CZ4046 Intelligent Agents

Assignment 2

Name: Pamela Sin Hui

Matriculation number: K1920110A

1. Agent Implementation

- The agent first cooperates on the first round (n = 0)

- For subsequent rounds, the agent keeps alternating between two approaches:
  - First approach:
    For odd rounds (n = 1, 3, ..., 99), the agent counts the total number of defects made by both opponents and the total number of cooperates made by both opponents. If the total number of defects is greater than cooperates, the agent chooses to defect. Or else, the agent chooses to cooperate.

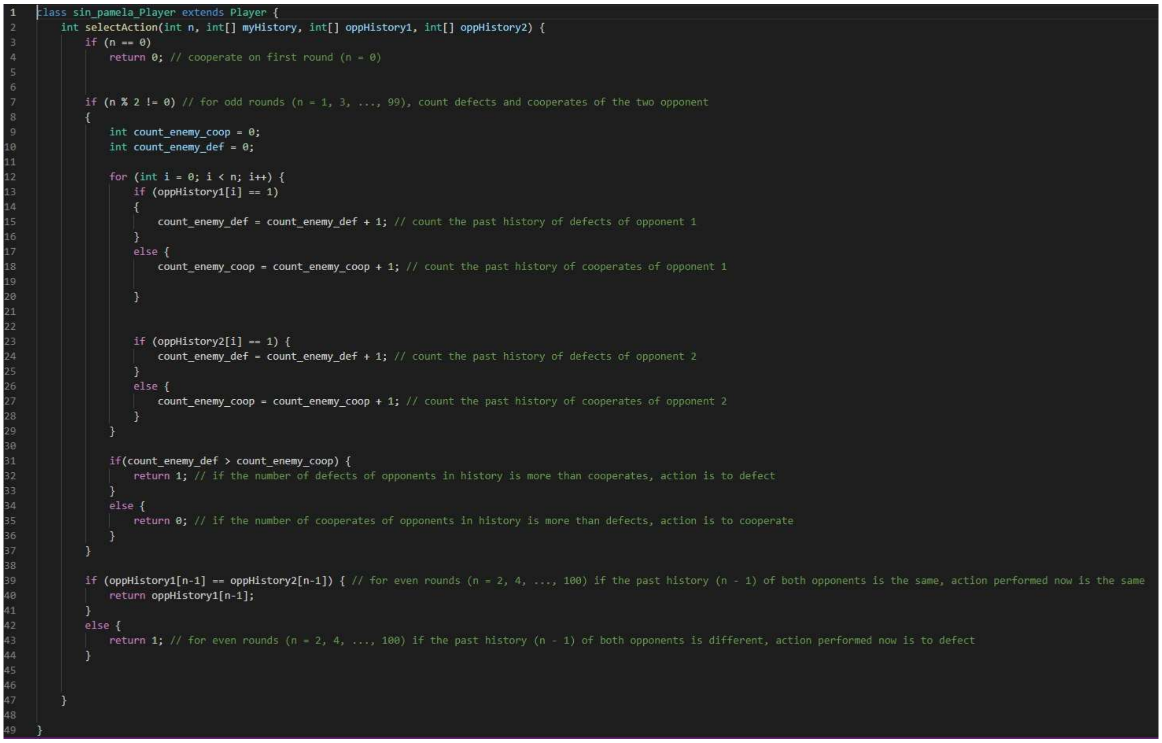
  - Second approach:
    For even rounds (n = 2,4, …, 98), the agent looks at the (n-1) choice of both opponents. If both opponents chose the same action, the agent chooses to follow that same action for round n. Or else, which is the case of one opponent choosing to cooperate and the other choosing to defect at (n-1) round, the agent chooses to defect for round n.

The alternating approaches balances between two arbitrary metrics throughout the round.

For the first approach, it considers the history of the match thus far to determine which actions the two opponents combined have taken previously. If the two opponents have a history of cooperating (at a 50% threshold) rather than defecting, then the agent decides to cooperate (and vice versa).

For the second approach, the previous action (which can be viewed as the short-term history) of both agents is considered. In this approach, the agent determines the strategy based on recent actions of both agents, and changes accordingly to keep in trend with recent events that has just elapsed.

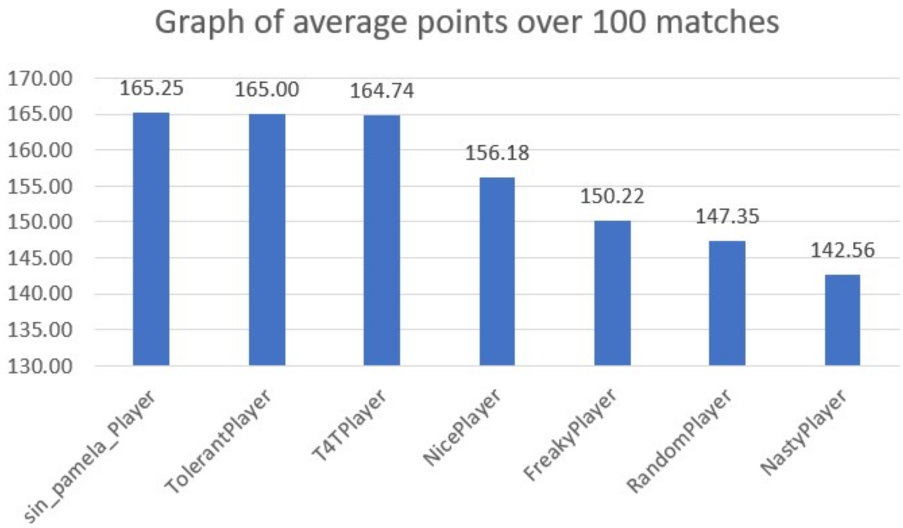
The alternating strategies strike a balance to take into account both the entire history of how the opponents have treated the agent as well as react to short-term changes in opponent strategies.



```java
class sin_pamela_Player extends Player {
    int selectAction(int n, int[] myHistory, int[] oppHistory1, int[] oppHistory2) {
        if (n == 0)
            return 0; // cooperate on first round (n = 0)


        if (n % 2 != 0) // for odd rounds (n = 1, 3, ..., 99), count defects and cooperates of the two opponent
        {
            int count_enemy_coop = 0;
            int count_enemy_def = 0;

            for (int i = 0; i < n; i++) {
                if (oppHistory1[i] == 1)
                {
                    count_enemy_def = count_enemy_def + 1; // count the past history of defects of opponent 1
                }
                else {
                    count_enemy_coop = count_enemy_coop + 1; // count the past history of cooperates of opponent 1

                }


                if (oppHistory2[i] == 1) {
                    count_enemy_def = count_enemy_def + 1; // count the past history of defects of opponent 2
                }
                else {
                    count_enemy_coop = count_enemy_coop + 1; // count the past history of cooperates of opponent 2
                }
            }

            if(count_enemy_def > count_enemy_coop) {
                return 1; // if the number of defects of opponents in history is more than cooperates, action is to defect
            }
            else {
                return 0; // if the number of cooperates of opponents in history is more than defects, action is to cooperate
            }
        }

        if (oppHistory1[n-1] == oppHistory2[n-1]) { // for even rounds (n = 2, 4, ..., 100) if the past history (n - 1) of both opponents is the same, action performed now is the same
            return oppHistory1[n-1];
        }
        else {
            return 1; // for even rounds (n = 2, 4, ..., 100) if the past history (n - 1) of both opponents is different, action performed now is to defect
        }


    }

}
```

2. Competition with example strategies

The agent was pitted against the example strategies in 100 matches, where each match consists of 100 rounds. The agent, sin_pamela_Player performed better than the example strategies, with an average total score of 165.25 for each match. TolerantPlayer and T4TPlayer trails closely behind at 165.00 and 164.74 points respectively, while there was a large margin for the remaining strategies.



The agent holds up against the NastyPlayer by producing a defect strategy in both the first approach (considering the entire history of events) and second approach (considering the most recent previous move of the player). This is a similar explanation for the NicePlayer and Freaky player, where such players swing to one extreme with only one outcome no matter what the actions of other players are. The agent can be thought of to hold up against the RandomPlayer by countering its randomness by looking at its short and long-term history in alternate turns and producing an action of its own. Though, the agent not only accounts for the action of the RandomPlayer, but it also considers the strategy of the other player to give a calibrated response.

It is noted that top players are the ones that consider the action of the other two players. An explanation for the agent doing marginally better than the TolerantPlayer is that the agent further considers the most recent actions of opponents and therefore has better reactivity to short-term swings. An explanation for the agent doing better than the T4TPlayer is that the T4TPlayer only takes into account the most recent action of one if its opponents it chooses at random in employing Tit-for-Tat, which is myopic and introduces a high degree of randomness at each turn.

3. <u>Conclusion</u>

As shown empirically, agents must take into account both the short-term changes their opponents deploy and the history of actions their opponents have taken to formulate an appropriate response at each turn. It is noted that for each of the 100 matches (where each match consists of 100 rounds), the top average player of the 100 matches did not consistently get the top position for each match. The random strategies of some players has an impact in introducing randomness to the game. Considering multiple iterations of matches gives us a flavour of which overarching strategy can get better rewards on consistently and on average.

References

[1] https://medium.com/thinking-is-hard/a-prisoners-dilemma-cheat-sheet-4d85fe289d87

[2] https://en.wikipedia.org/wiki/Repeated_game

[3] https://nordstromjf.github.io/IntroGameTheory/S_RepeatPD.html