New Winning Strategies for the Iterated Prisoner's Dilemma

(Extended Abstract)

Philippe Mathieu
CRIStAL Lab. UMR 9189 CNRS
Lille University
Villeneuve d'Ascq, France
philippe.mathieu@univ-lille1.fr

Jean-Paul Delahaye
CRIStAL Lab. UMR 9189 CNRS
Lille University
Villeneuve d'Ascq, France
jean-paul.delahaye@univ-lille1.fr

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Multiagent systems*

General Terms

Algorithms, Experimentation

Keywords

Game theory, group strategy, iterated prisoner's dilemma, IPD, agent's behaviour, memory, opponent identification

1. INTRODUCTION

In the iterated prisoner's dilemma game, new successful strategies are regularly proposed especially outperforming the well-known *tit_for_tat* strategy. New forms of reasoning have also recently been introduced to analyse the game. They lead W. Press and F. Dyson [4] to a double infinite family of strategies that -theoretically- should all be efficient strategies. We study and confront using severals experimentation the main strategies introduced since the discovery of *tit_for_tat*. The iterated prisoner's dilemma is a game that leads to understand various basic truths about social behaviour and how cooperation between entities is established and evolves. Several studies [1, 4] have led to consider other strategies than the famous *tit_for_tat*. We have begun to make a balance of the situation with the desire to reach clear and as unbiased as possible conclusions. Our method is based on three main ideas, each converging on robust results. (1) Confronting the candidate strategies on the principle of the `tournament` (mainly for information) and the method of `ecological competition` which gives results independent from initial conditions. (2) Using sets of strategies in which all strategies of a particular class (eg using the last move of past of each player) are in competition. This method of `complete classes` [2] avoids any subjective choice. (3) Taking a phased approach by not trying to find the best of all strategies in absolute terms, but by combining the results of progressive massive confrontation experiments.

Appears in: *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2015), Bordini, Elkind, Weiss, Yolum (eds.), May, 4–8, 2015, Istanbul, Turkey.*

2. RULES OF THE GAME

The prisoner's dilemma is that accorded to two entities with a choice between cooperation (`c`) and defection (`d`) and are remunerated by `R` points each if each plays `c`, `P` points if each plays `d` and receiving `T` respectively `S` points if one plays `d` and the other `c`. We describe these rules by writing: `[c, c] -> R + R , [d, d] -> P + P , [d, c] -> T + S`

In our experiments with use the classical values `T=5, R=3, P=1, S=0` and 1000 rounds for each meeting.

We make a distinction between deterministic strategies and probabilistic strategies (where choices can depend on chance). The study of literature about the dilemma led us to define a set of 17 basic deterministic strategies (including the simplest imaginable strategies). We have added 13 probabilistic strategies mainly taking into account the recent discoveries of Press and Dyson on extortion [4].

Let us present the set of 17 basic strategies. **all_c**: always cooperates. **all_d**: always defects. **tit_for_tat**: cooperates on the first move then plays what its opponent played the previous move. **spiteful**: (also called *grim*) cooperates until the opponent defects and thereafter always defects. **soft_majo**: begins by cooperating and cooperates as long as the number of times the opponent has cooperated is greater that or equal to the number of times it has defected; otherwise she defects. **hard_majo**: defects on the first move and defects if the number of defections of the opponent is greater than or equal to the number of times she has cooperated; else she cooperates. **per_ddc**: plays ddc periodically. **per_ccd**: plays ccd periodically. **mistrust**: (also called *suspicious_tft*) defects on the first move then play what my opponent played the previous move. **per_cd**: plays cd periodically. **pavlov**: (also called *win-stay-lose-shift*) cooperates on the first move and defects only if both the players did not agree on the previous move. **tf2t**: cooperates the two first moves, then defects only if the opponent has defected during the two previous moves. **hard_tft**: cooperates the two first moves, then defects only if the opponent has defected one of the two previous moves. **slow_tft**: cooperates the two first moves, then begin to defect after two consecutive defections of its opponent; returns to cooperation after two consecutive cooperations of its opponent. **gradual**: cooperates on the first move, then defect n times after n^{th} defections of its opponent, and calms down with 2 cooperations [1]. **prober**: plays the sequence `d,c,c`, then always defects if its opponent has cooperated in the moves 2 and 3; plays as `tit_for_tat` in other cases. **mem2**: behaves like *tit_for_tat* : in the first two moves, and then shifts among three strategies *all_d*, *tit_for_tat*, *tf2t* [3].

Memory(X,Y) is the complete class which is the class of all deterministic strategies using my X last moves and the Y last moves of my opponent. In each Memory(X,Y) complete class, all deterministic strategies can be completely described by their "genotype" i.e. a chain of C/D actions to do that begin with the $max(X, Y)$ first moves i.e. not depending on the past. These starting actions are written in lower case. The list of cases of the past is sorted by lexicographic order on my X last moves (from the older to the newer) followed by my opponent's Y last moves (from the older to the newer).

Our platform has allowed us to compete in tournament and ecological competitions families of 1000 and even 2000 strategies. Our experiments using large complete classes led us to discover four new strategies :
winner12 (mem12_ccCDCDDCDD), **winner21** (mem21-dcCDCDCDDD), **spiteful_cc** which is classical spiteful but with a cc forced start, **tft_spiteful** which starts with c, then plays *tit_for_tat* unless she has been betrayed two times consecutively, in which case she always betrays (plays *all_d*).

The 17 basic + 4 new strategies : The experiment A involves the 17 basic strategies with these 4 new strategies.

Tournament ranking			Ecological ranking		
1	spiteful_cc	58981	1	spiteful_cc	236
2	gradual	58814	2	gradual	235
3	tft_spiteful	57486	3	tft_spiteful	207
4	winner12	57072	4	winner12	200
5	slow_tft	55821	5	slow_tft	175
6	tit_for_tat	55666	6	tf2t	169
7	tf2t	55156	7	tit_for_tat	163
8	hard_tft	54679	8	all_c	150

It is remarkable that three among the four new introduced strategies are in the 4 first ecological ranking.

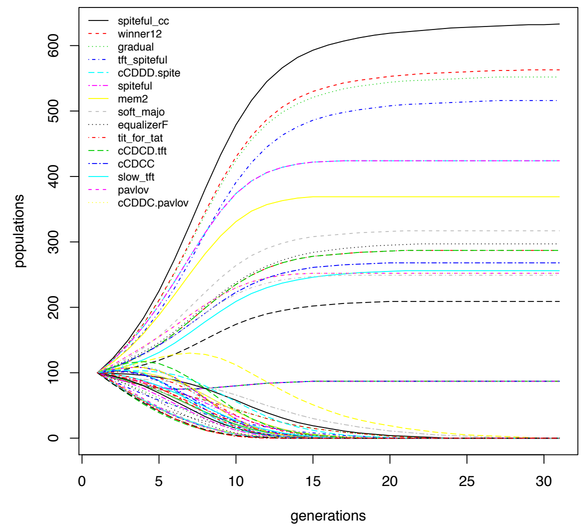
All deterministic + 4 new strategies : For this experiment B we add the Memory(1,1) complete class. This leads to a set of 53 strategies (17 + 32 + 4 new).

Tournament ranking			Ecological ranking		
1	spiteful_cc	152873	1	spiteful_cc	552
2	winner12	149466	2	winner12	493
3	spiteful	148934	3	gradual	480
4	cCDDD-spite	148934	4	tft_spiteful	461
5	gradual	148676	5	cCDDD-spite	359
6	mem2	144941	6	spiteful	359
7	tft_spiteful	144068	7	mem2	320
8	pavlov	132712	8	tit_for_tat	258

This time the four winners are exactly the same as in the previous experiment B but not exactly in the same order. This result shows the robustness of these four strategies.

All deterministic and probabilistic : This experiment C is built with all the basic deterministic startegies obtained with the 17 initial basic strategies and the Memory(1,1) complete class added with 13 probabilistic strategies coming from [4] and the four new strategies discovered thanks to the complete classes experiments. This leads to a set of 66 strategies.

Tournament ranking			Ecological ranking		
1	spiteful_cc	8708014	1	spiteful_cc	639
2	winner12	8588922	2	winner12	574
3	gradual	8540987	3	gradual	562
4	spiteful	8511448	4	tft_spiteful	518
5	cCDDD-spite	8510088	5	cCDDD-spite	421
6	tft_spiteful	8274128	6	spiteful	420
7	mem2	8115054	7	mem2	347
8	pavlov	8013314	8	soft_majo	319



These experiments show clearly that *winner21* seems less robust than the three other stategies.

3. CONCLUSION

According to the state of the art, we have collected the most well-known interesting strategies. Then we have used the systematic and objective complete classes method to evaluate them. These experiments led us to identify seven efficient and robust strategies: ***spiteful_cc, winner12, gradual, tft_spiteful, spiteful, mem2, soft_majo***. We note that they are almost all mixtures of two basic strategies : *tit_for_tat* and *spiteful*. This suggests that *tit_for_tat* is not severe enough, that *spiteful* is a little too much and that finding ways to build hybrids of these two strategies is certainly what gives the best and most robust results. We also note that using information about the past beyond the last move is helpful. Among the seven strategies that our tests put in the head of ranking some of them use the past from the beginning (*gradual* and *soft_majo*) and all the others use two moves of the past or a little more. This work illustrates the fact that using complete classes of increasingly size will allow to identify increasingly efficient strategies, and provides a broad framework to find new ones.

4. REFERENCES

[1] B. Beaufils, J.-P. Delahaye, and P. Mathieu. Our meeting with gradual, a good strategy for the iterated prisoner's dilemma. In *Proc of ALIFE'5 conf.*, pages 202–209. The MIT Press, 1996.

[2] B. Beaufils, J.-P. Delahaye, and P. Mathieu. Complete classes of strategies for the classical iterated prisoner's dilemma. In *Proc EP'7 conf.*, volume 1447 of *LNCS*, pages 33–41. Springer, 1998.

[3] J. Li and G. Kendall. The effect of memory size on the evolutionary stability of strategies in iterated prisoner's dilemma. *Evolutionary Computation, IEEE Transactions*, PP(99):1–8, 2013.

[4] W. H. Press and F. J. Dyson. Iterated prisoner's dilemma contains strategies that dominate any evolutionary opponent. *PNAS*, 109(26):10409–10413, 2012.