Partial Imitation Rule in Iterated Prisoner Dilemma Game on a Square Lattice

Degang Wu, Mathis Antony, and K.Y. Szeto⋆

Abstract. A realistic replacement of the general imitation rule in the Iterated Prisoner Dilemma (IPD) is investigated with simulation on square lattice, whereby the player, with finite memory, can only imitate those behaviors of the opponents observed in past games. In contrast to standard practice where all the possible behaviors of the opponents are accessible, the new partial imitation rule assumes that the player can at most access those behaviors of his opponent observed in the past few moves. This partial imitation of the behavior in IPD shows very different out-comes in the long time behavior of the games, such as the ranking of various strategies. In particular, the well known tit-for-tat (TFT) strategy loses its importance.

1 Introduction

Game theory [1] has attracted the attention of many scientists working in complex systems as an experimental playground in computer simulation of multi-agent systems is now easily available [2]. Since the introduction of evolutionary game theory by Maynard Smith and Price [3, 4], one of the important issues of this theory is to understand the spontaneous cooperation towards a more efficient outcome with agent interactions in the absence of a central planner [5, 6]. Among the many games, the most studied example by political scientists and sociologists is the Prisoner's Dilemma, as it provides a simple example of the difficulties of cooperation [7]. Prisoner Dilemma (PD) is described by the following set of rules. When two players play a PD game, each of them can choose to cooperate (C) or defect (D). Each player will gain a payoff depending jointly on his choice and the opponent's choice.

Degang Wu · Mathis Antony · K.Y. Szeto
Department of Physics,
Hong Kong University of Science and Technology,
Clear Water Bay, Hong Kong, HKSAR
e-mail: phszeto@ust.hk

⋆ Corresponding author.

J.R. González et al. (Eds.): NICSO 2010, SCI 284, pp. 141–150, 2010.
springerlink.com

Cooperation yields a payoff R (S) if the opponent cooperates (defects) and defection yields T (P) if the opponent cooperates (defects). R is the Reward for cooperation, S is the Sucker's payoff, T is the Temptation to defect and P is the Punishment. Typically, $T > R > P > S$ and $2R > T + P$. PD game is a non zero sum game because one player's loss does not equal the opponent's gain. In order to reduce the amount of parameters, one can follow Nowak et al. [8] and use the following simplified payoff table$A$,

$$A = \begin{pmatrix} R & S \\ T & P \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ b & 0 \end{pmatrix} . \tag{1}$$

In this setup, there remains a free parameter $b(= T)$ which should be in the range $(1, 2)$. The tragedy behind this simple PD game is that the best strategy for a selfish individual, who chooses to defect, will result in mutual defection. This entails the worst collective effect for the society. In this game, the expectation of defection (D) is greater than the expectation of cooperation (C), independent of the opponent's strategy, even though mutual cooperation yields a higher total payoff for the society. The only state where no player can gain more by unilaterally changing its own strategy, the state called the Nash Equilibrium, occurs when all players defect. Hence, if the players use the simple imitation rule so that the players will adapt the strategy of a more successful player, the dominant strategy is defection. In order to further investigate the emergence of cooperation, a variant of the PD game is to consider a set of players located on a lattice and play the so-called spatial PD game (SPDG). In this case, cooperators can support each other in more than one dimension [9]. There are other approaches which will favor the survival of cooperation, as can be found in the recent work of Hebling et al [10] and Nowak [6].

The total income of player i in the spatial PD game can be described by a two-state Potts model Hamiltonian [2, 11]:

$$H_i = \sum_{j(i)} \underset{\sim}{S}_i^T \underset{\sim}{A} \underset{\sim}{S}_j \quad \text{with} \underset{\sim}{S}_i^T , \underset{\sim}{S}_j \in \left\{ \overrightarrow{C}, \overrightarrow{D} \right\} \text{and} \overrightarrow{C} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \overrightarrow{D} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \tag{2}$$

Here is the state vector for player j who is a neighbor of player i and the state vector can be either one of the two unit vectors . The summation runs over all the neighbors of the player i sitting at node i, while the neighborhood is defined by the topology of the given network. We will also give the players the ability to remember a fixed number of the most recent events and supply each player with a rule to decide what move they should take to respond to a history. We call this rule a strategy. A complete strategy covers all the possible situations but in a real game only a subset of the strategy will be used. Players will adapt their strategies, imitating other more successful players following a certain imitation rule. The usual imitation rule assumes that the player will copy all the strategies of his idol, who is a more successful opponent in his encounter. However, if only a subset of all the strategies of the idol has been used, it is unrealistic for the player to copy all the strategies, including those that have never been observed. A realistic modification on the imitation rule is to copy only those strategies that have been observed. The modification of the traditional imitation rule is necessitated by the fact that all players can only have finite memory. This simple observation, together with the existence of a generally large set of possible strategies, motivates us to consider a new imitation rule. We call it a "partial imitation rule", as it permits the player to imitate at most those strategies his idol has used. In real life, a player cannot even remember all the observed moves of his idol. We will formulate our representation of the strategy and the numerical methods used in setting up the spatial iterated PD game in Section 2. We then present a detailed discussion on the various imitation rules in Section 3. The results of our simulation are summarized and discussed in Section 4. We conclude with some discussion on the implication of partial imitation rule and discuss future works in the final section.

2 Methods

2.1 Memory Encoding

A two-player PD game yields one of the four possible outcomes because each of the two independent players has two possible moves, cooperate (C) or defect (D). To an agent i, the "outcome" of playing a PD game with his opponent, agent j, can be represented by an ordered pair of strategies $s_i s_j$. Here s_i can be either C for "cooperate" or D for "defect". Thus, there are four possible scenarios for any one game between them: $\{s_i s_j\}$ takes on one of these four outcomes $\{CC, CD, DC, DD\}$. In general, for n games, there will be a total of $4n$ possible scenarios. A particular pattern of these n games will be one of these $4n$ scenarios, and can be described by an ordered sequence of the form $S_{i1}S_{j1} \ldots S_{in}S_{jn}$. This particular ordered sequence of outcomes for these n games is called a history of games between these two players, which consists of n pairs of outcome $\{S_i S_j\}$, with the leftmost one being the first game played, while the rightmost one being the outcome of the last gamed played, or the most recent outcome. For example, an ordered sequence of strategy pairs *DDDDDDCC* represents that the two players cooperate right after the past three defection $\{DD\}$, $\{DD\}$, $\{DD\}$. Note the convention for the outcome is that the in the pair $\{s_i s_j\}$, s_i is the move made by agent i, who is the player we address, while s_j is the move made by agent j, the opponent of our player.

We say that a player has a memory of fixed-length m, when this player can remember only m-pairs of outcomes. Obviously, a "Memory" is a sub-sequence of a history. In a PD-game with a fixed memory-length m, the players can get access to the outcomes of the past m games and decide the response to the specific outcomes in the present game. For example, for an agent with two-game memory ($m = 2$), given a history represented by *DDDDDDCC*, the memory of the player consists of only the substring *DDCC*. Because a given memory can be represented by a unique sequence of strategies, a memory can be conveniently designate by a unique number. In this paper, cooperation is represented by 1 and defection 0. Thus, the memory *DDCC* can be represented by the binary number 0011 or the decimal number 3. The number of all the possible memory, given that the agent can memorize the outcomes of the last m games, is $4m$. Next, we must address the beginning of the game between our players.

Let's consider a non-trivial example when $m = 3$. In this case there are $64 = 4m = 4 \cdot 3$ possible histories of the strategies used by the two players. Following a method proposed by Bukhari and Haider[12], we reserve one bit for the first move of our player: $\{D, C\}$, and use two more bits for the second move of our player when confronted with the two possibilities of the first move of the opponent $\{D, C\}$. (Our player can choose C or D when the opponent's first move is D, and our player also can choose C or D when the opponent's first move is C. Thus we need two more bits for our player). To account for the four possible scenarios of the last two moves of the opponents: $\{DD, DC, CD, CC\}$, we need to reserve 4 more bits to record the third move of our player. Thus, for a PD game played by prisoners who can remember 3 games, a player will need $1+2+4 = 7$ bits to record his first three moves. After this initial stage, the strategy sequence for our player will need to respond to the game history with a finite memory. Since there are a total of $64 = 4m = 4 \cdot 3$ possible Memory, i.e., 64 possible outcomes of the last three games, our player will need 64 more bits. In conclusion, the length of the strategy sequence is $7+64 = 71$ and there are a total of , possible strategies. Thus the space of strategies for a $m = 3$ game is already very large. Let's now denote the ensemble of m-step memory as M_m, then the total number of bits required to encode the possible strategy sequence is $b(m) = 2m - 1 + 4m$ and the total number of possible strategy sequences is $|Mm| = 2b(m)$. Table 1 summarizes the enumeration of the encoding of the possible strategies for $m = 1$. The representation of the strategy sequence in $M1$ is denoted as $S_0|S_1S_2S_3S_4$. Here $b(1) = 5$ and there are a total of 32 possible strategies, since each S_i can have two possible choices (C or D) for $i = 0, ..., 4$. For $m = 2$, we have $b(2) = 19$ and $|M2| = 524288$, allowing for an exhaustive enumeration of all possible strategies [13]. For $m = 3$, we see that the $|M3|$ is $2^{71} = 2.4 \cdot 10^{21}$, which is already very large.

Table 1 Representation of Strategy Sequence in M_1

Memorized History	First Move	DD	DC	CD	CC
Players' Strategy	S_0	S_1	S_2	S_3	S_4

2.2 Monte Carlo Simulation

In this paper, agents will be placed on a square lattice of size LxL, with periodic boundary condition. Each agent only interacts with its four nearest neighbors. For one confrontation we randomly choose an agent i and a neighbor j of i and let them play F games with each other. We can compute the payoff $U(i)$ and $U(j)$ of agent i and j over these games in this confrontation. The payoff parameters used are $T = 5.0, R = 3.0, P = 1.0, S = 0.0$, which are widely used and allow meaningful comparison with the existing results. Agent i will then imitate agent j with probability

$$P(\text{i imitates j}) = \frac{1}{1+\exp\left(\frac{U_i - U_j}{K}\right)} \quad (3)$$

K is similar to the temperature and represents the thermal noise level. The larger corresponds to smaller noise. We use $K = 100$. The reason that we decide that in one confrontation between agent i and j, they have to play $F(>1)$ games is that memory effect will not be evident unless there is some repeated encounter between the two players to let them learn about the selected strategies used. However, a fixed number for F is rather artificial. Different pairs of players may play different number of games. Furthermore, we find that fixing F does affect the results in a complex manner. In order to test the strategies for different F, we introduce a probability parameter p for a player to stop playing games with his chosen opponent. We further define one generation of the PD game on the square lattice when all LxL confrontations are completed. With this stopping probability p, one effectively control the average number of games played between pair of players, thereby determining F. The choice of F and the rest of the procedure in one independent simulation can be described by the pseudo code in algorithm 3 for a given p.

Algorithm 3. Iterated SPDG algorithm.

```
$P := 0.05$
$F := 1$
**for** $i = 0$ to 100 **do**
    **while** $rand() > p$ % where rand() generates a random number in $[0,1)$ drawn from a uniform distribution **do**
        $F := F + 1$
    **end while**
**end for**
**for** $j = 0$ to $L * L$ **do**
    randomly pick one site A
    A plays with its neighbors, each confrontation lasts for F games
    randomly pick one site B from A's neighborhood
    B plays with its neighbors, each confrontation lasts for F games
    **if** $rand() < \left\{1.0 + exp\left(\frac{A.payoff - B.payoff}{K}\right)\right\}^{-1}$ **then**
        A imitates B using different imitation rules
    **end if**
**end for**
```

3 Imitation Rule

The standard imitation rule for the spatial PD game without memory is that the focal agent i will adopt the pure strategy of a chosen neighbor depending on payoff. The generalized imitation rule for PD game with memory is adopting the entire set of strategy sequences. We call such imitation rule the traditional imitation rule (tIR). In this way, tIR impose that condition that every agent has complete information about the entire set of the strategy sequence of all its neighbors. Such assumption of complete information is unrealistic since the focal agent only plays a few games with its neighbors while the space of strategies used by the neighbor is generally astronomically larger than F. A more realistic situation is that the focal agent i only has partial information about the strategies of his neighbors. In this paper, every agent only knows a subset of the strategy sequence used by a chosen neighbor. For a pair of players (i, j), playing approximately F games, the focal player i will only observed a set $(S_j(i, j))$ of strategy sequences actually used by agent j. This set $S_j(i, j)$ is much smaller than the entire set of strategies available to agent j. With this partial knowledge of the strategies of the neighbors, the new imitation rule for agent i is called the partial imitation rule. We will give an example to illustrate the difference between partial imitation rule and the traditional one. Let's consider an agent i with $C|DDDD$ strategy confronts another agent j with the Tit-for-Tat strategy $(S_0|S_1S_2S_3S_4 = C|DCDC)$ and agent i decides to imitate the agent j's strategy. In tIR, we assume that agent i somehow knows all the five bits of Tit-for-Tat though in the confrontation with agent j only four bits of Tit-for-Tat have been used. On the other hand, with partial imitation rule (pIR), when a $C|DDDD$ agent confronts a Tit-for-Tat agent, the $C|DDDD$ will know only four bits of Tit-for-Tat $(S_0|S_1S_2S_3S_4 = C|DCDC)$, i.e., $S_0 = C$, $S_1 = D$, $S_2 = C$, $S)3 = D$ (c.f. table 1). Thus, when agent i imitates agent j using pIR, agent i will become $(C|DDDC)$, which corresponds to a Grim Trigger instead of Tit-for-Tat $(C|DCDC)$. We call this new imitation rule the type 1 partial imitation rule, denoted by pIR1. In a more relaxed scenario, we can slightly loosen the restriction on the access of our focal agent i to the information of neighbors' strategy sequences. If we denote the subset of agent j's strategy sequence used during the confrontation between agent i and agent j as $S_j(i, j)$, then we can assume that agent i knows the larger subset of strategy sequences of agent j described by

$$G_j(i, j) = \bigcup_{k \in \Omega(j)} S_j(k, j) \tag{4}$$

where $\Omega(j)$ denotes the nearest neighbors of agent j. Note that this set of strategy sequences of agent j is substantially larger than $S_j(i, j)$, but still should generally be much smaller than the entire set of strategies of player j. In pIR1, we provide agent i information on agent j defined by the set $S_j(i, j)$. We now introduce a second type of partial imitation rule, denoted by pIR2, if we replace $S_j(i, j)$ by the much larger set $G_j(i, j)$.

We now illustrate pIR2 with an example using the notation of table 1. Consider an always-cooperating agent i $(C|CCCC)$ confronting a Grim Trigger $(C|DDDC)$ agent j, who has four neighbors. One of them of course is the always cooperating agent i. Let's assume that the remaining three neighbors of agent j are always-defecting $(D|DDDD)$. Let's call these three neighbors agent a, b, and c. In the confrontation between agent i (who is $C|CCCC$) and agent j (Grim Trigger), S_0 and S_4 of Grim Trigger are used. However, in the confrontation between agent j (Grim Trigger) and its three neighbors (agent a, b and c), who are $D|DDDD$, agent j will use S_0, S_1

and S_3 of Grim Trigger. With pIR1, agent i imitates agent j, but the result will be unchanged as they will use C for S_0 and S_4 of Grim Trigger based on the set $S_j(i,j)$. However, for pIR2, agent i imitates agent j and changes from $C|CCCC$ to the Grim Trigger agent, which results in a change of its S_0, S_1, S_3 and S_4 to the corresponding bits of Grim Trigger, giving the new strategy of agent i as $C|DCDC$. This is not a Grim Trigger. Finally, if we use tIR, the traditional imitation rule, we of course will replace agent i with Grim Trigger $(C|DDDC)$. We see from this example, the result of tIR, pIR1 and pIR2 are all different.

4 Results

We first test our algorithm of SPDG with the published results [13]. We initialize our strategy sequence with each element assigned cooperation or defection at equal probability and reproduce results similar to figure 3a in [13] in figure 1 using the traditional imitation rule. Here, Tit-For-Tat (TFT) and Grim- Trigger (GT) dominate at long time. These two strategies together with Pavlov and $C|CCDC$ are the only four surviving strategies in the long run. In figure 2(a) we use partial imitation rule 1 (pIR1) and in 2(b), we use pIR2. In both cases, only GT dominates and the concentration of TFT is reduced greatly to the level of Pavlov and $C|CCDC$. Results are independent of the lattice size, provided that it is sufficiently large so that every strategy in M_1 can be visited several times. We next discuss the importance of game

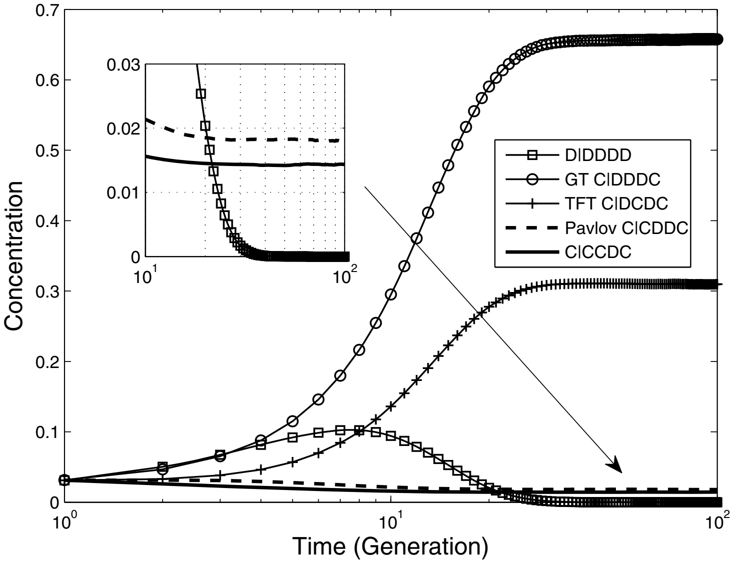
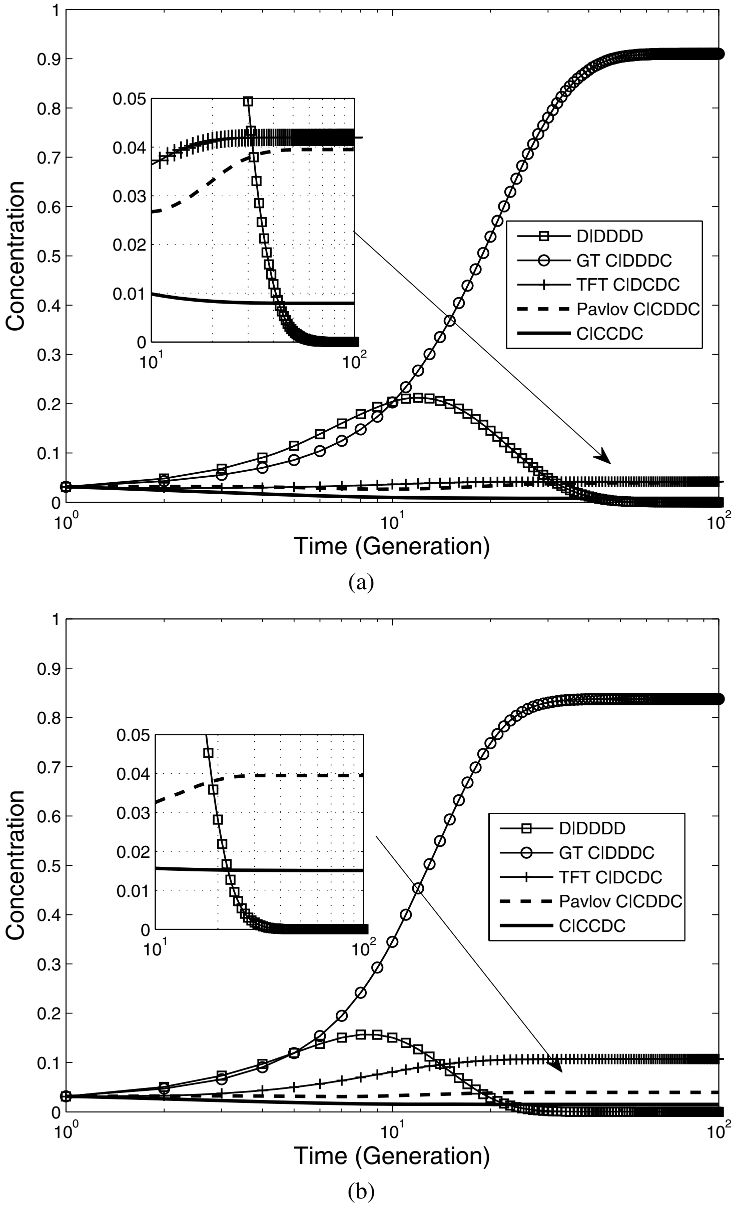


Fig. 1 Concentration of important strategies in SPDG on 100x100 square lattice with M_1. Result is averaged over 1000 independent simulations, with $K = 0.01$, using traditional Imitation Rule (tIR)



(a)

(b)

Fig. 2 Concentration of important strategies in SPDG on 100x100 square lattice with M_1. Result is averaged over 1000 independent simulations, with $K = 0.01$, using partial Imitation Rule 1 (pIR1) in figure 2(a) and partial Imitation Rule 2 (pIR2) in figure 2(b)

sampling in terms of the value of p, i.e. the probability to end one confrontation at time t. Our numerical experiments show that p affects the concentrations of all the strategies regardless of the imitation rule used. When $p = 1$, agents will always cooperate or defect without making use of the memory mechanism. When p is smaller than 1, agents can use their memory to access the benefits of different strategies. Recall that we have in general 32 strategies for M_1. For sufficiently small p, our numerical results indicate that the concentrations of these 32 strategies reach a constant value at long time. In this paper, we choose p to be 0.05, but in fact when p is smaller than 0.2, the results will not differ much.

5 Conclusion

Our introduction of memory effects on the players in spatial PD game indicates the importance of the imitation rule used in the learning process of the players. We started our spatial PD game with traditional imitation rule, which makes the unrealistic assumption that the players have a complete access of his opponent's strategies. When this assumption is relaxed and implemented with the partial imitation rules that assume the players only have the information on a selected subset of his opponent's strategies, the long time behavior of the concentration of various strategies are very different. Indeed, for the traditional imitation rule, where TFT and GT dominate at long time, only GT remains dominant when partial imitation rule is used. With the traditional imitation rule, TFT and GT dominate at the long run, while with either partial imitation rule 1 or 2, only GT dominate the population and the concentration of TFT is remarkably smaller than that with the traditional imitation rule. This work shows that with memory, more realistic imitation rules may have an impact on the concentration of the surviving strategies such as TFT and GT. In the scenario we set up in the paper, GT is favored by the partial imitation rules. This result has important implication of previous studies on PD game as partial knowledge of the opponents' strategies should be the norm rather than the exception in real life. In future work, we will investigate more on the generalization of our observation to longer memory cases.

Acknowledgements. K.Y. Szeto acknowledges the support of CERG grant 602506 and 602507.

References

[1] von Neumann, J., Morgenstern, O.: Theory of Games and Economic Behaviour. Princeton University Press, Princeton (1944)

[2] Szabo, G., Fath, G.: Evolutionary games on graphs. Physics Reports 446(4-6), 97–216 (2007)

[3] Smith, J.M., Price, G.M.: The logic of animal conflict. Nature 246, 15–18 (1973)

[4] Smith, J.M.: Evolution and the Theory of Games. Cambridge University Press, Cambridge (1982)

[5] Ohtsuki, H., Hauert, C., Lieberman, E., Nowak, M.A.: A simple rule for the evolution of cooperation on graphs and social networks. Nature 441, 502–505 (2006)

[6] Nowak, M.A.: Five Rules for the Evolution of Cooperation. Science 314(5805), 1560–1563 (December 8, 2006)

[7] Axelrod, R.: The Evolution of Cooperation. Basic Books, New York (1984)

[8] Nowak, M.A., May, R.M.: The spatial dilemmas of evolution. Int. J. of Bifurcation and Chaos 3(1), 35–78 (1993)

[9] Szabo, G., Vukov, J., Szolnoki, A.: Phase diagrams for an evolutionary prisoner's dilemma game on two-dimensional lattices. Phys. Rev. E 72(4), 047107 (2005)

[10] Helbing, D., Lozano, S.: Routes to cooperation and herding effects in the prisoner's dilemma game (May 2009)

[11] Ariosa, D., Fort, H.: Extended estimator approach for 2x2 games and its mapping to the Ising Hamiltonian. Phys. Rev. E 71, 016132 (2005)

[12] Bukhari, S., Adnan, H.A.S.: Using genetic algorithms to develop strategies for the prisoners dilemma. Asian Journal of Information Technology 8(5), 866–871 (2006)

[13] Baek, S.K., Kim, B.J.: Intelligent tit-for-tat in the iterated prisoner's dilemma game. Physical Review E (Statistical, Nonlinear, and Soft Matter Physics) 78(1), 011125 (2008)