Computational Game Theory
Integrated Master (BSc. + MSc.) of Computer Science and Engineering
Faculty of Sciences and Technology of New University of Lisbon of New University of Lisbon (FCT NOVA | FCT/UNL)
2018/2019 - 2nd Semester

STRATEGIES FOR THE PRISONERS' DILEMMA

—

By Rúben André Barreiro

INTRODUCTION

The ***Prisoners' Dilemma*** is a standard example of a *game analyzed in game theory* that shows why two completely rational individuals might not cooperate, even if it appears that it is in their best interests to do so. (*Wikipedia* - https://en.wikipedia.org/wiki/Prisoner%27s_dilemma)

Brief History

It was originally framed by ***Merrill Flood*** and ***Melvin Dresher*** while working at RAND in 1950. ***Albert W. Tucker*** formalized the game with prison sentence rewards and named it "prisoner's dilemma". (*Wikipedia* - https://en.wikipedia.org/wiki/Prisoner%27s_dilemma)

Problem

Brief Definition

Two members of a criminal gang are arrested and imprisoned. Each prisoner is in solitary confinement with no means of communicating with the other. The prosecutors lack sufficient evidence to convict the pair on the principal charge, but they have enough to convict both on a lesser charge. Simultaneously, the prosecutors offer each prisoner a bargain. Each prisoner is given the opportunity either to betray the other by testifying that the other committed the crime, or to cooperate with the other by remaining silent. The offer is:

- If **A** and **B** each **betray** the other, each of them serves **two years** in prison;
- If **A** betrays **B** but **B remains silent**, **A** will be set free and **B** will serve **three years** in prison (**and vice versa**);
- If **A** and **B** both **remain silent**, both of them will only serve **one year** in prison (**on the lesser charge**);

Player 1 \ Player 2	Cooperate	Defect
Cooperate	(-1,-1)	(-3,0)
Defect	(0,-3)	(-2,-2)

Table 1: The matrix of utilities of the original problem

(*Wikipedia* - https://en.wikipedia.org/wiki/Prisoner%27s_dilemma)

Solving the Problem

In this course, it was asked to solve a modified version of the original Prisoners' Dilemma. Where the matrix of utilities is the following one:

Player 1 \ Player 2	Cooperate	Defect
Cooperate	(3,3)	(4,0)
Defect	(0,4)	(1,1)

Table 2: The matrix of utilities of the considered problem

Strategies Implemented

1. Gradual Strategy

Initially, was implemented the known Gradual Strategy. This strategy consists, basically, in use the "Cooperate" action in the first action and then, continues to do so as long as the opponent make "Cooperate" actions too. Then, after the first "Defect" action of the opponent, it defects one time and cooperates two times. After the second "Defect" action of the opponent, it defects two times and cooperates two times... ***After the nth defection of the opponent, it reacts with n consecutive defections and then, with two cooperates (some kind of "calm down")***. ***When it's being made a defection from 1 to n, or two consecutive cooperatess, it will continuing to detect the opponent's "Defect" actions, to be made more punishments in the future, restarting all the process, but now, making a defection from 1 to (n+1), and then, two "Cooperate" actions***, and so on;

2. Hybrid Gradual Strategy

This strategy it's similar to the first one, ***but with the difference that, uses a harder concept of forgiveness***. In the normal version of the Gradual Strategy, the "calm down" it's always made by two consecutive "Cooperate" actions after nth "Defect" accordingly to the nth "Defect" of the opponent. **This version, follows the same definition of the Gradual Strategy before the first 8 initial "Defect" actions made by the opponent**. ***After that, the "calm downs" can be made with the support of just one or two consecutive "Cooperate" actions, it will depend on the current number of consecutive "Cooperate" actions made by the opponent***. ***If the opponent made such or more than 6 consecutive "Cooperate" actions, it will be applied the "soft calm down" with half of the number of "Defect" actions in the base strategy but, with the same two "Cooperate" actions, if not, will be applied the "hard calm down" with the number of "Defect" actions like the normal version but with just one "Cooperate" action, in order, to recover the utility points loss***. In some cases, it will be verified, ***if the opponent made the "Cooperate" action such or more than 4 consecutive times and in that case, since I'm not currently making "Defect" actions neither "calming down", I will respond with "Defect" actions, i order to gain utility points over my opponent and, fast recover, in sometimes. In the last iteration, since that's known, will be made a "Defect" action***;

3. Hybrid Gradual With Cooperate Leeway Strategy

This strategy differs in comparison to the previous one, in the way, that start with a "Defect" action. And additionally, ***in the case of, I'm not currently making "Defect" actions neither "calming down", I will respond with a "Cooperate" action since the opponent have more than the double of "Defect"actions made in comparison to the number of "Cooperate" actions made, until the moment. And will respond with a "Defect" action, otherwise. It's just a way, to try do some "Cooperate" actions, in the case of, I have some kind of "secure" leeway to do some "Cooperate" actions without get some penalty related with a loss of utility points. Furthermore, will be played the "Defect" action, if the probability of continue playing or iterating be lesser than (⅓) = 33,3333%. If it's known if it's being run the last iteration, will be made a "Defect" action.***

Some experimental tests' results [1]

- *Relevant 5 experimental tests performed with 1 000 Iterations and 100% of Probability of Continue to Iterate, against other Strategies:*

Num. of Test	Num. of Iterations	Probability of Continue [0.0 - 1.0]	Strategy used by Player #1	Strategy used by Player #2	Total of Points of Player #1	Total of Points of Player #2	Winner
Test #1	1 000	1.0	Random Strategy	Hybrid Gradual Strategy	1 110	4 998	**Player #2 (Hybrid Gradual Strategy)**
Test #2	1 000	1.0	Random Strategy	Hybrid Gradual Strategy	1 072	5 032	**Player #2 (Hybrid Gradual Strategy)**
Test #3	1 000	1.0	Mimic Strategy (Modified Tit For Tat Strategy)	Hybrid Gradual Strategy	2 057	2 061	**Player #2 (Hybrid Gradual Strategy)**
Test #4	1 000	1.0	Gradual Strategy	Hybrid Gradual Strategy	2 050	2 106	**Player #2 (Hybrid Gradual Strategy)**

Table 2: Experimental results with 1 000 iterations and 100% of probability of continue to iterate

The Strategy used for the first round of the Tournament - ***Hybrid Gradual Strategy***:

- As, accordingly with the experimental tests' results, for the **first round of the tournament**, was chosen the ***Hybrid Gradual Strategy*** (*see the code in annex*);

Some experimental tests' results [2]

- *Relevant 5 experimental tests performed with 10 000 000 Iterations and 80% of Probability of Continue to Iterate (considered only Games that achieved such or more than 10 Iterations), against other Strategies:*

Num. of Test	Num. of Iterations	Probability of Continue [0.0 - 1.0]	Strategy used by Player #1	Strategy used by Player #2	Total of Points of Player #1	Total of Points of Player #2	Winner
Test #1	10 000 000 Possible Iterations (14 Iterations Completed)	0.8	Random Strategy	Hybrid Gradual With Cooperate Leeway Strategy	**51**	**63**	**Player #2 (Hybrid Gradual With Cooperate Leeway Strategy)**
Test #2	10 000 000 Possible Iterations (14 Iterations Completed)	0.8	Random Strategy	Hybrid Gradual With Cooperate Leeway Strategy	**53**	**57**	**Player #2 (Hybrid Gradual With Cooperate Leeway Strategy)**
Test #3	10 000 000 Possible Iterations (12 Iterations Completed)	0.8	Mimic Strategy (Modified Tit For Tat Strategy)	Hybrid Gradual With Cooperate Leeway Strategy	**56**	**56**	------------
Test #4	10 000 000 Possible Iterations (17 Iterations Completed)	0.8	Gradual Strategy	Hybrid Gradual With Cooperate Leeway Strategy	**102**	**102**	------------
Test #5	10 000 000 Possible Iterations (13 Iterations Completed)	0.8	Hybrid Gradual Strategy	Hybrid Gradual With Cooperate Leeway Strategy	**54**	**54**	------------

Table 3: Experimental results with 10 000 000 iterations and 80% of probability of continue to iterate

The Strategy that will supposed to be used for the second and third round of the Tournament - ***Hybrid Gradual With Cooperate Leeway Strategy***:

- As, accordingly with the experimental tests' results, for the **second and third round of the tournament**, was chosen the ***Hybrid Gradual With Cooperate Leeway Strategy*** (*see the code in annex*);

Conclusions

The Prisoners' Dilemma isn't an easy computational game to solve, right?! ***The true it's that, there's no specific and "universal" strategy to solve it***. ***We can conclude that, perhaps, playing always in "defence", doing "Defect" actions always, instead of, "Cooperate" actions, can be a good strategy on games of type 1 vs. 1, because you have no round that you don't have any gain of utility, gaining always 1 or 4 utility points***.

But in, an environment where we are playing against more than one opponent in iterated games and considering the average of utility gained in every games, perhaps isn't the best strategy to use, because another two players, can be, possibly, cooperating, and gaining both, 3 utility points. And in some situations, can have more points, in average, than us.

So, maybe, isn't too bad idea, cooperate some times to try to maximise our global utility, since our opponent isn't doing "Defect" actions.

We can think that, implement a strategy that takes that in consideration, maybe should be the best approach for this situations in this computational game, since, of course, tries to avoid, the situations where we do "Cooperate" actions and our opponents do "Defect" actions.

Furthermore, ***winning the most number of single games, during a tournament of iterated games using Round Robin, doesn't implies the maximum global points average and consequently winning of the tournament***.

Was proved that, the Prisoners' Dilemma it's a very hard computational game to solve in an universally way. ***This game presents a very complex paradox between individual interests and the common good***. Maybe, the only way to "win" is to change this game itself, and that's should be the larger lesson to learn from the Prisoners' Dilemma. ***A good solution, thinking in a global well-being, would be define with the people involved that, everyone should be together and agree that cooperating is the best solution, and then, everyone agrees to cooperate, and most importantly, everyone agrees that anyone who "betrays" someone will be punished by the collective group*** when they get out of prison. ***Then, that will change the payoff table so that betraying costs more and, cooperating is the better selfish choice and better choice for everyone***.

In life, nothing it’s easy, like, per example, tattoo a map of the plant of a prison in the body to plan a successful escape from a prison where our partners in prison cooperates and helps each other in our plan, such as the story of Michael Scofield in the famous TV serie Prison Break… ;)

Some Bibliography and References

- https://en.wikipedia.org/wiki/Prisoner%27s_dilemma
- https://www.quora.com/What-is-the-best-strategy-for-the-Iterated-Prisoners-dilemma-game
- http://jasss.soc.surrey.ac.uk/20/4/12.html#toc-conclusion

Further Information

The Java implementations used for the strategies applied to this computational game, will be provided in the next pages and you can check it the following ***GitHub’s repository*** (such as, other additional strategies used):

- https://github.com/rubenandrebarreiro/computational-game-theory-tournaments/tree/master/temp/1st%20Tournament%20-%20Prisoners'%20Dilemma