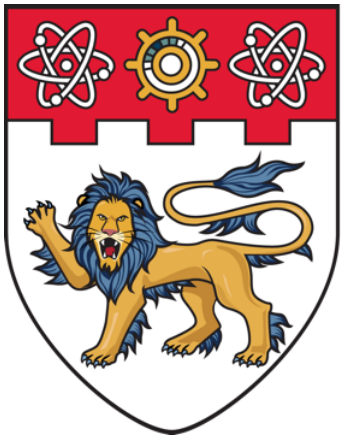

NANYANG TECHNOLOGICAL UNIVERSITY

SINGAPORE

CZ4046 Intelligent Agents

Assignment 2

Marathe Ajinkya Avinash
(U1522716K)

Objective:

The objective of this assignment is to devise a strategy for the Repeated Three Prisoners' Dilemma problem. Three players chosen randomly will play each other in a match that consists of multiple rounds of the three prisoners' dilemma game. Each player will be given the history of all the moves that the three players including himself have played in the match so far. With this information the player will have to decide the strategy in the next round.

Strategy:

```
int selectAction(int n, int[] myHistory, int[] oppHistory1, int[] oppHistory2){
    // cooperate in the first round
    if (n==0) return 0;
    // Add the payoff from the last round to compute the total payoff till now for every player
    myPayoff+=payoff[myHistory[n-1]][oppHistory1[n-1]][oppHistory2[n-1]];
    opp1Payoff+=payoff[oppHistory1[n-1]][myHistory[n-1]][oppHistory2[n-1]];
    opp2Payoff+=payoff[oppHistory2[n-1]][myHistory[n-1]][oppHistory1[n-1]];

    // If my player is winning against both the opponent, i.e. payoff is largest then cooperate
    if((myPayoff>=opp1Payoff)&&(myPayoff>=opp2Payoff)){
        return 0;
    }
    // If winning against only opponent 1, cooperate if the last move of opponent 2 was cooperate, otherwise defect
    else if(myPayoff>opp1Payoff){

        if(oppHistory2[n-1]==0) return 0;
        else return  1;
    }
    // If winning against only opponent 2, cooperate if the last move of opponent 1 was cooperate, otherwise defect
    else if(myPayoff>opp2Payoff){
        if(oppHistory1[n-1]==0) return 0;
        else return  1;
    }
    // If losing against both, i.e. payoff is the lowest, then defect
    else return 1;
}
```

Figure 0-1

The strategy adopted here decides whether to defect or cooperate based on the payoff that the agent has received so far and the last moves of the opponents. The agent adds the payoff of the last round to calculate the total payoff so far in every round. The agent starts off by cooperating in the first round and decides what to do based on one of the four cases.

Case 1: If the agent's total payoff is higher than the total payoffs of both other agents, then the agent cooperates. Here as the agent is winning against both its opponents, it decides to take a soft stand of cooperating.

Case 2: If the agent's total payoff is more than opponent 1 but lesser than opponent two, it cooperates if the last move of opponent 2 was cooperate, otherwise it defects. As the agent is winning only against opponent 1, it cooperates only if opponent 2 has shown signs of cooperating in the last round

Case 3: If the agent's total payoff is more than opponent 2 but lesser than opponent 1, it cooperates if opponent 1 cooperated in the last move, otherwise it defects. . As the agent is winning only against opponent 2, it cooperates only if opponent 1 has shown signs of cooperating in the last round

Case 4: defects if it is losing against both the agents

Figure 0-1 shows the code snippet for the selectAction() method

Results:

Evaluation against other players:

The strategy defeats the typical players listed in the assignment. Figure 0-1 shows the results for one tournament by running the assignment code.

```
Tournament Results
Marathe_Ajinkya_Player: 172.30853 points.
T4TPlayer: 164.51196 points.
TolerantPlayer: 158.59612 points.
RandomPlayer: 153.51874 points.
FreakyPlayer: 145.73045 points.
NicePlayer: 140.83682 points.
NastyPlayer: 139.70978 points.
```

Figure 0-1

When the strategy was tested out for 1000 tournaments, the strategy was victorious for almost half of the tournaments. The strategy was also tested for 5000 and 10000 tournaments and it gave similar results. The tolerant player emerges at the second place in each of these experiments. Figure 0-2 shows the results for 1000 tournaments.

PLAYER	MATCHES WON	
Marathe_Ajinkya_Player	486/1000	(48.60%)
TolerantPlayer	295/1000	(29.50%)
T4TPlayer	200/1000	(20.00%)
NicePlayer	12/1000	(1.20%)
FreakyPlayer	4/1000	(0.40%)
RandomPlayer	3/1000	(0.30%)
NastyPlayer	0/1000	(0.00%)

Figure 0-2

Selected Cases:

1. NastyPlayer scored 3.6153846 points, RandomPlayer scored 0.978022 points, and Marathe_Ajinkya_Player scored 3.5604396 points.
2. NastyPlayer scored 2.1 points, TolerantPlayer scored 1.9888889 points, and Marathe_Ajinkya_Player scored 2.0444446 points.
3. RandomPlayer scored 3.7884614 points, T4TPlayer scored 3.6923077 points, and Marathe_Ajinkya_Player scored 3.7884614 points.
4. NicePlayer scored 2.2747252 points, RandomPlayer scored 5.4065933 points, and Marathe_Ajinkya_Player scored 5.3516483 points.
5. NicePlayer scored 6.0 points, TolerantPlayer scored 6.0 points, and Marathe_Ajinkya_Player scored 6.0 points.
6. NicePlayer scored 6.0 points, NicePlayer scored 6.0 points, and Marathe_Ajinkya_Player scored 6.0 points.
7. RandomPlayer scored 1.3505155 points, TolerantPlayer scored 3.257732 points, and Marathe_Ajinkya_Player scored 3.2061856 points.
8. RandomPlayer scored 1.0582525 points, FreakyPlayer scored 3.485437 points, and Marathe_Ajinkya_Player scored 3.4368932 points.

Observations-

- As it was expected, the strategy does not win against the nasty player as defer is the dominant strategy. But the agent protects itself as it can be seen from examples 1 and 2.
- As seen in example 4, if one of the players is a Nice player, then the strategy outperforms it by a huge margin because it has to adjust for the other opponent which is not so nice and defect more. But in this case everyone gets lower payoffs. If both the players are nice players, then it chooses to cooperate and everyone scores a higher payoff as seen in example 6.
- The strategy performs better or at least as good as T4T, Random, Freaky and Tolerant player by adjusting its decisions as seen in the examples. When the third opponent is nasty, then the strategy outperforms these players, but otherwise it almost matches with them and everyone gets a higher payoff. Refer to examples 1,2,7,8 etc.