

# **Texas Hold'em Inference Bot Proposal**

By: Brian Mihok & Michael Terry

Date Due: Monday, April 11, 2005

# 1 Introduction

One of the key goals in Artificial Intelligence is to create cognitive systems that can perform human-competitive reasoning tasks. When a particular problem domain requires that inferences be made with limited information, an additional element of difficulty is introduced. In real-world problems, one typically does not have all of the information necessary to make a completely informed decision, and must either perform an action to receive that information, or make an educated guess regarding the best plan of action. A number of challenges arise when attempting to develop reasoning systems which emulate this type of intelligence.

One of the main challenges is formulating a well-defined problem and appropriate computational representation within which to perform experiments. Games have traditionally provided many problems in Computer Science with an excellent test-bench for new ideas, because they can be clearly described in terms of set rules and limited actions. Games of imperfect information (hidden state) provide models for an even broader range of problems, because they address the issues of uncertainty typically associated with most difficult inference tasks.

In addition, many games have expert strategists, which can provide additional insight into the design and evaluation of intelligent systems. As we will show in this paper, this incorporation of domain knowledge is an important theme amongst many techniques in Artificial Intelligence. Harnessing the knowledge accumulated by humans into the design process is often critical, and provides a great deal of benefit to the end-product.

A recent worldwide explosion in the popularity of Poker has stimulated interest in the game in many areas of Computer Science research. The most common form of poker, Texas Holdem, is a game of simple rules and complex strategies. As in most cards games, much of the state information is hidden to the player. In addition, because there are complex interactions between multiple adversaries, it provides a great application for reasoning methods associated with multi-agent environments. Finally, because it is a zero-sum game of chance (odds) and clear rewards (money), one could conceivably extend many of the concepts for reasoning in Holdem to disciplines such as Economics.

For this project, we will develop a system for making inferences in the game of Limit Texas Holdem using a number of advanced techniques. To address these tasks of reasoning under uncertainty, we will use a number

of probabilistic inference methods that fall under the umbrella of Statistical Learning. These techniques have been used in a number of problem domains, and demonstrate a great deal of success and recent advances in Computer Science research. In section 2 of this proposal, we give an overview of our two particular statistical inference methods of interest: Hidden Markov Models and Bayesian Nets. These methods provide a very intuitive, graphical way to represent information in a number of domains where modeling uncertainty is important. This section also describes additional work done in the area of Computer Science Poker research, primarily centered out of the University of Alberta. Section 3 provides a clear outline of the problem we have chosen to address. This is followed by Section 4, which presents the problem in more technical detail, and highlights a number of our ideas and methods for addressing the problem. Finally, Section 5 describes how we have chosen to organize and schedule this project.

## **2 Previous work**

### **2.1 Hidden Markov Models**

Two potential uses of Hidden Markov Models (HMMs) are present in this project. The first potential use is to use HMMs to model the progression of a hand. The opponent's hole cards are used to define the initial state of the model. The probability of receiving any two hole cards is a simple application of entry level probability, yielding a flat probability mass function (PMF) for all possible hands.

However, the model can be simplified by bucketing the possible hands. At the preflop stage of the game, it does not matter if the opponent has an ace of hearts and an ace of spades as opposed to an ace of diamonds and an ace of clubs. Instead, what matters is that the opponent has a pair of aces. Thus, in the preflop model, the suits of any pair are irrelevant. Similarly, the particular suit can be ignored for the remaining combinations. However, it is an important distinction in Hold 'em if the opponent has hole cards that are "suited," which means two cards of the same suit, versus "off-suited," which means the two cards are of different suits. Having suited cards provide the advantage of increasing the chance of obtaining a flush by the end of the game. As a result, the particular suit does not matter, but whether the cards are suited or off-suited is an important distinction to make. Thus,

the initial bucketing tracks pairs, suited non-pairs, and off-suited non-pairs, thereby substantially reducing the number of combinations, which limits the search space of the problem.

The probability of transitioning from any initial state to any other set of cards is again just an application of entry level probability and is known prior to any particular hand. This probability depends only on the known cards in play and the number of cards remaining from which to choose. Thus, the structure and transitional probabilities associated with the HMM are known before the game begins and never vary from hand to hand.

It is important to note that though the structure does not vary from hand to hand, it alters in shape throughout the course of any given hand. As mentioned before, the particular suits do not matter in the preflop stage of the game. However, after the flop hits, the suits of the opponent's hole cards are needed so that an evaluation with the community cards can be performed to see what type of poker hand he/she could potentially have. While this leads to an increased number of possible states, the number of states is reduced by continuing to bucket the potential outcomes into poker hands instead of modelling all the combinations that could occur with the ultimate 7 cards.

Furthermore, the model of the opponent is updated as new information about the state of the game becomes available. One such new information source is the appearance of the community cards. If a card show up in the community cards and only one deck is being used, as is the case for a standard Hold 'em game, then the card clearly cannot be one of the opponent's hole cards. As a result, any initial state containing any of the community cards can be eliminated as a possibility.

Another source of new information is the observations of the opponent, which are his/her bets. Again, the opponent's actions can be bucketed to limit the search space. The opponent's actions will be categorized as passive, normal, and aggressive.

As stated above, the structure and transitional probabilities associated with the HMM are known ahead of time and do not change. However, what does change from opponent to opponent, or even from hand to hand, is the observational probability, or the likelihood of the opponent taking a certain action at any state in a hand. The observations are the only information present that are directly related to what a particular opponent has for hole cards. As a result, the problem of inferring an opponent's hand is essentially one of conditional probability: given the current state of the game (the com-

munity cards and the opponent observations), what is the probability that the opponent has any particular pair of hole cards. This then becomes a problem of hypothesis testing. Each state can be evaluated for consistency with the opponent's previous actions and the community cards that are available. If the actions seem inconsistent with a particular hole card pair, then this pair is not a likely candidate for the opponent's cards.

The observational probabilities are clearly of vital importance to the inference process. The best way to populate the observational probabilities for each given state needs careful consideration if HMMs are to be used to model the game. One such way to determine the observational probabilities is through the use of Bayesian nets. This will be discussed at length in the next section. On a high level, however, Bayesian nets could be embedded in a particular state to determine the probability of an action given a number of discrete and continuous variables identified as important. The net could then output the probability of seeing any particular observation for a given state.

Another potential use for HMMs is to extract information from hand histories. The ability to use Bayesian nets or any other learning method depends upon having valid training data from which the method can learn. Hand histories, or databases of previously played games, exist on the internet. However, the format of these databases vary. The databases sometimes come as human-readable text, with extraneous information present. As a result, the information needs to be extracted and presented in a manner that is useful to the learning method. One potential solution to this problem is presented in Dayne Freitag's Information extraction with HMM structures learned by stochastic optimization. This paper discusses an alteration to HMMs to make the distinction between "target states" and "non-target states." The target states are used to extract the useful information from the document. The approach also divides the non-target states into prefix, suffix, and background states. These states provide the context for a given target, thereby leading to improved accuracy in the information extraction.

## 2.2 Bayesian Nets

Bayesian Nets, also known as belief networks, provide a powerful tool for making inferences under situation of uncertainty. They are a graphical method, like Hidden Markov Models, for representing quantities of information in the form of a connected graph. In our final report, we will present

a more detailed overview of these methods along with a survey of the most recent work done in this field of research. We also hope to incorporate the most advanced techniques and algorithms wherever applicable in our design.

### 3 Problem Statement

- To propose a concise and representation for state information in the Game of Texas Holdem Poker
- To use expert knowledge to generate candidate structures for these representations
- Use Statistical Learning to approximate parameters for the generated structures

### 4 Technical Approach

It is difficult to concisely describe a successful Holdem strategy, and the concept is highly debated amongst experts. However, there are critical pieces of information and inferences that must be factored into any winning strategy. A number of books have been published on the topic, and most experts place emphasis on concepts like odds, hand evaluation, opponent modeling, deception, and many others. Upon examining the breadth of research on poker today, most research groups are focusing on developing strategy without many of these critical underlying inference tools. In this project, we hope to implement a very general demonstration of an inference mechanism that will provide information about the most important variables in any situation that may arise in Limit Texas Holdem. From this, one can begin expanding the tools to include more sophisticated inference mechanisms and eventually work to develop a strategy. It is our firm belief that expert AI Poker systems cannot be built without establishing a foundation of these sub-goals and inference mechanisms first. Therefore, we will not be focusing on inferring strategy information.

To further break down the problem, we propose a Multi-Level Holdem Reasoning Model. This model is based upon the nature of the complexities of Holdem, and is mostly based on expert information from the domain. At the most basic level, Holdem is a game of inference regarding the strength of

a players own. In this model, we call this Level 0 Reasoning. The next level of reasoning, Level 1, involves making inferences regarding ones opponents hand strength based upon the betting patterns of that player and any other available information. Level 2 begins to introduce a lot of the complexity of Holdem, in attempting to infer what ones opponents may be thinking about your own hand. One can start to imagine the complexities introduced when a player begins to reason about Levels 3, 4 and higher. Experts are often considered to be reasoning about strategy at these higher levels . However, we would like to reduce the focus of our inference mechanism to Level 0 and 1 reasoning to emphasize the need for this foundation.

Our approach to developing this Level 0 and 1 reasoning system for Holdem can be divided into two distinct types of tasks. Our initial work will go into formulating and refining the models for representing all possible relevant information. Because we intend to use Bayesian Nets and Hidden Markov Models to perform inference, we will define all quantities of interest in terms of discrete and continuous variables. For example, the percentage of time a player performs an action raise can be quantified on the doman from 0 to 100

## 5 Project Organization

The project development will use the spiral method of management. The compressed timeline of the project only allows for the completion of two spirals. Three project plans are mapped to these two spirals. The plans include the minimal plan, the baseline plan, and the enhanced plan. The minimal plan is the minimum project needed to accomplish the learning goals. The baseline plan describes the amount of work that is expected to be done at the beginning of the project. Finally, the enhanced plan details the ideal project, which includes all the project goals that would be implemented given an sufficient amount of time and energy. The following two sections detail the goals of the two spirals and the schedule for the project.

### 5.1 Spiral 1

The goal of the first spiral is to implement the minimal plan. On a high level, this involves parsing hand histories, representing the progression of the hand through a variety of states based upon known information and observations,

and then returning a PMF after the last observation (after the river card). This PMF will then be compared to the actual result to determine the accuracy of the prediction, which will then be used to further train the model. The following paragraphs break down the high-level components into more detail.

Hand histories serve as input from which the bot can learn. A variety of hand history sources exist online, including a database provided by the University of Alberta. As a result, the bot does not initially have to analyze real-time hands, but instead can be limited to previously played hands. This eliminates the need for a human-computer interface, but instead only necessitates that the bot be able to parse information out of a database or file. This then becomes an information extraction problem, as discussed above.

The structure of hand histories depend upon the source. Some of the sources do, however, contain human-readable information, such as player's chatting back and forth with each other, that is not important to the bot learning to predict the opponent's cards. This then dictates a sparse extraction of information from human-readable text files, such as those described in the previously mentioned paper. Depending upon the hand history source selected, HMMs could be used to extract the relevant information.

A great deal of evaluation is required to figure out the variables necessary to accurately model an opponent's actions. Obvious factors include the strength of the opponent's cards and the amount of money that he/she possesses. However, other important factors include (but are not limited to) recent win/loss streaks, time of day, and how long the opponent has been playing. Some of this information can easily be extracted directly out of the hand history. A great deal of it is not simply obtained, thereby necessitating that methods be developed to produce this information.

The information extracted from the hand histories can then be used as input into the bot. The best representation for game and the learning method used by the bot need to be determined. Options are detailed in the previous work section. Part of the first spiral will be dedicated to experimentation with different methods to see which ones best fit the project's demands for accuracy while still operating in a compressed time-scale. For further details on the schedule, see the 5.3 subsection below.

Finally, the bot will calculate and return the PMF for an opponent's hole cards. Instead of being tailored to an individual person, however, the PMF calculation will be based upon an average player in the minimum plan. The tendency of an average player will be obtained through the evaluation of the



information extracted from the hand histories without regard to the player involved. This is done for simplicity and to model what any given person would do at a particular table. Though not specific, this information is still quite useful, as it serves as a starting point for modelling new players at a table for which no prior information exists.

A further simplification for the minimal plan is that the PMF will be calculated after the last observation of the game. This observation occurs during the betting round after the river card and before the showdown. At this point in the game, seven cards are known (the user's two hole cards and the five community cards) in addition to the string of observations generated by the opponent in the form of the betting rounds. This is the maximum amount of information that can be obtained in a given hand of Hold 'em, thereby simplifying the inference about the opponent's hole cards as much as possible. This also eliminates all the potential hands in the hand history that did not go to showdown. A further benefit other than maximizing the information is that when the showdown occurs, all parties still involved in the hand must show their cards. This then serves as a way to validate the model.

In summary, the first spiral of the project will first determine what the variables necessary to model an opponent's hole cards are, as well as the best way to extract these variables. The information will then be fed into the bot. The bot's representation of the game and its learning method still need to be examined and selected. With this done, the bot will then generate a PMF of the opponent's hole cards after the betting following the river card has occurred. The actual cards of the opponent will then be used to validate the model.

## 5.2 Spiral 2

The main thrust of the second spiral will be to implement the baseline plan. If any time is left over upon the completion of the baseline plan, then the tasks associated with the enhanced plan will be addressed.

Fundamentally, the baseline plan extends the work of the minimal plan to include the beginnings of an opponent-specific model. During the first 50 hands of information for a new opponent, the average player model developed in the first spiral will be used to predict the opponent's cards. After the 50th hand, the opponent's play will be evaluated, producing a classification of the opponent into one of several predefined categories. Such categories are

common in poker literature and include terms such as "rock," which is an opponent who plays extremely conservatively, and "calling station," which is an opponent that checks most of his/her bets as opposed to folding or raising.

This classification will be based upon Mike's expert knowledge of Hold 'em. His knowledge of the game can be used to create rules that can be used to rate players depending on how they acted in certain situations. Examples of key criteria include preflop aggression, amount of money put into the pot, the number of big bets per hour, and the percentage of flops seen. The benefit of categorizing the players is that it allows for a player-type specific model to be developed. As in any game, Texas Hold 'em is played by people with a wide variety of skill levels and sometimes quite contrasting strategies. While an average player model is useful when no other information is present, it is not as useful as models more tailored to an individual's tendencies. A player-type specific model attempts to give increased specificity for a given player without requiring an individual model for each opponent.

If time permits, the enhanced plan aims to advance this baseline plan further by attempting to make the bot useful in real-world style games. To do this, the bot will need to perform the calculations to generate the PMF at each stage of the game, not just after the last observation, and in real-time, which usually amounts to at most 20 seconds. Furthermore, the bot could potentially be made to interface with poker servers such as those offered by the University of Alberta. This would serve as a great way to test the bot with live data. Finally, the bot could be extended to model each opponent individually. These are extremely ambitious goals which almost certainly could not be implemented in 5 weeks, even if this was the authors' only commitment. However, conceptualizing such extensions can potentially help shape the design of the minimal and baseline bot.

### 5.3 Schedule

The authors plan to divide up the work required for the project as equitably as possible. Instead of having experts in one particular section of the project, the authors wish to work together so as to alleviate the frustration of attempting to debug a major section of code alone. However, given the division in the similarly-themed advanced lecture, it would seem natural for Michael to take the lead on the Bayesian net section while Brian could take the lead on the HMMs. These roles will be evaluated and adjusted constantly

through the project.

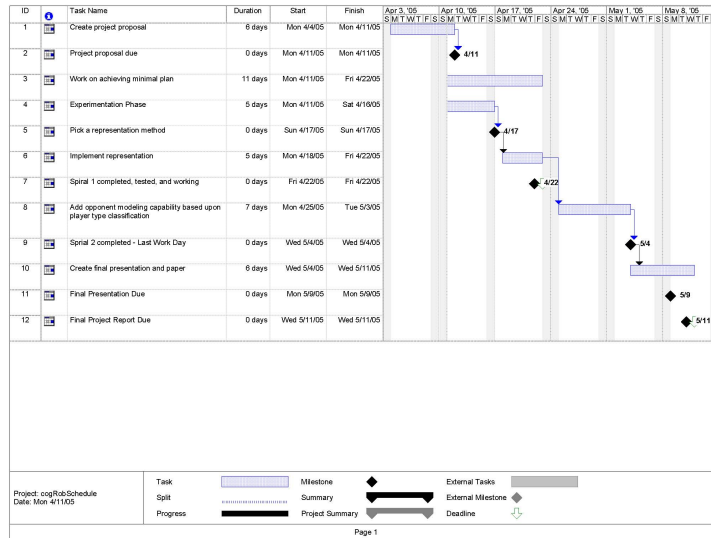


Figure 1: Schedule for the proposed project

Figure 1 shows the schedule for the proposed project. The first spiral is scheduled to last for the first two workweeks of the project. This is subdivided into an experimentation phase, where the different representations and learning methods are evaluated, and an actual implementation phase. The first spiral is scheduled to complete on Friday, April 22. From there, the project then moves into the second spiral. The authors want to leave at least one week to write the final paper and work on the final presentation. As a result, the last work day is scheduled for Wednesday, May 4, thereby allowing an uninterrupted week to work on the final deliverables.