NANYANG TECHNOLOGICAL UNIVERSITY

CZ4046 Assignment 2 Report

Chen Zhiwei
U1620741J

Introduction

In this assignment, I am assigned the task to develop a strategy for an agent in a three player repeated prisoners' dilemma. In this simulation, triples of players will play each other repeatedly in a match, where a match consists of about 100 rounds, and the score for the match is the average payoffs from each round of that match. At each round, the strategy I developed computes the next action for my agent given the historical actions of my opponents.

High Level Explanation of Implementation

The payoff given in the task can be summarised into the following:

$$
\begin{aligned}
U(CCC) &= 6; \\
U(CCD) &= U(CDC) = 3; \\
U(CDD) &= 0; \\
U(DCC) &= 8; \\
U(DCD) &= U(DDC) = 5; \\
U(DDD) &= 2;
\end{aligned}
$$

$where\ U(myAction, opp1Action, opp2Action).$

From the above payoff, we can infer the following observations:

1. The **dominant strategy** for all players is to do **D**. This is because every player is always getting higher reward by defecting, regardless of what the other 2 opponents do:

$$
\begin{aligned}
U(DCC) &> U(CCC) \\
U(DCD) &> U(CCD) \\
U(DDD) &> U(CDD)
\end{aligned}
$$

2. Every player is better off if **more of its opponents decide to do C**:

$$
\begin{aligned}
U(DCC) &> U(DCD) > U(DDD) \\
U(CCC) &> U(CCD) > U(CDD)
\end{aligned}
$$

Bearing in mind the second observation, the **main goal** of Chen_Zhiwei_Player will be to **incentivise its opponents to do C, thereby rendering Chen_Zhiwei_Player to mostly do C**. But if the opponents are detected as nasty or random, then it is **unlikely that my strategy will affect their strategy**, and so Chen_Zhiwei_Player will adopt dominant strategy of doing D instead. Besides incentivising opponents to do C, Chen_Zhiwei_Player shall also closely monitor its payoff, and do D when necessary. With these considerations, I developed the following laws (to be applied sequentially):

Law 1: If n < 2, then always C.

Law 2: Tolerate 2 consecutive D from both opponents in the nearest 2 time steps. If 2 consecutive D is detected from both opponents in the nearest 2 time steps, then D for the current time step.

Law 3: If one of the opponents is Nasty, then always defect.

Law 4: If one of the opponents is Random, then always defect.

Law 5: If my current score is lower than one of the opponents, then always defect.

Law 6: If Law 1 and Law 2 do not apply, then be a T4TPlayer by randomly picking an opponent at each play, and uses the 'tit-for-tat' strategy against them.

Implementation

```
class Chen_Zhiwei_Player extends Player {
	int myScore = 0;
	int opp1Score = 0;
	int opp2Score = 0;

	int selectAction(int n, int[] myHistory, int[] oppHistory1, int[] oppHistory2) {
		// First Law: Always cooperate in first 2 rounds
		if (n < 2) return 0;

		// Second Law: Tolerate 2 consecutive defects from both opp
		// If 2 consecutive defects from both opp, then defect
		if (oppHistory1[n-1] == 1 && oppHistory1[n-2] == 1 &&
			oppHistory2[n-1] == 1 && oppHistory2[n-2] == 1)
			return 1;

		// Third Law: if one of the opponents is Nasty, then always defect
		boolean isOpp1Nasty, isOpp2Nasty;
		isOpp1Nasty = isNasty(n, oppHistory1);
		isOpp2Nasty = isNasty(n, oppHistory2);
		if (isOpp1Nasty || isOpp2Nasty) return 1;

		// Fourth Law: if one of the opponents is Random, then always defect
		boolean isOpp1Random, isOpp2Random;
		isOpp1Random = isRandom(n, oppHistory1);
		isOpp2Random = isRandom(n, oppHistory2);
		if (isOpp1Random || isOpp2Random) return 1;

		// Fifth Law: if my score is lower than one of the opp, then defect
		myScore += payoff[myHistory[n-1]][oppHistory1[n-1]][oppHistory2[n-1]];
		opp1Score += payoff[oppHistory1[n-1]][oppHistory2[n-1]][myHistory[n-1]];
		opp2Score += payoff[oppHistory2[n-1]][oppHistory1[n-1]][myHistory[n-1]];
		 if (myScore < opp1Score || myScore < opp2Score) return 1;

		// Sixth Law: If above laws don't apply, then be a T4TPlayer
		if (Math.random() < 0.5) return oppHistory1[n-1];
		else return oppHistory2[n-1];
	}

	boolean isNasty(int n, int[] oppHistory) {
		int cnt = 0;
		for (int i=0; i<n; i++){
			if (oppHistory[i] == 1)
				cnt++;
		}

		if (cnt == n) return true;
		else return false;
	}

	boolean isRandom(int n, int[] oppHistory) {
		int sum = 0;
		double eps = 0.025;
		for (int i=0; i<n; i++) {
			sum += oppHistory[i];
		}

		// if ratio is roughly 0.5, then the opponent is highly likely to be random
		double ratio = (double) sum / n;
		if (Math.abs(ratio - 0.5) < eps) return true;
		else return false;
	}
}
```

Evaluation

The main goal of Chen_Zhiwei_Player is to maximise its own payoff. In view of the existence of the other 2 opponents.

The following section present the evaluations of performance between Chen_Zhiwei_Player and other agents under the **runTournament** function. For each example agent X, I will first report the performance when both opponents are agent X, and when one opponent is example agent X and the other is also Chen_Zhiwei_Player.

Against NicePlayer

Match Against	Agent	Payoff
2 NicePlayer	**Chen_Zhiwei_Player**	**6.0**
	NicePlayer	6.0
	NicePlayer	6.0
1 NicePlayer 1 Chen_ZhiweiPlayer	**Chen_Zhiwei_Player**	**6.0**
	Chen_Zhiwei_Player	6.0
	NicePlayer	6.0

When both opponents are NicePlayer, then from round 3 onwards, Chen_Zhiwei_Player will always do C under and **Law6**, such that all players of the match attained equal payoff of 6.0.

When one of the opponent is also Chen_Zhiwei_Player, then again, from round 3 onwards, only **Law6** will be triggered, such that Chen_Zhiwei_Player will only do C under **Law6** (because the NicePlayer will never defect). The end result is that all players of the match attain equal payoff of 6.0

Against Nasty Player

Match Against	Agent	Payoff
2 NastyPlayer	**NastyPlayer**	**2.06**
	NastyPlayer	2.06
	Chen_Zhiwei_Player	1.96
1 NastyPlayer 1 Chen_ZhiweiPlayer	**NastyPlayer**	**2.11**
	Chen_Zhiwei_Player	2.02
	Chen_Zhiwei_Player	2.02

When both opponents are NastyPlayer, then from round 3 onwards, Chen_Zhiwei_Player will do D under **Law2**. However, as Chen_Zhiwei_Player gained less points in the first 2 rounds, the end performance of Chen_Zhiwei_Player is thus marginally less than the 2 NastyPlayer.

When one of the opponent is a Chen_Zhiwei_Player, then from round 3 onwards, **Law3** will be triggered as NastyPlayer is detected, such that Chen_Zhiwei_Player will do D, which is the dominant strategy. Again, as Chen_Zhiwei_Player gained less points in the first 2 rounds, the end performances of the 2 Chen_Zhiwei_Player are thus marginally less than the NastyPlayer. Nonetheless, it is still satisfactory as the difference is marginal.

Against FreakyPlayer

As FreakyPlayer are like NicePlayer and NastyPlayer, in the following table, I present all possible scenarios of FreakPlayer being NicePlayer/NastyPlayer.

Match Against	Agent	Payoff
2 NicePlayer	**Chen_Zhiwei_Player**	**6.0**
	NicePlayer	6.0
	NicePlayer	6.0
1 NicePlayer 1 Chen_ZhiweiPlayer	**Chen_Zhiwei_Player**	**6.0**
	Chen_Zhiwei_Player	6.0
	NicePlayer	6.0
2 NastyPlayer	**NastyPlayer**	**2.06**
	NastyPlayer	2.06
	Chen_Zhiwei_Player	1.96
1 NastyPlayer 1 Chen_ZhiweiPlayer	**NastyPlayer**	**2.11**
	Chen_Zhiwei_Player	2.02
	Chen_Zhiwei_Player	2.02
1 NicePlayer 1 NastyPlayer	**NastyPlayer**	**5.07**
	Chen_Zhiwei_Player	4.96
	NicePlayer	0.07

The first 4 matches presented in the table are identical to the performances reported in the previous 2 sections against NicePlayer and against NastyPlayer.

Under the match against 1 NicePlayer and 1 NastyPlayer, from round 3 onwards, **Law3** will be triggered as one NastyPlayer is detected, such that Chen_Zhiwei_Player will do D, which is the dominant strategy. The end performance of Chen_Zhiwei_Player is thus marginally lower than the NastyPlayer.

Against RandomPlayer

Match Against	Agent	Payoff
2 RandomPlayer	**Chen_Zhiwei_Player**	**4.76**
	RandomPlayer	2.91
	RandomPlayer	2.63
1 RandomPlayer 1 Chen_ZhiweiPlayer	**Chen_Zhiwei_Player**	**3.87**
	Chen_Zhiwei_Player	3.87
	RandomPlayer	2.47

When both opponents are RandomPlayer, then for the first few rounds after round 3, **Law2/3/4/5/6** are all possible to be triggered. As time progress, mainly **Law4** will be triggered, as RandomPlayer shall be detected, which will make Chen_Zhiwei_Player do D i.e. the dominant strategy. The end performance of Chen_Zhiwei_Player is thus satisfactory, as it is the highest among all 3.

When one of the opponents is also Chen_Zhiwei_Player, then again, for the first few rounds after round 3, **Law2/3/4/5/6** are all possible to be triggered. As time progress, mainly **Law4** will be triggered, as one RandomPlayer is detected, which will make Chen_Zhiwei_Player do D i.e. the dominant strategy. The end performance of Chen_Zhiwei_Player is thus satisfactory, as it is the highest among all 3.

Against TolerantPlayer

Match Against	Agent	Payoff
2 TolerantPlayer	**Chen_Zhiwei_Player**	**6.0**
	TolerantPlayer	6.0
	TolerantPlayer	6.0
1 TolerantPlayer 1 Chen_ZhiweiPlayer	**Chen_Zhiwei_Player**	**6.0**
	Chen_Zhiwei_Player	6.0
	TolerantPlayer	6.0

When both opponents are TolerantPlayer, the analysis will be similar to that when against NicePlayer, as the threshold for TolerantPlayer to do D is very high. Then from round 3 onwards, Chen_Zhiwei_Player will always do C **Law6**, such that all players of the match attained equal payoff of 6.0.

When one of the opponent is also Chen_Zhiwei_Player, then again, from round 3 onwards, only **Law6** will be triggered (because the threshold for TolerantPlayer to do D is very high), such that Chen_Zhiwei_Player will do C. The end performance is that all players of the match attained equal payoff of 6.0.

Against T4TPlayer

Match Against	Agent	Payoff
2 T4TPlayer	**Chen_Zhiwei_Player**	**6.0**
	T4TPlayer	6.0
	T4TPlayer	6.0
1 T4TPlayer 1 Chen_ZhiweiPlayer	**Chen_Zhiwei_Player**	**6.0**
	Chen_Zhiwei_Player	6.0
	T4TPlayer	6.0

When both opponents are T4TPlayer, then from round 3 onwards, Chen_Zhiwei_Player will always do C under **Law6** (because the 2 T4TPlayer also start off by doing C), such that all players of the match attained equal payoff of 6.0.

When one of the opponents is also Chen_Zhiwei_Player, then again, from round 3 onwards, Chen_Zhiwei_Player will always do C under **Law6** (because both Chen_Zhiwei_Player and

T4TPlayer start off by doing C). The end performance is that all players of the match attained equal payoff of 6.0.