

Technical University of Crete

Reinforcement Learning & Dynamic Optimization - plh423

Simple Poker Game

Presented By:

Georgios Skoulas 2018030148

Ioannis Peridis 2018030069



A black and white photograph of a poker table. In the center, three playing cards are laid out: an Ace of Spades, a King of Spades, and a Ten of Clubs. Surrounding the cards are numerous poker chips of various denominations and designs. A small stack of cards is visible in the top right corner. The table has a dark, textured surface.

Environment

→ State Space

Deck: 20 cards : T,J,Q,K,A & 4 suits

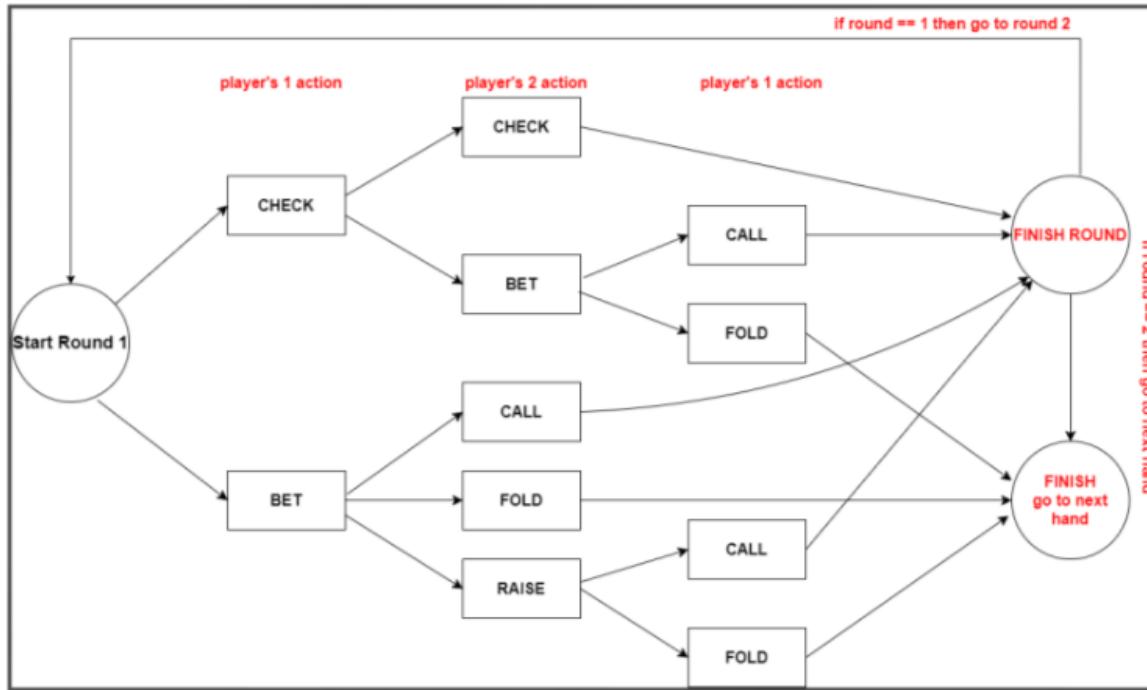
Player Hand: 1 card, Board: 2 cards

Blinds: 0,5 mandatory, Stack: 20 blinds

→ Actions

Fold, Check, Call, Bet, Raise

Actions Flow Diagram representing one game hand:



→ Heroes

Policy Iteration

Q-Learning

→ Villains

Random

Threshold Loose (aggressive)

Threshold Tight (passive)

Poker as a Markov Decision Process

Actions: (0) fold, (1) check, (2) call, (3) bet, (4) raise

States: divided by round 1 or 2 and by hand strength

Total 21 states:

$S(0-4)$: round 1 states: $T=s_0$, $J=s_1$, $Q=s_2$, $K=s_3$, $A=s_4$

$S(5-19)$: round 2 states: high card= s_5-s_9 , one pair= $s_{10}-s_{14}$, set = $s_{15}-s_{19}$

S_{20} = terminal state

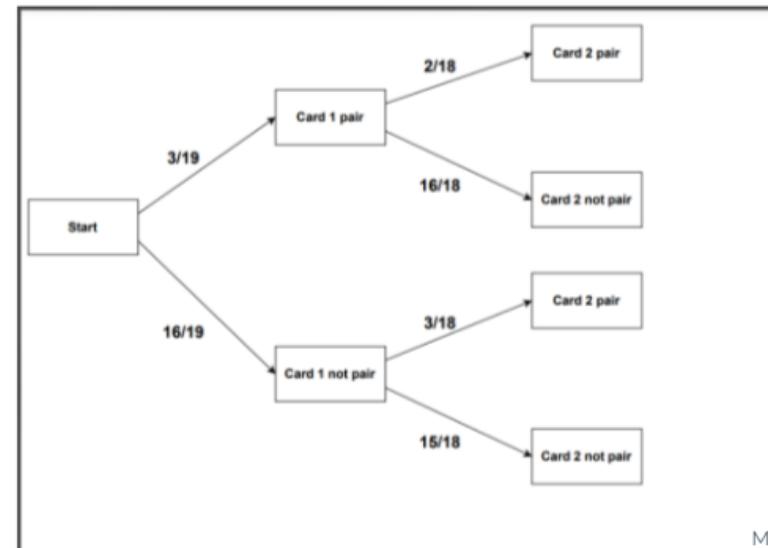
Transition Probabilities:

From S_{0-4} to S_{5-9} : $P=0,668$

From S_{0-4} to S_{10-14} : $P=0,315$

From S_{0-4} to S_{15-19} : $P=0,017$

From S_{5-19} to S_{20} : $P=1$



A black and white photograph of a poker table. In the center, three cards are laid out: an Ace of Spades, a King of Spades, and a Queen of Clubs. Stacks of poker chips are scattered around the cards. A dark, semi-transparent rectangular box is positioned over the top right corner of the cards, containing the text.

Policy Iteration

Strategy vs Opponents:

→ vs Threshold Loose
Play high strength hands
Tight/Passive actions
Win few big pots

→ vs Threshold Tight/Random
Play all hands
Aggressive actions/Bluffs
Win many small pots

Convergence:

Converges when new policy equals to old policy
Convergence after 2-3 iterations

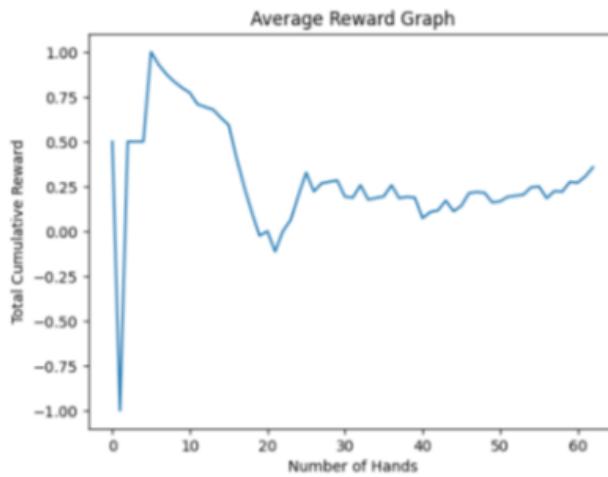
Optimality:

Optimal solution occurs from the reward system
PI agent decisions match the optimal policy

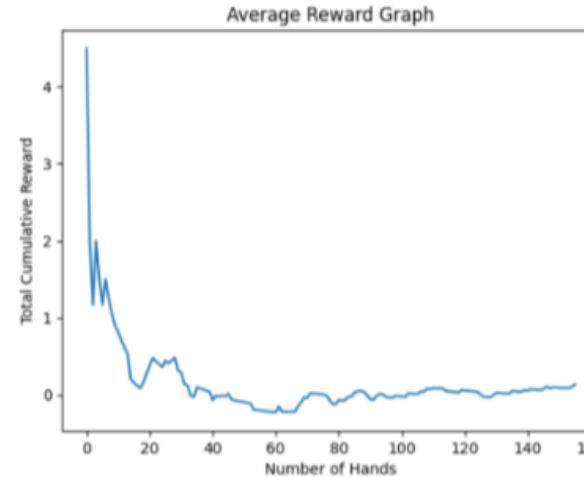
Parameters: discount factor (γ) = 0.98 convergence error (ϵ) = 1e-10

Average Reward vs Opponents

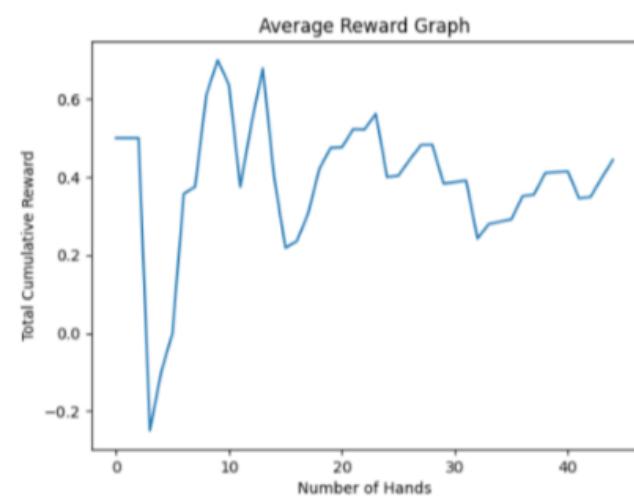
vs Random:



vs Threshold_Loose:



vs Threshold_Tight:



Experimental Results (10 games)

Win ratio:	Average #Hands:	Avg.Avg Blinds/Hand ratio:	Win ratio:	Average #Hands:	Avg.Avg Blinds/Hand ratio:	Win ratio:	Average #Hands:	Avg.Avg Blinds/Hand ratio:
100%	42,4	0,64	100%	73,3	0,49	100%	33	0,65

A black and white photograph of a poker table. In the center, three playing cards are laid out: an Ace of Spades, a King of Spades, and a Nine of Clubs. Stacks of poker chips are scattered around the cards. A stack of white and black checkered chips is prominent on the right. In the top center, a dark rectangular overlay contains the text "Q-Learning" in a yellow, italicized, sans-serif font.

Q-Learning

Parameters

discount factor (γ) = 0.9

learning rate (α) = 0.1

exploration rate (ϵ) = $\text{episode}^{-1/4}$ reduces over time

number of episodes = 20000

Convergence:

Convergence time is different against each opponent

Most time: vs threshold loose

Probably because we lose from threshold loose

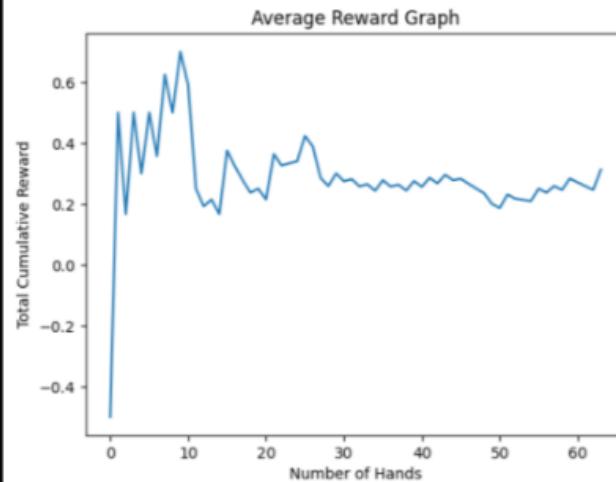
Optimality:

Tuning the parameters in different ways gives different results

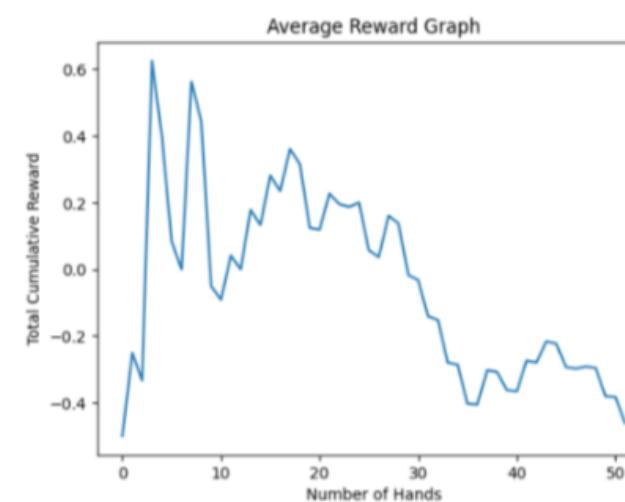
By experimenting we chose parameters that gave the best results

Average Reward vs Opponents

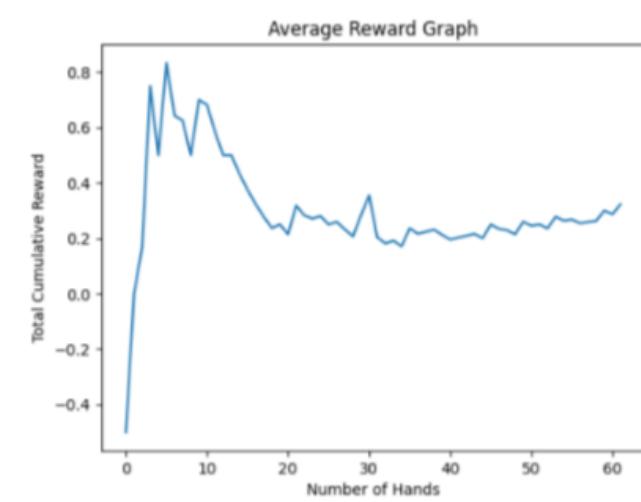
vs Random:



vs Threshold_Loose:



vs Threshold_Tight:



Experimental Results (10 games)

Win ratio:	Average #Hands:	Avg.Avg Blinds/Hand ratio:	Win ratio:	Average #Hands:	Avg.Avg Blinds/Hand ratio:	Win ratio:	Average #Hands:	Avg.Avg Blinds/Hand ratio:
100%	82,5	0,27	0%	40,6	negative0,5	90%	97,5	0,22

A black and white photograph of a poker table. In the center, three playing cards are laid out: an Ace of Spades, a King of Spades, and a Two of Clubs. The table is covered with a dark, textured cloth. Numerous poker chips are scattered around the cards, with larger stacks on the left and right sides. A deck of cards is visible in the bottom right corner.

Problem

Q-Learning vs threshold loose

The agent loses to the threshold-loose bot

We could not find if there was some problem
with the code

We assume maybe that the state space was not
big enough



THANK YOU