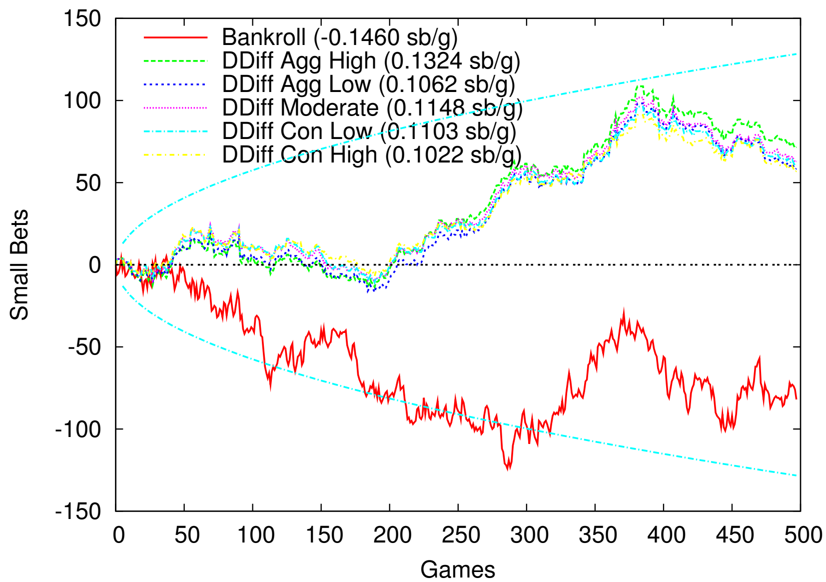


(b) Varying flop betting thresholds

Figure 4.10: Varying bet thresholds and fold offsets for the flop over a very short match.



(a) Varying turn folding thresholds



(b) Varying turn betting thresholds

Figure 4.11: Varying bet thresholds and fold offsets for the turn over a very short match.

would provide a smoothing over actions that are almost equally viable. We do not present any experiments with smoothing because these robustness experiments appear to show little need to do this smoothing. In cases where less than 500 games are played, the biggest issue is not errors in the DIVAT metric, but whether or not enough hands were played to balance opportunities for both players.

Chapter 5

Analysis and Insight

In this chapter, DIVAT analysis of several matches are presented that have either been seen before in the poker literature, or figure prominently in recent events in poker bot research and development. In particular, the match between Gautum Rao and PsOpti1 featured in the paper for IJCAI 2003, a poker competition held in Las Vegas 2005 featuring poker professional Phil Laak, and the AAAI poker competition in Boston 2006 are highlighted. In each case, the DIVAT analysis provides some interesting and sometimes surprising insights into the differences between the players.

5.1 Sparbot versus Vexbot

As a prelude to the other matches in this chapter, the first match presented is between two of the leading bots the University of Alberta poker group has produced. Sparbot is the general name for the PsOpti family of pseudo-optimal bots that approximate game-theoretic equilibrium strategy. This particular version was PsOpti4. Vexbot is an exploitive bot that is capable of finding weaknesses in its opponent [5]. Vexbot is expected to beat Sparbot over the long term, since Sparbot utilizes a static strategy with known exploits. However, Vexbot's learning is error prone and can get stuck with the wrong opponent model for many games. Analysis of this match is important because it represents a controlled experiment where one of the players is not stationary. In many of the matches presented in this chapter, one or both players may be non stationary so this is an important test. Certain

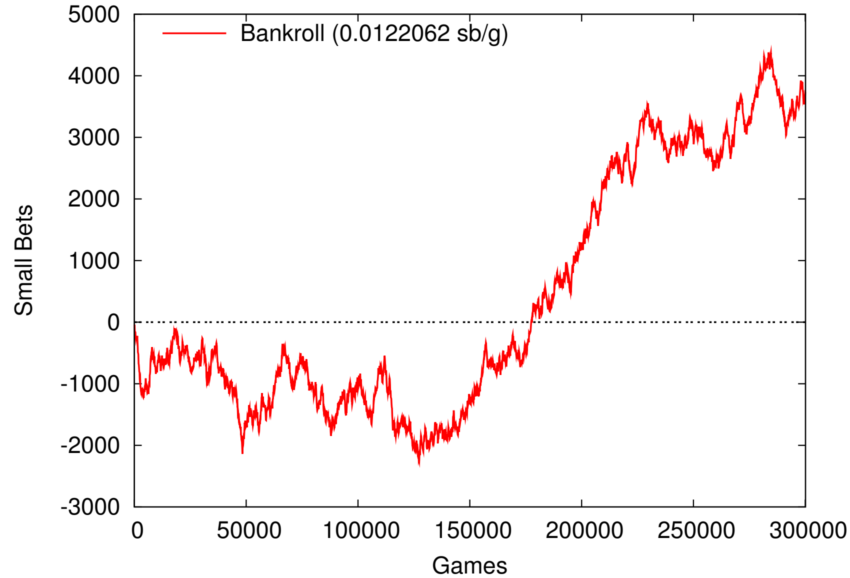


Figure 5.1: Bankroll result of Vexbot versus PsOpti4.

characteristics of these bots are well known and can be looked for in the DIVAT Difference analysis.

Note that this match was not run in duplicate. Duplicate matches are not quite as effective for non-stationary players. For this match in particular, Vexbot can learn very different models of Sparbot resulting in misleading conclusions for the match.

The bankroll graph for this match is shown in Figure 5.1. Early in the match, Vexbot flounders for awhile as it attempts to gain a model of Sparbot. Around game 125,000, Vexbot appears to have found an effective model of Sparbot and starts to exploit it at a rate of about +0.05 sb/g. Around hand 225,000 the rapid rate of exploitation appears to flatten out to the end of the match.

The DIVAT Difference analysis is plotted in Figure 5.2. According to DIVAT, it appears that Vexbot actually started gaining an advantage over Sparbot much earlier than the bankroll suggests. Perhaps bad luck is the cause of the break-even section from game 75,000 to 125,000 (where Vexbot's DIVAT Difference score starts to rise). The slower rate of exploitation near the end of the match seems to be reflected in the DIVAT Difference line (though not to the same extent as the money line). It appears that Vexbot actually "unlearned" an exploitation that it had used for roughly a 100,000 game stretch.

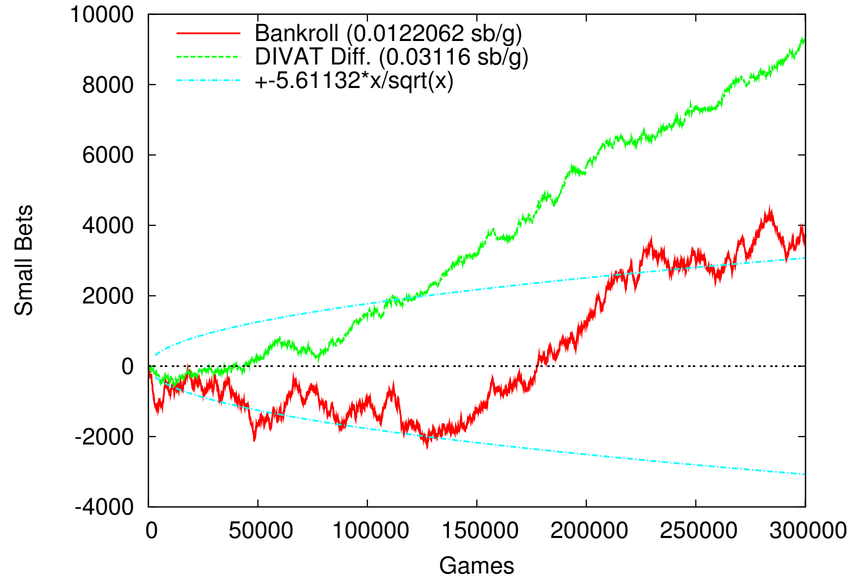


Figure 5.2: DIVAT analysis of Vexbot versus PsOpti4.

Note that despite the long match length, the DIVAT Difference does not overlap the bankroll at the end. The unbiased proof of DIVAT still works for players who are non-stationary. The issue is because Vexbot is constantly changing its strategy; once the two lines have deviated they will likely stay apart from each other unless another deviation is made to bring them back together. Instead of looking for overlapping DIVAT and bankroll lines, what is important is that they show similar *slopes*. In this case, the slope of the DIVAT Difference line appears to be the same as the bankroll for the middle 100,000 games. After that, VexBot might be changing to a less effective strategy (possibly due to some bad luck in its observations), but it is far from being conclusive.

The round-by-round DIVAT analysis of this match is shown in Figure 5.3, which leads to some interesting features of the match. Vexbot gives up a little equity before the flop (by folding too often). It then loses a great deal of value on the flop round. It roughly breaks even on the turn. However this is all offset by the large gains made on the river.

Vexbot is known to be a very aggressive player, so a likely explanation for its flop losses is the many raises and re-raises made with a weaker hand. These raises cost Vexbot equity in that particular round, but they also create a false impression

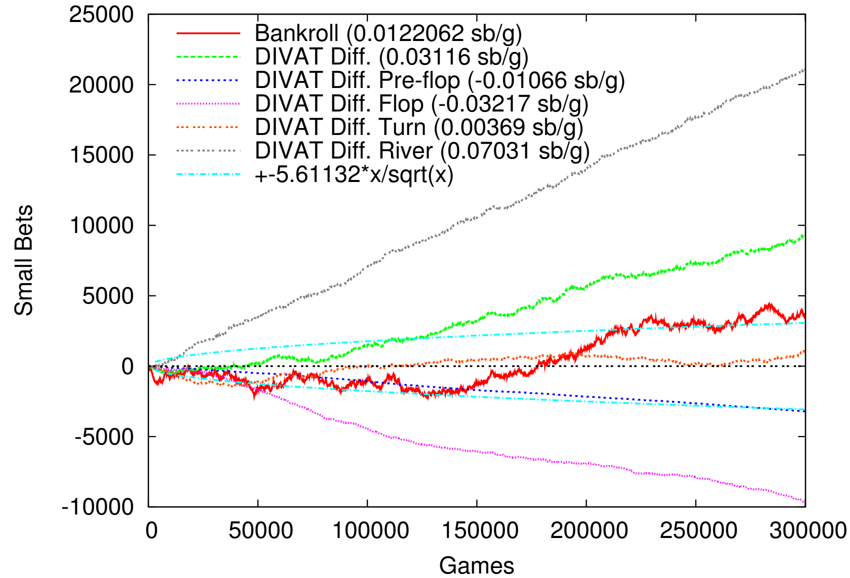


Figure 5.3: Round-by-round DIVAT analysis of Vexbot versus PsOpti4.

of strength to the opponent, which is related to the huge gains in equity that Vexbot experiences on the river. The false impressions of strength can cause Sparbot to fold marginal hands on the river that are strong enough to call and win.

Surprisingly, the DIVAT Difference value on the river maintains the same slope throughout the entire match. Vexbot learned almost immediately that aggressive river play was profitable against Sparbot. It also appears that Vexbot eventually learned that it didn't need quite so much aggression on the flop to setup that river exploitation. This is shown at around hand 80,000 where the flop losses are reduced, and the overall DIVAT Difference begins to break in favour of Vexbot. Thus, this particular counter-strategy to PsOpti4 can be summarized as: play aggressively on the flop, setting up a big bluff on the river.

5.2 thecount versus PsOpti1

In 2003 Gautum Rao, a world class poker player, played a series of games against the University of Alberta's pseudo-optimal program PsOpti1 [4]. Rao, known in online poker rooms by the aliases of "thecount" and "CountDracula", is known to play some of the highest limit poker games available online. The match lasted 7000 games, which is long by human standards (it was played over the course of several

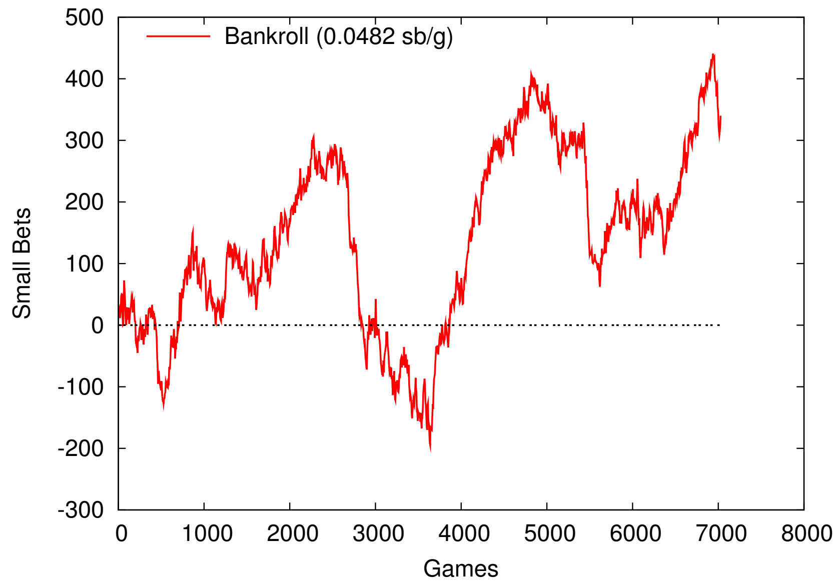


Figure 5.4: Bankroll result of "thecount" versus PsOpti1.

days), but is short in terms of statistical significance. Over the course of the match, Rao appeared to have learned how to beat PsOpti1. The progress of the match is graphed in Figure 5.4. The details of the match are available in [4].

The important features of the match include the large swings, such as the 300 sb drop in Mr. Rao's bankroll over a span of only 400 hands, and the turnaround after about 3700 games when Rao stopped to reassess his style of play. Are these dramatic changes in the bankroll indicative of any particular aspects of the player's skill? In Figure 5.5 the DIVAT analysis plot of the match is shown, leading to some interesting insights.

The DIVAT analysis does not show any of the wild fluctuations that the bankroll did, indicating that these swings can be attributed mostly to the luck of the cards. Indeed, the standard deviation of the DIVAT Difference is only ± 2.109 sb/g compared to ± 5.576 sb/g standard deviation of the money line, for a 7.4-fold reduction in variance.

Over the first 3000 hands of the match, it appears that PsOpti1 held a small advantage over "thecount". Despite the large winning swings in bankroll over these hands, the DIVAT line suggests that the players played close to break-even poker with each other. The dramatic collapse in bankroll looks to be entirely attributable

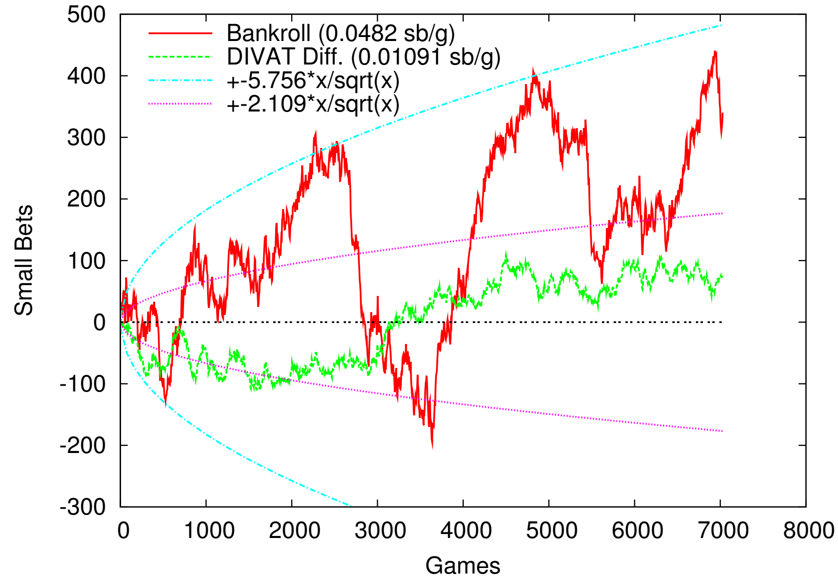


Figure 5.5: DIVAT analysis of "thecount" versus PsOpti1.

to bad luck. In fact, during the later stages of the collapse, "thecount" looks to have saved a few bets and gained value. At this point in the match, "thecount" reassessed his play and switched to a less aggressive style, which appears to have paid off. "thecount" gained the edge over PsOpti1 around that time, after which the players seem to be playing at a break-even pace.

Note that this match is significantly different than some of the test matches highlighted in earlier portions of this paper in that "thecount" is not a static player. It is interesting to note that the DIVAT line has an upward bend to it, indicating that "thecount" is learning and adapting to his opponent. PsOpti is not an easy bot to find exploits in due to its game theoretic approach. However, the bot is static, so its weaknesses can be found through studious play. It is interesting to see that the learning process of "thecount" is reflected in the DIVAT Difference.

Lastly, Figure 5.6 plots the round by round DIVAT Difference analysis for the same match, with some fascinating revelations. Immediately obvious is the very large negative preflop DIVAT Difference indicating that "thecount" was being badly outplayed in preflop play. This is particularly surprising because preflop errors tend to be small in magnitude compared to errors from the flop onwards. After discovering this statistic, the match was reviewed and it was found that "thecount" was

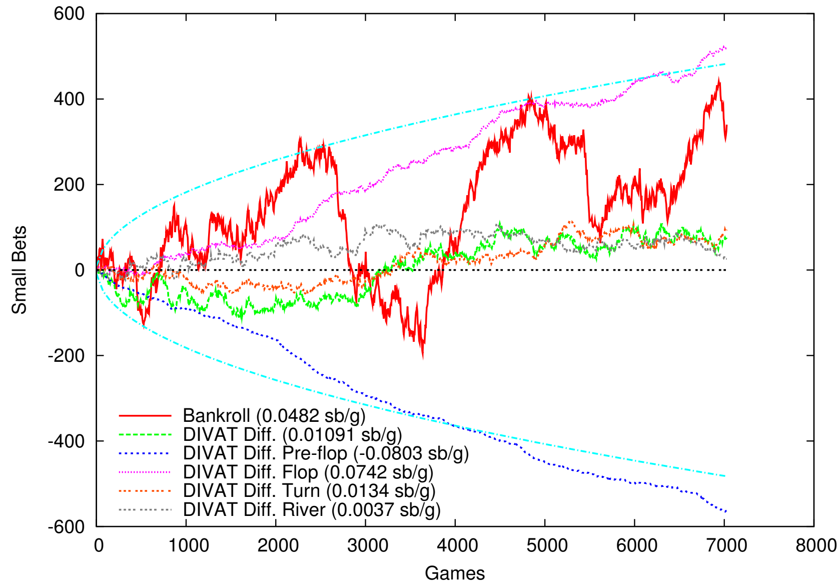


Figure 5.6: Round-by-round DIVAT analysis of "thecount" versus PsOpti1.

folding too frequently before the flop (about 16% of the time) which was causing this massive -0.08 sb/g loss.

The situation is reversed for the flop betting round, where "thecount" makes almost as much on the flop as he lost in the preflop. The cause of this interesting observation is more difficult to deduce than for the preflop. It has not been identified. Several candidate reasons include the fact that "thecount's" tight preflop play now gives him a decided advantage for postflop decisions, since his hands rank among the high 84% of hands. Perhaps "thecount" displays a better understanding of intricate poker strategy including implied odds. Another possibility is that PsOpti's weaknesses are most visible in the flop round of play against this particular opponent. Whatever the reason, "thecount" makes up for most of the ground he lost in the preflop round of play.

Both the turn and the river betting rounds are much less dramatic than the preflop and flop rounds, however they too are interesting. The turn appears to show the same learning trend noticable in the DIVAT Difference sum. In the end, both the turn and river show a small advantage for "thecount", which turns out to be close to the overall result since the preflop and flop metrics largely cancel each other out.

By looking at the round by round analysis, some significant insight into the player's various strengths and weaknesses has been gained. Perhaps the developers of PsOpti may wish to address the obvious problem of flop decisions, while "thecount" may want to address his preflop play. Remember though that these rounds are not independent. If "thecount" were to change his preflop folding policy, then his play in all remaining rounds will be affected. Although DIVAT is a powerful tool, it cannot tell you exactly how to change your strategy for the better!

5.3 Vegas 2005 Matches

In the summer of 2005, a series of matches were played in Las Vegas [1]. In the competition, six poker bots that were previously unknown played a multiplayer single table elimination tournament for $100,000 and the right to play against poker professional Phil Laak. The University of Alberta was present but was not allowed to participate for the prize money; but did play matches against both the winner of the tournament and against Phil Laak (known to the poker world as the "Unabomber"). The winner of the main tournament was PokerProbot.

The main tournament was multiplayer poker which the DIVAT tool does not currently support (this remains future work, and is discussed in Chapter 6). The matches that DIVAT can be applied to are the heads-up matches between Poki-X and PokerProbot, PokerProbot and Phil Laak, and Poki-X and Phil Laak. These matches are analyzed in detail in this section. Each match varies in length due to players having finite chip stacks with escalating blinds. This format adds a great deal of noise to the match because the escalating blinds make the final hands worth more than the rest of the match combined. All analysis is done assuming the players are always playing $10/$20 Hold'em with blinds of $5 and $10.

Unfortunately, the matches are very short, meaning that the results of each match are not statistically significant. The matches are so short that even our lower variance DIVAT estimator cannot decide which player is superior. Despite the power of the DIVAT tool, there is still no substitute for a larger statistical sample for determining the stronger player.

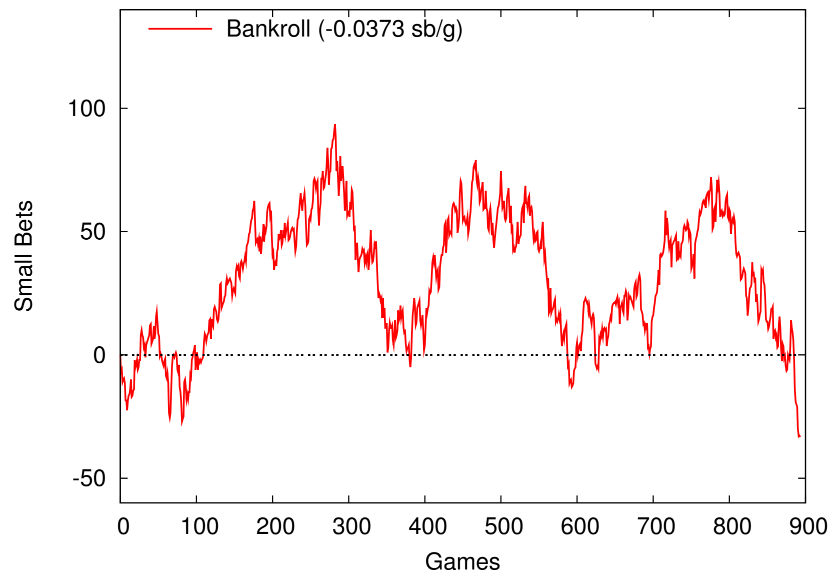


Figure 5.7: Bankroll result of Poki-X versus PokerProbot (fixed stakes of $10/20).

5.3.1 Poki-X versus PokerProbot

The University of Alberta, having done research on building poker bots for several years, was invited to play exhibition matches against the winner of the tournament and against Phil Laak. The bankroll graph for Poki-X against PokerProbot is presented in Figure 5.7. The match lasted just under 900 hands with Poki-X gaining large chip advantages at three stages of the match only to have a large negative swing bring it back to even. Due to the escalating blinds, the last downswing was the worst for Poki-X and caused it to lose all its chips. The question then is whether these swings could be attributed to luck or skill. For a hint at this, we turn to the DIVAT analysis.

DIVAT reveals a very different story than the bankroll (Figure 5.8). The DIVAT analysis indicates that Poki-X held the advantage for most of the match, with the large downswings being largely attributable to bad luck. The standard deviation for the DIVAT Difference is 2.168 compared to 4.680 for the bankroll; a reduction in variance by a factor of 4.7. While not quite statistically significant, the DIVAT score is close to a 70% confidence in declaring that Poki-X is the stronger player.

The round-by-round analysis of this match is presented in Figure 5.9. PokerProbot's greatest weakness appears to be its play in the flop betting round. Poki-X

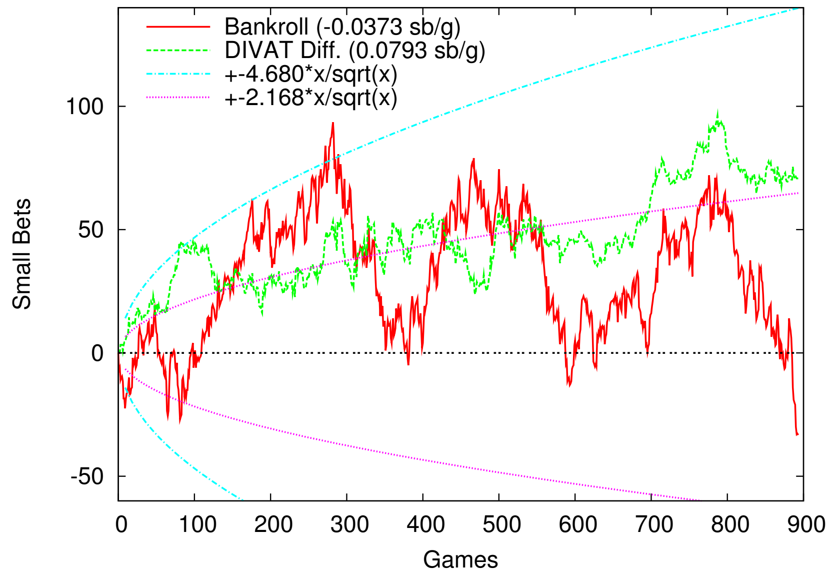


Figure 5.8: DIVAT analysis of Poki-X versus PokerProbot.

does not hold much of an advantage in the preflop or turn betting rounds and is punished on the river betting round. However, the mistakes that PokerProbot makes on the flop are the biggest reason it loses a great deal in the DIVAT Difference score. Once again, this agrees with the assessment made by University of Alberta observers. PokerProbot appeared to play much too tight on the flop which gave Poki-X the chance to earn a large amount of equity from the bot. Given PokerProbot's high selectivity on the flop, it held relatively strong hands on the river. Clearly, Poki-X did not account for this fact adequately, and should have folded more to increase its advantage.

5.3.2 Phil Laak versus PokerProbot

The match between Phil Laak and PokerProbot lasted 399 hands, and the result never seemed in doubt with Phil winning at the very respectable rate of about 0.2 sb/g. His bankroll over time is shown in Figure 5.10. Observers from the University of Alberta described the match as being one-sided with Phil Laak obviously being prepared to meet precisely the type of opponent that PokerProbot represented. PokerProbot is a rule-based bot based on the Poki architecture presented in UofA papers. Poki was designed for the multi-player game, and is known to be weak in

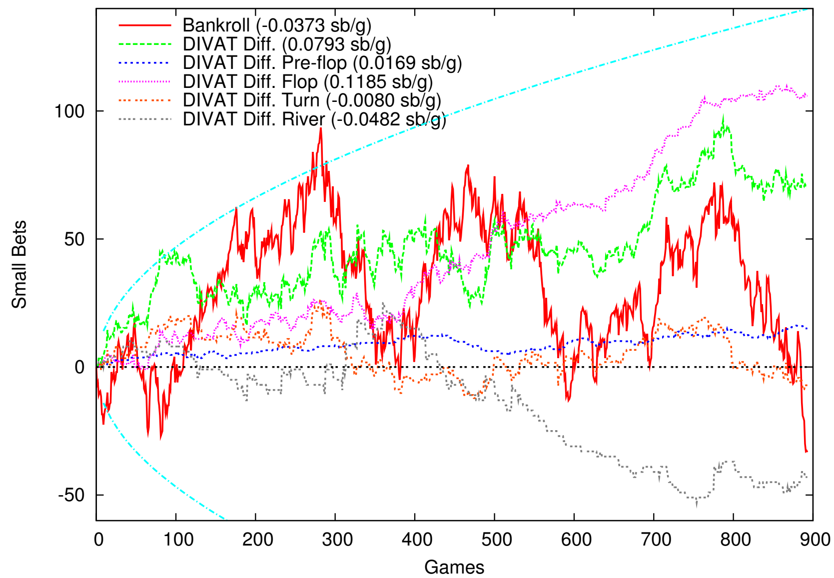


Figure 5.9: Round-by-round DIVAT analysis of Poki-X versus PokerProbot.

the two-player game. Poki is available in the Poker Academy software, which was used by Phil Laak in preparation for the match. As a result, PokerProbot plays a very conservative tight game and would too easily throw away mediocre hands that still had enough value to keep for at least another round.

The DIVAT analysis presented in Figure 5.11 agrees with the bankroll assessment of Phil Laak's play, but with much lower variance. The bankroll win rate is 0.1765 sb/g with a standard deviation of 3.2289, which is amazingly low for a game of heads up Texas Holdem.[1] The variance was so low because most hands ended early: either Phil bets and wins the pot, or he gives up knowing that PokerProbot must have a good hand. The DIVAT win rate is 0.1652 sb/g with a standard deviation of 2.081; a factor of 2.4 reduction in variance. It appears that Phil Laak was very much the better player in this match even though only 400 hands of poker were played.

In Figure 5.12, the round by round DIVAT analysis is plotted to attempt to gain insight into the specific weaknesses of PokerProbot. From this graph, it appears that Phil Laak outplayed PokerProbot mostly on the flop and on the turn. Observers

[1] A normal game of poker typically has a standard deviation of a little less than six. The intuitive reason for this is that most games of poker see each player contributing a bet to the pot each round. Because ties are possible, a player can win or lose six small bets most of the time and sometimes tie.

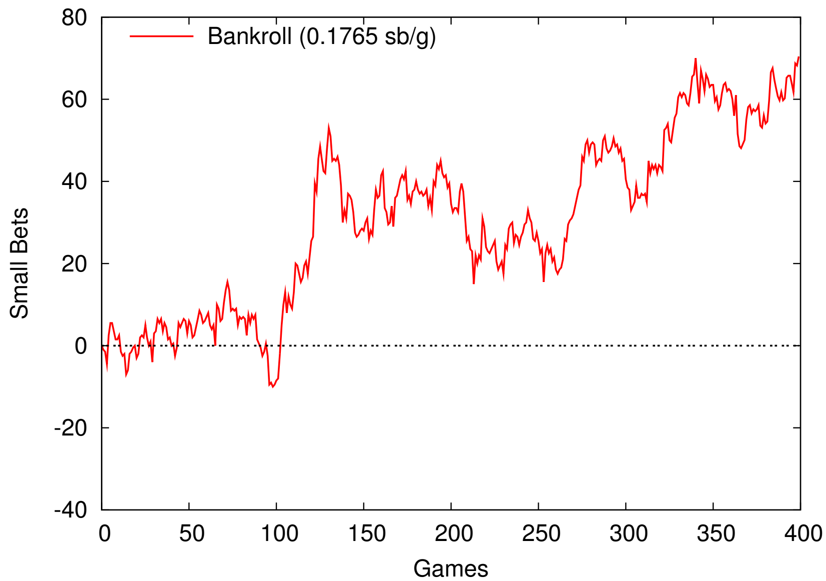


Figure 5.10: Bankroll result of Phil Laak versus PokerProbot (fixed stakes of $10/20).

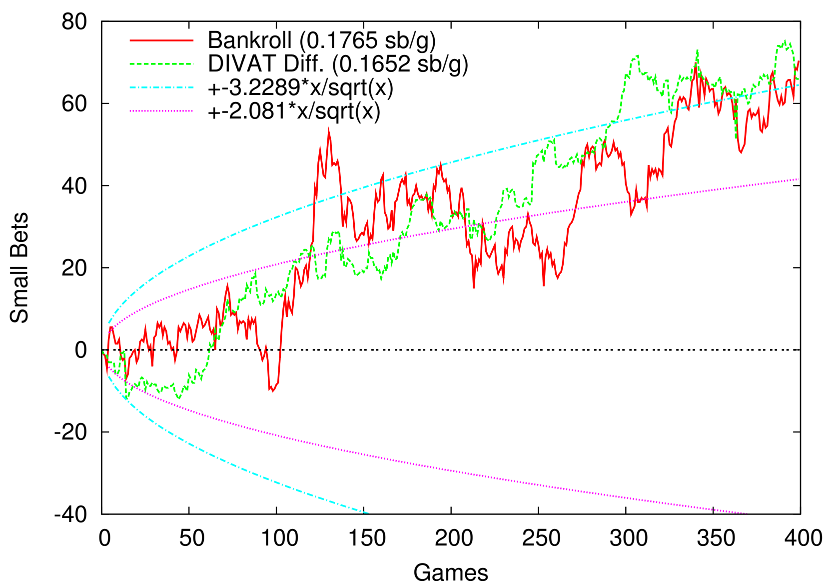


Figure 5.11: DIVAT analysis of Phil Laak versus PokerProbot.

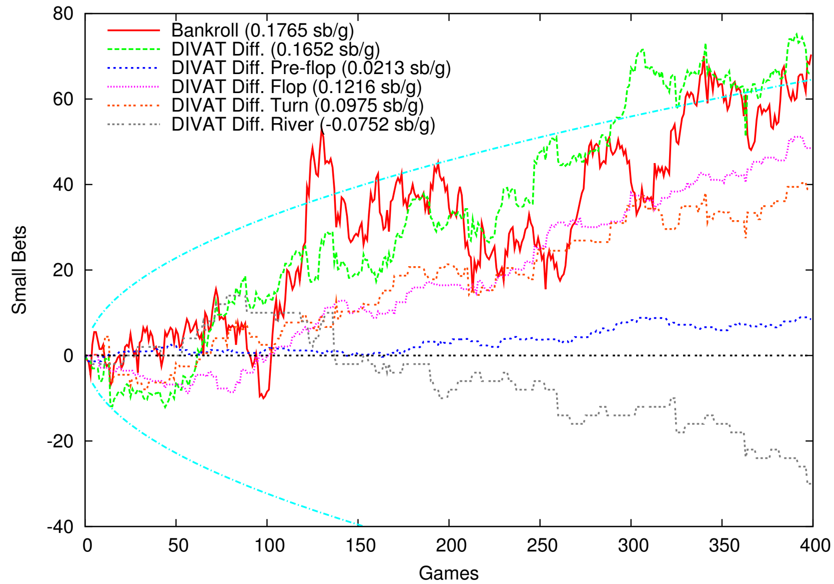


Figure 5.12: Round-by-round DIVAT analysis of Phil Laak versus PokerProbot.

at the match noticed that PokerProbot's play was much too tight on the flop and turn, and the DIVAT analysis seems to agree with that assessment. It appears that the players were close to break-even on preflop play but PokerProbot held a decisive edge on the river. Perhaps the bot's tight flop and turn play lead to easy river decisions with strong hands, and the bot is earning some compensation on those relatively rare occasions.

In all aspects, the DIVAT analysis seems to agree with the observations made at the time of the match.

5.3.3 Poki-X versus Phil Laak

The last match from the Vegas 2005 set of matches was between Phil Laak and Poki-X. The bankroll graph for Poki-X is presented in Figure Figure 5.13. Note that the match was an extremely short 290 hand match which means that even the DIVAT metric is not statistically significant. (The bankroll line truly cannot be trusted, being dominated by short-term chance outcomes). However, DIVAT is the best metric we have, so it is worth using to see what it can show.

Poki-X started the match out very well for the first 60 games. Most of those games ended with one player folding, usually Phil Laak. In the many cases where neither player had a good hand, it is fair to conclude that Poki-X outplayed the human opponent in the early stage of the match.

Game 58 was particularly memorable. Poki-X showed a great deal of aggression holding Q♡6♡ and Phil holding Q♣4♢ on the button. The preflop went call-raise-call with Poki-X doing the raising. The flop came 2♠K♣4♣. The flop went bet-raise-reraise-rereraise-call (the maximum betting which occurred several times early in the match). The turn was the 2♣ with the action being just bet-call. Finally, the river came the K♢. Poki-X executed a *check-raise bluff* which made Phil lay down his pair of fours, which was the best hand. At that moment, Phil remarked "If this is a bluff, it's over for humans". Needless to say, it definitely was a bluff!

Poki-X's advantage early in the match did not carry over into the rest of the match. After the first 60 games, Poki-X's bankroll took a large hit, and the program continued to lose chips for the rest of the match. These chips were worth an increasing amount over the course of the match as the blinds increased and in the end Poki-X lost.

A couple of interesting pieces of information are necessary to understand what happened in this match. The Poki-X architecture actually was composed of two programs. One was a highly aggressive version of the adaptive program Vexbot, and the other was the fourth incarnation of the pseudo-optimal game theoretic program PsOpti (otherwise known as Sparbot). Due to technical issues, the aggressive Vexbot was selected to start the match, with PsOpti4 available if Vexbot lost too many chips.[2] Somewhere around the 90 hand mark, Vexbot had hit enough of a negative streak that PsOpti was swapped in to attempt to stem the loss in chips. Over the next 250 hands or so the bot lost chips, but at a slower rate than the sudden drop that Vexbot experienced. However, the program also had a panic mode where it reverted to Vexbot near the end when there was not many chips left. Phil tightened up considerably near the end of the match in an attempt to get a knockout

[2] Often in tournament play with escalating blinds, the reverse strategy is employed. A tight conservative style of play is adopted for early hands when the blinds are low, but the player then *changes gears* and adopts an aggressive style to pick up antes and blinds as they increase. Because of some architectural issues, the adaptive Vexbot built its opponent model using all games regardless of whether or not it was the program deciding the actions. Because the bot is highly unpredictable with a small number of observations, the decision to let it start the match was almost required.

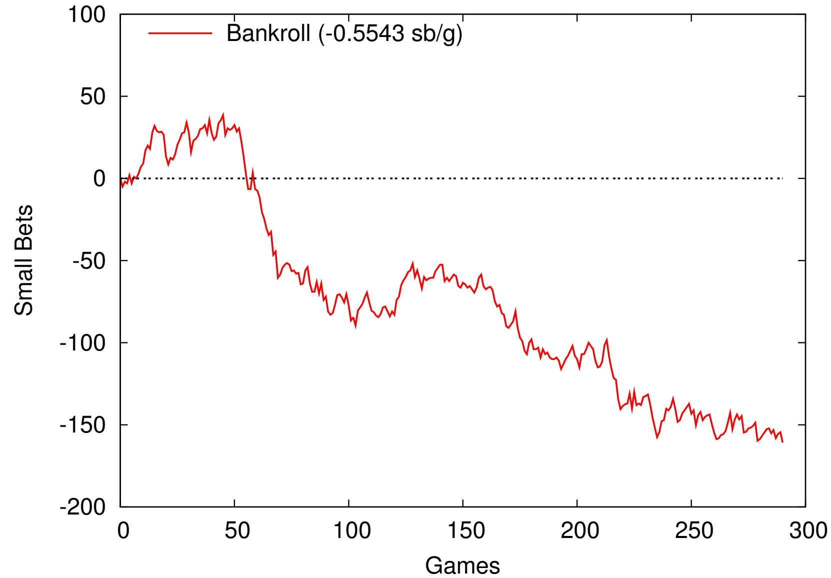


Figure 5.13: Bankroll result of Poki-X versus Phil Laak.

hand to finish off the bot, which he ended up doing.

The DIVAT analysis graph is presented in Figure 5.14. The first noticable thing is that DIVAT claims the match is far from decisive which makes sense given the short sample of hands. In the first 60 hands or so, DIVAT indicates that Poki-X played better. At hand 58, there is a very large spike in DIVAT Difference score showing the gain in equity from its amazing bluff. After PsOpti4 steps in to play around hand 90, the DIVAT score has a slight negative slope indicating that the pseudo-optimal bot is losing at a small rate of -0.049 sb/g. However, once Vexbot is reintroduced, it lost a large number of chips in a short period of time. Near the end of the match when Phil tightened up, Poki-X actually started gaining DIVAT value again as it stole pots from Phil.

It is too bad the match ends at 250 hands, as the match looks like it could be very interesting if it were perhaps 1000 hands in length. In particular, it was very obvious that the number of good opportunities was not balanced. After the first 60 games, the number of times Phil hit a pair on the flop versus no pair for Vexbot was skewed in his favour. Changing to the more conservative PsOpti4 partway through the match could have gained a lot if it had made winning hands, because Phil was conditioned to call by the earlier maniac behaviour of Vexbot. Switching back

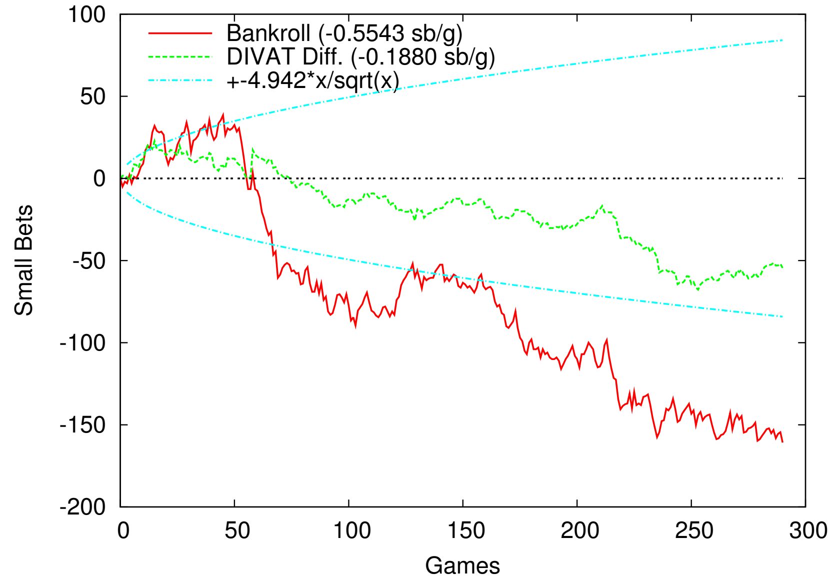


Figure 5.14: DIVAT analysis of Poki-X versus Phil Laak.

to the hyper-aggressive mode backfired, because Phil was still using a defensive style, having had no real reason to change. The DIVAT analysis seems to correctly identify some of the interesting features of the match including the points where Poki-X switched its internal bot, as well as the outrageous bluff that Vexbot pulled off on hand 58.

Finally, the round-by-round DIVAT analysis is shown in Figure 5.15. The preflop line shows that there was no significant difference between the way the players played the preflop for most of the match, except near the end when Phil Laak tightened up to finish the match. The flop line is the largest negative value for Poki-X, due to the hyper-aggressive style Vexbot employed, almost regardless of its hand. The last feature of note is the river round on hand 58 where the bluff took place.

5.4 AAAI 2006 Poker Bot Competition

The first American Association of Artificial Intelligence (AAAI) poker bot competition was held in Boston in 2006. There were few entries in the inaugural event, with the University of Alberta fielding a bot based on two versions of PsOpti. In this section, the results of the competition are examined and some interesting statistics are discussed which DIVAT helps us identify. For most of the matches, a statis-

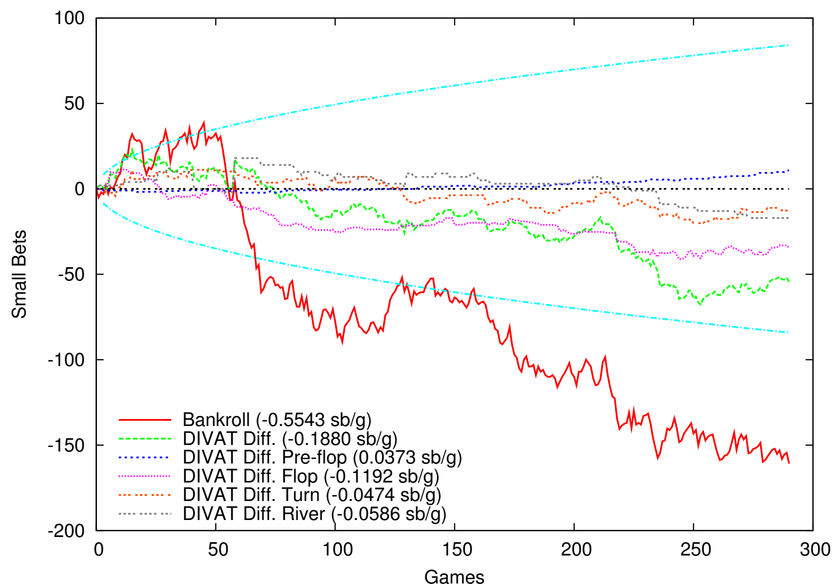


Figure 5.15: Round-by-round DIVAT analysis of Poki-X versus Phil Laak.

tically significant number of hands were played in duplicate format, therefore the DIVAT tool is not necessary to help identify the winner. However, the analysis can help us identify areas where the bots had relative difference in skill.

Tables 5.1 and 5.2 show the results of the competitions at AAAI. The first was the fast "bankroll" event which was the standard event where players received seven seconds per game to make all of their decisions. A player's total winnings were counted to determine the winner. The second was the slow "series" event where the amount of time each program received was much more generous due to constraints of one of the participants. Sixty seconds per game were allocated for each program. These programs were also allowed to use huge amounts of memory resources. One can think of the slow event as a futurity event – a demonstration of what might be possible in the future with more powerful computers.[3]

In all cases in this section, the analysis is done on the duplicate result of each match. A selection of the most interesting matches of the competition were selected for presentation in this thesis. For more complete match by match analysis, the

[3] The competitions were dubbed Bankroll and Series from the way that the final results were determined. In reality, these determinations are arbitrary. Both competitions can be judged using either a series or bankroll result. In the case of these competitions, the result does not change. These competitions will be referred to as the 'fast' and 'slow' competitions to distinguish them more simply.

Fast Event	Alberta	Bluffbot	Monash	Teddy	Average
Alberta	x	+0.05	+0.72	+0.41	+0.39
Bluffbot	-0.05	x	+0.53	-0.19	+0.10
Monash	-0.72	-0.53	x	+1.17	-0.03
Teddy	-0.41	+0.19	-1.17	x	-0.46

Table 5.1: Results of the AAAI 2006 Fast Competition

Slow Event	Alberta	Bluffbot	CMU	Monash	Average
Alberta	x	+0.11	+0.18	+0.73	+0.34
Bluffbot	-0.11	x	+0.12	+0.52	+0.18
CMU	-0.18	-0.12	x	+0.65	+0.12
Monash	-0.73	-0.52	-0.65	x	-0.64

Table 5.2: Results of the AAAI 2006 Slow Competition

reader is referred to [6].

5.4.1 The Competitors

This section is an overview of the competitors that took part. Some of the unique features of the programs including some known strengths and weaknesses are discussed.

Hyperborean is the entry from the University of Alberta. The program is a hybrid of two different versions of PsOpti. These programs approximate a Nash-equilibrium solution for poker by solving an abstracted version of the game. The entry includes a coach that selects which of the two programs will play the next hand based on the performance of each program. The performance was measured using a modified form of the DIVAT tool that helped reduce the variance in situations where the opponent's cards were known. While the programs involved are fairly strong, they aim for an equilibrium strategy, which means they do not try to exploit their opponent. Instead, each program focuses on not *being exploitable*, and neither program does any opponent modelling (which will increasingly be necessary as the competition gets stronger in future years). Two versions of this program were entered, one in each competition. The slow competition bot included a "trickiness filter" which was targeted at some weaknesses the UofA poker group felt the CMU program might exhibit. This filter increased the probability of slow-playing a strong hand and bluffing a weak hand on the flop round of play. The effect of this filter will be discussed in more detail later in this section.

Bluffbot was an entry by Teppo Salonen who implemented a program very similar to the UofA's PsOpti [18]. Bluffbot suffered from the same problems that the UofA bots did: namely, the lack of opponent modelling. An additional weakness of this program is that it only had the one solution, which meant that if it encountered problems against a particular program there is no way for it to escape by switching to another program. Note that there are an infinite number of Nash-equilibrium solutions to the game, and in practice a particular solution can be stronger due to its style of play being harder to play against. A particular feature of this program is that it appeared to fold too often. Evidence of this will be presented later in this section.

Monash BPP is an entry from the Monash University in Australia [14]. Monash's implementation is based on a Bayesian belief network which is updated after each hand. This was the lone entry to employ an adaptive approach. However, evidence of adapting to opponents in the tournament was inconclusive.

GS2 was an entry from Carnegie Mellon University which also attempted a game theoretic approach to poker [12] refining the UofA method of 2003. They used a precomputed solution for the first two rounds of play and then solved the turn and river with a "real-time" linear program. The program utilizes *GameShrink* [12] which is an automatic abstraction tool that makes this approach possible. Due to its approach, this program could not compete in the bankroll event because the real time solutions were much too slow, and used a huge amount of memory. In addition, this program suffered from two fundamental problems. First, because there was a disconnect between the flop and the turn rounds, potentially tricky play on the flop could cause errors in the program's implicit assumptions of its opponent's hand strength. Second, the real time solution sometimes caused the program to be "bumped off the solution tree". This can happen if the program chooses a turn action only to find through further computation that a better action should have been chosen. In this situation, GS2 reverted to simply calling with no knowledge of what situation it was in. We attribute GS2's poor showing to these two issues.

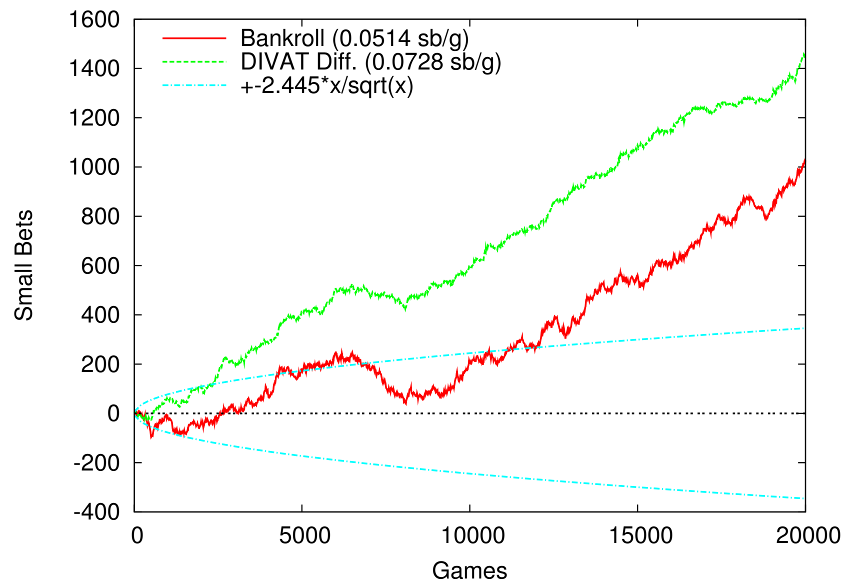


Figure 5.16: DIVAT analysis of Hyperborean versus Bluffbot in the AAAI'06 fast competition.

Teddy was a late entry from Morten Lynge in Denmark. Not much was known about the program, though from the match history it appears that this bot was not much different from an Always-Raise player. Monash was the only program to exploit Teddy to a high degree. However, even Monash did not exploit Teddy to the fullest extent possible.

5.4.2 Hyperborean versus Bluffbot

Hyperborean and Bluffbot are both pseudo-optimal programs. Both consist of pre-calculated solutions which tells each program how to play (a probability of each action) in each situation it encounters. In other words, this match is essentially a list of numbers against another list of numbers. In Hyperborean's case, there are two lists of numbers available for its coach to choose from. Neither bot contains any true adaptivity which means that the winner of this match simply has a style that is better suited to beating its opponent, or has fewer errors in its approximation of equilibrium play.

Figure 5.16 shows the result of the duplicate match with DIVAT analysis. Hyperborean won the match by a healthy margin with statistical significance. The

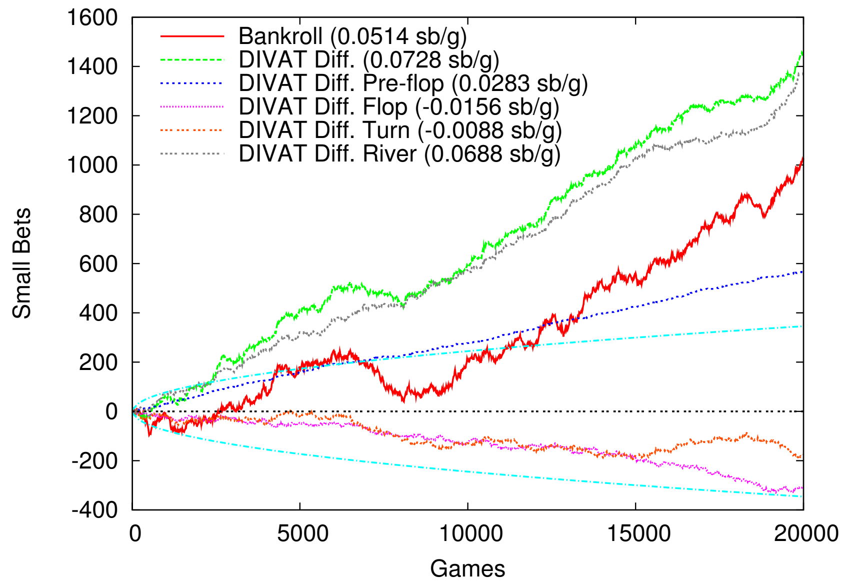


Figure 5.17: Round-by-round DIVAT analysis for Hyperborean versus Bluffbot in the AAAI'06 fast competition.

variance of the duplicate match was 2.445 sb/g. The DIVAT Difference appears to agree with the duplicate bankroll conclusion. The negative run found in the bankroll appears to be mirrored in the DIVAT line though not to the same degree. This dip appears to be a combination of poorer play and bad luck (situations which do not involve the skill advantages). Notice that a duplicate match does not eliminate all bad luck in a match. If one player chooses to fold preflop for instance, but the other player sees the flop, then the outcomes of these hands are very different – particularly when the player who sees the flop makes a strong hand against his opponent's strong hand. Whoever makes the best hand will win a large amount, which is almost entirely a luck outcome.

It appears that the reason the DIVAT line differs from the duplicate bankroll by so much is due to these differences during the first 7500 games. After this point, the DIVAT Difference runs almost directly parallel to the bankroll at a rate of about 0.07 sb/g.

Figure 5.17 shows the round-by-round DIVAT analysis. Of particular interest is that Hyperborean makes a large amount of its winnings on the pre-flop and river betting rounds, with small losses on the flop and turn. The river betting round in

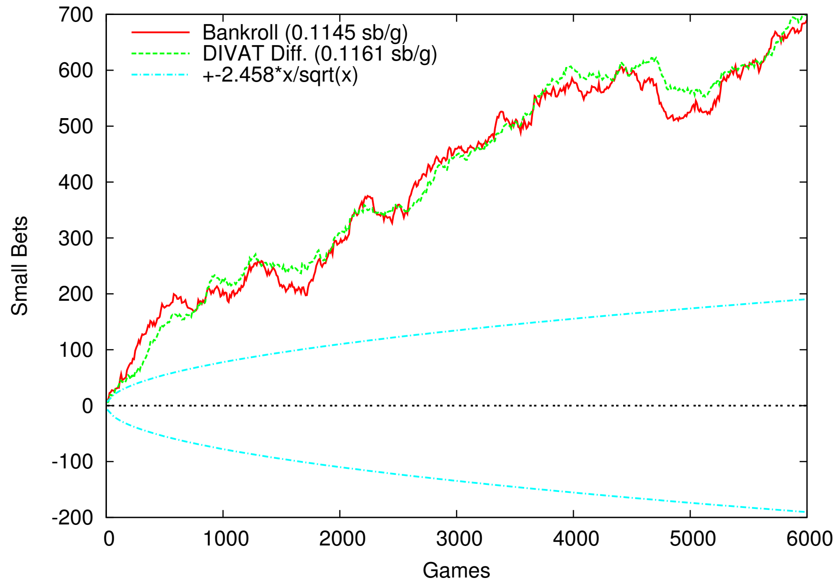


Figure 5.18: DIVAT analysis for Hyperborean versus Bluffbot in the AAAI'06 slow competition.

particular is comparable Hyperborean's overall DIVAT score.

In the match between Bluffbot and Teddy, a program that is essentially always raise, Bluffbot also shows a large loss on the river betting round (and a net loss overall). This seems to indicate that Bluffbot is far too conservative on the river, folding too many hands that should be called or raised.

It is interesting and instructive to look at the differences between the Hyperborean versus Bluffbot matches in both the fast and slow competitions. The Hyperborean program was slightly different for each competition, and this difference shows up in the analysis of the matches.

Figure 5.18 shows the 6000 hand duplicate match between Hyperborean and Bluffbot. The first thing to notice is while Hyperborean also wins this match, the win rate is 0.11 sb/g which is significantly larger than the 0.07 sb/g observed in the fast competition. Since Bluffbot was unchanged between the competitions, the only difference was the changes in Hyperborean. As discussed earlier, the lone difference between the two versions of Hyperborean was a "trickiness filter" that increased both slow-playing strong hands and bluffing weak hands on the flop betting round. The trickiness filter does not come into effect very often, but is shown

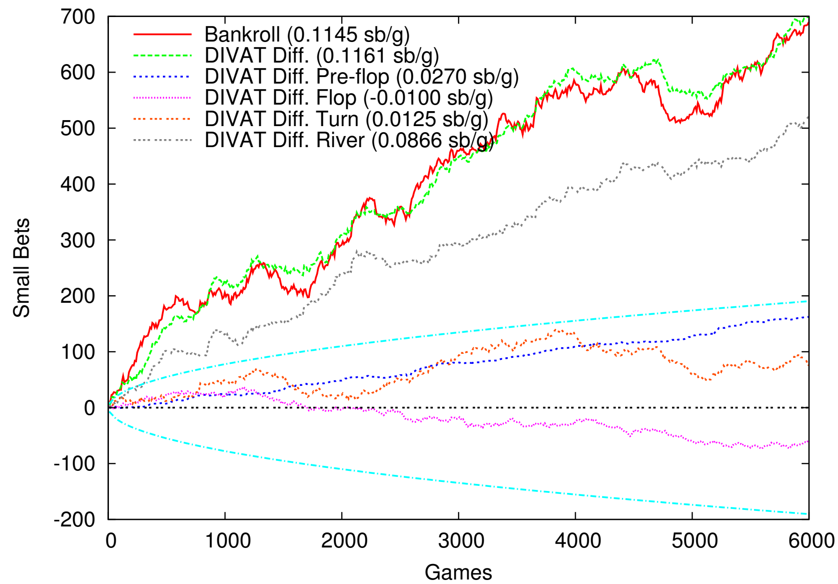


Figure 5.19: Round-by-round DIVAT analysis for Hyperborean versus Bluffbot in the AAAI'06 slow competition.

to be more effective against Bluffbot.

Figure 5.19 shows the same match with round-by-round DIVAT analysis. Note that the preflop rate of winning is unchanged which helps verify that the only difference between the programs is from the flop onward. The DIVAT analysis for the flop indicates that there wasn't a distinguishable difference in EV between the two variants of Hyperborean. However, on the turn and river, Hyperborean wins 0.04 sb/g more than in the fast competition. From this evidence, it appears that the trickiness filter was a success in confusing Bluffbot into making additional errors on the turn and river. Remember though that these are still two bots with lists of numbers competing. A truly adaptive opponent might be able to learn the effects of the trickiness filter and use that knowledge against Hyperborean.

5.4.3 Hyperborean versus GS2

GS2 was the latest version of a pseudo-optimal poker bot with several additions to the ideas used in PsOpti. Utilizing the *GameShrink* algorithm [12], the program was capable of discovering a solution for the game fairly quickly, and thus used this idea in a "real time" solver for the turn and river. Given that the program's

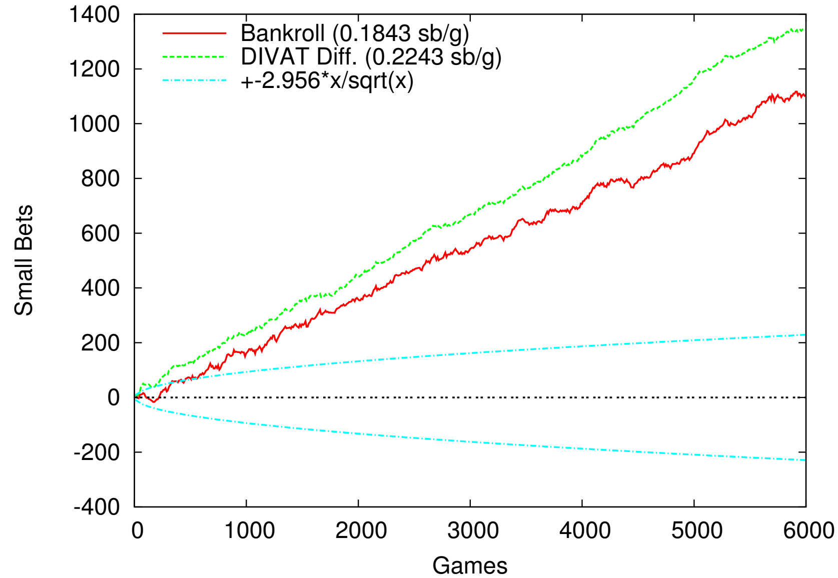


Figure 5.20: DIVAT analysis for Hyperborean versus GS2 in the AAAI'06 slow competition.

developers had some very solid ideas for improving the play of a pseudo-optimal program, the question becomes why did it not win? Figure 5.20 shows the duplicate bankroll alongside the DIVAT Difference graph for the match between Hyperborean and GS2. Hyperborean won by several standard deviations making it the runaway winner. The DIVAT analysis suggests that the margin of victory could have been even larger.

Some of GS2's disadvantages have already been discussed. There exists a disconnect between the play on the flop and the turn. The assumed priors for the opponent's possible cards for the turn and river can be wrong, which can lead to some disastrous results (as the University of Alberta's poker group has experienced with other approaches). Finally, the program can "fall off the game tree" when the game was solved for the turn and then resolved for the river. In such cases, the programs takes a default action of 'call'. Since it is not known how many times this happened, we do not know for sure which of these problems was the most serious. However, in reviewing the games by hand, GS2 frequently made bad calls on the turn and river where Hyperborean correctly folded in the same situation. This seems to indicate that the calling errors cost GS2 a lot of equity.

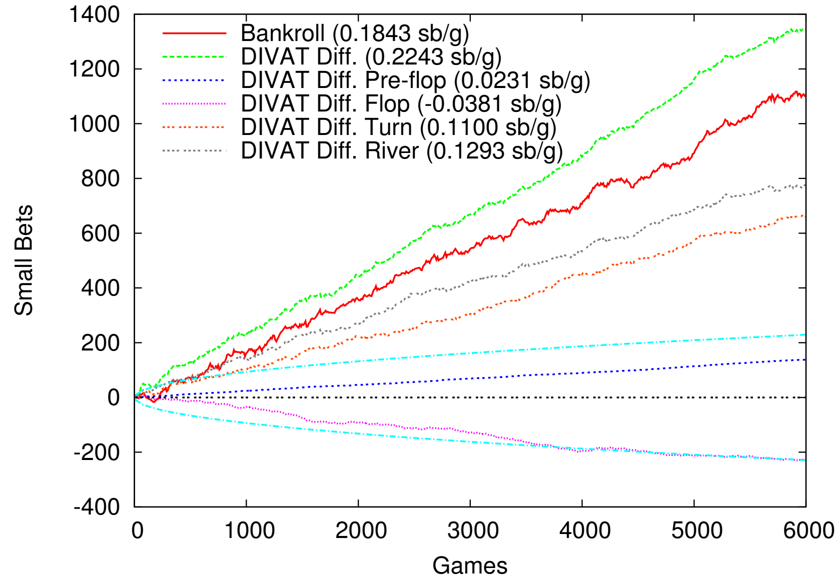


Figure 5.21: Round-by-round DIVAT analysis for Hyperborean versus GS2 in the AAAI'06 slow competition.

The DIVAT round-by-round analysis is shown in Figure 5.21. Hyperborean shows a small loss on the flop, perhaps some of this is from the trickiness filter which intentionally gives up equity on the flop, for purposes of deception, in the hope of earning much more when the bet size doubles on the turn. Hyperborean exhibits a small preflop edge, but this is dwarfed by the very large turn and river gains earned by Hyperborean. It is likely that some of these gains are attributable to the trickiness filter.

5.4.4 Monash versus Teddy

In one of the interesting turns in the tournament, the non-transitivity of poker was dramatically evidenced. Teddy is a bot that very closely resembles always raise. Post-tournament analysis suggested that Teddy was always raise even when a raise wasn't allowed (when the betting was already capped). The tournament software deemed this to be an illegal action, in which case the program's hand was folded. Monash is based on a Bayesian network, and has the possibility of learning from its opponent. However, it is not clear that this happened. An always raise bot can be exploited to a maximum degree of around 3 sb/g. As seen in Figure 5.22, Monash

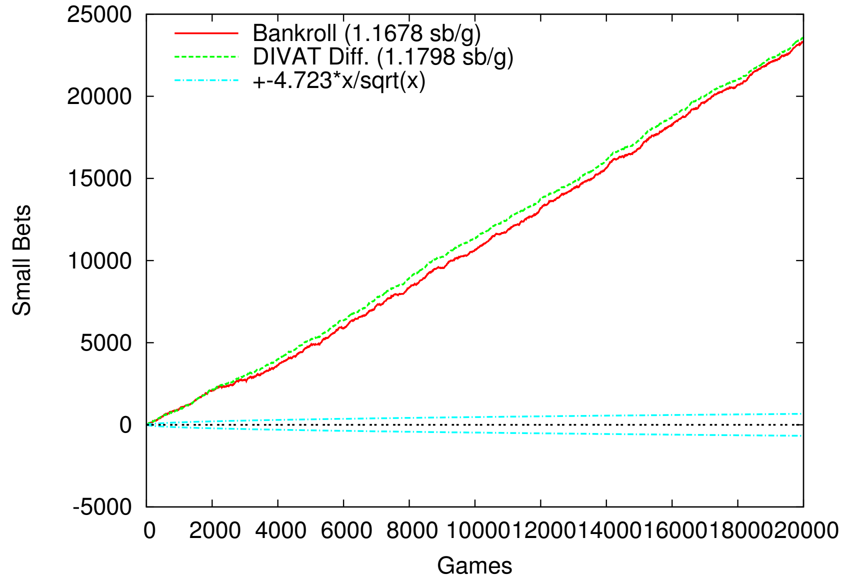


Figure 5.22: DIVAT analysis for Monash versus Teddy in the AAAI'06 fast competition.

acheived less than half of this win rate.

The tournament was played in twenty 1000 game matches, but the DIVAT lines do not seem to show any learning curve. So either Monash was quick to learn that its opponent raised a large amount, and did not learn to exploit the opponent for the full amount, or Monash's style coincidentally matched a fairly effective counter-strategy to Always-Raise. At this point, which of these cases is closer to the truth is unclear.

Figure 5.23 shows that Monash had slight losses in the preflop, slightly positive returns on the flop, and large gains on both the turn and river in roughly equal quantities. This is similar to a natural counter-strategy against Always-Raise. Lower quality hands may have been thrown away on the flop before they can lose too much, but higher quality hands should be raised to gain the most value, and usually played to the end. Thus, a conventional "tight but aggressive" approach would do very well against this extreme opponent.

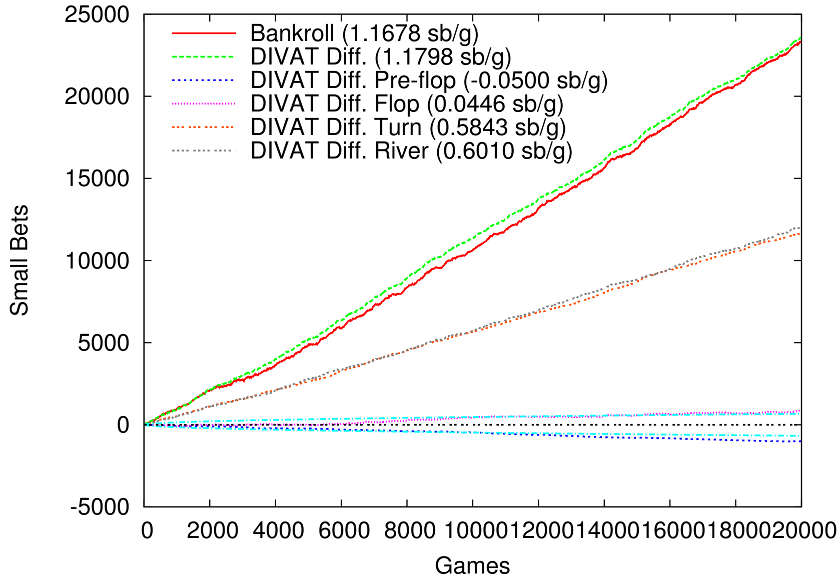


Figure 5.23: Round-by-round DIVAT analysis for Monash versus Teddy in the AAAI'06 fast competition.

Chapter 6

Conclusions and Future Work

The DIVAT metric has a great deal of promise for analyzing how well players play the game of poker. The variance reduction means that players need to play quadratically less number of games to get the same statistical significance. This makes statistically significant matches between humans and machines much more likely.

There are several obvious ways to continue development of the DIVAT metric. First, multiplayer DIVAT would likely have more interest in the poker community, since multiplayer poker is far more common than heads-up. While there was not enough time to complete an initial multiplayer DIVAT, some interesting observations and challenges have already been identified for it. A second extension to DIVAT would be for handling the No-Limit variety of Texas Hold'em. No-Limit heads-up Texas Hold'em should not be too difficult to consider, since the large number of bet sizes can be pruned to a small set of effective bet sizes. The third obvious extension is to make DIVAT handle other popular games of poker such as 7-card Stud, Omaha, Omaha Hi-Lo, or Razz. In principle, all of these games can be approached in much the same way the original DIVAT was formulated.

6.1 Multiplayer DIVAT

Probably of greatest interest for future DIVAT development is extending the analysis to multiplayer Limit Texas Hold'em. As was quickly discovered, even extending the computation to three players gives rise to a number of complexities.

The first idea is to apply the same structure as discussed in this thesis for the

DIVAT tool for the heads-up game. Obviously some changes need to be made for the baseline betting sequence. In the two-player game, if the opponent raises, then there is just one more bet to call. However, in the multiplayer case, it is possible that one player bets and another raises before action reaches you. In this case, you must call two bets to continue with the hand. Obviously, the amount to call influences the frequency of calling. One way to handle this would be to have M1, M2, M3, M4 and C thresholds as before, but also F1, F2, F3, and F4 values for folding to a number of bets. Once the baseline is defined, it is theoretically possible to calculate the DIVAT metric for multiplayer in the same way it was done in this thesis.

There are a couple of issues to overcome with this method, however. First, in the heads-up game, the preflop DIVAT ROE calculation already took a long time to compute. In the multiplayer game, the time required to calculate these tables in the same way is infeasible at the current time. It may be good enough to use Monte Carlo simulation in the place of exact enumeration, but this has yet to be explored.

A second, and more serious problem is in attributing credit or blame for a given outcome. In the two-player game, the equity lost by a player is always gained by the other. The situation is not as simple for the multiplayer game. It is possible in the multiplayer scenario to lose equity even though they played the hand correctly. Suppose there are three players and the baseline sequence is bet-call-call. Now suppose that the actual sequence went bet-raise-fold, and the third player correctly chose to fold because the second player incorrectly raised to force him out. Player 3 was forced to forfeit his non-zero pot equity in the hand because the second player deviated from the baseline.

This scenario causes the multiplayer DIVAT to look a little less useful, in principle. Ideally, an analysis tool would be capable of assigning "skill points" to the game for the actions the player made during the hand independent of the actions made by the opponents. Unfortunately, based on the way DIVAT is formulated, situations arise where a player cannot help losing DIVAT Difference score through no fault of their own.

A possible workaround for this problem is to take a difference between the equity that a player has before and after each *decision* (rather than before and after

	Player 1	Player 2	Player 3
Player 1	P1 influences self	P1 influences P2	P1 influences P3
Player 2	P2 influences P1	P2 incluences self	P2 influences P3
Player 3	P3 influences P1	P3 influences P2	P3 influences self
Sum	P1 DDiff	P2 DDiff	P3 DDiff

Table 6.1: Sample Multiplayer DIVAT Crosstable

all decisions for that round). This remains an unbiased estimator, with a proof similar to the proof for the original DIVAT formulation. In this example, however, there are some intriguing features which may be worth the added complexity.

Table 6.1 shows a sample crosstable obtained when DIVAT differences are calculated after each action instead of after each round. The diagonal in the crosstable shows how much each player's own actions affect their DIVAT score. Values outside of the diagonal show how each player's decisions affect their opponents' DIVAT scores. The column sums would equal the DIVAT scores for each player, but the real value of the table allows you to identify situations where a player has a disadvantage, not because of their own decisions but because of the actions of the other players. In multiplayer poker, there exists situations where a player who exhibits superior skill cannot win, simply because of the situations the player is put in. Some of these situations are the consequence of the various opponent strategies and the player's relative position to the opponents. Other examples involve collusion among opponents (which is illegal but not preventable).

This approach to the DIVAT Difference score should be able to identify situations which were not favourable to the player, whether they be the coincidental clash of strategies, or explicit collusion. Because of this very interesting feature, this method of calculating a DIVAT Difference score bears further investigation. Of course, this caculation will run much slower than the current DIVAT implementation, and will require more complicated tables to be calculated, which may not be possible at the current time. Instead of calculating DIVAT rollouts for each pair of player hands and four different pot sizes, a rollout table would need to be able to handle lookups from any point in the betting sequence with a set of *n-player* hands (in an *n-player* game). This table would probably be prohibatively expensive to compute at this time, so some research needs to go into making that efficient, or for other techniques to be developed.

6.2 No Limit DIVAT

To handle No Limit Texas Hold'em, it should be sufficient to consider several different bet sizes as the only available options. These options will likely be fold, check, bet half the pot, bet the pot, bet twice the pot, and go all-in. Instead of only three possible actions there are six, which constrains the large number of bet sizes down to a manageable number of options. Once again, it is important to realize that the baseline sequence does not need to be a perfect player. Instead, it only needs to be somewhat reflective of how much each player should be willing to put into the pot. Therefore, not all cases need to be considered – in fact, a subset should work quite well in practice.

6.3 Other Poker Games

Other poker games should not represent a huge challenge for the extension of the DIVAT technique. The concepts discussed in this thesis for the DIVAT method apply to other poker games equally well. For instance, a hand strength metric can be calculated in an analogous way for every variation of poker. Similarly, once a hand strength metric has been defined, betting thresholds for how much each player wishes to put into the pot can again be determined, and the tool would be well on its way to being complete.

6.4 Other Applications

The idea of using a baseline strategy for unbiased comparison between players should be applicable to many other domains. One possible example is an analysis of real time strategy (RTS) game battles [9]. Suppose there are two players with a set of armies on both sides. Typical RTS games include randomization of damage done to units, hidden information in the strength of the opponent units, and other details to consider. In such a match, a baseline action sequence could be used to determine which player is making better actions relative to each other. If the variance in the match was high enough, it is possible that the player with the better decisions could lose the match, but a baseline differential in moves would reveal this.

The components of a DIVAT-like tool are simple (hand strength metric, baseline policy, and expectation calculation) and could potentially be applied to evaluation in many domains. It is simple but effective in its purpose – to reduce much of the noise in high stochastic areas. Therefore, the key features of a domain where a DIVAT-like tool would succeed are: high variance stochastic events, and a domain where skill is the determining factor over the long term.

Appendix A

Glossary

This glossary defines poker terms used throughout this thesis. For a more complete and indepth glossary of poker terminology, the reader is referred to any of the following websites:

- http://www.seriouspoker.com/dictionary.html
- http://conjelco.com/pokglossary.html
- http://en.wikipedia.org/wiki/Poker_jargon
- ***Aggressive***. Refers to a player's style of play. An aggressive player will tend to raise and re-raise instead of calling. These players predominately raise or fold rather than call.
- ***Betting for Value***. Betting with the expectation that if the opponent calls you will win more. To bet for value, your hand needs to compare favourably with the probable opponent holdings.
- ***Big Bet***. In Limit Texas Hold'em, the amount a player is allowed to raise by in the turn and river betting rounds. The amount of a big bet is determined by the table's stakes and is usually twice that of a small bet.
- ***Big Blind***. In Texas Hold'em, the player two places to the left of the dealer posts a forced bet called the big blind. This amount is set depending on the stakes being played. The amount of the big blind is usually twice that of the small blind, and equal to a small bet.

- ***Bluff***. Making a bet when holding a weak hand that has little chance to win if called. Bluffs are used in balance with bets that are intended to be for value to create uncertainty about the strength of their hand. If a player does not bluff, they will be too predictable and therefore be exploitable to observant opponents.

- ***Capping the Betting***. In limit Texas Hold'em, there is usually a betting cap of four raises (including the initial bet) in a particular round. Once four raises have occurred in a given betting round, no more raises are allowed.

- ***Check-raise***. A trap play in which a player checks (feigning weakness) and encouraging an opponent to bet, and then raising at the next opportunity. Most often, this indicates a strong hand, but can also be used as a bluff.

- ***Community Card***. A public card that all players can use to make their best poker hand.

- ***Dealer Button***. In Texas Hold'em, the player who has the button is denoted the dealer, and acts last on every betting round after the first.

- ***Drawing Dead***. A player is drawing dead if no future combination of community cards can give them at least a tie with the best hand. As an example, the player with just a pair who cannot improve to more than three of a kind is drawing dead if his opponent already holds a flush.

- ***Flop***. (a) The first three community cards dealt in Texas Hold'em. (b) The betting round that commences after the first three community cards have been dealt.

- ***Flush***. A poker hand ranked higher than a straight, but lower than a full house. Consists of five cards of the same suit.

- ***Free Card***. When all players check through to the next round, the card dealt is deemed to be a free card. The player with the best hand is said to be "giving a free card" when they check rather than bet.

- ***Full House***. A poker hand ranked higher than a flush, but lower than a four of a kind. Consists of three of kind and a pair (eg. A-A-A-3-3).
- ***Heads-up***. To play poker with just two players (either from the outset of the game or at the late stages of a multiplayer game).
- ***Hand Rank***. A metric that compares the relative strength of a player's holding with all possible opponent holdings assuming a uniform distribution of opponent cards. The Hand Rank metrics used are defined in Chapter 3.
- ***Hole Card***. A player's private card in poker, unknown to the opponents.
- ***Implied Odds***. The expectation of winning bets in future rounds. It considers situations after the future cards are dealt and shows how the situation is better than the immediate pot odds indicate.
- ***Loose***. Refers to a player's style. A loose player plays comparatively more hands than a tight player.
- ***Nut Flush***. See Nuts.
- ***Nuts***. The best possible hand. Sometimes used to describe the best possible straight or flush (as in nut flush). Also can be combined to make the second best possible hand. For example the phrase "second nut flush" means the second highest flush possible losing only to the nut flush.
- ***Passive***. Refers to a player's style of play. A passive player will tend to check or call other player's bets rather than raise themselves. These players predominately call or fold rather than raise.
- ***Pot Equity***. A measure of how valuable a current hand is. It is a share of the current pot with regard to the likelihood that the player wins the pot. To calculate pot equity, we determine the percentage chance that the player wins the pot against the opponent, and assign that player the same percentage of the pot as his pot equity.

- ***Pot Odds***. The ratio of money in the pot to the money required to call. This ratio is often used to decide whether it is correct to call a bet with a weaker hand. If the hand can improve to the likely best hand with probability better than the pot odds offered, then the call is possibly correct. Pot odds are often influenced by implied odds and reverse implied odds.

- ***Protect your hand***. When a player has a strong hand that is vulnerable to the opponent making a stronger hand when future cards are dealt, it is usually a good idea to bet to give the opponent the chance to fold, rather than giving them a free card.

- ***Reverse Blinds***. A fairly common way of playing heads-up Texas Hold'em. The dealer posts the small blind and the other player posts the big blind. In the preflop betting round, the dealer acts first and then acts last in all remaining rounds. This is reversed from the normal method of having the player to the left of the dealer post the small blind.

- ***Reverse Implied Odds***. The expectation of losing bets in future rounds. It considers situations after the future cards are dealt and shows how the situation is worse than the immediate pot odds indicate.

- ***River***. (a) The fifth community card dealt in Texas Hold'em. (b) The final betting round after the fifth community card is dealt.

- ***Slowplaying***. To play a strong hand as though it were weak by checking or calling until a later round. Used to mix up a strategy, or as a value play with the expectation of winning more later in the hand.

- ***Small Bet***. In limit Texas Hold'em, the amount a player is allowed to raise in the pre-flop and flop betting rounds. Often used as a unit of measure to compare performance, as in small bets per game. The amount of a small bet is determined by the table's stakes and is usually half of a big bet.

- ***Small Blind***. In Texas Hold'em, the player immediately to the left of the dealer (unless playing heads-up with the reverse-blinds format) posts a forced bet called the small blind. This amount is set depending on the stakes being played. The amount of the small blind is typically half of the big blind.

- ***Starting hand***. In Texas Hold'em, the first two hole cards a player is dealt.

- ***Straight***. A poker hand ranked higher than three of a kind, but lower than a flush. Consists of five cards of differing suits in sequential order (eg. 3-4-5-6-7).

- ***Straight Flush***. The highest ranking poker hand. Consists of five cards of the same suit in sequential order (eg. 3-4-5-6-7, all hearts).

- ***Tight***. Refers to a player's style. A tight player plays comparatively fewer hands than a loose player.

- ***Trips***. When a player makes three of a kind. In Texas Hold'em, trips is when the player makes three of a kind using two community cards. In contrast, making a ***set*** refers to only using one community card to make three of a kind (when holding a matching pair in their hole cards).

- ***Turn***. (a) The fourth community card dealt in Texas Hold'em. (b) The betting round after the fourth community card is dealt.

Bibliography

[1] Poker Academy. World poker robot championship 2005 information page. WWW, 2005. `http://www.poker-academy.com/wprc`.

[2] D. Billings. Computer Poker. Master's thesis, Department of Computing Science, University of Alberta, 1995.

[3] D. Billings. *Algorithms and Assessment in Computer Poker*. PhD thesis, Department of Computing Science, University of Alberta, 2006.

[4] D. Billings, N. Burch, A. Davidson, T. Schauenberg, R. Holte, J. Schaeffer, and D. Szafron. Approximating game-theoretic optimal strategies for full-scale poker. In *The Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence, IJCAI*, pages 661–668, 2003.

[5] D. Billings, A. Davidson, T. Schauenberg, N. Burch, M. Bowling, R. C. Holte, J. Schaeffer, and D. Szafron. Game-tree search with adaptation in stochastic imperfect-information games. In *Computers and Games*, pages 21–34, 2004.

[6] D. Billings and M. Kan. Development of a tool for the direct assessment of poker decisions. Technical Report TR06-07, University of Alberta Department of Computing Science, April 2006.

[7] D. Billings and M. Kan. A tool for the direct assessment of poker decisions. *International Computer Games Association Journal*, 29(3):119–142, 2006.

[8] M. Buro. The Othello match of the year: Takeshi Murakami vs. Logistello. *International Computer Chess Association Journal*, 20(3):189–193, 1997.

[9] M Buro. Real-time strategy games: A new AI research challenge. In *Proceedings of the International Joint Conference on AI 2003*, pages 1534–1535, 2003.

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

[11] C. Donninger and U. Lorenz. Hydra chess webpage. World Wide Web, 2005. `http://hydrachess.com/`.

[12] A. Gilpin and T. Sandholm. A competitive Texas Hold'em poker player via automated abstraction and real-time equilibrium computations. In *American Association of Artificial Intelligence National Conference, AAAI*, pages 1007–1013, 2006.

[13] M. Guid and I. Bratko. Computer analysis of world chess champions. 29(2):65–73, 2006.

[14] K. B. Korb, A. E. Nicholson, and N. Jitnah. Bayesian poker. In *Proceedings 15th Conference on Uncertainty in Artificial Intelligence*, pages 343–350. Morgan Kaufmann, 1999.

[15] C. Liu. Gambling Tools. WWW, 2000. `http://www.gamblingtools.net/`.

[16] B. Mezrich. *Bringing Down the House: The Inside Story of Six M.I.T. Students Who Took Vegas for Millions*. Free Press, 2002.

[17] M. Newborn. *Kasparov versus deep blue: computer chess comes of age*. Springer-Verlag New York, Inc., New York, NY, USA, 1996.

[18] T. Salonen. The official website for Bluffbot. WWW, 2006. `http://www.bluffbot.com`.

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

[20] M. Shackleford. Wizard of Odds. WWW, 1998. `http://wizardofodds.com/`.

[21] B. Sheppard. World-championship-caliber scrabble. *Artificial Intelligence*, 134(1-2):241–275, 2002.

[22] D. Sklansky. *The Theory of Poker*. Two Plus Two Publishing, 1992.

[23] G. Tesauro. Temporal difference learning and TD-gammon. *Communications of the ACM*, 38(3):58–68, 1995.

[24] E Thorp. *Beat the Dealer*. Random House, 1962.

[25] Wikipedia. Law of iterated expectations. World Wide Web, 2006. `http://en.wikipedia.org/wiki/Law_of_iterated_expectations/`.

[26] D. Wolfe. Distinguishing gamblers from investors at the blackjack table. In J. Schaeffer, M. Müller, and Y. Björnsson, editors, *Computers and Games 2002*, LNCS 2883, pages 1–10. Springer-Verlag, 2002.

[27] M. Zinkevich, M. Bowling, N. Bard, M. Kan, and D. Billings. Optimal unbiased estimators for evaluating agent performance. In *American Association of Artificial Intelligence National Conference, AAAI*, pages 573–578, 2006.