NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

CZ4046 Intelligent Agents

Iterated Prisoners Dilemma – Assignment 2

Name: Singh Jasraj

Matriculation: U1940558D

Introduction

The prisoner's dilemma is a theoretical scenario in game theory where two suspects are arrested and interrogated separately. They face a choice: to cooperate with each other by staying silent, or betray each other by confessing. If both remain silent, they both get a lighter sentence; if both confess, they both receive a harsher sentence; if one confesses and the other remains silent, the confessor gets the lightest sentence while the other gets the harshest. The dilemma illustrates the challenge of trust, cooperation, and self-interest in decision-making.

In this task, our objective is to devise a tactic for a three-person, limited, repeated Prisoner's Dilemma, consisting of 90 to 100 turns per game. The table below shows all potential move combinations and their corresponding payoffs in the Prisoner's Dilemma. Cooperation is represented by 0, and defection by 1.

Player	Opponent 1	Opponent 2	Payoff
1	0	0	**8**
0	0	0	**6**
1	0	1	**5**
1	1	0	**5**
0	0	1	**3**
0	1	0	**3**
1	1	1	**2**
0	1	1	**0**

Several well-known strategies exist for a two-player Prisoner's Dilemma. Some examples are:

1. Tit-for-Tat Player: This player mirrors their opponent's previous move, starting with cooperation.
2. Freaky Player: This player randomly chooses an action at the beginning and maintains that action for the entire match.
3. Random Player: This player selects a random action in each round, regardless of previous moves.
4. Nice Player: This player always cooperates with their opponent.

Among the strategies for a 2-person Prisoner's Dilemma, the Tit-for-Tat approach (T4T) stands out as the most prominent. However, it's important to recognize that T4T aims for a draw rather than a victory. This method may not be as effective when applied to a three-player game. Specifically, if using the T4T strategy, an agent must decide which of the two opponents to imitate. The choice made could significantly impact the agent's performance in the game.

Thought Process

The problem with nasty agent

Based on Axelrod's principles, in a 2-player repeated prisoner's dilemma with a predetermined and known number of turns, the prevailing strategy is to consistently defect, essentially behaving as a nasty agent. By always defecting, an agent can ensure a victory or at least a draw, as the opponent will never have the opportunity to score points from the current player's actions.

This reasoning can also be extended to a single iterated three-player Prisoner's Dilemma match. In a game involving a nasty player, no other player can outscore the nasty player, as cooperating leads to a lower payoff compared to defecting. If opponents also choose to defect, they will only receive the same payoff as the nasty player. Consequently, even in a single three-player Prisoner's Dilemma match, the nasty player remains unbeaten.

In contrast, a tournament-style three-player Prisoner's Dilemma can alter the dynamics for an always-defect agent. This occurs when agents like Tit-for-Tat (T4T) and Tolerant Player, who are generally cooperative with each other, participate in a match. They maintain cooperation throughout the game, resulting in high payoffs for both. However, when facing a nasty player, these agents retaliate against the defections by defecting themselves, leading to reduced payoffs for all players involved.

The figure below demonstrates this issue using an example of a tournament conducted 1000 times, featuring a NastyPlayer (always-defect), a T4TPlayer (Tit-for-Tat), a NicePlayer, and a TolerantPlayer. As the results indicate, the Tolerant Player emerges victorious, as it cooperates with other cooperative players and punishes the nasty player during their match, thereby optimizing its score. This example highlights the importance of penalizing an always-defect strategy while cooperating with other cooperative players whenever feasible.

```
Enter number of runs for the tournament:
1000

TolerantPlayer: 72994.7 points, 1000 wins.
T4TPlayer: 72049.9 points, 0 wins.
NicePlayer: 70515.086 points, 0 wins.
NastyPlayer: 58919.863 points, 0 wins.
```

Defection should still be the dominant choice

Striking a balance between consistently defecting and occasionally cooperating is crucial for success in the Prisoner's Dilemma, as demonstrated in the previous discussion. An effective strategy should consider the importance of cooperation, trust, and adaptability, while also being prepared to retaliate or punish non-cooperative behavior. This balance allows an agent to capitalize on cooperative opportunities, maximize payoffs, and avoid falling into the pitfalls of being too cooperative or too aggressive in its interactions with other players.

Defection should indeed remain the dominant choice in many cases. Examining the table of all possible move combinations and their payoffs reveals that defection often leads to better outcomes – as indicated by the green rows. Additionally, it can be inferred that the current player benefits more when more opponents decide to cooperate.

The scenarios in the following table can thus be deduced, conclusively showing that defect would still be the dominant strategy in a single match tournament.

Possible Scenarios	Best Action	Payoff
If both opponents defect	Defect	2
If one opponent defects and other cooperates	Defect	5
If both opponents cooperate	Defect	8

Implementation

Guessing the Opponents' Strategy

Identifying an opponent's strategy using the available history array might be highly effective when the devised strategy competes solely against the six predefined players: NicePlayer, NastyPlayer, RandomPlayer, TolerantPlayer, FreakyPlayer, and T4TPlayer. By examining the history array, which records each player's past moves, it is possible to detect patterns and traits unique to each specific strategy. Once the opponent's strategy is discerned, the devised strategy can adjust its moves to capitalize on the opponent's weaknesses and optimize its payoff. Nevertheless, this method could face limitations when dealing with more sophisticated or adaptive strategies, or when opponents employ hybrid strategies that merge elements from different approaches. In these instances, accurately predicting and responding to the opponent's strategy may prove more difficult.

Nonetheless, according to this assignment's assessment, the proposed strategy will compete against agents developed by other students, encompassing various combinations. There is an infinite number of ways to design these strategies, making it nearly impossible to identify all possible opponents. Consequently, relying on predicting opponents' strategies or devising a strategy that performs well only against the six given players but falters against unknown ones is not feasible. It is crucial to create a more resilient and adaptable strategy to ensure success.

Strategy

A few strategies were trialed in this assignment to see which performed the best against the 6 provided players, along with some other common prisoners' dilemma agents, such as the Pavlov Player and a modified T4T Player. But they were carefully designed to be more generic instead of beating their strategies specifically.

Our approach is to start as a cooperative agent in order to entice other players to cooperate as well, thereby collecting the maximum reward. In the later rounds, we check if the other players can be trusted by checking the proportion of rounds they defected in. If either of the players cannot be defected, we defect as well. Otherwise, if we are unsure of their intent, we simply do what they did in the previous round. And, of course, if we can trust them both to cooperate, then we cooperate as well.

```java
class Singh_Jasraj_Player extends Player {

    // initialize the number of defects to 0
    int oppDefects1 = 0, oppDefects2 = 0;

    // check if the opponent can be truste based on their proportion fo defects from history
    float isTrusted(int oppDefects, int n){
        float defectRate = (float)oppDefects/n;
        // can be trusted if ratio <= 0.05
        if (defectRate <= 0.05) {
            return 1;
        }
        // unsure if 0.05 < ratio <= 0.20
        else if (defectRate <= 0.20) {
            return 0;
        }
        // cannot be trusted if ratio > 0.2
        return -1;
    }

    int selectAction(int n, int[] myHistory, int[] oppHistory1, int[] oppHistory2) {

        // start by cooperating
        if (n == 0) {
            return 0;
        }

        // update number of defects by each opponent
        oppDefects1 += oppHistory1[n-1];
        oppDefects2 += oppHistory2[n-1];

        // if either of the defected in the last round, we defect
        if (oppHistory1[n-1] == 1 || oppHistory2[n-1] == 1) {
            return 1;
        }
        // else if either one of them cannot be trusted, we defect
        else if (isTrusted(oppDefects1, n) == -1 || isTrusted(oppDefects2, n) == -1) {
            return 1;
        }
        // else if unsure if one of them can be trusted, we repeat their action
        else if (n > 2) {
            if (isTrusted(oppDefects1, n) == 0) {
                return oppHistory1[n-2];
            }
            else if (isTrusted(oppDefects2, n) == 0) {
                return oppHistory2[n-2];
            }
        }
        // else we cooperate
        return 0;
    }

}
```

Evaluation

This section's objective is to outline the evaluation process for the previously mentioned strategies. The primary approach involved conducting up to 1000 tournaments, calculating the agent's total score in the tournament, and determining the number of times the agent emerged victorious against other agents.

```
Enter number of runs for the tournament:
10000

Singh_Jasraj_Player: 1693434.6 points, 6391 wins.
TolerantPlayer: 1645346.1 points, 1885 wins.
T4TPlayer: 1636759.5 points, 1689 wins.
NicePlayer: 1516074.1 points, 30 wins.
FreakyPlayer: 1460311.2 points, 5 wins.
NastyPlayer: 1403632.5 points, 0 wins.
RandomPlayer: 1339488.6 points, 0 wins.
```

Conclusion

In summary, creating a consistently victorious strategy is an extremely challenging endeavor, given the countless ways a strategy can counter an existing one. The final chosen strategy presented in this report is designed to be as versatile as possible – predominantly defecting while permitting occasional cooperation in an attempt to attain the highest possible score.