Applied Mathematics and Computation 282 (2016) 117–127

Contents lists available at ScienceDirect

Applied Mathematics and Computation

journal homepage: www.elsevier.com/locate/amc

Iterated symmetric three-player prisoner's dilemma game

Essam El-Seidy[a], Karim.M. Soliman[b,*]

[a] Department of Mathematics, Faculty of Science, Ain Shams University, Cairo, Egypt
[b] University of Science and Technology, Zewail City for Science and Technology, Giza, Egypt

A R T I C L E I N F O

MSC:
91A06
91A10
91A15
91A20

Keywords:
Iterated games
Prisoner's dilemma
Payoff matrix
Symmetric games
Tit For Tat strategy
Evolutionary games

A B S T R A C T

Although, most game theory researches on the prisoner's dilemma have centered on two-player models, it is possible to create it to be consisted of three or even more players. In this paper, we are interested in the model of three-player iterated prisoner's dilemma game where, each player has two choices. The action of each strategy in this model depends on the previous action of the last round. Each strategy is presented by finite state of automata. We used a computer program to calculate the payoff values resulting from the actions of all possible strategies. We study the behavior of four different strategies related to Tit for Tat concept. The conditions of each strategy to be the best are determined. In Appendix section, we design an algorithm and implement it using the Java programing language to facilitate the calculations.

1. Introduction

The prisoner's dilemma (PD) is a traditional game model for the study of decision-making and self-interest [1,2]. It is only one of many illustrative examples of the logical reasoning and complex decisions involved in game theory. The mechanisms that drive the (PD) are the same as those that are faced by marketers, military strategists, poker players, and many other types of competitors [3–5]. This dilemma can multiply into hundreds of other more complex dilemmas. The dilemma has widely been addressed in different disciplines such as artificial intelligence, economics [6,7], biology [8], physics, networks [9], business [10], mathematics [11,12], philosophy, public health, ecology [13], traffic engineering [14], sociology and computer science [15].

In the prisoner's dilemma, two players are faced with a choice, they can either cooperate or defect. Each player is awarded points (called payoff) depending on the choice they made compared to the choice of the opponent. Each player's decision must be made without knowledge of the other player's next move. Prior agreement between the players concerning the game is not allowed. If both players cooperate they both receive a reward, *R*. If both players defect they both receive a punishment, *P*. If one player defects and the other cooperate, the defector receives a reward, *T* the temptation to defect, while the player who cooperated is punished with the sucker's payoff, *S* [16]. We can represent the payoff matrix as the following:

$$\begin{array}{cc} & \begin{array}{cc} C & D \end{array} \\ \begin{array}{c} C \\ D \end{array} & \begin{pmatrix} R & S \\ T & P \end{pmatrix} \end{array} \tag{1}$$

Where, $T > R > P > S$ should be satisfied [17].

* Corresponding author. Tel.: +20 1017597979.
E-mail addresses: esam_elsedy@hotmail.com (E. El-Seidy), kamohamed@zewailcity.edu.eg (Karim.M. Soliman).

http://dx.doi.org/10.1016/j.amc.2016.02.009
0096-3003/

In single game of prisoner's dilemma (known as one shot), defection is the dominant strategy (the Nash equilibrium) [18]. Thus both players will defect earning rewards of P points rather than the R points that mutual cooperation could have yielded. The iterated prisoner's dilemma (*IPD*) is an interesting variant of (*PD*) where, the dominant mutual defection strategy relies on the fact that it is a one shot game with no future while, the key of the (IPD) is that the two players may meet each other again, and develop their strategies based on the previous game interactions [19]. Therefore a player's move now may affect how his/her opponent behaves in the future and thus affect the player's future payoffs, and this removes the single dominant strategy of mutual defection because, the players will use more complex strategies which depend on the game history to maximize the payoffs that they receive. In fact, under the correct circumstances mutual cooperation can emerge [10,20].

Wang et al. have studied the evolution of public cooperation on two interdependent networks that are connected by means of a utility function, which determines to what extent payoffs in one network influence the success of players in the other network [9,21]. Also, they have shown that the percolation threshold of an interaction graph constitutes the optimal population density for the evolution of public cooperation, and they have demonstrated this by presenting outcomes of the public goods game on the square lattice with and without an extended imitation range, as well as on the triangular lattice [22–24]. More importantly, they have found that for cooperation to be optimally promoted, the interdependence should stem only from an intermediate fraction of links connecting the two networks, and that those links should affect the utility of players significantly [25]. Recently, they have studied the evolution of cooperation in the public goods game on interdependent networks. They have shown that, increasing the relevance of the average payoff of nearest neighbors on the expense of individual payoffs in the evaluation of utility increases the survivability of cooperators [26,27]. They have showed that the interdependence between networks self-organizes so as to yield optimal conditions for the evolution of cooperation [28].

Perc and Szolonki have worked on studying the enhancement of cooperation, and the impact of diverse activity patterns on the evolution of cooperation in evolutionary social dilemmas [29–34]. They have showed that spatially and temporally white additive Gaussian noise introduced in the payoff matrix of an evolutionary spatial prisoner's dilemma game can facilitate and maintain cooperation in a resonant manner depending on the level of random variations [35]. Moreover, they have showed that extortion is evolutionary stable in structured populations if the strategy updating is governed by a myopic best response rule [36,37]. While, Xia et al. have focused on the weak prisoner's dilemma on random and scale-free (SF) networks, and have shown that degree-uncorrelated activity patterns on scale-free networks significantly impair the evolution of cooperation, and they have studied how the heterogeneous coupling strength affects the evolution of cooperation in the prisoner's dilemma game with two types of coupling schemes (symmetric and asymmetric ones) [38]. Their results convincingly demonstrated that the emergence or persistence of cooperation within many real-world systems can be accounted for by the interdependency between meta-populations or sub-systems [39]. Moreover, they have put forward an improved traveler's dilemma game model on two coupled lattices to investigate the effect of coupling effect on the evolution of cooperation [40]. Their results are surprisingly conducive to understanding the cooperation behavior of traveler's dilemma game within many real world systems, especially for coupled and interdependent networked systems [41].

Game theory has been extended into evolutionary biology, which has generated great insight into the evolution of strategies under both biological and cultural evolution. The replicator equation, which consists of sets of differential equations describing how the strategies of a population evolve over time under selective pressures, has also been used to study learning in various scenarios [42–44]. There are various approaches to construct dynamics in repeated games [45–47]. Kleimenov and Schneider have proposed approach of constructing dynamics in the repeated three-person game to give a tool for solving various optimization problems, for example, the problem of minimizing time of using abnormal behavior types. In their approach, two players act in the class of mixed strategies and the third player acts in the class of pure strategies [48,49]. Matsushima and Ikegami have discussed the similarity between a noisy $2p-IPD$ and a noiseless $3p-IPD$ game where the role of noise in the two-person game is replaced by the third player in the three-person game. It is known that, due to the noise, Tit for Tat loses its robustness and is taken over by more complex strategies in a noiseless *IPD* game, but in the $3p-IPD$ game, even without noise, tit for tat loses its robustness and is also taken over by more complex strategies. They found that similar strategies take over tit for tat in both situations. It is also found that game strategies in an automaton form can be understood as a combination of defensive and offensive substructures. A recognition of these substructures enabled them to study the mechanism of robustness in the strategies of the $3p-IPD$ game [50].

The existence and implications of alternative stable states in ecological systems have been investigated extensively within deterministic models. Sun et al. have studied the role of noise on the pattern formation of a spatial predator-prey model with Allee effect, and have showed that the spatially extended system exhibits rich dynamic behavior. More specifically, the stationary pattern can be induced to be a stable target wave when the noise intensity is small. As the noise intensity increases, patchy invasion emerges. Their results indicate that the dynamic behavior of predator–prey models may be partly due to stochastic factors instead of deterministic factors, which may also help to understand the effects arising from the undeniable susceptibility to random fluctuations of real ecosystems [51]. Also, they have presented a spatial version of the predator–prey model with *HollingIII* functional response, which includes some important factors such as external periodic forces, random fluctuations, and diffusion processes. They found that, noise can lead the spiral waves to be chaotic patterns [52]. Additionally, they have presented a numerical evidence of complicated phenomenon controlled by noise in a spatial epidemic model, where the number of the spot decreases as the noise intensity being increased. Their results showed that noise plays an important role in the pattern formation of the epidemic model, which may provide guidance to prevent and control the spread of disease [53]. Finally, because of vegetation plays an important role in the ecological environment, Sun et al. have presented a numerical analysis of the effect of noise on the spatial pattern of a vegetation model, and gave numerical evidences of a transition from spotted pattern to stripe growth induced by noise. They have shown that noise exerts a tremendous influence in the pattern formation of the vegetation model, which may provide guidance to protect the vegetation from desertification [54]. Also, Li and Jin have presented a numerical analysis of the effect of noise on the pattern formation of the predator-prey model. They have found that the number of the spotted pattern increases as the noise intensity increases. When the noise intensity and temporal correlation are in appropriate levels, the model exhibits phase transition from spotted to stripe pattern. Moreover, they have showed the number of the spotted and stripe pattern, with respect to both noise intensity and temporal correlation. Their studies raised important questions on the role of noise in the pattern formation of the populations, which may well explain some data obtained in the ecosystems [55].

2. Three-player prisoner's dilemma game (3p-PD)

2.1. One shot game

Most game theory research on the prisoner's dilemma has focused on two player games, but it is possible to create it involving three or even more players where, the strategies from the two player game do not necessarily extend to a three player game in a natural way. We consider a simple game with three players such that, each player has two pure strategies C and D, and each round in the game leads to one of the eight possible outcomes CCC, CCD, CDC, CDD, DCC, DCD, DDC or DDD, where the first position represents the player under consideration, the second and the third positions represent the opponents. For example, DCD represents the payoff to a defecting player if one of his two opponents cooperates and the other opponent defects. Since we assume a symmetric game matrix, XCD could be written as XDC, where X may be C or D. These outcomes are specified by the player's payoff R, K, S, T, L or P which can be numbered by $i = 1, 2, 3, 4, 5, 6$ respectively. We impose three rules about the payoffs for the (3p-PD):

1. Defection should be the dominant choice for each player. In other words, it should be always better for a player to defect, regardless what the opponents do. This rule gives three constraints:
   (a) $DCC > CCC$ ($T > R$) (both opponents cooperate).
   (b) $DCD > CCD$ ($L > K$) (one opponent cooperates, the other defects).
   (c) $DDD > CDD$ ($P > S$) (both opponents defect).
2. A player should always be better off if more of his opponents choose to cooperate. This rule gives two constraints:
   (a) $CCC > CCD > CDD$ ($R > K > S$).
   (b) $DCC > DCD > DDD$ ($T > L > P$).
3. If one player's choice is fixed, the other two players should be left in (2p-PD). This rule gives the two constraints:
   (a) $CCD > DDD$ ($K > P$).
   (b) $CCC > DCD$ ($R > L$).

Finally, suppose the payoff matrix of the (3p-PD) is

$$\begin{array}{cc} & \begin{matrix} CC & CD & DD \end{matrix} \\ \begin{matrix} C \\ D \end{matrix} & \begin{pmatrix} R & K & S \\ T & L & P \end{pmatrix} \end{array} \tag{2}$$

where, $T > R > L > K > P > S$.

Assume a rational player under consideration who wants to maximize his/her reward and thinks that his/her opponents will cooperate, then he/she will defect to receive a reward, T points as opposed to the cooperation which would have earned him/her only, R points. If the rational player thinks that his/her opponents will defect, then he/she will also defect and receive, P points rather than cooperate and receive the sucker's payoff, S points. Moreover if the rational player thinks that one of his/her opponents will defect and the other will cooperate, then he/she will also defect and receive, L points rather than cooperate and receive, K points. Therefore the rational decision is to always defect. But assuming the opponents are also rational, they will come to the same conclusion and they will defect. Thus all players will always defect, earning rewards of, P points rather than the, R points that cooperation could have yielded. Therefore defection is the dominant strategy for (3p-PD) (the Nash equilibrium), this holds true as long as the payoffs follow the relationship, $T > R > L > K > P > S$.

2.2. Infinitely iterated game (3p-IPD)

Consider now the iterated game repeating the simple game infinitely i.e. with probability 1 to repeat the game. In the above discussion of the (3p-PD), the dominant defection strategy relies on the fact that it is a one shot game with no future. The key to the (3p-IPD) is that the three players may meet each other again, this allows the players to develop strategies based on the previous game interactions. Therefore a player's move now may affect how his/her opponents behave in the future and thus affect the player's future payoffs. This removes the single dominant strategy of defection because, the players use more complex strategies dependent on game history to maximize the payoffs they will receive. We assume that the

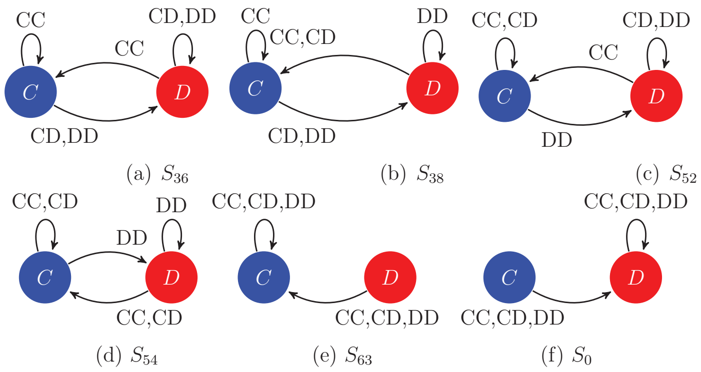

Fig. 1. Some examples of automata: (a) automaton represents (*TFT*1) S_{36} where, it still play *C* only if its two opponents play *C* together while, it moves from the state *C* to the state *D* if only one of its two opponents play *D*. In addition, it still play *D* unless its two opponents plays *C* together again. (b) Automaton represents (TFT2) S_{38} where, it still play *C* if its two opponents play *C* together while, it moves from the state *C* to the state *D* if only one of its two opponents play *D*. In addition, it still play *D* only if its two opponents plays *D* together again. (c) Automaton represents (TFT3) S_{52} where, it still play *C* if only one of its two opponents play *C* while, it moves from the state *C* to the state *D* if only one of its two opponents play *D*. In addition, it still play *D* unless its two opponents plays *C* together again. (d) Automaton represents (TFT4) S_{54} where, it still play *C* if one of its two opponents play *C* while, it moves from the state *C* to the state *D* if its two opponents play *D*. In addition, it still play *D* unless one of its two opponents plays *C* again. (e) Automaton represents (*ALLC*) S_{63} where, it still play *C* forever. (f) Automaton represents (*ALLD*) S_0 where, it still play *D* for ever.

three players take their decision according to the last choice of the opponents, only the last choice, by this assumption we call our game, (3p-IPD) with memory one. The length of the (3p-IPD) (i.e. number of repetitions of the dilemma played) must not be known to the players, unless all players would defect.

In the (3p-IPD), every player has two choices either to defect or to cooperate after each outcome of the six outcomes *T*, *R*, *L*, *K*, *P*, *S*, so the total number of strategies can be composed as $2^6 = 64$ different strategies. The 64 possible strategies can be labeled by $(u_1, u_2, u_3, u_4, u_5, u_6)$ of zeros and ones. Here, u_i is 1 if the player plays *C* and 0 if he/she plays *D* after outcome $i (i = 1, 2, 3, 4, 5, 6)$. For convenience, we label these rules by S_j, where j ranges from 0 to 63 and j is the integer given by (in binary notation) $u_1u_2u_3u_4u_5u_6$ [11,17].

We can describe our strategies by finite state automata, more precisely, two state automata only. Each of the three players is now an automaton which can be in one of two states through any given round of the (3p-IPD). These states correspond to the two possible moves *C* and *D*. The state of the player in the following round depends on the present state and on the opponent's move. Hence each such automaton is specified by a graph with two nodes *C* and *D* and three oriented edges issuing from each node, one labeled *C* and the other *D*, which specify the transition from the current state to the state in the next round [17]. For examples, the transition rule (1, 0, 0, 1, 1, 0) represents the strategy S_{38} represented in Fig. 1.

How one rule fares against another depends, of course, on the initial condition of this rule [10]. Let us consider, for instance, an automaton with rule S_{36} (a retaliator never relents after defection from any one of his/her opponents unless they both cooperate again) against the two automata S_{38} (a retaliator but is more forgiving than S_{36}) and S_{52} (slow to anger than his/her opponents and slow to forgive than S_{36}).

(a) If all three automata start with *C*, they will keep playing *C* forever. The sequence looks as follows:

$$\begin{array}{llllllllll} S_{36}: & C & C & C & C & C & C & C & C & C\ldots \\ S_{38}: & C & C & C & C & C & C & C & C & C\ldots \\ S_{52}: & C & C & C & C & C & C & C & C & C\ldots \end{array}$$

(b) If all three automata start with *D*, the sequence looks as follows:

$$\begin{array}{llllllllll} S_{36}: & D & D & D & D & D & D & D & D & D\ldots \\ S_{38}: & D & D & D & D & D & D & D & D & D\ldots \\ S_{52}: & D & D & D & D & D & D & D & D & D\ldots \end{array}$$

(c) If S_{36} and S_{38} start with *C* while S_{52} starts with *D*, the sequence looks as follows:

$$\begin{array}{llllllllll} S_{36}: & C & D & D & D & D & D & D & D & D\ldots \\ S_{38}: & C & D & C & D & D & D & D & D & D\ldots \\ S_{52}: & D & C & D & D & D & D & D & D & D\ldots \end{array}$$

(d) If S_{36} and S_{52} start with *C* while S_{38} starts with *D*, the sequence looks as follows:

$$\begin{array}{llllllllll} S_{36}: & C & D & C & D & C & D & C & D & C\ldots \\ S_{38}: & D & C & D & C & D & C & D & C & \ldots \\ S_{52}: & C & C & C & C & C & C & C & C & C\ldots \end{array}$$

(e) If S_{38} and S_{52} start with D while S_{36} starts with C, the sequence looks as follows:

$$\begin{array}{llllllllll} S_{36}: & C & D & \mathrm{D} & \mathrm{D} & \mathrm{D} & \mathrm{D} & \mathrm{D} & \mathrm{D} & \mathrm{D}\ldots \\ S_{38}: & \mathrm{D} & C & \mathrm{D} & \mathrm{D} & \mathrm{D} & \mathrm{D} & \mathrm{D} & \mathrm{D} & \mathrm{D}\ldots \\ S_{52}: & \mathrm{D} & \mathrm{D} & \mathrm{D} & \mathrm{D} & \mathrm{D} & \mathrm{D} & \mathrm{D} & \mathrm{D} & D\ldots \end{array}$$

(f) If S_{38} and S_{52} start with C while S_{36} starts with D, the sequence looks as follows:

$$\begin{array}{llllllllll} S_{36}: & D & C & D & C & D & C & D & C & D\ldots \\ S_{38}: & C & D & C & D & C & D & C & D & C\ldots \\ S_{52}: & C & C & C & C & C & C & C & C & C\ldots \end{array}$$

(g) If S_{36} and S_{52} start with D while S_{38} starts with C, the sequence looks as follows:

$$\begin{array}{llllllllll} S_{36}: & D & D & D & D & D & D & D & D & D\ldots \\ S_{38}: & C & D & D & D & D & D & D & D & D\ldots \\ S_{52}: & D & D & D & D & D & D & D & D & D\ldots \end{array}$$

(h) If S_{36} and S_{38} start with D while S_{52} starts with C, the sequence looks as follows:

$$\begin{array}{llllllllll} S_{36}: & D & D & D & D & D & D & D & D & D\ldots \\ S_{38}: & D & C & D & D & D & D & D & D & D\ldots \\ S_{52}: & C & D & D & D & D & D & D & D & D\ldots \end{array}$$

The payoff in the infinitely repeated game is simply the average payoff per round. In our example, for the player under consideration who is using the strategy S_{36}, the payoff is R in case (a), $\frac{K+T}{2}$ in cases (d) and (f), and P in cases (b), (c), (e), (g) and (h). This directly means that for the player using the strategy S_{38}, the payoff is R in case (a), $\frac{K+T}{2}$ in cases (d) and (f), and P in cases (b), (c), (e), (g) and (h). Finally for the player using the strategy S_{52}, the payoff is R in case (a), $\frac{K+K}{2}$ in cases (d) and (f), and P in cases (b), (c), (e), (g) and (h). Noting that the payoffs are independent of the moves of the players in the first round.

We can use a more direct approach [17] where the eight possible initial conditions lead (in unperturbed runs) to three possible regimes A, B and E, where A denotes the run where the three players use C, while B is the run where the S_{52}-player always plays C and the other two opponents one of them plays C and the other plays D, finally E denotes the run where the three players use D. Suppose we are in regime A, rare perturbation causes one of the three players to play D that follows either scenario (f), (d) or (c), and hence leads after few steps with probability $\frac{2}{3}$ to regime B and with probability $\frac{1}{3}$ to regime E. Suppose now that a perturbation occurs in regime B, it leads with probability $\frac{1}{3}$ to regime A and with probability $\frac{2}{3}$ to regime E. Suppose now that a perturbation occurs in regime E, it leads always to regime E. The corresponding transition matrix is

$$\begin{pmatrix} 0 & \frac{2}{3} & \frac{1}{3} \\ \frac{1}{3} & 0 & \frac{2}{3} \\ 0 & 0 & 1 \end{pmatrix} \tag{3}$$

The corresponding stationary distribution vector (the eigen vevtor corresponding to the eigen value one) is (0, 0, 1). Thus, an iterated game between S_{36}, S_{38} and S_{52} will be always in the regime E, i.e the S_{36}-player receives an average payoff P per round. By repeating this technique for the combinations between the strategies of all three players, we get $64 \times 64 \times 64 = 262144$ payoff values for strategy S_i of the player under consideration against the opponents strategies S_j of playerII and S_k of playerIII where, $i, j, k = 0, 1, 2, \ldots, 63$.

It is clear that, the previous manner for calculations takes long time. Therefor it is more practical to design an algorithm and implement it using one of the programing languages to facilitate these calculations, and we present the designed algorithm in the Appendix section, where the Algorithm takes the three strategies corresponding to the three players as an input and gives us the regimes and their transition matrix as an output. For the player who use the strategy S_{52} against the two players using $S_0(ALLD)$ and $S_{63}(ALLC)$, the output will be the only two regimes K and L such that their transition matrix is:

$$\begin{pmatrix} \frac{1}{3} & \frac{2}{3} \\ \frac{2}{3} & \frac{1}{3} \end{pmatrix} \tag{4}$$

With a stationary distribution vector $(\frac{1}{2}, \frac{1}{2})$. It means that the player using S_{52} receives a payoff $\frac{K+L}{2}$, the player using S_0 receives a payoff $\frac{T+L}{2}$ and the player using S_{63} receives a payoff $\frac{K+S}{2}$. For example, we present only the payoff values for three players, where they play using either S_{10}, S_{20} or S_{30} in Table 1.

3. General Tit For Tat strategy

Although there is no one best strategy for all circumstances, one that works extremely well over a wide variety of environments is a simple Tit For Tat strategy. In this strategy, one begins by cooperating and then mimics the other player's

Table 1
The payoff values for the three players when they use either S_{10}, S_{20} or S_{30}.

1st player strategy	2nd player Strategy	3rd player strategy	1st player payoff	2nd player payoff	3rd player payoff
10	10	10	P	P	P
10	10	20	$\frac{2P+3K+3L}{8}$	$\frac{2P+3K+3L}{8}$	$\frac{2P+3T+3S}{8}$
10	10	30	P	P	P
20	20	20	P	P	P
20	20	10	$\frac{9L+6K+5P}{20}$	$\frac{9L+6K+5P}{20}$	$\frac{9S+6T+5P}{20}$
20	20	30	$\frac{P+L}{2}$	$\frac{P+L}{2}$	$\frac{P+S}{2}$
30	30	30	P	P	P
30	30	10	P	P	P
30	30	20	P	P	P
10	20	30	$\frac{P+3T}{4}$	$\frac{P+3K}{4}$	$\frac{P+3K}{4}$

Table 2
The $6-tuble$ $(v_1, v_2, v_3, v_4, v_5, v_6)$ is a stochastic vector for the first player asserting that he/she will get a payoff value $= \frac{v_1R+v_2K+v_3S+v_4T+v_5L+v_6P}{v_1+v_2+v_3+v_4+v_5+v_6}$, such that each row denotes the first player strategy, each column denotes the second player strategy, and the third player is using the strategy S_{52}. For example, the $6-tuble$ (0, 1, 1, 0, 0, 0) in the sixth row and seventh column indicates that, the player who is using the strategy S_{63} against S_0 and S_{52} will get a payoff $= \frac{K+S}{2}$.

Third player using S_52	52	38	54	36	63	0
***TFT*-3 (52)**	(1,0,0,0,0,1)	(1,0,0,0,0,1)	(1,0,0,0,0,1)	(1,0,0,0,0,1)	(1,0,0,0,0,0)	(0,0,0,0,0,1)
***TFT*-2 (38)**	(1,0,0,0,0,1)	(1,2,1,1,2,1)	(1,0,0,0,0,0)	(0,0,0,0,0,1)	(1,0,0,0,0,0)	(0,0,0,0,0,1)
***TFT*-4 (54)**	(1,0,0,0,0,1)	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(1,0,0,0,0,1)	(1,0,0,0,0,0)	(0,0,0,0,0,1)
***TFT*-1 (36)**	(1,0,0,0,0,1)	(0,0,0,0,0,1)	(1,0,0,0,0,1)	(0,0,0,0,0,1)	(1,0,0,0,0,0)	(0,0,0,0,0,1)
***ALLC* (63)**	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(0,1,1,0,0,0)
***ALLD* (0)**	(0,0,0,0,0,1)	(0,0,0,0,0,1)	(0,0,0,0,0,1)	(0,0,0,0,0,1)	(0,0,0,1,1,0)	(0,0,0,0,0,1)

moves. Tit For Tat is “nice” in that it is willing to cooperate and it does not bear a grudge. It also cannot be exploited because any “defection” from cooperation will be returned. If a player uses a Tit for Tat strategy, he/she will cooperate on the first round. If he/she discovers that the second player has defected, he/she will defect on the next round. If, after he/she realizes that the second player cooperated again, he/she becomes ready to cooperate, a Tit-For-Tat strategy is ready to begin cooperating [3,4]. The power of Tit For Tat in encouraging cooperation in unusual places has been explored by Axelrod in The Evolution of Cooperation [1].

In this section, we are searching for the meaning of Tit For Tat strategy in the (*I3PD*). We attempt to find an answer for the following questions: Should the player defect when either the opponents defected on the previous round or only when both opponents are defected? Is either of the strategies nearly as effective in the (I3PD) game as Tit For Tat is in the (I2PD) game? In order to do that, we will present four different strategies in which we can discuss the Tit For Tat concept in details. These four strategies are called *TFT*1 (S_{36}), *TFT*2 (S_{38}), *TFT*3 (S_{52}) and *TFT*4 (S_{54}) presented by the automatons in Fig. 1, such that:

- The player who uses the *TFT*1 strategy plays the iterated prisoner’s dilemma game as follows:
  If his/her two opponents have chosen the same choice, on the previous move, then he/she makes the choice which they did, else he/she always plays *D*.
- The player who uses the *TFT*2 strategy plays the iterated prisoner’s dilemma game as follows:
  If his/her two opponents have chosen the same choice together, on the previous move, then he/she makes the choice which they did, else he/she shifts his/her choice.
- The player who uses the *TFT*3 strategy plays the iterated prisoner’s dilemma game as follows:
  If his/her two opponents have chosen the same choice together, on the previous move, then he/she makes the choice which they did, else he/she stays on his/her choice.
- The player who uses the *TFT*4 strategy plays the iterated prisoner’s dilemma game as follows:
  If his/her two opponents have chosen the same choice, on the previous move, then he/she makes the choice which they did, else he/she always plays *C*.

4. Competition between Tit For Tat, *ALLC* and *ALLD*

Due to the extensive payoffs that we get it in the (I3PD) game including 64 strategies, we focus on the six strategies *TFT*1, *TFT*2, *TFT*3, *TFT*4, *ALLC* and *ALLD* and we work on calculating the payoff values of their competitions and we exhibit it in table form, see from Tables 2–7, such that the third player uses, in each table, a fixed strategy, where each row denotes the first player strategy, each column denotes the second player strategy, and every $6-tuble$ $(v_1, v_2, v_3, v_4, v_5, v_6)$ is a stochastic vector for the first player asserting that he/she will get a payoff value $= \frac{v_1R+v_2K+v_3S+v_4T+v_5L+v_6P}{v_1+v_2+v_3+v_4+v_5+v_6}$. Although, a complete analysis of the dynamical system is a hopeless task, we noticed that, there is no dominant strategy among them, and no pure strategy is evolutionary stable. But patchy invasion emerges, as every pure strategy can be invaded whenever

Table 3
The $6-tuble$ $(v_1, v_2, v_3, v_4, v_5, v_6)$ is a stochastic vector for the first player asserting that he/she will get a payoff value $= \frac{v_1R+v_2K+v_3S+v_4T+v_5L+v_6P}{v_1+v_2+v_3+v_4+v_5+v_6}$, such that each row denotes the first player strategy, each column denotes the second player strategy, and the third player is using the strategy S_{38}.

Third player using $S_3$8	52	38	54	36	63	0
***TFT*-3 (52)**	(1,0,0,0,0,1)	(1,2,1,1,2,1)	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(0,0,0,0,0,1)
***TFT*-2 (38)**	(1,2,1,1,2,1)	(1,2,1,1,2,1)	(5,9,0,6,3,1)	(1,3,6,0,9,5)	(2,1,0,1,0,0)	(0,0,1,0,1,2)
***TFT*-4 (54)**	(1,0,0,0,0,0)	(5,12,3,3,0,1)	(1,0,0,0,0,0)	(2,6,3,0,3,2)	(1,0,0,0,0,0)	(0,0,3,0,3,2)
***TFT*-1 (36)**	(0,0,0,0,0,1)	(1,0,3,3,12,5)	(2,3,0,3,6,2)	(0,0,0,0,0,1)	(2,3,0,3,0,0)	(0,0,0,0,0,1)
***ALLC* (63)**	(1,0,0,0,0,0)	(1,1,0,0,0,0)	(1,0,0,0,0,0)	(1,3,0,0,0,0)	(1,0,0,0,0,0)	(0,1,1,0,0,0)
***ALLD* (0)**	(0,0,0,0,0,1)	(0,0,0,0,1,1)	(0,0,0,0,3,1)	(0,0,0,0,0,1)	(0,0,0,1,1,0)	(0,0,0,0,0,1)

Table 4
The $6-tuble$ $(v_1, v_2, v_3, v_4, v_5, v_6)$ is a stochastic vector for the first player asserting that he/she will get a payoff value $= \frac{v_1R+v_2K+v_3S+v_4T+v_5L+v_6P}{v_1+v_2+v_3+v_4+v_5+v_6}$, such that each row denotes the first player strategy, each column denotes the second player strategy, and the third player is using the strategy S_{54}.

Third player using $S_5$4	52	38	54	36	63	0
***TFT*-3 (52)**	(1,0,0,0,0,1)	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(1,0,0,0,0,1)	(1,0,0,0,0,0)	(0,0,0,0,0,1)
***TFT*-2 (38)**	(1,0,0,0,0,0)	(5,9,0,3,6,1)	(1,0,0,0,0,0)	(2,3,3,3,3,2)	(1,0,0,0,0,0)	(0,0,3,0,3,2)
***TFT*-4 (54)**	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(0,14,6,0,6,4)
***TFT*-1 (36)**	(1,0,0,0,0,1)	(2,3,0,3,6,2)	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(0,0,0,0,0,1)
***ALLC* (63)**	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(0,1,0,0,0,0)
***ALLD* (0)**	(0,0,0,0,0,1)	(0,0,0,0,3,1)	(0,0,0,14,12,4)	(0,0,0,0,0,1)	(0,1,0,0,0,0)	(0,0,0,0,0,1)

Table 5
The $6-tuble$ $(v_1, v_2, v_3, v_4, v_5, v_6)$ is a stochastic vector for the first player asserting that he/she will get a payoff value $= \frac{v_1R+v_2K+v_3S+v_4T+v_5L+v_6P}{v_1+v_2+v_3+v_4+v_5+v_6}$, such that each row denotes the first player strategy, each column denotes the second player strategy, and the third player is using the strategy S_{36}.

Third player using $S_3$6	52	38	54	36	63	0
***TFT*-3 (52)**	(1,0,0,0,0,1)	(0,0,0,0,0,1)	(1,0,0,0,0,1)	(0,0,0,0,0,1)	(1,0,0,0,0,0)	(0,0,0,0,0,1)
***TFT*-2 (38)**	(0,0,0,0,0,1)	(1,3,6,0,9,5)	(2,3,3,3,3,2)	(0,0,0,0,0,1)	(2,3,0,3,0,0)	(00,0,0,0,1)
***TFT*-4 (54)**	(1,0,0,0,0,1)	(2,6,3,0,3,2)	(1,0,0,0,0,0)	(0,0,0,0,0,1)	(1,0,0,0,0,0)	(0,0,0,0,0,1)
***TFT*-1 (36)**	(0,0,0,0,0,1)	(0,0,0,0,0,1)	(0,0,0,0,0,1)	(0,0,0,0,0,1)	(0,0,0,0,1,0)	(0,0,0,0,0,1)
***ALLC* (63)**	(1,0,0,0,0,0)	(1,3,0,0,0,0)	(1,0,0,0,0,0)	(0,0,1,0,0,0)	(1,0,0,0,0,0)	(0,0,1,0,0,0)
***ALLD* (0)**	(0,0,0,0,0,1)	(0,0,0,0,0,1)	(0,0,0,0,0,1)	(0,0,0,0,0,1)	(0,0,0,0,1,0)	(0,0,0,0,0,1)

Table 6
The $6-tuble$ $(v_1, v_2, v_3, v_4, v_5, v_6)$ is a stochastic vector for the first player asserting that he/she will get a payoff value $= \frac{v_1R+v_2K+v_3S+v_4T+v_5L+v_6P}{v_1+v_2+v_3+v_4+v_5+v_6}$, such that each row denotes the first player strategy, each column denotes the second player strategy, and the third player is using the strategy S_{63}.

Third player using S_0	52	38	54	36	63	0
***TFT*-3 (52)**	(0,0,0,0,0,1)	(0,0,0,0,0,1)	(0,0,0,0,0,1)	(0,0,0,0,0,1)	(0,1,0,0,1,0)	(0,0,0,0,0,1)
***TFT*-2 (38)**	(0,0,0,0,0,1)	(0,0,1,0,1,2)	(0,0,3,0,3,2)	(0,0,0,0,0,1)	(0,1,0,0,1,0)	(0,0,0,0,0,1)
***TFT*-4 (54)**	(0,0,0,0,0,1)	(0,0,3,0,3,2)	(0,14,6,0,6,4)	(0,0,0,0,0,1)	(0,1,0,0,0,0)	(0,0,0,0,0,1)
***TFT*-1 (36)**	(0,0,0,0,0,1)	(0,0,0,0,0,1)	(0,0,0,0,0,1)	(0,0,0,0,0,1)	(0,0,0,0,1,0)	(0,0,0,0,0,1)
***ALLC* (63)**	(0,1,1,0,0,0)	(0,1,1,0,0,0)	(0,1,0,0,0,0)	(0,0,1,0,0,0)	(0,1,0,0,0,0)	(0,0,1,0,0,0)
***ALLD* (0)**	(0,0,0,0,0,1)	(0,0,0,0,0,1)	(0,0,0,0,0,1)	(0,0,0,0,0,1)	(0,0,0,0,1,0)	(0,0,0,0,0,1)

Table 7
The $6-tuble$ $(v_1, v_2, v_3, v_4, v_5, v_6)$ is a stochastic vector for the first player asserting that he/she will get a payoff value $= \frac{v_1R+v_2K+v_3S+v_4T+v_5L+v_6P}{v_1+v_2+v_3+v_4+v_5+v_6}$, such that each row denotes the first player strategy, each column denotes the second player strategy, and the third player is using the strategy S_0.

Third player using $S_6$3	52	38	54	36	63	0
***TFT*-3 (52)**	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(0,1,0,0,1,0)
***TFT*-2 (38)**	(1,0,0,0,0,0)	(2,1,0,1,0,0)	(1,0,0,0,0,0)	(2,3,0,3,0,0)	(1,0,0,0,0,0)	(0,1,0,0,1,0)
***TFT*-4 (54)**	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(0,1,0,0,0,0)
***TFT*-1 (36)**	(1,0,0,0,0,0)	(2,3,0,3,0,0)	(1,0,0,0,0,0)	(0,0,0,0,1,0)	(1,0,0,0,0,0)	(0,0,0,0,1,0)
***ALLC* (63)**	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(1,0,0,0,0,0)	(0,1,0,0,0,0)
***ALLD* (0)**	(0,0,0,1,1,0)	(0,0,0,1,1,0)	(0,0,0,1,0,0)	(0,0,0,0,1,0)	(0,0,0,1,0,0)	(0,0,0,0,1,0)

we fix the third player's strategy, and study the invasion among the other two players. In Table 8, we notice that, for example, the strategy S_{63} *ALLC* is not invaded, but only in case (e) by S_{52} and S_{54}. Also, in case (f) when the third player uses S_0, we can notice that the five strategies S_{38}, S_{54}, S_{36}, S_{63}, and S_0 are not invaded by any other strategy.

Table 8
A list of the strategies invading S_i where, we fix the strategy of the third player such that: we consider the third player uses S_{52} in column (a), S_{38} in column (b), S_{54} in column (c), S_{36} in column (d), S_{63} in column (e), and S_0 in column (f).

	(a)	(b)	(c)	(d)	(e)	(f)
$S_{52} <<$	54,63	54,63	54,63	54	–	38,36,0
$S_{38} <<$	54,63	54	54,63	54	52,0	–
$S_{54} <<$	63	–	63	–	52,63	–
$S_{36} <<$	52,38,54,63	52,38,54,63	52,54,63	52,38,54,0	0	–
$S_{63} <<$	–	–	–	–	52,54	–
$S_0 <<$	52,38,54,36,63	–	52,38,54,36,63	52,38,54,36	–	–

5. Conclusion

We concluded that: The strategy *TFT*4 is more relenting than the strategy *TFT*3 as the player who uses *TFT*4 will be relent (transfer from the state *D* to the state *C*) as soon as only one of his/her opponents plays *C* in the previous round, while *TFT*3 never moves from the state *D* to the state *C* unless his/her two opponents play *C* together. Also, *TFT*2 is more relenting than *TFT*1 for the same reason. The strategy *TFT*4 forgives more than the strategy *TFT*2 as the player who uses *TFT*4 never punishes (transfer from the state *C* to the state *D*) his/her two opponents unless his/her two opponents play *D* together in the previous round, while *TFT*2 punishes his/her two opponents as soon as only one of his/her opponents play *D*. Also, The strategy *TFT*3 forgives more than the strategy *TFT*1 for the same reason.

In summary, we can classify the strategies according to their behaviors of the respond against the other strategies as follows:

- *TFT*1 is fast to anger (after being on the state *C*), where *TFT*1 will defect if one of its two opponents plays *D*, and also is slow to forgive (after being on the state *D*), where *TFT*1 will defect forever like S_0 (*ALLD*) if only one of its two opponents plays *D*. So, the behavior of *TFT*1 is approaching to defection than cooperation.
- *TFT*2 is fast to anger, where it will defect if only one of its two opponents plays *D*, and also it is fast to forgive, where it will cooperate if one of its two opponents plays *C*.
- *TFT*3 is slow to anger, where it will cooperate forever like S_{63} (*ALLC*) if only one of its two opponents plays *C*, and also is slow to forgive, where it will defect forever like S_0 (*ALLD*) if only one of its two opponents plays *D*.
- *TFT*4 is slow to anger, where it will cooperate forever like S_{63} (*ALLC*) if only one of its two opponents plays *C*, and also is fast to forgive, where it will cooperate if one of his two opponents plays *C*. So, the behavior of *TFT*4 is approaching to cooperation than defection.

Appendix

The following is the designed algorithm used for the calculations of the (I3PD) game:
constant Integer ITERATIONS = 9;

```
Declare Integer row1[ITERATIONS], row2[ITERATIONS], row3[ITERATIONS];
Declare character w [ITERATIONS];
For player1 = 0 to 63 do
  For player2 = 0 to 63 do
    For player3 = 0 to 63 do
        runGame(player1, player2, player3)
    ENDFOR
  ENDFOR
ENDFOR
Procedure runGame(player1, player2, player3)
  Declare Integer x[6] = calculate_binary_of (player1);
  Declare Integer y[6] = calculate_binary_of (player2);
  Declare Integer z[6] = calculate_binary_of (player3);
  Declare List of Character resultsOfBloks;
  FOR count = 0 to 8 do
    initialize_rows(count);
    For q = 0 to ITERATIONS - 1 do
        excute_core_game (q);
    ENDFOR
    For i = 0 to ITERATIONS do
        construct_result_row (i);
    ENDFOR
    Declare Character blockOfResult[ ] = calculateRepeatedSequence()
```

```
    resultsOfBloks.add(blockOfResult);
  ENDFOR
  Declare Integer transitionMatrix[ ][ ] =
      transferToTransitionMatrix(x, y, z, resultsOfBloks);
End procedure
Function calculateRepeatedSequence()
  Declare Integer repeatIndex = 0;
  Declare Integer startindex = 0;
  Declare Integer endIndex = 0;
  Declare Boolean found = false;
  FOR i = 0 to q-1
    For k = i+1 to q-1
    IF (w[i] = w[k])
        found = true
        startindex = i
        endIndex = k - 1
        "Leave the loop"
    ENDIF
    ENDFOR
    IF (found = true)
        "Leave the loop"
    ENDIF
  ENDFOR
  Declare Character elements[endIndex - startindex + 1]
  FOR i = startindex to endIndex -1
    elements[i-startindex] = w[i]
    repeatIndex= repeatIndex+1
  ENDFOR
  elements[endIndex - startindex] = w[endIndex];
  return elements;
End function
Procedure transferToTransitionMatrix (x, y, z, resultsOfBloks)
  LOOP resultsOfBloks
  Declare List of Character[ ] newResults
  For i = 0 to blokResult.size-1
    Declare Integer entity[ ] = elements.get(blokResult[i]);
    runNewInitials(entity, newResults); // play game 3 times with 3
  ENDFOR
  LOOP newResults
    int blockResultPosition = getBlockPosition(resultsOfBloks, blokResult);
    int newResultPosition = getBlockPosition(resultsOfBloks, newResult);
    transitionMatrix[blockResultPosition][newResultPosition] += 1;
  ENDLOOP
  ENDLOOP
ENDProcedure
Procedure runNewInitials(Integer[ ] entity, List of Character[ ] newResults)
  Integer newEntity1[ ] = swapDigitOfEntity(entity, 0);
  Integer newEntity2[ ] = swapDigitOfEntity(entity, 1);
  Integer newEntity3[ ] = swapDigitOfEntity(entity, 2);
  newResults.add(excute_core_game(newEntity1));
  newResults.add(excute_core_game(newEntity2));
  newResults.add(excute_core_game(newEntity3));
ENDProcedure
Function getBlockPosition(List of Character[ ] resultsOfBloks, Character[ ] block)
  LOOP resultsOfBloks
    IF block = resultsOfBlok
    return i
    ENDIF
  ENDFOR
    return -1;
ENDfunction
```

Where swapDigitOfEntity just replaces 0 by 1 and vice versa,
the function initialize_rows (count) is declared as:

If count =	Set $r_1[0] =$	$r_2[0] =$	$r_3[0] =$
0	1	1	1
1	0	0	0
2	1	1	0
3	1	0	1
4	1	0	0
5	0	1	1
6	0	1	0
7	0	0	1

, the function excute_core_game (q) is declared as:

If $r_1[q] =$	and $r_2[q] =$	and $r_3[q] =$	Set $r_1[q+1] =$	Set $r_2[q+1] =$	Set $r_3[q+1] =$
1	1	1	$x[0]$	$y[0]$	$z[0]$
0	1	1	$x[3]$	$y[1]$	$z[3]$
1	1	0	$x[1]$	$y[1]$	$z[3]$
0	1	0	$x[4]$	$y[2]$	$z[4]$
1	0	1	$x[1]$	$y[3]$	$z[1]$
0	0	1	$x[4]$	$y[4]$	$z[2]$
1	0	0	$x[2]$	$y[4]$	$z[4]$
0	0	0	$x[5]$	$y[5]$	$z[5]$

, and construct_result_row (i) is declared as:

If $r_1[q] =$	and $r_2[q] =$	and $r_3[q] =$	Set $w[q] =$
0	0	0	P
0	0	1	L
0	1	0	l
0	1	1	T
1	0	0	S
1	0	1	K
1	1	0	k
1	1	1	R

References

[1] R. Axelrod, The Evolution of Cooperation, Basic Books, New York, 2006.
[2] R. Axelrod, W. Hamilton, The evolution of cooperation, Science 211 (1981) 1390.
[3] L. A.Imhof, D. Fundenberg, M.A. Nowak, Tit-for-tat or win-stay, lose-shift? J. Theor. Biol. 247 (2007) 574–580.
[4] M. Nowak, K. Sigmund, A strategy of win-stay lose-shift that outperforms tit-for-tat in the prisoner's dilemma game, Nature 364 (1993) 56–58.
[5] A. Szolnoki, M. Mobilia, L. Jiang, B. Szczesny, A.M. Rucklidge, M. Perc, Cyclic dominance in evolutionary games: a review, J. R. Soc. Interface 11 (2014) 20140735.
[6] J.v. Neumann, O. Morgenstern, Theory of Games and Economic Behavior, Princeton University Press, Princeton, NJ, 1944.
[7] J. W.Friedman, Game Theory with Applications to Economics, Oxford University Press New York, 1990.
[8] P. Hammerstein, R. Selten, Game theory and evolutionary biology, in: Hand-Book of Game Theory with Economic Applications, vol. 2, 1994, p. 929993.
[9] Z. Wang, A. Szolonki, M. Perc, Evolution of public cooperation on interdependent networks: The impact of biased utility functions, EPL 97 (2012) 48001.
[10] A. Haider, Using Genetic Algorithms to Develop Strategies for the Prisoners Dilemma, University Library of Munich, Germany, 2006.
[11] E. ElSeidy, H.K. Arafat, M.A. Taha, The trembling hand approach to automata in iterated games, Math. Theory Model. 3 (2013) 47–65.
[12] F. Chen, A mathematical analysis of public avoidance behavior during epidemics using game theory, J. Theor. Biol. 302 (2012) 1828.
[13] G.Q. Sun, Z. Jin, L.P. Song, A. Chakraborty, B.L. Li, Phase transition in spatial epidemics using cellular automata with noise, Ecol. Res. 26 (2011) 333–340.
[14] J. Tanimoto, T. Fujiki, Z. Wang, A. Hagishima, N. Ikegaya, Dangerous drivers foster social dilemma structures hidden behind a traffic flow with lane changes, J. Stat. Mech. Theory Exp. 11 (2014) P11027.
[15] Y. Shoham, Computer science and game theory, Commun. ACM 8 (2008) 7479.
[16] A. Rapoport, A. Chammah, The Prisoner's Dilemma: A Study in Conflict and Cooperation, Univ. of Michigan Press, Ann Arbor, 1965.
[17] M.A. Nowak, K. Sigmund, E. ElSedy, Automata repeated games and noise, Math. Biol. 33 (1995) 703–722.
[18] J. Andreoni, J.H. Miller, Rational cooperation in the finitely repeated prisoner's dilemma: Experimental evidence, Econ. J. 103 (1993) 570–585.
[19] A. Errity, Evolving Strategies for the Prisoner's Dilemma, Dublin City University, Ireland, 2003.
[20] C. Hilbe, M.A. Nowak, K. Sigmund, Evolution of extortion in iterated prisoner's dilemma games, Proc. Natl. Acad. Sci. 110 (2013) 6913–6918.
[21] Z. Wang, M. Perc, Aspiring to the fittest and promotion of cooperation in the prisoner's dilemma game, Phys. Rev. E82 (2010) 021115.
[22] Z. Wang, A. Szolonki, M. Prec, Percolation threshold determines the optimal population density for public cooperation, Phys. Rev. E85 (2012) 037101.
[23] Z. Wang, A. Szolonki, M. Perc, Interdependent network reciprocity in evolutionary games, Sci. Rep. 3 (2013) 1183.
[24] A. Szolnoki, Z. Wang, M. Perc, Wisdom of groups promotes cooperation in evolutionary social dilemmas, Sci. Rep. 2 (2012) 576.
[25] Z. Wang, A. Szolonki, M. Perc, Optimal interdependence between networks for the evolution of cooperation, Sci. Rep. 3 (2013) 2470.
[26] M. Perc, Z. Wang, Heterogeneous aspirations promote cooperation in the prisoners dilemma game, PLoS ONE 5 (2010) e15117.
[27] Z. Wang, L. Wang, A. Szolnoki, M. Perc, Evolutionary games on multi-layer networks: a colloquium, Eur. Phys. J. B88 (2015) 115.
[28] A.S. Z.Wang, M. Perc, Self-organization towards optimally interdependent networks by means of co-evolution, New J. Phys. 16 (2014) 033041.
[29] M. Perc, A. Szolonki, Social diversity and promotion of cooperation in the spatial prisoners dilemma game, Phys. Rev. 77 (2008) 01190.
[30] M. Perc, A. Szolonki, Does strong heterogeneity promote cooperation by group interactions? New J. Phys. 13 (2011) 123027.
[31] M. Perc, A. Szolonki, Co-evolution of teaching activity promotes cooperation, New J. Phys. 10 (2008) 043036.
[32] M. Perc, A. Szolonki, Evolution of cooperation on scale-free networks subject to error and attack, New J. Phys. 11 (2009) 033027.
[33] A. Szolnoki, M. Perc, Z. Danku, Making new connections towards cooperation in the prisoner's dilemma game, EPL 84 (2008) 5000.

[34] Z. Wang, A. Szolonki, M. Perc, Rewarding evolutionary fitness with links between populations promotes cooperation, J. Theor. Biol. 349 (2014) 50–56.
[35] M. Perc, Coherence resonance in a spatial prisoner's dilemma game, New J. Phys. 8 (2006) 22.
[36] A. Szolnoki, M. Perc, Evolution of extortion in structured populations, Phys. Rev. E89 (2014) 022804.
[37] A. Szolnoki, M. Perc, Defection and extortion as unexpected catalysts of unconditional cooperation in structured populations, Sci. Rep. 4 (2014) 5496.
[38] C.Y. Xia, S. Melomi, M. Perc, Y. Moreno, Dynamic instability of cooperation due to diverse activity patterns in evolutionary social dilemmas, EPL 109 (2015) 58002.
[39] C.Y. Xia, X.K. Meng, Z. Wang, Heterogeneous coupling between interdependent lattices promotes the cooperation in the prisoners dilemma game, Plos One 10 (2015) e0129542.
[40] C.Y. Xia, Q. Miao, J. Wang, S. Ding, Evolution of cooperation in the travelers dilemma game on two coupled lattices, Appl. Math. Comput. 246 (2014) 389398.
[41] M. Perc, A. Szolonki, Co-evolutionary games, a mini review, Bio Syst. 99 (2010) 109–125.
[42] P.D. Taylor, L.B. Jonker, Evolutionary stable strategies and game dynamics, Math. Biosci. 40 (1978) 145–156.
[43] K. Sigmund, M.A. Nowak, Evolutionary game theory, Curr. Biol. 9 (1999) R503–R505.
[44] Z. Wang, M.A. Andrews, Z.X. Wu, L. Wang, C.T. Bauch, Coupled disease behavior dynamics on complex networks: A review, Phys. Life Rev. 15 (2015) 1–29.
[45] J.M. Smith, Evolution and the Theory of Games, Cambridge University Press, Cambridge, 1982.
[46] J. Hofbauer, K. Sigmund, The Theory of Evolution and Dynamic Systems, Cambridge University Press, Cambridge, 1988.
[47] D. Friedman, Evolutionary games in economics, Econometrica 59 (1991) 637–666.
[48] A.F. Kleimenov, Construction of dynamics in repeated three-person game with two strategies for players, in: Proceedings of the IFAC Workshop GSCP, vol. 4, 2004, pp. 132–137.
[49] A.F. Kleimenov, M.A. Schneider, Cooperative dynamics in a repeated three-person game with finite number of strategies, in: Proceedings of the Sixteenth IFAC World Congress, Prague, Czech Republic, 2005.
[50] M. Matsushima, T. Ikegami, Evolution of strategies in the three-person iterated prisoner's dilemma game, J. Theor. Biol. 195 (1998) 53–67.
[51] G.Q. Sun, Z. Jin, L. Li, Q.X. Liu, The role of noise in a predator-prey model with Allee effect, J. Bio. Phys. 35 (2009) 185–196.
[52] G.Q. Sun, Z. Jin, Q.X. Liu, B.L. Li, Rich dynamics in a predator-prey model with both noise and periodic force, Bio Syst. 100 (2010) 14–22.
[53] G.Q. Sun, L. Li, Z. Jin, B.L. Li, Effect of noise on the pattern formation in an epidemic model, Numer. Methods PDE (2010) 1168–1179.
[54] G.Q. Sun, J. Li, B. Yu, Z. Jin, Noise induced pattern transition in a vegetation model, Appl. Math. Comput. 221 (2013) 463–468.
[55] L. Li, Z. Jin, Pattern dynamics of a spatial predator-prey model with noise, Nonlinear Dyn. 67 (2012) 1737–1744.