Programming Assignment 2: CS 747

Md Kamran

Due: 11 October 2021

Contents

Task 1

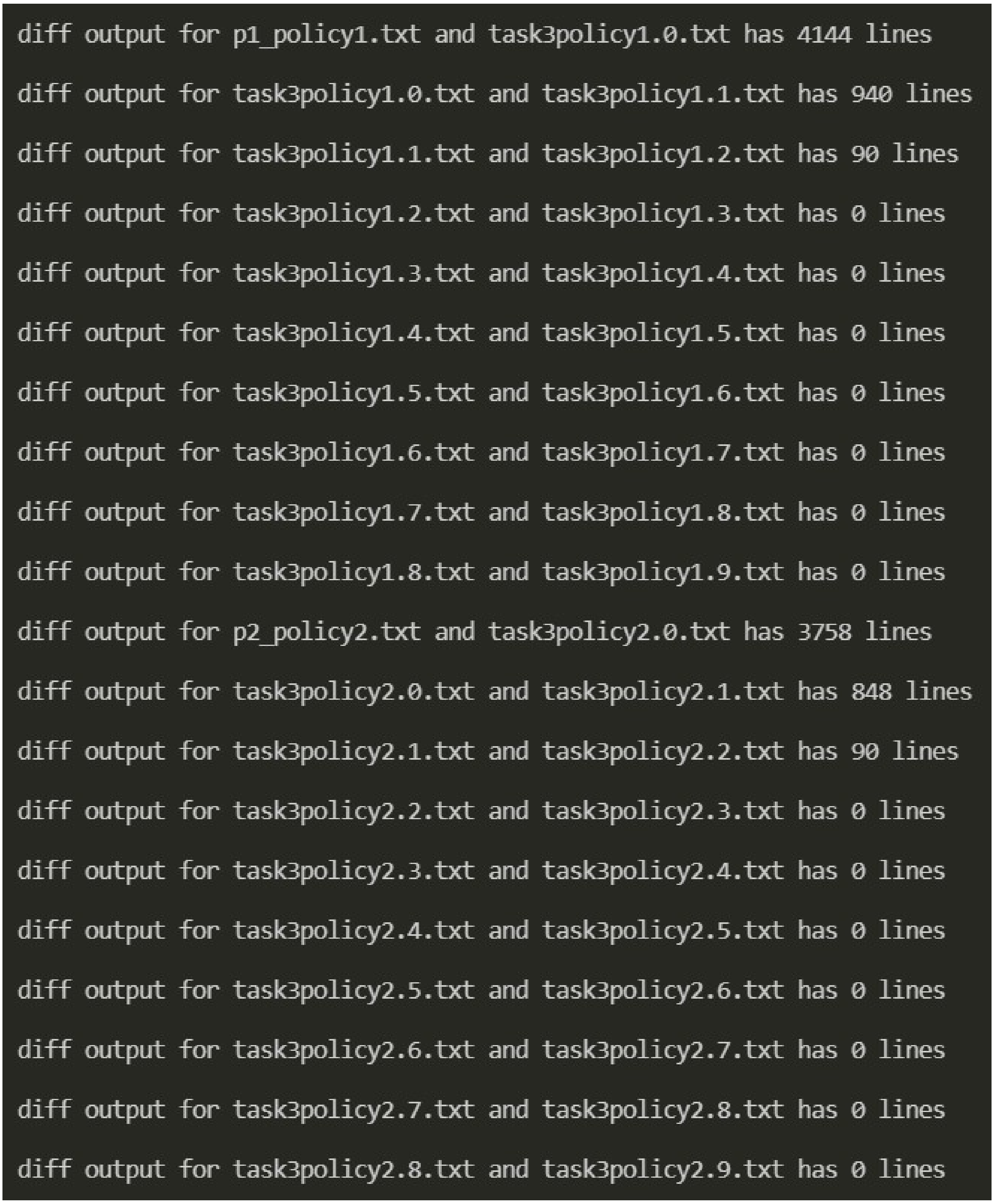
For choosing the default action to start with for each state, I chose the action with smallest index available for that state.
Terminals states were included while doing Linear Programming formulation and solving the system of Linear Equations but were excluded while computing value function and action value function because their value will always be zero whatsoever.
In value iteration, the convergence criteria was set to the difference between successive value function vector being $1e-7$. The difference between successive value function vector is calculated by summing up the absolute difference between each vector element.
In Howard policy iteration, the convergence criteria was set to two successive policies being exactly the same.
In linear programming and value iteration, after computing the value function, the policies were calculated by doing an argmax over the action value functions. And in Howard policy iteration, after computing the optmial policy, the value function was computed by solving the Bellman equations.

Task 2

MDP formulation:-
The number of states I have taken is equal to the number of states provided in the statesfile plus 1 (as an end state).
The total number of possible actions will always be 9.
The end state only has the last state which is declared as the end state apart from the states given in the statesfile.
The mdptype is chosen to be episodic as it has an end state.
Discount factor is chosen to be 1.
Now in the transitions part, first a state (except end state) is chosen from where the transition will take place. Then possible actions will be all the places which has a zero. Then for every state action pair, the state is updated by replacing the zero at action place to be equal to the agent's player id. Then if the new state formed is not present in the rival player's list of states, it means the agent has lost. So the final state of transition is set to the end state with reward 0 and probability 1. Else it's now the rival player's (environment's) turn with possible actions from 1 to 9 (0 to 8 in the code). So in environment's policy file, if the action has 0 probability then it is not included. Else for every state action pair, we update the state by replacing the zero at action place to be equal to the rival's player id. Then if the new state formed is not present in the agent's list of states, it means the agent has won. So the final state of transition is set to the end state with reward 1 and probability equal to the environment's probability of taking that action. Else the transition takes place to the next state with reward 0 and probability equal to the environment's probability of taking that action.
Finally in decoder while creating the policy file, the best action found by the planner for every state is set to 1 and other actions are set to 0.

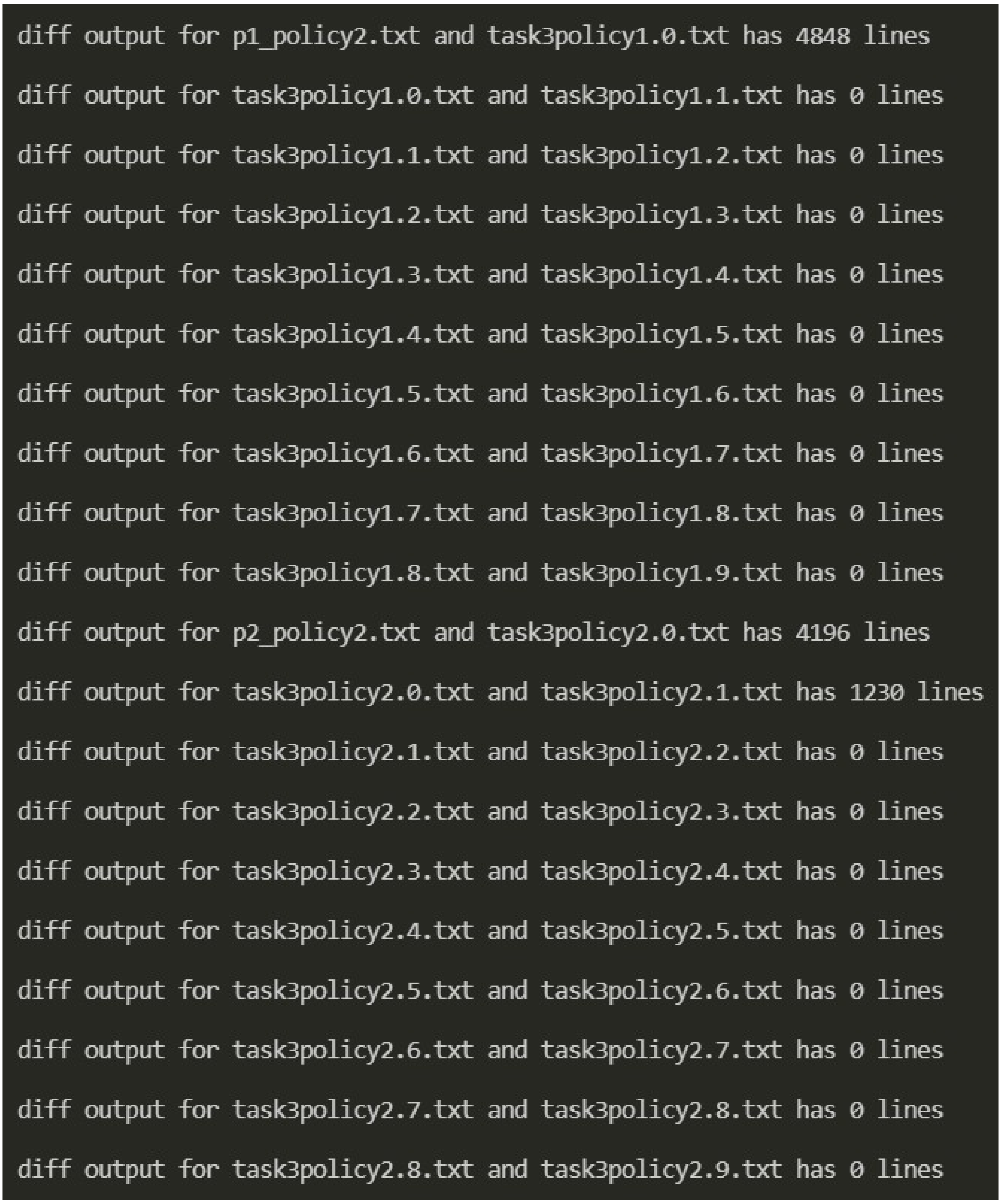
Task 3

Yes, the sequence of policies generated for each player is guaranteed to converge.



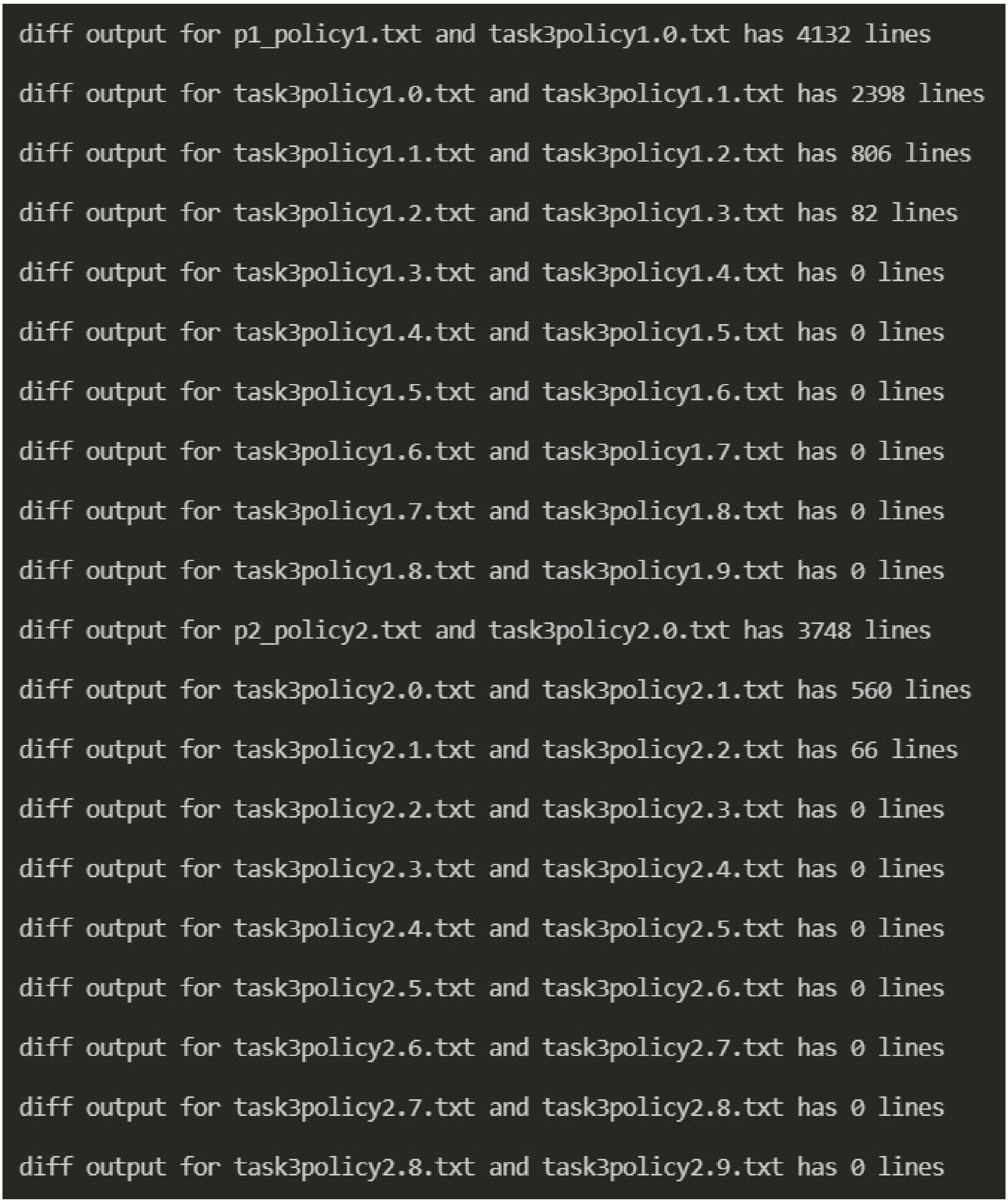
```
diff output for p1_policy1.txt and task3policy1.0.txt has 4144 lines

diff output for task3policy1.0.txt and task3policy1.1.txt has 940 lines

diff output for task3policy1.1.txt and task3policy1.2.txt has 90 lines

diff output for task3policy1.2.txt and task3policy1.3.txt has 0 lines

diff output for task3policy1.3.txt and task3policy1.4.txt has 0 lines

diff output for task3policy1.4.txt and task3policy1.5.txt has 0 lines

diff output for task3policy1.5.txt and task3policy1.6.txt has 0 lines

diff output for task3policy1.6.txt and task3policy1.7.txt has 0 lines

diff output for task3policy1.7.txt and task3policy1.8.txt has 0 lines

diff output for task3policy1.8.txt and task3policy1.9.txt has 0 lines

diff output for p2_policy2.txt and task3policy2.0.txt has 3758 lines

diff output for task3policy2.0.txt and task3policy2.1.txt has 848 lines

diff output for task3policy2.1.txt and task3policy2.2.txt has 90 lines

diff output for task3policy2.2.txt and task3policy2.3.txt has 0 lines

diff output for task3policy2.3.txt and task3policy2.4.txt has 0 lines

diff output for task3policy2.4.txt and task3policy2.5.txt has 0 lines

diff output for task3policy2.5.txt and task3policy2.6.txt has 0 lines

diff output for task3policy2.6.txt and task3policy2.7.txt has 0 lines

diff output for task3policy2.7.txt and task3policy2.8.txt has 0 lines

diff output for task3policy2.8.txt and task3policy2.9.txt has 0 lines
```

Figure 1: diff output for 10 successive policy files for both players when starting with p1 policy 1



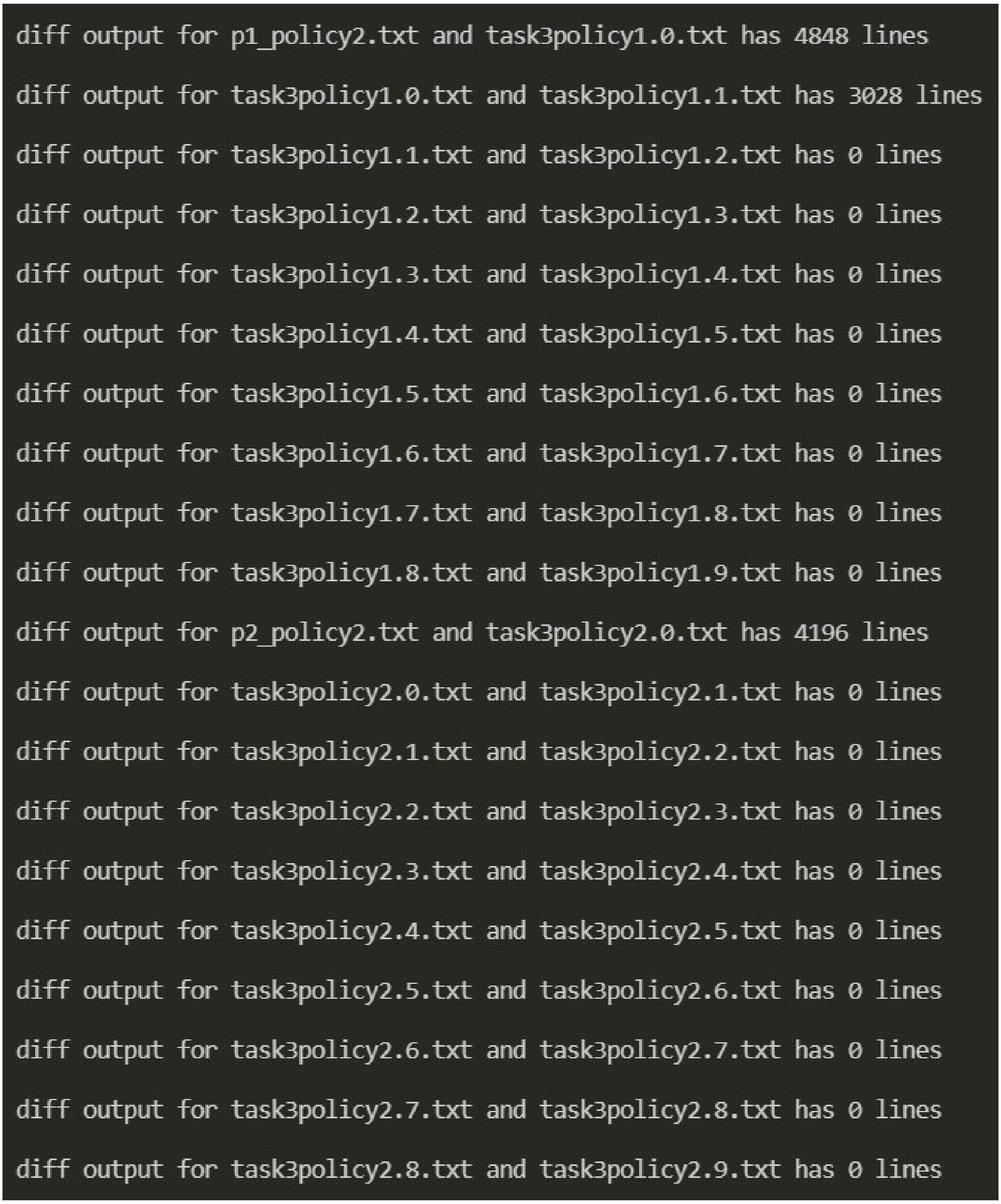
```
diff output for p1_policy2.txt and task3policy1.0.txt has 4848 lines

diff output for task3policy1.0.txt and task3policy1.1.txt has 0 lines

diff output for task3policy1.1.txt and task3policy1.2.txt has 0 lines

diff output for task3policy1.2.txt and task3policy1.3.txt has 0 lines

diff output for task3policy1.3.txt and task3policy1.4.txt has 0 lines

diff output for task3policy1.4.txt and task3policy1.5.txt has 0 lines

diff output for task3policy1.5.txt and task3policy1.6.txt has 0 lines

diff output for task3policy1.6.txt and task3policy1.7.txt has 0 lines

diff output for task3policy1.7.txt and task3policy1.8.txt has 0 lines

diff output for task3policy1.8.txt and task3policy1.9.txt has 0 lines

diff output for p2_policy2.txt and task3policy2.0.txt has 4196 lines

diff output for task3policy2.0.txt and task3policy2.1.txt has 1230 lines

diff output for task3policy2.1.txt and task3policy2.2.txt has 0 lines

diff output for task3policy2.2.txt and task3policy2.3.txt has 0 lines

diff output for task3policy2.3.txt and task3policy2.4.txt has 0 lines

diff output for task3policy2.4.txt and task3policy2.5.txt has 0 lines

diff output for task3policy2.5.txt and task3policy2.6.txt has 0 lines

diff output for task3policy2.6.txt and task3policy2.7.txt has 0 lines

diff output for task3policy2.7.txt and task3policy2.8.txt has 0 lines

diff output for task3policy2.8.txt and task3policy2.9.txt has 0 lines
```

Figure 2: diff output for 10 successive policy files for both players when starting with p1 policy 2



```
diff output for p1_policy1.txt and task3policy1.0.txt has 4132 lines

diff output for task3policy1.0.txt and task3policy1.1.txt has 2398 lines

diff output for task3policy1.1.txt and task3policy1.2.txt has 806 lines

diff output for task3policy1.2.txt and task3policy1.3.txt has 82 lines

diff output for task3policy1.3.txt and task3policy1.4.txt has 0 lines

diff output for task3policy1.4.txt and task3policy1.5.txt has 0 lines

diff output for task3policy1.5.txt and task3policy1.6.txt has 0 lines

diff output for task3policy1.6.txt and task3policy1.7.txt has 0 lines

diff output for task3policy1.7.txt and task3policy1.8.txt has 0 lines

diff output for task3policy1.8.txt and task3policy1.9.txt has 0 lines

diff output for p2_policy2.txt and task3policy2.0.txt has 3748 lines

diff output for task3policy2.0.txt and task3policy2.1.txt has 560 lines

diff output for task3policy2.1.txt and task3policy2.2.txt has 66 lines

diff output for task3policy2.2.txt and task3policy2.3.txt has 0 lines

diff output for task3policy2.3.txt and task3policy2.4.txt has 0 lines

diff output for task3policy2.4.txt and task3policy2.5.txt has 0 lines

diff output for task3policy2.5.txt and task3policy2.6.txt has 0 lines

diff output for task3policy2.6.txt and task3policy2.7.txt has 0 lines

diff output for task3policy2.7.txt and task3policy2.8.txt has 0 lines

diff output for task3policy2.8.txt and task3policy2.9.txt has 0 lines
```

Figure 3: diff output for 10 successive policy files for both players when starting with p2 policy 1



```
diff output for p1_policy2.txt and task3policy1.0.txt has 4848 lines

diff output for task3policy1.0.txt and task3policy1.1.txt has 3028 lines

diff output for task3policy1.1.txt and task3policy1.2.txt has 0 lines

diff output for task3policy1.2.txt and task3policy1.3.txt has 0 lines

diff output for task3policy1.3.txt and task3policy1.4.txt has 0 lines

diff output for task3policy1.4.txt and task3policy1.5.txt has 0 lines

diff output for task3policy1.5.txt and task3policy1.6.txt has 0 lines

diff output for task3policy1.6.txt and task3policy1.7.txt has 0 lines

diff output for task3policy1.7.txt and task3policy1.8.txt has 0 lines

diff output for task3policy1.8.txt and task3policy1.9.txt has 0 lines

diff output for p2_policy2.txt and task3policy2.0.txt has 4196 lines

diff output for task3policy2.0.txt and task3policy2.1.txt has 0 lines

diff output for task3policy2.1.txt and task3policy2.2.txt has 0 lines

diff output for task3policy2.2.txt and task3policy2.3.txt has 0 lines

diff output for task3policy2.3.txt and task3policy2.4.txt has 0 lines

diff output for task3policy2.4.txt and task3policy2.5.txt has 0 lines

diff output for task3policy2.5.txt and task3policy2.6.txt has 0 lines

diff output for task3policy2.6.txt and task3policy2.7.txt has 0 lines

diff output for task3policy2.7.txt and task3policy2.8.txt has 0 lines

diff output for task3policy2.8.txt and task3policy2.9.txt has 0 lines
```

Figure 4: diff output for 10 successive policy files for both players when starting with p2 policy 2

As we can see in the above output snippets of code, after 3 to 4 iterations, the policy is converging for both the players and then doesn't seem to change afterwards. For tie breaking in the code, if all the actions give value function = 0, then the action with smallest index is chosen optimal and if there are multiple actions which give value function = 1, then again the action with smallest index is chosen optimal.

Proof of convergence:-

Let player 1 start with policy π_1^0 and player 2 start with policy π_2^0. After 2i iterations of policy updates, each of player 1 and player 2 would have updated their policy by i times. Let player 1 policy after 2i iterations be π_1^i and player 2 policy be π_2^i.

In every iteration, the policy for one of the players is updated. Let the expected $\#wins$ for player 1 initially be x_0 and that for player 2 be y_0. Since there are are only two players, therefore $x_0 = 1 - y_0$. Now suppose player 1 updates his policy such that his expected $\#wins$ become x_1 and for player 2 will become y_1. Now $x_1 = 1 - y_1$ and $x_1 \geq x_0$. So $y_1 \leq y_0$. Later player 2 updates his policy resulting into $x_2 = 1 - y_2$, $x_2 \leq x_1$, and $y_2 \geq y_1$.

There are only finitely many possible policies for both player 1 and player 2 precisely 9 times the $\#states$. So when player 1 improves its policy, it is not that player 2 has worsened its policy, player 2 policy is same but expected $\#wins$ may have decreased. Although the policy depends on the environment, at this point we can safely say that when environment improves its policy, we can not go back to a policy we already improved from thus the acyclicity of policies can be confirmed.

If we are no more able to increase the expected $\#wins$, it means we have reached the equality for x_is and y_is so we will get $x_i = x_{i-1}$ and $y_i = y_{i-1}$. This further means that our policies have saturated which is more likely given that we have only finitely many possible policies for both the players. So no more improvable expected $\#wins$ means no more improvable policies and hence $\pi_1^i = \pi_1^{i-1}$ and $\pi_2^i = \pi_2^{i-1}$. Hence, the convergence of policies.

Properties of converged policy:-

(i) Every game will result into the same output ie. either every game will result into "player 1 wins" or "player 2 wins" or "draw game". This is because since the policies for both the players are fixed therefore all the moves for both of them will also be fixed from first move to last move and hence the result will also be same.

(ii) Since the policies have converged, it means neither of the players have any more scope of improvement in their expected $\#wins$.