Creating a Hand Solver for the Card Game Bridge

Nikhil Alladi | Computer Systems Lab under Dr. Yilmaz

Introduction

Bridge is a highly dynamic card game involving 13 tricks of 1-card plays across 4 players, where the highest card in the suit that the first played played wins the trick. The game revolves around a “trump suit” that overpowers the main suit of a trick. Modeling the game is notoriously difficult due to its lack of information (2-3 unknown hands totaling to nearly 40 unknown cards). In this project, I present a solution to model hands based off of the current game state and bid information, to attempt a solution to the unknown information issue that limits most bridge solvers.

Methods

I used a heuristic that simulated hands based off of the bids that players made in the bidding stage of the game. The heuristic takes into account the bids that were made across all players and the order they were made in, the amount of repeat bids that each player had throughout the bidding session, how bids worked across partners and the shared weights of their hands, and the played cards in the game to inform its weights [Figure 1]. I then accentuated this hand knowledge with game state constraints (played cards, suit densities, etc.) to inform a guess on the rest of the hand. This guess is then fed into a Double Dummy Solver [Figure 2] to evaluate the next possible move for the state.

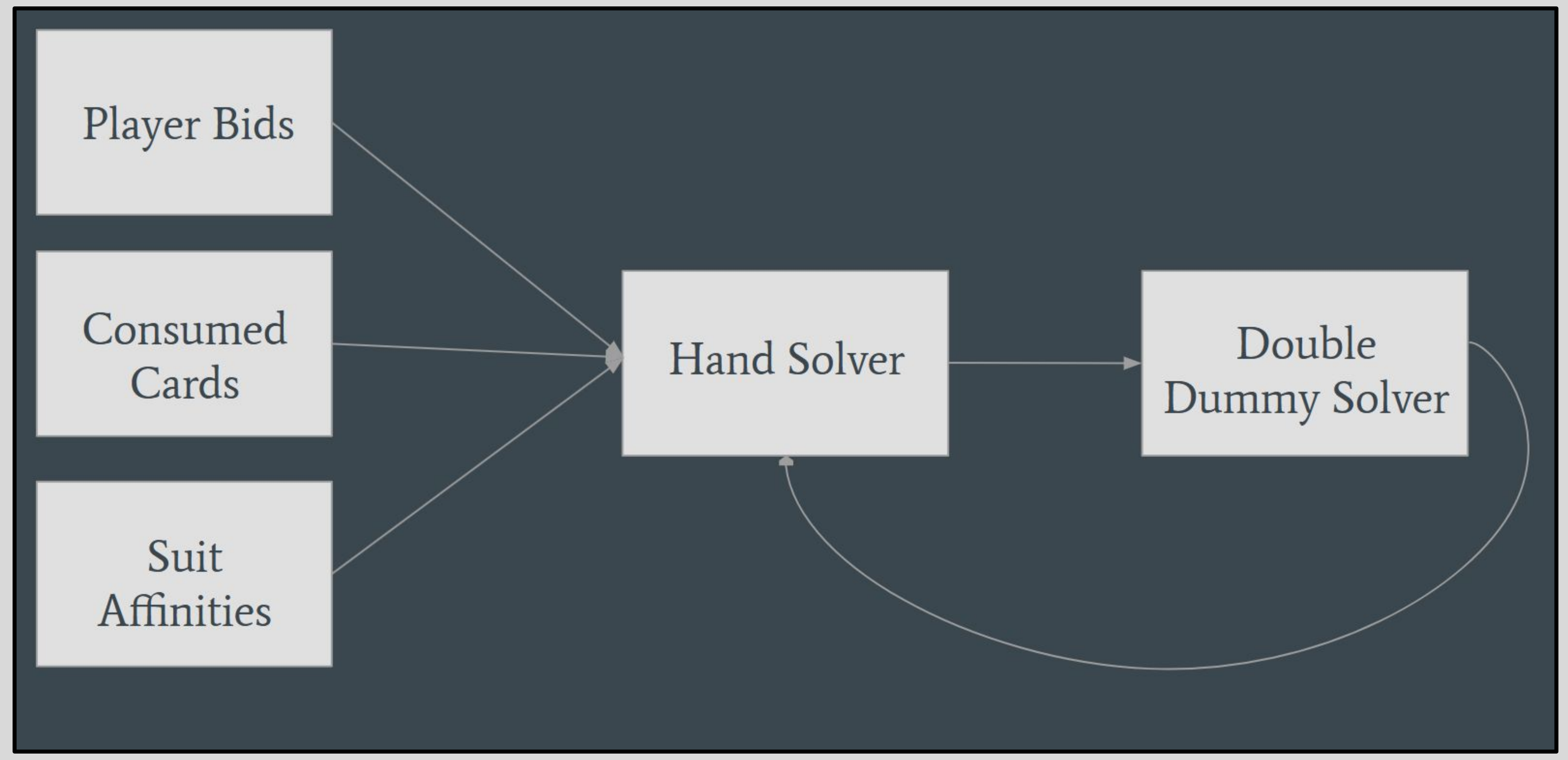

Figure 1: A high-level overview of the systems architecture for the project. Bids are declared, then in the process of evaluating hands, a hand estimator is used given the game state and bids to evaluate a hand which is then fed into the Double Dummy Solver to solve that hand.

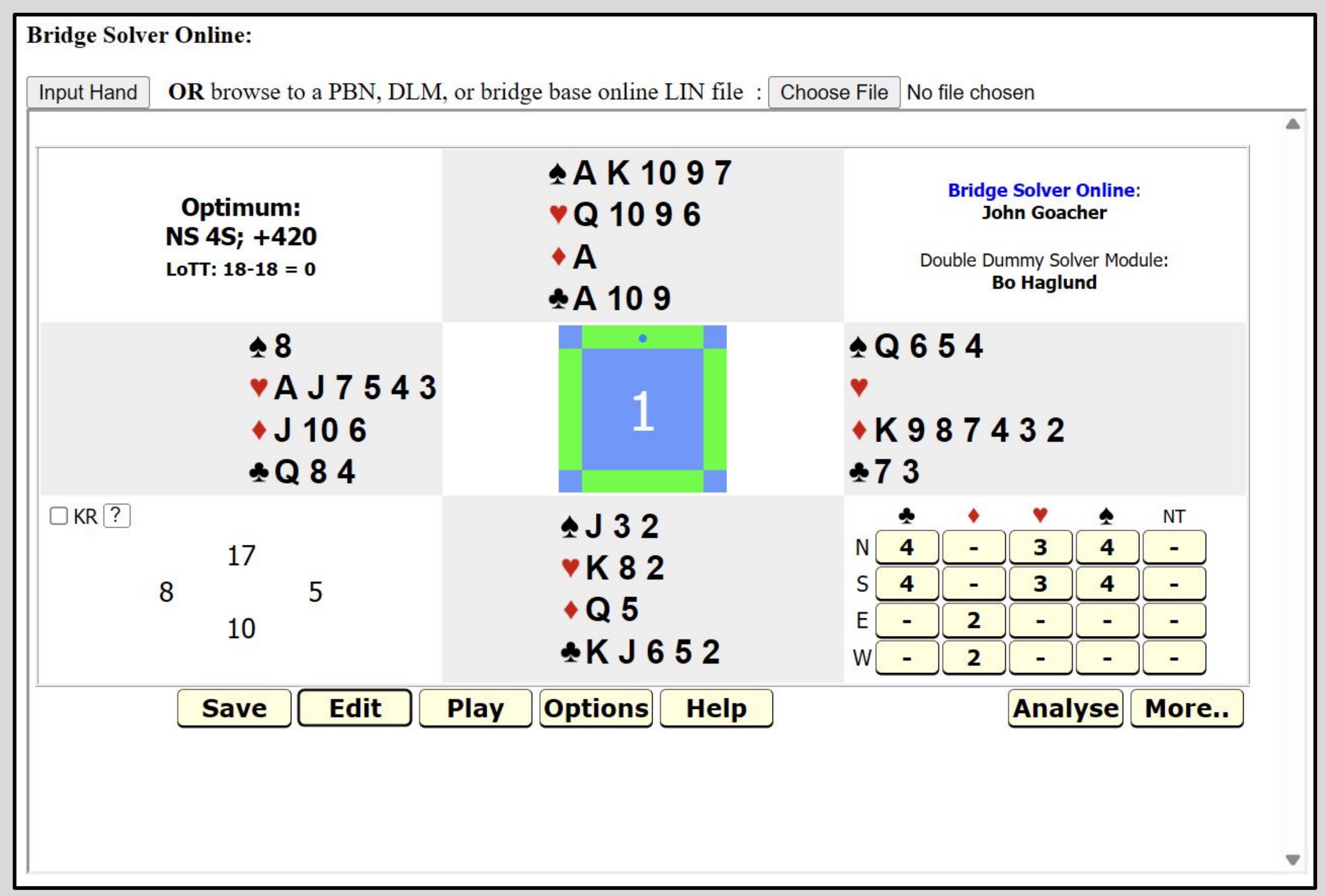

Figure 2: Example view of a double dummy solver using the Bridge Solver Online module. Knowing the game state is required to make a DDS work, but if estimated, a DDS becomes a powerful tool to evaluate a game from a state.

Results

In about 80% of cases, the hand solver is able to accurately predict the state needed to win the game to the trick level provided. In all cases, the hand solver was able to provide moves that resulted in making all but one of the tricks that the hand was able to make under perfect play. The hand solver predicts pessimistic outcomes, due to the hand solver weighting. When ambiguities are faced over powerful cards, it tended to give those cards to the opponents to create a harder scenario. This would be more likely to make the expected trick amount higher in practical play.

Conclusion

Bridge is a uniquely difficult card game to model due to its incomplete information problem and lack of gaps to fill in that information. This project attempted to resolve that by predicting the hands using the information we did have to simulate a complete information state from where we could perform standard operations to evaluate the best move from that full information state, later applying it to the incomplete state we started with. While it did not have perfect efficacy, it provided a powerful replication of the state that was able to provide insight into the real game at a high level of play.