Effects of Configuration of Agents with Different Strategy Representations on the Evolution of Cooperative Behavior in a Spatial IPD Game

Hisao Ishibuchi, Keisuke Takahashi, Kouichirou Hoshino, Junpei Maeda, and Yusuke Nojima

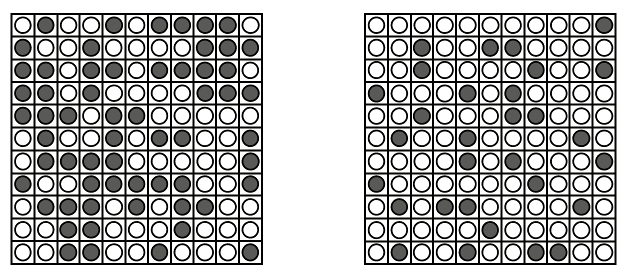
***Abstract*—It is well-known in the literature that the choice of a strategy representation scheme has a large effect on the evolution of cooperative behavior in an IPD (Iterated Prisoner's Dilemma) game. Different representation schemes often lead to totally different results. The evolution of cooperative behavior is easy under some representation schemes while it is difficult under other schemes. An interesting research issue is to examine the evolution of cooperative behavior in an inhomogeneous population consisting of heterogeneous agents with different representation schemes. In our former study, we used an inhomogeneous population where one of two representation schemes was randomly assigned with the same probability to each cell in a two-dimensional grid-world. In this setting, every agent is likely to have both types of representation schemes in its neighborhood. That is, an IPD game is likely to be played not only between homogeneous agents with the same representation scheme but also between heterogeneous agents with different representation schemes everywhere in the grid-world. In this paper, we examine the effect of spatial configurations of two types of agents on the evolution of cooperative behavior in a spatial IPD game. In computational experiments, agents with one representation scheme are clustered in a specific region of a two-dimensional grid-world (e.g., 7x7 cells at the center of an 11x11 grid-world). Agents with the other representation scheme are placed in the remaining cells. In this setting, the interaction through the IPD game between heterogeneous agents is limited only around the boundary between the two types of agents in the grid-world. Experimental results are compared between this setting and the random assignment in our former study.**

I. Introduction

Iterated prisoner's dilemma (IPD) games [1]-[3] have often been used to examine the effect of the choice of a strategy representation scheme on strategy evolution. It has been demonstrated by Ashlock et al. [4]-[6] that the choice of a representation scheme has large effects on the evolution of cooperative behavior. They examined various representation schemes such as a finite-state machine, a neural network, and a look-up table. Totally different results were often obtained from different representation schemes. The evolution of cooperative behavior was very fast under some representation schemes while it was very slow under other schemes. It should be noted that a population of agents with the same representation scheme was used in each trial in those studies in order to examine the characteristic features of each representation scheme (while a wide variety of different representation schemes were examined in those studies).

H. Ishibuchi, K. Takahashi, K. Hoshino, J. Maeda, and Y. Nojima are with Department of Computer Science and Intelligent Systems, Osaka Prefecture University, 1-1 Gakuen-cho, Naka-ku, Sakai, Osaka 599-8531, Japan (e-mail: {hisaoi@, keisuke.takahashi@ci., kouichirou.hoshino@ci., jumpei.maeda@ci., nojima@}cs.osakafu-u.ac.jp).

An interesting research issue is to examine the evolution of cooperative behavior in an inhomogeneous population of heterogeneous agents with different representation schemes. In our former study [7], we generated an inhomogeneous population with two types of agents for a spatial IPD game by randomly assigning one of two representation schemes to each cell in a two-dimensional grid-world (see Fig. 1 (a)). As a result, each of the two representation schemes was randomly and uniformly distributed over the grid-world. In this setting, many agents are likely to have neighbors with different representation schemes. This means that the IPD game is likely to be played between heterogeneous agents (as well as between homogeneous ones) everywhere in the grid-world. In Fig. 1 (a), one of two representation schemes is selected with the same probability (i.e., a 50% chance) for each cell. It is also possible to use different probabilities for different representation schemes. In some computational experiments of our former study [7], one representation scheme was assigned to each cell with a 75% chance. Fig. 1 (b) shows an example of such a biased random assignment.



(a) Totally random assignment. (Each type of agents: 50%)

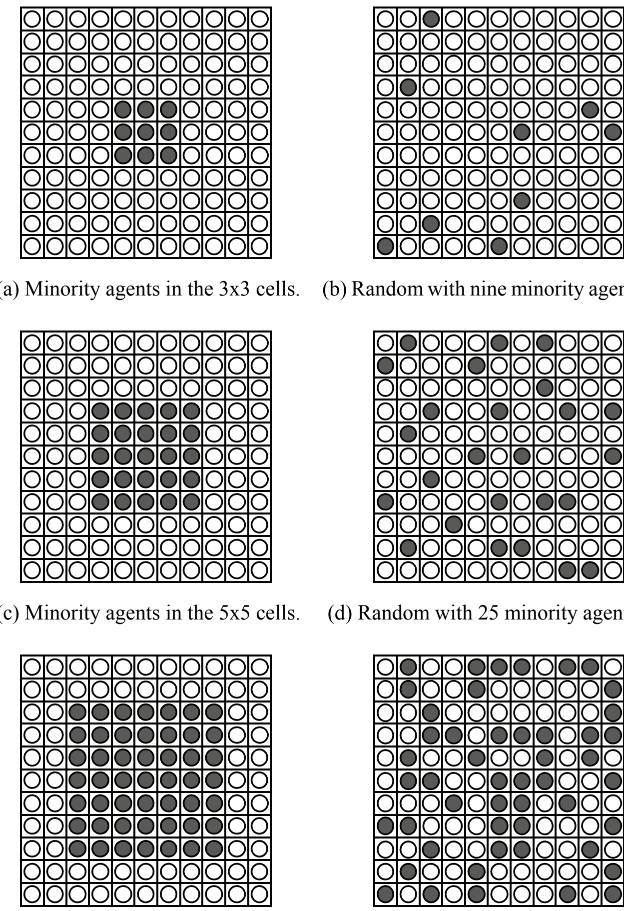
(b) Biased random assignment. (Majority agents: 75%)

Fig. 1. Illustration of the spatial configuration of heterogeneous agents in our former study [7]. For each cell, one of two types of agents is randomly chosen. Both types have a 50% chance to be selected in (a) while the majority one has a 75% chance in (b).

The aim of this paper is to examine the effect of spatial configurations of heterogeneous agents on the evolution of cooperative behavior in a special IPD game. The main motivation is that the random assignment in Fig. 1 in our former study is not necessarily a natural setting for the spatial configuration of heterogeneous agents. One may think that

978-1-4577-0011-8/11/$26.00 ©2011 IEEE

the same type of agents may exist close together. So we compare two configurations with each other. One is random assignment used in our former study. The other is clustered assignment as shown in Fig. 2 (a), (c) and (e) where minority agents are clustered around the center of the two-dimensional grid-world. These clustered configurations are compared with the corresponding random assignment configurations with the same number of minority agents in Fig. 2 (b), (d) and (f), respectively. For example, Fig. 2 (b) includes nine minority agents as in Fig. 2 (a).



(a) Minority agents in the 3x3 cells. (b) Random with nine minority agents.

(c) Minority agents in the 5x5 cells. (d) Random with 25 minority agents.

(e) Minority agents in the 7x7 cells. (f) Random with 49 minority agents.

Fig. 2. Spatial configurations of heterogeneous agents examined in this paper. Each clustered configuration is compared with the corresponding random configuration with the same number of minority agents.

In each of the clustered configurations in Fig. 2 (a), (c) and (e), only agents around the boundary between the two regions have neighbors with different representation schemes. This means that the clustered configurations have less interaction between two types of agents through the IPD game than the corresponding random configurations. Thus it seems that cooperative behavior is more likely to be evolved within the same type of agents in the clustered configurations than the corresponding random configurations. In this paper, we examine the effect of the configuration of heterogeneous agents on the evolution of cooperative behavior.

This paper is organized as follows. In Section II, we explain a spatial IPD game with two neighborhood structures [7]. In Section III, we explain an evolutionary algorithm for strategy evolution. Some representation schemes are also explained in Section III. Experimental results under various settings of neighborhood structures and agent configurations are reported in Section IV. Section V concludes this paper.

II. Our Spatial IPD Game

A. IPD Game

In our spatial IPD game, we assume a frequently-used standard payoff matrix in Table I. Two agents are supposed to simultaneously choose one of the two actions: cooperate and defect. Each agent receives a payoff depending on their actions. When both agents cooperate, each receives three points as shown in Table I as "Agent: 3" and "Opponent: 3". When both agents defect, each receives one point (see the right-bottom cell of Table I). The highest payoff of five points is obtained by defecting when the opponent cooperates. In this case, the opponent receives the lowest payoff of zero point. The simultaneous choice of actions by two agents can be viewed as a single round of the prisoner's dilemma game. In its iterated version (i.e., iterated prisoner's dilemma: IPD), the prisoner's dilemma game is played by the two agents for a pre-specified number of rounds. Usually it is assumed that no agent knows in advance how many rounds of the game will be iterated (i.e., no agent knows when the iteration of the game will be terminated). In our computational experiments, the iteration of the game is terminated after 100 rounds.

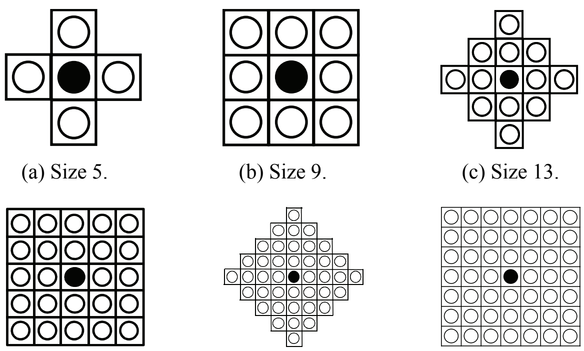
TABLE I
Standard Payoff Matrix of Prisoner's Dilemma Game

Agent's Action	Opponent's Action	
	C: Cooperate	D: Defect
C: Cooperate	Agent: 3 Opponent: 3	Agent: 0 Opponent: 5
D: Defect	Agent: 5 Opponent: 0	Agent: 1 Opponent: 1

B. Spatial IPD Game with Two Neighborhood Structures

Two-dimensional grid-worlds have been usually used in spatial versions of the IPD game [7]-[11]. In this paper, we use an 11x11 grid-world with 121 cells in Fig. 1 and Fig. 2. Each cell has a single agent. Thus the total number of agents is 121. Each agent has its own IPD game strategy. Strategies of the 121 agents are evolved by selection, crossover and mutation as we will explain in Section III and Section IV.

In spatial versions of the IPD game, it is assumed that each agent plays the game against only its neighbors. Neighbors of each agent are specified by a neighborhood structure. Some examples of neighborhood structures are shown in Fig. 3. We denote the set of neighbors of Agent i as $N_{\mathrm{IPD}}(i)$. Agent i itself is included in its neighbors. As in many other studies, we use this setting whereas it is somewhat strange (since each agent can play against itself). Opponents of Agent i for the IPD game are always selected from $N_{\mathrm{IPD}}(i)$. Thus $N_{\mathrm{IPD}}(i)$ can be viewed as the neighborhood structure for local opponent selection. We examine seven specifications of the size of $N_{\mathrm{IPD}}(i)$: 5, 9, 13, 25, 41, 49 and 121. Fig. 3 shows the six specifications of $N_{\mathrm{IPD}}(i)$ of the size 5, 9, 13, 25, 41, and 49. The other specification with the size 121 is the entire 11x11 grid-world. In this case, opponents of each agent are chosen from all the 121 agents in the 11x11 grid-world.



(a) Size 5. (b) Size 9. (c) Size 13.

(d) Size 25. (e) Size 41. (f) Size 49.

Fig. 3. Neighborhood structures used in this paper. Each open circle shows a neighbor of the closed circle agent at the center of each plot. The closed circle agent itself is included in its neighborhood structure as one of its neighbors.

Agent i plays the IPD game against a neighbor in $N_{\mathrm{IPD}}(i)$ for 100 rounds. When $N_{\mathrm{IPD}}(i)$ includes five neighbors, Agent i plays the IPD game with 100 rounds against each of the five neighbors. The average payoff of Agent i per round is calculated over the five executions of the IPD game. The calculated average payoff is used as the fitness of Agent i. However, if $N_{\mathrm{IPD}}(i)$ includes more than five neighbors, randomly selected different five neighbors from $N_{\mathrm{IPD}}(i)$ are used as opponents of Agent i (i.e., random sampling of five neighbors from $N_{\mathrm{IPD}}(i)$ without replacement).This setting of opponent selection is to avoid the increase in the computation time when a neighborhood structure with many neighbors (e.g., 121 neighbors) is used. This setting is also to calculate the average payoff of each agent over the same number of executions of the IPD game even when we use various specifications of the neighborhood structure $N_{\mathrm{IPD}}(i)$.

In some computational experiments, we also assume an additional condition on opponent selection for comparison: Agent i plays the IPD game against its neighbors with the same representation scheme. In this case, $N_{\mathrm{IPD}}(i)$ includes less qualified neighbors as opponents than its size. If the number of qualified neighbors is less than or equal to five, the IPD game is played against all the qualified neighbors. If Agent i has more than five qualified neighbors, it plays the IPD game against randomly selected five qualified neighbors.

As in cellular versions of evolutionary algorithms, a new strategy for each agent is generated from its neighbors by crossover and mutation. In order to separately examine the effect of local opponent selection from the effect of local parent selection on the evolution of cooperative behavior, we use two neighborhood structures [7], [11]: One is for local opponent selection, and the other is for local parent selection. This setting is motivated by the idea of structured demes [12]-[15] with two neighborhood structures. One is for the modeling of daily interaction with others. The other is for the modeling of mating. There exist a number of real-world situations with those two neighborhood structures in nature. For example, neighboring plants fight with each other for water and sunlight in one neighborhood structure, which is much smaller than the other neighborhood structure for mating where they can disperse their pollen.

Let $N_{\mathrm{GA}}(i)$ be the set of neighbors from which parents are selected for generating a new strategy for Agent i. $N_{\mathrm{GA}}(i)$ can be viewed as the neighborhood structure for local parent selection. We use the same seven specifications for $N_{\mathrm{GA}}(i)$ as those for $N_{\mathrm{IPD}}(i)$: the six neighborhood structures in Fig. 3 and the 11x11 grid-world itself. We examine all the 7x7 combinations of those seven specifications for $N_{\mathrm{IPD}}(i)$ and $N_{\mathrm{GA}}(i)$ in our computer experiments. In the case of an inhomogeneous population, $N_{\mathrm{GA}}(i)$ may include neighbors with different representation schemes. In our computational experiments, neighbors with the same representation scheme as Agent i are selected as parents to generate its new strategy.

III. Evolution of IPD Game Strategy

A. Representation Schemes of IPD Game Strategies

An agent's strategy determines its next action based on a finite history of previous plays of the game. In our computational experiments, we use only the previous action of the opponent to determine the agent's action. Binary strings have often been used to represent strategies where "1" and "0" usually mean "cooperate" and "defect", respectively. In Table II, we show a three-bit binary string strategy "101" called TFT (tit-for-tat). An agent with this strategy cooperates at the first round and then cooperates at each round only when the opponent cooperated in the previous round. Each bit value can be also interpreted as the probability of "cooperate". For example, the first bit value "1" of the TFT strategy "101" can be interpreted as the probability with which the TFT agent cooperates in the first round.

Probabilities of "cooperate" in the three situations in Table II can be represented by a real number string of length three such as "0.9 0.1 0.7". This real number string shows the following stochastic strategy: Cooperate with the probability 0.9 in the first round, cooperate with the probability 0.1 in response to the opponent's previous defection, and cooperate with the probability 0.7 in response to the opponent's previous cooperation. All elements of real number strings should be in the closed interval [0, 1]. In our computational experiments, we use three-bit binary strings and real number strings of length three. Real number strings show stochastic strategies while binary strings show deterministic strategies.

TABLE II
THREE-BIT STRING "101" REPRESENTING THE TIT-FOR-TAT STRATEGY

Agent's first action: Cooperate		1
Opponent's previous action	Suggested action	
D: Defect	D: Defect	0
C: Cooperate	C: Cooperate	1

B. Evolutionary Algorithm

As an initial population, we randomly generate a new strategy for each agent. In the case of three-bit binary strings, we randomly choose 0 or 1 with the same probability as each bit value. In the case of real number strings of length three, we randomly generate a real number in the unit interval [0, 1] using the uniform probability distribution. In this manner, we generate an initial population of 121 new strategies (i.e., a new strategy for each agent in the 11x11 grid-world).

The fitness of each agent is calculated as its average payoff per round over multiple executions of the IPD game with 100 rounds against its neighbors. Five neighbors in $N_{\text{IPD}}(i)$ for local opponent selection are chosen randomly to calculate the average payoff of Agent i in many cases in our computational experiments. Only when the number of qualified neighbors in $N_{\text{IPD}}(i)$ is less than or equal to five, all the qualified neighbors in $N_{\text{IPD}}(i)$ are used as opponents.

A new strategy of Agent i is generated by crossover and mutation from two neighbors in $N_{\text{GA}}(i)$ for local parent selection. Binary tournament selection with replacement is applied twice to qualified neighbors in $N_{\text{GA}}(i)$ in order to select a pair of parents to generate a new string for Agent i. The average payoff of each neighbor is used as its fitness in binary tournament selection.

In our former study [7], we used roulette wheel selection with a linear scaling. In this paper, we use binary tournament selection. This is because somewhat strange behavior was observed in [7] when roulette wheel selection was used together with small neighborhood structures for local parent selection. Simplicity is another reason for the use of binary tournament selection. Except for binary tournament selection, we use the same setting as in our former study [7] in our evolutionary algorithm for IPD game strategy evolution.

A new strategy for each agent is generated by applying crossover and mutation to the selected two parents. For binary strings, we use one-point crossover and bit-flip mutation. One-point crossover randomly chooses a crossing point between two bits. Then it exchanges all bit values after the crossing point between the two parents. We apply the one-point crossover operator to each pair of parents with the crossover probability 1.0. Bit-flip mutation changes the bit value from 1 to 0 and from 0 to 1. We apply the bit-flip mutation operator to each bit of binary strings with the mutation probability $1/(5\times 121)$ after crossover.

For real number strings, we use blend crossover (BLX-α [16]) with $\alpha = 0.25$ and the uniform mutation. If a real number becomes more than 1 (or less than 0) by the crossover operator, it is repaired to be 1 (or 0) before mutation. The same crossover probability 1.0 and the mutation probability $1/(5\times 121)$ are used for binary and real number strategies.

The current strategy of each agent is always replaced with a newly generated strategy for that agent. That is, the current population of 121 strategies is entirely replaced with newly generated 121 strategies. The fitness evaluation through the IPD game and the generation update by genetic operations are iterated for a pre-specified number of generations. In our computational experiments, the evolution of game strategies is performed for 1000 generations.

IV. EXPERIMENTAL RESULTS

All of our computational experiments in this paper were performed using the following parameter specifications:

Parameters in the IPD Game:
Grid-world: 11x11 grid-world with the torus structure,
Number of agents: 121,
Neighborhood structure for local opponent selection:
Seven structures (Fig. 3 and the grid-world itself),
Number of opponents: Maximum five neighbors,
Number of rounds: 100 rounds.

Parameters in the Evolutionary Algorithm:
Population size: 121 in the 11x11 grid-world,
Representation schemes: Three-bit binary strings, and
real number strings of length 3,
Initial population: Randomly generated strings,
Neighborhood structure for local parent selection:
Seven structures (Fig. 3 and the grid-world itself),
Parent selection:
Binary tournament selection with replacement,
Crossover: One-point crossover for binary strings, and
BLX-α with $\alpha = 0.25$ for real number strings,
Mutation: Bit-flip mutation for binary strings, and
uniform mutation for real number strings,
Crossover probability: 1.0,
Mutation probability: $1/(5\times 121)$,
Constraint handling for real number strings:
Repair to 1(if larger than 1) or 0 (if smaller than 0),
Termination condition: 1000 generations.

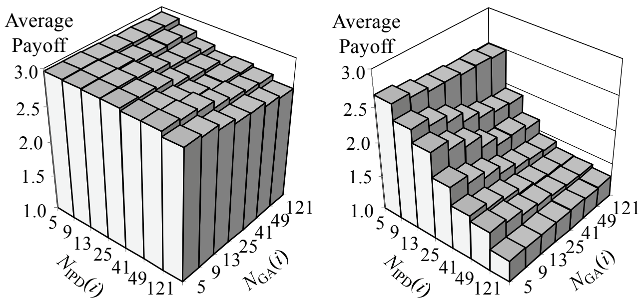
All results reported in this paper are average results over 500 runs for each setting in our computational experiments.

A. Results with Homogeneous Population

In this subsection, we report experimental results from a homogeneous population. First we assigned three-bit binary strings to all the 121 agents in the 11x11 grid-world. Under this setting, we calculated the average payoff of all agents over 500 runs through 1000 generations for each of the 7x7 = 49 combinations of the two neighborhood structures: $N_{\text{IPD}}(i)$ for local opponent selection in the IPD game and $N_{\text{GA}}(i)$ for local parent selection in the evolution algorithm for IPD game strategies. We show experimental results in Fig. 4 (a). High average payoff close to 3.0 was obtained in all combinations of the two neighborhood structures.

The same computational experiments were also performed using real number strings of length three. Experimental results were summarized in Fig. 4 (b). High average payoff close to 3.0 was obtained only when we used the smallest neighborhood structure with five neighbors for local opponent selection (i.e., $|N_{\mathrm{IPD}}(i)| = 5$). There exist clear differences between Fig. 4 (a) and Fig. 4 (b). This observation supports that the choice of a representation scheme has a large effect on the evolution of cooperative behavior.

It should be noted that the average payoff 3.0 is obtained only when all agents always cooperate. The increase in the number of “defect” actions always leads to the decrease in the average payoff over all agents (whereas it may increase the average payoff over a small number of agents who defect). Thus we can use the average payoff over all agents as an index to measure the evolution of cooperative behavior.



(a) Three-bit binary strings. (b) Real number strings of length three.

Fig. 4. Average payoff obtained from a homogeneous population of 121 agents with the same representation scheme. All the possible 49 combinations in the size of the two neighborhood structures were examined.

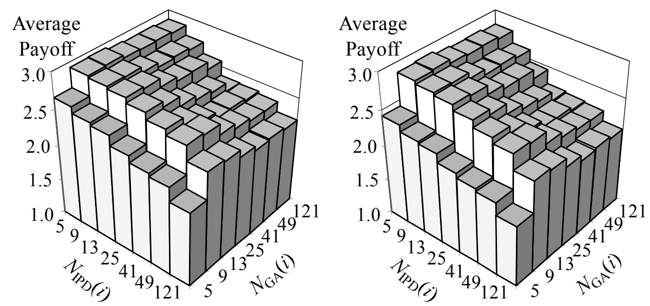
B. Results with Random Assignment

As in our former study [7], we randomly assigned one of the two representation schemes to each agent with the same probability. Fig. 1 (a) is an example of a random assignment configuration of the two representation schemes. Using such a random assignment configuration, we compared two cases with each other. In one case, the IPD game was performed independent of the representation scheme of each agent. That is, the IPD game was performed between heterogeneous agents with different representation schemes as well as between homogeneous ones. In the other case, the IPD game was performed only between homogeneous agents with the same representation scheme. That is, the IPD game was not played between a binary string agent and a real number string agent in the latter case.

As in the previous subsection, we examined each of the 49 combinations of the two neighborhood structures for each of the two cases (i.e., with and without the interaction through the IPD game between the two representation schemes). In our computational experiments in this subsection, the average payoff was calculated for agents with each representation scheme over 500 runs. In each run, we used a different random assignment configuration. Fig. 5 shows experimental results in the case with the interaction through the IPD game between the two representation schemes. As shown in our former study [7], similar results were obtained by binary string agents in Fig. 5 (a) and real number string agents in Fig. 5 (b) when the IPD game was played between the two representation schemes.

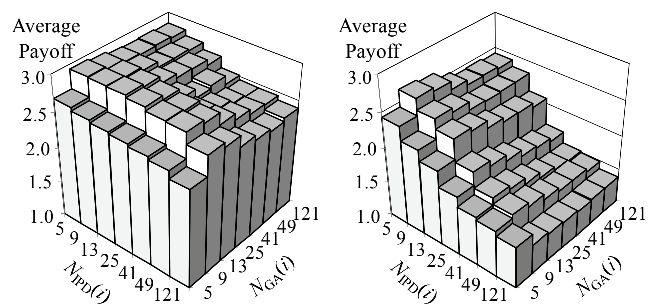
When the IPD game was not played between the two representation schemes, totally different experimental results were obtained from each representation scheme as shown in Fig. 6. In Fig. 6, there was no interaction between the two representation schemes. Thus Fig. 6 is similar to Fig. 4 with a homogeneous population (especially Fig. 6 (b) is similar to Fig. 4 (b)).

From Fig. 5 and Fig. 6, we can see that the average payoff was decreased by the use of the smallest neighborhood structure with five neighbors for local parent selection (i.e., $|N_{\mathrm{GA}}(i)| = 5$). This observation will be discussed in the next subsection.



(a) Three-bit binary strings. (b) Real number strings of length three.

Fig. 5. Average payoff obtained from an inhomogeneous population with 50% agents of each representation scheme. The IPD game was played between the two representation schemes.



(a) Three-bit binary strings. (b) Real number strings of length three.

Fig. 6. Average payoff obtained from an inhomogeneous population with 50% agents of each representation scheme. The IPD game was not played between the two representation schemes.

C. Results with 49 Minority Agents

In this subsection, we report experimental results from the spatial configuration in Fig. 2 (e) with clustered 49 minority agents in the 7x7 cells at the center of the 11x11 grid-world. In our computational experiments, we always assigned three-bit binary string agents to the 7x7 cells at the center of the 11x11 grid-world. Real number string agents were always assigned to all the remaining cells. For comparison, we also examine random assignment configurations with 49 minority agents. Fig. 2 (f) is an example of such a random assignment configuration. Three-bit binary string agents were always used as minority agents.

Before reporting experimental results, let us briefly discuss the difference between clustered configuration and random configuration. In Fig. 7, we illustrate the relation between the spatial configuration of 49 minority agents and the smallest neighborhood structure with five neighbors.

When we use the smallest neighborhood structure with five neighbors for the clustered configuration in Fig. 7 (a), many agents have only homogeneous neighbors with the same representation scheme. Some minority agents (e.g., agent A in Fig. 7 (a)) have a single majority agent as a heterogeneous neighbor. Some majority agents (e.g., agent C) have a single minority agent as a heterogeneous neighbor. Only four agents (e.g., agent B) have two heterogeneous neighbors in Fig. 7 (a). The number of heterogeneous neighbors of each agent in Fig. 7 (a) can be calculated as follows:

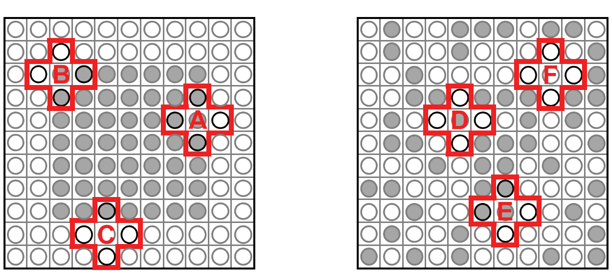
Among the 49 minority agents (denoted by closed circles):
- 4 agents have two heterogeneous neighbors,
- 20 agents have a single heterogeneous neighbor,
- 25 agents have no heterogeneous neighbors.

Among the 72 majority agents (denoted by open circles):
- 28 agents have a single heterogeneous neighbor,
- 44 agents have no heterogeneous neighbors.

Among all the 121 agents in Fig. 7 (a):
- 4 agents have two heterogeneous neighbors,
- 48 agents have a single heterogeneous neighbor,
- 69 agents have no heterogeneous neighbors.

That is, 57% agents have no heterogeneous neighbors in Fig. 7 (a). Only 3% have two heterogeneous neighbors. On the other hand, in Fig. 7 (b), more agents (e.g., agent E) have multiple heterogeneous neighbors. Some agents (e.g., agent D) have four heterogeneous agents while others (e.g., agent F) have no heterogeneous neighbors. In Fig. 7 (b), the number of heterogeneous neighbors of each agent is as follows:

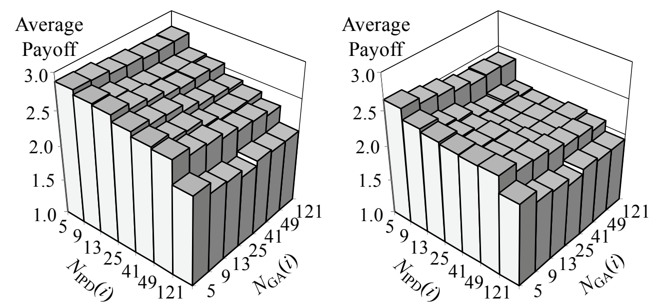
- 9 agents have four heterogeneous neighbors,
- 30 agents have three heterogeneous neighbors,
- 47 agents have two heterogeneous neighbors,
- 32 agents have a single heterogeneous neighbor,
- 3 agents have no heterogeneous neighbors.



(a) Minority agents in the 7x7 cells. (b) Random with 49 minority agents.

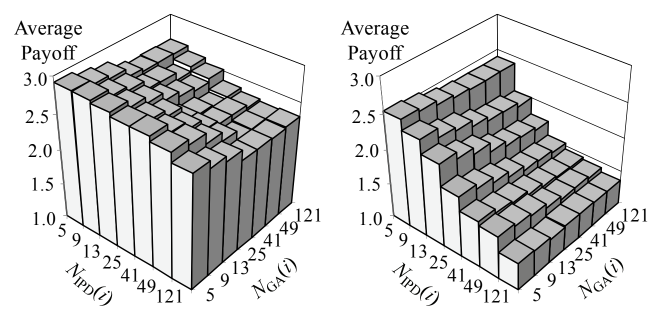
Fig. 7. Relation between the spatial configuration of 49 minority agents and the smallest neighborhood structure with five neighbors.

In Fig. 8 and Fig. 9, we show experimental results using the clustered configuration of 49 binary string agents (i.e., Fig. 7 (a)). The IPD game was played in Fig. 8 between the two representation schemes while there was no interaction in Fig. 9 between the two representation schemes. For comparison, we show experimental results from random assignment of 49 binary string agents in Fig. 10. Different random assignment configurations (e.g., Fig. 7 (b)) were used in each run in Fig. 10 while the same clustered configuration in Fig. 7 (a) was used in all runs in Fig. 8 and Fig. 9. In Fig. 10, we can observe a clear similarity between Fig. 10 (a) and Fig. 10 (b) obtained from random assignment configurations with the interaction between the two representation schemes as in Fig. 5.



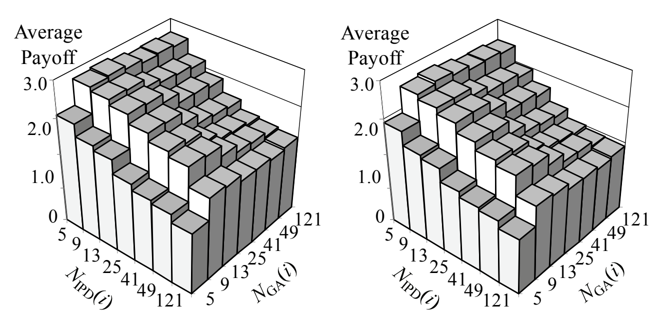
(a) Three-bit binary strings. (b) Real number strings of length three.

Fig. 8. Average payoff obtained from an inhomogeneous population with 49 clustered agents with three-bit binary strings in the 7x7 cells. The IPD game was played between the two representation schemes.



(a) Three-bit binary strings. (b) Real number strings of length three.

Fig. 9. Average payoff obtained from an inhomogeneous population with 49 clustered agents with three-bit binary strings in the 7x7 cells. The IPD game was not played between the two representation schemes.



(a) Three-bit binary strings. (b) Real number strings of length three.

Fig. 10. Average payoff obtained from an inhomogeneous population with randomly placed 49 agents with three-bit binary strings. The IPD game was played between the two representation schemes.

In Fig. 10, we can also see that the average payoff was decreased by the use of the smallest neighborhood structure with five neighbors for local parent selection (i.e., $|N_{\mathrm{GA}}(i)| = 5$) as in Fig. 5 and Fig. 6. This is because some agents had four heterogeneous neighbors as explained in Fig. 7 (b). In this case, each of those agents had only a single qualified neighbor as its parent (i.e., that agent itself). This mean that the same agent was always selected as its parent since no other qualified neighbors were included in $N_{\mathrm{IPD}}(i)$. Thus no selection pressure was applied to those agents with only a single homogeneous neighbor in $N_{\mathrm{GA}}(i)$. This leads to low average payoff when $|N_{\mathrm{IPD}}(i)| = 5$ in Fig. 5, Fig. 6 and Fig. 10.

In Fig. 8 with the clustered configuration of 49 binary string agents, we can observe some similarity between Fig. 8 (a) and Fig. 8 (b). Since the two plots are totally different from each other in Fig. 9 with no interaction between the two representation schemes through the IPD game, we can conclude that the interaction through the IPD game led to somewhat similar evolution of cooperative behavior among agents with each representation scheme. However, the two plots in Fig. 8 are not so similar if compared with the case of random assignment configurations in Fig. 10. This is because the interaction between the two representation schemes was limited in Fig. 8 as explained using Fig. 7 (a).

In order to further examine the evolution of cooperative behavior in the clustered configuration of 49 binary string agents in Fig. 7 (a), we calculated the average payoff of each agent at each generation over 500 runs. Experimental results are summarized in Fig. 11 for the case of $|N_{\mathrm{IPD}}(i)| = 5$ and $|N_{\mathrm{GA}}(i)| = 5$ with the interaction through the IPD game between the two representation schemes. We chose the combination of the two neighborhood structures from which the highest average payoff was obtained in each of the two plots in Fig. 8. That is, Fig. 11 corresponds to the highest bar in each plot of Fig. 8 at $|N_{\mathrm{IPD}}(i)| = 5$ and $|N_{\mathrm{GA}}(i)| = 5$.

The basal plane of each plot in Fig. 11 corresponds to the 11x11 grid-world. Each bar in Fig. 11 shows the average payoff of the agent at each cell of the grid-world. Dark gray bars show binary string agents while light gray bars show real number string agents. In Fig. 11, binary string agents (i.e., dark gray bars) had lower average payoff around the boundary of the 7x7 region than the center of the grid-world due the interaction with real number string agents. This negative effect became smaller in later generations.

For comparison, in Fig. 12, we show experimental results with no interaction between the two representation schemes through the IPD game. Except for the interaction through the IPD game between the two representation schemes, we used the same setting between Fig. 11 and Fig. 12. From the comparison between Fig. 11 and Fig. 12, we can see that the interaction with binary string agents helped real number string agents (i.e., light gray bars) to evolve cooperative behavior faster in Fig. 11 than Fig. 12. In Fig. 12, agents with one representation scheme evolved cooperative behavior totally independent from agents with the other scheme.

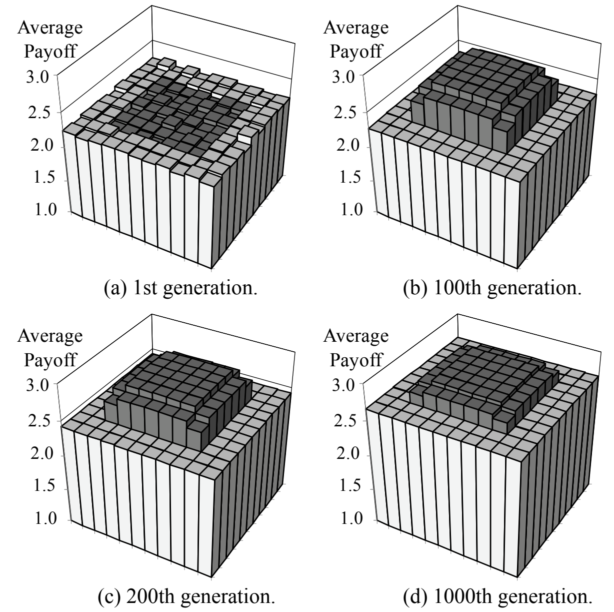


Fig. 11. Average payoff of each agent at each generation over 500 runs. The IPD game was played between the two representation schemes in the clustered configuration of 49 minority binary string agents in Fig. 7 (a).

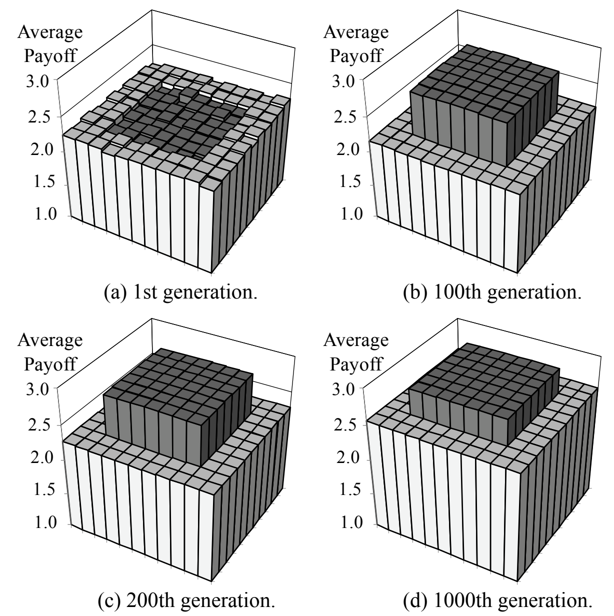
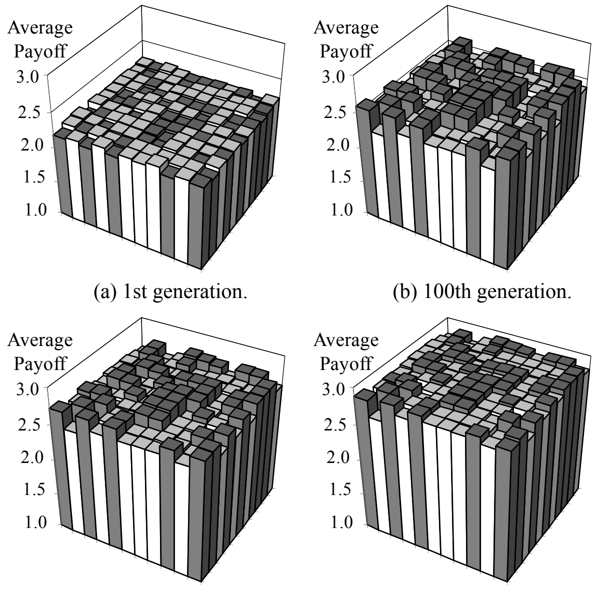


Fig. 12. Average payoff of each agent at each generation over 500 runs. The IPD game was not played between the two representation schemes in the clustered configuration of 49 minority binary string agents in Fig. 7 (a).

In Fig. 13, we show experimental results using the random assignment configuration in Fig. 7 (b) with the interaction between the two representation schemes through the IPD game. The two neighborhood structures were specified as $|N_{\mathrm{IPD}}(i)| = 5$ and $|N_{\mathrm{GA}}(i)| = 9$ in Fig. 13 since high average payoff was not obtained from $|N_{\mathrm{GA}}(i)| = 5$ in Fig. 10. In Fig. 13, we used the same random assignment (i.e., Fig. 7 (b)) for all the 500 runs while a different random assignment was used in each run in Fig. 10. Thanks to heavy interaction between heterogeneous agents, almost all agents had similar average payoff at the 1000th generation in Fig. 13.



(a) 1st generation. (b) 100th generation.

(c) 200th generation. (d) 1000th generation.

Fig. 13. Average payoff of each agent at each generation over 500 runs. The IPD game was played between the two representation schemes in the random configuration of 49 minority binary string agents in Fig. 7 (b).

D. *Results with 9 and 25 Minority Agents*

Due to the page limitation, we cannot report experimental results with 9 and 25 minority binary string agents in detail. Similar observations were obtained from the three settings with respect to the number of binary string agents (i.e., 9, 25 and 49) whereas there were large differences in the average payoff among the three settings. The obtained observations are summarized as follows:

1. Different results were obtained from each representation scheme when the IPD game was not played between the two representation schemes.
2. Similar results were obtained from each representation scheme when the IPD game was played between the two representation schemes. Higher similarity was observed in random configurations than clustered configurations.

V. CONCLUSIONS

In this paper, we examined the effect of spatial configurations of heterogeneous agents on the evolution of cooperative behavior in a spatial IPD game with an inhomogeneous population. In our spatial IPD game, each agent has a different representation scheme. In this sense, agents are heterogeneous, and a population of heterogeneous agents is inhomogeneous. In computational experiments with two representation schemes, we compared two spatial configurations: clustered configuration of minority agents and their random assignment configuration. Experimental results showed that similar results were obtained from each representation scheme when the IPD game was played between the two representation schemes. More similar results were obtained from each representation scheme by random assignment configuration than clustered one. This is because heterogeneous agents with different representation schemes are more likely to interact with each other in the case of their random assignment than their clustered assignment. One may think that the evolution of cooperative behavior is easier among clustered homogeneous agents than the case of their random assignment. However, we observed no clear positive effects of clustered homogeneous agents on the evolution of cooperative behavior in comparison with their random assignment case except when the smallest neighborhood structure with five neighbors was used for parent selection.

We examined only one clustered configuration of 49 minority binary string agents in detail in this paper. More computational experiments with different settings may be needed for analyzing the evolution of cooperative behavior in an inhomogeneous population with heterogeneous agents. Simultaneous use of more than two representation schemes in a large population is an interesting topic for future research.

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, 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. 67-81, March 2011.
[8] M. A. Nowak, R. M. May, and K. Sigmund, "The arithmetics of mutual help," *Scientific American*, vol. 272, no. 6, pp. 50-53, June 1995.
[9] A. L. Lloyd, "Computing bouts of the Prisoner's Dilemma," *Scientific American*, vol. 272, no. 6, pp. 80-83, June 1995.
[10] K. Brauchli, T. Killingback, and M. Doebeli, "Evolution of cooperation in spatially structured populations," *Journal of Theoretical Biology*, vol. 200, no. 4, pp. 405-417, October 1999.
[11] 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, December 2005.
[12] D. S. Wilson, "Structured demes and the evolution of group-advantageous traits," *The American Naturalist*, vol. 111, no. 977, pp. 157-185, January-February 1977.
[13] D. S. Wilson, "Structured demes and trait-group variation," *The American Naturalist*, vol. 113, no. 4, pp. 606-610, April 1979.
[14] M. Slatkin and D. S. Wilson, "Coevolution in structured demes," *Proc. of the National Academy of Sciences*, vol. 76, no. 4, pp. 2084-2087, April 1979.
[15] B. Charlesworth, "A note on the evolution of altruism in structured demes," *The American Naturalist*, vol. 113, no. 4, pp. 601-605, April 1979.
[16] L. J. Eshelman and J. D. Schaffer, "Real-coded genetic algorithms and interval-schemata," in *Foundations of Genetic Algorithms 2*. San Mateo, CA: Morgan Kaufman, pp. 187-202, 1993.