2023 8th International Conference on
Advanced Robotics and Mechatronics
July 8-10, Sanya, China

Nash Equilibrium Solution based on Safety-guarding Reinforcement Learning in Nonzero-sum Game

Junkai Tan, Shuangsi Xue, Hui Cao and Huan Li

Abstract— **In this paper, a safety-guarding controller is introduced to keep the safety of exploration in constrained state space. The controller is utilized to obtain the nonzero-sum game Nash equilibrium solution via a model-based reinforcement learning architecture. To deal with the uncertainty of persistent excitation, a concurrent learning approach is applied and both historical and transient data are employed in the learning process. In order to reduce the computational load, a single-critic network is utilized for approximation. To demonstrate the effectiveness of the proposed method, a two-player nonzero-sum game is developed, toward both convex/non-convex safe state-space constraints.**

I. Introduction

In recent years, reinforcement learning (RL) has gained popularity as a means of solving complex control problems in a variety of fields, including robotics, manufacturing, and aerospace[1], [2], [3], [4]. The importance of ensuring safety during the learning process is now widely recognized, and much research has focused on safety-aware control design, including Barrier Transformation [5], Reward-barrier functions [6], Lyapunov-like control barrier functions, and compensating controller [7]. A state-constrained RL method is proposed to ensure safety in discrete time systems [8]. In [9], a method is proposed for safe control under sensor and actuator attacks. Greene [10] improved the result of work [11] by sampling the Bellman error and using sparse neural networks for training, which reduces computational pressure.

Safety is a crucial aspect of N-player games, where multiple players interact with each other, pursuing a Nash equilibrium while ensuring that control remains within the safety limit. In [12], a barrier transformation architecture is proposed to guarantee the asymmetric safety limit. An online actor-critic algorithm is developed for N-player nonzero sum games that involve one single value function and N-actor controller. [13]. The work of [14] proposed a robust deep neural network (DNN) with an identifier for developing a controller and approximating value function. The result in [15] proposed an ACI architecture for online learning in games, which relaxed the traditional persistent excitation requirement. In [16], an online policy iteration method is developed to simultaneously evaluate two players in the nonlinear zero-sum game. An off-policy method is proposed for discrete-time systems to achieve state- and input-constrained control in fully cooperative games in [17]. A new approach to H∞ optimal control using a single critic network is developed in [18], which approximates the value, control, and disturbance strategies. However, excitation risk is not always considered in N-player games.

For multi-player systems, concurrent learning is a popular method for solving persistent excitation problems. The result in [19] proposed a method that uses historical and synchronous data effectively for the learning process of controllers. In [20], an online optimal control method is proposed that depends only on non-strictly excitation conditions. The method from [21] combines RL and experience replay to achieve better control performance compared to the previous result [22]. A model-based method is designed to improve the efficiency of the computation in [23]. The result in [24] shows that a method based on DNN and concurrent learning solves the optimal tracking problem efficiently. However, safety is ignored in the above research.

Motivated by the above discussion, this paper introduces a safety-guarding controller to ensure safety during exploration in constrained state space. The controller is used to obtain a feedback equilibrium solution for the nonzero-sum game through a system identification-based RL architecture. To address the uncertainty of persistent excitation, the concurrent learning method is applied, which uses both historical and instantaneous data in the learning process. To relax the computation load, a single-critic network is utilized for approximation. To demonstrate the effectiveness of the proposed method, a two-player nonzero-sum game is developed that addresses both convex and non-convex safe state space constraints.

This paper is organized as follows. Section II illustrates the basic setup for the N-player game and barrier function. Section III outlines the principles for the safety-guarding controller design. Section IV developed a model-based reinforcement learning structure to implement the online approximation. Section V presented the Lyapunov stability analysis for the overall system. Section VI verifies the effectiveness of the proposed method. Section VII concludes the work of this paper.

II. Preliminaries and Problem Formulation

A. Multi-player Nonzero-sum Games

Consider the N-player continuous-time affine system with nonlinear dynamics

$$\dot{x} = f(x) + \sum_{i=1}^{N} g_i(x)u_i \tag{1}$$

Junkai Tan, Shuangsi Xue, Hui Cao and Huan Li are with the Shaanxi Key Laboratory of Smart Grid, Xi'an Jiaotong University, Xi'an 710049, China, and also with the State Key Laboratory of Electrical Insulation and Power Equipment, School of Electrical Engineering, Xi'an Jiaotong University, Xi'an 710049, China. (e-mails: 15958024@stu.xjtu.edu.cn; xssxjtu@stu.xjtu.edu.cn; huicao@mail.xjtu.edu.cn; lh2000dami@stu.xjtu.edu.cn)

2023 International Conference on Advanced Robotics and Mechatronics (ICARM) | 979-8-3503-0017-8/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICARM58088.2023.10218910

where $x(t) = [x_1, x_2, ..., x_N] \in \mathbb{R}^n$, $u_i(t) \in \mathbb{R}^{m_j}$ and $g_i(x) \in \mathbb{R}^{n \times m_j}$, $f(x) \in \mathbb{R}^n$ is the drifting nonlinear dynamics. Assume $f(0) = 0$, $f(x)$ is local Lipschitz. Let $U = [u_1, u_2, ..., u_N]$ be the tuple of admissible control.

Then, We denote the cost function for ith-player as V_i, which in the form of

$$V_i(x(0), u_1, u_2, ..., u_N) = \int_t^\infty r_i(x(t), u_1, u_2, ..., u_N)dt = \int_t^\infty (Q_i(x) + \sum_j^N u_j^T R_{ij} u_j)dt \quad (2)$$

where $r_i \in \mathbb{R}_{\geq 0}$ is the instantaneous reward function, defined as $r(x(t), u_1, u_2, ..., u_N) = Q_i(x) + \sum_i^N u_j^T R_{ij} u_j$. The objective of the nonzero-sum game is to find a tuple of Nash equilibrium solutions $U^* = [u_1^*, u_2^*, ..., u_N^*]$, which minimizes the overall value function, the corresponding value function can be expressed as

$$V_i^*(x(0), u_1^*, u_2^*, ..., u_N^*) = \min_{u_i} \int_t^\infty r_i(x(t), u_1^*, u_2^*, ..., u_N^*)dt \quad (3)$$

The corresponding control can be expressed as

$$u_i^* = \underset{u_i}{argmin} V_i \quad (4)$$

To obtain the analytical solution, we differentiate the value function V^*, which results in the Hamilton-Jacobi equation in the form of

$$0 = r_i(x(t), u_1, ..., u_N) + (\Delta V_i^{*T}(f(x) + \sum_{j=1}^N g_j(x) u_i) \quad (5)$$

According to optimal control theory[15], Nash equilibrium control solutions $U^* = [u_1^*, u_2^*, ..., u_N^*]$ can be expressed as

$$u_i^* = -\frac{1}{2} R_{ii}^{-1} g_i^T (\Delta V_i)^T \quad (6)$$

Substituting eq. (6) into eq. (5), the closed-loop Hamilton-Jacobi equation is expressed as

$$0 = r_i(x(t), u_1, ..., u_N) + (\Delta V_i^{*T}(f(x) + \frac{1}{4} \sum_{j=1}^N u_j^T R_{ij} u_j^*)) \quad (7)$$

B. Control Barrier Function

First, define the *forward invariant* property for a set $c \subset \mathbb{R}^n$. if, for any $x_0 \in c$, system dynamic(1)) 's solution satisfy $x(t) \in c$ in a pre-defined period $t \in \mathscr{I}(x_0)$, where $\mathscr{I}(x_0)$ is the maximum interval corresponding to initial state x_0 . The *forward invariant* set c is a *safe set*, which consists of interior and boundary

$$\begin{cases} c = \{x \in \mathbb{R}^n \mid h(x) \geq 0\} \\ \partial c = \{x \in \mathbb{R}^n \mid h(x) = 0\} \\ \text{Int}(c) = \{x \in \mathbb{R}^n \mid h(x) > 0\} \end{cases} \quad (8)$$

where $h \in \mathbb{R}^n$ is the boundary function, which vanishes in the boundary of c. For any state $x(t) \in \partial c$, we mark that the system (1) is a safe system.

Definition 1. If a continuous function $b(x)$ satisfies three important properties below, it is called as *barrier function*

1) The function $b(x)$ does not go to infinity when $x(t) \in Int(c)$, that is, $|b(x)| < \infty$.

2) As the state x approaches the boundary of the forward invariant set, the function $b(x)$ goes to infinity, expressed as $\lim_{z \to \partial c} b(x) = \infty$.

3) The equilibrium value of the barrier function vanishes, that is, $b(0) = 0$.

Then, as to facilitate the subsequent design of safety-guarding controller. We select the barrier function $b(x)$ in the form of

$$b(x) = \left(\frac{1}{h(x)} - \frac{1}{h(0)}\right)^2 \quad (9)$$

where $h(x)$ is a nonzero continuous boundary function that ensures $b(x)$ meet all three properties of Definition 1.

III. Safety-Guarding Controller Design

The multi-player non-zero sum game and the definition of the barrier function are introduced in the previous section. Motivated by [7], we introduce the safety-guarding controller

$$u_b(x) = -\alpha_i g_i(x)^T \Gamma\left(\nabla b(x)^T\right) \quad (10)$$

where α_i is the selected control gain. $b(x)$ is the barrier function as we defined in the last section.

Lemma 1. For N-player dynamical system (1), assume that the interior set $Int(c)$ contains the origin x_0. If for all $t \in \mathscr{I}(x_0)$, the barrier function doesn't approach infinity, that is, $\|b(x(t))\| < \infty$, the interior set $Int(c)$ has the property of *forward invariant*.

According to the fact of Lemma 1, the safety of the system is guaranteed under the condition that the barrier function is finite. To design the controller for the specific multi-player nonzero-sum game, we give the following assumption hold.

Assumption 1. For N-player dynamical system (1), given a *forward invariant* set c, assume that the following properties hold:

1) Nonlinear dynamic $f(x)$ is bounded by a non-negative increasing function $\overline{f} \in \mathbb{R}_{\geq 0}$, that is, $\|f(x)\| \leq \overline{f}(x)$ and $\lim_{x \to \partial c} \overline{f}(x) < \infty$.

2) There exist a lower bound for $g(x)$, that is, $\underline{g} \leq \|g(x)\|$ for all $x \in c$, where $\underline{g} \in \mathbb{R}_{>0}$ is a positive constant.

3) The non-zero neighborhood of the boundary set ∂c is defined as $\mathscr{N}(\partial c)$, which satisfies that for all $x \in \mathscr{N}(\partial c)$, safety-guarding controllers will not vanish, that is, $\|\Gamma(\nabla b(x)) g(x)\| \neq 0$.

Based on Assumption 1 and Lemma 1, we have the following theorem to obtain the safe control policy, which renders the interior set $Int(x)$ *forward invariant* for the system (1)).

Lemma 2. For N-player dynamical system (1), a *forward invariant* set $c \subset \mathbb{R}^n$ from eq. (8) which satisfies $0 \in Int(c)$, and define b as a barrier function for the multi-player game (1). Given that Assumption 1 holds, the safety-guarding controller of eq. (10) $u = u_b(x)$ ensures the interior set $Int(c)$ is *forward invariant*, in which the safety of the system (1) is protected.

The above result shows when the safety-guarding term is used as a controller, interior set $Int(c)$ is ensured to be

forward invariant. Next, a regular adaptive dynamic programming (ADP) controller is obtained for solving the Nash equilibrium solution. Later, it is combined with the safety-guarding controller, which guarantees safe exploration within the state constraints of any convex/non-convex set.

Lemma 3. Assume continuous-time controller $u_i(x,t)$ is designed to be locally Lipschitz in state space and meets the condition that $u_i(0,t)=0$ for $t \in \mathscr{I}(x_0)$. Assume Assumption 1 holds and the drift dynamic is bounded as $\|g(x)u_i(x,t)\| \leq \overline{g}_u$, where $\overline{g}_u$ has the same definition as $\overline{f}$. The controller is obtained as

$$u_{b,i} = u_i(x,t) + u_b(x) \tag{11}$$

ensures that the interior set $Int(C)$ is a forward invariant set for (1).The controller also guarantees that the origin of state space is the final equilibrium solution of the system (1).

With a nominal controller and a safety-guarding term, we derived a controller $u_{b,i}$ that maintains $Int(c)$ *forward invariant*. In the next section, we will detail the use of RL for the online approximation of the nominal controller to obtain eq. (11).

IV. Online Approximation based on Reinforcement Learning

In previous sections, we introduced the safe-guarding controller to keep the safety of exploration under a state boundary. In this section, the structure of RL is used to implement the approximation of the control and value function. To avoid the danger of persistence excitation exceeding the safety limits, the technique of concurrent learning is utilized.

A. Approximation for the Value Function

To obtain the analytical solution of control policies u_i and value functions V_i, we utilize a single critic network for the approximation of value function V_i, which in the form of

$$V_i = \omega_i^T \phi(x) + \varepsilon_i(x)^T \tag{12}$$

where $\omega_i \in \mathbb{R}^{p_i}$ is the ideal weight for the single network and $\phi(x) \in \mathbb{R}^{n \times p_i}$ is the vector of the activation function, p_i is the hidden layer neuron number and $\varepsilon_i(x)$ is the critic network's approximation error. The gradient of V_i is expressed as

$$\nabla V_i = \nabla \phi(x)^T \omega_i + \nabla \varepsilon_i(x) \tag{13}$$

The estimated approximation of the ideal value function V_i is defined as

$$\hat{V}_i = \hat{\omega}_i^T \phi(x) \tag{14}$$

where $\hat{\omega}_i \in \mathbb{R}^{p_i}$ is the estimated weight of the single network, which is implemented in the single network to estimate the actual value of V_i.

B. Single Neural Network

To reduce the computational load, the approximation of control u_i is implemented through the single network method, which is in the form of

$$u_i = -\frac{1}{2} R_{ii}^{-1} g_i^T (\Delta \phi_i^T(x) \omega_i + \Delta \varepsilon_i^T(x)) \tag{15}$$

With the estimated value's gradient using the weights ω_i in eq. (13), the actual controller can be expressed in the form of

$$\hat{u}_i = -\frac{1}{2} R_{ii}^{-1} g_i^T \nabla \phi_i^T(x) \hat{\omega}_i \tag{16}$$

Then we add the safety-guarding term (10) to the control policy in the term of

$$u_{b,i} = \hat{u}_i - \frac{\alpha_i}{2} R_{ii}^{-1} g_i(x)^T \nabla b(x)^T \tag{17}$$

C. Critic Learning using Concurrent Learning

Based on eq. (7), (14), and (16), we can define the error of approximating the Hamilton-Jacobi equation, in the form of

$$\delta_i = \Omega_i^{\mathrm{T}} \sigma_i + x^{\mathrm{T}} Q_i x + \sum_{j=1}^{N} \frac{1}{4} \omega_j^{\mathrm{T}} \sigma_j' G_{ij} \sigma_j'^{\mathrm{T}} \omega_j + \nabla \varepsilon_i^T \Omega_i \tag{18}$$

where $G_j = g_j R_{jj}^{-1} g_j^{\mathrm{T}}$, $G_{ij} = g_j R_{jj}^{-1} R_{ij} R_{jj}^{-1} g_j^{\mathrm{T}}$, $\sigma_j = \nabla \phi(x)(f + \sum_{k=1}^{N} g_k u_{b,k})$ and $\Omega_i := \sigma_i' f - \frac{1}{2} \sum_{j=1}^{N} \sigma_i' G_j \sigma_j'^{\mathrm{T}} \hat{\omega}_j$. To simplify the notation, we denote $e_i = \Omega_i^{\mathrm{T}} \sigma_i + x^{\mathrm{T}} Q_i x + \sum_{j=1}^{N} \frac{1}{4} \hat{\omega}_j^{\mathrm{T}} \sigma_j' G_{ij} \sigma_j'^{\mathrm{T}} \hat{\omega}_j$ and $\nabla \varepsilon_i^T \Omega_i = -\varepsilon_{ham,i}$, which result in

$$\delta_i = e_i - \varepsilon_{ham,i} \tag{19}$$

To obtain an admissible control policy u and facilitate the following optimization, we first combine the historical and instantaneous data in the form of the total energy-like objective E_i, which is expressed as

$$E_i = \frac{1}{2} \left[\frac{\sigma_i^2}{\left(1 + \sigma_i^T \sigma_i\right)^2} + \sum_{k=1}^{M} \frac{(\sigma_i^k)^2}{\left(1 + (\sigma_i^k)^T \sigma_i^k\right)^2} \right] \tag{20}$$

where σ_i^k is the k-th historical data of σ_i. M is the total number of historical data.

According to the property of the above objective function, we can obtain the adaptation law based on least squares for the estimated critic network weight $\hat{\omega}_i$ as follows.

$$\begin{aligned} \dot{\hat{\omega}}_i &= -\beta_i \frac{\partial E_i}{\partial \omega_i} \\ &= -\beta_i \frac{\sigma_i e_i}{\left(1 + \sigma_i^T \sigma_i\right)^2} - \beta_i \sum_{k=1}^{M} \frac{\sigma_i^k e_i^k}{\left(1 + (\sigma_i^k)^T \sigma_i^k\right)^2} \end{aligned} \tag{21}$$

where β_i is the learning gain for each player, determine the convergence speed of each player's single network weight ω_i. Although a higher learning rate β_i may accelerate the convergence speed, it could also be a training disaster due to the cautiousness of the safety-guarding controller.

V. Stability Analysis

In this section, to investigate the stability of the proposed regular ADP controller, the Lyapunov stability analysis is presented. First, we introduce the following assumption to facilitate the stability proof.

Assumption 2. To facilitate the following Lyapunov analysis, assume that the following bounded conditions hold.

1. The ideal weight ω_i of the single network is bounded, that is, $\|\omega_i\| \leq \overline{\omega}_i$.

2. The approximation error $\varepsilon_i(x)$ of the single network and its corresponding gradient are bounded, that is, $\|\varepsilon_i(x)\| \leq \overline{\varepsilon}_i$ and $\|\nabla\varepsilon_i(x)\| \leq \nabla\overline{\varepsilon}_{max,i}$.

3. The activation vector $\phi_i(x)$ of the single network and its corresponding gradient are bounded, that is, $\|\phi_i(x)\| \leq \overline{\phi}_i$ and $\|\nabla\phi_i(x)\| \leq \nabla\overline{\phi}_i$.

4. The Hamiltonian residual is bounded, that is, $\left\|\varepsilon_{\text{ham},i}\right\| \leq \overline{\varepsilon}_{\text{ham},i}$.

5. $g_i(x)$ is bounded on Ω, that is, $g_i(x) \leq \overline{g}_i$.

Theorem 1. For N-player dynamical system (1), a *forward invariant* set $c \subset \mathbb{R}^n$ from (18) which satisfies $0 \in Int(c)$, and define b as a barrier function for the multi-player game 1. Given Assumption $1 \sim 2$ holds. and

$$\begin{cases} \overline{g}_i\overline{\phi}_j < 0 \\ \rho < 0 \\ \beta_i\left(\frac{p+1}{2} - 2\lambda_{\min}(\Gamma_k)\right) < 0 \end{cases} \tag{22}$$

where $\rho = \sum_{i=1}^{N}\left[\beta_i\frac{p+1}{2}\overline{\varepsilon}_i^2 - (\overline{\omega}_i\overline{\phi}_i + \overline{\varepsilon}_i)\sum_{j=1}^{N}\left(\frac{1}{2}G_j\overline{\phi}_i\|\hat{\omega}_j\| - g_i\overline{\varepsilon}_i\right)\right]$ for a short notation.

Then the control policy in (17) and the concurrent learning-based updating law in eq. (21), ensure that the interior set $Int(c)$ is *forward invariant* for the multi-player game (1). In addition, there exists a global equilibrium point, and the state converges to zero asymptotically.

Proof. We define the following Lyapunov function for stability analysis:

$$V_L = \sum_{i=1}^{N}(V_i + V_{\omega,i}) \tag{23}$$

where $V_{\omega,i} = \frac{1}{2}\tilde{\omega}_i^{\mathrm{T}}\tilde{\omega}_i$ is an additional error term for the single network weights.

For each player, we have

$$\dot{V}_i = \left(\frac{\partial V_i(x)}{\partial x}\right)^T\left[f(x) - \frac{1}{2}\sum_{j=1}^{N}g_j(x)R_{jj}^{-1}g_j^T\nabla\phi_i^T(x)\hat{\omega}_j\right] \tag{24}$$

Combining the controller (16), Hamilton-Jacobi equation (7) and Assumption 3, we get

$$\dot{V}_i \leq -r_i - (\overline{\omega}_i\overline{\phi}_i + \overline{\varepsilon}_i)\sum_{j=1}^{N}\left(\frac{1}{2}G_j\overline{\phi}_i\|\hat{\omega}_j\| - g_i\overline{\varepsilon}_i\right) \tag{25}$$

Differentiating each player's weight-error term $V_{\omega,i}$ result in the following equation

$$\dot{V}_{\omega,i} = \tilde{\omega}_i^T\dot{\tilde{\omega}}_i \tag{26}$$

Then, given the updating law form (21), the dynamic of each player's single network weight error can be expressed as

$$\dot{\tilde{\omega}}_i = -\beta_i\left[\Gamma_a(t) + \Gamma_k\right]\tilde{\omega}_i(t) + \beta_i\Lambda_a \tag{27}$$

where

$$\Gamma_a(t) = \frac{\sigma_i(\sigma_i)^T}{\left[1 + (\sigma_i)^T\sigma_i\right]^2}, \quad \Gamma_k = \sum_{k=1}^{p}\frac{\sigma_i\left(\sigma_i^k\right)^{\mathrm{T}}}{\left[1 + \left(\sigma_i^k\right)^{\mathrm{T}}\sigma_i\right]^2} \tag{28}$$

$$\Lambda_a = \frac{\sigma_i\varepsilon_{\text{ham},i}}{\left[1 + (\sigma_i)^T\sigma_i\right]^2} + \sum_{k=1}^{p}\frac{\sigma_i^k\varepsilon_{\text{ham},i}^k}{\left[1 + \left(\sigma_i^k\right)^T\sigma_i^k\right]^2} \tag{29}$$

Inserting eq. (27) into eq. (26) yields

$$\dot{V}_{\omega,i} \leq \beta_i\left[\frac{p+1}{2} - 2\lambda_{\min}(\Gamma_k)\right]\|\tilde{\omega}_i\|^2 + \beta_i\frac{p+1}{2}\overline{\varepsilon}_{\text{hmax},i}^2 \tag{30}$$

Combining inequality (25) and (30) yields

$$\begin{aligned}\dot{V} \leq & -\sum_{i=1}^{N}r_i + \rho \\ & + \sum_{i=1}^{N}\left[\overline{g}_i\overline{\phi}_i + \beta_i\left(\frac{p+1}{2} - 2\lambda_{\min}(\Gamma_k)\right)\right]\|\tilde{\omega}_i\|^2\end{aligned} \tag{31}$$

For each player, if (22) in Assumption 2 holds, we have $\dot{V}_L \leq 0$. Then, according to the theorem of Lyapunov stability, the stability of the proposed controller (17) is guaranteed. By the property of *forward invariant* and the asymptotic stability, the safety of forcing multi-player game (1) to Nash equilibrium is guaranteed.

VI. Simulation Results

A. Two-Player Problem Setup

To verify the safety-guarding capability of the proposed method, we investigate in a nonlinear case of nonzero-sum game for two-player. To test the generality of the safety-guarding controller, the barrier function is chosen in the form of both non-convex and convex. We select the nonlinear control-affine system

$$\dot{x} = f(x) + g_1(x)u_1 + g_2(x)u_2 \tag{32}$$

where $x \in \mathbb{R}^2, u_1, u_2 \in \mathbb{R}$, and the nonlinear dynamic is chosen as

$$f = \begin{bmatrix} x_2 - 2x_1 \\ -\frac{1}{2}x_1 - x_2 + \frac{1}{4}x_2(\cos(2x_1) + 2)^2 \\ + \frac{1}{4}x_2\left(\sin\left(4x_1^2\right) + 2\right)^2 \end{bmatrix}$$

$$g_1 = \begin{bmatrix} 0 \\ \cos(2x_1) + 2 \end{bmatrix} \quad g_2 = \begin{bmatrix} 0 \\ \sin\left(4x_1^2\right) + 2 \end{bmatrix}$$

The construction of the value function is depicted in (2), with weights $Q_1 = 2Q_2 = \|x\|$ and $R_{11} = R_{12} = 2R_{21} = 2R_{22} = 2$.

To obtain an approximate online Nash equilibrium solution for the provided nonzero-sum two-player game, first we specify a safety-guarding controller design from Section III, then the concurrent learning-based ADP controller from Part IV is used together.

To stabilize the two-player game, the objective of our proposed controller is to guarantee that the state $x(t)$ converges to zero, while making sure that $x(t)$ does not move out of the safe boundary set ∂c . In this numerical simulation, the boundary set ∂c have a specified boundary function $h(x) = px_2^2 - x_1 + 1$ defined in (8), where p is the coefficient to decide the convexity property of the safe set c. For simplicity,

we choose $p=-1$ for convex set, and $p=1$ for non-convex set.

The initial state is selected as $x_0=[-4,2.2]$ which is close enough to the safe set boundary. By creating the barrier function $b(x)=(1/h(x)-1/h(0))^2$ and the controller gain $\alpha_i=0.1$, a safety-guarding controller is obtained for each evaluation. To obtain the approximation control policy, we utilize the Staf kernels from [25] as the activation function. The learning rates for each player are selected as $\beta_1=1$, $\beta_2=0.1$, and the weights are initialized as $\hat{\omega}_i(t_0)=[0.5,0.5,0.5]^T$ for the convex set, and $[3,3,3]^T$ for the nonconvex set.

B. Analytical Solution

To compare with the proposed result and the ideal solution, first we analyze the equilibrium point's value function of the system from (30) as

$$V_1^*=\begin{bmatrix}0.5\\0\\1\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}x_1^2\\x_1x_2\\x_2^2\end{bmatrix}\quad V_2^*=\begin{bmatrix}0.25\\0\\0.5\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}x_1^2\\x_1x_2\\x_2^2\end{bmatrix} \tag{33}$$

and the corresponding ADP controller for each player are expressed as

$$\begin{cases}u_1^*=-\dfrac{1}{2}R_{11}^{-1}g_1^{\mathrm{T}}\begin{bmatrix}2x_1&0\\x_2&x_1\\0&2x_2\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}0.5\\0\\1\end{bmatrix}\\u_2^*=-\dfrac{1}{2}R_{22}^{-1}g_2^{\mathrm{T}}\begin{bmatrix}2x_1&0\\x_2&x_1\\0&2x_2\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}0.25\\0\\0.5\end{bmatrix}\end{cases} \tag{34}$$

C. Simulation Results

To demonstrate the effectiveness of the proposed technique, the system is simulated with and without the safety-guarding controller in both non-convex/convex state constraints scenarios. For the convex case, we set $p=-1$, the main result is presented in Fig. 1 and the learning process is shown in Fig. 2. As presented in Fig. 1, by utilizing the control strategy from (17) with the added safety-guarding term, the state $x(t)$ is stabilized to zero state, while never move out of the safe set. In contrast, the control policy without the safety-guarding term stabilizes the state to zero, but in the beginning, the state trajectory violates the security constraint.

As shown in Fig. 2 and 3, the value functions of Player 1 and Player 2 converge to [0.5000, 0, 1.0002] and [0.2500, 0, 0.5001], respectively, and the value functions and controllers obtained is:

$$\hat{V}_1=\begin{bmatrix}0.5000\\0\\1.0002\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}x_1^2\\x_1x_2\\x_2^2\end{bmatrix}\quad \hat{V}_2=\begin{bmatrix}0.2500\\0\\0.5001\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}x_1^2\\x_1x_2\\x_2^2\end{bmatrix}$$

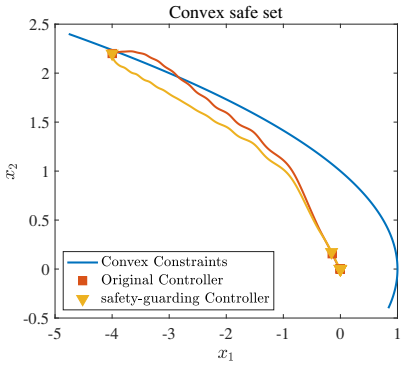

Fig. 1. Trajectories of the two-player nonzero-sum game system with a convex safe boundary

$$\hat{u}_1=-\frac{1}{2}R_{11}^{-1}g_1^{\mathrm{T}}\begin{bmatrix}2x_1&0\\x_2&x_1\\0&2x_2\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}0.5000\\0\\1.0002\end{bmatrix}$$

$$\hat{u}_2=-\frac{1}{2}R_{22}^{-1}g_2^{\mathrm{T}}\begin{bmatrix}2x_1&0\\x_2&x_1\\0&2x_2\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}0.2500\\0\\0.5001\end{bmatrix}$$

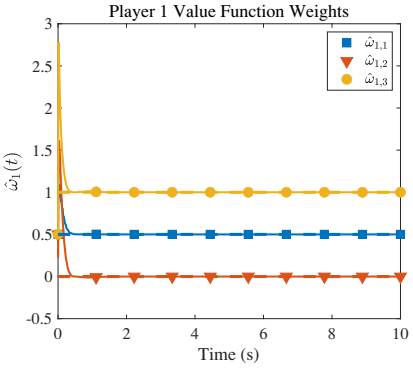

Fig. 2. Player 1 value function weights

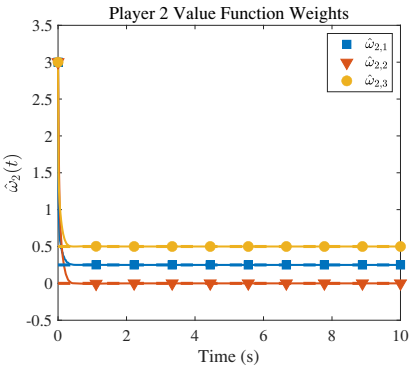

Fig. 3. Player 2 value function weights

To demonstrate the generality of the safety-guarding controller, we set a non-convex boundary restriction. Fig. 4 shows the difference in the control effect of the controller with/without the safety-guarding term. The initial state is

$x_0 = [1.5, 3]$, and the state trajectories of both controllers are the same in the initial period. However, when the safety-guarding controller approaches the boundary, the safety-guarding term drags the state trajectory away from the direction of the reverse gradient of the barrier function. When the distance from the boundary gradually increases, the effect of the safety-guarding term gradually disappears and finally converges to zero. However, the state trajectory of the original controller directly crosses the state limitation without any sign of tending to move away from the boundary.

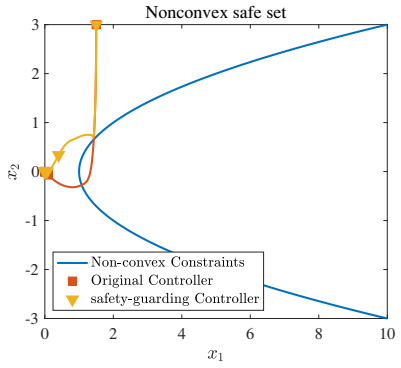

Fig. 4. Trajectories of the nonlinear system in a non-convex safe set

VII. Conclusions

In this paper, we introduce a safety-guarding controller to keep safe exploration in constrained state space. The non-zero sum game Nash equilibrium solution is obtained by developing a model-based reinforcement learning architecture. To deal with the uncertainty of persistence excitation, we apply concurrent learning methods using both historic and instantaneous data to train the network without excitation risks. In order to relax the computation load, we utilize the single-network technique for the approximation. In the future, we will investigate the use of a Lyapunov-based deep neural network to improve approximation accuracy.

References

[1] D. Liu, S. Xue, B. Zhao, B. Luo, and Q. Wei, "Adaptive Dynamic Programming for Control: A Survey and Recent Advances," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, pp. 142–160, Jan. 2021.

[2] K. G. Vamvoudakis, H. Modares, B. Kiumarsi, and F. L. Lewis, "Game Theory-Based Control System Algorithms with Real-Time Reinforcement Learning: How to Solve Multiplayer Games Online," *IEEE Control Systems Magazine*, vol. 37, pp. 33–52, Feb. 2017.

[3] Z. Li, G. Li, X. Wu, Z. Kan, H. Su, and Y. Liu, "Asymmetric cooperation control of dual-arm exoskeletons using human collaborative manipulation models," *IEEE Transactions on Cybernetics*, vol. 52, no. 11, pp. 12126–12139, 2021.

[4] D. Wang, H. He, and D. Liu, "Adaptive Critic Nonlinear Robust Control: A Survey," *IEEE Transactions on Cybernetics*, vol. 47, pp. 3429–3451, Oct. 2017.

[5] Y. Yang, Y. Yin, W. He, K. G. Vamvoudakis, H. Modares, and D. C. Wunsch, "Safety-Aware Reinforcement Learning Framework with an Actor-Critic-Barrier Structure," in *2019 American Control Conference (ACC)*, pp. 2352–2358, July 2019. ISSN: 2378-5861.

[6] Z. Marvi and B. Kiumarsi, "Safe reinforcement learning: A control barrier function optimization approach," *International Journal of Robust and Nonlinear Control*, vol. 31, no. 6, pp. 1923–1940, 2021.

[7] M. H. Cohen and C. Belta, "Safe exploration in model-based reinforcement learning using control barrier functions," *Automatica*, vol. 147, p. 110684, Jan. 2023.

[8] J. Xu, J. Wang, J. Rao, Y. Zhong, and H. Wang, "Adaptive dynamic programming for optimal control of discrete-time nonlinear system with state constraints based on control barrier function," *International Journal of Robust and Nonlinear Control*, vol. 32, pp. 3408–3424, Apr. 2022.

[9] Y. Zhou, K. G. Vamvoudakis, W. M. Haddad, and Z.-P. Jiang, "A Secure Control Learning Framework for Cyber-Physical Systems Under Sensor and Actuator Attacks," *IEEE Transactions on Cybernetics*, vol. 51, pp. 4648–4660, Sept. 2021.

[10] M. L. Greene, P. Deptula, S. Nivison, and W. E. Dixon, "Sparse Learning-Based Approximate Dynamic Programming With Barrier Constraints," *IEEE Control Systems Letters*, vol. 4, pp. 743–748, July 2020.

[11] Y. Yang, D.-W. Ding, H. Xiong, Y. Yin, and D. C. Wunsch, "Online barrier-actor-critic learning for H control with full-state constraints and input saturation," *Journal of the Franklin Institute*, vol. 357, pp. 3316–3344, Apr. 2020.

[12] Y. Yang, K. G. Vamvoudakis, and H. Modares, "Safe reinforcement learning for dynamical games," *International Journal of Robust and Nonlinear Control*, vol. 30, pp. 3706–3726, June 2020.

[13] K. G. Vamvoudakis and F. L. Lewis, "Multi-player non-zero-sum games: Online adaptive learning solution of coupled Hamilton–Jacobi equations," *Automatica*, vol. 47, pp. 1556–1569, Aug. 2011.

[14] M. Johnson, R. Kamalapurkar, S. Bhasin, and W. E. Dixon, "Approximate N-Player Nonzero-Sum Game Solution for an Uncertain Continuous Nonlinear System," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, pp. 1645–1658, Aug. 2015.

[15] R. Kamalapurkar, J. R. Klotz, and W. E. Dixon, "Concurrent learning-based approximate feedback-Nash equilibrium solution of N-player nonzero-sum differential games," *IEEE/CAA Journal of Automatica Sinica*, vol. 1, pp. 239–247, July 2014.

[16] K. G. Vamvoudakis and F. L. Lewis, "Online actor–critic algorithm to solve the continuous-time infinite horizon optimal control problem," *Automatica*, vol. 46, pp. 878–888, May 2010.

[17] S. Liu, L. Liu, and Z. Yu, "Safe reinforcement learning for discrete-time fully cooperative games with partial state and control constraints using control barrier functions," *Neurocomputing*, vol. 517, pp. 118–132, Jan. 2023.

[18] Q. Zhang, D. Zhao, and Y. Zhu, "Event-Triggered H Control for Continuous-Time Nonlinear System via Concurrent Learning," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, pp. 1071–1081, July 2017.

[19] G. Chowdhary, T. Yucelen, M. Mühlegg, and E. N. Johnson, "Concurrent learning adaptive control of linear systems with exponentially convergent bounds," *International Journal of Adaptive Control and Signal Processing*, vol. 27, no. 4, pp. 280–301, 2013.

[20] S. M. N. Mahmud, S. A. Nivison, Z. I. Bell, and R. Kamalapurkar, "Safe Model-Based Reinforcement Learning for Systems With Parametric Uncertainties," *Frontiers in Robotics and AI*, vol. 8, p. 733104, Dec. 2021.

[21] H. Modares, F. L. Lewis, and M.-B. Naghibi-Sistani, "Integral reinforcement learning and experience replay for adaptive optimal control of partially-unknown constrained-input continuous-time systems," *Automatica*, vol. 50, pp. 193–202, Jan. 2014.

[22] M. Abu-Khalaf and F. L. Lewis, "Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach," *Automatica*, vol. 41, pp. 779–791, May 2005.

[23] R. Kamalapurkar, L. Andrews, P. Walters, and W. E. Dixon, "Model-Based Reinforcement Learning for Infinite-Horizon Approximate Optimal Tracking," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, pp. 753–758, Mar. 2017.

[24] M. L. Greene, Z. I. Bell, S. Nivison, and W. E. Dixon, "Deep Neural Network-based Approximate Optimal Tracking for Unknown Nonlinear Systems," *IEEE Transactions on Automatic Control*, pp. 1–8, 2023.

[25] C. Peng, H. Zhang, Y. He, and J. Ma, "State-Following-Kernel-Based Online Reinforcement Learning Guidance Law Against Maneuvering Target," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 58, pp. 5784–5797, Dec. 2022.