T. Sudo, K. Goto, Y. Nojima, and H. Ishibuchi, Strange evolution behavior of 7-bit binary string strategies in iterated prisoner's dilemma game," Proc. of 2015 IEEE Congress on Evolutionary Computation, pp. 3346-3353, Sendai, Japan, May 25-28, 2015.

Strange Evolution Behavior of 7-bit Binary String Strategies in Iterated Prisoner's Dilemma Game

Takahiko Sudo, Kazushi Goto, Yusuke Nojima, and Hisao Ishibuchi
Department of Computer Science and Intelligent Systems
Graduate School of Engineering, Osaka Prefecture University
Sakai, Osaka 599-8531, Japan
{takahiko.sudo@ci., kazushi.goto@ci., nojima@, hisaoi@}cs.osakafu-u.ac.jp

***Abstract*—The prisoner's dilemma (PD) game is a well-known non-zero sum game. Its iterated version (IPD game) has been widely used to study the evolution of cooperative strategies. In this paper, we assume a noisy environment where a player chooses a different action from the suggested one by its own strategy with a pre-specified error probability. Generally, the noise in action selection makes the evolution of cooperation difficult because the player cannot distinguish between an intentional defection by the opponent's strategy and an unintentional defection by error. However, when a 7-bit binary string with a memory about opponent's two actions was used as a strategy of each player, we observed strange evolution behavior where the use of a small error probability increased the average payoff to the level close to the complete mutual cooperation. That is, the use of a small error probability seems to help the evolution of cooperation. Such a strange behavior was not clearly observed by other types of strategies (e.g., 3-bit binary string with a memory about opponent's single action, 15-bit binary strings with a memory about opponent's three actions). In this paper, we report our simulation results where our focus is placed on the strange evolution behavior of 7-bit binary string strategies. We also try to analyze their strange evolution behavior.**

Keywords—iterated prisoner's dilemma (IPD); evolutionary games; game strategies; evolution of cooperation; error probability.

I. Introduction

The prisoner's dilemma (PD) game provides an interesting decision making model where the rational action selection by a player and an opponent leads to a worse result for them than their irrational cooperative action selection. The iterated prisoner's dilemma (IPD) game is its iterated version, which has been studied in various fields since the 1980s [1]-[3] including the field of evolutionary computation [4]-[6]. In our former studies [7]-[10], we examined effects of various factors on the evolution of cooperation among players with binary string strategies. Examined factors include spatial structures (e.g., two-dimensional gird-worlds, networks), neighborhood size, error probability, strategy type (e.g., deterministic binary string, probabilistic real number string, ensemble of multiple strategies). In general, the evolution of cooperation depends on various factors including them, especially, on the type of strategies and the specification of neighbors of each player for opponent selection and parent selection.

In our previous study [10], we examined the effect of using an ensemble of multiple strategies for action selection. In our computer simulations in [10], we observed strange effects of the error probability on the evolution of cooperation only when we used a 7-bit binary string as a strategy with a memory length two for opponent's two actions. Usually, errors of action selection in the IPD game make the evolution of cooperation difficult. However, mutual cooperation was almost always evolved under the noisy setting with a small error probability when all players used a 7-bit binary string strategy. Moreover, higher average payoff was obtained from the noisy setting with a small error probability than the noise-free setting with the error probability zero.

Since the focus of our former study [10] was action selection by an ensemble of multiple strategies, we did not report the above-mentioned strange evolution behavior of 7-bit binary string strategies. In this paper, we report our simulation results focusing on the strange behavior of 7-bit binary string strategies. We also try to explain why such a strange evolution behavior is observed among players with 7-bit binary string strategies. Through computer simulations, first we examine the evolution of cooperation among players under various settings in order to find the difference in simulation results between 7-bit binary string strategies and other strategies. Then we closely monitor all strategies in the population in order to calculate the percentage of each strategy at each generation in each run. Using the percentage of each strategy, we try to explain drastic increase, drastic decrease and no change of the average payoff over generations in our computer simulations.

This paper is organized as follows. In Section II, we explain our spatial and non-spatial IPD models, 7-bit binary string strategies, and our genetic algorithm for the evolution of IPD game strategies. In Section III, we report the strange evolution behavior of 7-bit binary string strategies and analyze it in detail. Finally, we conclude this paper in Section IV.

II. IPD Game and Strategy Evolution

A. IPD Game

In the prisoner's dilemma (PD) game, a player and an opponent choose either "C: Cooperation" or "D: Defection" separately and simultaneously. Each of them receives a payoff depending on their actions. We use a standard payoff matrix in Table I. The best average payoff over the player and the opponent in Table I is 3, which is obtained from mutual cooperation. The worst average payoff is 1, which is obtained from mutual defection. When one cooperates and the other defects, the average payoff is (0 + 5)/2 = 2.5. If both choose their actions randomly, their expected payoff is calculated for each of them as 2.25 = (3 + 1 + 5 + 0)/4. Average payoff values similar to 2.25 are observed in an initial population in our computer simulations where strategies of all players are randomly initialized.

In the iterated prisoner's dilemma (IPD) game, the PD game is iterated between the same player and the same opponent for a pre-specified number of rounds (100 rounds in our computer simulations). Since the average payoff over the player and the opponent in each round is one of the three possible values in Table I (i.e., 3, 1 or 2.5), we can see that average payoff values larger than 2.75 over many rounds need mutual cooperation in at least more than a half rounds (since $3 \times 0.5 + 2.5 \times 0.5 = 2.75$). For the same reason, average payoff values larger than 2.85 and 2.90 need mutual cooperation in at least more than 70% and 80% of rounds, respectively. From these calculations, we can see that the increase in average payoff shows the evolution of cooperation in the IPD game. On the contrary, the decrease in average payoff shows the evolution of defection. For example, average payoff values smaller than 1.75 means that mutual defection occurs in at least more than 50% of round (since $1 \times 0.5 + 2.5 \times 0.5 = 1.75$).

TABLE I PAYOFF MATRIX OF OUR IPD GAME

Player's Action	Opponent's Action	
	C: Cooperation	D: Defection
C: Cooperation	Player Payoff: 3 Opponent Payoff: 3	Player Payoff: 0 Opponent Payoff: 5
D: Defection	Player Payoff: 5 Opponent Payoff: 0	Player Payoff: 1 Opponent Payoff: 1

B. Strategy Representation by Binary Strings

In the IPD game, a player determines an action according to its strategy based on the opponent's actions in the previous rounds. In this paper, we focus on the behavior of 7-bit binary string strategies, which are explained in Table II. A 7-bit binary string strategy has a memory of length 2, which contains opponent's actions in the previous two rounds. In Table II, the first bit x_1 determines the player's action in the first round. The next two bits x_2x_3 determine the player's action in the second round based on the opponent's action in the first round. If the opponent's action in the first round is D, x_2 is used as the action of the player in the second round (i.e., D ==> x_2). If it is C, x_3 is used (i.e., D ==> x_3). The other four bits $x_4x_5x_6x_7$ determine the player's action in the subsequent rounds depending on opponent's actions in the previous two rounds. For example, x_5 is used as the action of the player in the tth round if the opponent's actions in the (t–2)th round and the (t–1)th round are "D" and "C", respectively (i.e., DC ==> x_5).

TABLE II BINARY STRATEGY OF LENGTH 7

Opponent's action at the (t-2)th round	–	–	–	D	D	C	C
Opponent's action at the (t-1)th round	–	D	C	D	C	D	C
Player's action at the t-th round	x_1	x_2	x_3	x_4	x_5	x_6	x_7

C. Neighborhood Structures

In this paper, we examine two settings of players' spatial structure: a spatial model and a non-spatial model. The non-spatial model is a standard population model where players have no spatial structure. Thus, all the other players are viewed as the neighbors of each player. As a spatial model, we use a two-dimensional grid-world with the torus structure where a player is assigned to each cell. A 7×7 grid-world is shown in Fig. 1 where the von Neumann neighborhood structure with four neighbors is illustrated. As a spatial model, we use a 40×40 two-dimensional grid-world with the von Neumann neighborhood structure in computer simulations in this paper.

Each player selects a pre-specified number of opponents from its neighbors to play the IPD game. Four opponents are selected in our computer simulations. Each player also selects parent strategies from its neighbors and the player itself to generate its new strategy by genetic operators.

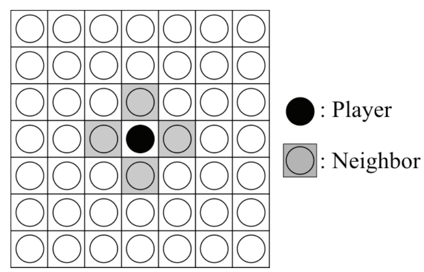


Fig. 1. Illustration of a two-dimensional 7×7 grid-world.

D. Errors in Action Selection

In the IPD game, we assume that each player chooses a different action from the suggested one by its strategy with a pre-specified error probability. If the error probability is 0.01, a player chooses a different action once in the IPD game with 100 rounds on average. The error probability 0 means that each player always uses the suggested action by its strategy.

E. Strategy Evolution

Each player plays the IPD game against a pre-specified number of opponents, which are selected randomly from its neighbors without replacement. The average payoff obtained from the IPD game over those opponents is used as the fitness of the player. After all players complete the IPD game against their opponents, each player updates its own strategy from neighbors' strategies (including the player's own strategy) with high fitness values.

First, a pair of two parents is selected for each player by binary tournament selection with replacement from its neighbors and the player itself. Next, one-point crossover is applied to the strategies of the selected parents to generate two offspring. Then one of the generated offspring is randomly selected. Bit-flip mutation is applied to the selected offspring. The selected offspring after mutation is used as the strategy of the player. The strategy update is performed by all players in a synchronized manner. That is, the IPD game is played by all players in the current population. Then a new strategy is generated for each player from the strategies in the current population. All players update their strategies simultaneously.

III. Computer Simulations

A. Setting of Our Computer Simulation

We use the following setting in our computer simulations:

[Setting of the IPD game]
The number of players (N): $40 \times 40 = 1{,}600$,
Spatial structure: Non-spatial, 2-dimensional grid-world,
Error probability: 0.00, 0.01, 0.02, ..., 0.50,
The number of opponents for each player: 4,
The number of rounds in the IPD game: 100.

[Setting of the genetic algorithm for strategy evolution]
Coding of strategies: binary strings,
String length: 3, 7, 15, 31, 63, 127,
Parent selection: binary tournament selection,
Crossover: One-point crossover with the probability 1.0,
Mutation: bit-flip mutation with the probability: $1/LN$
(L: string length, N: population size),
Number of runs for each setting: 50 runs.

In Section II, we explained the coding of 7-bit binary string strategies in detail. Binary string strategies of different length are coded in the same manner. Let L be the memory length. The string length of binary string strategies of memory length L is calculated as $1+2+2^2+\ ...\ +2^L = 2^{(L+1)}-1$ where the first bit is used for the first round, the next two bits are used for the second round, the next four bit are used for the third round, ..., and the last 2^L bits are used for action selection based on opponent's actions in the previous L rounds.

B. Comparison between 7-bit Strategies and Other Strategies

First, we compare the behavior of 7-bit binary string strategies with other strategies of various length. Fig. 2 shows the results from the spatial model where the von Neumann neighborhood structure is used in the 40×40 grid-world. Fig. 3 shows the results from the non-spatial model where the standard non-spatial population of 1600 players is used. These figures show the average payoff over 50 runs at the 1000th generation. The horizontal axis is the error probability. In Fig. 2 and Fig. 3, each plot shows the average results by binary string strategies with a different string length specification: string length of 3, 7, 15, 31, 63 and 127 (i.e., memory length 1, 2, 3, 4, 5 and 6).

In Fig. 2, we have a similarity curve in each of the six plots. In all the six plots, the highest average payoff close to 3.0 is obtained when the error probability is zero (i.e., the error-free case). When the error probability is 0.5, the action selection becomes random since a different action from the suggested one is selected with the probability 0.5. As a result, the average payoff by the error probability 0.5 is close to 2.25 in all the six plots. As we explained in Section II, 2.25 is the expected payoff by random action selection.

In each of the six plots in Fig. 2, the increase of the error probability from 0.0 to a value in [0.25, 0.3] decreases the average payoff from 3.0 (100% mutual cooperation) to about 1.75 (at least 50% mutual detection). The increase of the error probability in this range gradually makes mutual cooperation difficult and mutual defection easy. After the minimum average payoff, the increase in the error probability gradually increases the average payoff since the behavior of the IPD game becomes close to random action selection. In each plot in Fig. 2, we can also observe an increase of the average payoff by the increase of the error probability around 0.1. Such a hill in each plot around the error probability 0.1 becomes small and unclear by increasing the string length in Fig. 2.

The comparison between Fig. 2 and Fig. 3 shows that the spatial structure helps the evolution of cooperation in Fig. 2. Whereas the average payoff close to 3.0 is obtained in all the six plots in Fig. 2, such a high average payoff is obtained only in Fig. 3 (c) with 15-bit binary string strategies. Moreover, errors in action selection decrease the average payoff more severely in Fig. 3 than Fig. 2.

The strange evolution behavior in Fig. 3 (b) by 7-bit binary string strategies is the increase of the average payoff by a small increase of the error probability from zero. As a result, higher average payoff is obtained from the noisy setting with a small error probability (e.g., 0.02) than the error-free setting with the error probability zero in Fig. 3 (b).

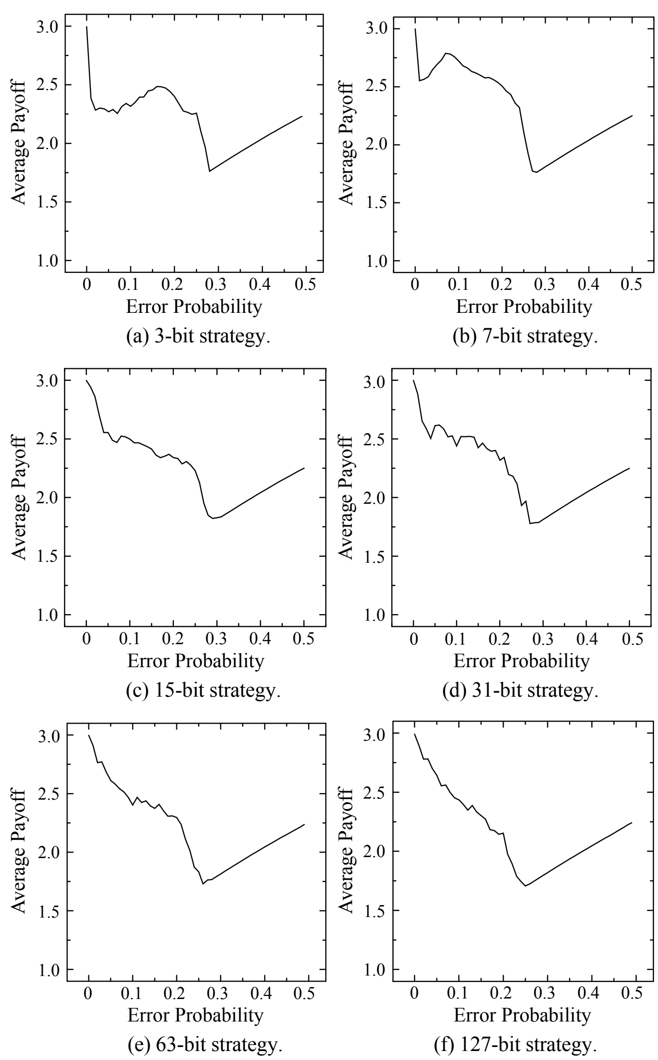
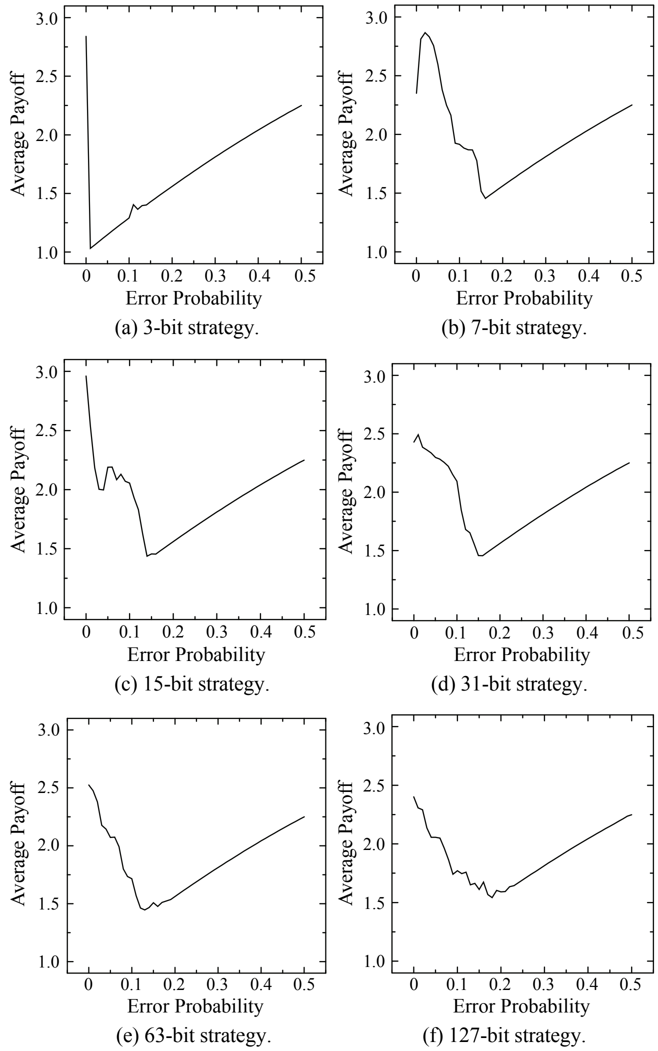


Fig. 2. Average payoff at the 1000th generation in the spatial IPD game with four neighbors in the von Neumann neighborhood structure.



(a) 3-bit strategy. (b) 7-bit strategy.

(c) 15-bit strategy. (d) 31-bit strategy.

(e) 63-bit strategy. (f) 127-bit strategy.

Fig. 3. Average payoff at the 1000th generation of the non-spatial IPD model where all the over players are neighbors of each player.

C. *Analysis of the Strange Behavior of 7-bit Strategies*

Let us examine the strange evolution behavior among players with 7-bit binary string strategies by observing a single run in detail. Fig. 4 shows single run results in the non-spatial IPD game in Fig. 3 (b) with 7-bit binary string strategies. The error probability is specified as zero in Fig. 4 (a) and 0.02 in Fig. 4 (b). There is no other difference in the setting of computer simulations in Fig. 4.

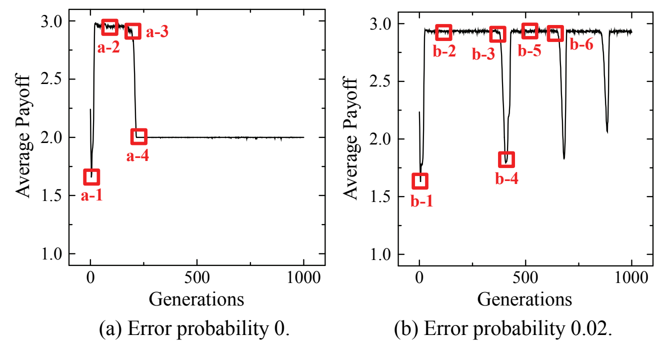
In Fig. 4 (a) in the noise-free setting with no errors, the average payoff rapidly increases to 3.0 in the first 20 generations. Then it continues to be close to 3.0 for about 200 generations. After those stable generations with the average payoff close to 3.0, the average payoff quickly drops to 2.0. After that, it is stable at the same level of the average payoff until the termination of this run at the 1000th generation. That is, the average payoff is 2.0 in the last 750 generations.

In Fig. 4 (b) with the error probability 0.02, the average payoff is close to 3.0 except for very early generations and the three deep and narrow valleys. It is interesting to observe that the average payoff quickly recovers from the three sharp drops to about 3.0 in Fig. 4 (b) whereas it continues to be 2.0 after the sharp drop in Fig. 4 (a).

In Fig. 4 (a) and Fig. 4 (b), we mark some turning points and stable generations for further examination as follows (the close-up of each part is shown in Fig. 5 for Fig. 4 (a) and Fig. 6 for Fig. 4 (b)):

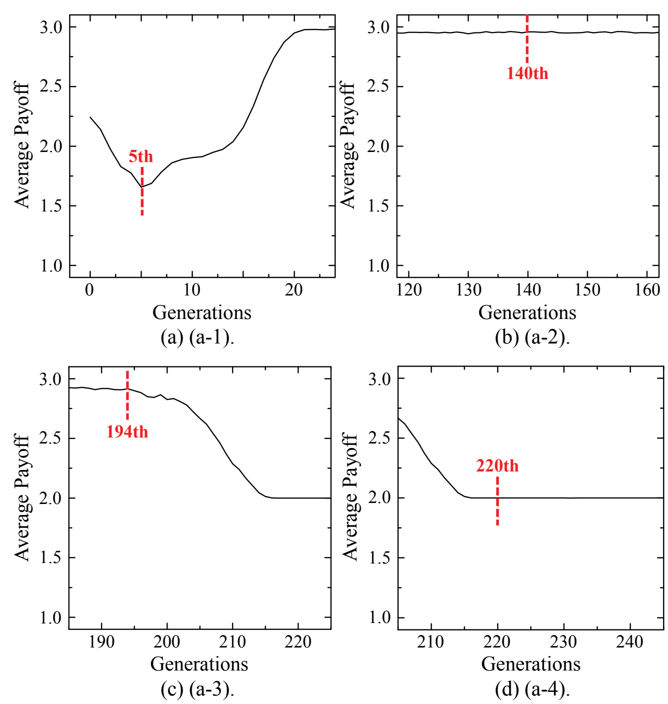
Four Red Boxes in Fig. 4 (a):
- (a-1): Start of the rapid increase to the level of (a-2)
- (a-2): Stable generations with high average payoff
- (a-3): Start of the rapid drop to the level of (a-4)
- (a-4): Stable generations with the average payoff 2.0

Six Red Boxes in Fig. 4 (b)
- (b-1): Start of the rapid increase to the level of (b-2)
- (b-2): Stable generations with high average payoff
- (b-3): Start of the rapid drop from the level of (b-2)
- (b-4): Start of the rapid recovery to the level of (b-5)
- (b-5): Stable generations with high average payoff
- (b-6): Start of the rapid drop from the level of (b-5)



(a) Error probability 0. (b) Error probability 0.02.

Fig. 4. Single run results by 7-bit strategies in the non-spatial IPD model.



(a) (a-1). (b) (a-2).

(c) (a-3). (d) (a-4).

Fig. 5. Close-ups of the marked parts in Fig. 4 (a).

In Fig. 5, we shows close-ups of the marked four parts in Fig. 4 (a). It seems from Fig. 5 (a) that the 5th generation is a turning point generation, So, we choose this generation for close examination. We also choose the 140th, 194th and 220th generations for close examination as shown in Fig. 5. At each generation, we count the number of players with each strategy. Results are summarized in Tables III-VI where "1" and "0" mean "cooperation" and "defection" respectively. In each table, strategies are sorted based on the number of players using each strategy. For example, the second row of Table III shows that the 7-bit binary string strategy 0100010 is used by 84 players in the 5th generation.

TABLE III STRATEGIES IN A-1 (5TH GENERATION)

Strategy	The number of players
0100010	84
0110010	80
1000010	79
1010010	74
0000010	71
0010010	69
1100010	62
1110010	53
0000000	41

TABLE IV STRATEGIES IN A-2 (140TH GENERATION)

Strategy	The number of players
0100111	1057
0110111	322
0100011	168
0110011	53

TABLE V STRATEGIES IN A-3 (194TH GENERATION)

Strategy	The number of players
0100111	1245
0100011	171
0110111	98
0100110	61
0110011	18
0110110	4

TABLE VI STRATEGIES IN A-4 (220TH GENERATION)

Strategy	The number of players
0100010	1470
0100011	81
0110010	46
0110011	3

From Table III, we can see that various strategies exist at the 5th generation. The average payoff at this generation is about 1.7 (see Fig. 5 (a)). This means that more than 50% rounds in the IPD game are mutual defection. In this situation, mutual cooperation is rapidly evolved within the next 15 generations as shown in Fig. 5 (a). Since the same behavior is observed in Fig. 4 (a) and Fig. 4 (b) in very early generations (i.e., in Fig. 5 (a) and Fig. 6 (a)), this part is not likely to explain the strange behavior of 7-bit binary string strategies in the non-spatial noisy IPD model in Fig. 3 (b). So, we do not examine each strategy in Table III in detail.

In Table IV, four strategies at the 140th generation are shown. It should be noted that the sum of the number of players is 1600, which is the same as the total number of players. This means that there is no other strategy at the 140th generation.

As we have explained using Table II in Section II, the first bit x_1 of the 7-bit binary string strategy $x_1x_2x_3x_4x_5x_6x_7$ is used as the player's first action. The next two bits x_2x_3 are used as the player's second action based on the opponent's first action as follows: D ==> x_2, and C ==> x_3. The last four bits are used for action selection in the other rounds based on the opponent's previous two actions as follows: DD ==> x_4, DC ==> x_5, CD ==> x_6, and CC ==> x_7. Using these action selection rules, let us examine the four strategies in Table IV.

Since the first bit of all the four strategies is "0", all players defect at the first generation. After the defection by the opponent in the first round, all players cooperate in the second round since the second bit of all strategies is "1". In the noise-free setting with no error, the third bit is never used since no players cooperate in the first round. Thus the first two strategies in the 140th generation in Table IV can be viewed as the same strategy 01*0111 where "*" is a wild card (i.e., "don't care" symbol). We refer to this strategy as the majority strategy in the 140th generation. For the same reason, the other two strategies can be viewed as the same strategy 01*0011. We refer to this strategy as the minority strategy in the 140th generation. Since the first two bits of all strategies are "01", the first two actions of all players in the 140th generation are "DC".

When two players with the majority strategy 01*0111 play the IPD game, their third actions are C by the fifth bit "1" after DC. Then their fourth actions are C by the seventh bit "1" after CC. After the fourth round, C is always selected by the two players with the majority strategy. That is, their actions in the 100 rounds are DCCC ... C. In this case, the average payoff of each player is 2.98: (Majority, Majority) = (2.98, 2.98). When two players with the minority strategy 01*0011 play the IPD game, their actions in the 100 rounds are DCDC ... DC with the average payoff 2.00: (Minority, Minority) = (2.00, 2.00). When the minority strategy 01*0011 plays against the majority strategy 01*0111, the minority strategy's actions are DCDCC ... C with the average payoff 3.00 while the majority strategy's actions are DCCCC ... C with the average payoff 2.95: (Minority, Majority) = (3.00, 2.95). These calculations of the average payoff are summarized in Table VII.

TABLE VII Average PAYOFF OF A PLAYER WITH EACH STRATEGY AT THE 140TH GENERATION IN TABLE IV

Player's Strategy	Opponent's Strategy	
	Majority 01*0111	Minority 01*0011
Majority 01*0111	Player 2.98	Player: 2.95
Minority 01*0011	Player: 3.00	Player: 2.00

As shown in Table VII, the minority strategy can obtain a slightly higher average payoff 3.00 than the majority strategy 2.95 when the two strategies play the IPD game. However, the average payoff from the IPD game between the two minority strategies (i.e., 2.00) is much lower than that between the two majority strategies (i.e., 2.98). If the number of players with the minority strategy increases, their average payoff decreases since the probability of choosing an opponent with the minority strategy increases. Thus the number of minority strategy players is not likely to increase. However, since the minority strategy can obtain a slightly higher payoff than the majority strategy from the IPD game between them, the number of minority players is not likely to decrease to zero. In our computational experiments in Table IV, four opponents are randomly selected for each player from the other 1599 players. For a minority strategy player, there exist the following five possible cases with respect to its average payoff over the IPD game against the four opponents:

2.00: Against four minority strategy players.
2.25: Against three minority and one majority strategy players.
2.50: Against two minority and two majority strategy players.
2.75: Against one minority and three majority strategy players.
3.00: Against four majority strategy players.

Among these five cases, a minority strategy player can obtain higher average payoff than a majority strategy player only in the last case since the average payoff of the majority strategy is at least 2.95 (see Table VII). The probability of this case among the possible five cases for a minority player can be calculated as $n(n-1)(n-2)(n-3)/(1599\times 1598\times 1597\times 1596)$ where n is the number of players with the majority strategy. This probability is about 0.5 when n = 1344.83. That is, when the number of minority strategy players is 1600 − 1345 = 255, about a half of them have higher average payoff than majority strategy players. Thus it is likely that the number of minority strategy players does not increase or decrease when about 255 players have the minority strategy in the population of 1600 players. The actual number of minority strategy players in Table IV is 221, which is similar to 255.

The above calculation also suggests that the majority strategy in the 140th generation continues to be used by a majority of players (i.e., by about 1345 players) if no other strategies invade into the population. Actually, in the 194th generation in Table V, 1343 players still have the majority strategies 01*0111. Since there is no large change in the number of players with the majority strategy between the 140th generation (i.e., 1379) and the 194th generation (i.e., 1343), the average payoff does not change during the 54 generations after the 140th generation in Fig. 4 (a) (also see Fig. 5 (b)).

From the comparison between Table IV and Table V, we can see that another type of strategies with the form "01*0110" emerges during the 54 generations after the 140th generation. We refer to this type "01*0110" as the new strategy in the 194th generation. In Table V, 61 + 4 = 65 players have this type. In Table V, we have 1343 players with the majority strategy 01*0111, 189 players with the minority strategy 01*0011, and 65 players with the new strategy (Other three players have different strategies). The main feature of the new strategy is "0" of the last bit x_7, which means the choice of D after the opponent's action C in the previous two rounds: CC ==> D.

When the new strategy plays the IPD game against the majority strategy, the actions and the average payoff of each strategy are as follows:

01*0110: DCCDDD CDCDDD CDCDDD ... CDCD: 3.16
01*0111: DCCCCD DCCCCD DCCCCD ... DCCC: 1.51

The new strategy can obtain high average payoff from the IPD game against the majority strategy. The average payoff obtained from each strategy in Table V in the 194th generation is summarized in Table VIII. From Table VIII, it is very likely that the number of players with the strategy continues to increase. This is because (i) the new strategy can obtain the highest average payoff 3.16 from the IPD game against the majority strategy, (ii) the increase in the number of new strategy players severely decreases the expected average payoff of the majority strategy, and (iii) the expected average payoff of the minority strategy is lower than that of the new strategy independent of the number of players with each strategy.

TABLE VIII Average Payoff of a Player with Each Strategy at the 194th Generation in Table V

Player's Strategy	Opponent's Strategy		
	Majority 01*0111	Minority 01*0011	New 01*0110
Majority 01*0111	Player: 2.98	Player: 2.95	Player: 1.51
Minority 01*0011	Player: 3.00	Player: 2.00	Player: 2.32
New 01*0110	Player: 3.16	Player: 2.32	Player: 2.32

However, the new strategy 01*0110 is not in Table VI at the 220th generation where 01*0010 is used by many players. In the IPD game by these two strategies, the actions and the average payoff of each strategy over the 100 rounds are

01*0110: DCCCDD CCDD CCDD ... CCDD CD: 1.48
01*0010: DCDDDC DDDC DDDC ... DDDC DC: 2.78

Let us assume a population of players with these strategies. The average payoff from the IPD game in such a population is calculated as shown in Table IX. Independent of the number of players with the new strategy, the expected average payoff of the strategy 01*0010 in Table IX is larger than that of the new strategy. Thus, the strategy 01*0010 in the 220th generation can take over the population. Actually, no player in the 220th generation in Table VI has the new strategy 01*0110.

TABLE IX Average Payoff from 01*0110 and 01*0010

Player's Strategy	Opponent's Strategy	
	New 01*0110	01*0010
New 01*0110	Player: 2.32	Player: 1.48
01*0010	Player: 2.78	Player: 2.00

It should be noted that the same results are obtained from the majority strategy 01*0111 in the 194th generation against the strategy 01*0010 in the 220th generation as follows:

01*0111: DCCCDD CCDD CCDD ... CCDD CD: 1.48
01*0010: DCDDDC DDDC DDDC ... DDDC DC: 2.78

This is because the last bit x_7 (CC => x_7) is not used in the IPD game against 01*0010. That is, both the new strategy 01*0110 and the majority strategy 01*0111 in the 194th generation can be viewed as 01*011* in the IPD game against the strategy 01*0010 in the 220th generation.

In the 220th generation in Table VI, all the four strategies can be denoted as 01*001*. When the IPD game is played between players with strategies of this type, their actions are always DCDCDC ... DC. The average payoff is 2.00. After the 220th generation, the average payoff continues to be 2.00 until the end of this run (i.e., 1000th generation) in Fig. 4 (a). This observation shows the difficulty of the evolution of mutual cooperation (and also mutual defection) in a population with a majority of players having the strategy 0100010 in Table VI.

Now, let us examine the result of a single run in Fig. 4 (b) under the noisy setting with the error probability 0.02. Under this noisy setting, each player chooses a different action from the suggested one by its own strategy twice on average over the 100 rounds in the IPD game. Fig. 6 shows close-ups of the marked six parts in Fig. 4 (b). From Fig. 6, we choose the 5th, 140th, 377th, 409th, 535th and 651th generations for close examination. Tables X-XV show the number of players with each strategy observed in each generation.

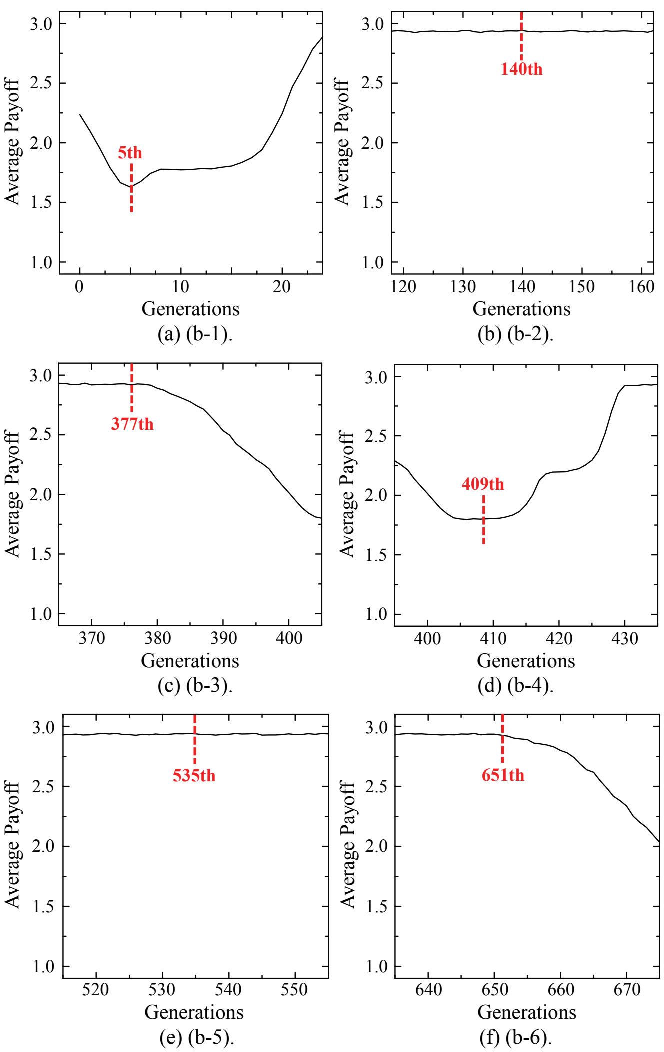


Fig. 6. Close-ups of the marked parts in Fig. 4 (b).

In the very early generations around the 5th generation, we can observe a similar rapid increase of the average payoff in Fig. 5 (a) and Fig. 6 (a). Table X for Fig. 6 (a) is also similar to Table III for Fig. 5 (a) in the sense that various strategies are included in the population in both Table III and Table X.

In the 140th generation in Table IV in the noise-free setting, we had only two types of strategies: 01*0111 (1379 players) and 01*0011 (221 players). In the noisy setting in Table XI, the number of players with each type is as follows: 01*0111 (1332 players) and 01*0011 (240 players). Since the same type of strategies are used in many players in the 140th generations in the noise-free and noisy settings, the average payoff is almost the same in these two settings in Fig. 5 (b) and Fig. 6 (b).

Strategies in Table XII in the 377th generation in the noisy setting are similar to those in Table V in the 194th generation in the noise-free setting: 01*0111 (1103 players), 01*0011 (189 players) and 01*0110 (13 players) in Table XII, and 01*0111 (1343 players), 01*0011 (189 players) and 01*0110 (65 players) in Table V. The shape of the decrease of the average payoff is similar between Fig. 5 (c) and Fig. 6 (c). These observations suggest that the first sharp decrease from high average payoff happens in similar situations in the noise-free setting and the noisy setting. One clear difference between Table V and Table XII is the value of the third bit x_3 (i.e., C => x_3). In Table V in the noise-free setting, x_3 is never used since the first bit x_1 is 0 in all strategies in Table V (i.e., D in the first round). However, in the noisy-setting with the error probability 0.02, the first action is D with the probability 0.02 whereas the first bit x_1 is 0 in all strategies in Table XII. In order to choose C in the second round even when C is selected by error in the first round, x_3 is evolved as 1 in all strategies in Table XII. Another difference is the existence of 0111111 in Table XII.

TABLE X STRATEGIES IN B-1 (5TH GENERATION)

Strategy	The number of players
0010010	85
0100010	82
1100010	75
1110010	73
1010010	72
0110010	68
0000010	63
1100100	55
1000010	54

TABLE XI STRATEGIES IN B-2 (140TH GENERATION)

Strategy	The number of players
0110111	1222
0110011	225
0100111	110
0100011	15
0111011	12
0111111	12

TABLE XII STRATEGIES IN B-3 (377TH GENERATION)

Strategy	The number of players
0110111	1103
0111111	229
0110011	189
0111011	38
0110101	20
0110110	13

TABLE XIII STRATEGIES IN B-4 (409TH GENERATION)

Strategy	The number of players
0110010	1327
0110011	273

TABLE XIV STRATEGIES IN B-5 (535TH GENERATION)

Strategy	The number of players
0110111	1361
0110011	238
0100110	1

TABLE XV STRATEGIES IN B-6 (651TH GENERATION)

Strategy	The number of players
0110111	1158
0111111	190
0110011	186
0111011	30
0110110	18

In the 409th generation in Table XIII at the bottom of the valley in Fig. 6 (d), we have only two strategies: 0110010 (1327 players) and 0110011 (273 players). These strategies are the same as the two strategies in the 220th generation in the noise-free settings: 01*0010 (1516 players) and 01*0011 (84 players) except for the third bit x_3. Whereas no cooperation is evolved after the 220 generation in Fig. 5 (d), the average payoff recovers rapidly in Fig. 6 (d). Then, 0110111 becomes the majority strategy used by 1361 players in Table XIV in the 535th generation in Fig. 6 (e) as in Table XI in the 140th generation in Fig. 6 (c) where 0110111 is used by 1222 players. After that, Table XV in the 651th generation is similar to Table XII in the 377th generation. That is, similar situations seem to be iterated, which lead to the three valleys in Fig. 4 (b).

From these discussions, we can see that the main difference between Fig. 5 in the noise-free setting and Fig. 6 in the noisy setting is the recovery from a population of 01*0010 players and 01*0011 players in Fig. 6 (d). In Table XVI and Table XVII, we show the average payoff obtained by 0110010, 0110011 and 0110111. Table XVII suggests the possibility of the increase in the number of 0110011 players in a population with 0110010 and 0110011 (see Table XIII) and the increase in the number of 0110111 players in a population with 0110011 and 0110111 (see Table XIV). These possibilities may be the reason for the rapid recovery only in the Fig. 6 (d).

Finally, we show simulation results by 7-bit binary string strategies in the non-spatial model for various specifications of the population size (from 25 to 2500) in Fig. 7. In this figure, the strange behavior is clearly observed only when we use a large population with more than 200-300 players.

TABLE XVI Average PAYOFF IN THE NOISE-FREE SETTING

Player's Strategy	Opponent's Strategy		
	0110010	0110011	0110111
0110010	Player: 2.00	Player: 2.00	Player: 2.78
0110011	Player: 2.00	Player: 2.00	Player: 3.00
0110111	Player: 1.48	Player: 2.95	Player: 2.98

TABLE XVII Average PAYOFF IN THE NOISY SETTING

Player's Strategy	Opponent's Strategy		
	0110010	0110011	0110111
0110010	Player: 1.79	Player: 1.83	Player: 2.63
0110011	Player: 1.77	Player: 2.19	Player: 3.01
0110111	Player: 1.46	Player: 2.87	Player: 2.96

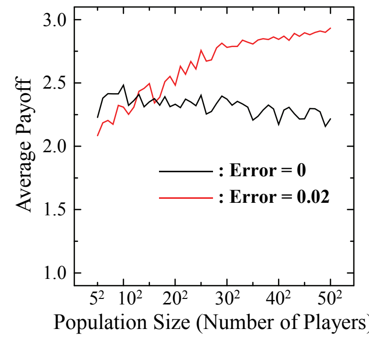


Fig. 7. Average payoff at the 1,000th generation over 50 runs.

IV. CONCLUSIONS

In this paper, we demonstrated that the average payoff can be increased by increasing the error probability from 0 to 0.02 in the IPD model with 7-bit binary string strategies. One possible reason for such a strange behavior was suggested by our computer simulations: An appropriate probability of errors can help a rapid recovery from a rapid drop of the average payoff by assigning different average payoff to strategies that have the same average payoff in the noise-free setting. We also showed that the strange behavior happened only when the population size was larger than 200-300 in the non-spatial model. Our observation seems to be similar to a positive effect of noises reported in [6] where the possibility of the noise level optimization was suggested for IPD game strategy evolution.

REFERENCES

[1] R. Axelrod, "The evolution of strategies in the iterated prisoner's dilemma," in L. Davis (ed.), *Genetic Algorithms and Simulated Annealing*, Morgan Kaufmann, pp. 32-41, 1987.

[2] D. B. Fogel, "Evolving behaviors in the iterated prisoner's dilemma," *Evolutionary Computation*, vol. 1, no. 1, pp. 77-97, 1993.

[3] G. Kendall, X. Yao, and S. Y. Chong (eds.), *The Iterated Prisoners' Dilemma: 20 Years*, World Scientific, Singapore, 2007.

[4] D. Ashlock, E. Y. Kim, and N. Leahy, "Understanding representational sensitivity in the iterated prisoner's dilemma with fingerprints," *IEEE Trans. on Systems, Man, and Cybernetics: Part C*, vol. 36, no. 4, pp. 464-475, July 2006.

[5] D. Ashlock and E. Y. Kim, "Fingerprinting: Visualization and automatic analysis of prisoner's dilemma strategies," *IEEE Trans. on Evolutionary Computation*, vol. 12, no. 5, pp. 647-659, October 2008.

[6] D. Ashlock, E. Y. Kim, and W. Ashlock, "Fingerprint analysis of the noisy prisoner's dilemma using a finite-state representation," *IEEE Trans. on Computational Intelligence and AI in Games*, vol. 1, no. 2, pp. 154-167, June 2009.

[7] H. Ishibuchi and N. Namikawa, "Evolution of iterated prisoner's dilemma game strategies in structured demes under random pairing in game playing," *IEEE Trans. on Evolutionary Computation*, vol. 9, no. 6, pp. 552-561, 2005.

[8] H. Ishibuchi, H. Ohyanagi, and Y. Nojima, "Evolution of strategies with different representation schemes in a spatial iterated prisoner's dilemma game," *IEEE Trans. on Computational Intelligence and AI in Games*, vol. 3, no. 1, pp. 67-82, March 2011.

[9] H. Ishibuchi, K. Hoshino, and Y. Nojima, "Strategy evolution in a spatial IPD game where each agent is not allowed to play against itself," *Proc. of 2012 IEEE Congress on Evolutionary Computation*, pp. 688-695, Brisbane, Australia, June 10-15, 2012.

[10] T. Sudo, Y. Nojima, and H. Ishibuchi, "Effects of ensemble action selection on the evolution of iterated prisoner's dilemma game strategies," *Proc. of 2014 IEEE Congress on Evolutionary Computation*, pp. 1195-1201, Beijing, China, July 6-11, 2014.