Mathematical Biosciences and Engineering

http://www.aimspress.com/journal/mbe

MBE, 20(9): 17116–17137.
DOI: 10.3934/mbe.2023763
Received: 11 May 2023
Revised: 16 August 2023
Accepted: 21 August 2023
Published: 04 September 2023

Research article

Nash equilibrium realization of population games based on social learning processes

Zhiyan Xing, Yanlong Yang* and Zuopeng Hu

School of Mathematics and Statistics, Guizhou University, Guiyang 550025, China

* **Correspondence:** Email: yylong1980@163.com.

Abstract: In the two-population game model, we assume the players have certain imitative learning abilities. To simulate the learning process of the game players, we propose a new swarm intelligence algorithm by combining the particle swarm optimization algorithm, where each player can be considered a particle. We conduct simulations for three typical games: the prisoner's dilemma game (with only one pure-strategy Nash equilibrium), the coin-flip game (with only one fully-mixed Nash equilibrium), and the coordination game (with two pure-strategy Nash equilibria and one fully-mixed Nash equilibrium). The results show that when the game has a pure strategy Nash equilibrium, the algorithm converges to that equilibrium. However, if the game does not have a pure strategy Nash equilibrium, it exhibits periodic convergence to the only mixed-strategy Nash equilibrium. Furthermore, the magnitude of the periodical convergence is inversely proportional to the introspection rate. After conducting experiments, our algorithm outperforms the Meta Equilibrium Q-learning algorithm in realizing mixed-strategy Nash equilibrium.

Keywords: population game; social learning; imitation learning; Nash equilibrium; stable limit ring

1. Introduction

Game theory is an important branch of operations research, in which population games are a classical game model. Because population games not only can reveal the essential features of collaboration and competition but also can provide profound insights and revelations among populations, it has been widely used in social sciences, biology, economics, and other fields. Therefore, population games have been a hot topic of academic research. In the game theory, Nash equilibrium [1,2] is an important concept. However, the traditional Nash equilibrium requires that players are perfectly rational and have complete information. Fudenberg and Levine [3] propose an alternative interpretation of equilibrium, "The equilibrium is the long-term outcome of the process by which imperfectly rational players seek to optimize over time". Influenced by Fudenberg's interpretation of equilibrium, we consider how to find

the path to Nash equilibrium under conditions of imperfect rationality and incomplete information. In reality, players aim to maximize their benefits, and equilibrium emerges after repeated games. Nash equilibrium is an integral component of this equilibrium, and due to its challenging establishment, investigating the process of Nash equilibrium formation holds intrinsic value. These players are not smart enough, and their ability is limited. To depict the strategic interactions among these players, we develop an algorithm to simulate their gaming processes. Among these algorithms, the particle swarm optimization (PSO) algorithm [4, 5] is based on the feeding behavior of a bird flock. Both the PSO algorithm and the realization of Nash equilibrium are based on the concept of optimization, albeit with distinct approaches. The PSO algorithm emphasizes collective optimization, whereas Nash equilibrium realization centers around individual optimization. Therefore, we can glean insights from the PSO algorithm to develop an algorithm suitable for achieving Nash equilibrium. The algorithmic realization of Nash equilibrium is rooted in the decisions made by imperfectly rational players, developing algorithms to simulate equilibrium evolution. However, limited research exists regarding achieving Nash equilibrium in population games using swarm intelligence algorithms.

In the field of game theory, Nash equilibrium theory holds significant importance. Learning rules provide a perspective for studying Nash equilibrium from the players' viewpoint. Currently, three primary types of learning models exist. The first type is the virtual action learning theory [6–13], which was first proposed by Fudenberg and Levine [3]. It is believed that the opponent's strategy remains uncertain in each game, requiring the anticipation of the opponent's moves. The theory of virtual action learning considers the opponent's prior strategy choices, assigning weight to these choices and using the weighted outcome to determine the opponent's subsequent strategy. The second type is the social learning model [14–18] for population games, which is also proposed by Fudenberg and Levine [19]. Within this model, players can glean information about fellow players who achieve superior benefits within the population. This collective learning process eventually converges the system to a stable state. The third type is the reinforcement learning model [20–24], Littman [25] proposed a two-player model in zero-sum games. The model assumes that players can retain the memory of strategies and their associated benefits from previous games. Through continuous reflective learning, players strive to achieve Nash equilibrium. In addition, Borgers and Sarin [26] proposed the stimulus-response learning model based on the reinforcement learning model. In this model, players can solely recall their past strategy selections and the associated benefits. Consequently, they are inclined to employ their previous actions to guide future strategic decisions. The model posits that well-performing actions are positively reinforced, while poorly performing actions are negatively reinforced. Jordan [27] proposed the Bayesian learning model, and Camerer and Hua [28] proposed the experience-adding weight affinity (EWA) model. These two models are also important learning models based on virtual, social, and reinforcement learning models.

Previous research on realizing Nash equilibrium has been mainly based on reinforcement learning theory. In 2000, Singh et al. [29] first proposed the infinitesimal gradient algorithm (IGA), which enables each player to adjust its strategy based on the gradient of its expected benefit. This algorithm converges to a particular Nash equilibrium. After that, Zinkevich [30] proposed the generalized infinitesimal gradient algorithm (GIGA), which extends the applicability of the IGA algorithm from just two strategies to encompass multi-strategy scenarios. Wang et al. [31] expanded their investigation to meta-games, achieving a path towards meta-equilibrium using Q-learning. However, the existing realizations of Nash equilibrium mainly focus on inter-player games, and further exploration and re-

finement are necessary for the realization of Nash equilibrium in population games.

This paper develops the population game particle swarm optimization (PGPSO) algorithm, which uses social learning and population imitation as theoretical sources. We theoretically prove the convergence of the PGPSO algorithm, and the Nash equilibrium of a single mixed strategy is proved to be the center of a stable limit ring in the algorithm. Using the PGPSO algorithm, we simulate the evolution of three two-population games and search their Nash equilibriums' realization paths. The experimental outcomes validate the efficacy of the PGPSO algorithm in uncovering Nash equilibria. Additionally, the effect of introspection rate and initial state on the PGPSO algorithm to realize Nash equilibrium is further explored.

2. Preliminaries

This section will mainly introduce the fundamental concepts of population games, social learning theory, and population imitation theory.

2.1. Concepts related to two-population game

The two-population game is denoted by $\{\Gamma, X, F\}$ [32].

1) $\Gamma = \{1, 2\}$ denotes the two populations. For each population, $p \in \Gamma$, $S^p = \{1, 2\}$ denotes the set of pure strategies available to population p.

2) For population 1, x_1 denotes the proportion of players who choose strategy 1; for population 2, y_1 denotes the proportion of players who choose strategy 1. Further, $x = (x_1, x_2)$ denotes the pure strategy distribution state of population 1, $y = (y_1, y_2)$ denotes the pure strategy distribution state of population 2. The strategy choice of the i-th player in population 1 is denoted as x^i, and likewise, the strategy choice of the j-th player in population 2 is denoted as y^j. Given that players can only select pure strategies, $x^i, y^j \in \{0, 1\}$. If x^i (y^j) = 1, it means that the i-th (j-th) player has chosen strategy 1; If x^i (y^j) = 0, it means that the i-th (j-th) player has chosen strategy 2. The combined set $X = (x, y)$ represents the social state of the two populations Γ.

3) For any given population p, $F_s^p : X \to R$ denotes the expected benefit associated with pure strategy s, $s \in S^p$. Therefore, the corresponding set of pure strategies for population p is denoted as S^p, and $F^p : X \to R^2$ represents the expected benefit of population p. The overall expected benefit function of the entire society Γ is denoted as $F = (F^1; F^2) : X \to R^4$.

The following definition is the Nash equilibrium definition of the two populations game $\{\Gamma, X, F\}$ [32].

Definition 1. Let $\{\Gamma, X, F\}$ be a two-population game. If the social state $\bar{z} = (\bar{x}, \bar{y}) \in X$ satisfies $\forall p \in \Gamma$, $\bar{x}_s > 0$, $\bar{y}_s > 0 \Rightarrow F_s^p(\bar{z}) = \max_{r \in S^p} F_r^p(\bar{z})$, $\forall s \in S^p$, then we define $\bar{z} = (\bar{x}, \bar{y})$ as the Nash equilibrium of the population game $\{\Gamma, X, F\}$, and denote the set containing all Nash equilibria as $E(F)$.

According to the above definition of Nash equilibrium for the two-population game, the Nash equilibrium of the prisoner's dilemma game is $(\bar{x}, \bar{y}) = ((1, 0), (1, 0))$. The Nash equilibrium of the coin-flip game is $(\bar{x}, \bar{y}) = ((\frac{1}{2}, \frac{1}{2}), (\frac{1}{2}, \frac{1}{2}))$. The Nash equilibrium set of the coordination game is $E(F) = \{((1, 0), (1, 0)), ((0, 1), (0, 1)), ((\frac{1}{3}, \frac{2}{3}), (\frac{1}{3}, \frac{2}{3}))\}$. The benefit matrices for the three games are presented in Examples 4.1–4.3 later on.

2.2. Social learning theory

Fudenberg and Levine [3] proposed the social learning theory to explain the formation of the Nash equilibrium of the population game. In a single iteration, the initial population is called the "parent", denoted as $q(t)$. The population that completes strategy adjustment is called the "offspring", denoted as $q(t+1)$. There is an excessive generation from the parent to the offspring, called the pending generation, denoted as $q(t')$. It is crucial that the overall strategy distribution of the pending generation is the same as that of the parent, i.e., $x_s(t') = x_s(t)$, with pending players corresponding one-to-one with their parent players.

During each iteration, for one of the populations p, a proportion α of pending players chooses to adjust their strategies, while the remaining pending players will keep their original strategy. Björnerstedt and Weibull [15] interpret the phenomenon as an introspective phenomenon, where certain players in the population actively imitate and learn from others. Players who adjust their strategies are called "introspective players", whereas those who keep their original strategies are called "non-introspective players". In the context of a game, following the principle of random matching, all players choose the pure strategy. This process can be illustrated using the game model presented in [3] as an example.

$$\begin{array}{c} \\ U \\ D \end{array} \begin{array}{c} \begin{array}{cc} L & R \end{array} \\ \left[\begin{array}{cc} 9,0 & 0,0 \\ 2,0 & 2,0 \end{array}\right] \end{array}$$

The model is based on a game framework featuring a virtual population 2, as proposed by Fudenberg and Levine [3]. The social learning theory for strategy updating is as follows: $x_1(t)$ denotes the proportion of parents in population 1 who choose strategy U, $x_2(t)$ denotes the proportion of parents in population 1 who choose strategy D, $y_1(t)$ denotes the proportion of parents in population 2 who choose strategy L, and $y_2(t)$ denotes the proportion of parents who choose strategy R. For population 1, the proportion of direct choice strategy U without introspection is $(1-\alpha)x_1(t)$. According to the social learning theory, a player's strategy remains unchanged if their strategy is consistent with their parent's, and the proportion is $\alpha x_1(t)^2$.

When a player's strategy does not match their parent's strategy, in that case, it is divided into two small populations according to the encountered opponents, and the player imitates the strategy of the small population with the highest expected benefit. For example, when a player encounters an opponent who chooses strategy L, he will choose strategy U, and the proportion is $2\alpha y_1(t)x_1(t)x_2(t)$. Similarly, if he encounters an opponent who chooses strategy R, he will choose strategy D, and the proportion is $2\alpha y_2(t)x_1(t)x_2(t)$.

With the above variation of strategies, the proportional update formula for the offspring selection strategy U of population 1 can be obtained as:

$$x_1(t+1) = (1-\alpha)x_1(t) + \alpha(x_1(t)^2 + 2y_1(t)x_1(t)x_2(t)). \tag{2.1}$$

2.3. Population imitation theory

For the population game model, $\{\Gamma, X, F\}$, Schlag [14] presents an alternative perspective, asserting that each player can observe the strategies and expected benefits of others within their population. This perspective eliminates the notion of parent and offspring from social learning theory. Schlag argues that

the emergence of Nash equilibrium in population games hinges solely on the phenomenon of imitation. The rules of imitation, as defined by Schlag, are as follows:

1) Following imitative behavior, i.e., change behavior exclusively by imitating others.

2) Never imitate someone who performs worse than you do.

Rule 2) means that the players will only imitate those who exhibit superior expected benefits. This implies that players evaluate their strategies based on the expected benefit of other players. And players choose to imitate other players with superior expected benefits to improve their own benefit. The essence of the imitation rule is that players adjust their strategies based on the strategies and benefits of other players.

3. Design of the PGPSO algorithm

3.1. The idea of the PGPSO algorithm

In 1995, inspired by the regularity of birds' flock feeding behavior, Kennedy and Eberhart [4, 5] developed a simplified algorithm model, which later evolved into the particle swarm optimization (PSO) algorithm through subsequent enhancements. The idea of the PSO algorithm originated from studying the birds' flock feeding behavior, where the birds share information collectively so that the flock can find the optimal destination. In the PSO algorithm, the feasible solution of each optimization problem can be considered as a point in the d-dimensional search space. Let the position of the i-th particle be denoted as $l_i = (l_{i1}, l_{i2}, ..., l_{id})$, and the best position it has experienced is denoted as $p_i = (p_{i1}, p_{i2}, ..., p_{id})$, and also known as p_{best}. The index number of the best position experienced by all particles is denoted by the symbol g_{best}. The velocity of the i-th particle is denoted as $v_i = (v_{i1}, v_{i2}, ..., v_{id})$. For each iteration, the velocity and position of the particle change according to the following:

$$v_i^{k+1} = wv_i^k + c_1r_1(p_{best}^k(i) - l_i^k) + c_2r_2(g_{best}^k - l_i^k), \tag{3.1}$$

$$l_i^{k+1} = l_i^k + v_i^{k+1}, \tag{3.2}$$

where c_1, c_2 are learning factors; r_1, r_2 are random numbers varying within (0, 1); w are inertia weights; k is the number of iterations.

The PGPSO algorithm builds upon the framework of the PSO algorithm to realize Nash equilibrium and find the realization path, which introduces social learning and population imitation theory into the PSO algorithm. In the PGPSO algorithm, each Nash equilibrium represents a solution to the problem, and each player in the population is considered a particle of the PSO algorithm. Players with different strategies reflect the differences in position among particles, and different benefits reflect the differences in expected benefits among particles, both of which constitute particle diversity.

According to population imitation theory, in the population updating process of particles (players), particles (players) with high expected benefits are learned. However, if all players with low benefits change their strategies by imitating in the first iteration, the algorithm stops directly, resulting in Nash equilibrium losing the opportunity to be learned. Therefore, this paper takes the introspection rate to the PSO algorithm. This serves two purposes: firstly, only some particles (players) choose to introspect, effectively maintaining the diversity of particles (players) in each iteration. Secondly, the introspection rate fits the lag of strategy update of the players in the actual game.

During each iteration, the player chooses a pure strategy, i.e., the position of the particles. Given the nature of the two-population game, the PGPSO algorithm accommodates two distinct particle populations. The benefit matrices for a two-population game are defined as follows:

$$A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}, B = \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix}.$$

At the k-th iteration, the expected benefit of the i-th player in population 1 is calculated as per references [33, 34].

$$\begin{aligned} F_s^1(k) &= (x^i, 1 - x^i)A(y_1, 1 - y_1)^{\mathrm{T}} \\ &= (a_{11} - a_{12} - a_{21} + a_{22})x^i y_1 + (a_{12} - a_{22})x^i + (a_{21} - a_{22})y_1 + a_{22}, \end{aligned} \tag{3.3}$$

where x^i denotes the i-th player strategy choice in population 1, $x^i \in \{0, 1\}$. If $x^i = 1$, then the player has chosen the strategy 1; if $x^i = 0$, then the player has chosen the strategy 2. y_1 indicates the proportion of players in population 2 who choose strategy 1.

The expected benefit of the j-th player in population 2 is

$$\begin{aligned} F_s^2(k) &= (x_1, 1 - x_1)B(y^j, 1 - y^j)^T \\ &= (b_{11} - b_{12} - b_{21} + b_{22})x_1 y^j + (b_{12} - b_{22})x_1 + (b_{21} - b_{22})y^j + b_{22}, \end{aligned} \tag{3.4}$$

where y^i denotes the i-th player strategy choice in population 2, $y^i \in \{0, 1\}$. If $y^i = 1$, then the player has chosen the strategy 1; if $y^i = 0$, then the player has chosen the strategy 2. x_1 denotes the proportion of players in population 1 who choose strategy 1.

The Nash equilibrium of the population game represents a state where all players maximize their benefits. The players are selfish and aim to maximize their benefits, so the benefits defined in the Eqs (3.3) and (3.4) are obtained based on that consideration. Since the benefits of players in the population who choose the same strategy are indistinguishable, it is assumed that determining the benefits corresponding to each strategy becomes an optimization problem. In this case, solving the Nash equilibrium is equivalent to finding the optimal solution of this optimization problem. The key difference is that an ordinary optimization problem is a single-player optimization problem. At the same time, the game studies a multi-player optimization problem, which is the essential distinction between the two.

In the iterative process of particles, population imitation theory guides introspective players to adopt the strategy associated with the highest expected benefit. In the k-th iteration of a population, the particle with the highest expected benefit is determined by the Eqs (3.3) and (3.4), denoted as i_{best}^k, and the introspective particle changes its strategy to i_{best}^k. In contrast, the non-introspective particle keeps its strategy unchanged. At the k-th iteration, suppose the set of all particles' ordinal numbers is I^k, and the set of introspective players' ordinal numbers is denoted as I_α^k.

Take the Eqs (3.1) and (3.2) as the basis, the PGPSO algorithm iteration function is

$$v_i^{k+1} = i_{best}^k - l_i^k, \tag{3.5}$$

$$l_i^{k+1} = \begin{cases} l_i^k + v_i^{k+1}, i \in I_\alpha^k \\ l_i^k, i \notin I_\alpha^k \end{cases}. \tag{3.6}$$

This is the iterative formulation of the PGPSO algorithm, which draws on the formula form of the Eqs (3.1) and (3.2). However, the idea is derived from the theory of population imitation and social learning, where players with relatively lower benefits adopt strategies observed from the players with the highest benefit.

3.2. Algorithm construction

The implementing steps of the PGPSO algorithm are as follows.

(a) The PGPSO algorithm initializes the parameter values. These include the introspection rates α, β, the lower bound of the search space *popmin*, the upper bound of the search space *popmax*, the population size m, n, and the number of iterations *genmax*.

(b) Two populations are created, each containing m and n particles, respectively. The algorithm randomly generates the strategy x^i for population 1 of m particles, x^i satisfies $x^i = 0$ or 1. Then the algorithm randomly generates the strategy y^j for population 2 of n particles, y^j satisfies $y^j = 0$ or 1.

(c) Expected benefits of all particles in both populations are computed using Eqs (3.3) and (3.4), and the particle strategy with the highest expected benefit is found, denoted as i_{best1} and i_{best2}, respectively.

(d) All particles update their positions according to Eqs (3.5) and (3.6).

(e) Whether to end the iteration is determined according to the number of iterations *genmax*. If the iteration ends, the algorithm outputs the two populations' optimal benefits and particle position figures. Otherwise, the algorithm turns to (c).

4. PGPSO algorithm convergence

4.1. PGPSO algorithm update formula

In the PGPSO algorithm, the update rule of players' strategies is as follows: the players with α proportion choose to adjust the strategy in population 1, and the remaining players do not. The players with β proportion choose to adjust their strategy in population 2, and the remaining players do not. For the players who choose to adjust their strategies, they can observe the expected benefits of the players in their population, i.e., the benefits are derived from the Eqs (3.3) and (3.4). Subsequently, these players apply population imitation theory to select an optimal strategy. After the above adjustment, the Eq (2.1) is the basis. The updated formula for the proportion of two populations choosing strategy 1 is obtained as follows:

$$\begin{cases} x_1(t+1) = (1-\alpha)x_1(t) + \alpha[p_1(t)x_1 + p_2(t)(1-x_1(t))] \\ y_1(t+1) = (1-\beta)y_1(t) + \beta[q_1(t)y_1 + q_2(t)(1-y_1(t))] \end{cases}. \tag{4.1}$$

where $x_1(t+1)$, $y_1(t+1)$ is the proportion of players choosing strategy 1 at $t+1$ iterations for both populations, respectively $x_1(t+1), y_1(t+1) \in [0,1]$. If $x_1(t+1)$ and $y_1(t+1)$ are both 0, it implies that both populations choose strategy 2. According to the benefit matrices A and B, all players in the first population receive the benefit a_{22}, and all players in the second population receive the benefit b_{22}.

$$p_1(t) = \begin{cases} 1, F_1^1(t) \geq F_2^1(t) \\ 0, F_1^1(t) < F_2^1(t) \end{cases}; \; p_2(t) = \begin{cases} 1, F_1^1(t) > F_2^1(t) \\ 0, F_1^1(t) \leq F_2^1(t) \end{cases};$$

$$q_1(t) = \begin{cases} 1, F_1^2(t) \geq F_2^2(t) \\ 0, F_1^2(t) < F_2^2(t) \end{cases}; \; q_2(t) = \begin{cases} 1, F_1^2(t) > F_2^2(t) \\ 0, F_1^2(t) \leq F_2^2(t) \end{cases}.$$

The updating Eq (4.1) essentially represents a discretized form of the differential Eqs (3.5) and (3.6). Compared with the Eq (2.1) in social learning theory, Eq (4.1) can be effectively applied to any two-population and two-strategy game model. Second, the social learning rule specifies the concepts of a parent, pending generation, and child, and the parent influences strategy updating. At the same time, the Eq (4.1) removes the concepts of a parent and pending generation, thereby reducing the dependency on heritability as a condition for strategy adaptation.

4.2. PGPSO algorithm convergence analysis

For the benefit matrices of the two-population game:

$$A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix},\ B = \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix}$$

Let $a_1 = a_{11} - a_{21}$, $a_2 = a_{22} - a_{12}$, $b_1 = b_{11} - b_{12}$, $b_2 = b_{22} - b_{21}$.

The benefit matrices simplify to

$$A = \begin{pmatrix} a_1 & 0 \\ 0 & a_2 \end{pmatrix}, B = \begin{pmatrix} b_1 & 0 \\ 0 & b_2 \end{pmatrix}$$

where $a_1 \neq 0, a_2 \neq 0, b_1 \neq 0, b_2 \neq 0$.

Theorem 4.1. In a non-cooperative repeated two-population and two-strategy game with benefit matrices A and B, when each population updates its strategy following the Eq (4.1), the convergence outcome corresponds to either the Nash equilibrium or its stable limit ring.

Proof. The updating Eq (4.1) is transformed to continuous time, and the imitation dynamic equation is obtained using the difference method.

$$\begin{cases} \frac{dx_1}{dt} = -\alpha x_1 + \alpha[p_1(t)x_1 + p_2(t)(1 - x_1)] \\ \frac{dy_1}{dt} = -\beta y_1 + \beta[q_1(t)y_1 + q_2(t)(1 - y_1)] \end{cases}. \tag{4.2}$$

The two-population game is classified into three types according to their equilibria: one pure strategy Nash equilibrium, one mixed strategy Nash equilibrium, and three Nash equilibria (two pure strategy Nash equilibria, one mixed strategy Nash equilibrium). Next, we discuss the classification.

For the first type, if Nash equilibrium $(\bar{x}, \bar{y}) = ((1, 0), (1, 0))$, then $a_1 > 0,\ a_2 < 0,\ b_1 > 0,\ b_2 < 0$, deducing $F_1^1 = a_1 y_1$, $F_2^1 = a_2(1 - y_1)$, $F_1^2 = b_1 x_1$ and $F_2^2 = b_2(1 - x_1)$, implying that $F_1^1 > F_2^1, F_1^2 > F_2^2$. From the Eq (4.2), we get $\frac{dx_1}{dt} = -\alpha x_1 + \alpha \geq 0$, $\frac{dy_1}{dt} = -\beta y_1 + \beta \geq 0$. The imitation dynamic Eq (4.2) converges to $x_1 = y_1 = 1$. That is, it converges to Nash equilibrium $((1, 0), (1, 0))$. When $(\bar{x}, \bar{y}) = ((1, 0), (0, 1))$, $(\bar{x}, \bar{y}) = ((0, 1), (1, 0))$ and $(\bar{x}, \bar{y}) = ((0, 1), (0, 1))$, the analysis is similar. The Eq (4.2) converges to Nash equilibrium, which is proved.

For the second type, Nash equilibrium $(\bar{x}, \bar{y}) = ((\frac{b_2}{b_1+b_2}, \frac{b_1}{b_1+b_2}), (\frac{a_2}{a_1+a_2}, \frac{a_1}{a_1+a_2}))$, then $a_1 < 0,\ a_2 < 0,\ b_1 > 0,\ b_2 > 0$. For the Eq (4.1), we get:

$$p_1(t) = \begin{cases} 0, y_1(t) > \frac{a_2}{a_1+a_2} \\ 1, y_1(t) \leq \frac{a_2}{a_1+a_2} \end{cases}; p_2(t) = \begin{cases} 0, y_1(t) \geq \frac{a_2}{a_1+a_2} \\ 1, y_1(t) < \frac{a_2}{a_1+a_2} \end{cases};$$

$$q_1(t) = \begin{cases} 1, x_1(t) \geq \frac{b_2}{b_1+b_2} \\ 0, x_1(t) < \frac{b_2}{b_1+b_2} \end{cases}; q_2(t) = \begin{cases} 1, x_1(t) > \frac{b_2}{b_1+b_2} \\ 0, x_1(t) \leq \frac{b_2}{b_1+b_2} \end{cases}.$$

When $x_1(0) = \frac{b_2}{b_1+b_2}$, $y_1(0) = \frac{a_2}{a_1+a_2}$, $\frac{dx_1}{dt} = \frac{dy_1}{dt} = 0$, i.e., the imitation dynamic Eq (4.2) will converge to Nash equilibrium $((\frac{b_2}{b_1+b_2}, \frac{b_1}{b_1+b_2}), (\frac{a_2}{a_1+a_2}, \frac{a_1}{a_1+a_2}))$.

If the initial point is not Nash equilibrium, let $b' = \frac{b_2}{b_1+b_2}$, $a' = \frac{a_2}{a_1+a_2}$ and $x^1 = x_1 - \frac{b_2}{b_1+b_2}$, $y^1 = y_1 - \frac{a_2}{a_1+a_2}$, then we can obtain that the Eq (4.2) is a differential equation with zero solutions.

Then the Eq (4.2) becomes

$$\begin{cases} \dot{x^1} = \alpha p_1(t)(x^1 + b') + \alpha p_2(t)(1 - b' - x^1) - \alpha(x^1 + b') \\ \dot{y^1} = \beta q_1(t)(y^1 + a') + \beta q_2(t)(1 - a' - y^1) - \beta(y^1 + a') \end{cases}. \tag{4.3}$$

Taking the polar coordinates $x^1 = r\cos\theta$, $y^1 = r\sin\theta$, we can obtain that the Eq (4.3) is

$$\begin{aligned} \dot{r} = \alpha(p_1(t) - p_2(t) - 1)r\cos^2\theta + \alpha(b'p_1(t) - b'p_2(t) - b' + p_2(t))\cos\theta + \\ \beta(q_1(t) - q_2(t) - 1)r\sin^2\theta + \beta(a'q_1(t) - a'q_2(t) - a' + q_2(t))\sin\theta. \end{aligned} \tag{4.4}$$

Divide the Eq (4.4) into $0 < \theta < \frac{\pi}{2}$, $\frac{\pi}{2} < \theta < \pi$, $\pi < \theta < \frac{3\pi}{2}$, $\frac{3\pi}{2} < \theta < 2\pi$ and $\theta = 0, \frac{\pi}{2}, \pi, \frac{3\pi}{2}, 2\pi$. These cases are classified and discussed.

1) When $0 < \theta < \frac{\pi}{2}$, we get $x^1(t), y^1(t) > 0$, which is obtained from the Eq (4.4) as

$$\dot{r} = -(\alpha\cos^2\theta + \beta\sin^2\theta)r - \alpha b'\cos\theta + \beta(1 - a')\sin\theta.$$

From Eq (4.4), it follows that when $r = \frac{-\alpha b'\cos\theta + \beta(1-a')\sin\theta}{\alpha\cos^2\theta + \beta\sin^2\theta}$, $\dot{r} = 0$, i.e., there is a special solution

$$r = \frac{-\alpha b'\cos\theta + \beta(1 - a')\sin\theta}{\alpha\cos^2\theta + \beta\sin^2\theta}.$$

The solution is a curve in the phase plane centered at the origin.

When $r > \frac{-\alpha b'\cos\theta + \beta(1-a')\sin\theta}{\alpha\cos^2\theta + \beta\sin^2\theta}$, we get $\dot{r} < 0$ from the Eq (4.4). That is, the trajectory converges to the curve from outside the curve. When $r < \frac{-\alpha b'\cos\theta + \beta(1-a')\sin\theta}{\alpha\cos^2\theta + \beta\sin^2\theta}$, we get $\dot{r} > 0$ from the Eq (4.4). That is, the trajectory converges to the curve from inside the curve. Therefore the system has a stable limit ring centered at the origin at $0 < \theta < \frac{\pi}{2}$. A stable limit ring is a periodic solution around a non-isolated equilibrium point. When the solution trajectory evolves from a point in the solution space, it converges to this limit ring and makes a periodic movement on this limit ring.

2) When $\frac{\pi}{2} < \theta < \pi, \pi < \theta < \frac{3\pi}{2}$, $\frac{3\pi}{2} < \theta < 2\pi$, $\dot{r}$ is respectively,

$$-(\alpha\cos^2\theta + \beta\sin^2\theta)r - \alpha b'\cos\theta - \beta a'\sin\theta,$$

$$-(\alpha\cos^2\theta + \beta\sin^2\theta)r + \alpha(1 - b')\cos\theta - \beta a'\sin\theta,$$

$$-(\alpha\cos^2\theta + \beta\sin^2\theta)r + \alpha(1 - b')\cos\theta + \beta(1 - a')\sin\theta.$$

The analysis idea and result are similar to $0 < \theta < \frac{\pi}{2}$.

3) When $\theta = 0, \frac{\pi}{2}, \pi, \frac{3\pi}{2}, 2\pi$, $\dot{r} = 0$.

The polar differential Eq (4.4) has a stable limit ring centered at the origin. That is, the imitation dynamic Eq (4.2) has a stable limit ring centered at $((\frac{b_2}{b_1+b_2}, \frac{b_1}{b_1+b_2}), (\frac{a_2}{a_1+a_2}, \frac{a_1}{a_1+a_2}))$. The imitation dynamic Eq (4.2) will converge to the Nash equilibrium or a stable limit ring centered on the Nash equilibrium.

After the above analysis, the second type proves to be completed.

For the third type, if Nash equilibrium set $E(F) = \{((1,0),(1,0)),((0,1),(0,1)),((\frac{b_2}{b_1+b_2},\frac{b_1}{b_1+b_2}),(\frac{a_2}{a_1+a_2},\frac{a_1}{a_1+a_2}))\}$, then $a_1 > 0,\ a_2 > 0,\ b_1 > 0,\ b_2 > 0$. For the Eq (4.1), we get

$$p_1(t) = \begin{cases} 1, y_1(t) \geq \frac{a_2}{a_1+a_2} \\ 0, y_1(t) < \frac{a_2}{a_1+a_2} \end{cases}; p_2(t) = \begin{cases} 1, y_1(t) \geq \frac{a_2}{a_1+a_2} \\ 0, y_1(t) < \frac{a_2}{a_1+a_2} \end{cases};$$

$$q_1(t) = \begin{cases} 1, x_1(t) \geq \frac{b_2}{b_1+b_2} \\ 0, x_1(t) < \frac{b_2}{b_1+b_2} \end{cases}; q_2(t) = \begin{cases} 1, x_1(t) > \frac{b_2}{b_1+b_2} \\ 0, x_1(t) \leq \frac{b_2}{b_1+b_2} \end{cases}.$$

According to the method of variation of parameters, the solution of the imitation dynamic Eq (4.2) is

$$\begin{cases} x_1(t) = \frac{e^{\alpha(p_1(t)-p_2(t)-1)t}[\alpha(p_1(t)-p_2(t)-1)x_1(0)+\alpha p_2(t)]-\alpha p_2(t)}{\alpha(p_1(t)-p_2(t)-1)} \\ y_1(t) = \frac{e^{\beta(q_1(t)-q_2(t)-1)t}[\beta(q_1(t)-q_2(x_1)-1)y_1(0)+\beta q_2(t)]-\beta q_2(t)}{\beta(q_1(t)-q_2(t)-1)} \end{cases}.$$

For $\forall\varepsilon > 0$, there exists $\delta > 0$, when $\sqrt{(x_1(0)-1)^2+(y_1(0)-1)^2} < \delta$, there is $p_1(t) = p_2(t) = q_1(t) = q_2(t) = 1$. From the Eq (4.2), we get

$$\lim_{t\to+\infty}\sqrt{(x_1(t)-1)^2+(y_1(t)-1)^2} = \sqrt{(1-1)^2+(1-1)^2} = 0.$$

This proves the asymptotic stability of $((1,0),(1,0))$.

When $\sqrt{x_1(0)^2+y_1(0)^2} < \delta$, we have $p_1(t) = p_2(t) = q_1(t) = q_2(t) = 0$. From the Eq (4.2), we get

$$\lim_{t\to+\infty}\sqrt{x_1(t)^2+y_1(t)^2} = \sqrt{0^2+0^2} = 0.$$

This proves the asymptotic stability of $((0,1),(0,1))$.

If and only if $x_1(0) = \frac{b_2}{b_1+b_2}$, $y_1(0) = \frac{a_2}{a_1+a_2}$, we get $\lim_{t\to+\infty} x_1(t) = \frac{b_2}{b_1+b_2}$, $\lim_{t\to+\infty} y_1(t) = \frac{a_2}{a_1+a_2}$.

According to the above analysis, the solution trajectory of the Eq (4.2) starts from any position in the solution space. It will converge to an element of the Nash equilibrium set $E(F)$.

When the Nash equilibrium set

$$E(F) = \{((1,0),(0,1)),((0,1),(1,0)),((\frac{b_2}{b_1+b_2},\frac{b_1}{b_1+b_2}),(\frac{a_2}{a_1+a_2},\frac{a_1}{a_1+a_2}))\},$$

the analysis is similar. The Eq (4.2) converges to Nash equilibrium, which is proved.

Example 4.1. Take the prisoner's dilemma game with the following benefit matrices. We set the introspection rate $\alpha = 0.1, \beta = 0.2$, the horizontal or vertical separation of the initial position is 0.2, and the arrows represent the direction of the solution trajectory. The solution trajectory is shown in Figure 1.

$$A_1 = \begin{pmatrix} -5 & 0 \\ -8 & -1 \end{pmatrix},\ B_1 = \begin{pmatrix} -5 & -8 \\ 0 & -1 \end{pmatrix}$$

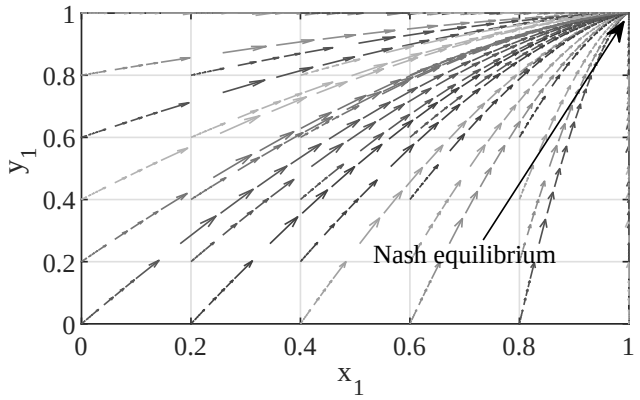


Figure 1. Solution trajectories of the imitation dynamic Eq (4.2) for the prisoner's dilemma game.

From the Figure 1, it can be seen that solution trajectories eventually converge to $(1, 1)$ for all initial points, i.e., the solution trajectories converge to Nash equilibrium $(\bar{x}, \bar{y}) = ((1, 0), (1, 0))$.

Example 4.2. Take the coin-flip game with the following benefit matrices. We set the introspection rate $\alpha = 0.1, \beta = 0.2$, the horizontal or vertical separation of the initial position is 0.25, and the arrow represents the direction of the solution trajectory. The solution trajectory is shown in Figure 2.

$$A_2 = \begin{pmatrix} -1 & 1 \\ 1 & -1 \end{pmatrix}, \ B_2 = \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}$$

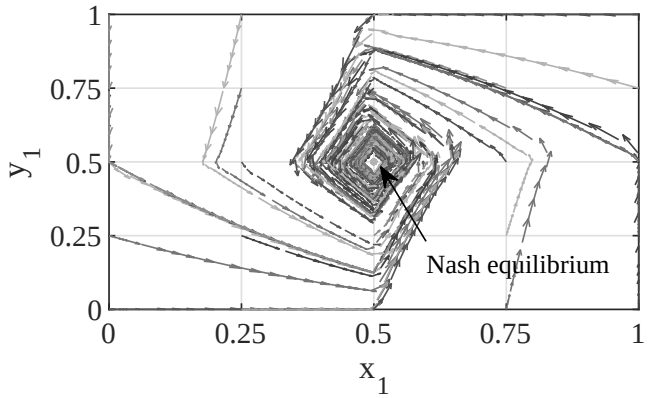


Figure 2. Solution trajectory of the imitation dynamic Eq (4.2) for the coin-flip game.

From the Figure 2, we can see that when the initial point is $(0.5, 0.5)$, the solution trajectory converges to $(0.5, 0.5)$, i.e., Nash equilibrium $(\bar{x}, \bar{y}) = ((\frac{1}{2}, \frac{1}{2}), (\frac{1}{2}, \frac{1}{2}))$; when the initial point is other than $(0.5, 0.5)$, the solution trajectory evolves counterclockwise to $(0.5, 0.5)$ and converges to the limit ring centered at $(0.5, 0.5)$, i.e., a stable limit ring centered at Nash equilibrium $(\bar{x}, \bar{y}) = ((\frac{1}{2}, \frac{1}{2}))$.

Example 4.3. Take the coordination game with the following benefit matrices. We set the introspection rate $\alpha = 0.1$, $\beta = 0.2$, the horizontal or vertical separation of the initial position is $\frac{1}{6}$, and the arrow represents the direction of the solution trajectory. The solution trajectory is shown in Figure 3.

$$A_3 = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix},\ B_3 = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}$$

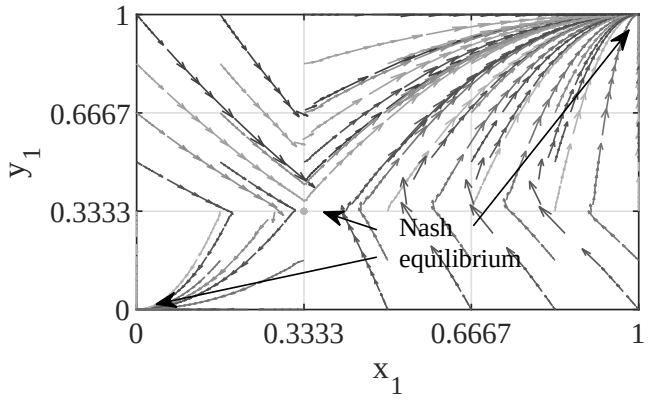


Figure 3. Solution trajectory of the imitation dynamic Eq (4.2) for the coordination game.

From the Figure 3, when the initial point is $(\frac{1}{3}, \frac{1}{3})$, the solution trajectory converges to $(\frac{1}{3}, \frac{1}{3})$, i.e., Nash equilibrium $(\bar{x}, \bar{y}) = ((\frac{1}{3}, \frac{2}{3}), (\frac{1}{3}, \frac{2}{3}))$; when the initial point is to the upper right of $(\frac{1}{3}, \frac{1}{3})$, the solution trajectory converges to $(1, 1)$, i.e., Nash equilibrium $((1, 0), (1, 0))$. When the initial point is to the lower left of $(\frac{1}{3}, \frac{1}{3})$, the solution trajectory converges to $(0, 0)$, i.e., Nash equilibrium $((0, 1), (0, 1))$.

Research in the realm of realizing or computing Nash equilibrium through algorithmic approaches has led to various contributions. Zhang et al. [35] proposed the PMR-IGA algorithm, and Zhang et al. [36] proposed the SA-IGA algorithm. Both algorithms are based on the reinforcement learning theory, striving to guide individuals within a population through iterative processes that ultimately lead to Nash equilibrium convergence. For the computation of Nash equilibrium, Li et al. [37] proposed the GPDEPSO algorithm to compute the Nash equilibrium of a finite non-cooperative game. This approach equates solving the Nash equilibrium to solving an optimization problem and considers the algorithmic process stochastic. Stochastic generalized function theory is adopted to prove the convergence to the Nash equilibrium.

On the other hand, the PGPSO algorithm's inspiration differs from the above-mentioned approaches, as it is based on the theory of social learning and population imitation. Its convergence is demonstrated by transforming the iterative process into differential equations. It proves the pure Nash equilibrium strategy's asymptotic stability from an equation-driven standpoint. In this case, the unique mixed strategy Nash equilibrium is the center of the stable limit ring in the solution space, and the solution trajectories from any point will converge to that limit ring.

5. PGPSO algorithm simulation experiments

In this section, we study the Nash equilibrium realizations of the prisoner's dilemma game, the coin-flip game, and the hawk-dove game using the PGPSO algorithm, respectively. The prisoner's dilemma game has only one pure strategy Nash equilibrium, the coin-flip game has only one mixed

strategy Nash equilibrium, and the hawk-dove game has three Nash equilibria, i.e., two pure strategy Nash equilibria and one mixed strategy Nash equilibrium.

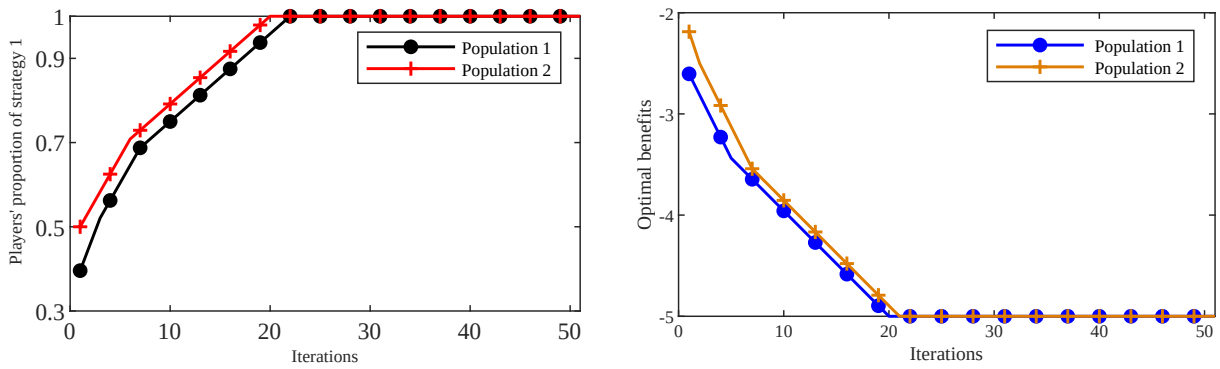
5.1. Test 1: Prisoner's dilemma game

The prisoner's dilemma game, typically a two-person game, has been employed by Wang [38] to corroborate the presence of the "baiting effect" within social populations. Notably, the benefit matrices characteristic of the inter-player game can be seamlessly applied to the population game model. Therefore, this game model can be used as an example of the population game. The prisoner's dilemma game has only one pure-strategy Nash equilibrium. Both populations choose strategy 1, i.e., $(\bar{x}, \bar{y}) = ((1,0),(1,0))$. The following are the benefit matrices and expected benefits of the prisoner's dilemma game:

$$A_1 = \begin{pmatrix} -5 & 0 \\ -8 & -1 \end{pmatrix}, \; B_1 = \begin{pmatrix} -5 & -8 \\ 0 & -1 \end{pmatrix},$$

$$F^1 = 2x^i y_1 + x^i - 7y_1 - 1 \; ; \; F^2 = 2x_1 y^j - 7x_1 + y^j - 1.$$

In the PGPSO algorithm of the prisoner's dilemma game, we set the introspection rate $\alpha = \beta = \begin{cases} \frac{1}{12}, & k \le 25 \\ \frac{1}{24}, & k > 25 \end{cases}$, the population size $m = n = 48$, the search space range from $popmin = 0$ to $popmax = 1$, and the number of iterations $genmax = 50$. The initial states of the populations are chosen randomly by the system. The two populations' optimal benefit and location figures are shown in Figure 4.



(a) The particle position

(b) The optimal benefit

Figure 4. The figures of two populations in the prisoners' dilemma game.

From the Figure 4(a),(b) we can see that all the particles of population 1 converge to $x_1 = 1$, i.e., all players of population 1 choose the strategy 1, and the best benefit of population 1 converges to -5. All particles of population 2 converge to $y_1 = 1$, i.e., all players of population 2 choose strategy 1, and the best benefit of population 2 converges to -5. This outcome is consistent with the Nash equilibrium $((1,0),(1,0))$ of the prisoner's dilemma game. Therefore, the PGPSO algorithm accurately finds the particles' positions and benefits corresponding to its Nash equilibrium strategy. It completely records the path to the Nash equilibrium of the prisoner's dilemma game.

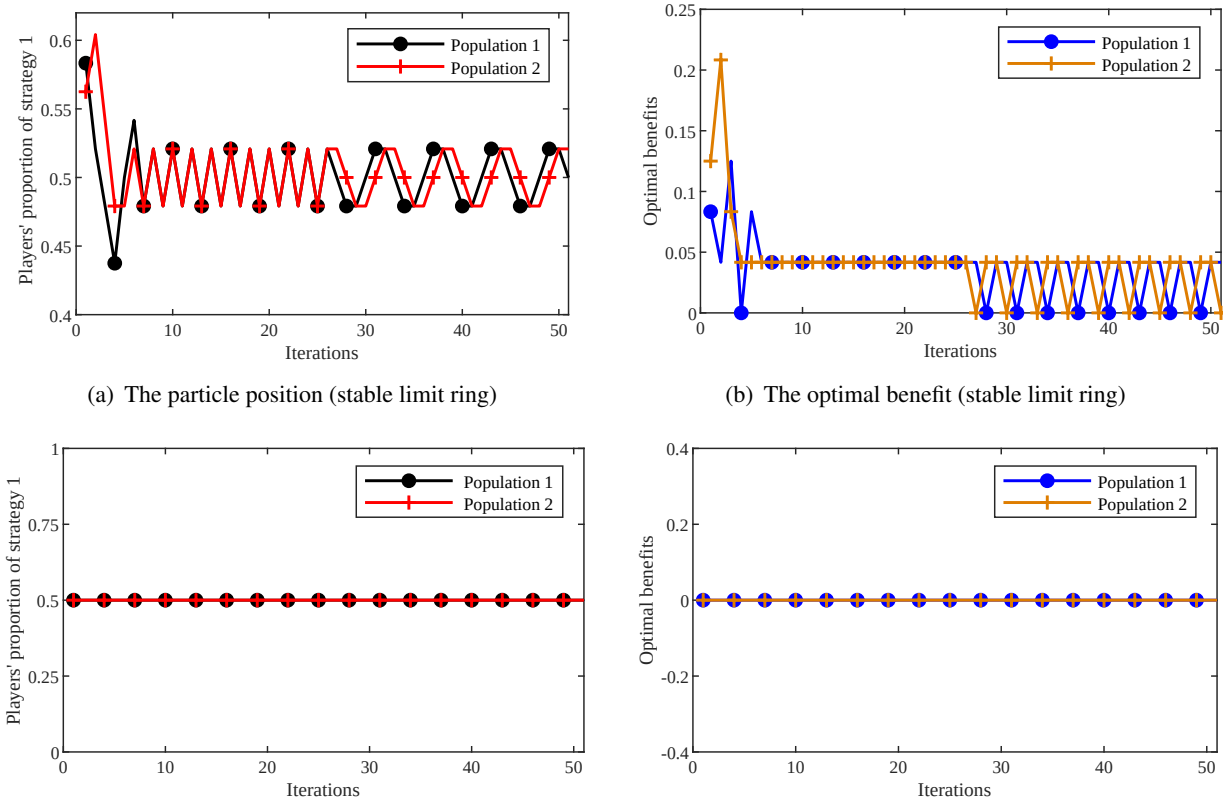
5.2. *Test 2: Coin-flip game*

Consider the coin-flip game as a population game, where two populations play against each other according to the benefit matrices of the coin-flip game. The coin-flip game has a mixed-strategy Nash equilibrium, i.e., $(\bar{x}, \bar{y}) = ((\frac{1}{2}, \frac{1}{2}), (\frac{1}{2}, \frac{1}{2}))$. The following are the benefit matrices and expected benefits of the coin-flip game:

$$A_2 = \begin{pmatrix} -1 & 1 \\ 1 & -1 \end{pmatrix}, \; B_2 = \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix},$$

$$F^1 = -4x^i y_1 + 2x^i + 2y_1 - 1 \; ; \; F^2 = 4x_1 y^j - 2x_1 - 2y^j + 1.$$

In the PGPSO algorithm of the coin-flip game, we set the introspection rate $\alpha = \beta = \begin{cases} \frac{1}{12}, & k \le 25 \\ \frac{1}{24}, & k > 25 \end{cases}$, the population size $m = n = 48$, the search space range from $popmin = 0$ to $popmax = 1$ and the number of iterations $genmax = 50$. The initial states of the populations are chosen randomly by the system. The two populations' optimal benefit and location figures are shown in Figure 5.



(a) The particle position (stable limit ring)

(b) The optimal benefit (stable limit ring)

(c) The particle position (Nash equilibrium)

(d) The optimal benefit (Nash equilibrium)

Figure 5. The figures of two populations in the coin-flip game.

From the Figure 5(a),(b), we can see that all particles of population 1 converge cyclically to $x_1 = \frac{1}{2}$, i.e., nearly half of the players choose the strategy 1 in population 1. The optimal benefit of population

1 converges cyclically at 0.025. All particles of population 2 converge cyclically to $y_1 = \frac{1}{2}$, i.e., nearly half of the players choose strategy 1 in population 2. The optimal benefit of population 2 converges cyclically to 0.025. All particles converge cyclically to $((\frac{1}{2}, \frac{1}{2}), (\frac{1}{2}, \frac{1}{2}))$. It represents the evolution of the two populations toward Nash equilibrium, which eventually converges to a neighborhood centered on Nash equilibrium. Corresponding to the mixed-strategy Nash equilibrium is a stable limit ring in the imitation dynamic equation Eq (4.2). And from the Figure 5(c),(d), we can see that when the initial strategy distribution is Nash equilibrium for the two populations, all particles of population 1 converge to $x_1 = \frac{1}{2}$. The optimal benefit of population 2 converges to 0. All particles of population 2 converge to $y_1 = \frac{1}{2}$, and the optimal benefit of population 2 converges to 0. This outcome is consistent with the Nash equilibrium $((\frac{1}{2}, \frac{1}{2}), (\frac{1}{2}, \frac{1}{2}))$ of the coin-flip game. Therefore, it can be seen that the PGPSO algorithm accurately finds the particle positions and benefits corresponding to its Nash equilibrium strategy. It completely records the path to the Nash equilibrium of the coin-flip game.

5.3. Test 3: Hawk-dove game

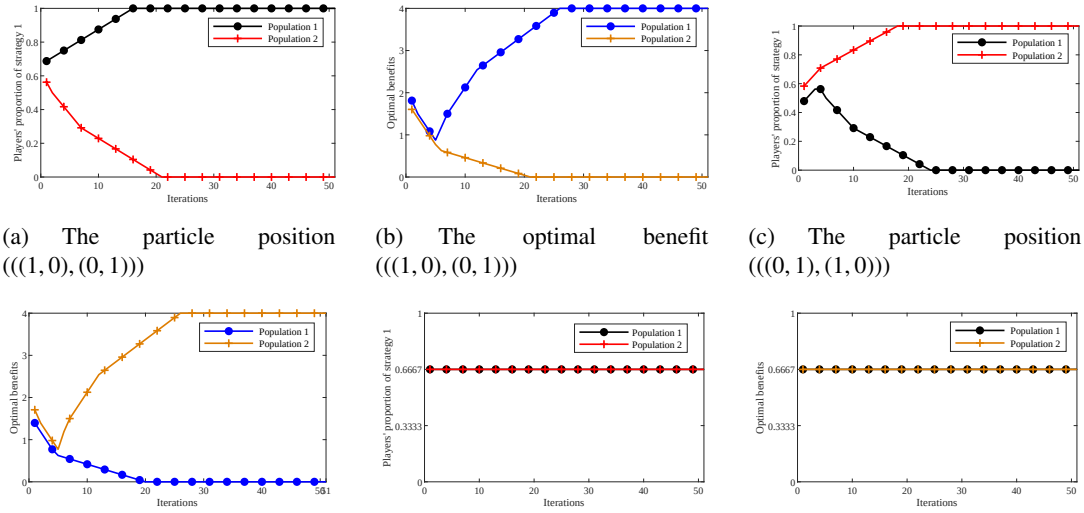
The hawk-dove game is an important population game model in evolutionary game theory. The model has three Nash equilibria, and the set of Nash equilibria is $E(F) = \{((1,0),(0,1)),((0,1),(1,0)),((\frac{2}{3}, \frac{1}{3}),(\frac{2}{3}, \frac{1}{3}))\}$. The following are the benefit matrices and expected benefits of the hawk-dove game:

$$A_4 = \begin{pmatrix} -1 & 4 \\ 0 & 2 \end{pmatrix}, B_4 = \begin{pmatrix} -1 & 0 \\ 4 & 2 \end{pmatrix},$$

$$F^1 = -3x^i y_1 + 2x^i - 2y_1 + 2\ ;\ F^2 = -3x_1 y^j - 2x_1 + 2y^j + 2.$$

In the PGPSO algorithm of the hawk-dove game, we set the introspection rate $\alpha = \beta = \begin{cases} \frac{1}{12}, & k \le 25 \\ \frac{1}{24}, & k > 25 \end{cases}$, the population size $m = n = 48$, the search space range from $popmin = 0$ to $popmax = 1$, and the number of iterations $genmax = 50$. The initial states of the populations are chosen randomly by the system. The two populations' optimal benefit and location figures are shown in Figure 6.

From the Figure 6(a),(b), it can be seen that all particles of population 1 converge to $x_1 = 1$, and the optimal benefit converges to 4. All particles of population 2 converge to $y_1 = 0$, and the optimal benefit converge to 0. From the Figure 6(c),(d), all particles of population 1 converge to $x_1 = 0$, and the optimal benefit converges to 0; all particles of population 2 converge to $y_1 = 1$, and the optimal benefit converges to 4. From the Figure 6(e),(f), when the initial strategy distribution of two populations is Nash equilibria $((\frac{2}{3}, \frac{1}{3}), (\frac{2}{3}, \frac{1}{3}))$, all particles of population 1 converge to $x_1 = \frac{2}{3}$, and the optimal benefit converges to $\frac{2}{3}$. All particles of population 2 converge to $y_1 = \frac{2}{3}$, and the optimal benefit converges to $\frac{2}{3}$. This outcome is consistent with the three Nash equilibria of the hawk-dove game, $E(F) = \{((1,0),(0,1)),((0,1),(1,0)),((\frac{2}{3}, \frac{1}{3})),(\frac{2}{3}, \frac{1}{3}))\}$. Therefore, it can be seen that the PGPSO algorithm accurately finds the particle positions and benefits corresponding to its Nash equilibrium strategy. It completely records the path to the Nash equilibrium of the hawk-dove game.



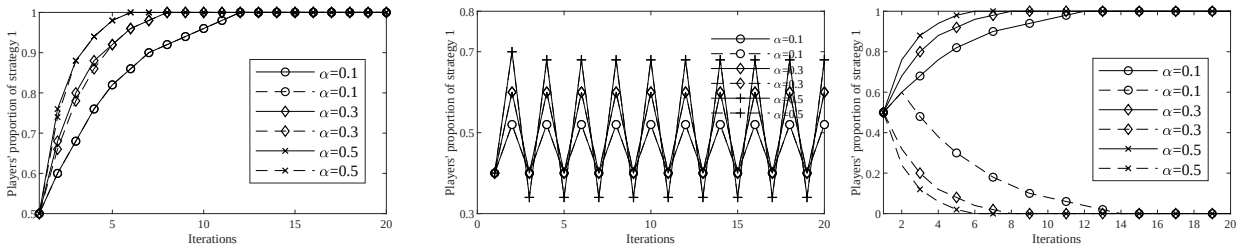
(a) The particle position $(((1,0),(0,1)))$ (b) The optimal benefit $(((1,0),(0,1)))$ (c) The particle position $(((0,1),(1,0)))$

(d) The optimal benefit $(((0,1),(1,0)))$ (e) The particle position $(((\frac{2}{3},\frac{1}{3}),(\frac{2}{3},\frac{1}{3})))$ (f) The optimal benefit $((\frac{2}{3},\frac{1}{3}),(\frac{2}{3},\frac{1}{3}))$

Figure 6. The figures of two populations in the hawk-dove game.

5.4. Effect of introspection rate on Nash equilibrium realization

In the PGPSO algorithm with introspection rate sensitivity of the three games, we set the introspection rate $\alpha = \beta$, the population size $m = n = 48$, the search space range from $popmin = 0$ to $popmax = 1$, and the number of iterations $genmax = 20$. The initial states of the prisoner's dilemma and the hawk-dove game are $x_1(0) = y_1(0) = 0.5$, and the initial states of the coin-flip game are $x_1(0) = y_1(0) = 0.4$. The position figures of the two populations are shown in Figure 7.



(a) The prisoner's dilemma game (b) The coin-flip game (c) The hawk-dove game

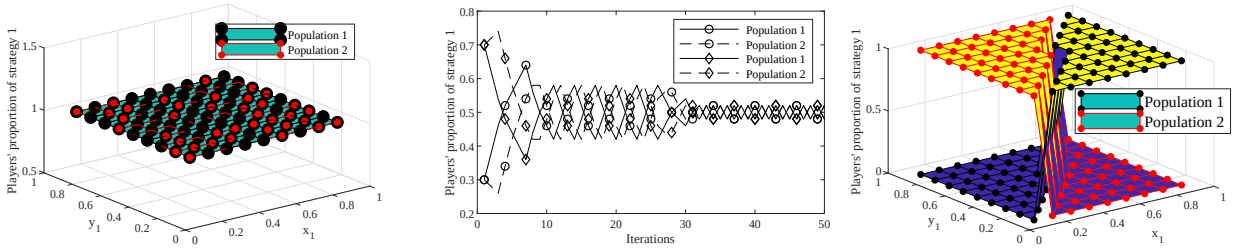
Figure 7. The relationship between Nash equilibrium realization and introspection rate for the three games. The solid line represents population 1, and the dashed line represents population 2.

From the Figure 7(a),(c), it can be seen that in the prisoner's dilemma and the hawk-dove game, for the two populations, the number of players converging to Nash equilibrium decreases as the introspection rate α increases, representing that the increase of the introspection rate α speeds up the evolution of Nash equilibrium realization. From the Figure 7(b), it can be seen that in the coin-flip

game, the magnitude of the cycle convergence increases with the increase of the introspection rate α, which means that the increase of the introspection rate α expands the range of cycle fluctuations.

5.5. *Effect of initial state on Nash equilibrium realization*

In the PGPSO algorithm for the initial state sensitivity of the three games, we set the introspection rate $\alpha = \beta = \begin{cases} \frac{1}{12}, k \leq 25 \\ \frac{1}{24}, k > 25 \end{cases}$, the population size $m = n = 48$, the search space range from $popmin = 0$ to $popmax = 1$ and the number of iterations $genmax = 50$. For the prisoner's dilemma and the hawk-dove game, the initial state range of population 1 is 0.1–0.9, and the positions are chosen every 0.1. The initial state range of population 2 is chosen with the same rules as population 1. A total of 81 parameter configurations are generated by combining the two populations. For each parameter configuration, a series of 50 experiments are conducted to observe the system's steady state. Two initial states (0.1, 0.1) and (0.9, 0.9) are selected for the coin-flip game. The location figures of the two populations are shown in Figure 8.



(a) The prisoner's dilemma game (b) The coin-flip game (c) The hawk-dove game

Figure 8. The relationship between Nash equilibrium realization and initial state for the three games. The solid line represents population 1, and the dashed line represents population 2.

From Figure 8(a), it can be seen that the populations start from any initial state in the prisoner's dilemma game. The two populations converge to the Nash equilibrium $((1, 0), (1, 0))$, which means that the change of initial state can't affect the Nash equilibrium realization. From Figure 8(b), it can be seen that the population starts from two initial states in the coin-flip game. The two populations converge cyclically to the mixed strategy Nash equilibrium $((\frac{1}{2}, \frac{1}{2}), (\frac{1}{2}, \frac{1}{2}))$, representing that the initial state changes can't affect the Nash equilibrium realization. From Figure 8(c), it can be seen that in the hawk-dove game, the two populations converge to the different Nash equilibrium from the different initial states. Specifically, when the initial state is on the right side of the diagonal with $x = y$, the two populations converge to the Nash equilibrium $((1, 0), (0, 1))$. Conversely, when the initial state is on the left side of this diagonal, the two populations converge to Nash equilibrium $((0, 1), (1, 0))$. Subtle nuances emerge when considering different positions along the diagonal. Specifically, for the first five positions, the populations converge to $((0, 1), (1, 0))$, while the last four positions lead to $((0, 1), (1, 0))$. These results represent that the initial state will affect the Nash equilibrium realization.

5.6. Comparison with Meta Equilibrium Q-learning algorithm

Taking the welfare game in [31] as an example, for finding its mixed-strategy Nash equilibrium realization path, we compare the difference between the PGPSO algorithm and the Meta Equilibrium Q-learning algorithm.

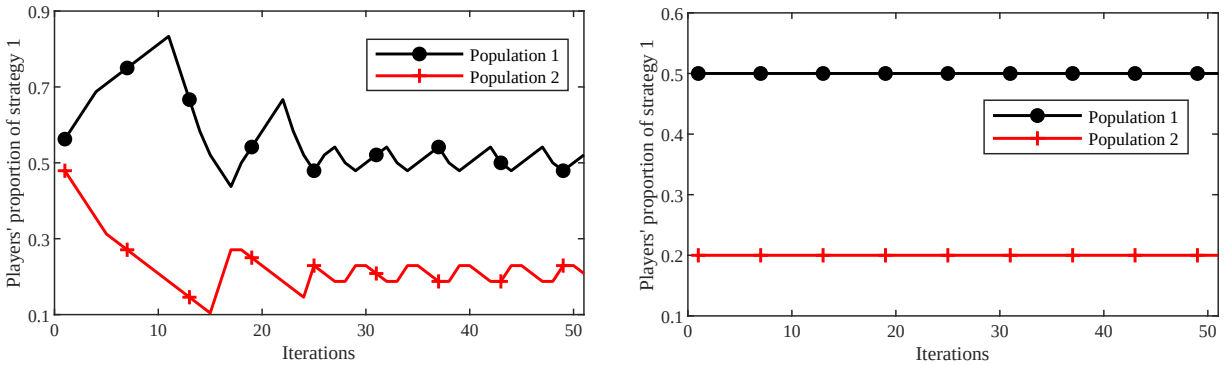
Example 5.1. This game model has a mixed-strategy Nash equilibrium, i.e., $(\bar{x}, \bar{y}) = ((\frac{1}{2}, \frac{1}{2}), (\frac{1}{4}, \frac{3}{4}))$, and the followings are the benefit matrices and expected benefits of the welfare game:

$$A_5 = \begin{pmatrix} 3 & -1 \\ -1 & 0 \end{pmatrix}, B_5 = \begin{pmatrix} 2 & 3 \\ 1 & 0 \end{pmatrix},$$

$$F^1 = 5x^i y_1 - x^i - y_1 \; ; \; F^2 = -2x_1 y^j + 3x_1 + y^j.$$

The Meta Equilibrium Q-learning algorithm represents an enhancement over the Nash Q-learning algorithm. The rationale behind this improvement stems from the Nash Q-learning algorithm's limitation C specifically, its inability to devise a pathway to realize a mixed-strategy Nash equilibrium when each player opts for a pure strategy. In order to address this problem, the Meta Equilibrium Q-learning algorithm transforms the welfare game into a meta-game. It uses the pure strategy meta-equilibrium to represent the mixed-strategy Nash equilibrium in the welfare game. The meta-equilibrium's realization path replaces the mixed strategy's realization path. Thus, we can obtain the path to find the mixed-strategy Nash equilibrium. However, it is important to note that this transformation comes at the expense of increased complexity in locating the realization path for the mixed-strategy Nash equilibrium.

In the PGPSO algorithm of the welfare game, we set the introspection rate $\alpha = \beta = \begin{cases} \frac{1}{12}, k \leq 25 \\ \frac{1}{24}, k > 25 \end{cases}$, the population size $m = n = 48$, the search space range from $popmin = 0$ to $popmax = 1$ and the number of iterations $genmax = 50$. The computer system randomly selects the initial states of the populations. The position figures of the two populations are shown in Figure 9.



(a) The stable limit ring

(b) The Nash equilibrium

Figure 9. The figures of two populations in the welfare game.

From Figure 9(a), it can be seen that after 25 iterations, all particles of population 1 converge cyclically to $x_1 = \frac{1}{2}$ and all particle of population 2 converge cyclically to $y_1 = \frac{1}{4}$. The difference

between the maximum and minimum values of the cycle convergence is 0.0625. From the Figure 9(b), when the initial strategy distribution of the two populations is Nash equilibrium, all particles of population 1 converge to $x_1 = \frac{1}{2}$, and all particles of population 2 converge to $y_1 = \frac{1}{4}$. This outcome is consistent with the Nash equilibrium of the welfare game. Therefore, for players who choose a pure strategy in the welfare game, the PGPSO algorithm converges to the mixed strategy Nash equilibrium and finds the path to realize the equilibrium.

In the welfare game, for the problem of finding the realization path of the mixed-strategy Nash equilibrium, the Meta Equilibrium Q-learning algorithm transforms the welfare game into a meta-game. The meta-equilibrium's realization path replaces the mixed strategy's realization path. From Figure 9(a), it can be seen that the PGPSO algorithm converges the stable limit ring centered on the mixed-strategy Nash equilibrium. The PGPSO algorithm directly finds the realization path, which means that the algorithm reduces the complexity by eliminating the operation of transforming the welfare game into a meta-game.

6. Conclusions

The population's Nash equilibrium exists commonly in human societies and biological populations. Its realization is crucial from the perspective of individual players within a population. In a population of finite players, players can optimize their strategies by imitating other players with higher benefits, depending on their knowledge of their strategic environment. So far, the rule of imitation has been widely studied in different game models. In this paper, we combine social learning theory and population imitation theory, develop the PGPSO algorithm, and apply it to the Nash equilibrium realization of three two-population game models.

The motivation for studying the imitation learning rule is to explore a problem: whether imitation learning rules can realize the Nash equilibrium realization of population games. Specifically, the new learning rule is transformed into a swarm intelligence algorithm, which is used to simulate the behavioral dynamics of the players in the game. For the PGPSO algorithm iterative formulation, the convergence analysis is performed from the perspective of differential equations. The result is that the solution trajectory of differential equations converges completely to the pure strategy Nash equilibrium. The solution trajectory will converge completely to the mixed strategy Nash equilibrium when the initial position is the mixed strategy Nash equilibrium. Also, in the coin-flip game, the mixed-strategy Nash equilibrium is the center of a stable limit ring of the differential equation. When the initial position is not on the mixed strategy Nash equilibrium, all initial points converge to this limit ring.

Using the PGPSO algorithm, we simulate the Nash equilibrium realization process for three two-population games. Simulation outcomes demonstrate that the PGPSO algorithm successfully realizes Nash equilibrium realization. Meanwhile, the PGPSO algorithm clearly shows the path of realizing Nash equilibrium. According to the analysis of the effect of introspection rate and initial state on the realization of Nash equilibrium, the increase of introspection rate accelerates the evolution of pure strategy Nash equilibrium realization. However, it expands the range of cycle fluctuations of mixed strategy Nash equilibrium. The change in the initial state can't affect the Nash equilibrium realization of the prisoner's dilemma and the coin-flip game, but it causes the hawk-dove game to converge to the different Nash equilibrium.

Abbreviations

The following abbreviations are used in this manuscript.

PSO	Particle swarm optimization
EWA	Experience-adding weight affinity
IGA	Infinitesimal gradient algorithm
GIGA	Generalized infinitesimal gradient algorithm
PGPSO	Population game particle swarm optimization

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No. 71961003) and Science and Technology Program of Guizhou Province (Grant No. 7223), and the Doctoral Foundation Project of Guizhou University (Grant No. 49).

Conflict of interest

The authors declare there is no conflicts of interest.

References

1. J. F. Nash, Equilibrium points in n-person games, *PNAS*, **36** (1950), 48–49. https://doi.org/10.1073/pnas.36.1.48
2. J. Nash, Non-cooperative games, *Ann. Math.*, **54** (1951), 286–295. https://doi.org/10.2307/1969529
3. D. Fudenberg, D. K. Levine, *The Theory of Learning in Games*, MIT Press Books, 1998.
4. J. Kennedy, R. Eberhart, Particle swarm optimization, in *Proceedings of ICNN'95 - International Conference on Neural Networks*, **4** (1995), 1942–1948. https://doi.org/10.1109/ICNN.1995.488968
5. R. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, in *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, (1995), 39–43. https://doi.org/10.1109/MHS.1995.494215
6. D. Fudenberg, D. K. Levine, Consistency and cautious fictitious play, *J. Econ. Dyn. Control*, **19** (1995), 1065–1089. https://doi.org/10.1016/0165-1889(94)00819-4
7. D. Fudenberg, D. M. Kreps, *Lectures on Learning and Equilibrium in Strategic form Games*, Core Foundation, 1992.

8. J. Robinson, An iterative method of solving a game, *Ann. Math.*, **54** (1951), 296–301. https://doi.org/10.2307/1969530
9. J. H. Nachbar, "Evolutionary" selection dynamics in games: convergence and limit properties, *Int. J. Game Theory*, **19** (1990), 59–89. https://doi.org/10.1007/BF01753708
10. V. Krishna, T. Sjostrom, On the convergence of fictitious play, *Math. Oper. Res.*, **23** (1998), 479–511. https://doi.org/10.1287/moor.23.2.479
11. D. Monderer, D. Samet, A. Sela, Belief affirming in learning processes, *J. Econ. Theory*, **73** (1997), 438–452. https://doi.org/10.1006/jeth.1996.2245
12. D. Fudenberg, D. M. Kreps, Learning mixed equilibria, *Games Econ. Behav.*, **5** (1993), 320–367. https://doi.org/10.1006/game.1993.1021
13. P. Milgrom, J. Roberts, Adaptive and sophisticated learning in normal form games, *Games Econ. Behav.*, **3** (1991), 82–100. https://doi.org/10.1016/0899-8256(91)90006-Z
14. K. H. Schlag, Why imitate, and if so, how?: A boundedly rational approach to multi-armed bandits, *J. Econ. Theory*, **78** (1998), 130–156. https://doi.org/10.1006/jeth.1997.2347
15. J. Bjornerstedt, J. W. Weibull, *Nash Equilibrium and Evolution by Imitation*, Working Paper Series, 1994. Available from: https://www.econstor.eu/bitstream/10419/94705/1/wp407.pdf.
16. K. Binmore, L. Samuelson, Muddling through: noisy equilibrium selection, *J. Econ. Theory*, **74** (1997), 235–265. https://doi.org/10.1006/jeth.1996.2255
17. J. Bjornerstedt, *Experimentation, Imitation and Evolutionary Dynamics*, University of Stockholm, mimeo, 1993.
18. K. Binmore, L. Samuelson, Evolutionary drift and equilibrium selection, *Rev. Econ. Stud.*, **66** (1999), 363–393. https://doi.org/10.1111/1467-937X.00091
19. D. Fudenberg, D. Levine, Learning in games, *Eur. Econ. Rev.*, **42** (1998), 631–639. https://doi.org/10.1016/S0014-2921(98)00011-7
20. J. Heinrich, M. Lanctot, D. Silver, Fictitious self-play in extensive-form games, in *Proceedings of the 32nd International Conference on Machine Learning*, **37** (2015), 805–813. Available from: http://proceedings.mlr.press/v37/heinrich15.
21. J. Heinrich, D. Silver, Deep reinforcement learning from self-play in imperfect-information games, preprint, arXiv:1603.01121.
22. P. Muller, S. Omidshafiei, M. Rowland, K. Tuyls, J. Perolat, S. Liu, et al., A generalized training approach for multiagent learning, preprint, arXiv:1909.12823.
23. L. Marris, P. Muller, M. Lanctot, K. Tuyls, T. Graepel, Multi-agent training beyond zero-sum with correlated equilibrium meta-solvers, in *Proceedings of the 38th International Conference on Machine Learning*, **139** (2021), 7480–7491. Available from: https://proceedings.mlr.press/v139/marris21a.html.
24. D. Balduzzi, M. Garnelo, Y. Bachrach, W. Czarnecki, J. Perolat, M. Jaderberg, et al., Open-ended learning in symmetric zero-sum games, in *Proceedings of the 36th International Conference on Machine Learning*, **97** (2019), 434–443. Available from: https://proceedings.mlr.press/v97/balduzzi19a.html.

25. M. L. Littman, Markov games as a framework for multi-agent reinforcement learning, *Mach. Learn. Proc. 1994*, (1994), 157–163. https://doi.org/10.1016/B978-1-55860-335-6.50027-1

26. T. Borgers, R. Sarin, Learning through reinforcement and replicator dynamics, *J. Econ. Theory*, **77** (1997), 1–14. https://doi.org/10.1006/jeth.1997.2319

27. J. S. Jordan, Bayesian learning in normal form games, *Games Econ. Behav.*, **3** (1991), 60–81. https://doi.org/10.1016/0899-8256(91)90005-Y

28. C. Camerer, T. H. Ho, Experience-weighted attraction learning in normal form games, *Econometrica*, **67** (1999), 827–874. https://doi.org/10.1111/1468-0262.00054

29. S. Singh, M. Kearns, Y. Mansour, Nash convergence of gradient dynamics in general-sum games, *UAI*, (2000), 541–548.

30. M. Zinkevich, Online convex programming and generalized infinitesimal gradient ascent, in *Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003)*, (2003), 928–936. Available from: https://cdn.aaai.org/ICML/2003/ICML03-120.pdf.

31. J. Wang, L. Cao, X. Chen, Y. Chen, Z. Zhao, Game reinforcement learning for pure strategy Nash equilibrium (in Chinese), *Comput. Eng. Appl.*, **58** (2022), 78–86. https://doi.org/10.3778/j.issn.1002-8331.2112-0167

32. W. H. Sandholm, *Population Games and Evolutionary Dynamics*, MIT press, 2010.

33. A. Nmeth, K. Takcs, The paradox of cooperation benefits, *J. Theor. Biol.*, **264** (2010), 301–311. https://doi.org/10.1016/j.jtbi.2010.02.005

34. Z. Wang, S. Kokubo, M. Jusup, J. Tanimoto, Universal scaling for the dilemma strength in evolutionary games, *Phys. Life Rev.*, **14** (2015), 1–30. https://doi.org/10.1016/j.plrev.2015.04.033

35. Z. Zhang, D. Wang, D. Zhao, Q. Han, T. Song, A gradient-based reinforcement learning algorithm for multiple cooperative agents, *IEEE Access*, **6** (2018), 70223–70235. https://doi.org/10.1109/ACCESS.2018.2878853

36. C. Zhang, X. Li, J. Hao, S. Chen, K. Tuyls, W. Xue, et al., SA-IGA: a multiagent reinforcement learning method towards socially optimal outcomes, *Auton. Agents Multi-Agent Syst.*, **33** (2019), 403–429. https://doi.org/10.1007/s10458-019-09411-3

37. H. Li, S. Xiang, Y. Yang, C. Liu, Differential evolution particle swarm optimization algorithm based on good point set for computing Nash equilibrium of finite noncooperative game, *AIMS Math.*, **6** (2021), 1309–1323. https://doi.org/10.3934/math.2021081

38. Z. Wang, M. Jusup, L. Shi, J. Lee, Y. Iwasa, S. Boccaletti, Exploiting a cognitive bias promotes cooperation in social dilemma experiments, *Nat. Commun.*, **9** (2018), 2954. https://doi.org/10.1038/s41467-018-05259-5

AIMS Press