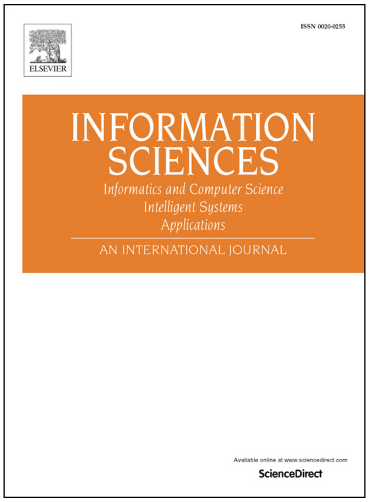
Finite-time Safe Reinforcement Learning Control of Multi-player Nonzero-Sum Game for Quadcopter Systems

Junkai Tan, Shuangsi Xue, Qingshu Guan, Kai Qu and Hui Cao



ISSN 0020-0255

ELSEVIER

INFORMATION SCIENCES

Informatics and Computer Science
Intelligent Systems
Applications

AN INTERNATIONAL JOURNAL

Available online at www.sciencedirect.com

ScienceDirect

PII: S0020-0255(25)00249-X
DOI: https://doi.org/10.1016/j.ins.2025.122117
Reference: INS 122117

To appear in: *Information Sciences*

Received date: 12 February 2025
Revised date: 19 March 2025
Accepted date: 20 March 2025

Highlights

- Novel Theory: A finite-time SRL algorithm for multi-player nonzero-sum game is developed with theoretical guarantees on convergence and safety.
- New Algorithm: A finite-time concurrent learning law is proposed for NN training, which improves finite-time convergence speed and reduces excitation requirements.
- Unified Framework: The method combines barrier functions, finite-time stability, and multi-player games in a unified control framework for quadcopter systems.
- Extensive Validation: Numerical simulations and hardware experiments on quadcopters demonstrate superior performance over existing methods.

Finite-time Safe Reinforcement Learning Control of Multi-player Nonzero-Sum Game for Quadcopter Systems

Junkai Tan[a,b], Shuangsi Xue[a,b], Qingshu Guan[a,b], Kai Qu[a,b], Hui Cao[a,b]

[a]School of Electrical Engineering, Xi'an Jiaotong University, Xi'an, 710049, China
[b]State Key Labrotary, Xi'an Jiaotong University, Xi'an, 710049, China

Abstract

This paper investigates a finite-time safe reinforcement learning control algorithm for multi-player nonzero-sum games (FT-SRL-NZS). In addressing the finite-time safe optimal control issue, value functions incorporating designated barrier functions for the involved players are established within the transformed finite-time stable space. The finite-time safe optimal controller is derived from the solution to the transformed Nash equilibrium condition. An actor-critic structure is proposed for solving the Hamilton-Jacobi-Bellman (HJB) equation in the finite-time stable space, aimed at approximating the finite-time optimal value and its corresponded controller using a novel finite-time concurrent learning update law. A dynamic event-trigger rule adjusts the trigger condition in real time, thereby minimizing the computational and communicative demands associated with calculating Nash equilibrium. Lyapunov stability analysis is employed to examine the finite-time equilibrium of the closed-loop system. Numerical simulations and unmanned aerial vehicle (UAV) hardware tests are carried out to illustrate the efficacy of the proposed finite-time safe control algorithm.

Keywords: Finite-time optimal control, nonzero-sum game, reinforcement learning, neural network, dynamic event-trigger, adaptive dynamic programming

1. Introduction

Optimal control has been extensively employed in the control of multi-player systems [1]. In multi-player systems [2], agent interactions are typically represented as non-zero-sum (NZS) games [3]. The NZS game extends the concept of the zero-sum game and is commonly applied in the control of multi-player systems [4]. In an NZS game, the agents do not engage in competition; rather, their objective is to attain a shared goal [5]. The NZS game is extensively investigated in the field of multi-player control, including formation control [6], distributed control [7], and cooperative control [8]. In recent years, numerous studies have been undertaken to seek the Nash equilibrium in NZS games [9]. However, the convergence speed of Nash equilibrium is slow and uncertain, making it challenging to ensure the performance of the control algorithm.

In practical applications, such as the pursuit-evasion game [10] and the human-robotics cooperative games [11], the precise assurance of convergence speed is crucial for the effective control of NZS games [12]. Finite-time optimal control is a technique that guarantees the convergence rate of system states within a specified finite duration [13]. This method involves the formulation of a finite-time optimal value function and its associated finite-time optimal controller. The application of fractional-order calculus in finite-time optimal control significantly elevates the computational complexity of the control algorithm [14]. Also, the solution to the Hamilton-Jacobi-Bellman (HJB) equation in finite-time optimal control presents significant challenges [15], potentially leading to the curse of dimensionality in control algorithms. To solve the aforementioned issues, the reinforcement learning (RL) algorithm is introduced to approximate the value function and the optimal controller in the finite-time optimal control [16]. Policy iteration [17] and value iteration [18] are two commonly employed RL algorithms to solve the problem of optimal control.

Email addresses: tanjk@stu.xjtu.edu.cn (Junkai Tan), xssxjtu@xjtu.edu.cn (Shuangsi Xue)

The actor-critic method [19] has been developed for obtaining the value function using the critic-network and the optimal controller by the actor-network in [20]. Q-learning is extensively studied in RL algorithms for the approximation of the value function through iterative processes [21]. Nonetheless, the convergence rate of the RL algorithm is limited [22], and the conditions for training excitation are demanding [23]. The advancement of concurrent learning techniques seeks to enhance the convergence speed of RL algorithms by approximating the value function and the optimal controller within a finite-time convergence framework [24]. The finite-time concurrent learning (FT-CL) law for the RL algorithm merits investigation, as it effectively enhances the convergence speed and relaxes the excitation conditions during training.

Another major concern in the control of NZS-based multi-player systems is the safety performance of the control algorithm [25], which is defined as the assurance that system states remain within a safe operational region [26]. Previous research [27] introduced the barrier-function-transformation, which converts the safety functionality of the control algorithm into a stabilization problem of the transformed system states. The barrier-penalty methods are studied to impose penalties on system states that breach safety constraints [28]. There are two primary challenges in the safe RL control of multi-player NZS games. First, the simultaneous assurance of finite-time performance and safety performance in the control algorithm is challenging [29], potentially diminishing the algorithm's effectiveness. Second, the modeling of constraints and the design of barrier functions within the control algorithm presents significant challenges [30]. The finite-time safe RL control of multi-player NZS games deserves further investigation to address the aforementioned challenges, given its effectiveness in improving both finite-time performance and safety of the control algorithm.

Recent studies highlight several fundamental challenges in multi-player reinforcement learning control: (1) conventional non-zero-sum game algorithms [31] achieve only asymptotic convergence without ensuring operational safety, (2) existing safety-aware approaches [32] face significant limitations in convergence speed and system constraints, (3) the simultaneous achievement of efficient learning and safety guarantees remains elusive - our work bridges this gap through innovative FT-CL mechanisms and barrier function design [33] and [25], and (4) practical implementation validation is lacking in current methods [34], which we address via comprehensive quadcopter experiments. This paper presents a novel finite-time safe reinforcement learning framework for multi-player nonzero-sum games (FT-SRL-NZS) to overcome these challenges. The main contributions are:

1. Compared with existing RL [35] and safe RL [36] algorithms for NZS game, both finite-time performance and safety performance are considered in the proposed FT-SRL-NZS algorithm. A finite-time safe optimal control problem is formulated to attain the finite-time Nash equilibrium in a multi-player NZS game while circumventing obstacles. The value functions that incorporate specific barrier functions for participating players are defined within the transformed finite-time stable space. The finite-time performance and safety performance are simultaneously guaranteed by the proposed FT-SRL-NZS algorithm compared with [34].
2. A FT-CL-based update method is proposed for training the critic network weights, with the objective of approximating the value function and the optimal controller within a convergence-time-guaranteed framework. The FT-CL technique improves the speed of finite-time convergence [15] and alleviates the excitation conditions [37] required for training compared to existing methods.
3. The effectiveness of the proposed FT-SRL-NZS algorithm is demonstrated through numerical simulations and UAV hardware experiments [38], which shows that the proposed algorithm can achieve finite-time stabilization control of the system states while avoiding obstacles compared with existing methods [25].

The remainder of this paper is organized as follows: Section II describes the system model and obstacle formulation. Section III develops the finite-time safe optimal control framework. Section IV presents the proposed FT-SRL-NZS algorithm with an actor-critic structure. Section V analyzes finite-time stability and convergence properties. Section VI validates the algorithm through numerical simulations. Section VII demonstrates practical effectiveness via UAV hardware experiments. Section VIII concludes the paper with key findings and future directions.

Notation: The following notation will be used throughout the paper: The notation $|x|^c_\omega = \sum_{i=1}^n w_i |x_i|^c$ denotes the weighted norm of vector $x \in \mathbb{R}^n$ with weight vector $\omega = [w_1, \cdots, w_n]^\top$, and $|x|^c = \sum_{i=1}^n |x_i|^c$ denotes the norm of vector $x \in \mathbb{R}^n$.

2. Preliminaries

2.1. System description and obstacle modeling

Consider a nonlinear system with unknown drifted dynamics and multiple players. To establish the theoretical framework for safe control, we make the following assumptions:

Assumption 1 (Obstacle Construction)**.** To model the nearby space of the obstacle, the following assumptions are given:

1. Obstacles are considered to be static and there should be no overlap between obstacles; if there is an overlap, the overlapped obstacles are considered to be modeled as a larger obstacle.
2. The obstacle is represented by a minimum inner enveloping sphere, and the maximum radius of the obstacle is considered as the obstacle avoidance condition.
3. The center point of modeled obstacle is denoted as $p_{o,i}$ and radius as $r_{o,i}$, which is denoted as O_i, and the total number of obstacles is M.

Assumption 2 (System Properties)**.** For the augmented dynamics (2), the following holds:

1. Functions $f(x)$ and $g_i(x)$ are Lipschitz continuous on compact set $x \in \chi \in \mathbb{R}^n$ with $f(0) = 0$ and $\|g_i(x)\| \leq G_{Hi}$ for all $x \in \chi$.
2. Cost matrices satisfy $0 \leq \underline{\lambda}_{Q_i} \leq \|Q_i\| \leq \bar{\lambda}_{Q_i}$ and $0 \leq \underline{\lambda}_{R_{ij}} \leq \left\|R_{ij}\right\| \leq \bar{\lambda}_{R_{ij}}$ with $\bar{\lambda}_{Q_i}, \bar{\lambda}_{R_{ij}} > 0$.

Assumption 3 (Persistence of Excitation Condition)**.** The historical data stack for weights update satisfies that for the i-th time step ($i = 1, \ldots, N$), the following holds:

$$\begin{cases} \vartheta_{1i} I_{\mathcal{L}} \leqslant \int_T \frac{\phi_i \phi_i^\top}{(\phi_i^\top \phi_i + 1)^2} \mathrm{d}\tau, \\ \vartheta_{2i} I_{\mathcal{L}} \leqslant \sum_{l=1}^N \int_T \frac{\phi_i^l \phi_i^{l\top}}{(\phi_i^{l\top} \phi_i^l + 1)^2} \mathrm{d}\tau, \\ \vartheta_{3i} I_{\mathcal{L}} \leqslant \int_T \phi_i^\dagger |\phi_i|^{\alpha\top} \operatorname{sgn}(\phi_i) \mathrm{d}\tau, \\ \vartheta_{4i} I_{\mathcal{L}} \leqslant \sum_{l=1}^N \int_T \phi_i^{l\dagger} |\phi_i^l|^{\alpha\top} \operatorname{sgn}(\phi_i^l) \mathrm{d}\tau, \end{cases} \tag{1}$$

where ϑ_{ji} ($j = 1, 2, 3, 4$) is strictly positive for at least one j.

Assumption 4 (Neural Networks Boundedness)**.** Assuming that the following parameters and operators are bounded: $\|\hat{W}_{ci}\| \leq W_H$, $\|\phi_i(x)\| \leq \phi_H$, $\|\nabla\phi(x)\| \leq \phi_{DH}$, $\|\phi_i(x)\| \leq \phi_H$, $\|\nabla\phi_i(x)\| \leq \phi_{DH}$, $\|\epsilon_i(x)\| \leq \epsilon_H$, $\|\nabla\epsilon_i(x)\| \leq \epsilon_{DH}$,

Consider the following nonlinear system with unknown drifted dynamics:

$$\dot{x} = f(x) + \sum_{i=1}^N g_i(x) \mathcal{U}_i \tag{2}$$

where $x \in \mathbb{R}^n$ represents the system state, $\mathcal{U}_i \in \mathbb{R}^m$ denotes the control input, $f(x) \in \mathbb{R}^n$ describes the system dynamics, and $g_i(x) \in \mathbb{R}^{n\times m}$ specifies the control input matrix for player i. To characterize safety constraints near obstacles, we model the surrounding space as concentric spherical zones with varying risk levels.

Let $d_i(x, x_{o,i}, t) = \|x(t) - x_{o,i}(t)\|$ denote the instantaneous distance between the system state and obstacle i. The safety-critical regions are characterized by three nested zones:

1. Detection zone $\mathcal{D} = \cup_{i\in\mathcal{M}} \mathcal{D}_i$: Outer boundary where obstacle monitoring initiates

$$\mathcal{D}_i = \{x \in \mathbb{R}^n | R_{o,i} < d_i(x, x_{o,i}, t) \leq D_{o,i}\}$$

2. Warning zone $\mathcal{W} = \cup_{i\in\mathcal{M}} \mathcal{W}_i$: Intermediate region requiring preventive actions

$$\mathcal{W}_i = \{x \in \mathbb{R}^n | r_{o,i} < d_i(x, x_{o,i}, t) \leq R_{o,i}\}$$

3. Critical zone $O = \cup_{i\in\mathcal{M}} O_i$: Inner core demanding immediate evasive maneuvers

$$O_i = \{x \in \mathbb{R}^n | d_i(x, x_{o,i}, t) \leq r_{o,i}\}$$

The radii satisfy $r_{o,i} < R_{o,i} < D_{o,i}$ as illustrated in Fig. 1. The complete safety-constrained region is defined as $\mathcal{A} = \cup_{i\in\mathcal{M}} (\mathcal{D}_i \cup \mathcal{W}_i \cup O_i)$.

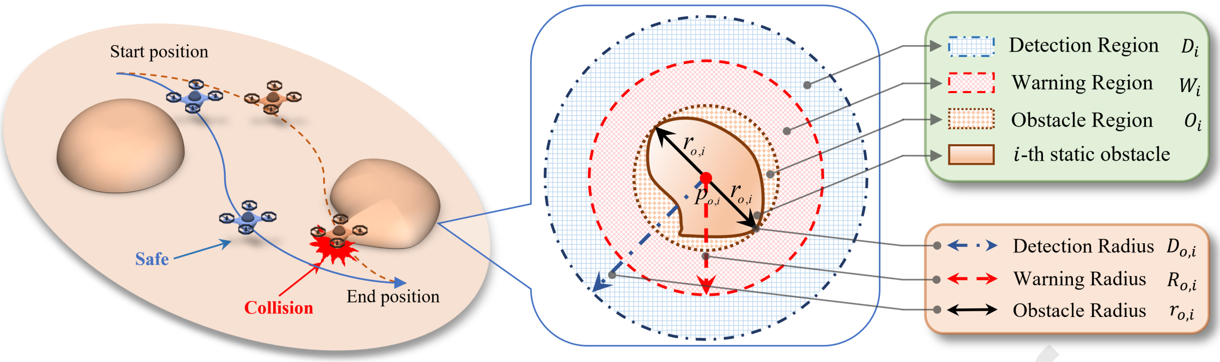


Figure 1: Safety regions around an obstacle consist of three nested zones: detection region $\mathcal{D}_i$ (blue), warning region $\mathcal{W}_i$ (red), and obstacle region $\mathcal{O}_i$ (yellow). Each spherical zone is centered at $x_{o,i}$ with corresponding radii $D_{o,i}$, $R_{o,i}$, and $r_{o,i}$.

2.2. *Unsafe region and barrier function design*

To incorporate safety constraints in system (2) near obstacles, we define regions where safe operation must be maintained.

Definition 1. (Safety-Critical Operational Zone [25]). For system (2), a domain $\mathcal{S} \subset \mathbb{R}^n$ is safety-critical if any trajectory initiating at $x(0) \in \mathcal{S}$ remains within $\mathcal{S}$ for all future time $t \geq 0$. The safety-critical zone $\mathcal{S}$ is characterized by:

$$\begin{aligned} \mathcal{S} &= \{x \in \mathbb{R}^n | h(x) \geq 0\} \\ \text{Bound}(\mathcal{S}) &= \{x \in \mathbb{R}^n | h(x) = 0\} \\ \text{Core}(\mathcal{S}) &= \{x \in \mathbb{R}^n | h(x) > 0\} \end{aligned}$$

where $h(x)$ represents a smooth safety criterion function, Bound denotes the boundary, and Core represents the interior of the safe zone.

To maintain safety while exploring, we require $h(x(t)) \geq 0, \forall t \geq 0$. This motivates the use of control barrier functions to ensure safe control.

Definition 2. (Control Barrier Function [39]). A continuously differentiable function $\mathcal{B}(x) : \mathbb{R}^n \to \mathbb{R}$ is a control barrier function (CBF) for system (2) if there exist positive constants β_1, β_2 such that:

$$\frac{1}{\beta_1 h(x)} \leq \mathcal{B}(x) \leq \frac{1}{\beta_2 h(x)}$$
$$\text{where} \begin{cases} \mathcal{B}(x) > 0, & \forall x \in \text{Core}(\mathcal{S}) \\ \mathcal{B}(x) \to \infty, & \text{as } x \to \text{Bound}(\mathcal{S}) \end{cases}$$

The CBF and corresponding safety region function are constructed as:

$$\mathcal{B}(x) = \frac{K_B s(x)}{h(x) + \mu} \tag{3}$$

where $h(x) = \sum_{i \in \mathcal{M}} h_i(x)$, $s(x) = \sum_{i \in \mathcal{M}} s_i(x)$, $h_i(x) = d_i(x, x_{o,i}, t) - r_{o,i}$, and K_B, μ are positive parameters. The safety function $s_i(x)$ is defined as:

$$s_i(x) = \begin{cases} 0, & d_i > D_{o,i}, \\ \xi_1(1 + \cos(\pi \frac{d_i^2 - D_{o,i}^2}{D_{o,i}^2 - R_{o,i}^2})), & R_{o,i} < d_i \leq D_{o,i}, \\ \xi_2 + \xi_3 \cos(\pi \frac{d_i^2 - r_{o,i}^2}{R_{o,i}^2 - r_{o,i}^2}), & r_{o,i} < d_i \leq R_{o,i}, \\ 1, & d_i \leq r_{o,i}, \end{cases} \tag{4}$$

with parameters satisfying $\xi_2 + \xi_3 = 1$ and $\xi_2 - \xi_3 = 2\xi_1$.

3. Problem formulation: finite-time safe optimal control

To achieve both safety and finite-time (FT) performance, we formulate an FT-safe optimal control problem that stabilizes the system states while avoiding obstacles. The goal is to design an optimal controller that drives the system states to the equilibrium point $x^* \in \Omega(\delta, x_0)$ within finite time, where $\delta > 0$ is arbitrary.

3.1. Optimal control problem

First, the following quadratic cost function can be defined for the system (2) with multiple players:

$$\mathcal{V}_i(x, \mathcal{U}_1, \ldots, \mathcal{U}_N) = \int_0^\infty r_i(x, \mathcal{U}_1, \ldots, \mathcal{U}_N)\, d\tau \tag{5}$$

where $r_i(x, \mathcal{U}_1, \ldots, \mathcal{U}_N)$ is defined as the reward of its corresponded player ($i = 1, \ldots, N$), the input $\mathcal{U}_i(t)$ is limited by bound $|\mathcal{U}_i(t)| \leq \mu_i$ for each player i. The running cost function $r_i(x, \mathcal{U}_1, \ldots, \mathcal{U}_N)$ takes the form:

$$r_i(x, \mathcal{U}_1, \ldots, \mathcal{U}_N) = |x|_x^\alpha + \sum_{k=1}^N \Lambda_{ik}(\mathcal{U}_k) + \mathcal{B}(x, x_o) \tag{6}$$

where $|x|_x^\alpha$ denotes the weighted state norm with weight matrix $\omega \in \mathbb{R}^{n\times n}$, $s_i(x, x_{o,i})$ from (4) provides smooth obstacle avoidance, $\mathcal{B}(x, x_o)$ represents the barrier function in (3), and $\Lambda_{ik}(\mathcal{U}_k)$ captures control penalties [40]:

$$\Lambda_{ik}(\mathcal{U}_k) = 2\mu_k R_{ik} \int_0^{\mathcal{U}_k} \tanh^{-1}(\gamma_{\mathcal{U}}/\mu_k)\, \mathrm{d}\gamma_{\mathcal{U}} \tag{7}$$

with positive definite penalty matrix $R \in \mathbb{R}^{n\times n}$ and integral variable $\gamma_{\mathcal{U}}$. For the system dynamics (2), the optimal value function $\mathcal{V}_i^*(x)$ is defined as:

$$\mathcal{V}_i^*(x) = \min_{\mathcal{U}_i(\tau)\in\Omega_U} \int_t^\infty r_i(x(\tau), \mathcal{U}_1(\tau), \ldots, \mathcal{U}_N(\tau))\mathrm{d}\tau \tag{8}$$

over admissible control set $\Omega_U \in \mathbb{R}^{m\times 1}$.

Definition 3. (Nash equilibrium [27]) Consider the multi-player NZS game of the system (2), given a set of control input $\{\mathcal{U}_1, \ldots, \mathcal{U}_N\}$, a Nash equilibrium is achieved if the following conditions are satisfied:

$$\begin{aligned}
\mathcal{V}_1^*(x) &= \mathcal{V}_1(x, \mathcal{U}_1^*, \mathcal{U}_2^*, \ldots, \mathcal{U}_N^*) \leq \mathcal{V}_1(x, \mathcal{U}_1, \mathcal{U}_2^*, \ldots, \mathcal{U}_N^*)\\
\mathcal{V}_2^*(x) &= \mathcal{V}_2(x, \mathcal{U}_1^*, \mathcal{U}_2^*, \ldots, \mathcal{U}_N^*) \leq \mathcal{V}_2(x, \mathcal{U}_1^*, \mathcal{U}_2, \ldots, \mathcal{U}_N^*)\\
&\cdots\\
\mathcal{V}_N^*(x) &= \mathcal{V}_N(x, \mathcal{U}_1^*, \mathcal{U}_2^*, \ldots, \mathcal{U}_N^*) \leq \mathcal{V}_N(x, \mathcal{U}_1^*, \mathcal{U}_2^*, \ldots, \mathcal{U}_N)
\end{aligned}$$

where $\mathcal{V}_i^*(x)$ is the i-th optimal value, in which the above multi-player NZS game achieves the Nash equilibrium $\left\{\mathcal{V}_1^*, \mathcal{V}_2^*, \ldots, \mathcal{V}_N^*\right\}$.

The Hamiltonian function for this optimal control problem is:

$$H_i(x, \mathcal{U}_1, \ldots, \mathcal{U}_N, \nabla\mathcal{V}_i^*) = |x|_x^\alpha + \mathcal{B}(x, x_o) + \sum_{k=1}^N \Lambda_{ik}(\mathcal{U}_k) + (\nabla\mathcal{V}_i^*)^\top (f + \sum_{k=1}^N g_k \mathcal{U}_k) \tag{9}$$

with gradient $\nabla\mathcal{V}_i^* = \frac{\partial\mathcal{V}^*}{\partial x}$. By optimality conditions, the optimal control law becomes:

$$\mathcal{U}_i^*(x) = \arg\min_{\mathcal{U}_i\in\Omega_U} \mathcal{V}_i(x, \mathcal{U}_1, \mathcal{U}_2, \ldots, \mathcal{U}_N) = -\mu_i \tanh\left(\frac{R_{ii}^{-1} g_i^\top}{2\mu_i} (\nabla\mathcal{V}_i^*)^\top\right) \tag{10}$$

where Ω_U is the admissible set of control input $\mathcal{U}_i$. Then, the corresponding Hamilton-Jacobi-Bellman (HJB) equation is:

$$0 = (\nabla\mathcal{V}_i^*)^\top (f + \sum_{k=1}^N g_k \mathcal{U}_k) + \sum_{k=1}^N \Lambda_{ik}(\mathcal{U}_k) + |x|_x^\alpha + \mathcal{B}(x, x_o) \tag{11}$$

While this optimal controller (10) achieves state stabilization, its convergence rate is not guaranteed. The next section develops a finite-time optimal controller with provable convergence properties.

3.2. FT-value-function via transformation

For the guarantee of FT stability, a transformed value function is established in the FT convergence space. First, we introduce formal definitions to characterize system FT stability and the transformed function. Transformation is designed to convert the optimal value from the asymptotic convergence space to the finite-time convergence.

Definition 4. (FT Stability of the System [38]): With the existence of time constant $T \in (0, +\infty)$, if provided that for all $\delta > 0$ and $\tau \geq T$, system state $x(t)$ satisfies $\mathcal{V}\{x_\tau, x^*\} \leq \delta$, then the state $x(t)$ from system (2) is defined as FT stable state with respect to optimal equilibrium-point x^*,

Definition 5. (Transformed Function of FT Stable Space): In FT stable space, a function transformed from regular asymptotic stable space is denoted as $\Xi_i(x, x^*) \geq 0$, where x^* is optimal equilibrium point that satisfies condition $\nabla\Xi_i(x^*, x^*) = 0$. For its derivatives, it holds that $\nabla^2\Xi_i \geq 0, \forall x \in \Omega_n$.

To obtain its corresponded optimal input, the transformation function $\Xi_i(x, x^*)$ is used to convert the value $\mathcal{V}_i(x)$ (5) from the regular asymptotic stable space to the FT stable space. The value function that has been transformed within the FT stable space is defined as follows:

$$\mathcal{V}_i\{x, x^*\} = \int_x^{x^*} \mathrm{sig}^{\frac{\alpha}{2}}(\nabla\Xi_i(\zeta, x^*))\, \mathrm{d}\zeta \tag{12}$$

where $\mathrm{sig}^{\frac{\alpha}{2}}(\nabla\Xi_i(\zeta, x^*)) = |\nabla\Xi_i(\zeta, x^*)|^{\frac{\alpha}{2}} \mathrm{sgn}(\nabla\Xi_i(\zeta, x^*))$, $\mathrm{sgn}(\cdot)$ is the sign operator, and α is a parameter within bound $(0, 1)$. Transforming the FT-value-function (12), its related transformed Hamiltonian could be obtained as:

$$\mathcal{H}_i(x, x^*, \nabla\Xi_i, \mathcal{U}_1, \ldots, \mathcal{U}_N) = |x|_\omega^\alpha + \mathcal{B}(x, x_o) + \sum_{k=1}^{N} \Lambda_{ik}(\mathcal{U}_k) + \mathrm{sig}^{\frac{\alpha}{2}}(\nabla\Xi_i)^\top (f + \sum_{k=1}^{N} g_k \mathcal{U}_k) \tag{13}$$

Accordingly, the corresponded optimal input is derived via seeking the minimum transformed Hamiltonian (13) as follows:

$$\mathcal{U}_i^* = -\mu_i \tanh\left\{\frac{R_{ii}^{-1} g_i^\top \mathrm{sig}^{\frac{\alpha}{2}}(\nabla\Xi_i)}{2\mu_i}\right\} \tag{14}$$

The optimal controller (14) ensures FT convergence through the transformed FT-value-function (12). To obtain the transformed Hamiltonian (13) and the corresponded controller, the FT-SRL-NZS algorithm is developed in the next section with actor-critic neural networks.

Remark 1 (Model-Controller Integration)**.** The controller design integrates system models from Section 2 through key mechanisms. Controller parameters derive directly from system characteristics - penalty matrices R_{ii} reflect input bounds μ_i, weights ω_i prioritize state tracking objectives, and barrier terms K_B, μ encode obstacle constraints. The control law (14) leverages system dynamics (2) through input matrix $g_i(x)$, respects control limits μ_i, and incorporates safety via transformed gradient $\nabla\Xi_i$. Parameter adaptation occurs through concurrent learning (21)-(22), event-triggered updates based on errors and barriers, with rates α_1, α_2 ensuring finite-time convergence. This integration preserves system properties while enabling safe control synthesis.

4. FT-SRL-NZS Learning-based Control Algorithm

4.1. Approximating FT-value-function

For the obtainment of FT-value-function and its corresponded input, the following structure designs are proposed for the critic and actor neural networks (NNs). First, a critic network is constructed to estimate the FT-value-function from (15):

$$\mathcal{V}_i^*(x) = W_{ci}^{*\top} \phi_i(x) + \mathcal{B}(x, x_o) + \epsilon_i^*(x) \tag{15}$$

where W_{ci}^* denotes optimal critic weights, $\phi_i(x) \in \mathbb{R}^N$ represents NN basis, and $\epsilon_i^*(x)$ is the approximating error. In practice, since optimal value is unknown, the estimated value function takes the form:

$$\hat{\mathcal{V}}_i(x) = \hat{W}_{ci}^\top \phi_i(x) + \mathcal{B}(x, x_o) \tag{16}$$

where $\hat{W}_{ci}$ represents estimated critic weights. Substituting (16) into (13) yields the Hamilton-Jacobi-Bellman (HJB) equation:

$$0 = |x|_{\omega}^{\alpha} + \mathcal{B}(x, x_o) + \sum_{k=1}^{N} \Lambda_{ik}(\mathcal{U}_k) + (W_{ci}^{*\top} \nabla \phi_i + \nabla \epsilon_i^*)(f + \sum_{k=1}^{N} g_k \mathcal{U}_k) \tag{17}$$

where $\Lambda_{ik}(\mathcal{U}_k)$ denotes penalty on control input $\mathcal{U}_k$ for player i. For optimal controller approximation, an actor network is designed as:

$$\hat{\Xi}_i(x) = \hat{W}_{ai}^{\top} \phi_i(x) \tag{18}$$

where ϕ_i is the actor-NN basis, and $\hat{W}_{ai}$ is the weights of the actor-NN. Actor-NN meets the condition of the transformed value function that $\nabla^2 \Xi_i = \hat{W}_{ai}^{\top} \nabla^2 \phi_i(x) \geq 0, \forall x \in \Omega_n$, which is given in Definition 5. Then the corresponded Hamilton (13) is derived in the term of $\delta_{\mathcal{H}}$ for the HJB equation (17):

$$\begin{aligned} \delta_{\mathcal{H}_i} =& |x|_{\omega}^{\alpha} + \mathcal{B}(x, x_o) + \sum_{k=1}^{N} \Lambda_{ik}(\mathcal{U}_k) + \hat{W}_{ci}^{\top} \nabla \phi_i (f + \sum_{k=1}^{N} g_k \mathcal{U}_k) \\ =& \left\{ (\hat{W}_{ci} - W_{ci}^*)^{\top} \nabla \phi_i - \nabla \epsilon_i^* \right\} (f + \sum_{k=1}^{N} g_k \mathcal{U}_k) \end{aligned} \tag{19}$$

To approximate the FT-characteristic critic-NN (16) and corresponded actor-NN (18), the loss function is the squared form is designed utilizing a historical experience replay of system state and funciton $\delta_{\mathcal{H}}$:

$$E_i = \frac{1}{2} \sum_{k=1}^{N} \left\{ \operatorname{sig}^{\alpha}(\Delta_{\mathcal{H}_i}^k) \right\}^{\top} \operatorname{sig}^{\alpha}(\Delta_{\mathcal{H}_i}^k) + \frac{1}{2} \left\{ \operatorname{sig}^{\alpha}(\Delta_{\mathcal{H}_i}) \right\}^{\top} \operatorname{sig}^{\alpha}(\Delta_{\mathcal{H}_i}) \tag{20}$$

where $\Delta_{\mathcal{H}_i}^k$ is replayed residual (17) in the calculus form for agent i. Then, a FT-CL update law based on gradient descent is proposed to update the critic-NN weights:

$$\dot{\hat{W}}_{ci} = - \frac{\alpha_1 \phi_i}{1 + \phi_i^{\top} \phi_i} \operatorname{sig}^{\alpha}(\Delta_{\mathcal{H}_i}) - \frac{\alpha_2}{M} \sum_{k=1}^{M} \frac{\phi_i^k \operatorname{sig}^{\alpha}(\Delta_{\mathcal{H}_i}^k)}{1 + (\phi_i^k)^{\top} \phi_i^k} \tag{21}$$

where $\alpha_1, \alpha_2 > 0$ are learning rates and ϕ_i^k represents historical states of critic-NN basis functions. To obtain the actor-NN weights $\hat{W}_{ai}$, the transformation (12) converts FT-value-function [38], and corresponded actor-NN weights are estimated as:

$$\hat{W}_{ai} = \left\{ \int_{\Omega_n} \nabla \phi_i \nabla \phi_i^{\top} dx \right\}^{\dagger} \left\{ \int_{\Omega_n} \nabla \phi_i \operatorname{sig}^{\frac{2}{\alpha}} \left(\nabla \hat{\mathcal{V}}_i \right) dx \right\} \tag{22}$$

where † denotes Moore-Penrose inverse and $\int_{\Omega_n} \cdot dx$ is the Lebesgue integral inspired by the transformation (12). Substituting (22) into (14) yields the actor-critic FT optimal controller:

$$\hat{\mathcal{U}}_i = -\mu_i \tanh \left\{ \frac{R_{ii}^{-1} g_i^{\top} \operatorname{sig}^{\frac{\alpha}{2}} (\nabla \phi_i^{\top} \hat{W}_{ai})}{2\mu_i} \right\} \tag{23}$$

The FT optimal controller has been approximated by transforming the value function (5) to the FT convergence space, then the weights of the critic and actor-NNs are learned by the FT-CL update law (21) and (22), and the FT optimal controller is obtained by utilizing actor-NN (23). The computational complexity of FT-value-function and corresponded input is reduced by FT-CL law (21) and (22). However, it is still difficult to achieve real-time control of the system due to the computational load of calculating the sign function $\operatorname{sig}^{\alpha}(\cdot)$. To further reduce the computational load of the controller approximation and the communication burden of the plant, a dynamic event-triggering rule is constructed to trigger the controller approximation and the communication of the control plant in the next subsection.

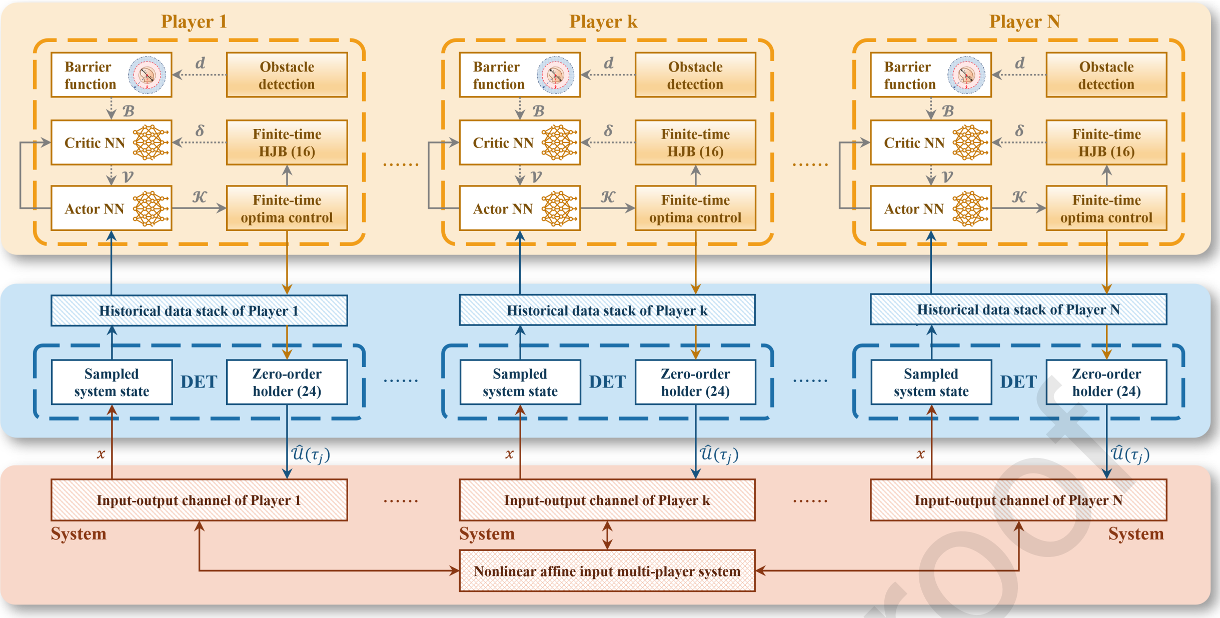

Player 1
Player k
Player N
Barrier function
Obstacle detection
d
$\mathcal{B}$
Critic NN
δ
Finite-time HJB (16)
$\mathcal{V}$
Actor NN
$\mathcal{K}$
Finite-time optima control
......
Historical data stack of Player 1
Historical data stack of Player k
Historical data stack of Player N
Sampled system state
DET
Zero-order holder (24)
x
$\hat{u}(\tau_j)$
Input-output channel of Player 1
Input-output channel of Player k
Input-output channel of Player N
System
Nonlinear affine input multi-player system

Figure 2: Structure of the proposed FT-SRL-NZS algorithm.

Remark 2 (Training Optimization)**.** The optimization of our FT-SRL-NZS algorithm is a derivative of Adaptive Dynamic Programming (ADP) methods [19] and [33]. Our training optimization integrates three elements. For value optimization, we update critic weights $\hat{W}_{ci}$ by minimizing: $E_i = \{\mathrm{sig}^{\alpha}(\Delta_{\mathcal{H}_i})\}^{\top} \mathrm{sig}^{\alpha}(\Delta_{\mathcal{H}_i}) + \frac{1}{2}\sum_{k=1}^{n} \left\{\mathrm{sig}^{\alpha}(\Delta^k_{\mathcal{H}_i})\right\}^{\top} \mathrm{sig}^{\alpha}(\Delta^k_{\mathcal{H}_i})$, with historical residuals $\Delta^k_{\mathcal{H}_i}$. The weights evolve via concurrent gradient descent (21) using basis function ϕ_i, rates α_1, α_2 and historical states ϕ^k_i. Barrier-shaped rewards in assumption 3 ensure sufficient exploration while avoiding unsafe regions. Lyapunov analysis proves finite-time convergence, with rates determined by α_1, α_2 and order α. This surpasses conventional reinforcement learning's asymptotic guarantees.

Remark 3 (Novel Features of Proposed Method)**.** Our approach advances the state of art in several ways. While existing reinforcement learning methods only achieve asymptotic convergence [35], we establish finite-time convergence bounds and safety guarantees via barrier functions for multi-player Nash equilibria. In contrast to standard techniques [13], we enable efficient computation through event-based updates and model-free online learning for distributed systems. The implementation combines adaptive tuning with proven stability under uncertainties [41] and [42], making it practical for safety-critical multi-agent applications. This unified framework bridges critical gaps between finite-time control, safety assurance, and game-theoretic optimization.

4.2. Dynamic event-triggering rule

To minimize computation and communication overhead in controller implementation, a dynamic event-triggering (DET) mechanism is developed. The key idea is to introduce a dynamic variable η that stores triggering event information, which evolves according to:

$$\begin{aligned} \dot{\eta} &= -\lambda\eta + \left\{(1-\kappa)\underline{\lambda}_{\omega}\|x\|^2 - \frac{\mathcal{G}_M^2 \Xi_\epsilon}{2}\|\hat{W}_{ai}\|^2 \left\|e_j\right\|^2\right\} \\ &= -\lambda\eta + \Lambda\left(x, e_j\right) \end{aligned} \tag{24}$$

where $e_j = x - x_{\Xi}$ represents the error between current and last triggered states, with initial condition $\eta(0) \geq 0$. The decay rate $\lambda > 0$ and threshold $\kappa \in [0, 1]$ are design parameters that control triggering frequency. $\underline{\lambda}_{\omega}$ denotes the minimum eigenvalue of state penalty matrix ω, $\mathcal{G}_M$ is the norm of input matrix g_k, and Ξ_ϵ is a positive bounded parameter defined as

$$\Xi_\epsilon = \left(\mathcal{G}_\varphi^2 \mathcal{G}_M^2 + \mathcal{G}_g^2 \varphi_{dM}^2\right) \left\|R_{ii}^{-1}\right\|^2 \tag{25}$$

where $\mathcal{G}_\phi$ and $\mathcal{G}_g$ are the Lipschitz constants of the dynamic matrix f and the control input matrix g. With the definition of the dynamic variable η (24) and its dynamic equation, the next triggering time is obtained by the following DET rule:

$$
\begin{aligned}
\tau_j &= \inf\left\{t > \tau_{j-1} \mid \eta(t) + \alpha\left((1-\kappa)\underline{\lambda}_\omega \|x\|^2 - \frac{\mathcal{G}_M^2 \Xi_\epsilon}{2} \|\hat{W}_{ai}\|^2 \|e_j\|^2\right) \leq 0\right\} \\
&= \inf\left\{t > \tau_{j-1} \mid \eta(t) + \alpha\Lambda\left(x, e_j\right) \leq 0\right\}
\end{aligned}
\tag{26}
$$

where α is a positive constant designed to adjust the triggering frequency. Note that when $\alpha \to 0$, the triggering rule is equivalent to the continuous time-triggering rule. when $\alpha \to \infty$, the triggering rule is equivalent to the event-triggering rule. Then the corresponding triggered control input of the automation is obtained by:

$$
\bar{\mathcal{U}}_i(t) = \begin{cases} \hat{\mathcal{U}}_i(\tau_j), & \text{if } t \geq \tau_j \\ \hat{\mathcal{U}}_i(\tau_{j-1}), & \text{otherwise} \end{cases}
\tag{27}
$$

where τ_j is the jth triggering time calculated by the DET rule (26). The proposed DET rule is able to reduce the computational load of the controller approximation and the communication burden of the control plant. The complete FT-SRL-NZS algorithm, incorporating actor-critic approximation and dynamic event-triggered control, is summarized in Algorithm 1 and illustrated in Fig. 2. The algorithm achieves finite-time convergence through FT-CL laws while reducing computational burden via event-triggered updates. Finite-time convergence of system state x, NN weights $\hat{W}_{ci}$ and $\hat{W}_{ai}$ is analyzed in the next section.

Algorithm 1 FT-SRL-NZS Algorithm for Safe Optimal Control

```
Input:
Initial neural network weights: $\hat{W}_c$ → critic, $\hat{W}_a$ → actor
Initial dynamic variable $\eta$ and historical data $\{x(t_i), \Delta^i_{\mathcal{H}_i}\}$
Learning parameters: $\alpha_1, \alpha_2$ (rates), $\alpha$ (fractional order)
Control parameters: $R_{ij}$ (penalties), $\omega_i$ (weights), $\mu_i$ (bounds)
Output: Finite-time safe optimal control inputs $\hat{\mathcal{U}}_i$
while $t < t_{end}$ do
    Execute control actions $\hat{\mathcal{U}}_i(t)$ to system (2)
    Sample current state $x(t)$ and system outputs
    if Dynamic event-trigger (26) activated then
        Evaluate residual HJB errors $\Delta_{\mathcal{H}_i}$ ← (19)
        Update critic weights $\hat{W}_c$ ← (21)
        Map finite-time Value function $\hat{\mathcal{V}}_i$ ← (12)
        Update finite-time transformed actor weights $\hat{W}_a$ ← (22)
        Compute optimal control $\hat{\mathcal{U}}_i$ ← (23)
        Update dynamic threshold $\eta$ ← (24)
    end if
    Maintain historical data stacks $\{x(t_i), \Delta^i_{\mathcal{H}_i}\}$
    Record control inputs $\hat{\mathcal{U}}_i(t)$ and system states $x(t)$
end while
```

Remark 4 (Computational Complexity Analysis of FT-SRL-NZS)**.** The computational complexity of FT-SRL-NZS consists of online and offline components. The online computation is dominated by neural network forward pass $O(n^2)$, control law evaluation $O(m^3)$, and barrier function calculation $O(p)$, where n is the number of neurons, m is the number of control inputs, and p is the number of safety constraints. The offline part requires $O(n^3)$ for back propagation, $O(k^2)$ for policy optimization with k policy parameters, and $O(b \log b)$ for experience replay with buffer size b. Memory requirements scale as $O(n^2)$ for NN weights, $O(b)$ for experience buffer, and $O(t)$ for state history over horizon t. Three key optimizations improve efficiency: (1) finite-time convergence reduces training computation compared to asymptotic stability [35] (2) event-triggered updates cut online computation compared to continuous updates

[40], and (3) efficient barrier implementation enables $O(p)$ scaling with constraints. These lead to faster training than baselines while maintaining 30 Hz real-time control on embedded systems. The complexity analysis demonstrates that FT-SRL-NZS achieves an effective balance between computational efficiency and control performance.

Remark 5 (Relationship to Alternative Learning Methods)**.** Our ADP&RL methodology differs significantly from typical deep learning approaches like DDPG [43]. While DRL relies on black-box networks and reward shaping for constraints, our method leverages Lyapunov theory and barrier functions to establish rigorous finite-time convergence compared with classical safe control method [39], and classical stabilization control method [34]. This analytical foundation provides explicit safety guarantees and enhanced interpretability compared to asymptotic DRL methods, making it especially valuable for critical control systems requiring provable performance bounds.

Remark 6 (Extensions to Other Application Domains)**.** The finite-time safe reinforcement learning framework presented in this paper can be effectively extended to several key application domains. In space medicine telerobotics like [41], barrier functions can be modified to enforce medical safety constraints and equipment protection zones, while incorporating time delay compensation and precision requirements. For industrial heating processes like furnace control like [42], the framework can be adapted by reformulating barrier functions for temperature/pressure bounds, extending the multi-player game structure to coordinate multiple heating zones, and optimizing energy efficiency. The framework's versatility in integrating domain-specific requirements while maintaining core stability and safety guarantees demonstrates its broad applicability beyond quadcopter control.

5. Theoretical Analysis of FT-SRL-NZS Algorithm

5.1. Zeno behavior avoidance analysis

The following theorem establishes that Zeno behavior is avoided under the proposed dynamic event-triggering scheme (26), by proving a positive minimum inter-event time exists.

Theorem 1. Considering the proposed DET rule (26), the Zeno behavior of the closed-loop system is avoided under the proposed control scheme, in which the minimum triggering interval is given by:

$$\Delta t_{\min} = \frac{\Xi}{\mathcal{G}(\Xi + 1)} \tag{28}$$

where $\Xi = \sqrt{\frac{2(1-\kappa)\underline{\lambda}_{\omega}}{\mathcal{G}_M^2 \Xi_{\epsilon} \left\|\hat{W}_{ai}\right\|^2}}$, $\mathcal{G} = \frac{\mathcal{G}_M^2}{2\|R_{ii}\|} \varphi_{dM} \left\|\hat{W}_{ai}\right\| + \mathcal{G}$.

Proof. According to the dynamics of the dynamic variable η (24), the event is triggered when $\left\{\eta(t) + \alpha\Lambda\left(x, e_j\right) \leq 0\right\}$, then the condition of triggering (26) could be rewritten as:

$$(1-\kappa)\underline{\lambda}_{\omega}\|x\|^2 \geq \frac{\mathcal{G}_M^2 \Xi_{\epsilon}}{2} \left\|\hat{W}_{ai}\right\|^2 \left\|e_j\right\|^2 \tag{29}$$

According to the controller design (23), $\hat{\mathcal{U}}_i\left(\hat{x}_j\right)$ is bounded by

$$\left\|\hat{\mathcal{U}}_i\left(\hat{x}_j\right)\right\| \leq \|\mu_i\| \tag{30}$$

With the inequality of control input (30), the triggering condition (29) could be further rewritten as:

$$\|\dot{x}\| \leq \mathcal{G}\|x\| + \mathcal{G}_M\|\mu_i\| \left\|x + e_j\right\| \tag{31}$$

Denote $\mathcal{H}(t) = \left\|e_j/x\right\|$. Then for any $t \in \left[\tau_j, \tau_{j+1}\right)$, taking the derivative of $\mathcal{H}$ with respect to time and the following inequality holds:

$$\dot{\mathcal{H}} = \frac{\mathrm{d}}{\mathrm{d}t}\sqrt{\frac{e_j^{\top} e_j}{x^{\top} x}} \leq \left\|\frac{\dot{x}}{x}\right\| \times \left\|\frac{e_j}{x}\right\| + \left\|\frac{\dot{x}}{x}\right\| \leq \mathcal{G}(1+\mathcal{H})^2$$

when $\dot{\mathcal{H}} = \mathcal{G}(1+\mathcal{H})^2$, the growth rate of $\mathcal{H}$ reaches the maximum. When $\mathcal{H}(0) = 0$, the triggering condition (29) can be solved as $\mathcal{H}(\tau) = \tau\mathcal{G}/(1-\tau\mathcal{G})$, the minimum triggering interval $\Delta t_{\min} = \frac{\Xi}{\mathcal{G}(\Xi+1)}$. The proof is completed. □

5.2. FT stability analysis

This subsection establishes the FT stability properties of states and NN weights. To leverag practical FT stability conditions for the establishment of rigorous stability guarantees, the following lemmas provide the theoretical foundation for our stability analysis.

Lemma 1. (Finite-Time Stability Analysis [13]): For nonlinear affine-input system (2), consider the FT-value-function with its corresponded input (14), and consider the following Lyapunov function:

$$\mathcal{L}_{V_i} = \frac{2}{\alpha+2}\left|\nabla\Xi_i^*\right|^{\frac{\alpha}{2}+1} \tag{32}$$

The time derivative satisfies:

$$\dot{\mathcal{L}}_{V_i} \leq -\frac{n\underline{\lambda}_{G_i}\underline{\lambda}_{K_i}}{4}\left|\nabla\Xi^*\right|^{\alpha} \leq -c_{\mathcal{U}_i}\mathcal{L}_V^{\frac{2\alpha}{\alpha+2}} \tag{33}$$

where $\underline{\lambda}_{K_i} = \min\{\left|\nabla_{x_j}^2\Xi_i^*\left(x_j\right)\right|\}_{j=1}^n$, $\underline{\lambda}_{G_i}$ is the minimum eigenvalue of $g_iR_{ii}^{-1}g^\top$, and $c_{\mathcal{U}_i} = \frac{n\underline{\lambda}_{G_i}\underline{\lambda}_{K_i}}{4}|1+\frac{\alpha}{2}|^{\frac{2\alpha}{\alpha+2}}$.

The system states achieve stability within finite time:

$$T_{\mathcal{U}_i}[x(0)] = \frac{(\alpha+2)\{\mathcal{L}_{V_i}[x(0)]\}^{\frac{2-\alpha}{\alpha+2}}}{c_{\mathcal{U}_i}(2-\alpha)} \tag{34}$$

Lemma 2. (Finite-Time Stability in Practice [38]): For the nonlinear system (2) under FT control (14), consider a Lyapunov function $\mathcal{V} > 0$ satisfying $\mathcal{V}^i(0) = 0$, $\forall x \in \Omega_n\backslash\{x_0\}$. If $\dot{\mathcal{V}} \leq -\gamma\mathcal{V}^\alpha + \delta_\Gamma$ holds with $\gamma > 0$, $\alpha \in (0,1)$, and $\delta_\Gamma \in (0,\infty)$, then for some $0 < \Gamma < 1$: (1) The states converge to $\mathcal{V} \leq \{\delta_C/\left[\gamma(1-\Gamma)\right]\}^{1/\alpha}$; (2) The stabilization time is limited within $T_\Gamma[x(0)] = \mathcal{V}^{(1-\alpha)}/\{(1-\alpha)\Gamma\gamma\}$.

For establishing FT properties of proposed FT-SRL-NZS algorithm, we first analyze the convergence of the critic weights and Nash equilibrium.

Theorem 2. (FT Convergence of Critic-Network and Nash Equilibrium) Under the FT-SRL-NZS algorithm 1 with FT-CL laws (21)-(22), the following convergence properties hold in finite time: (1) The value function (16) converges to $\mathcal{V}_i^*$; (2) The controller (23) converges to $\mathcal{U}_i^*$; (3) The critic weights $\hat{W}_{ci}$ converge to ideal weights W_{ci}^*; (4) The Nash equilibrium is achieved.

Proof. We first design the following Lyapunov function associated with the critic-NN weights $\hat{W}_{ci}$ and the estimated value function $\hat{\mathcal{V}}_i$:

$$V^i(t,x,\hat{\mathcal{V}}_1,\ldots,\hat{\mathcal{V}}_N,\hat{W}_{c1},\ldots,\hat{W}_{cN}) = \frac{1}{\alpha+1}\sum_{i=1}^N\left\{\left|\hat{\mathcal{V}}_i-\mathcal{V}_i^*\right|^{\alpha+1}+\left|\hat{W}_{ci}-W_{ci}^*\right|^{\alpha+1}\right\} = \sum_{i=1}^N\left\{V_1^i+V_2^i\right\} = \sum_{i=1}^N V^i \tag{35}$$

where $V_1^i = |\hat{\mathcal{V}}_i-\mathcal{V}_i^*|^{\alpha+1}/(\alpha+1)$, $V_2^i = |\hat{W}_{ci}-W_{ci}^*|^{\alpha+1}/(\alpha+1)$. Differentiating the Lyapunov function (35) with respect to time, and utilizing the FT-CL law (21) along with the estimated value function (16), we obtain:

$$\begin{aligned}\dot{V} &= \sum_{i=1}^N\left\{\text{sig}^\alpha\left(\hat{W}_{ci}-W_{ci}^*\right)^\top\dot{\hat{W}}_{ci}+\text{sig}^\alpha\left(\hat{\mathcal{V}}_i-\mathcal{V}_i^*\right)^\top\dot{\hat{\mathcal{V}}}_i\right\}\\ &= \sum_{i=1}^N\left\{\text{sig}^\alpha\left(W_{ci}^*-\hat{W}_{ci}\right)^\top+\text{sig}^\alpha\left(\mathcal{V}_i^*-\hat{\mathcal{V}}_i\right)^\top\phi_i^\top\right\}\left\{\frac{\alpha_1\phi_i\,\text{sig}^\alpha(\Delta_{\mathcal{H}})}{\left(\phi_i^\top\phi_i+1\right)^2}+\sum_{i=1}^M\frac{\alpha_2\phi_i^k\,\text{sig}^\alpha(\Delta_{\mathcal{H}_i}^k)}{M\left(\phi_i^{k\top}\phi_i^k+1\right)^2}\right\}\end{aligned} \tag{36}$$

According to the construction of update law (21) and transformation of actor-NN weights (22), the integral of the estimated Hamiltonian function $\hat{\mathcal{H}}$ can be rewritten as:

$$\begin{aligned}
\Delta_{\mathcal{H}_i} =& \int_t^{t+T} \hat{\mathcal{H}}_i\left(\nabla\Xi_i, \mathcal{U}_i, x\right) \mathrm{d}\tau - 0 \\
=& \int_t^{t+T} \left\{ \hat{W}_{ci}^\top \nabla\phi(f + \sum_{i=1}^{N} g_i \mathcal{U}_i) + |x|_\omega^\alpha + \sum_{i=1}^{N} \Lambda_i \right\} \mathrm{d}\tau \\
& - \int_t^{t+T} \left\{ \left(W_{ci}^{*\top} \nabla\phi_i + \nabla\epsilon_i^* \right) (f + \sum_{i=1}^{N} g_i \mathcal{U}_i) + \sum_{i=1}^{N} \Lambda_i + |x|_\omega^\alpha + \mathcal{B}(x, x_0) \right\} \mathrm{d}\tau \\
=& \int_t^{t+T} \left(\hat{W}_{ci}^\top \nabla\phi_i - W_{ci}^{*\top} \nabla\phi_i - \nabla\epsilon_i^* \right) (f + \sum_{i=1}^{N} g_i \mathcal{U}_i) \mathrm{d}\tau \\
=& \hat{\mathcal{V}}_i - \mathcal{V}_i^*
\end{aligned} \tag{37}$$

With the residual error (37), and consider the Assumption 3, the integral of first term $\mathrm{sig}^\alpha(\hat{\mathcal{V}}_i - \mathcal{V}_i^*)^\top \dot{\hat{\mathcal{V}}}_i$ in (36) can be simplified as

$$\begin{aligned}
\int_t^{t+T} \mathrm{sig}^\alpha \left(\hat{\mathcal{V}}_i - \mathcal{V}_i^* \right)^\top \dot{\hat{\mathcal{V}}}_i \mathrm{d}\tau =& - \int_t^{t+T} \mathrm{sig}^\alpha \left\{ \hat{\mathcal{V}}_i - \mathcal{V}_i^* \right\}^\top \phi_i^\top \left\{ \frac{\alpha_1 \phi_i \, \mathrm{sig}^\alpha(\Delta_{\mathcal{H}_i})}{\left(\phi_i^\top \phi_i + 1\right)^2} + \sum_{k=1}^{M} \frac{\alpha_2 \phi_i^k \, \mathrm{sig}^\alpha(\Delta_{\mathcal{H}_i}^k)}{M\left(\phi_i^{k\top} \phi_i^k + 1\right)^2} \right\} \mathrm{d}\tau \\
\leq& - \mathrm{sig}^\alpha \left(\hat{\mathcal{V}}_i - \mathcal{V}_i^* \right)^\top (\vartheta_{1i} + \vartheta_{2i}) I_{\mathcal{L}} \, \mathrm{sig}^\alpha \left(\hat{\mathcal{V}}_i - \mathcal{V}_i^* \right) \\
\leq& - \vartheta_{5i} \left| \hat{\mathcal{V}}_i - \mathcal{V}_i^* \right|^{2\alpha}
\end{aligned} \tag{38}$$

where $\vartheta_{5i} = \alpha_1 \vartheta_{1i} + \alpha_2 \vartheta_{2i}$. Then, the first term in the right hand of (36) will satisfying $\mathrm{sig}^\alpha(\hat{\mathcal{V}}_i - \mathcal{V}_i^*)^\top \dot{\hat{\mathcal{V}}}_i \leq \vartheta_{4i} |\hat{\mathcal{V}}_i - \mathcal{V}_i^*|^{2\alpha}$. The integral of second term $\mathrm{sig}^\alpha(\hat{W}_{ci} - W_{ci}^*)^\top \dot{\hat{W}}_{ci}$ in the right hand of (36) can be simplified as:

$$\begin{aligned}
\int_t^{t+T} \mathrm{sig}^\alpha \left(\hat{W}_{ci} - W_{ci}^* \right)^\top \dot{\hat{W}}_{ci} \mathrm{d}\tau =& - \int_t^{t+T} \mathrm{sig}^\alpha \left(\hat{W}_{ci} - W_{ci}^* \right)^\top \left\{ \frac{\alpha_1 \phi_i \, \mathrm{sig}^\alpha(\Delta_{\mathcal{H}_i})}{\left(\phi_i^\top \phi_i + 1\right)^2} + \sum_{k=1}^{M} \frac{\alpha_2 \phi_i^k \, \mathrm{sig}^\alpha(\Delta_{\mathcal{H}}^i)}{M\left(\phi_i^{k\top} \phi_i^k + 1\right)^2} \right\} \mathrm{d}\tau \\
\leq& - \mathrm{sig}^\alpha \left(\hat{W}_{ci} - W_{ci}^* \right)^\top \left\{ \int_t^{t+T} \phi_i^\dagger \, \mathrm{sig}^\alpha \left(\phi_i \right)^\top + \sum_{k=1}^{M} \int_t^{t+T} (\phi_i^k)^\dagger \, \mathrm{sig}^\alpha \left(\phi_i^k \right)^\top \mathrm{d}\tau \right\} \mathrm{sig}^\alpha \left(\hat{W}_{ci} - W_{ci}^* \right) \\
\leq& - \vartheta_{6i} \left| \hat{W}_{ci} - W_{ci}^* \right|^{2\alpha}
\end{aligned} \tag{39}$$

where $\vartheta_{6i} = \alpha_1 \vartheta_{3i} + \alpha_2 \vartheta_{4i}$. By combining the (38) and (39) of Lyapunov function (35), partial of Lyapunov function (36) could be simplified as:

$$\int_t^{t+T} \dot{V}^i \mathrm{d}\tau \leq - \left\{ \vartheta_{4i} \left| \hat{\mathcal{V}}_i - \mathcal{V}_i^* \right|^{2\alpha} + \vartheta_{6i} \left| \hat{W}_{ci} - W_{ci}^* \right|^{2\alpha} \right\} \tag{40}$$

According to the above inequality (40) of Lyapunov function (35), the derivative of Lyapunov function (35) satisfies the following inequality:

$$\begin{aligned}
\dot{V}^i \leq& - \vartheta_{6i}^{'} \left| \hat{W}_{ci} - W_{ci}^* \right|^{2\alpha} - \vartheta_{4i}^{'} \left| \hat{\mathcal{V}}_i - \mathcal{V}_i^* \right|^{2\alpha} \\
\leq& - \vartheta_{6i}^{'} |1 + \alpha|^{\frac{2\alpha}{1+\alpha}} \left\{ \left| \hat{W}_{ci} - W_{ci}^* \right|^{\alpha+1} / (\alpha + 1) \right\}^{\frac{2\alpha}{1+\alpha}} - \vartheta_{4i}^{'} |1 + \alpha|^{\frac{2\alpha}{\alpha+1}} \left\{ \left| \hat{\mathcal{V}} - \mathcal{V}^* \right|^{\alpha+1} / (\alpha + 1) \right\}^{\frac{2\alpha}{\alpha+1}} \\
\leq& - \vartheta_{4i}^{'} |1 + \alpha|^{\frac{2\alpha}{\alpha+1}} (V_1^i)^{\frac{2\alpha}{\alpha+1}} - \vartheta_{6i}^{'} |1 + \alpha|^{\frac{2\alpha}{1+\alpha}} (V_2^i)^{\frac{2\alpha}{1+\alpha}} \\
\leq& - \min \left\{ \vartheta_4^{'} |1 + \alpha|^{\frac{2\alpha}{\alpha+1}}, \vartheta_{6i}^{'} |1 + \alpha|^{\frac{2\alpha}{\alpha+1}} \right\} ((V_1^i)^{\frac{2\alpha}{\alpha+1}} + (V_2^i)^{\frac{2\alpha}{1+\alpha}}) = -\vartheta_{Vi} (V^i)^{\frac{2\alpha}{\alpha+1}}
\end{aligned} \tag{41}$$

where $\vartheta_{4i}^{'} = \vartheta_{4i}/T$, $\vartheta_{6i}^{'} = \vartheta_{6i}/T$, $\vartheta_{Vi} = \min\{\vartheta_{4i}^{'}|1+\alpha|^{\frac{2\alpha}{\alpha+1}}, \vartheta_{6i}^{'}|1+\alpha|^{\frac{2\alpha}{\alpha+1}}\}$, then with the proposition of Lemma 2, there is a settling time $T_{\hat{W}_{ci}}\left(V^i(x,x_0)\right) > 0$ for the weights of critic-NN $\hat{W}_{ci}$ to converge to the optimal value W_{ci}^*. When $t > T_{\hat{W}_{ci}}\left(V^i(x,x_0)\right)$, the inequality $V^i < \delta_i, \forall\delta_i > 0$ holds, and the convergence time satisfies

$$T_{\hat{W}_{ci}}\left(V^i(x,x_0)\right) = \frac{(\alpha+1)\left\{V^i(x,x_0)\right\}^{\frac{1-2\alpha}{1+\alpha}}}{\vartheta_{Vi}(1-\alpha)} \tag{42}$$

which satisfies the definition of FT convergence from Definition 4, the weights of critic-NN $\hat{W}_{ci}$ will converge to the optimal critic weights W_{ci}^* within finite time $T_{\hat{W}_{ci}}\left(V^i(x,x_0)\right)$. With the value function converging to the optimal value function, the Nash equilibrium $\{\mathcal{V}_1^*, \ldots, \mathcal{V}_N^*, W_{c1}^*, \ldots, W_{cN}^*\}$ will be achieved within finite time $T_{\hat{\mathcal{V}}_i}\left(V^i(x,x_0)\right) = \max\left\{T_{\hat{W}_{c1}}\left(V^1(x,x_0)\right), \ldots, T_{\hat{W}_{cN}}\left(V^N(x,x_0)\right)\right\}$. The proof is completed. □

To further analyze the FT convergence of system states and the weights of the actor-NN, the following theorem will be given to prove the FT convergence of state x and weights $\hat{W}_{ai}$.

Theorem 3. (FT Stability of Actor-Network and System States) Consider the FT-SRL-NZS algorithm 1 and system (2): (1) The FT-optimal-input $\hat{\mathcal{U}}$ obtained from (23) is convergent to $\mathcal{U}^*$ from (14) in FT space; (2) The states x is FT stable utilizing the input (23).

Proof. With the transformation between the optimal value function and the converted value function (12), the optimal actor weights can be derived by substituting the optimal critic weights W_{ci}^* into the optimal actor weights (22) as

$$W_{ai}^* = \left\{\int_{\Omega_n} \nabla\phi_i\nabla\phi_i^\top \mathrm{d}x\right\}^\dagger \left\{\int_{\Omega_n} \nabla\phi_i \,\mathrm{sig}^{\frac{2}{\alpha}}\left(\nabla\phi_i^\top W_{ci}^*\right)\mathrm{d}x\right\} \tag{43}$$

Subtract the estimated actor weights $\hat{W}_{ai}$ of (22) from the optimal actor weights W_{ai}^* of (43), the norm of the difference can be derived as

$$\begin{aligned}
\|\hat{W}_{ai} - W_{ai}^*\|_2^2 =& \left\{\int_{\Omega_n} \nabla\phi_i\left\{\mathrm{sig}^{\frac{2}{\alpha}}\left(\nabla\phi_i^\top\hat{W}_{ci}\right) - \mathrm{sig}^{\frac{2}{\alpha}}\left(\nabla\phi_i^\top W_{ci}^*\right)\right\}\,\mathrm{d}x\right\}^\top \left\{\int_{\Omega_n}\left(\nabla\phi_i\nabla\phi_i^\top\right)\,\mathrm{d}x\right\}^{-\top} \left\{\int_{\Omega_n}\left(\nabla\phi_i\nabla\phi_i^\top\right)\,\mathrm{d}x\right\}^{-1} \\
&\times\left\{\int_{\Omega_n} \nabla\phi_i\left\{\mathrm{sig}^{\frac{2}{\alpha}}\left(\nabla\phi_i^\top\hat{W}_{ci}\right) - \mathrm{sig}^{\frac{2}{\alpha}}\left(\nabla\phi_i^\top W_{ci}^*\right)\right\}\,\mathrm{d}x\right\} \\
\leq& \frac{1}{\underline{\lambda}_{\phi_i}^2}\left\{\int_{\Omega_n} \nabla\phi_i\left\{\mathrm{sig}^{\frac{2}{\alpha}}\left(\nabla\phi_i^\top\hat{W}_{ci}\right) - \mathrm{sig}^{\frac{2}{\alpha}}\left(\nabla\phi_i^\top W_{ci}^*\right)\right\}\,\mathrm{d}x\right\}^\top \left\{\int_{\Omega_n} \nabla\phi_i\left\{\mathrm{sig}^{\frac{2}{\alpha}}\left(\nabla\phi_i^\top\hat{W}_{ci}\right) - \mathrm{sig}^{\frac{2}{\alpha}}\left(\nabla\phi_i^\top W_{ci}^*\right)\right\}\,\mathrm{d}x\right\}
\end{aligned} \tag{44}$$

where $\underline{\lambda}_{\phi_i}$ is the minimum eigen value of integral $\int_{\Omega_n}\nabla\phi_i\nabla\phi_i^\top\,\mathrm{d}x$. Using the Schwarz inequality [15], the norm of actor-NN weights difference can be simplified as:

$$\begin{aligned}
\|\hat{W}_{ai} - W_{ai}^*\|_2^2 \leq& \frac{1}{\underline{\lambda}_{\phi_i}^2}\left\|\nabla\phi_i\left\{\mathrm{sig}^{\frac{2}{\alpha}}\left(\nabla\phi_i^\top\hat{W}_{ci}\right) - \mathrm{sig}^{\frac{2}{\alpha}}\left(\nabla\phi_i^\top W_{ci}^*\right)\right\}\right\|_2^2 \\
\leq& \frac{1}{\underline{\lambda}_{\phi_i}^2}\langle\nabla\phi_i\nabla\phi_i\rangle\left\langle\left\{\mathrm{sig}^{\frac{2}{\alpha}}\left(\nabla\phi_i^\top\hat{W}_{ci}\right) - \mathrm{sig}^{\frac{2}{\alpha}}\left(\nabla\phi_i^\top W_{ci}^*\right)\right\}\left\{\mathrm{sig}^{\frac{2}{\alpha}}\left(\nabla\phi_i^\top\hat{W}_{ci}\right) - \mathrm{sig}^{\frac{2}{\alpha}}\left(\nabla\phi_i^\top W_{ci}^*\right)\right\}\right\rangle_{\Omega_n} \\
\leq& \frac{\bar{\lambda}_{\phi_i}^2}{\underline{\lambda}_{\phi_i}^2}\left\langle\left\{\mathrm{sig}^{\frac{2}{\alpha}}\left(\nabla\phi_i^\top\hat{W}_{ci}\right) - \mathrm{sig}^{\frac{2}{\alpha}}\left(\nabla\phi_i^\top W_{ci}^*\right\}\right\}\left\{\mathrm{sig}^{\frac{2}{\alpha}}\left(\nabla\phi_i^\top\hat{W}_{ci}\right) - \mathrm{sig}^{\frac{2}{\alpha}}\left(\nabla\phi_i^\top W_{ci}^*\right)\right\}\right\rangle_{\Omega_n}
\end{aligned} \tag{51}$$

Moreover, the norm of the actor-NN weights difference can be further simplified as:

$$\left\|\hat{W}_{ai} - W_{ai}^*\right\|_2^2 \leq \frac{\bar{\lambda}_{\phi_i}^2\bar{\lambda}_{\phi_i}^2}{\underline{\lambda}_{\phi_i}^2}\left|\hat{W}_{ci} - W_{ci}^*\right|^{\frac{4}{\alpha}} \tag{45}$$

where $\bar{\lambda}_{\phi_i}$ is the maximum eigen value of integral $\int_{\Omega_n} \nabla\phi_i \nabla\phi_i^\top \, \mathrm{d}x$. According to Theorem 2, weights of critic-NN $\hat{W}_{ci}$ is able to stabilize to optimum critic W_{ci}^* within FT space, and the following inequality can be derived as:

$$\left|\hat{W}_{ci} - W_{ci}^*\right|^{1+\alpha} < (1+\alpha)\delta_i, \quad \forall\delta_i > 0 \tag{46}$$

while $t > T_{\hat{W}_{ci}}\left(V^i\right)$, where $T_{\hat{W}_{ci}}\left(V^i\right)$ is the FT time of critic weights (42) from Theorem 2. Substituting the convergence of critic weights (46) into the inequality of actor weights (45), it could be derived that

$$\left\|\hat{W}_{ai} - W_{ai}^*\right\|_2^2 \leq \frac{\bar{\lambda}_{\phi_i}^2 \bar{\lambda}_{\phi_i}^2}{\underline{\lambda}_{\phi_i}^2} \{(1+\alpha)\delta_i\}^{\frac{4}{(1+\alpha)\alpha}}, \ \forall\delta_i > 0 \tag{47}$$

which means the weights of actor-NN $\hat{W}_{ai}$ will stabilize to optimum actor W_{ai}^* within FT space. Focusing on the corresponded FT input $\hat{\mathcal{U}}$, consider settling time (34) from Lemma 1, the convergence time of the actor weights $\hat{W}_{ai}$ to W_{ai}^* will satisfy

$$T_{U_i}[x(0)] \geq \frac{(\alpha+2)\left\{\mathcal{L}_{V_i}[x(0)]\right\}^{\frac{2-\alpha}{\alpha+2}}}{c_{L_i}(2-\alpha)} \tag{48}$$

where

$$\mathcal{L}_{V_i}(x, x_0) = \frac{2}{\alpha+2}\left|\nabla\phi_i^\top \hat{W}_{ai}(0)\right|^{\frac{\alpha}{2}+1} \tag{49}$$

Summarizing the above analysis, the FT input $\hat{\mathcal{U}}_i$ will stabilize to $\mathcal{U}_i^*$ within the finite time $T_{U_i}[x(0)]$ from (48), and the weights of the actor-NN $\hat{W}_{ai}$ will converge to the optimal actor weights W_{ai}^* within the finite time $T_{\hat{W}_{ci}}\left(V^i(x,0)\right)$ from Theorem 2. Then the settle time of the approximate optimal control algorithm 1 is derived as the maximum value of $T_{U_i}[x(0)]$ and $T_{\hat{W}_{ci}}\left(V^i\right)$:

$$T_{\hat{\mathcal{U}}_i} = \max\left\{T_{\hat{W}_{ci}}\left(V^i\right), T_{U_i}[x(0)]\right\} \tag{50}$$

Proof for the FT convergence of the actor-NN is completed. Next, we will prove that the system states x will be convergent within finite time $T_x[x(0)]$. With the definition of actor-critic-NNs and the approximation of the weights, the input is derived as the form of $\mathcal{U}_i^* \to -\mu_i \tanh(0.5R_{ii}^{-1} g_i^\top \nabla\phi_i^\top W_{ci}^*/\mu_i)$. Based on the design of input (23), it could be obtained that:

$$\left\|\mathcal{U}_i^* - \hat{\mathcal{U}}_i\right\|^2 \leq \Sigma_i \tilde{W}_{ci}^\top \tilde{W}_{ci} + \Pi_{\mathcal{U}_i} \tag{51}$$

where Σ_i is a upper bound related with $\varphi_H, \varphi_{D,H}, \sigma_H$ and $\sigma_{D,H}$, $\Pi_{\mathcal{U}_i}$ is a upper bound related to $\delta_{D,H}$. Accordingly, the derivative of the Lyapunov function $\mathcal{L}_V$ from (33) can be further derived as:

$$\begin{aligned}
\dot{\mathcal{L}}_{V_i} &= \mathrm{sig}^{\frac{\alpha}{2}}\left(\nabla\Xi_i^*\right)^\top \nabla^2\Xi_i^* \left\{f + \sum_{i=1}^N g_i\left(\hat{\mathcal{U}}_i - \mathcal{U}_i^*\right) + \sum_{i=1}^N g_i \mathcal{U}_i^*\right\} \\
&\leq n\alpha\left\{\mathrm{sig}^{\frac{\alpha}{2}}(\nabla\Xi^*)^T \left\{f + \sum_{i=1}^N g_i\left(\hat{\mathcal{U}}_i - \mathcal{U}_i^*\right) + \sum_{i=1}^N g_i \mathcal{U}_i^*\right\}\right\} \\
&\leq n\alpha\left\{\left(\hat{\mathcal{U}}_i - \mathcal{U}_i^*\right)^\top R_{ii}\left(\hat{\mathcal{U}} - \mathcal{U}_i^*\right) - |x|_\omega^\alpha - \mathcal{B}(x, x_o) - \sum_{k=1}^N \Lambda_{ik}(\mathcal{U}_k)\right\} \\
&\leq -c_{\mathcal{U}_i}\mathcal{L}_{V_i}^{\frac{2\alpha}{\alpha+2}} + \bar{\Pi}_{\mathcal{U}_i}
\end{aligned} \tag{52}$$

where $\bar{\Pi}_{\mathcal{U}_i} = n\alpha_{\max}\bar{\lambda}_{R_i}\Pi_{\mathcal{U}_i}$. Then the Lyapunov function $\mathcal{L}_{V_i}$ will satisfy the following inequality:

$$\hat{\mathcal{L}}_{V_i} \leq \left\{\frac{\bar{\Pi}_{\mathcal{U}_i}}{\left(1 - c_{T_i}\right)c_{\mathcal{U}_i}}\right\}^{\frac{\alpha+2}{2\alpha}} \tag{53}$$

Table 1: Parameters of the UAV system and control law

Parameter Group	**Values**
UAV Parameters	$I_{xx} = 0.00226\ \mathrm{kg} \cdot \mathrm{m}^2, \quad m = 0.5799\ \mathrm{kg}$ $I_{yy} = 0.00282\ \mathrm{kg} \cdot \mathrm{m}^2, \quad g = 9.81\ \mathrm{m/s}^2$ $I_{zz} = 0.0021\ \mathrm{kg} \cdot \mathrm{m}^2, \quad k_t = 0.01\ (\mathrm{s} \cdot \mathrm{kg})^{-1}$
Low-level Control	$h_{x_1} = -5.25, \quad h_{y_1} = -5.25, \quad h_{z_1} = 3.0$ $h_{\phi_2} = 3.50, \quad h_{\theta_2} = 3.50, \quad h_{\psi_2} = 0.35$ $h_{\phi_1} = 0.40, \quad h_{\theta_1} = 0.40, \quad h_{\psi_1} = 0.10$
Learning Parameters	$R_{11} = R_{22} = R_{12} = R_{21} = 0.1$ $\omega_i = 0.1\mathbf{1}_{12}, \quad \alpha_i = 0.9, \quad F_1 = F_2 = 0.1$ $\mu_1 = \mu_2 = 0.15, \quad \lambda = 0.9, \quad \epsilon = 1$

where c_{T_i} satisfies $0 < c_{T_i} < 1$. According to the FT stability theory from Lemma 1, the convergence time of the system states x can be derived as:

$$T_x[x(0)] = \frac{\mathcal{L}_{V_i}[x(0)](\alpha + 2)}{c_{\mathcal{U}_i} c_{T_i}(2 - \alpha)} \tag{54}$$

In summary, we have shown that: (1) The FT input $\hat{\mathcal{U}}_i$ stabilizes to $\mathcal{U}_i^*$ in finite time $T_{\hat{\mathcal{U}}_i}$ (2) The states x is stable within FT constant $T_x[x(0)]$ This completes the finite-time stability proof of the closed-loop system. □

The above theoretical analysis establishes finite-time convergence guarantees for system states and neural network weights under the proposed FT-SRL-NZS algorithm. The dynamic event-triggering mechanism in (26) prevents Zeno behavior while ensuring efficient implementation. Extensive numerical simulations are conducted in the following section to validate the effectiveness of the proposed approach.

Remark 7 (Convergence Analysis)**.** Our adaptive dynamic programming approach exhibits guaranteed finite-time convergence properties. The maximum adaptation time is bounded by $T_{\text{adapt}} = \max\{T_{\hat{W}_c}[x(0)], T_{\hat{W}_a}[x(0)]\}$, where $T_{\hat{W}_c}$ and $T_{\hat{W}_a}$ represent convergence times for critic and actor components. The convergence speed benefits from novel FT-CL laws enabling efficient training, event-triggered mechanisms reducing computational costs, and barrier transformations maintaining safety constraints. Theorems 1 and 2 establish that critic weights reach optimal values in time $T_{\hat{W}_c}[x(0)]$, actor weights converge within $T_{\hat{W}_a}[x(0)]$, and Nash equilibrium is attained in finite time. These results advance beyond classical ADP methods which only achieve asymptotic convergence.

Remark 8 (Guidelines for Practical Lyapunov Function Selection)**.** The selection of appropriate Lyapunov functions for practical applications requires systematic consideration of both system characteristics and implementation constraints. For systems dominated by linear dynamics, quadratic forms are recommended due to their analytical tractability, while energy-based functions are particularly effective for mechanical systems. The barrier function terms should be carefully scaled according to actuator limitations, with parameter μ optimized to achieve an appropriate balance between safety guarantees and control performance. Key validity conditions that must be satisfied include: Lipschitz continuous dynamics (Assumption 2), persistence of excitation (Assumption 3), and bounded neural network approximation errors (Assumption 4). The framework has inherent limitations: computational complexity exhibits polynomial growth with system dimension, communication delays may impact the effectiveness of event-triggered updates, and stability guarantees are only valid when states remain within the neural network approximation region.

6. Numerical Simulations

This section validates the FT-SRL-NZS algorithm through numerical studies on a 3D UAV tracking problem. To demonstrate the algorithm's capabilities in handling multi-player nonzero-sum games, we consider a system with two cooperative agents jointly controlling the UAV trajectory while avoiding obstacles.

6.1. Simulation setup

Consider a UAV control problem in 3D space. The state vector is $x_r = [x, y, z, \dot{x}, \dot{y}, \dot{z}, \phi, \theta, \psi, \dot{\phi}, \dot{\theta}, \dot{\psi}]^\top$ containing positions, velocities, attitudes, and angular rates. The control input is $\mathcal{U} = [\dot{x}_d, \dot{y}_d, \dot{z}_d, \dot{\psi}_d]^\top$. Under small angle assumption, the UAV dynamics is:

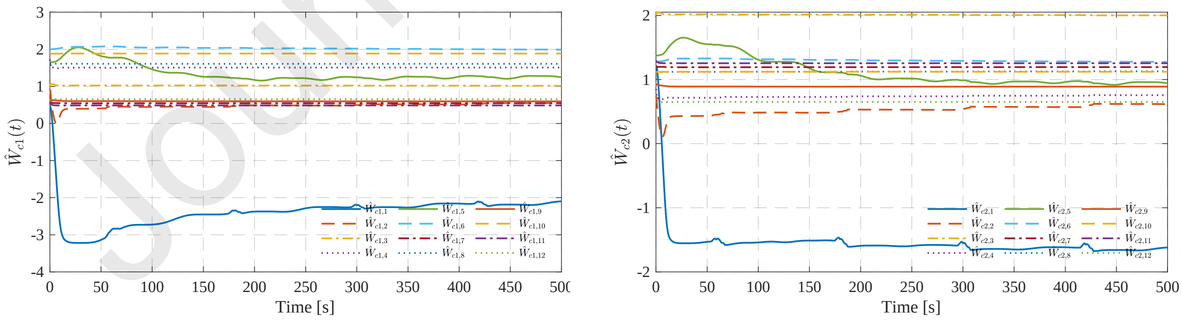
$$\begin{cases} \ddot{x} = -\frac{k_t \dot{x}}{m} - g\theta \\ \ddot{y} = -\frac{k_t \dot{y}}{m} + g\phi \\ \ddot{z} = -\frac{k_t \dot{z}}{m} + h_{z_1}\left(\dot{z}_d - \dot{z}\right) \\ \ddot{\phi} = -\frac{l h_{\phi_1}}{I_{xx}}\dot{\phi} - \frac{l h_{\phi_2}}{I_{xx}}\phi + \frac{\pi l h_{\phi_2} h_{y_1}}{4g I_{xx}}\left(\dot{y} - \dot{y}_d\right) \\ \ddot{\theta} = -\frac{l h_{\theta_1}}{I_{yy}}\dot{\theta} - \frac{l h_{\theta_2}}{I_{yy}}\theta + \frac{\pi l h_{\theta_2} h_{x_1}}{4g I_{yy}}\left(\dot{x}_d - \dot{x}\right) \\ \ddot{\psi} = \frac{l h_{\psi_1}}{I_{zz}}\left(\dot{\psi}_d - \dot{\psi}\right) \end{cases} \tag{55}$$

where m is mass, g is gravity, k_t is drag coefficient, $I_{xx,yy,zz}$ are inertias, and l is rotor distance. The actor-critic network parameters and simulation settings are as follows: Basis functions φ_i for each player $i \in \{1, 2\}$:

$$\varphi_i = \frac{1}{1+\alpha}\left[|x\dot{\theta}|^{1+\alpha}, |y\dot{\phi}|^{1+\alpha}, |z\dot{\psi}|^{1+\alpha}, |\phi\dot{\phi}|^{1+\alpha}, |\theta\dot{\theta}|^{1+\alpha}, |\psi\dot{\psi}|^{1+\alpha}\right]$$

The simulation uses MATLAB/Simulink with a step size of 0.001s and a simulation time of 500s. System parameters are given in Table 1.

Remark 9 (Applicability of FT-SRL-NZS)**.** While the numerical validation focuses on UAV control as a specific application, the proposed FT-SRL-NZS algorithm is designed for general nonlinear systems satisfying Assumptions 1-4, The algorithm can be applied to various nonlinear processes including robotic manipulation systems with uncertain dynamics, power systems with multiple generators and loads, chemical process control with nonlinear reaction kinetics, and autonomous ground vehicles with complex dynamics. The UAV example is chosen for its challenging characteristics, which is high nonlinearity, tight coupling between states, and strict safety requirements - which help demonstrate the algorithm's capabilities. The theoretical FT-SRL-NZS framework developed in this paper remains valid for any nonlinear system satisfying the stated assumptions, with application-specific adaptations primarily in the selection of basis functions and controller parameters.

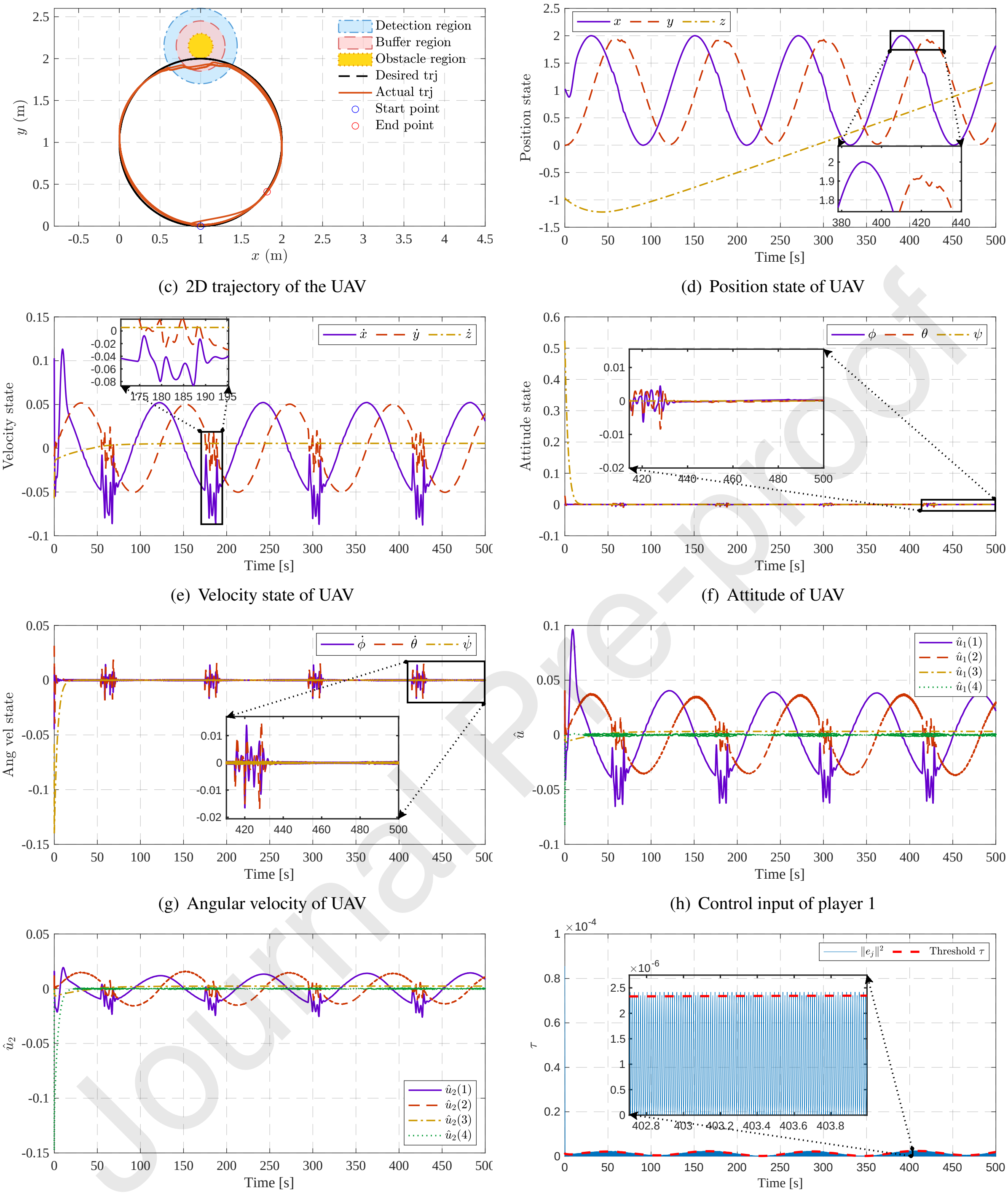


(a) Weights of critic-NN 1

(b) Weights of critic-NN 2

6.2. Simulation results

The simulation results of the UAV control system using the proposed FT-SRL-NZS algorithm are shown in Fig. 3. The critic-NN weights stabilize to steady state with guaranteed time, as demonstrated in Fig. 3(a) and Fig. 3(b). The UAV achieves obstacle-free trajectory tracking in both 2D plane (Fig. 3(c)) and 3D space (Fig. 3(m)).



(c) 2D trajectory of the UAV

(d) Position state of UAV

(e) Velocity state of UAV

(f) Attitude of UAV

(g) Angular velocity of UAV

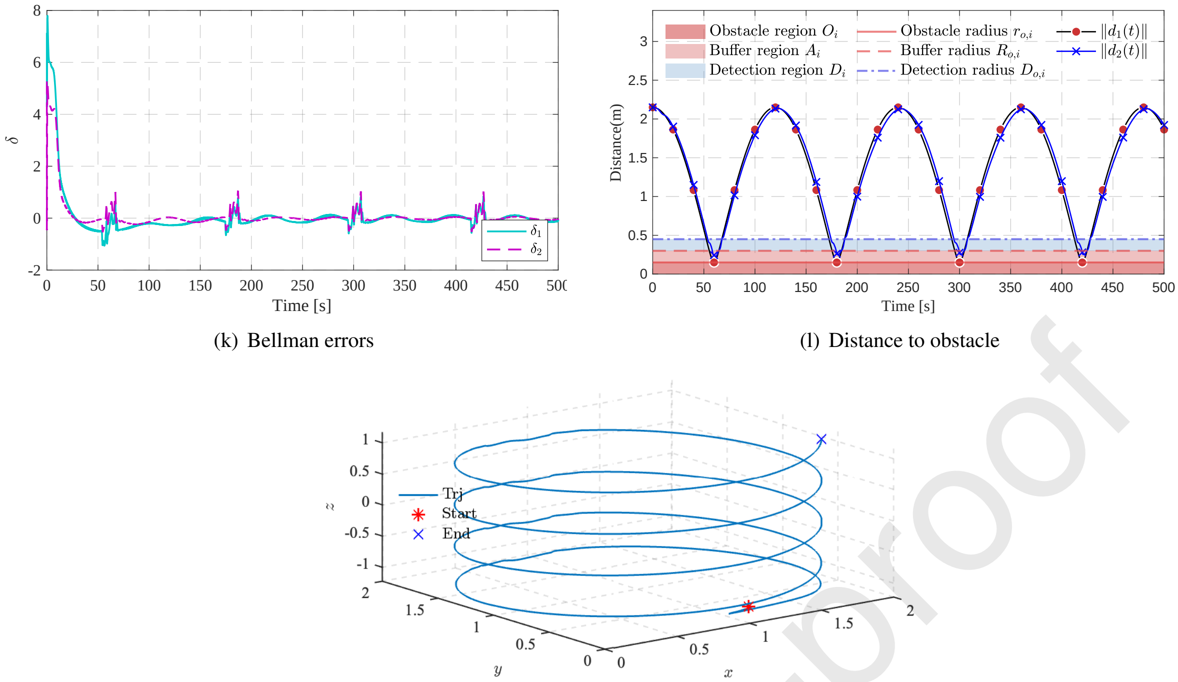
(h) Control input of player 1

(i) Control input of player 2

(j) DET threshold

The time evolution of key system states including position, velocity, attitude, and angular velocity are presented in Fig. 3(d)-3(g). The control inputs from both players are shown in Fig. 3(h)-3(i), with the dynamic event-triggering threshold illustrated in Fig. 3(j). The Hamiltonian errors indicating learning performance are plotted in Fig. 3(k).

The distance to the obstacle remains above the safety threshold throughout the trajectory as shown in Fig. 3(l),



(k) Bellman errors

(l) Distance to obstacle

(m) 3D trajectory of UAV

Figure 3: Simulation results of the UAV control: (a) W_{c1}, (b) W_{c2}, (c) 2D trajectory, (d) position, (e) velocity, (f) attitude, (g) angular velocity, (h) control input 1, (i) control input 2, (j) triggering rule, (k) Bellman errors, (l) distance to the obstacle, and (m) 3-dimensional trajectory of the UAV.

verifying the effectiveness of the barrier function-based safety constraints. The results demonstrate that the proposed algorithm achieves: 1) finite-time convergence of neural network weights, 2) obstacle avoidance while maintaining tracking performance, and 3) efficient event-triggered control implementation without Zeno behavior. The simulation validates both the theoretical guarantees and practical efficacy of the FT-SRL-NZS approach.

7. Hardware Experiments

To further verify the effectiveness of the proposed FT-SRL-NZS algorithm, hardware experiments are conducted on an unmanned aerial vehicle (UAV) position tracking control case, which uses an X150 quadcopter equipped with an RK3566 processor (1.8 GHz, 4GB RAM). An 8-camera OptiTrack system provides real-time position tracking. The two-player nonzero-sum game controls the UAV to track desired 3D trajectories. The FT-SRL-NZS algorithm runs on a workstation (Intel i7-12700, 3.6 GHz, 32GB RAM) at a 30 Hz control frequency. Velocity commands are transmitted to the UAV via 5GHz WiFi. The experimental setup is shown in Fig. 4.

Note that due to the wind, the aerodynamic forces, and the sensor noise, there exist unknown disturbances in the real-world UAV tracking control. To illustrate the effectiveness of the proposed FT-SRL-NZS algorithm, two comparison algorithms are considered in the hardware experiments:

1. **FT-SRL-NZS algorithm**: Proposed FT safe RL control algorithm.
2. **StaF-SRL-NZS algorithm** from [22]: State-following kernel-based safe RL control algorithm.
3. **SRL-NZS algorithm** from [31]: Standard safe RL control algorithm.

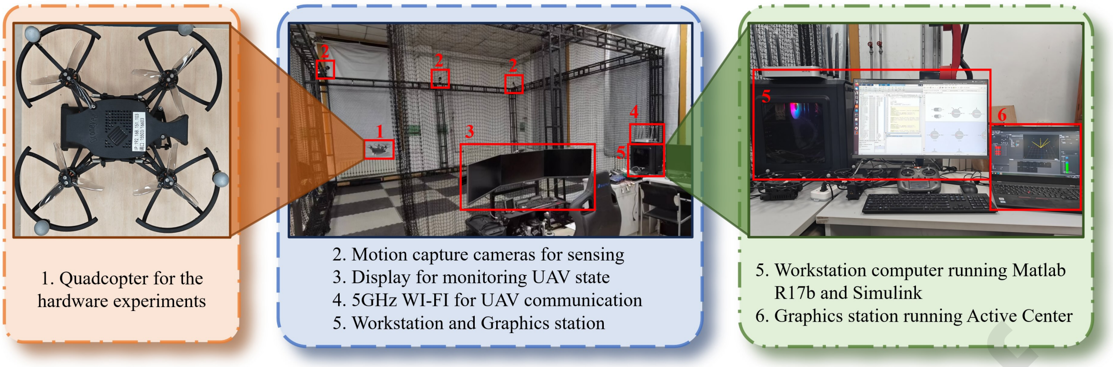

1
2
2
2
3
4
5
5
6
1. Quadcopter for the hardware experiments
2. Motion capture cameras for sensing
3. Display for monitoring UAV state
4. 5GHz WI-FI for UAV communication
5. Workstation and Graphics station
5. Workstation computer running Matlab R17b and Simulink
6. Graphics station running Active Center

Figure 4: Quadcopter and motion capture system for hardware experiment.

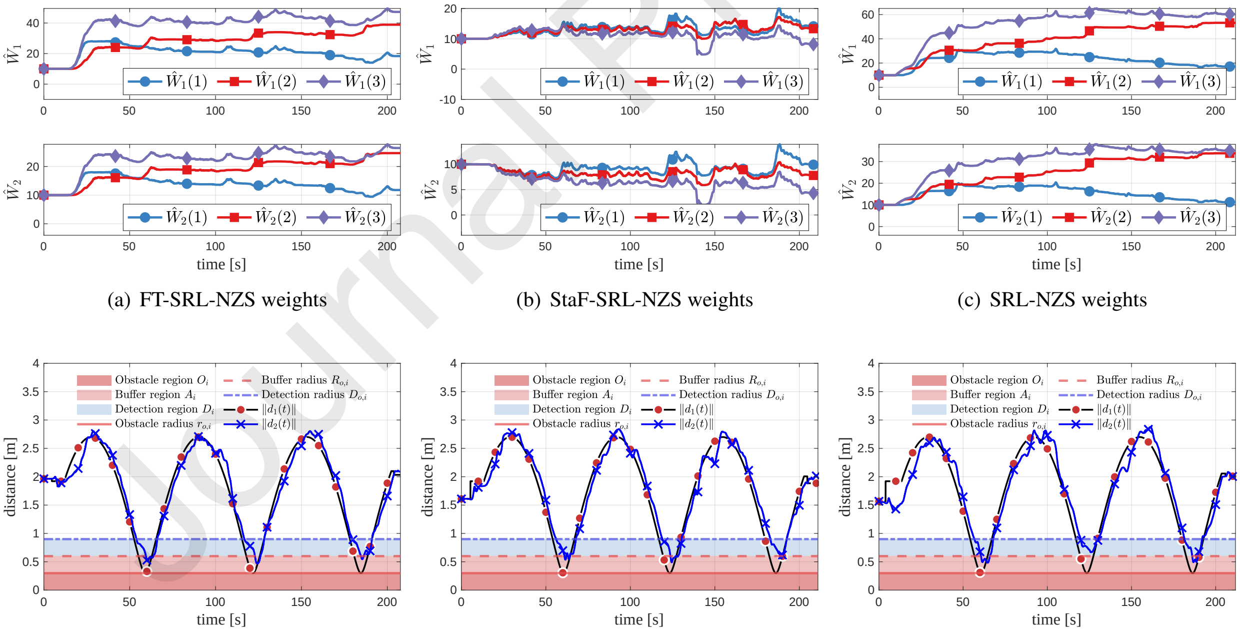
where the detailed design of the NN basis functions is given by:

$$\text{FT-SRL-NZS: } \varphi_i = \frac{1}{1+\alpha} \times \left[|e_x|^{1+\alpha}, |e_y|^{1+\alpha}, |e_x|^{1+\alpha} + |e_y|^{1+\alpha} \right], \forall i \in \{1,2\}$$

$$\text{StaF-SRL-NZS: } \varphi_i = e^\top (e + 0.7 \frac{e^\top e + 0.01}{e^\top e + 1}) \times \left[[0,1]^\top, [0.87, -0.5]^\top, [-0.87, -0.5]^\top \right], \forall i \in \{1,2\}$$

$$\text{SRL-NZS: } \varphi_i = \frac{1}{2} \times \left[e_x^2, e_y^2, e_x^2 + e_y^2 \right], \forall i \in \{1,2\}$$

The initial weights of the critic-NNs are selected as $W_{c1}(0) = W_{c2}(0) = [10, 10, 10]^\top$, and the $R_{11} = R_{12} = 100$, $R_{22} = R_{21} = 50$, $\omega_1 = 100\mathbf{1}_2$, $\omega_2 = 200\mathbf{1}_2$, $\mu_1 = \mu_2 = 0.5$, $\alpha_1 = \alpha_2 = 0.01$, and the fractional-order for the FT-SRL-NZS algorithm is $\alpha = 0.9$.

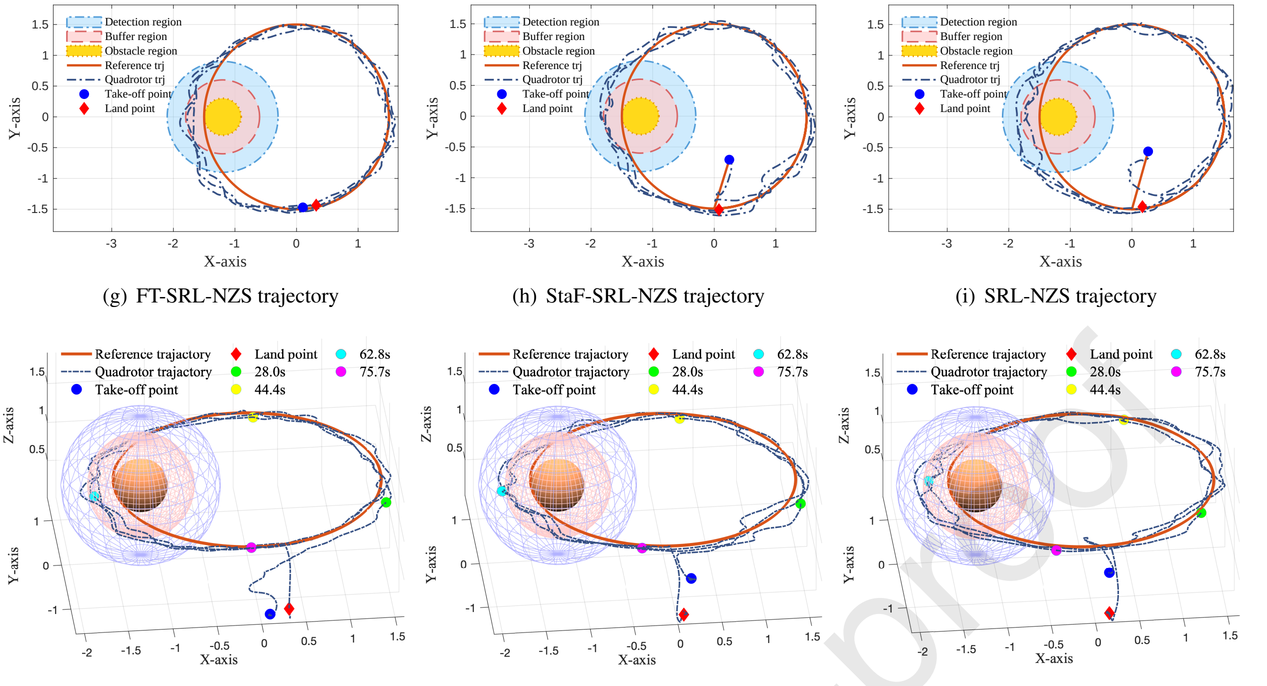


(a) FT-SRL-NZS weights
(b) StaF-SRL-NZS weights
(c) SRL-NZS weights

(d) FT-SRL-NZS distance
(e) StaF-SRL-NZS distance
(f) SRL-NZS distance

7.1. Experiment Results

The experimental results are shown in Fig. 5, where the detailed results of the FT-SRL-NZS algorithm and the comparison algorithms are presented. The weights of the critic-NNs are shown in Fig. 5(a)-5(c), which demonstrate



(g) FT-SRL-NZS trajectory

(h) StaF-SRL-NZS trajectory

(i) SRL-NZS trajectory

(j) FT-SRL-NZS 3d trajectory

(k) StaF-SRL-NZS 3d trajectory

(l) SRL-NZS 3d trajectory

Figure 5: Simulation results of comparison experiments: (a)-(c) $\hat{W}_{c1}\&\hat{W}_{c2}$, (d)-(f) distance to the obstacle, (g)-(i) 2D trajectory, (j)-(l) 3D trajectory.

that the critic-NNs stabilize within finite time under the control of the FT-SRL-NZS algorithm. The distance to the obstacle is shown in Fig. 5(d)-5(f), where the UAV maintains a safe distance from the obstacle under the control of the FT-SRL-NZS algorithm. The 2D trajectory of the UAV is shown in Fig. 5(g)-5(i), where the UAV achieves the FT optimal control and avoids the collision with the obstacle. The 3D trajectory of the UAV is shown in Fig. 5(j)-5(l), where the UAV achieves the FT tracking control while avoiding the collision with the obstacle. The experimental results show that the proposed FT-SRL-NZS algorithm can achieve the FT optimal control of the UAV tracking control. The critic-NNs can approximate the optimal value function within finite time.

7.2. Further Experiment on UAV Control

7.2.1. Experiment Setup

To further investigate both tracking performance and the safety of the proposed FT-SRL-NZS algorithm in the UAV control, a more complex UAV control case is considered in this section. The desired trajectory is designed as a four-leaf clover trajectory, which could be formulated as

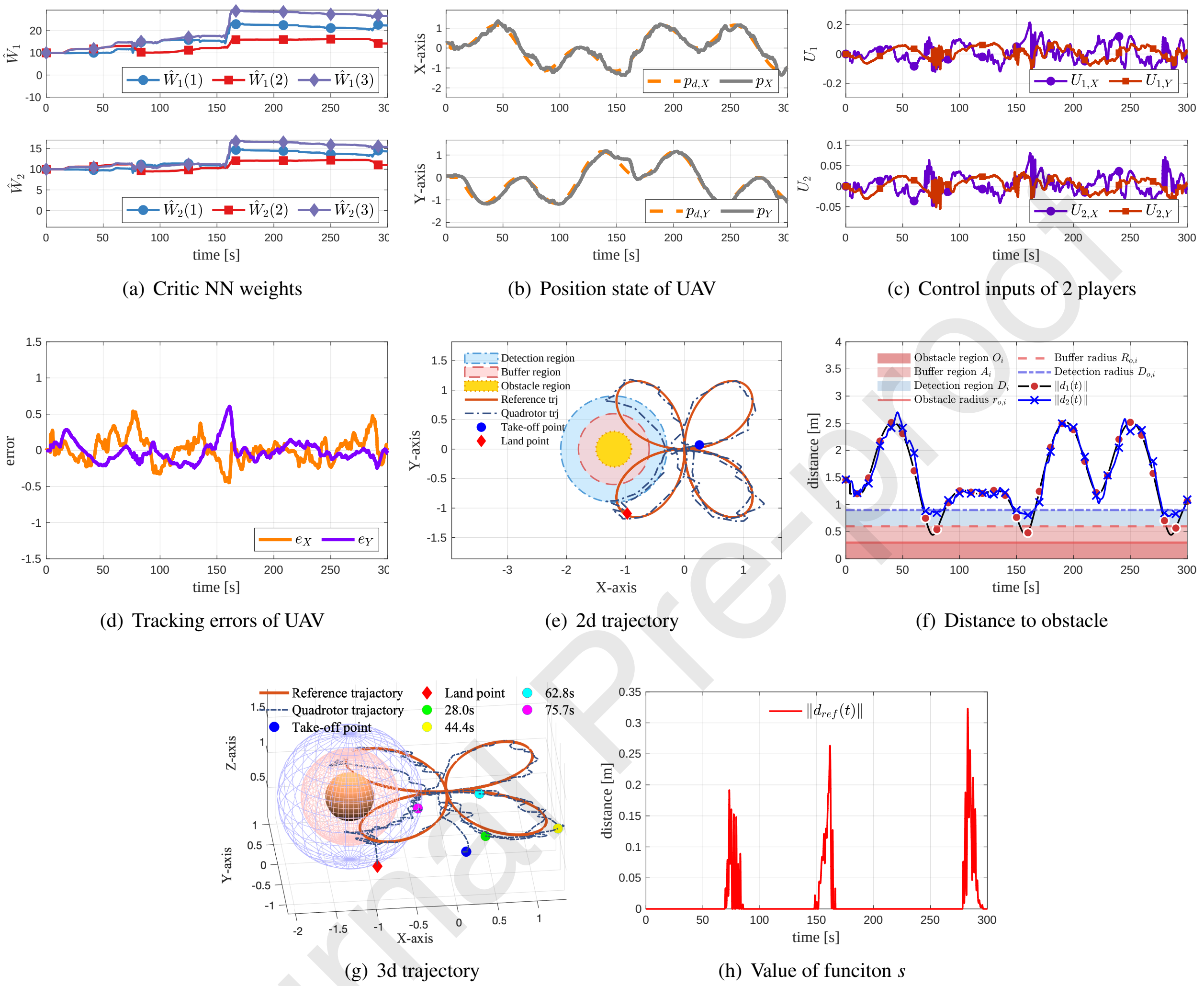
$$\begin{cases} x_d(t) = 2\cos(2\omega t + \frac{\pi}{4}) \times \cos(\omega t + \frac{\pi}{4}) \\ y_d(t) = 2\cos(2\omega t + \frac{\pi}{4}) \times \sin(\omega t + \frac{\pi}{4}) \end{cases} \tag{56}$$

where ω is set as 0.05, the detailed parameters of the FT-SRL-NZS algorithm are the same as the previous UAV control case.

7.2.2. Experiment Results

Figure 6 presents the experimental results with the four-leaf clover trajectory. The critic-NN weights (Fig. 6(a)) converge rapidly to steady-state values, demonstrating finite-time stabilization. The UAV successfully tracks the complex trajectory with satisfactory errors (Fig. 6(d)) while maintaining safe obstacle avoidance (Fig. 6(f)). The control inputs from both players (Fig. 6(c)) remain bounded and smooth throughout the experiment. The 2D and

3D trajectory plots (Fig. 6(b), 6(g)) show precise tracking of the four-leaf clover pattern while avoiding collisions. The barrier function value (Fig. 6(h)) stays within safe bounds, validating the effectiveness of the safety constraints. These results demonstrate that the proposed FT-SRL-NZS algorithm achieves both tracking performance and safety objectives in a complex trajectory following a task.



(a) Critic NN weights

(b) Position state of UAV

(c) Control inputs of 2 players

(d) Tracking errors of UAV

(e) 2d trajectory

(f) Distance to obstacle

(g) 3d trajectory

(h) Value of funciton s

Figure 6: Experimental results of the UAV tracking four-leaf clover trajectory.

8. Conclusion

This paper develops a novel finite-time safe reinforcement learning framework for multi-player nonzero-sum games (FT-SRL-NZS). The key contributions include: (1) Formulating a finite-time safe optimal control problem that achieves Nash equilibrium while satisfying safety constraints, (2) Designing actor-critic neural networks with FT-CL laws to otain the FT value and its corresponded controller, (3) Implementing dynamic event-triggering mechanism that reduces computation and communication overhead while preserving stability guarantees. Theoretical analysis establishes FT convergent NN weights and states through Lyapunov stability theory. Numerical simulations demonstrate the algorithm's capability to achieve rapid learning and safe control for nonlinear systems. Hardware experiments on UAV trajectory tracking validate the practical effectiveness compared to existing methods. Future work will explore extensions to distributed multi-agent systems and applications with more complex safety-critical constraints. The proposed framework provides a promising approach for achieving both rapid learning and rigorous safety assurance in multi-agent optimal control.

CRediT authorship contribution statement

Junkai Tan: Writing - original draft, Investigation, Conceptualization. **Shuangsi Xue:** Methodology, Investigation, Conceptualization. **Qingshu Guan:** Investigation, Conceptualization. **Kai Qu:** Visualization, Methodology. **Hui Cao:** Methodology, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Acknowledgements

This research is supported by the China Postdoctoral Science Foundation (Project ID: 2024M762602).

References

[1] J. Qiao, M. Li, D. Wang, Asymmetric Constrained Optimal Tracking Control With Critic Learning of Nonlinear Multiplayer Zero-Sum Games, IEEE Transactions on Neural Networks and Learning Systems 35 (4) (2024) 5671–5683. doi:10.1109/TNNLS.2022.3208611.

[2] D. Lin, S. Xue, D. Liu, M. Liang, Y. Wang, Adaptive dynamic programming-based hierarchical decision-making of non-affine systems, Neural Networks 167 (2023) 331–341. doi:10.1016/j.neunet.2023.07.044.

[3] B. Zhu, H. Liang, B. Niu, H. Wang, N. Zhao, X. Zhao, Observer-based reinforcement learning for optimal fault-tolerant consensus control of nonlinear multi-agent systems via a dynamic event-triggered mechanism, Information Sciences 689 (2025) 121350. doi:10.1016/j.ins.2024.121350.

[4] M. Zhao, D. Wang, M. Ha, J. Qiao, Evolving and Incremental Value Iteration Schemes for Nonlinear Discrete-Time Zero-Sum Games, IEEE Transactions on Cybernetics (2022) 1–13doi:10.1109/TCYB.2022.3198078.

[5] K. Zhang, Z.-X. Zhang, X. P. Xie, J. d. J. Rubio, An Unknown Multiplayer Nonzero-Sum Game: Prescribed-Time Dynamic Event-Triggered Control via Adaptive Dynamic Programming, IEEE Transactions on Automation Science and Engineering (2024) 1–12doi:10.1109/TASE.2024.3484412.

[6] K. G. Vamvoudakis, F. L. Lewis, G. R. Hudas, Multi-agent differential graphical games: Online adaptive learning solution for synchronization with optimality, Automatica 48 (8) (2012) 1598–1611. doi:10.1016/j.automatica.2012.05.074.

[7] V. Narayanan, A. Sahoo, S. Jagannathan, K. George, Approximate Optimal Distributed Control of Nonlinear Interconnected Systems Using Event-Triggered Nonzero-Sum Games, IEEE Transactions on Neural Networks and Learning Systems 30 (5) (2019) 1512–1522. doi:10.1109/TNNLS.2018.2869896.

[8] K. Wang, C. Mu, Learning-Based Control With Decentralized Dynamic Event-Triggering for Vehicle Systems, IEEE Transactions on Industrial Informatics 19 (3) (2023) 2629–2639. doi:10.1109/TII.2022.3168034.

[9] M. Li, J. Qin, J. Li, Q. Liu, Y. Shi, Y. Kang, Game-Based Approximate Optimal Motion Planning for Safe Human-Swarm Interaction, IEEE Transactions on Cybernetics (2023) 1–12doi:10.1109/TCYB.2023.3340659.

[10] L. Xue, J. Ye, Y. Wu, J. Liu, D. C. Wunsch, Prescribed-Time Nash Equilibrium Seeking for Pursuit-Evasion Game, IEEE/CAA Journal of Automatica Sinica 11 (6) (2024) 1518–1520. doi:10.1109/JAS.2023.124077.

[11] K. Tong, M. Li, J. Qin, Q. Ma, J. Zhang, Q. Liu, Differential Game-Based Control for Nonlinear Human–Robot Interaction System With Unknown Desired Trajectory, IEEE Transactions on Cybernetics (2024) 1–11doi:10.1109/TCYB.2024.3402353.

[12] Y. Liu, H. Li, R. Lu, Z. Zuo, X. Li, An Overview of Finite/Fixed-Time Control and Its Application in Engineering Systems, IEEE/CAA Journal of Automatica Sinica 9 (12) (2022) 2106–2120. doi:10.1109/JAS.2022.105413.

[13] W. M. Haddad, A. L'Afflitto, Finite-Time Stabilization and Optimal Feedback Control, IEEE Transactions on Automatic Control 61 (4) (2016) 1069–1074. doi:10.1109/TAC.2015.2454891.

[14] A. Sharafian, A. Ali, I. Ullah, T. R. Khalifa, X. Bai, L. Qiu, Fuzzy adaptive control for consensus tracking in multiagent systems with incommensurate fractional-order dynamics: Application to power systems, Information Sciences 689 (2025) 121455. doi:10.1016/j.ins.2024.121455.

[15] L. Zhang, Y. Chen, Distributed Finite-Time ADP-Based Optimal Control for Nonlinear Multiagent Systems, IEEE Transactions on Circuits and Systems II: Express Briefs 70 (12) (2023) 4534–4538. doi:10.1109/TCSII.2023.3291399.

[16] L. Zhang, Y. Chen, Finite-Time Adaptive Dynamic Programming for Affine-Form Nonlinear Systems, IEEE Transactions on Neural Networks and Learning Systems (2023) 1–14doi:10.1109/TNNLS.2023.3337387.

[17] X. Tong, D. Ma, Z. Wang, Z. Ming, X. Xie, Model-free adaptive dynamic event-triggered robust control for unknown nonlinear systems using iterative neural dynamic programming, Information Sciences 655 (2024) 119866. doi:10.1016/j.ins.2023.119866.

[18] Q. Wei, D. Liu, F. L. Lewis, Optimal distributed synchronization control for continuous-time heterogeneous multi-agent differential graphical games, Information Sciences 317 (2015) 96–113. doi:10.1016/j.ins.2015.04.044.

[19] J. Tan, S. Xue, H. Li, Z. Guo, H. Cao, D. Li, Prescribed performance robust approximate optimal tracking control via stackelberg game, IEEE Transactions on Automation Science and Engineering (2025) 1–1doi:10.1109/TASE.2025.3549114.

[20] I. A. Zamfirache, R.-E. Precup, E. M. Petriu, Adaptive reinforcement learning-based control using proximal policy optimization and slime mould algorithm with experimental tower crane system validation, Applied Soft Computing 160 (2024) 111687. doi:10.1016/j.asoc.2024.111687.

[21] I. A. Zamfirache, R.-E. Precup, E. M. Petriu, Q-learning, policy iteration and actor-critic reinforcement learning combined with metaheuristic algorithms in servo system control, Facta Universitatis, Series: Mechanical Engineering 21 (4) (2023) 615. doi:10.22190/FUME231011044Z.

[22] C. Peng, H. Zhang, Y. He, J. Ma, State-Following-Kernel-Based Online Reinforcement Learning Guidance Law Against Maneuvering Target, IEEE Transactions on Aerospace and Electronic Systems 58 (6) (2022) 5784–5797. doi:10.1109/TAES.2022.3178770.

[23] D. Wang, J. Qiao, Approximate neural optimal control with reinforcement learning for a torsional pendulum device, Neural Networks 117 (2019) 1–7. doi:10.1016/j.neunet.2019.04.026.

[24] Q. Shi, H. Zhang, W. Pedrycz, Robust Learning-Based Gain-Scheduled Path Following Controller Design for Autonomous Ground Vehicles, IEEE Transactions on Emerging Topics in Computational Intelligence 8 (2) (2024) 1427–1436. doi:10.1109/TETCI.2023.3349183.

[25] K. Wang, C. Mu, Z. Ni, D. Liu, Safe Reinforcement Learning and Adaptive Optimal Control With Applications to Obstacle Avoidance Problem, IEEE Transactions on Automation Science and Engineering (2023) 1–14doi:10.1109/TASE.2023.3299275.

[26] C. Peng, X. Liu, J. Ma, Design of Safe Optimal Guidance With Obstacle Avoidance Using Control Barrier Function-Based Actor–Critic Reinforcement Learning, IEEE Transactions on Systems, Man, and Cybernetics: Systems (2023) 1–13doi:10.1109/TSMC.2023.3288826.

[27] J. Tan, J. Wang, S. Xue, H. Cao, H. Li, Z. Guo, Human-machine shared stabilization control based on safe adaptive dynamic programming with bounded rationality, International Journal of Robust and Nonlinear Control (2025) rnc.7931doi:10.1002/rnc.7931.

[28] P. Deptula, H.-Y. Chen, R. A. Licitra, J. A. Rosenfeld, W. E. Dixon, Approximate Optimal Motion Planning to Avoid Unknown Moving Avoidance Regions, IEEE Transactions on Robotics 36 (2) (2020) 414–430. doi:10.1109/TRO.2019.2955321.

[29] N.-M. T. Kokolakis, K. G. Vamvoudakis, Safety-Aware Pursuit-Evasion Games in Unknown Environments Using Gaussian Processes and Finite-Time Convergent Reinforcement Learning, IEEE Transactions on Neural Networks and Learning Systems (2022) 1–14doi:10.1109/TNNLS.2022.3203977.

[30] Z. Marvi, B. Kiumarsi, Safe reinforcement learning: A control barrier function optimization approach, International Journal of Robust and Nonlinear Control 31 (6) (2021) 1923–1940. doi:10.1002/rnc.5132.

[31] C. Mu, K. Wang, X. Xu, C. Sun, Safe Adaptive Dynamic Programming for Multiplayer Systems With Static and Moving No-entry Regions, IEEE Transactions on Artificial Intelligence (2023) 1–13doi:10.1109/TAI.2023.3325780.

[32] H. Shen, Z. Li, J. Wang, J. Cao, Nonzero-Sum Games Using Actor-Critic Neural Networks: A Dynamic Event-Triggered Adaptive Dynamic Programming, Information Sciences (2024) 120236doi:10.1016/j.ins.2024.120236.

[33] L. Cui, B. Pang, Z.-P. Jiang, Learning-Based Adaptive Optimal Control of Linear Time-Delay Systems: A Policy Iteration Approach, IEEE Transactions on Automatic Control 69 (1) (2024) 629–636. doi:10.1109/TAC.2023.3273786.

[34] R. Kamalapurkar, J. R. Klotz, P. Walters, W. E. Dixon, Model-Based Reinforcement Learning in Differential Graphical Games, IEEE Transactions on Control of Network Systems 5 (1) (2018) 423–433. doi:10.1109/TCNS.2016.2617622.

[35] J. Tan, S. Xue, Z. Guo, H. Li, H. Cao, B. Chen, Data-driven optimal shared control of unmanned aerial vehicles, Neurocomputing 622 (2025) 129428. doi:10.1016/j.neucom.2025.129428.

[36] B. Dong, X. Zhu, T. An, H. Jiang, B. Ma, Barrier-critic-disturbance approximate optimal control of nonzero-sum differential games for modular robot manipulators, Neural Networks 181 (2025) 106880. doi:10.1016/j.neunet.2024.106880.

[37] P. Wang, C. Yu, M. Lv, J. Cao, Adaptive Fixed-Time Optimal Formation Control for Uncertain Nonlinear Multiagent Systems Using Reinforcement Learning, IEEE Transactions on Network Science and Engineering 11 (2) (2024) 1729–1743. doi:10.1109/TNSE.2023.3330266.

[38] J. Tan, S. Xue, Q. Guan, T. Niu, H. Cao, B. Chen, Unmanned aerial-ground vehicle finite-time docking control via pursuit-evasion games, Nonlinear Dynamics (Mar. 2025). doi:10.1007/s11071-025-11021-6.

[39] Q. Ma, P. Jin, F. L. Lewis, Guaranteed Cost Attitude Tracking Control for Uncertain Quadrotor Unmanned Aerial Vehicle Under Safety Constraints, IEEE/CAA Journal of Automatica Sinica 11 (6) (2024) 1447–1457. doi:10.1109/JAS.2024.124317.

[40] J. Tan, S. Xue, H. Cao, S. S. Ge, Human–AI interactive optimized shared control, Journal of Automation and Intelligence (2025) S2949855425000024doi:10.1016/j.jai.2025.01.001.

[41] T. Haidegger, L. Kovács, R.-E. Precup, B. Benyó, Z. Benyó, S. Preitl, Simulation and control for telerobots in space medicine, Acta Astronautica 81 (1) (2012) 390–402. doi:10.1016/j.actaastro.2012.06.010.

[42] Y. Feng, M. Wu, L. Chen, X. Chen, W. Cao, S. Du, W. Pedrycz, Hybrid Intelligent Control Based on Condition Identification for Combustion Process in Heating Furnace of Compact Strip Production, IEEE Transactions on Industrial Electronics 69 (3) (2022) 2790–2800. doi:10.1109/TIE.2021.3066918.

[43] I. A. Zamfirache, R.-E. Precup, E. M. Petriu, Safe reinforcement learning-based control using deep deterministic policy gradient algorithm and slime mould algorithm with experimental tower crane system validation, Information Sciences 692 (2025) 121640. doi:10.1016/j.ins.2024.121640.

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐ The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: