Forgiveness in the Iterated Prisoner's Dilemma and Cooperation in Multi-Agent Systems

Colm O' Riordan,
colmor@geminga.nuigalway.ie
Phone: 353 91 524411 ext 3669
email: colmor@geminga.nuigalway.ie
Fax: 353 91 750501

Dept. of Information Technology,
National University of Ireland,
Galway, Ireland.

Abstract. This paper discusses results in game-theoretic modelling of agent-based systems. The paper includes a brief discussion of cooperation in multi-agent systems. We concentrate on the prisoner's dilemma and the iterated prisoner's dilemma as an abstraction of some of the salient features involved in repeated interactions of agents in a multi-agent system. The paper describes results of a new family of strategies in the setting of the iterated prisoner's dilemma and discusses its applicability to multi-agent systems.

1 Introduction

This paper discusses results in game-theoretic modelling of agent-based systems. The paper includes a discussion of cooperation in multi-agent systems. We concentrate on the prisoner's dilemma and the iterated prisoner's dilemma as an abstraction of some of the salient features involved in repeated interactions of agents in a multi-agent system. The paper describes results of a new strategy in the setting of the iterated prisoner's dilemma and discusses its applicability to multi-agent systems.

Section 2 discuss agent and multi-agent systems, paying particular attention to models of coordination and cooperation. The next section discusses well-known games such as the prisoner's dilemma and the iterated prisoner's dilemma. Section 4 discusses results obtained, with a new strategy, in the iterated version of the game. Section 5 discusses this strategy and its applicability to multi-agent systems.

2 Multi-Agent Systems

2.1 Introduction

The term *agent* has become one of the more pervasive buzzwords over the past few years. The number of products and companies using, or claiming to use, ‘agent’ technology has steadily increased; this trend seems set to continue.

The existence of the many various definitions and interpretations of agents, is due mainly to the fact that numerous classes of agent exist, each with their own set of properties. The concept of an agent was first introduced by Hewitt, in his *Actor Model* [10]. Today, numerous strands of agent research exist, each applying different methodologies to different types of problems. Hence, the difficulty in defining the term *agent*. Nwana [14] uses different means of classifying agents—mobility, reactiveness, possession of certain basic properties and, finally, classification by the role they fulfill.

Much work has been carried out in the multi-agent systems arena largely concentrating on the development and deployment of such systems. To achieve the desired goal of intelligent, adaptive, potentially distributed solutions in complex environments much effort has been applied to the development of agent theories, architectures(reactive, BDI) agent languages (AGENT0 etc), packages (Agglets, JINI), communication languages (KQML, FIPA ACL) and coordination protocols (contract net protocol, negotiation strategies).

Irrespective of the approach adopted, the potential for complex emergent behaviour exists. The behaviour is likely to be seen if the agents are allowed to adapt (learn) over time, or if the involved agents are be improved over time.

2.2 Cooperation in Multi-agent Systems

Introduction Thomas Hobbes presented a rather pessimistic explanation as to how cooperation can be maintained in a group of interacting agents; he argued that prior to the existence of governments nature was dominated by selfish agents resulting in a life that was “solitary, poor, nasty, brutish and short”. He believed that without a controlling authority, cooperation was impossible.

Olson [16] agreed with the classical conclusion that coercion or selective incentives are necessary to achieve cooperation.

Subsequent work in the field of game theory and related fields have indicated that this is not necessarily the case.

Multi-Agent Systems Much work in multi-agent systems has, either explicitly or implicitly, made a number of assumptions regarding the nature of the agents and the communication allowed between them. Common assumptions of benevolence and altruism of the agents and of a noise-free communication channel between the agents are not necessarily valid for many real-world scenarios.

It is easy to envisage an environment (e.g. a set of trading agents) where each agent has their own goals to satisfy. Satisfying these goals may involve actively not cooperating with other agents, while possibly maintaining some degree of benevolence to maintain the fitness of the society as whole. The rewards obtained by the agents is dependent on their own choice and the choice of the other agents.

Scenarios of this type have often been modelled as games such as the prisoner's dilemma and the Voter's paradox.

3 Prisoner's Dilemma and the Iterated Prisoner's dilemma

In the prisoner's dilemma game, there are two players who are both faced with a decision—to either cooperate or defect. The decision is made by a player with no knowledge of the other player's choice. If both cooperate, they receive a specific punishment (or reward). If both defect they receive a larger punishment. However, if one defects, and one cooperates, the defecting strategy receives no punishment and the cooperator a punishment (the sucker's payoff). The game is often expressed in the canonical form in terms of pay-offs:

	Player 1	
	C	**D**
Player 2	**C** (3,3)	(0,5)
	D (5,0)	(1,1)

where the pairs of values represent the pay-offs for players `Player 1` and `Player 2` respectively. The prisoner's dilemma is much studied problem due to it's far-reaching applicability in many domains. In game theory, the IPD can be viewed as a two-person, non-zero-sum, non-cooperative and simultaneous game. In order to have a dilemma $S < P < R < T$ must hold where S is the sucker's payoff, P is the punishment for mutual defection, R is the reward for mutual cooperation and T is the temptation to defect. The prisoner's dilemma and applications has been described in [6][7][12] (biology) [17] (economics) and [4] (politics).

More interesting is the iterated version where 2 players will play numerous games (the exact number not known to either player). To favour cooperation the following inequality should hold: $S + T < 2R$. Note that some research has indicated that it is not necessary to look to the iterated versions for more interesting behaviour to occur. Work by Epstein [8] into spatial zones indicate that more interesting behaviour (e.g mutual cooperation) can emerge and exist in the non-iterated version of the game.

Similar work on the effect of spatial organisation of strategies was undertaken by Oliphant [15] who showed, via a series of simulations, that in the absence of spatial constraints, the population quickly fell into defection; obversely, spatial populations were able to evolve and maintain cooperative behaviour.

A computer tournament (Axelrod)[1] was organised to pit strategies against each other in a round-robin manner. The winning strategy was *tit-for-tat*; this strategy involved cooperating on the first move and then mirroring opponents moves on all subsequent moves.

No best strategy exists; the success of a strategy depends on the other strategies present. For example, in a collection of strategies who defect continually (*ALL-D*) the best strategy to adopt is *ALL-D*. In a collection of strategies adopting a *tit-for-tat* strategy, an *ALL-D* strategy would not perform well.

Strategies It is instructive to look at other approaches and classify the strategies. The list is by no means exhaustive:

- **periodic:** strategies play C or D in a periodic manner. Common strategies: *ALL-C*, *ALL-D*, *CD**, *DC**, *CCD**, etc.
- **random**: strategies that have some random behaviour. Totally random, or one of the other types (e.g. periodic) with a degree of randomness.
- **based on some history of moves:** *tit-for-tat* (C initially, then D if opponent defects, C if opponent cooperates), *spiteful* (C initially, C as long as opponent cooperates, then D forever), probers (play some fixed string, example (DDC) and then decides to play *tit-for-tat* or *ALL-D* (to exploit non-retaliatory), *soft-majo* (C initially, then cooperate if opponent is not defecting more than cooperating).

There are many variations on the above type of strategies.

The initial results and analysis (which were echoed in later tournaments) showed that the following properties seemed necessary for success—niceness (cooperate first), retaliatory, forgiving and clear.

In a second tournament [1], of the top 16 strategies, 15 were found to be nice. These results seem to indicate that cooperative strategies are useful if there is a high chance the strategies will meet again.

Beaufils et al [2] question that last property and develop a strategy *gradual*[1] which is far more complex than *tit-for-tat* and outperforms *tit-for-tat* in experiments.

[1] gradual performs like *tit-for-tat*, in that it cooperates on the first move. It retaliates upon defection. On the first defection it responds with a defection, followed by 2 cooperations. Following the second defection, it responds with 2Ds, followed by 2Cs and so forth.

4 The *forgiving* approach

If we look at *tit-for-tat*, we can identify techniques to overcome it's shortcomings. There are certain aspects where new heuristics (with parallels in human interactions in the real world) could be used to improve upon 'weak' parts of *tit-for-tat's* performance.

If we consider *(CD)** against *tit-for-tat*, we get:

```
tit-for-tat: C C D C D C D C D C . . .
(CD)*:       C D C D C D C D C D . . .
```

The *per-cd* repeatedly plays a C followed by a D irrespective of *tit-for-tat*'s move. This results in the two strategies alternating between receiving the maximum payoff and the sucker's payoff.

Tit-for-tat and other strategies could improve performance by recognising non-nice, naive strategies. Upon recognition of these strategies, a less forgiving approach can be adopted. The goal here is to be more far-sighted with respect to clear non-nice strategies.

Another (and possibly more serious) shortcoming of *tit-for-tat* is the prevalence of spiraling into ongoing retaliation. Consider *tit-for-tat* player against *nasty-tit-for-tat* (*tit-for-tat* but attempts to exploit non-retaliatory strategies by playing a `DD` with some probability. Note this can also happen with *tit-for-tat* in noisy environments).

```
tit-for-tat       C C C C C C C C C C C D D D...
nasty-tit-for-tat C C C C C C C C C C D D D D . . .
```

Once *nasty-tit-for-tat* plays 2 `Ds`, the two strategies are locked in a spiral of mutual defections. This will not be broken until one of the two strategies plays two `Cs`.

Our strategy attempts to take these two factors into account:

- Don't be exploited by periodic strategies
- Try to re-establish cooperation by forgiving

Note that the above modifications do not violate the first three recommendations (generally accepted) forwarded by Axelrod—never defect first, be retaliatory, be forgiving. There are cases when the exploitation of periodic strategies can damage performance: where a pattern is recognised as a periodic strategy and we adopt an `ALL-D` approach to avoid exploitation. This can quickly result in a spiral of mutual defections if the opponent is not really periodic (but appears to be).

The degree of forgiveness in our strategy is of length 2, i.e we play two consecutive Cs. The length of the DD strings is set to 5 (i.e., once 5 pairs of defections are encountered an effort is made to re-establish cooperation).

In summary, our strategy is tit-for-tat like, with the following amendments—exploit periodic strategies and forgive when interactions are spiraling into an on going defection.

4.1 Results

The initial experiments carried out with the strategy was in a round-robin tournament with 37 other well-known strategies. The strategies chosen were those included as default in the simulation package created by Beaufils and Delahaye[2]. This very useful package allows experiments in both a round-robin and an evolutionary setting. A set of well-known strategies are included in the package and the addition of new strategies is facilitated. We added our strategies to the pool of strategies provided.

The initial results, Table 1, showed that the strategy performed well, reaching a position of second. The only strategy that outperformed it was the *gradual* strategy. We see, however, in the next section that in an environmental setting the *forgiving* strategy performs better.

1	gradual	106277
2	forgiving	104086
3	soft_spiteful	102553
4	soft_joss	102487
5	c_then_per_dc	102345
6	hard_prober	102341
7	tit_for_tat	100319
8	doubler	99409
9	prober4	97761
10	worse_and_worse3	95484

In the evolutionary simulation, each successive generation contains strategies with frequency proportional to their score in the current generation. Again, 38 strategies were included initially. The performance of the strategies over a number of generations was plotted (Figure 1). We also include the top ten results following the first and twenty-first generations:

	Strategy	Generation 1 proportion
1	gradual	119
2	forgiving	116
3	soft_spiteful	115
4	soft_joss	114

[2] Available at http://www.lifl.fr/IPD/ipd.frame.html

5	c_then_per_dc	114
6	hard_prober	114
7	tit_for_tat	112
8	doubler	111
9	prober4	109
10	worse_and_worse3	107

	Strategy	Proportion in next generation
1	forgiving	449
2	gradual	419
3	soft_joss	393
4	soft_spiteful	388
5	tit_for_tat	335
6	doubler	309
7	c_then_per_dc	263
8	soft_tf2t	206
9	hard_prober	155
10	worse_and_worse3	143

This trend continues as illustrated in Figure 1, with *forgiving*'s representation in the population greater than that of any of the others.

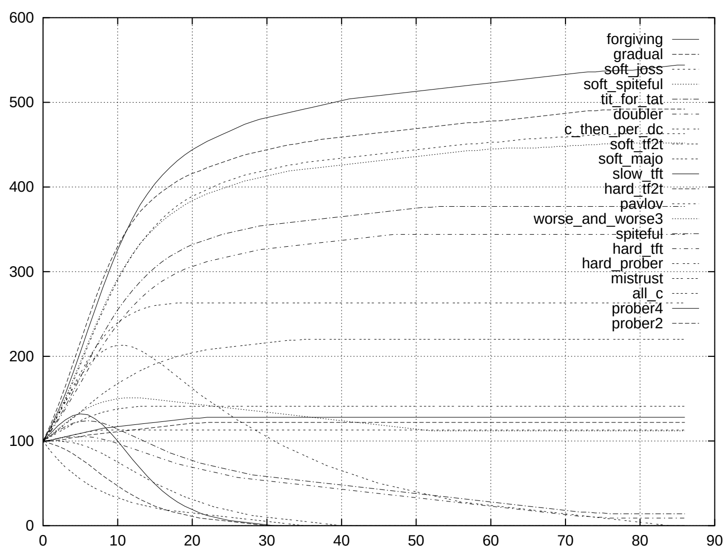

Fig. 1. Evolutionary Setting

Following these initial results we wished to investigate which of the two aspects of the *forgiving* strategy accounted for its good performance (its exploitation of periodic strategies or its ability to re-establish cooperation). The performance of the strategy in the environmental setting indicates that its exploitation of periodic strategies, while useful, is not necessary for its success as the strategies (periodic) upon which it preys die off at a relatively early stage (e.g., *ALL-D*, *per-ddc*).

To provide empirical evidence, we also include two variations—*forgiving-1* which does not exploit periodic strategies but attempts to re-establish cooperation and *forgiving-2* which attempts to exploit periodic strategies only. The graph in Figure 2 shows their performance. As can be seen, the two strategies that attempt to forgive and re-establish cooperation flourish.

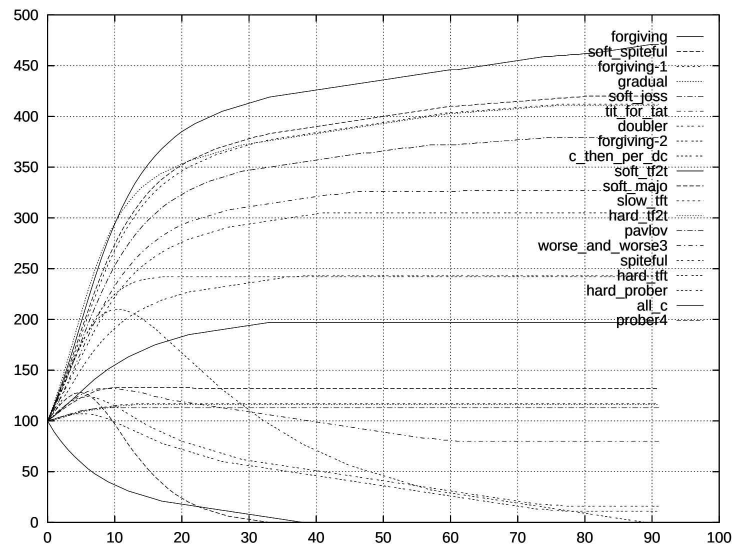

Fig. 2. Evolution of different versions of forgiving

4.2 Evolutionary Search for *forgiving* strategies

The above evidence provides some justification for the incorporation of forgiveness into strategies. To explore more fully the effect of degrees of forgiveness, an evolutionary computation method has been adopted. We use a genetic algorithm to breed successful strategies to play against a well-known collection of strategies (the same used in the above strategies).

Evolutionary computation approaches have been used previously by other researchers to explore the range of strategies. These include Nowak and Sigmund[13], Linster[11], Beaufils[3], Harrald and Fogel[9], Cohen et al.[5].

In our initial experiments, we encode, in each chromosome, five different aspects of a strategy's behaviour. These include:

- behaviour on the first move
- behaviour following a defection by opponent
- behaviour following a cooperation by opponent
- number of mutual defections to allow before entering to *forgiving* behaviour
- number of successive cooperative gestures to make in order to *forgive*

On all trials the first three genes converge extremely quickly and confirm findings by many others—cooperate on first move, cooperate following a cooperation and defect following a defection.

The fifth gene does not converge to zero in any of the experiments, indicating a *forgiving* behaviour is a useful feature for any strategy to maintain. The degree of *forgiveness* and the length of the spiral have varied in the trials but are much larger than those in our designed strategy.

Further experimentation is required to incorporate some other features and to test strategies in noisy environments where one would expect to see a higher number of spirals of defection.

5 Application to Multi-Agent Systems

5.1 Introduction

In the area of agent-based systems, particularly that of multi-agent systems, we believe that the properties of *forgiving*-like strategies can be useful. The goal in multi-agent systems is to allow the design and development of flexible, dynamic, robust, possibly distributed solutions to complex problems in a dynamic environment. The systems usually comprise a set of autonomous, adaptive, possibly heterogeneous agents who can communicate with each other.

The vast majority of work assumes the following properties:

- benevolent agents: each agent is working for the common good.
- absence of noise: that the agents are operating in a noise-free environment; i.e., no message from one agent to another is corrupted or that no agent misinterprets the communicated message.

We believe these assumptions are not valid in all scenarios. If one is attempting to build truly flexible software, where new agents with new functionality can be added or removed, and if these agents are autonomous, the a priori assumption of benevolence is not valid.

Misinterpretation of behaviour or messages regarding behaviour, goals etc., cannot be assumed to work in a noise-free environment. Conflicting goals, conflicting representations, communication primitives etc. can all lead to a noisy environment.

The forgiving strategy helps agents (and the society of agents in general) obtain a high score. The strategy, due to its willingness to forgive has proved useful in noisy environments.

5.2 Example Domain

It is easy to envisage the following scenario arising: a set of agents are used to provide information from a set of diverse sources (web-based, document repositories, information feeds, databases). Furthermore agents within the community may combine and collate information provided by other agents. Agents may provide a set of services—either information or interfaces to query the information.

Some agents have consumers, who pay for the agent-provided information. Each agent is trying to maximise their profit; by selling information (and access to information) to other agents or human consumers. Hence, there exist a number of interactions between agents.

The content of the information provided can be verified to a degree by IR techniques (vector space model, latent semantic indexing etc.) or via the use of structural clues and tags. The quality of the information produced by the agents is gauged by the users.

The provision of quality information (relevant, high-quality, timely, at a reasonable price) can be seen as a cooperative act; failure to do so can be seen as a defection. The payment (timely) can also be seen as a cooperative act; defection would correspond to the failure to pay. Thus, we can see the society of agents as being involved in a series of 2-player interactions.

In this scenario, each agent has the motivation to defect: avoid paying and thereby increase profit and/or to sell sub-standard information and hence increase profit. So, the assumption of benevolence is not valid. We believe, in scenarios like this, where new agents may join or leave the society, that a *forgiving* like strategy would be successful.

5.3 Conclusion

In this paper we discussed a strategy for the iterated prisoner's dilemma which in experiments has performed well. We believe that the properties of this strategy can be useful in modelling and as an aid in designing truly multi-agent systems.

The ability to promote cooperation by never being the first to defect but also by re-establishing cooperation is a useful feature which may lead to more fit societies of agents.

Current and future work involves further experimentation with noisy environments, and with a continuous range of actions ranging from cooperation to defection. We are currently developing an multi-agent information trading environment where subsets of the agents are guided by *forgiving*-like strategies.

References

1. R. Axelrod. *The Evolution of Cooperation*. Basic Books, New York, 1984.
2. B. Beaufils, J. P. Delahaye, and P. Mathieu. Our meeting with gradual, A good strategy for the iterated prisoner's dilemma. In C. Langton, editor, *Artificial Life, Proc. 5th International Workshop on the Synthesis and Simulation of Living Systems*, volume 5, 1996. Artificial Life 5, Nara, Japan, May 16-18.
3. B. Beaufils, P. Mathieu, and J.-P. Delahaye. Complete classes of strategies for the classical iterated prisoner's dilemma. In *Proceedings of Evolutionnary Programming VII, LNCS vol 1447*, pages 33–41. Springer-Verlag, 1998.
4. J. Bendor and D. Mookherjee. Institutional structure and the logic of ongoing collective action. *American Political Science Review*, (81):129–154, 1987.
5. M. D. Cohen, R. L. Riolo, and R. Axelrod. The emergence of social organization in the prisoner's dilemma: How context-preservation and other factors promote cooperation. Technical report, SantaFe Institute, 1999.
6. L. Dugatkin. Do guppies play tit for tat during predator inspection visits? *Behavioral Ecology and Sociobiology*, 2(3):395–399, 1988.
7. L. Dugatkin. N-person games and the evolution of co-operation: A model based on predator inspection in fish. *Journal of Theoretical Biology*, (142):123–135, 1990.
8. J. M. Epstein. Zones of cooperation in demographic prisoner's dilemma. Technical report, Santa Fe Institute, 1997.
9. P. Harrald and D. Fogel. Evolving continuous behaviours in the iterated prisoner's dilemma. *Biosystems*, 37:135–145, 1996.
10. C. Hewitt. Viewing control structures as patterns of passing messages. *Artificial Intelligence*, 8(3):323–364, 1977.
11. B. Linster. Evolutionary stability in the infinitely repeated prisoner's dilemma played by two-state moore machines. *Southern Economic Journal*, (58):880–903, 1992.
12. M. Lombardo. Tree swallows and tit for tat: Response to koenig. *Ethology and Sociobiology*, 11:521–528, 1990.
13. M. Nowak and K. Sigmund. The evolution of stochastic strategies in the prisoner's dilemma. *Acta Applicandae Mathematicae*, (20):247–265, 1990.
14. H. S. Nwana. Software agents : An overview. *Knowledge Engineering Review*, 11(3):1–40, 1996.
15. M. Oliphant. Evolving cooperation in the non-iterated prisoner's dilemma: The importance of spatial organization. pages 349–352, 1998.
16. M. Olson. *The Logic of Collective Action: Public Goods and the Theory of Groups*. Harvard University Press, 1965.
17. E. Sober. Stable cooperation in iterated prisoner's dilemmas. *Economics and Philosophy*, 8:127–139, 1992.