CS747

Assignment #2
Report

Sibasis Nayak (190050115)

Department of Computer Science and Engineering
Indian Institute of Technology Bombay

2021-2022

Task 1

In `planner.py` the initial values have been initialised to a very large negative value wherever required. For pi in episodic states we output 0 as the default action. We break the ties with equal valued actions by choosing the valid action that has numerically smallest index. (basically doing `argmax`)

Task 2

The MDP is formulated by the following. We take all the states from the states file, and also add a terminal state T. We say a particular state as T, if the game has ended with a draw/win/loose in that state.

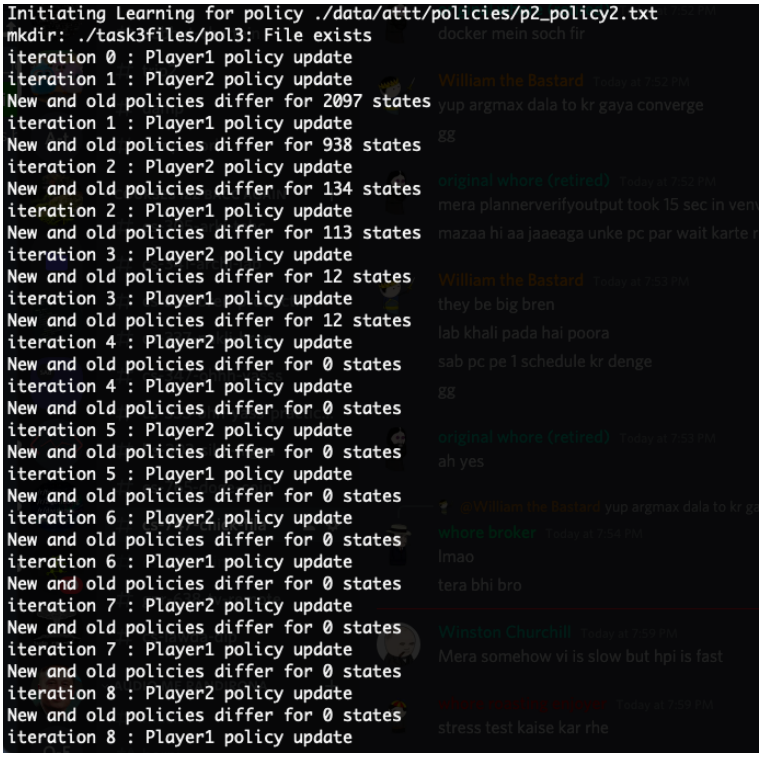
For a give transition let's say we have current state s and we play a move by taking action a, the new state generated x must be a valid state of the opponent. Now we have the opponent's policy with us. So we will take the opponent's policy and and check to which states of player can the opponent move using it's policy. Let's say for state x we have actions $ai1, ai2, ...$ with transitions to state $si1, si2...$ with non zero probability. This means that our transition function of the MDP will have $Tr(s, a, s') = Pr(x, a')$ where x reaches s' by taking action a'.

For rewards, we define all transitions to non terminal state to have reward 0. For transition to terminal state, we define the reward to be 0 if the action lead to a draw/loosing and reward to be 1 if the action leads to a win for the player.

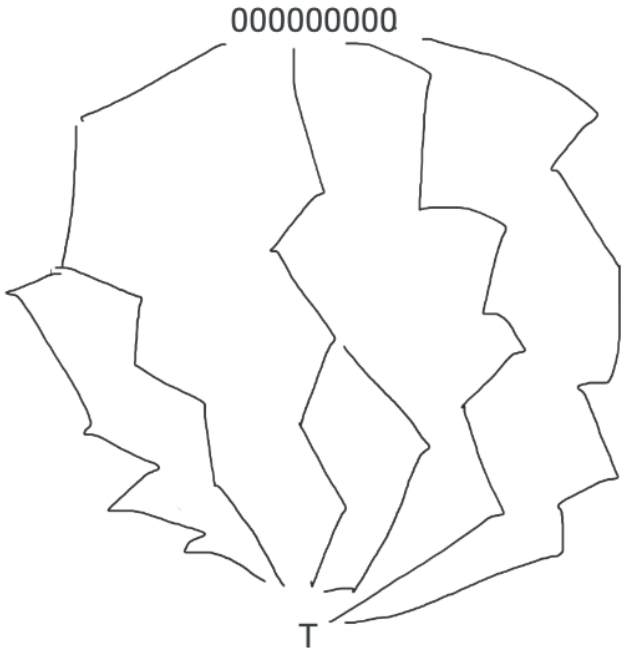
To completely define the transition and reward function we take all invalid actions and make a transition from the current state to the terminal state with reward -1 (a negative reward so that the action will never be considered to calculate the optimal policy).

Task 3

We see from the above that the policies generated converge after a certain iterations.



The proof of the above can be seen in this way. In each stage of our game we will have a state corresponding to player $P1$ or $P2$, and he has to choose an action. There will be 9 such stages of actions with the game starting at $P1$, and we will obtain states pertaining to a particular player in alternating stages. We can draw a graph of such states with edges corresponding to valid actions that a player can take. The graph will have a source (state 000000000) where the game starts, and a sink T where the game ends. The graph will look somewhat like this with all edges downwards, there would not be any loops (since game is progressive).



We also observe that any path from state 000000000 to T has atmost 9 vertices in between them. We also observe that at a particular level(i.e states that are at a particular distance from 000000000) will correspond to a particular player. Also observe that consecutive levels (except T) correspond to states of different players. Now we know that every such path will have a fixed result at the end either P1 wins/ P2 wins/ Draws. We also observe that there are a finite number of paths from any given state to the terminal state.

Now let our players have any particular policies π_1 and π_2. Let by combining this policies we traverse from a particular state s(which is a from the current improvising player) in the above graph and we reach a particular conclusion. Let it be a player P's chance of improving the policy (it can be anyone arbitrarily). Let s be such a state that the current path with the combined policy is resulting in a draw/loss of P and no player cannot improvise further by changing any action of a state that is present in the current taken path (i.e s is the closest such state). The existence of such a state can be shown because we have the results of each path as fixed. So from s if any other action is present currently that favors P i.e P ends up winning, will be taken as per the definition of finding the optimal policy. But we see that the branch that we excluded will never be taken again because no further improvement was possible. In the next iteration our opponent cannot change our action because it is not a state that it can interfere with and since our changing branch state was the closest such state to terminal, our opponent cannot improvise in lower states. It has the chances of improvising an upper state which has a path through s only if he changes the action in the upper state itself, such that it does not come to s. If any such action is found we see for all upper states changed we don't encounter s again. So the combined policy has reduced the branches. So either it never happens again or there is no change. For equally improvising actions, or if no such successful branching actions were possible

then we have the same action as per the new policy too, since our tie breaking rule is consistent.

Since we have a finite number of branches and our improvising actions result in a branched out action never getting chosen again the policies together will finally converge to fixed actions for each state.

We can also observe another property of the convergent policy that it does not depend upon the initial policy provided. The final policies will be the strongest players ever no matter how well we initialise them to play.