Scott Layton

Robbie Plankenhorn

Rob Webb

**Milestone 4**

**Revised Project Description:**

(The Project Description has not changed from milestone 2.)

Our project is to create an intelligent system (dubbed “The Poker Coach”) that will assist a human player in the card game of Texas Hold’em.

The main tasks include:

* Create a user interface that can be used while a human user is playing a hand of Texas Hold’em.
* Create statistical functions that can be used by the system to determine how the human user should play their hand.
* Keep a database of past hands as well as data on individual opponents that the user will encounter while playing and use this data to further increase the user’s chance of winning.

Domain:

A windows application to be used on a personal computer.

Users:

One human user.

The main goals for the “Poker Coach” are as follows:

* To provide the human user with worthwhile statistical advice about their two-card hand (standard in Texas Hold’em).
* To provide the user with updated statistics as each hand progresses from pre-flop, flop, turn, river, and showdown.
* To provide intelligent advice on how the user should play based on the users current chip count (unit of value) and the bets being made by opposing players.
* Keep data on opponents betting strategies and use this data to offer advice to the user.
* Win, the ultimate goal of this system is to win more poker hands.

**Requirements:**

Functional:

* Provide users with advice on how to play their hands with the goal being to win more poker hands and larger pots. This advice would be based on statistics, betting strategy, and data kept on previous hands and opponents.
* The output would take two forms: first, a suggestion to bet/raise, call, check, or fold but more importantly, second, a list of factors that the computer has used to determine this suggestion.
* Factors will include statistics, possible other hands the opponent could have, and observations the computer has made about the opponents style of play.

Performance:

* Provide said advice in a timely manner; in a way that would not slow down the general flow of a game of poker between up to 10 human players.
* Provide the statistics and observations that the computer used to the make the advice in an accessible manner so that the user can quickly digest the relevant points.
* Allow the user to quickly enter the relevant information without unnecessary distraction.

Optional:

* Add graphics that include card faces, etc.
* Allow for varying styles of play; aggressive, tight, etc.

**User Feedback:**

As the developers themselves are the primary targets, we have attempted to deploy the poker coach in the field ourselves. Simply, we have used our program while playing poker on-line or with friends. Our results at this stage are mixed.

The UI needs to be improved because of the time demands of the game. It is difficult to enter all of the relevant information about each hand before the computer would need to have given advice. This results in clumsy play and annoyance from the other players. To rectify this situation, we suggest that the user input data through buttons instead of a pull down menu.

The advice provided by the system is also limited at this point to pre-flop play. While this within the progress schedule of milestone 2, it makes the usefulness of the program somewhat limited. For the advice to really be useful, relevant information must propagate throughout the hand, and ideally, information learned in a given hand would be applied to subsequent hands.

**System Design:**

The system is designed using Microsoft Visual C# 2008. It consists of a GUI front end where all of the relevant information about the state of play is entered into the computer, and a dialog box is then used to dispense the commentary and advice that the program generates.

The structure of the program depends primarily on the card class. This class defines all the relevant information about a single card. This class is then extended into the deck class and the hand class. These classes define the current state of play, as all the player really knows is what he himself holds, and everything else is assumed to remain in the deck.

Further classes called pre-flopOdds and post-flopOdds are included, however pre-flopOdds is currently functional and post-flopOdds is not. All of these classes are used by a Coach class that weights the relevant information at each stage to generate a recommendation.

**Prototype and Implementation:**

**Evaluation Plan:**

Simply, does following our programs advice to help you win more hands of Texas Hold’em than you would on your own? A simple way of checking this would be to have a player play 5 games on their own, while keeping statistics on their play. The player would then play 5 games with our poker coach, following the poker coach’s advice on every play. If the poker coach increases the player winnings or places them higher in tournament play, after multiple experiments of this type, it can be judged a success.

To provide a move concrete metric of the success of the program, a concept called the Fundamental Theorem of Poker could be used. This principle was first articulated by poker theorist David Sklansky as the foundation of theoretical poker discussions. It 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.”*

To evaluate the performance of the poker coach, the fundamental theorem could be used, because it can be determined mathematically what the optimal play is at any point given the information about all the cards. The performance of the player can then be determined by defining the error as the deviation from the optimal play given all the information.