**Environment Implementation:**

There are three possible states: Check or Bet (Bet is the first raise done at each round) , Call Raise or Fold and Call ,Fold. The state of the player is decided by the last action the opponent has done. In the beginning of a game, player 1 always starts with check or bet available.

A picture containing text, screenshot, font, design

Description automatically generated

The environment keeps track of the actions of both players, the bets that have been done, the available cards and updates the available state for each player .Playing , Round 1 starts with only players cards revealed to them. Player 1 takes an action Check or Bet which will decide if player 2 will be in state Check or Bet or in state Call Raise Fold. If a bet is done by player 1 player 2 can raise also leading player 1 in state Call or Fold. If player 2 calls or checks back we move on to the next round where the public cards are revealed, and players can bet again. A game ends when a player folds or a call is done in round 2 by any player. Judger, at the end of a game a judger is called which checks both players’ hands and public hands, in order to decide the winner. An amount of chips is returned to its player as a reward , positive if he won , negative if he lost , or 0 if it was a draw.

Example of the environment on 0.0:

A screenshot of a computer screen

Description automatically generated with medium confidence

**Policy Iteration:**

The dictionary implemented for the policy iteration follows 11 states for player 1 and 9 states for player 2. Each state has the 5 available actions Check, Bet, Call, Raise, Fold with the unavailable actions at each state marked with a huge negative reward. In detail for player 1 there are 2 possible states Check or Bet and Call or Fold. But to keep track of the rewards we have multiple Check or Bet states with same transition probs but different rewards and states to go. Transition probabilities are given by a function which calculates the probability of the opponent taking a certain action in case of Threshold opponent or there are uniformly distributed in case of a Random opponent. The rewards are given when there is an action leading to a terminating state win or lose. The reward is equal to the amount of chips each player has bet in a certain game. In 0.0 example player 1 has to decide to fold leading to -2.5 reward which is the number of chips he has contributed so far to the game. Transition probabilities leading to terminating states are given by a function which calculates the possibility of winning taking into consideration the agent’s hand and public cards too. In order to encourage the agent to take action at the first round, which does not lead to a terminating state a small reward is given too. There are 2 opponent types, one Threshold which bets/calls with K or A in the first round and a pair on the second round and one Random who randomly chooses an action. The environment is different for each opponent leading to separate training.

**Running 10000 episodes**

In each plot we can observe the policy agent’s cumulative reward playing against a certain opponent compared to a Random Agent playing with the opponent. Testing is done for 10000 games.

Agent playing first against a Random agent, Agent playing second against a Random agent.

A picture containing text, line, plot, screenshot

Description automatically generated A picture containing text, line, plot, screenshot

Description automatically generated

Average Reward is 0.50 Average Reward is 0.95

In both cases we prefer a gamma close to 0.5 which strikes a balance between present reward and later stages , and an epsilon close to 0.05 which converges faster giving a decent average reward.

In player’s 2 case agent plays kind optimally, he basically chooses to bet or raise most of the time having a good hand like A or K and sometimes checks with Q J T. In most cases agent bets or raises which is expected because of the high probability of player 1 folding with 50% chance or losing as he calls randomly with occasionally having a good hand. On hands like T J Q, he chooses to check instead of betting with lower chances to win on showdown. Average reward is also expected to be between 0.5 and 1.5 as player 1 will check or bet randomly and player 2 will bet or raise leading to player 1 folding with reward 0.5 or 1.5. In 10000 episodes the agent won 8091 times and 5431 of them with a reward of 0.5 or 1.5 with e percentage of 0.67.

In player’s 1 case agent chooses most of the time to bet or call again with higher card like K or A, but chooses to fold when raised with lower value cards like T or J and no big bets have been placed yet meaning he prefers not to call with a great probability of losing risking for a low amount of chips which makes sense. Average reward is also expected as in most cases agent will bets and opponet will fold winning 0.5 chips. In 10000 episodes the agent won 7259 times and 3259 of them were a 0.5 reward with a percentage of 0.449 which means player 1 bet and player 2 folded.

A screenshot of a game

Description automatically generated with medium confidenceA screenshot of a computer screen

Description automatically generated with low confidenceA screenshot of a card game

Description automatically generated with medium confidence