Convergence of Multi-Agent Learning with a Finite Step Size in General-Sum Games

Xinliang Song
Tsinghua University
Beijing, China
songxinliang@outlook.com

Tonghan Wang
Tsinghua University
Beijing, China
tonghanwang1996@gmail.com

Chongjie Zhang
Tsinghua University
Beijing, China
chongjie@tsinghua.edu.cn

ABSTRACT

Learning in a multi-agent system is challenging because agents are simultaneously learning and the environment is not stationary, undermining convergence guarantees. To address this challenge, this paper presents a new gradient-based learning algorithm, called Gradient Ascent with Shrinking Policy Prediction (GA-SPP), which augments the basic gradient ascent approach with the concept of shrinking policy prediction. The key idea behind this algorithm is that an agent adjusts its strategy in response to the forecasted strategy of the other agent, instead of its current one. GA-SPP is shown formally to have Nash convergence in larger settings than existing gradient-based multi-agent learning methods. Furthermore, unlike existing gradient-based methods, GA-SPP's theoretical guarantees do not assume the learning rate to be infinitesimal.

KEYWORDS

Multi-Agent Learning; Nash Equilibrium; Convergence; Finite Step Size

ACM Reference Format:
Xinliang Song, Tonghan Wang, and Chongjie Zhang. 2019. Convergence of Multi-Agent Learning with a Finite Step Size in General-Sum Games. In *Proc. of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), Montreal, Canada, May 13–17, 2019,* IFAAMAS, 9 pages.

1 INTRODUCTION

Multi-agent learning (MAL) is concerned with a set of agents that learn to maximize their expected rewards. There are a number of important applications that involve MAL, including competitive settings such as self-play in AlphaZero [23] and generative adversarial networks in deep learning [15, 20], cooperative settings such as when learning to communicate [13, 25] and multiplayer game [14], or some mix of the two [17, 26]. Although promising empirical results, establishing a theoretical guarantee of convergence for MAL, especially for gradient-based methods, is fundamentally challenging because of its non-stationary environment.

Recent multi-agent learning (MAL) algorithms [7, 9–11, 19, 22] with satisfactory empirical results are proposed, but most of them do not provide theoretical analyses of convergence. There are only a few worksthat provide theoretical results before them. Singh *et al.* [24] first consider the theoretical convergence of gradient-based methods in MAL. After that, several variants [1, 4, 5, 24, 27] are proposed and they provide theoretical convergence in general-sum games, but theoretical guarantees are restricted in 2-agent, 2-action games and they assume that the learning rate is infinitesimal, which is not practical. Some other online learning algorithms [8, 12, 16] have also been proposed with theoretical guarantees, but just for specific settings, such as congestion games and potential games.

In this paper, we propose a new multi-agent learning algorithm that augments a basic gradient ascent algorithm with *shrinking* policy prediction, called Gradient Ascent with Shrinking Policy Prediction (GA-SPP). The key idea behind this algorithm is that an agent adjusts its strategy in response to the forecasted strategy of the other agent, instead of its current one. This paper makes three major novelties. First, to our best knowledge, GA-SPP is the first gradient-ascent MAL algorithm with a finite learning rate that provides convergence guarantee in general-sum games. Second, GA-SPP provides convergence guarantee in larger games than existing gradient-ascent MAL algorithms, which include $m \times n$ positive semi-definite games, a class of $2 \times n$ general-sum games, and 2×2 general-sum games. Finally, GA-SPP guarantees to converge to a Nash Equilibrium when converging in any $m \times n$ general-sum game.

Although GA-SPP shares some similar ideas about using policy prediction with IGA-PP [27] and the extra-gradient method [2], it has several major differences from them. For example, apart from using a finite step size, another significant difference between GA-SPP and IGA-PP is that forecasted strategies of the opponent are projected to the valid probability space. This improvement enables GA-SPP's Nash convergence when converging, which does not hold for IGA-PP. In contrast to the extra-gradient approach, GA-SPP uses shrinking prediction lengths which can be different from the policy update rate. This improvement makes GA-SPP not only more flexible in practice but also stronger in terms of theoretical guarantees.

Like IGA-PP, we assume that agents know the other agent's strategy and its current strategy gradient, but we do not require the learning rate to be infinitesimal. Even though GA-SPP needs some restricted assumptions, it pushes forward the state of the art of MAL with theoretical analysis. We expect that our work can shed a light for theoretical understanding of dynamics and complexity of MAL problems and like IGA-PP and WoLF-IGA [6], can encourage broadly applicable multi-agent reinforcement learning algorithms. Our proposed learning algorithm also provides a different approach for computing Nash Equilibiria of subsets of larger games, other than well-established offline algorithms [18, 21], whose computation complexity increases sharply with the number of actions.

Notation

We use following notations in this paper:

Δ denotes the valid strategy space (*i.e.*, a simplex).

Proc. of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), N. Agmon, M. E. Taylor, E. Elkind, M. Veloso (eds.), May 13–17, 2019, Montreal, Canada.

$\Pi_\Delta\colon \Re^n \to \Delta$ denotes the convex projection to the valid space,

$$\Pi_\Delta[\boldsymbol{x}] = argmin_{z\in\Delta}\|\boldsymbol{x} - z\|.$$

$P_\Delta(\boldsymbol{x},\ \boldsymbol{v})$ denotes the projection of a vector $\boldsymbol{v}$ on $\boldsymbol{x} \in \Delta$,

$$P_\Delta(\boldsymbol{x},\ \boldsymbol{v}) = \lim_{\eta\to 0}\frac{\Pi_\Delta[\boldsymbol{x} + \eta\boldsymbol{v}] - \boldsymbol{x}}{\eta}.$$

$(\boldsymbol{v}_1; \boldsymbol{v}_2)$ denotes $\begin{pmatrix} \boldsymbol{v}_1 \\ \boldsymbol{v}_2 \end{pmatrix}$, where $\boldsymbol{v}_1,\ \boldsymbol{v}_2$ are column vectors.

2 GRADIENT ASCENT

We begin with a brief overview of normal-form games and then review the basic gradient ascent algorithm.

2.1 Normal-Form Games

A 2-agent, $m \times n$ -action, general-sum normal-form game is defined by a pair of matrices

$$R = \begin{bmatrix} r_{11} & \dots & r_{1n} \\ \dots & \dots & \dots \\ r_{m1} & \dots & r_{mn} \end{bmatrix} \quad \text{and} \quad C = \begin{bmatrix} c_{11} & \dots & c_{1n} \\ \dots & \dots & \dots \\ c_{m1} & \dots & c_{mn} \end{bmatrix}$$

specifying the payoffs for the row agent and the column agent, respectively. The agents simultaneously select an action from their available set, and the joint action of the agents determines their payoffs according to their payoff matrices. If the row agent selects action $a \in \{1,\ \dots,\ m\}$ and the column agent selects action $b \in \{1,\ \dots,\ n\}$, respectively, then the row agent receives a payoff r_{ab} and the column agent receives a payoff c_{ab}.

The agents can choose actions stochastically based on some probability distribution over their available actions. This distribution is said to be a mixed strategy. Let $\alpha_i \in [0,\ 1]$ denote the probability of choosing the i-th action by the row agent and $\beta_j \in [0,\ 1]$ denote the probability of choosing the j-th action by the column agent, where $i \in \{1,\ \dots,\ m-1\}$, $j \in \{1,\ \dots,\ n-1\}$, $\sum_1^{m-1}\alpha_i \le 1$, $\sum_1^{n-1}\beta_j \le 1$. We use $\boldsymbol{\Delta_1}$ to denote a m-1 dimensional simplex and $\boldsymbol{\Delta_2}$ to denote a n-1 dimensional simplex. This $(m-1)\times(n-1)$ representation is equivalent to the $m \times n$ representation, and, following the previous work on gradient-based methods, we choose the former one. Let

$$\boldsymbol{\alpha} = [\alpha_1 \dots \alpha_{m-1}]^{\mathrm{T}}, \qquad \boldsymbol{e_{m-1}} = [1 \dots 1]^{\mathrm{T}},$$
$$\boldsymbol{\beta} = [\beta_1 \dots \beta_{n-1}]^{\mathrm{T}}, \qquad \boldsymbol{e_{n-1}} = [1 \dots 1]^{\mathrm{T}},$$

where the dimension of $\boldsymbol{e_{m-1}}$ is $m-1$, the dimension of $\boldsymbol{e_{n-1}}$ is $n-1$.

Then $\boldsymbol{\alpha} \in \boldsymbol{\Delta_1},\ \boldsymbol{\beta} \in \boldsymbol{\Delta_2}$. With a joint strategy $(\boldsymbol{\alpha}, \boldsymbol{\beta})$, the row agent's and column agent's expected payoffs are

$$\begin{aligned} V_r(\boldsymbol{\alpha}, \boldsymbol{\beta}) &= (\boldsymbol{\alpha};\ 1 - \boldsymbol{e}_{\boldsymbol{m-1}}^{\mathrm{T}}\boldsymbol{\alpha})^{\mathrm{T}} R(\boldsymbol{\beta};\ 1 - \boldsymbol{e}_{\boldsymbol{n-1}}^{\mathrm{T}}\boldsymbol{\beta}), \\ V_c(\boldsymbol{\alpha}, \boldsymbol{\beta}) &= (\boldsymbol{\alpha};\ 1 - \boldsymbol{e}_{\boldsymbol{m-1}}^{\mathrm{T}}\boldsymbol{\alpha})^{\mathrm{T}} C(\boldsymbol{\beta};\ 1 - \boldsymbol{e}_{\boldsymbol{n-1}}^{\mathrm{T}}\boldsymbol{\beta}). \end{aligned} \tag{1}$$

A joint strategy $(\boldsymbol{\alpha^*},\ \boldsymbol{\beta^*})$ is called a Nash equilibrium if for any mixed strategy $\boldsymbol{\alpha}$ of the row agent, $V_r(\boldsymbol{\alpha^*},\ \boldsymbol{\beta}^*) \ge V_r(\boldsymbol{\alpha},\ \boldsymbol{\beta}^*)$, and for any mixed strategy $\boldsymbol{\beta}$ of the column agent, $V_c(\boldsymbol{\alpha^*},\ \boldsymbol{\beta}^*) \ge V_c(\boldsymbol{\alpha^*},\ \boldsymbol{\beta})$. It is well-known that every game has at least one Nash equilibrium.

2.2 Learning using Gradient Ascent in Iterated Games

In an iterated normal-form game, agents repeatedly play the same game. Each agent seeks to maximize its expected payoff in response to the strategy of the other agent. Using the basic gradient ascent algorithm, a agent can increase its expected payoff by updating its strategy with a step size along the gradient of the current strategy. The gradient is computed as the partial derivative of the agent's expected payoff with respect to its strategy:

$$\begin{aligned} \boldsymbol{\partial_\alpha} V_r(\boldsymbol{\alpha},\ \boldsymbol{\beta}) &= \frac{\partial V_r(\boldsymbol{\alpha},\ \boldsymbol{\beta})}{\boldsymbol{\partial\alpha}} = (\boldsymbol{I_{m-1}}\ \boldsymbol{e_{m-1}})R(\boldsymbol{\beta};\ 1 - \boldsymbol{e}_{\boldsymbol{n-1}}^{\mathrm{T}}\boldsymbol{\beta}), \\ \boldsymbol{\partial_\beta} V_c(\boldsymbol{\alpha},\ \boldsymbol{\beta}) &= \frac{\partial V_c(\boldsymbol{\alpha},\ \boldsymbol{\beta})}{\boldsymbol{\partial\beta}} = (\boldsymbol{I_{n-1}}\ \boldsymbol{e_{n-1}})C^{\mathrm{T}}(\boldsymbol{\alpha};\ 1 - \boldsymbol{e}_{\boldsymbol{m-1}}^{\mathrm{T}}\boldsymbol{\alpha}), \end{aligned} \tag{2}$$

where $\boldsymbol{I_{m-1}}$ is (m-1)-order identity matrix and $\boldsymbol{I_{n-1}}$ is (n-1)-order identity matrix.

If $(\boldsymbol{\alpha_k},\ \boldsymbol{\beta_k})$ are the strategies on the k-th iteration and both agents use gradient ascent, then the new strategies will be:

$$\begin{aligned} \boldsymbol{\alpha_{k+1}} &= \Pi_{\Delta_1}[\boldsymbol{\alpha_k} + \eta\boldsymbol{\partial_\alpha} V_r(\boldsymbol{\alpha_k},\ \boldsymbol{\beta_k})], \\ \boldsymbol{\beta_{k+1}} &= \Pi_{\Delta_2}[\boldsymbol{\beta_k} + \eta\boldsymbol{\partial_\beta} V_c(\boldsymbol{\alpha_k},\ \boldsymbol{\beta_k})], \end{aligned} \tag{3}$$

where η is the gradient step size. If the updates move the strategies out of the valid probability space, the function Π_Δ will project it back.

Singh *et al.* [24] analyzed the gradient ascent algorithm by examining the dynamics of the strategies in the case of an infinitesimal step size ($\lim_{\eta\to 0}$). This algorithm is called Infinitesimal Gradient Ascent (IGA). IGA cannot converge in some 2-agent 2-action zero-sum game. GIGA-WoLF and IGA-PP extended IGA and provide theoretical guarantee of Nash equilibrium in 2-agent 2-action game through similar methods. However, these algorithms require a infinitesimal step size, which is not practical. We will describe a new gradient ascent algorithm that enables the agents' strategies to converge to a Nash equilibrium with a finite step size in a larger game setting.

3 GRADIENT ASCENT WITH SHRINKING POLICY PREDICTION (GA-SPP)

As shown in Eq. 3, the gradient used by IGA to adjust the strategy is based on current strategies. Suppose that an agent can estimate the change direction of the opponent's strategy, *i.e.*, its strategy derivative, in addition to its current strategy. Then the agent can forecast the opponent's strategy and adjust its own strategy in response to the forecasted strategy. With this idea, we design a gradient ascent algorithm with shrinking policy prediction (GA-SPP). Its updating rule consists of three steps.

In Step 1, the new derivative terms with γ serve as a short-term prediction of the opponent's strategy. If the opponent's forecasted strategy is out of boundary of simplex, it will be projected back to the valid space.

In Step 2, agents update their strategies on the basis of the forecasted strategy of its opponent.

In Step 3, agents terminate or adjust their prediction lengths. If predicted strategies are equal to the current strategies, the algorithm will terminate. Step 3 can make sure GA-SPP only converges to Nash equilibrium (NE) instead of other points. Because when

$(\boldsymbol{\alpha}_{\boldsymbol{k+1}}, \boldsymbol{\beta}_{\boldsymbol{k+1}}) = (\boldsymbol{\alpha}_{\boldsymbol{k}}, \boldsymbol{\beta}_{\boldsymbol{k}})$, GA-SPP will stop, if there is no Step 3, then $(\overline{\boldsymbol{\alpha}}_{\boldsymbol{k+1}}, \overline{\boldsymbol{\beta}}_{\boldsymbol{k+1}}) \neq (\boldsymbol{\alpha}_{\boldsymbol{k}}, \boldsymbol{\beta}_{\boldsymbol{k}})$ may happen. In this situation, GA-SPP may converge to a non-NE point. We will prove this property of GA-SPP in Proposition 1.

Algorithm 1: Updating rule of GA-SPP

```
1 $\overline{\boldsymbol{\alpha}}_{\boldsymbol{k+1}} = \Pi_{\Delta_1}[\boldsymbol{\alpha}_{\boldsymbol{k}} + \gamma_k \boldsymbol{\partial}_{\boldsymbol{\alpha}} V_r(\boldsymbol{\alpha}_{\boldsymbol{k}}, \boldsymbol{\beta}_{\boldsymbol{k}})]$;
  $\overline{\boldsymbol{\beta}}_{\boldsymbol{k+1}} = \Pi_{\Delta_2}[\boldsymbol{\beta}_{\boldsymbol{k}} + \gamma_k \boldsymbol{\partial}_{\boldsymbol{\beta}} V_c(\boldsymbol{\alpha}_{\boldsymbol{k}}, \boldsymbol{\beta}_{\boldsymbol{k}})]$;
2 $\boldsymbol{\alpha}_{\boldsymbol{k+1}} = \Pi_{\Delta_1}[\boldsymbol{\alpha}_{\boldsymbol{k}} + \eta \boldsymbol{\partial}_{\boldsymbol{\alpha}} V_r(\boldsymbol{\alpha}_{\boldsymbol{k}}, \overline{\boldsymbol{\beta}}_{k+1})]$;
  $\boldsymbol{\beta}_{\boldsymbol{k+1}} = \Pi_{\Delta_2}[\boldsymbol{\beta}_{\boldsymbol{k}} + \eta \boldsymbol{\partial}_{\boldsymbol{\beta}} V_c(\overline{\boldsymbol{\alpha}}_{\boldsymbol{k+1}}, \boldsymbol{\beta}_k)]$;;
3
if $(\overline{\boldsymbol{\alpha}}_{\boldsymbol{k+1}}, \overline{\boldsymbol{\beta}}_{\boldsymbol{k+1}}) == (\boldsymbol{\alpha}_{\boldsymbol{k}}, \boldsymbol{\beta}_{\boldsymbol{k}})$ then
    terminate;
else
    if $(\boldsymbol{\alpha}_{\boldsymbol{k+1}}, \boldsymbol{\beta}_{\boldsymbol{k+1}}) == (\boldsymbol{\alpha}_{\boldsymbol{k}}, \boldsymbol{\beta}_{\boldsymbol{k}})$ & $(\overline{\boldsymbol{\alpha}}_{\boldsymbol{k+1}}, \overline{\boldsymbol{\beta}}_{\boldsymbol{k+1}}) \neq (\boldsymbol{\alpha}_{\boldsymbol{k}}, \boldsymbol{\beta}_{\boldsymbol{k}})$
     then
        $\gamma_{k+1} = \mu_k \gamma_k$, $(0 < \mu_k < 1)$, back to (1);
    else
        $\gamma_{k+1} = \gamma_k$, back to (1);
    end
end
```

The prediction length γ_k and gradient step size η will affect the convergence of the GA-SPP algorithm. With a too large prediction length, the gradient computed with the forecasted strategy will deviate too much from the gradient computed with the opponent's current strategy. As a result, the agent may adjust its strategy in the improper direction and cause their strategies to fail to converge.

Following conditions ensure that γ and η are appropriate:

Condition 1: $\gamma_0 > 0, \eta > 0$

Condition 2: $4\gamma_0^2 \delta_r \delta_c < 1$

Condition 3: $\eta, \gamma_0 < \frac{1}{\delta_r + \delta_c}$

where $\delta_r = r_{max} - r_{min}$, $\delta_c = c_{max} - c_{min}$, r_{max} and c_{max} is the maximum reward for the row and column agent, r_{min} and c_{min} is the minimum reward for the row and column agent.

Condition 3 makes sure that the theoretical guarantee of Nash convergence in the game settings analyzed in Section 4. In experiments, the algorithm can still work in some other games if we choose larger prediction length or let agents have different prediction lengths.

3.1 Analysis of GA-SPP

In this section, we will show that if agents' strategies converge by following GA-SPP, then they must converge to a Nash equilibrium, which is described by Proposition 1. Using this proposition, we will then prove the Nash convergence of GA-SPP in three classes of games: $m \times n$ positive semi-definite games, a class of $2 \times n$ general-sum games, and 2×2 general-sum games, respectively, in the following sections.

Before proving Proposition 1, we will first show that if the projected gradients of a strategy pair are zero, then this strategy must be a Nash equilibrium, which is described by Lemma 3.1. For brevity, let $\boldsymbol{\partial}_{\boldsymbol{\alpha}}$ denotes $\boldsymbol{\partial}_{\boldsymbol{\alpha}} V_r(\boldsymbol{\alpha}, \boldsymbol{\beta})$, $\boldsymbol{\partial}_{\boldsymbol{\beta}}$ denotes $\boldsymbol{\partial}_{\boldsymbol{\beta}} V_r(\boldsymbol{\alpha}, \boldsymbol{\beta})$.

Lemma 3.1. *In $(m \times n)$-action games, if the projected partial derivatives at a strategy pair $(\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*)$ are zero, that is, $P_{\Delta_1}(\boldsymbol{\alpha}^*, \boldsymbol{\partial}_{\boldsymbol{\alpha}^*}) = 0$ and $P_{\Delta_2}(\boldsymbol{\beta}^*, \boldsymbol{\partial}_{\boldsymbol{\beta}^*}) = 0$, then $(\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*)$ is a Nash equilibrium.*

Proof. Assume that $(\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*)$ is not a Nash equilibrium. Then at least one agent, for example, the column agent, can increase its expected payoff by changing its strategy unilaterally. Assume that the improved point is $(\boldsymbol{\alpha}^*, \boldsymbol{\beta})$. Because of the convexity of the strategy space $\boldsymbol{\Delta}_2$ and the linear dependence of $V_c(\boldsymbol{\alpha}, \boldsymbol{\beta})$ on $\boldsymbol{\beta}$, then, for any $\epsilon > 0$, $(\boldsymbol{\alpha}^*, (1-\epsilon)\boldsymbol{\beta}^* + \epsilon\boldsymbol{\beta})$ must also be an improved point, which implies that the projected gradient of $\boldsymbol{\beta}$ at $(\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*)$ is not zero. By contradiction, $(\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*)$ is a Nash equilibrium. □

Proposition 1. *In 2-agent, $m \times n$ games, if two agents follow GA-SPP with appropriate γ, η (satisfying Condition 1, 2, and 3) and GA-SPP converges, then $(\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*)$ is a Nash equilibrium.*

Here is a proof sketch (the detailed formal proof is described in supplementary material[1]). According to Step 3 in the algorithm 1, if the strategy pair trajectory converges at $(\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*)$, then $(\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) = (\overline{\boldsymbol{\alpha}}_{\boldsymbol{k+1}}, \overline{\boldsymbol{\beta}}_{\boldsymbol{k+1}}) = (\boldsymbol{\alpha}_{\boldsymbol{k}}, \boldsymbol{\beta}_k)$ or $(\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) = \lim_{k\to\infty}(\overline{\boldsymbol{\alpha}}_{\boldsymbol{k+1}}, \overline{\boldsymbol{\beta}}_{\boldsymbol{k+1}}) = \lim_{k\to\infty}(\boldsymbol{\alpha}_{\boldsymbol{k}}, \boldsymbol{\beta}_k)$. For both cases, we can have $\boldsymbol{\alpha}^* = \Pi_{\Delta_1}[\boldsymbol{\alpha}^* + \eta\boldsymbol{\partial}_{\boldsymbol{\alpha}^*}]$ and $\boldsymbol{\beta}^* = \Pi_{\Delta_2}[\boldsymbol{\beta}^* + \eta\boldsymbol{\partial}_{\boldsymbol{\beta}^*}]$. From here, we can show that, for any arbitrary small $\epsilon > 0$, $\boldsymbol{\alpha}^* = \Pi_{\Delta_1}[\boldsymbol{\alpha}^* + \epsilon\boldsymbol{\partial}_{\boldsymbol{\alpha}^*}]$ and $\boldsymbol{\beta}^* = \Pi_{\Delta_2}[\boldsymbol{\beta}^* + \epsilon\boldsymbol{\partial}_{\boldsymbol{\beta}^*}]$, which imply $P_{\Delta_1}(\boldsymbol{\alpha}^*, \boldsymbol{\partial}_{\boldsymbol{\alpha}^*}) = \mathbf{0}$ and $P_{\Delta_2}(\boldsymbol{\beta}^*, \boldsymbol{\partial}_{\boldsymbol{\beta}^*}) = \mathbf{0}$. Then according to Lemma 3.1, $(\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*)$ is a Nash equilibrium.

4 CONVERGENCE OF GA-SPP

We will show the Nash convergence of GA-SPP in three classes of games in this section.

4.1 $m \times n$ Positive Semi-Definite Games

A function $\Phi(\boldsymbol{v}, \boldsymbol{w})$ is called a positive semi-definite function if it obeys the inequality defined in [3]:

$$\Phi(\boldsymbol{w}, \boldsymbol{w}) - \Phi(\boldsymbol{w}, \boldsymbol{v}) - \Phi(\boldsymbol{v}, \boldsymbol{w}) + \Phi(\boldsymbol{v}, \boldsymbol{v}) \geq 0. \tag{4}$$

To facilitate the proof, we define the normalized value function for a game:

$$\Phi(\boldsymbol{v}, \boldsymbol{w}) = V_r(\boldsymbol{\alpha}^1, \boldsymbol{\beta}^2) + V_c(\boldsymbol{\alpha}^2, \boldsymbol{\beta}^1), \tag{5}$$

where $\boldsymbol{v} = (\boldsymbol{\alpha}^1, \boldsymbol{\beta}^1) \in \{\Delta_1 \times \Delta_2\}$, $\boldsymbol{w} = (\boldsymbol{\alpha}^2, \boldsymbol{\beta}^2) \in \{\Delta_1 \times \Delta_2\}$.

Definition 1. *A 2-agent $m \times n$ game is called positive semi-definite (PSD) game if its normalized value function obeys*

$$\Phi(\boldsymbol{w}, \boldsymbol{w}) - \Phi(\boldsymbol{w}, \boldsymbol{v}) - \Phi(\boldsymbol{v}, \boldsymbol{w}) + \Phi(\boldsymbol{v}, \boldsymbol{v}) \geq 0. \tag{6}$$

It means that for a PSD game, its payoff matrices satisfies

$$\begin{aligned} &V_r(\boldsymbol{\alpha}^1, \boldsymbol{\beta}^1) + V_c(\boldsymbol{\alpha}^1, \boldsymbol{\beta}^1) + V_r(\boldsymbol{\alpha}^2, \boldsymbol{\beta}^2) + V_c(\boldsymbol{\alpha}^2, \boldsymbol{\beta}^2) \\ \geq &V_r(\boldsymbol{\alpha}^1, \boldsymbol{\beta}^2) + V_c(\boldsymbol{\alpha}^1, \boldsymbol{\beta}^2) + V_r(\boldsymbol{\alpha}^2, \boldsymbol{\beta}^1) + V_c(\boldsymbol{\alpha}^2, \boldsymbol{\beta}^1) \\ &\forall \boldsymbol{\alpha}^1, \boldsymbol{\alpha}^2 \in \Delta_1, \quad \forall \boldsymbol{\beta}^1, \boldsymbol{\beta}^2 \in \Delta_2. \end{aligned} \tag{7}$$

Zero-sum games are a subset of PSD games, because their value functions satisfy $V_r(\boldsymbol{\alpha}, \boldsymbol{\beta}) + V_c(\boldsymbol{\alpha}, \boldsymbol{\beta}) = 0$, then both sides of inequality 7 are equal to zero.

[1] https://drive.google.com/file/d/1TZeRf0xp4g4wg-JX7zA9TjqC2S619pAp/view?usp=sharing

For a PSD game, if $(\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*)$ is a Nash equilibrium and $\boldsymbol{v}^* = (\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*)$, then its normalized function obeys

$$\langle \nabla_2\Phi(\boldsymbol{w}, \boldsymbol{w}), \boldsymbol{w} - \boldsymbol{v}^*\rangle \geq 0 \quad \forall \boldsymbol{w} \in \{\Delta_1 \times \Delta_2\}. \tag{8}$$

In the proof of Theorem 1, we will use this inequality.

THEOREM 1. *If, in a 2-agent, $m \times n$ iterated positive semi-definite norm-form game, both agents follow the GA-SPP algorithm (with Condition 1, 2, and 3), then their strategies will converge to a Nash equilibrium.*

PROOF. Motivated by [2], our proof will use some variational inequalities techniques.

From the first and second step of GA-SPP (Algorithm 1), we have estimates

$$\begin{aligned}
|\overline{\boldsymbol{\alpha}}_{\boldsymbol{k+1}} - \boldsymbol{\alpha}_{\boldsymbol{k+1}}| &\leq |\gamma_k \boldsymbol{\partial}_{\boldsymbol{\alpha}} V_r(\boldsymbol{\alpha}_{\boldsymbol{k}}, \boldsymbol{\beta}_{\boldsymbol{k}}) - \eta \boldsymbol{\partial}_{\boldsymbol{\alpha}} V_r(\boldsymbol{\alpha}_{\boldsymbol{k}}, \overline{\boldsymbol{\beta}}_{\boldsymbol{k+1}})|,\\
|\overline{\boldsymbol{\beta}}_{\boldsymbol{k+1}} - \boldsymbol{\beta}_{\boldsymbol{k+1}}| &\leq |\gamma_k \boldsymbol{\partial}_{\boldsymbol{\beta}} V_c(\boldsymbol{\alpha}_{\boldsymbol{k}}, \boldsymbol{\beta}_{\boldsymbol{k}}) - \eta \boldsymbol{\partial}_{\boldsymbol{\beta}} V_c(\overline{\boldsymbol{\alpha}}_{\boldsymbol{k+1}}, \boldsymbol{\beta}_{\boldsymbol{k}})|.
\end{aligned} \tag{9}$$

We present the first and second step of GA-SPP in the form of variational inequalities:

$$\begin{aligned}
&\langle \overline{\boldsymbol{\alpha}}_{\boldsymbol{k+1}} - \boldsymbol{\alpha}_{\boldsymbol{k}} - \gamma_k \boldsymbol{\partial}_{\boldsymbol{\alpha}} V_r(\boldsymbol{\alpha}_{\boldsymbol{k}}, \boldsymbol{\beta}_{\boldsymbol{k}}),\ z_1 - \overline{\boldsymbol{\alpha}}_{\boldsymbol{k+1}}\rangle \geq 0 \quad \forall z_1 \in \Delta_1,\\
&\langle \overline{\boldsymbol{\beta}}_{\boldsymbol{k+1}} - \boldsymbol{\beta}_{\boldsymbol{k}} - \gamma_k \boldsymbol{\partial}_{\boldsymbol{\beta}} V_c(\boldsymbol{\alpha}_{\boldsymbol{k}}, \boldsymbol{\beta}_{\boldsymbol{k}}),\ z_2 - \overline{\boldsymbol{\beta}}_{\boldsymbol{k+1}}\rangle \geq 0 \quad \forall z_2 \in \Delta_2;
\end{aligned} \tag{10}$$

$$\begin{aligned}
&\langle \boldsymbol{\alpha}_{\boldsymbol{k+1}} - \boldsymbol{\alpha}_{\boldsymbol{k}} - \eta \boldsymbol{\partial}_{\boldsymbol{\alpha}} V_r(\boldsymbol{\alpha}_{\boldsymbol{k}}, \overline{\boldsymbol{\beta}}_{\boldsymbol{k+1}}),\ z_1 - \boldsymbol{\alpha}_{\boldsymbol{k+1}}\rangle \geq 0 \quad \forall z_1 \in \Delta_1,\\
&\langle \boldsymbol{\beta}_{\boldsymbol{k+1}} - \boldsymbol{\beta}_{\boldsymbol{k}} - \eta \boldsymbol{\partial}_{\boldsymbol{\beta}} V_c(\overline{\boldsymbol{\alpha}}_{\boldsymbol{k+1}}, \boldsymbol{\beta}_{\boldsymbol{k}}),\ z_2 - \boldsymbol{\beta}_{\boldsymbol{k+1}}\rangle \geq 0 \quad \forall z_2 \in \Delta_2.
\end{aligned} \tag{11}$$

Let $\boldsymbol{v} = \begin{pmatrix} \boldsymbol{\alpha}^1 \\ \boldsymbol{\beta}^1 \end{pmatrix}$. Put $\boldsymbol{z_1} = \boldsymbol{\alpha}^*, z_2 = \boldsymbol{\beta}^*$ in Eq. 11, then set $\boldsymbol{z_1} = \boldsymbol{\alpha}_{\boldsymbol{k+1}}, z_2 = \boldsymbol{\beta}_{\boldsymbol{k+1}}$ in Eq. 10, and take into account of Eq. 9, we can get (the detailed computation is listed in our supplementary material)

$$\begin{aligned}
&\langle \boldsymbol{v}_{\boldsymbol{k+1}} - \boldsymbol{v}_{\boldsymbol{k}},\ \boldsymbol{v}^* - \boldsymbol{v}_{\boldsymbol{k+1}}\rangle + \langle \overline{\boldsymbol{v}}_{\boldsymbol{k+1}} - \boldsymbol{v}_{\boldsymbol{k}},\ \boldsymbol{v}_{\boldsymbol{k+1}} - \overline{\boldsymbol{v}}_{\boldsymbol{k+1}}\rangle\\
&+\eta\langle \nabla_2\Phi(\overline{\boldsymbol{v}}_{\boldsymbol{k+1}},\ \overline{\boldsymbol{v}}_{\boldsymbol{k+1}}),\ \boldsymbol{v}^* - \overline{\boldsymbol{v}}_{\boldsymbol{k+1}}\rangle\\
&+h^2\|\nabla_2\Phi(\boldsymbol{v}_{\boldsymbol{k}},\ \boldsymbol{v}_{\boldsymbol{k}}) - \nabla_2\Phi(\overline{\boldsymbol{v}}_{\boldsymbol{k+1}},\ \overline{\boldsymbol{v}}_{\boldsymbol{k+1}})\|^2 \geq 0,
\end{aligned} \tag{12}$$

where $h = max\{\gamma_0, \eta\}$. By means of identity, the first two scalar products in Eq. 12 can be rewritten as

$$\begin{aligned}
&\frac{1}{2}\|\boldsymbol{v}_{\boldsymbol{k}} - \boldsymbol{v}^*\|^2 - \frac{1}{2}\|\boldsymbol{v}_{\boldsymbol{k+1}} - \boldsymbol{v}^*\|^2 -\\
&\frac{1}{2}\|\boldsymbol{v}_{\boldsymbol{k+1}} - \overline{\boldsymbol{v}}_{\boldsymbol{k+1}}\|^2 - \frac{1}{2}\|\overline{\boldsymbol{v}}_{\boldsymbol{k+1}} - \boldsymbol{v}_{\boldsymbol{k}}\|^2.
\end{aligned} \tag{13}$$

Set $\boldsymbol{w} = \overline{\boldsymbol{v}}_{\boldsymbol{k+1}}$ in Eq. 8, then the third term in Eq.12 is non-positive. For the last term of Eq. 12, if $\nabla_2\Phi(\boldsymbol{v}_{\boldsymbol{k}},\ \boldsymbol{v}_{\boldsymbol{k}})$ satisfies the Lipschitz condition with constant L, then following estimate is correct

$$|\nabla_2\Phi(\boldsymbol{v}_{\boldsymbol{k}},\ \boldsymbol{v}_{\boldsymbol{k}}) - \nabla_2\Phi(\overline{\boldsymbol{v}}_{\boldsymbol{k+1}},\ \overline{\boldsymbol{v}}_{\boldsymbol{k+1}})| \leq L|\overline{\boldsymbol{v}}_{\boldsymbol{k+1}} - \boldsymbol{v}_{\boldsymbol{k}}|. \tag{14}$$

Now put Eq. 13 and Eq. 14 in Eq. 12, we can yield

$$\begin{aligned}
&\|\boldsymbol{v}_{\boldsymbol{k+1}} - \boldsymbol{v}^*\|^2 + \|\boldsymbol{v}_{\boldsymbol{k+1}} - \overline{\boldsymbol{v}}_{\boldsymbol{k+1}}\|^2 +\\
&(1 - 2h^2L^2)\|\overline{\boldsymbol{v}}_{\boldsymbol{k+1}} - \boldsymbol{v}_{\boldsymbol{k}}\|^2 \leq \|\boldsymbol{v}_{\boldsymbol{k}} - \boldsymbol{v}^*\|^2.
\end{aligned} \tag{15}$$

Note that $\nabla_2\Phi(\boldsymbol{v}_{\boldsymbol{k}},\ \boldsymbol{v}_{\boldsymbol{k}}) = \boldsymbol{\partial}_{\boldsymbol{\alpha}} V_r(\boldsymbol{\alpha},\ \boldsymbol{\beta}) + \boldsymbol{\partial}_{\boldsymbol{\beta}} V_c(\boldsymbol{\alpha},\ \boldsymbol{\beta})$. According to Eq. 2, $\boldsymbol{\partial}_{\boldsymbol{\alpha}} V_r(\boldsymbol{\alpha},\ \boldsymbol{\beta})$ is a function of $\boldsymbol{\beta}$, $\boldsymbol{\partial}_{\boldsymbol{\beta}} V_c(\boldsymbol{\alpha},\ \boldsymbol{\beta})$ is a function of $\boldsymbol{\alpha}$. The maximum value of 2-norm of $\boldsymbol{\partial}_{\boldsymbol{\alpha}}$ is not greater than ${\delta_r}^2/2$, and not greater than ${\delta_c}^2/2$ for 2-norm of $\boldsymbol{\partial}_{\boldsymbol{\beta}}$. So the Lipschitz constant $L \leq \frac{\delta_r}{\sqrt{2}} + \frac{\delta_c}{\sqrt{2}}$. According to Condition 3, $h = max\{\gamma_0,\ \eta\} < \frac{1}{\delta_c + \delta_r}$, so $hL < \frac{\sqrt{2}}{2}$ and $1 - 2h^2L^2 > 0$. Sum up inequality Eq. 15 from $k = 0$ to $k = K$, we get

$$\begin{aligned}
&\|\boldsymbol{v}_{\boldsymbol{K+1}} - \boldsymbol{v}^*\|^2 + \sum_{k=0}^{K}\|\boldsymbol{v}_{\boldsymbol{k+1}} - \overline{\boldsymbol{v}}_{\boldsymbol{k+1}}\|^2 +\\
&(1 - 2h^2L^2)\sum_{k=0}^{K}\|\overline{\boldsymbol{v}}_{\boldsymbol{k+1}} - \boldsymbol{v}_{\boldsymbol{k}}\|^2 \leq \|\boldsymbol{v}_{\boldsymbol{0}} - \boldsymbol{v}^*\|^2.
\end{aligned} \tag{16}$$

From the gained inequality (Eq. 16) the bound of trajectory follows

$$\|\boldsymbol{v}_{\boldsymbol{K+1}} - \boldsymbol{v}^*\|^2 \leq \|\boldsymbol{v}_{\boldsymbol{0}} - \boldsymbol{v}^*\|^2, \tag{17}$$

and the series are convergent

$$\sum_{k=0}^{K}\|\boldsymbol{v}_{\boldsymbol{k+1}} - \overline{\boldsymbol{v}}_{\boldsymbol{k+1}}\|^2 < \infty,\quad \sum_{k=0}^{K}\|\overline{\boldsymbol{v}}_{\boldsymbol{k+1}} - \boldsymbol{v}_{\boldsymbol{k}}\|^2 < \infty.$$

As a result, $\lim_{k\to\infty}\|\boldsymbol{v}_{\boldsymbol{k+1}} - \overline{\boldsymbol{v}}_{\boldsymbol{k+1}}\|^2 = 0$, $\lim_{k\to\infty}\|\overline{\boldsymbol{v}}_{\boldsymbol{k+1}} - \boldsymbol{v}_{\boldsymbol{k}}\|^2 = 0$, so $\lim_{k\to\infty}\|\boldsymbol{v}_{\boldsymbol{k+1}} - \boldsymbol{v}_{\boldsymbol{k}}\|^2 = 0$. It implies $\lim_{k\to\infty}\|\boldsymbol{\alpha}_{\boldsymbol{k+1}} - \boldsymbol{\alpha}_{\boldsymbol{k}}\|^2 = 0$ and $\lim_{k\to\infty}\|\boldsymbol{\beta}_{\boldsymbol{k+1}} - \boldsymbol{\beta}_{\boldsymbol{k}}\|^2 = 0$.

So GA-SPP can converge. With Proposition 1, GA-SPP must converge to a Nash equilibrium. Therefore, proof of Theorem 1 is completed. □

THEOREM 2. *If, in a 2-agent, $m \times n$ iterated positive semi-definite norm-form game, one agent follows the GA-SPP algorithm (with Condition 1, 2, and 3), another agent uses GA, then their strategies will converge to a Nash equilibrium.*

The proof of this theorem is omitted, which is similar to that of Theorem 1.

4.2 A Subclass of $2 \times n$ General-Sum Games

In this section, we will show that GA-SPP converges to a Nash equilibrium in a subclass of 2-agent $2\times n$ general games (Theorem 3).

A 2-agent, $2 \times n$, general-sum normal-form game's payoff matrices can be written as

$$R = \begin{bmatrix} r_{11} & \dots & r_{1n} \\ r_{21} & \dots & r_{2n} \end{bmatrix},\quad C = \begin{bmatrix} c_{11} & \dots & c_{1n} \\ c_{21} & \dots & c_{2n} \end{bmatrix}.$$

Let

$$\begin{aligned}
\boldsymbol{r_1} &= [r_{11} \dots r_{1,n-1}]^{\mathrm{T}}, & \boldsymbol{r_2} &= [r_{21} \dots r_{2,n-1}]^{\mathrm{T}},\\
\boldsymbol{c_1} &= [c_{11} \dots c_{1,n-1}]^{\mathrm{T}}, & \boldsymbol{c_2} &= [c_{21} \dots c_{2,n-1}]^{\mathrm{T}}.
\end{aligned}$$

Then agents' expected payoffs (Eq. 1) are

$$\begin{aligned}
V_r(\alpha,\ \boldsymbol{\beta}) = &(\alpha\boldsymbol{\beta}^{\mathrm{T}})\boldsymbol{r_1} + r_{1n}(\alpha(1 - \boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{e_{n-1}}))\\
&+ (1-\alpha)\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{r_2} + r_{2n}((1-\alpha)(1 - \boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{e_{n-1}})),\\
V_c(\alpha,\ \boldsymbol{\beta}) = &(\alpha\boldsymbol{\beta}^{T})\boldsymbol{c_1} + c_{1n}(\alpha(1 - \boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{e_{n-1}}))\\
&+ (1-\alpha)\boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{c_2} + c_{2n}((1-\alpha)(1 - \boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{e_{n-1}})).
\end{aligned} \tag{18}$$

The gradients (Eq. 2) can be written as

$$\begin{aligned}
\partial_\alpha V_r(\alpha, \boldsymbol{\beta}) &= \frac{\partial V_r(\alpha, \boldsymbol{\beta})}{\partial\alpha} = \boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{u_r} + b_r,\\
\boldsymbol{\partial_\beta} V_c(\alpha, \boldsymbol{\beta}) &= \frac{\partial V_c(\alpha, \boldsymbol{\beta})}{\partial\boldsymbol{\beta}} = \alpha\boldsymbol{u_c} + \boldsymbol{b_c},
\end{aligned} \tag{19}$$

where $b_r = r_{1n} - r_{2n}$, $\boldsymbol{b_c} = \boldsymbol{c}_2 - c_{2n}\boldsymbol{e}_{n-1}$, $\boldsymbol{u_r} = \boldsymbol{r}_1 - \boldsymbol{r}_2 - b_r\boldsymbol{e}_{n-1}$, and $\boldsymbol{u_c} = \boldsymbol{c}_1 - \boldsymbol{c}_2 - (c_{1n} - c_{2n})\boldsymbol{e}_{n-1}$.

THEOREM 3. *If, in a 2-agent, $2 \times n$, norm-form game, if there exists a $\delta > 0$ such that the payoff matrices obey*

$$\boldsymbol{u_r} + \delta\boldsymbol{u_c} = 0, \tag{20}$$

and both agents follow the GA-SPP algorithm (with Condition 1, 2, and 3), then their strategies will converge to a Nash equilibrium.

Proof. For a 2-agent $2\times n$ game, if we put Eq. 18 into Definition 1, then we derive $\boldsymbol{u_r} + \boldsymbol{u_c} = 0$. It shows that $2 \times n$ games in Theorem 3 with $\delta = 1$ are PSD games.

First we consider $2 \times n$ positive semi-definite games ($\boldsymbol{u_r} + \boldsymbol{u_c} = 0$). According to Theorem 1, in this particular case, GA-SPP can converge to a Nash Equilibrium. It means the following iteration can converge:

$$\begin{aligned}
\overline{\alpha}_{k+1} &= \Pi_{\Delta_1}[\alpha_k + \gamma_k(\boldsymbol{\beta}_{\boldsymbol{k}}^{\mathrm{T}}\boldsymbol{u}_{\boldsymbol{r}_1} + b_{r_1})],\\
\overline{\boldsymbol{\beta}}_{\boldsymbol{k+1}} &= \Pi_{\Delta_2}[\boldsymbol{\beta_k} - \gamma_k(\alpha_k\boldsymbol{u}_{\boldsymbol{r}_1} + \boldsymbol{b}_{\boldsymbol{c}_1})];\\
\alpha_{k+1} &= \Pi_{\Delta_1}[\alpha_k + \eta(\overline{\boldsymbol{\beta}}_{\boldsymbol{k}}^{\mathrm{T}}\boldsymbol{u}_{\boldsymbol{r}_1} + b_{r_1})],\\
\boldsymbol{\beta}_{\boldsymbol{k+1}} &= \Pi_{\Delta_2}[\boldsymbol{\beta_k} - \eta(\overline{\alpha_k}\boldsymbol{u}_{\boldsymbol{r}_1} + \boldsymbol{b}_{\boldsymbol{c}_1})].
\end{aligned} \tag{21}$$

For brevity, we omit step 3 of GA-SPP.

For a $2 \times n$, norm-form game that obeys Eq. 23, we have $\boldsymbol{u}_{\boldsymbol{r}_2} + \delta\boldsymbol{u}_{\boldsymbol{c}_2} = 0$. Let $x = \frac{\alpha}{\sqrt{\delta}}$, $\boldsymbol{y} = \sqrt{\delta}\boldsymbol{\beta}$. If α and $\boldsymbol{\beta}$ follows GA-SPP, then the update rule of x and $\boldsymbol{y}$ is

$$\begin{aligned}
\overline{x}_{k+1} &= \Pi_{\Delta_x}[x_k + \gamma_k(\boldsymbol{y}_{\boldsymbol{k}}^{\mathrm{T}}\boldsymbol{u}_{\boldsymbol{r}_2} + \frac{b_{r_2}}{\sqrt{\delta}})],\\
\overline{\boldsymbol{y}}_{\boldsymbol{k+1}} &= \Pi_{\Delta_{\boldsymbol{y}}}[\boldsymbol{y_k} - \gamma_k(\alpha_k\boldsymbol{u}_{\boldsymbol{r}_2} + \sqrt{\delta}\boldsymbol{b}_{\boldsymbol{c}_2})];\\
x_{k+1} &= \Pi_{\Delta_x}[x_k + \eta(\overline{\boldsymbol{y}}_{\boldsymbol{k}}^{\mathrm{T}}\boldsymbol{u}_{\boldsymbol{r}_2} + \frac{b_{r_2}}{\sqrt{\delta}})],\\
\boldsymbol{y}_{\boldsymbol{k+1}} &= \Pi_{\Delta_{\boldsymbol{y}}}[\boldsymbol{y_k} - \eta(\overline{x_k}\boldsymbol{u}_{\boldsymbol{r}_2} + \sqrt{\delta}\boldsymbol{b}_{\boldsymbol{c}_2})].
\end{aligned} \tag{22}$$

Comparing Eq. 22 with Eq. 21, $(x, \boldsymbol{y})$ can be viewed as a strategy pair of another $2 \times n$ PSD game following GA-SPP. Notice that the proof of Theorem 1 only requires that the valid space is a bounded convex set. Therefore, if $(\alpha, \boldsymbol{\beta})$ follows GA-SPP, $(x, \boldsymbol{y})$ can converge, then $(x,\ \boldsymbol{y})$ can still converge in $2 \times n$, norm-form game can converge.

With Proposition 1, we finish the proof of Theorem 3. □

THEOREM 4. *If, in a 2-agent, $2 \times n$, norm-form game, if there exists $\delta > 0$, and the payoff matrices obey*

$$\boldsymbol{u_r} + \delta\boldsymbol{u_c} = 0, \tag{23}$$

and one agent follow the GA-SPP algorithm (with Condition 1, 2, and 3), another agent uses GA, then their strategies will converge to a Nash equilibrium.

The proof of this theorem is omitted, which is similar to that of Theorem 3.

4.3 2 × 2 General-Sum Games

In this section, we will prove the Nash convergence of GA-SPP in 2×2 general-sum games.

THEOREM 5. *If, in a 2-agent, 2×2, iterated general-sum game, both agents follow the GA-SPP algorithm (with Condition 1, 2, and 3), then their strategies will converge to a Nash equilibrium.*

Proof. With Proposition 1, in order to prove Theorem 5, we just need to prove the convergence of GA-SPP in 2×2 games, which is accomplished by Lemma 4.1, 4.2, and 4.3. □

Next, we will analyze the structure of 2×2 games firstly, and then show the convergence in different cases respectively.

In a 2-agent, 2-action game, the reward functions (Eq. 1) can be written as

$$\begin{aligned}
V_r(\alpha,\ \beta) &= r_{11}(\alpha\beta) + r_{12}(\alpha(1-\beta)) + r_{21}((1-\alpha)\beta)\\
&\quad + r_{22}((1-\alpha)(1-\beta)),\\
V_c(\alpha,\ \beta) &= c_{11}(\alpha\beta) + c_{12}(\alpha(1-\beta)) + c_{21}((1-\alpha)\beta)\\
&\quad + c_{22}((1-\alpha)(1-\beta)).
\end{aligned}$$

And the gradient function (Eq. 2) can be written as

$$\begin{aligned}
\partial_\alpha V_r(\alpha,\ \beta) &= \frac{\partial V_r(\alpha,\ \beta)}{\partial\alpha} = u_r\beta + b_r,\\
\partial_\beta V_c(\alpha,\ \beta) &= \frac{\partial V_c(\alpha,\ \beta)}{\partial\beta} = u_c\alpha + b_c,
\end{aligned}$$

where $u_r = r_{11}+r_{22}-r_{12}-r_{21}$, $b_r = r_{12}-r_{22}$, $u_c = c_{11}+c_{22}-c_{12}-c_{21}$, and $b_c = c_{21} - c_{22}$. We have $|u_r| \le 2\delta_r,\ |u_c| \le 2\delta_c$.

We can formulate the first two update rules of GA-SPP (1):

$$\begin{aligned}
\alpha_{k+1} &= \Pi_\Delta[\alpha_k + \eta\partial_{\alpha_k}V_r(\alpha_k,\ \Pi_\Delta[\beta_k + \gamma\partial_{\beta_k},\ \beta_k])],\\
\beta_{k+1} &= \Pi_\Delta[\beta_k +\ \eta\partial_{\beta_k}V_c(\beta_k,\ \Pi_\Delta[\alpha_k + \gamma\partial_{\alpha_k},\ \alpha_k])],
\end{aligned} \tag{24}$$

where $\Delta = \Delta_1 = \Delta_2 = [0,\ 1]$.

To prove the Nash convergence of GA-SPP, we will examine the dynamics of the strategy pair following GA-SPP. In a 2-agent, 2-action, general-sum game, $(\alpha,\ \beta)$ can be viewed as a point in $\mathbb{R}^2$ constrained to lie in the unit space.

According to Eq. 24, if $(\alpha_k,\ \beta_k)$ is a unconstrained point, then value of $(\alpha_{k+1},\ \beta_{k+1})$ is

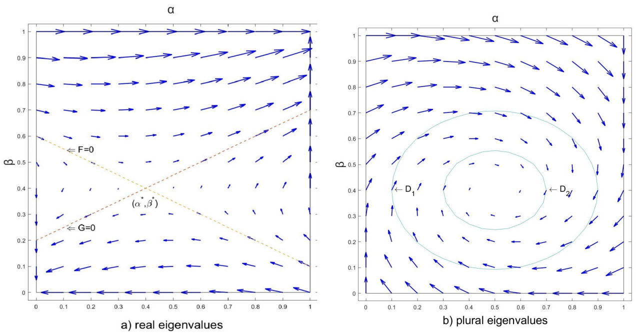
$$\begin{aligned}
\begin{bmatrix}\alpha_{k+1}\\ \beta_{k+1}\end{bmatrix} - \begin{bmatrix}\alpha_k\\ \beta_k\end{bmatrix} &= \eta\begin{bmatrix}\gamma_k u_r u_c & u_r\\ u_c & \gamma_k u_r u_c\end{bmatrix}\begin{bmatrix}\alpha_k\\ \beta_k\end{bmatrix}\\
&\quad +\eta\begin{bmatrix}\gamma_k u_r b_c + b_r\\ \gamma_k u_c b_r + b_c\end{bmatrix}.
\end{aligned} \tag{25}$$

We denote the 2×2 matrix in Eq. 25 as U. If the matrix U is invertible, in the unconstrained condition, there exists and only exists one point so that the left hand side of Eq. 25 is zero. We call this point the center (or origin) and denote it as $(\alpha^c,\ \beta^c)$. The eigenvalues of U is given by

$$\lambda_1 = \gamma_k u_r u_c + \sqrt{u_r u_c} \text{ and } \lambda_2 = \gamma_k u_r u_c - \sqrt{u_r u_c}. \tag{26}$$

According to Condition 2 ($4\gamma_k^2\delta_r\delta_c < 1$) and $|u_r| \le 2\delta_r,\ |u_c| \le 2\delta_c$, then $\gamma_k^2 u_r u_c < 1$. There are three cases of U:

- Case 1: $u_r u_c = 0$, *i.e.*, U is not invertible;
- Case 2: $u_r u_c < 0$, *i.e.*, having two imaginary conjugate eigenvalues with negative real;
- Case 3: $u_r u_c > 0$, *i.e.*, having two real eigenvalues.



a) real eigenvalues

b) plural eigenvalues

Figure 1: Strategy updating directions of the GA-SPP. a) when U has real eigenvalues and b) when U has imaginary eigenvalues with negative real part.

To prove Theorem 5, we only need to show that GA-SPP always leads the strategy pair to converge in these three cases.

Lemma 4.1. *If, in a 2-agent, 2-action, iterated general-sum game, U is not invertible, for any initial strategy pair, GA-SPP leads the strategy pair trajectory to converge to a Nash equilibrium (NE) with finite steps.*

Proof. From Eq. 26, if U is not invertible, then $u_r u_c = 0$. Assume $u_c = 0$ (the analysis for the case $u_r = 0$ is analogous and thus omitted for brevity).

According to Eq. 25, $\beta_{k+1} = \Pi_\Delta[\beta_k + \eta b_c]$. Because ηb_c is constant and $\beta \in [0,\ 1]$, strategy β will no longer change after finite steps. We denote this value by β^*. Then, $\alpha_{k+1} = \Pi_\Delta[\alpha_k + \eta(u_r \beta^* + b_r)]$. Because $\eta(u_r\beta^* + b_r)$ is a constant and $\alpha \in [0,\ 1]$, after a certain number (finite) of steps, strategy α will also stop changing. We denote this value by α^*. So in this case, the strategy pair will converge to $(\alpha^*,\ \beta^*)$, and with Proposition 1, this is a Nash equilibrium.

Note that the index of η_k was omitted in the proof, which is because the situation of $(\alpha_{k+1},\ \beta_{k+1}) = (\alpha_k,\ \beta_k)$ & $(\overline{\alpha}_{k+1},\ \overline{\beta}_{k+1}) \neq (\alpha_k,\ \beta_k)$ did not occur in this case. Lemma 4.3 also has this property. □

Lemma 4.2. *If, in a 2-agent, 2-action, iterated general-sum game, U has two imaginary conjugate eigenvalues with negative real, for any initial strategy pair, GA-SPP leads the strategy pair trajectory to converge to a NE.*

Proof. Since $u_r u_c < 0$, there exists a δ such that $u_r + \delta u_c = 0$. This is a 2-dimensional situation of Theorem 3. So GA-SPP can converge to a NE. □

In the rest of this section, we will introduce Lemma 4.3 and the basic idea of proof. For the detailed mathematical proof, we refer readers to the supplementary material.

Lemma 4.3. *If, in a 2-agent, 2-action, iterated general-sum game, U has real eigenvalues, for any initial strategy pair, GA-SPP leads the strategy pair trajectory to converge to a point that is a NE.*

Before proof, we first introduce some variables to simplify the expressions.

If U is invertible, then $u_r u_c \neq 0$. Let $x = \alpha + \frac{b_c}{u_c}$, $y = \beta + \frac{b_r}{u_r}$, Eq. 25 can be reformulated as:

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \end{bmatrix} - \begin{bmatrix} x_k \\ y_k \end{bmatrix} = \eta \begin{bmatrix} \gamma u_r u_c & u_r \\ u_c & \gamma u_r u_c \end{bmatrix} \begin{bmatrix} x_k \\ y_k \end{bmatrix} \tag{27}$$

By setting the left hand side of Eq. 27 to zero, we can get an equation, the only solution of which is $x = 0$, $y = 0$.

Now we give the proof of Lemma 4.3.

Proof. From Eq. 26, real eigenvalues imply $u_r u_c > 0$. So without the loss of generality, we assume that $u_r > 0$ and $u_c > 0$ (the analysis for the case $u_r < 0$ and $u_c < 0$ is analogous and thus omitted).

Proof of Lemma 4.3 depends on the location of $(\alpha^c,\ \beta^c)$, which has three possibilities:

(1) both α^c and β^c are in the valid probability range [0,1],
(2) only one of α^c, β^c is in the valid probability range [0,1],
(3) neither α^c nor β^c is in the valid probability range [0,1].

Proofs of convergence in these three cases are given in Property 4.1, 4.2, and 4.3, respectively. □

Notice that U has two real eigenvalues: $\lambda_1 > 0$ and $\lambda_2 < 0$ and two nonparallel eigenvectors. The central point $(x = 0,\ y = 0)$ with two eigenvectors ($\boldsymbol{v_1} = [\sqrt{u_r},\ \sqrt{u_c}]$, $\boldsymbol{v_2} = [\sqrt{u_r},\ -\sqrt{u_c}]$) can form a new 2D coordinate system. The basic idea of proof is to analyze coordinates of the strategy pair update trajectory in the new coordinate system. To be brief, we introduce two functions to compute it instead of converting coordinate.

$$F = x + \sqrt{\frac{u_r}{u_c}}y, \quad G = x - \sqrt{\frac{u_r}{u_c}}y. \tag{28}$$

Property 4.1. *If U has real eigenvalues, both of α^c, β^c are in the valid probability range([0, 1]), GA-SPP leads the strategy pair trajectory to converge to a NE.*

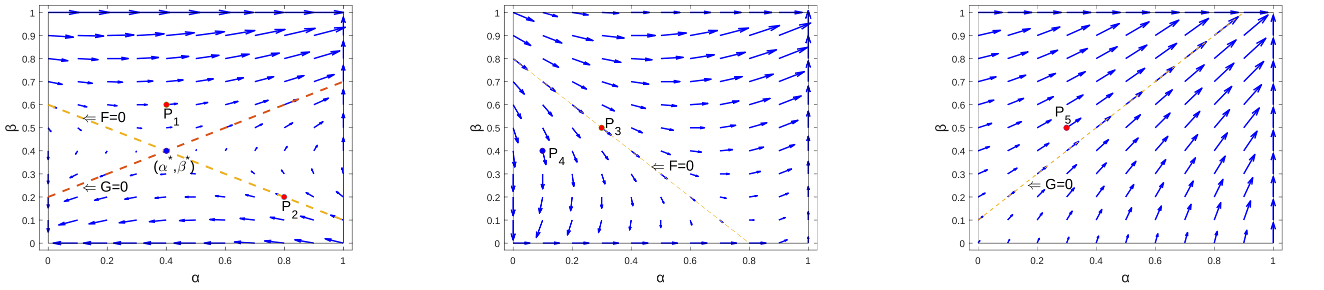
Proof. As shown in Fig. 2(a), the initial point will affect the Nash convergence result because there are three Nash equilibrium points.

The first case is $F_0 = 0$. Then the strategy pair point will keep staying in the line $F = 0$ while the absolute value of G decreases, which means that the point moves to the center point $(F = 0, G = 0)$, *i.e.*, $(\alpha^*,\ \beta^*)$. For example, if P_2 is the initial point, the point will travel along the line $F = 0$ and moves to $(F = 0, G = 0)$. We can compute F_{k+1} and G_{k+1} by

$$G_{k+1} = (1 + \eta\lambda_2)G_k, \qquad F_{k+1} = (1 + \eta\lambda_1)F_k. \tag{29}$$

According to Condition 1, 2 and 3, $0 < (1 + \eta\lambda_2) < 1$, so the GA-SPP will converge to $(F = 0,\ G = 0)$, *i.e.*, $(x = 0,\ y = 0)$.

Another case is when $F_0 > 0$, from Fig. 2(a), we can tell that the strategy pair point first touches the boundary x_{max} ($\alpha = 1$) or y_{max} ($\beta = 1$) after finite iteration steps, after then it travels along the boundary and moves to $(x_{max},\ y_{max})$. For example, if P_1 is the initial point, the point will touch the boundary $\alpha = 1$ (*i.e.*, x_{max}), then it travels along $\alpha = 1$ and moves to $(x_{max},\ y_{max})$. Without no more than one exceptional case, we can derive $F_{k+1} > F_k$ in each iteration. From the monotone bounded theorem, the GA-SPP will converge to F_{max}, *i.e.*, $(x_{max},\ y_{max})$.



(a) (b) (c)

Figure 2: Updating directions of strategy pair in Lemma 4.3. (a) Both α^c and β^c are in the valid probability range, $F(P_1) > 0$, $F(P_2) = 0$; (b) only one of α^c and β^c is in the valid range, $F(P_3) = 0$, $F(P_4) < 0$; (c) neither α^c and β^c is in the valid range.

Situation when $F_0 < 0$ is similar to that when $F_0 > 0$, so we omit it for brevity.

In all, GA-SPP can converge for any initial strategy pair. □

PROPERTY 4.2. *When* U *has real eigenvalues and only one of α^c and β^c is in the valid probability range, GA-SPP leads the strategy pair trajectory to converge to a NE.*

Proof of Property 4.2 can be classified into 4 cases. Without loss of generality, we just consider one of them, where $\beta^c < 0$ and $\alpha^c \in [0, 1]$. As shown in Fig. 2(b), we can also divide the proof into 3 parts: $F_0 > 0$, $F_0 < 0$, and $F_0 = 0$. If the point is in the part $F > 0$, according to Property 4.1, the algorithm will converge to (x_{max}, y_{max}). If $F_0 \leq 0$, we can see that the point will first touch a boundary of the valid probability space, after that it will move into the part $F > 0$. For example, if P_3 $(F = 0)$ is the initial point, the point will travel along line $F = 0$ until it hits the boundary, then it will be projected to the subspace where $F > 0$. If P_4 $(F < 0)$ is the initial point, it will touch the boundary $\beta = 0$ (y_{min}) and then y remains y_{min} while x increases until it move into the subspace where $F > 0$.

PROPERTY 4.3. *If* U *has real eigenvalues and neither α^c nor β^c is in the valid probability range [0, 1], GA-SPP leads the strategy pair trajectory to converge to a NE.*

Updating directions of strategy pair is shown in Fig. 2(c) for this case. For the detailed proof, please refer to supplementary material.

THEOREM 6. *If, in a 2-agent, 2 × 2, iterated general-sum game, one agent follow the GA-SPP algorithm (with Condition 1, 2, and 3), another agent uses GA, then their strategies will converge to a NE*

The proof is omitted, which is similar to that of Theorem 5.

5 EXPERIMENTAL ANALYSIS IN NORMAL-FORM GAMES

In this section, we will illustrate GA-SPP in games with experiments and compare GA-SPP with IGA-PP and GIGA-WoLF, both of which have theoretical guarantees, in some larger games.

5.1 Benchmark games

We first illustrate the results of GA-SPP on four representative benchmark games presented in Tab. 1. GA-SPP converges to NE in all of these games (Fig. 3).

Table 1: Benchmark games

(a) Prisoners' Dilemma

	Silent	Betray
Silent	(-1,-1)	(-3,0)
Betray	(0,-3)	(-2,-2)

(b) Chicken

	Swerve	Straight
Swerve	(-2,-2)	(1,-1)
Straight	(-1,1)	(-1,-1)

(c) Battle of Sexes

	Opera	Football
Opera	(3,2)	(1,1)
Football	(0,0)	(2,3)

(d) Rock-Paper-Scissors

	R	P	S
R	(0,0)	(-1,1)	(1,-1)
P	(1,-1)	(0,0)	(-1,1)
S	(-1,1)	(1,-1)	(0,0)

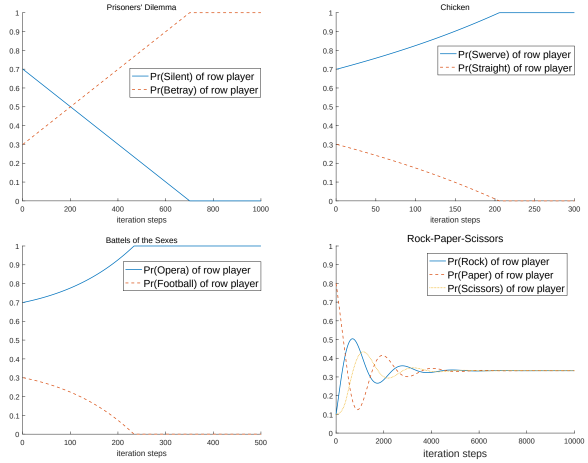


Figure 3: Action probabilities of row agent following GA-SPP in four benchmark games. Parameters: $\eta = 0.001$, $\gamma = 0.1$. **Initial polices:** $(0.7, 0.3)$ **and** $(0.3, 0.7)$**.**

5.2 Games beyond theoretical settings

We also evaluate GA-SPP in *Shapley's game* and a 2 × 3 game, presented in Tab. 2. Although the theoretical analyses of GA-SPP have not covered these games, empirical results show that it still converge. We now compare GA-SPP, GIGA-WoLF, and IGA-PP in these two games.

Fig. 4 shows the row player's action probabilities over time if both players follow GA-SPP, GIGA-WoLF, or IGA-PP in *Shapley's game* respectively. GIGA-WoLF fails to converge in this non-zero sum game, but GA-SPP and IGA-PP can converge to a Nash equilibrium.

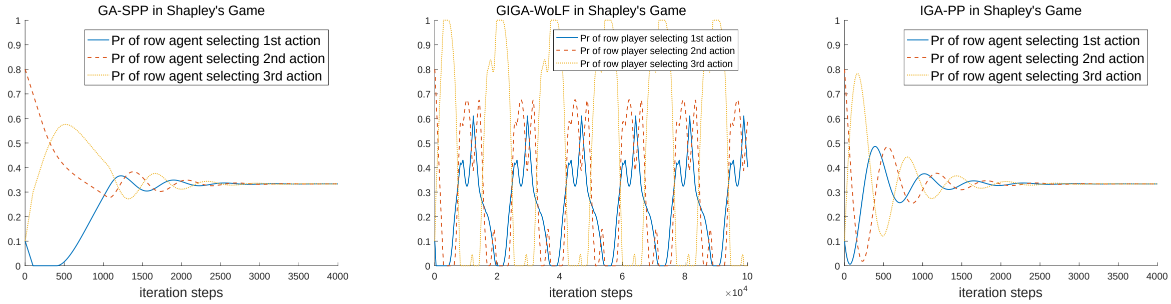


Figure 4: Comparison between GA-SPP, GIGA-WoLF, and IGA-PP in *Shapley's game*. GIGA-WoLF cannot converge while GA-SPP and IGA-PP converge to NE. GA-SPP has a slighter oscillation. Parameters: $\gamma = 3, \ \eta = 0.001$. Initial polices: $(0.1, \ 0.8, \ 0.1)$ and $(0.8, \ 0.1, \ 0.1)$.

Table 2: Games with larger settings

(a) Shapley's Game

	C1	C2	C3
R1	(0,0)	(1,0)	(0,1)
R2	(0,1)	(0,0)	(1,0)
R3	(1,0)	(0,1)	(0,0)

(b) A 2x3 Game

	C1	C2	C3
R1	(3,3)	(0,5)	(1,-2)
R2	(2,2)	(1,1)	(-1,0)

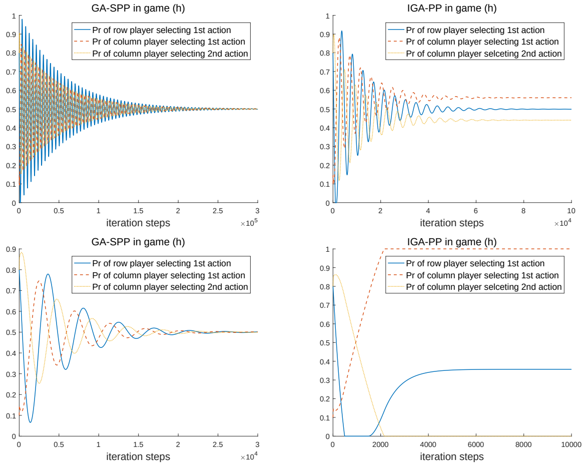


Figure 5: Comparison between GA-SPP, and IGA-PP in a 2×3 game under different prediction lengths. IGA-PP's convergence to Nash Equilibrium is affected by prediction length, while GA-SPP can always converge to NE. Parameters: $\eta = 0.001$, $\gamma = 0.01$ in upper and $\gamma = 0.1$ in lower. Initial polices: $(0.8, \ 0.2)$ and $(0.1, \ 0.8, \ 0.1)$.

Fig. 5 shows results of GA-SPP and IGA-PP in a 2×3 game under different prediction lengths. Although IGA-PP can converge, it does not converge to a Nash equilibrium. On the contrary, the strategies lead by GA-SPP successfully converge to Nash equilibrium under different prediction lengths. The essential reason is that GA-SPP projects the predicted strategies to a valid space at every step.

By examining with different learning rates, we observe that GA-SPP often converges faster than GIGA-WoLF. A possible explanation is introduced in [27]. We do not show these results for sake of space.

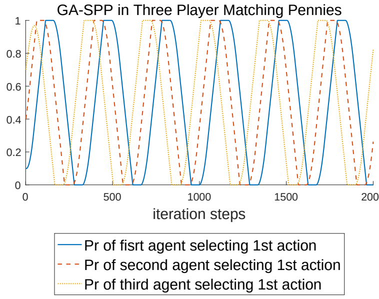


Figure 6: **Following GA-SPP, action probabilities of agents fail to converge in three player matching pennies. Parameters: $\eta = 0.001$, $\gamma = 0.3$. Initial polices: $(0.1, \ 0.9)$, $(0.4, \ 0.6)$ and $(0.7, \ 0.3)$.**

5.3 Problem games

Although GA-SPP has better performance than other MAL algorithms, the convergence of GA-SPP is not perfect. As shown in Fig. 6, in the *three player matching pennies*, GA-SPP cannot converge with a constant prediction length.

This failed case show the difficulties of MAL work and indicate that gradient method may not be the ideal way to handle a complex game. Because dynamic of gradient method in such game is not linearly, the chaotic phenomenon will occur. We may need different approaches to deal with such problems. In order to make MARL work effectively in more cases, it is important to analyze and solve these problems.

6 CONCLUSION

This paper introduced a new gradient-based multi-agent learning algorithm, called gradient-ascent with shrinking policy prediction (GA-SPP). We proved Nash convergence of GA-SPP with a finite step size in three classes of general-sum games: $m \times n$ positive semi-definite games, a subclass of $2 \times n$ general-sum games, and 2×2 general-sum games, respectively, which provide a stronger theoretical guarantee than existing gradient-based MAL algorithms. We also empirically verified the strong convergence property of GA-SPP with example games. In future work, we aim to relax assumptions of GA-SPP and extend it to stochastic games where each agent only has observations of their in-game payoffs and has no gradient information about other agents.

REFERENCES

[1] Sherief Abdallah and Victor R. Lesser. 2008. A multiagent reinforcement learning algorithm with non-linear dynamics. *Journal of Artificial Intelligence Research* 33, 1 (2008), 521–549.

[2] Anatoly Antipin. 2003. *Extragradient approach to solution of two person non-zero sum games.* Optimization And Optimal Control. 1–28 pages.

[3] A. S. Antipin. 1995. The convergence of proximal methods to fixed points of extremal mappings and estimates of their rate of convergence. *Computational Mathematics and Mathematical Physics* 35, 5 (1995), 539–551.

[4] Bikramjit Banerjee and Jing Peng. 2007. Generalized multiagent learning with performance bound. *Autonomous Agents and Multi-Agent Systems* 15, 3 (2007), 281–312.

[5] Michael Bowling. 2005. Convergence and no-regret in multiagent learning. In *Advances in neural information processing systems*. 209–216.

[6] Michael Bowling and Manuela Veloso. 2001. Convergence of gradient dynamics with a variable learning rate. In *ICML*. 27–34.

[7] Branislav BoÅąanskÃ¡, Viliam LisÃ¡, Marc Lanctot, JiÅŹÃŋ ÄŇermÃąk, and Mark H.M. Winands. 2016. Algorithms for computing strategies in two-player simultaneous move games. *Artificial Intelligence* 237 (2016), 1–40.

[8] Johanne Cohen, Amélie Héliou, and Panayotis Mertikopoulos. 2017. Learning with bandit feedback in potential games. In *Proceedings of the 31th International Conference on Neural Information Processing Systems*.

[9] Jacob W. Crandall. 2014. Towards minimizing disappointment in repeated games. *Journal of Artificial Intelligence Research* 49, 1 (2014), 111–142.

[10] Jacob W. Crandall and Michael A. Goodrich. 2011. Learning to compete, coordinate, and cooperate in repeated games using reinforcement learning. *Machine Learning* 82, 3 (2011), 281–314.

[11] Steven Damer and Maria L. Gini. 2017. Safely Using Predictions in General-Sum Normal Form Games. *adaptive agents and multi agents systems* (2017), 924–932.

[12] Constantinos Daskalakis, Alan Deckelbaum, and Anthony Kim. 2011. Near-optimal no-regret algorithms for zero-sum games. In *Proceedings of the twenty-second annual ACM-SIAM symposium on Discrete Algorithms*. Society for Industrial and Applied Mathematics, 235–254.

[13] Jakob Foerster, Ioannis Alexandros Assael, Nando de Freitas, and Shimon Whiteson. 2016. Learning to communicate with deep multi-agent reinforcement learning. In *Advances in Neural Information Processing Systems*. 2137–2145.

[14] Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. 2017. Counterfactual multi-agent policy gradients. *arXiv preprint arXiv:1705.08926* (2017).

[15] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in neural information processing systems*. 2672–2680.

[16] Walid Krichene, Benjamin Drighès, and Alexandre M Bayen. 2015. Online learning of nash equilibria in congestion games. *SIAM Journal on Control and Optimization* 53, 2 (2015), 1056–1081.

[17] Joel Z Leibo, Vinicius Zambaldi, Marc Lanctot, Janusz Marecki, and Thore Graepel. 2017. Multi-agent reinforcement learning in sequential social dilemmas. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*. International Foundation for Autonomous Agents and Multiagent Systems, 464–473.

[18] C. E. Lemke and J. T. Howson. 1964. Equilibrium Points of Bimatrix Games. *J. Soc. Indust. Appl. Math.* 12, 2 (1964), 413–423.

[19] Reshef Meir, Maria Polukarov, Jeffrey S. Rosenschein, and Nicholas R. Jennings. 2017. Iterative voting and acyclic games. *Artificial Intelligence* 252 (2017), 100–122.

[20] Luke Metz, Ben Poole, David Pfau, and Jascha Sohl-Dickstein. 2016. Unrolled Generative Adversarial Networks. *CoRR* abs/1611.02163 (2016). arXiv:1611.02163 http://arxiv.org/abs/1611.02163

[21] Ryan Porter, Eugene Nudelman, and Yoav Shoham. 2004. Simple search methods for finding a Nash equilibrium. In *National Conference on Artifical Intelligence*.

[22] H. L. Prasad, L A Prashanth, and Shalabh Bhatnagar. 2015. Two-Timescale Algorithms for Learning Nash Equilibria in General-Sum Stochastic Games. *adaptive agents and multi-agents systems* (2015), 1371–1379.

[23] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. 2017. Mastering chess and shogi by self-play with a general reinforcement learning algorithm. *arXiv preprint arXiv:1712.01815* (2017).

[24] Satinder P. Singh, Michael J. Kearns, and Yishay Mansour. 2000. Nash Convergence of Gradient Dynamics in General-Sum Games. In *UAI '00: Proceedings of the 16th Conference in Uncertainty in Artificial Intelligence, Stanford University, Stanford, California, USA, June 30 - July 3, 2000*. 541–548.

[25] Sainbayar Sukhbaatar, Rob Fergus, et al. 2016. Learning multiagent communication with backpropagation. In *Advances in Neural Information Processing Systems*. 2244–2252.

[26] Ardi Tampuu, Tambet Matiisen, Dorian Kodelja, Ilya Kuzovkin, Kristjan Korjus, Juhan Aru, Jaan Aru, and Raul Vicente. 2017. Multiagent cooperation and competition with deep reinforcement learning. *PloS one* 12, 4 (2017), e0172395.

[27] Chongjie Zhang and Victor R Lesser. 2010. Multi-Agent Learning with Policy Prediction.. In *AAAI*.