

Evolution of Grim Trigger in Prisoner Dilemma Game with Partial Imitation

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

Department of Physics,
Hong Kong University of Science and Technology,
Clear Water Bay, Hong Kong, HKSAR
*email: phszeto@ust.hk

Abstract The emergence of Grim Trigger as the dominant strategy in the Iterated Prisoner Dilemma (IPD) on a square lattice is investigated for players with finite memory, using three different kinds of imitation rule: the traditional imitation rule where the entire data base of the opponent's moves is copied, and the two more realistic partial imitation rules that copy only a subset of opponent's moves based on information of games played. We find that the dominance of Grim Trigger is enhanced at the expense of some well known strategies such as tit-for-tat (TFT) when a player has access only to those moves observed in past games played with his opponents. The evolution of the clusters of Grim Trigger in the early stage of the games obeys a common pattern for all imitation rules, before these clusters of Grim Triggers coalesce into larger patches in the square lattice. A physical explanation for this pattern evolution is given. Implication of the partial imitation rule for IPD on complex networks is discussed.

Keywords: Evolutionary Game, Imitation Rule, Prisoner Dilemma

1. Introduction

Evolutionary game [1-4] provides a rich playground for the simulation of multi-agent systems with complex dynamics revealed through the evolving patterns of various strategies used by the players. These spatial-temporal patterns are of interest to many scientists working in various fields, ranging from computer science, physics, ecology and biology. One of the most studied games by political scientists and sociologists is

the Prisoner's Dilemma, as it provides a simple model of the difficulties of cooperation [5-7] in a world populated by egoists. In the Prisoner Dilemma game (PD) two players can choose to cooperate (C) or defect (D). Each player will gain a payoff depending jointly on his choice and the opponent's choice. Cooperation yields a payoff R (S) if the opponent cooperates (defects) and defection yields T (P) if the opponent cooperates (defects). We call R the *Reward* for cooperation, S the *Sucker's payoff*, T the *Temptation to defect* and P the *Punishment*. Typically, $T > R > P > S$ and $2R > T + P$. The Prisoner Dilemma game is a non zero sum game because one player's loss does not equal the opponent's gain. For player without memory, the best strategy for a selfish individual is to defect, although this will result in mutual defection and lead to the worst collective effect for the society. In this PD game, the expectation of defection (D) is greater than the expectation of cooperation (C), independent of the opponent's strategy, even though cooperation yields a higher total payoff for the society. In order to further investigate the emergence of cooperation, a variant of the PD game is the spatial PD game (SPDG), which describes the evolution pattern for a set of players fixed on a lattice, with each player playing the PD game with nearest neighbors. Since now there is a spatial restriction on the players, cooperators can support each other [8,9] and enhance the survival of cooperators [10,11]. For the SPDG, the problem can be mapped onto the statistical mechanics of the two-state Potts model Hamiltonian [2,12] that describes the total income of player i by

$$H_i = \sum_{j(i)} \tilde{S}_i^T A \tilde{S}_j \quad \text{with} \quad A = \begin{pmatrix} R & S \\ T & P \end{pmatrix}, \tilde{S}_i, \tilde{S}_j \in \{\bar{C}, \bar{D}\} \quad \text{and} \quad \bar{C} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}; \quad \bar{D} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad (1)$$

Here \tilde{S}_j is the state vector for the player j who is the neighbor of player i and the state vector can be either one of the two unit vectors $\{\bar{C}, \bar{D}\}$. 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. In the PD game, complication arises for players with the ability to remember a fixed number of the most recent events and supply each player with a set of answers to respond to every possible given history of the game. We call such an answer a "Move". The finite history of the responses of the players is recorded.. The rule that describes what move should be used given a particular history of interaction is called a "Strategy". A complete strategy should include an answer to every possible situation. Players will adapt their strategies, imitating other more successful players following certain "Imitation Rule". Although each player at a particular lattice site has a favorite strategy at time t , he may change to a different strategy at a later time as he realizes from his opponents (his neighbors in SPDG) a better choice. Consequently, the lattice at a particular time can be labeled by a colorful collection of strategies defined on the N sites of the lattice, corresponding to the favorite strategy of the N players at that time. The evolution of this pattern of strategies is one of the main topics of analysis in our present work. Memory is the key that leads to the possible switch of strategy of player i to a new strategy after observing the success of his neighbors, who are his opponents. Without memory, player i will not be able to remember the move of his successful neighbor j ,

thereby imitating the strategy of player j . Now, the key idea of our model comes from the modification of the traditional imitation rule used in the past research on the PD game. The usual imitation rule assumes that the player will copy the complete strategy of his idol, who is a more successful opponent in his encounter. However, if only a subset of the complete strategy of the idol has been used, then it is unrealistic for the player to copy the whole strategy, including the subset that has never been observed. A realistic modification on the imitation rule is to copy only those subsets of the complete strategy 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 observation motivates us to consider a new imitation rule called “partial imitation rule”, as it permits the player to imitate at most the subset of the strategy his idol has used. In real life, a player cannot even remember all the observed moves of his idol. We will formulate our model in Section 2 and the imitation rule in Section 3. The results are discussed in Section 5.

2. 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 moves $S_i S_j$. Here S_i can be either C for “cooperate” or D for “defect”. In any one game between them: $\{S_i S_j\}$ takes on one of these four outcomes $\{CC, CD, DC, DD\}$. For n games, there will be a total of 4^n possible scenarios. A particular pattern of these n games will be one of these 4^n scenarios, and can be described by an ordered sequence of length $2n$ of the form $S_{i1} S_{j1} \dots S_{in} S_{jn}$. 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 pair being the first game played, while the rightmost pair being the outcome of the last game played, or the most recent outcome. (We use capital S to denote the value of either C or D in the history. For example, an ordered sequence of move pairs DDDDDCC represents that the two players cooperate right after the past three mutual defection $\{DD\}, \{DD\}, \{DD\}$.) We use the convention that the outcome $\{S_i S_j\}$, corresponds to S_i being the move made by agent i , who is the player we address, and S_j is the move made by agent j , the opponent of our player. Depending on the player we address, the representation of the same history is not unique. In SPDG, agent j is one of the neighbors of agent i . We say that a player has a memory of fixed-length m , when this player can remember exactly *the* outcomes of the most recent m games. A “Memory” is a sub-sequence of a history. For example, for an agent i with two-game memory ($m=2$), will only has a “Memory” DDCC given a history represented by DDDDDCC. We encode the memory by a bit string using the convention that cooperation is represented by 1 and defection by 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 4^m . (Here 4 refers to the four possible outcomes of one game

which is 00,01,10, and 11). To start the game, let's Consider a non-trivial example when $m=3$. In this case there are $64=4^{m=3}$ possible histories of the strategies used by the two players. We need to reserve 1 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 [13]. After this initial stage, the strategy for our player will need to respond to the game history with a finite memory. Since there are a total of $64=4^{m=3}$ 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 $2^{71} \sim 2.4 \times 10^{21}$ possible strategies. Thus the strategy space for a $m=3$ game is 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 $b(m)=2^m-1+4^m$ and the total number of possible strategies is $|M_m|=2^{b(m)}$. For $m=1$, the enumeration of the encoding of the possible strategies shows that there are 32 possible strategies. This can be seen from Table 1 below.

Table 1. Representation of Strategy Sequence in M_1

| Memorized History | The first move | DD | DC | CD | CC |
|-------------------|----------------|-------|-------|-------|-------|
| Players' strategy | S_0 | S_1 | S_2 | S_3 | S_4 |

The strategy in M_1 can be denoted by $S_0|S_1S_2S_3S_4$. Here the first move is S_0 . If the memory is DD, then the move is S_1 . If the memory is DC, then the move is S_2 . If the memory is CD, then the move is S_3 . If the memory is CC, then the move is S_4 .

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 the complete strategy. 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 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 strategy of his neighbors. In this paper, every agent only knows a subset of the strategy used by a chosen neighbor. For a pair of players (i, j) , playing F games, the focal player i will only observed a set $(S_j(i, j))$ of moves actually used by agent j . This set $S_j(i, j)$ is usually much smaller than the entire set of possible moves corresponding to the strategy of agent j . With this partial knowledge of the moves of the neighbors, the new imitation rule for agent i is called the partial imitation rule. We now give an example to illustrate the difference between partial imitation rule and the traditional one for one step memory. 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 at most 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., $S_0=C$, $S_1=D$, $S_2=C$, $S_3=D$, since S_4 is not applicable as we do not run into S_4 situation since it corresponds to the last pair of moves is CC and our agent i always use D. . 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. If we denote the subset of agent j 's sequence of moves 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 of agent j described by

$$G_j(i, j) = \bigcup_{k \in \Omega(j)} S_j(k, j) \quad (2)$$

where $\Omega(j)$ denotes the nearest neighbors of agent j . Note that this set of sequences of moves used by 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 larger set $G_j(i, j)$.

Let's illustrate pIR2 with an example. 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), agent j uses only S_0 and S_4 of Grim Trigger. 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 is still C|CCCC as they will use C for S_0 and S_4 of Grim Trigger based on the set $S_j(i, j)$. Now, for pIR2, agent i imitates agent j and changes from C|CCCC to the new strategy of agent i as C|DCDC, which is TFT. 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. For tIR, agent i will change from C|CCCC to C|DDDC. For pIR1, agent i will remain C|CCCC. For pIR2, agent i will change from C|CCCC to C|DCDC.

4. Results of Monte Carlo Simulation

In this paper, agents will be placed on a fixed square lattice of size $L \times L$, 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 a number (F) of games with each other. 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. 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 $L \times L$ confrontations are completed. With this stopping probability p , one effectively control the average number of games played between pair of players, thereby determining F . According to our Hamiltonian in Eq.(1), the payoff of agent i after playing a game with agent j is given by the interaction term $\tilde{S}_i^T A \tilde{S}_j$. After F games between these two agents, we obtain the average payoff $U(i)$ and $U(j)$ of agent i and j over these games in this confrontation. The payoff parameters used are $T=5.0$, $R=3.0$, $P=1.0$, $S=0.0$. Agent i will then imitate agent j with probability $P(S_i \rightarrow S_j) = \left(1 + e^{\beta(U(i)-U(j))}\right)^{-1}$. Here, $1/\beta$ represents the thermal noise level. We use $\beta=100$.

In order to verify the correctness of our program on SPDG, we first test our algorithm using the traditional imitation rule. We initialize our strategy sequence with each element assigned “cooperation (C)” or “defection (D)” at equal probability and our results of the simulation shown in Fig.1a is very similar to the published result of Baek and Kim (Fig.3a in [14]). 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. We then use the same program but with the partial imitation rule. In Fig.1(b), we use partial imitation rule 1 (pIR1) and in Fig.1(c), we use pIR2. In both cases, only GT dominates and the concentration of TFT is dramatically reduced to the level of Pavlov and C|CCDC. Results are independent of the lattice size. Next, we should note that the number of games between two players in one encounter is controlled by the probability p to end one confrontation between the players. 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 as the game ends with certainty. When p is smaller than 1, there is a finite probability $(1-p)$ that the agents continue playing games, thereby making use of their memory to activate the various moves of their strategies. In general, we should choose p

sufficiently small so that the number of games played is sufficiently large and memory effect is evident. As our main concern is on the effect of using partial imitation rule on the dominance of various strategies, we use $p=0.05$ so that there are about 20 games played between neighbors. Indeed, we have verified that the general results of our analysis are not sensitive to the values of p , provided that it is below 0.2.

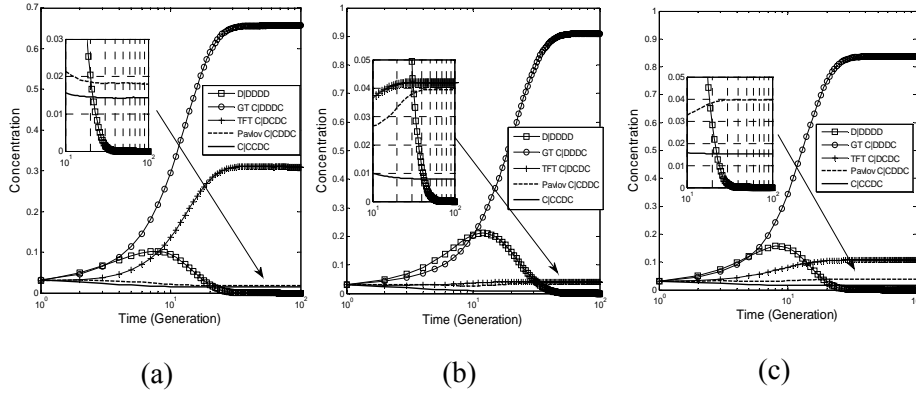


Fig.1. Concentration of important strategies in SPDG on 100x100 square lattice with M1. Result is averaged over 1000 independent simulations, with $\beta=100$ using (a) traditional Imitation Rule (tIR), (b) partial Imitation Rule 1 (pIR1), and (c) partial Imitation Rule 2 (pIR2).

In Fig.2, we show a particular run of the Monte Carlo simulation starting with a randomized initial configuration of players, using three kinds of imitation rules: tIR, pIR1, and pIR2. The time that we make the snapshot are $t=5, 10$ and 20 . The white clusters are the players adopting the GT strategies. These clusters grow till they begin to merge into larger connected clusters. In order to understand the evolution of strategies and the emergence of the dominant clusters of GT, we introduce the following measures for the characterization of the topology of the GT clusters. At a given time, the total number of players adopting the GT strategies can be measured by the total area of the square lattice occupied by GT. Let this total area be $A(t)$. We can also count the length of the boundary between GT and non GT players, and let's denote this boundary as $L(t)$. If we have a single cluster of GT, we can approximate the relation between $L(t)$ and $A(t)$ using a disk of radius $R(t)$, so that $A(t) = \pi R^2(t)$, $L(t) = 2\pi R(t)$. Now, if there are n equal size disks of GT clusters of radius $R(t)$, then we have $A_n(t) = n\pi R^2(t)$, $L_n(t) = 2n\pi R(t)$. Therefore the number of GT clusters can be estimated to be $n(t) = L_n^2(t) / (4\pi A_n(t))$. Since both the total area $A_n(t) = n\pi R^2(t)$ and boundary length $L_n(t) = 2n\pi R(t)$ are measurable, we can obtain the approximate number of GT clusters. Once we obtain $n(t)$, we can obtain the average area of the GT clusters by dividing the total area of GT by $n(t)$: $a(t) = A(t) / n(t)$. Here the total area of GT clusters is denoted by $A(t) \equiv A_n(t)$.

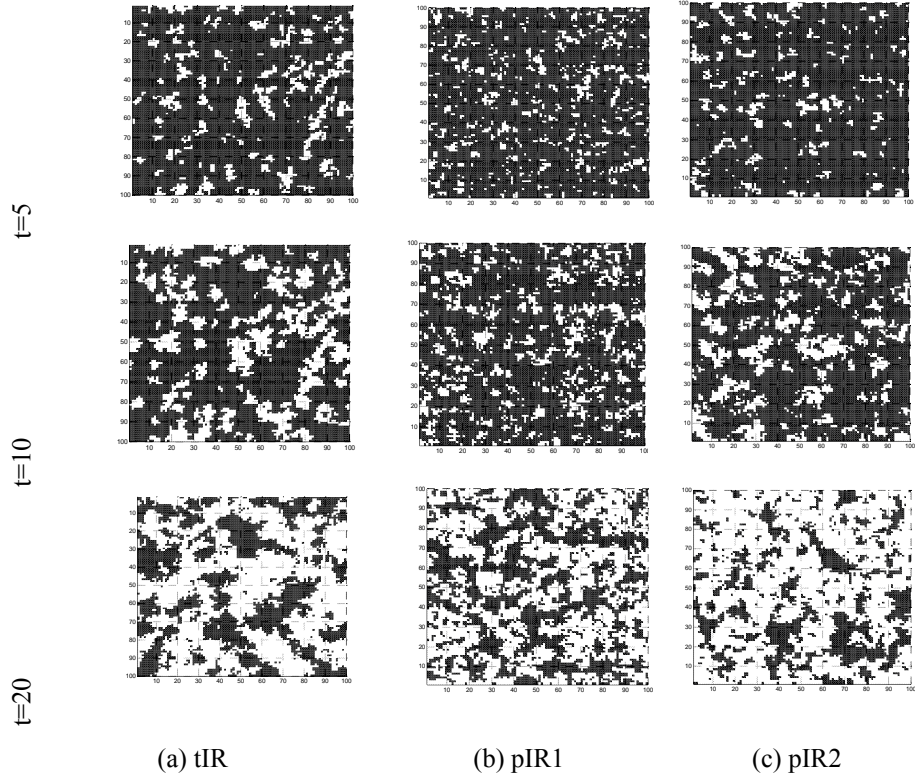


Fig. 2. Evolution patterns of the GT clusters for (a) tIR (b) pIR1 and (c) pIR2 imitation rules at time measured by generation number $t=5, 10$, and 20 .

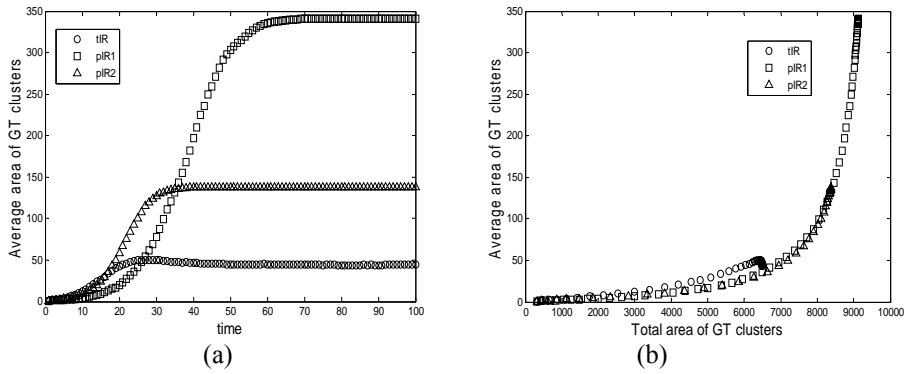


Fig.3. (a) Evolution of the total area occupied by players using GT strategies in the 100x100 square lattice. Time is measured by generation number. (b) Average area per GT clusters vs total area of the GT clusters. It shows a collapse of data for three different imitation rules.

In Fig.3a, we summarize the results by plotting the average total area of GT players in the 100x100 square lattice as a function of time. We perform this analysis of the GT clusters based on the evolution patterns of the SPDG simulation results using different imitation rules as shown in Fig.2. In Fig.3b, we observe an interesting universal curve relating the average area $a(t)$ of a GT cluster and the total area $A(t)$ of GT clusters. We see that for all three imitation rules, the data collapse onto the same curve. The collapse of the data is better at small total area, corresponding to the early stage of evolution shown in Fig.3a: for tIR, the time is less than 25, for pIR2, the time is less than 30, and for pIR1, the time is less than 60. Since the measurement of time is different for different imitation rules, it is easier to measure time of evolution using the total area occupied by GT players. Therefore, the data collapse for the three imitation rules shown in Fig.3b indicates some intrinsic scaling relation of the dynamics of the game. Indeed, for tIR, the saturation of the average area of GT clusters in Fig.3a occurs sooner at time around 25, since there is a complete knowledge of the opponent's moves before imitation. This saturation effect comes from the coalescence of the various GT clusters to form larger and irregular shaped GT clusters. This phenomenon is shown clearly in Fig.2 for a particular run of the evolution of the GT patterns. When the imitation rule is partial, the knowledge of the possible moves by the GT player is less, so the time needed for the saturation of the average area in Fig.3a will be longer for games with partial imitation rule. The fact that the time for saturation for pIR1 is more than pIR2 is then clear, since there is less information on the moves known to the player using pIR1 than pIR2, so saturation occurs sooner in pIR2 than in pIR1. When the information on the moves of one's opponent is less available, it will take more time to realize the advantage of the GT strategy, so that the time for saturation of the average area is longer. Thus, in Fig.2, we see that at time $t=10$, the white clusters for pIR1 (Fig.2b), which has less information on the opponent's moves, are generally smaller than the white clusters for pIR2 (Fig.2c), which has more information. For tIR (Fig.2a), there is complete information, so GT clusters are even larger. After saturation, the system enters into a state of dynamic equilibrium.

5. Conclusion

The memory of the players has important implication on the PD game. In view of the fact that the traditional imitation rule is unrealistic in assuming that a player can copy all the moves of the opponent, we introduce two kinds of partial imitation rules, different by the size of subset of moves observed in past games, and we find very different evolution patterns of various strategies. One major difference is that GT now becomes dominant, and TFT succumbs to the same miserable level of usage as Pavlov. We also observe a universal scaling of the average area of the cluster of GT for all three different imitation rules. This observation implies that there is some hidden scaling relation on the dynamics of SPDG with memory, and the level of partial imitation, as demonstrated by pIR1 and pIR2, corresponds to different region of the universal scaling curve. One generalization that we will further our investigation is to

relate the sequence of partial imitation rule to the propagation of information on the moves of an opponent through his interaction with his nearest neighbors, (pIR1 and pIR2), and next nearest neighbors and so on. In this way, a social network based on the propagation of information on the history of moves by this opponent can be established. It will be a very interesting problem to relate this to “rumors propagation” in complex networks. Finally, our analysis indicates that more realistic players in PD game will prefer using GT than TFT, when they use memory and access local information about the opponent before imitation. This result has important implication of previous studies on PD game as partial knowledge of the opponents’ moves should be the norm rather than the exception in real life.

Acknowledgment K.Y. Szeto acknowledge the support of CERG 602506 and 602507.

References

- [1] J.von Neumann and O. Morgenstern, *Theory of Games and Economic Behaviour*, Princeton University Press, Princeton (1944).
- [2] Gyorgy Szabo, Gabor Fath, *Evolutionary games on graphs*, Physics Reports Volume 446, Issues 4-6, July 2007, Pages 97-216, (2007).
- [3] J. Maynard Smith and G.R. Price, *The logic of animal conflict*, Nature 246, pp. 15–18 (1973).
- [4] J. Maynard Smith, *Evolution and the Theory of Games*, Cambridge University Press, Cambridge (1982).
- [5] Hisashi Ohtsuki, Christoph Hauert, Erez Lieberman, and Martin A. Nowak; *A simple rule for the evolution of cooperation on graphs and social networks*, Nature 441, 502-505 (2006)
- [6] Martin A. Nowak; *Five Rules for the Evolution of Cooperation*; Science 8 December 2006: Vol. 314. no. 5805, pp. 1560 – 1563 (2006)
- [7] R. Axelrod, *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) (1993) 35-78
- [9] Gyrgy Szab, J.V., Szolnoki, A.: *Phase diagrams for an evolutionary prisoner's dilemma game on two-dimensional lattices*. Phys. Rev. E 72(4) (2005) 047107
- [10] Helbing, D., Lozano, S.: *Routes to cooperation and herding effects in the prisoner's dilemma game*. (May 2009)
- [11] Nowak, M.A.: *Five rules for the evolution of co-operation*. Science 314(5805) (December 2006) 1560-1563
- [12] D. Ariosa and H. Fort; “*Extended estimator approach for 2×2 games and its mapping to the Ising Hamiltonian*”, Phys. Rev. E 71, 016132 (2005)
- [13] Bukhari, S.A.H.A.S.: *Using genetic algorithms to develop strategies for the prisoners dilemma*. Asian Journal of Information Technology 8(5) (2006) 866-871
- [14] 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) (2008) 011125