Unmanned Systems, Vol. 0, No. 0 (2013) 1–19

Cooperative Game-based Optimal Shared Control of Unmanned Aerial Vehicle

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

[a] *State Key Laboratory of Electrical Insulation and Power Equipment, Xi'an Jiaotong University, Xi'an 710049, P. R. China*

[b] *School of Electrical Engineering, Xi'an Jiaotong University, Xi'an 710049, P. R. China*

Collaboration between human operators and unmanned aerial vehicles (UAV) is an emerging research area in UAV control, focused on enhancing UAV performance and alleviating the workload of human operators. This paper proposes a dynamic event-triggered cooperative control method for the optimal shared control of UAV. The proposed method formulates a cooperative non-zero sum game between human operators and UAV, and the optimal shared control of UAV is achieved by solving the Nash equilibrium of the game. The optimal shared control of UAV is approximated by the actor-critic algorithm, which learns the optimal control policy from the historical operation data of UAV. Then a continuous and efficient shared mechanism is established to allocate the relationship between optimal and human control input. To reduce the online computation burden of the optimal shared control in the UAV system, a dynamic event-triggered mechanism is proposed to update the shared control input only when the event-triggered condition is satisfied. The performance of the proposed method is evaluated by numerical simulations. The results show that the proposed method could achieve optimal shared control of UAV with human operators efficiently and continuously compared with conventional shared control methods.

Keywords: Unmanned aerial vehicles; cooperative game; optimal control; shared control; reinforcement learning.

1. Introduction

Unmanned aerial vehicles (UAV) are extensively utilized in complex and emergent scenarios, including transportation [1,2], surveillance [3,4], search and rescue [5,6], owing to their flexibility, cost-effectiveness, and operational efficiency [7, 8]. The performance of UAV is constrained by the workload of supervising human operators [9, 10], presenting a significant challenge in UAV control. The conventional control methods for UAV primarily rely on their autonomy, wherein the human operator monitors the UAV's state and provides control inputs [11,12]. The UAV's autonomy is designed to accomplish the intended trajectory and task execution. Nevertheless, the autonomy of UAV may struggle in complex and emergent scenarios, necessitating human operator intervention to maintain safety and efficiency [13,14]. The performance of UAV may decline when human operators are required to intervene frequently, resulting in an increased workload for these operators [15]. Improving UAV performance and reducing human operator workload necessitates the development of a cooperative control method for optimal shared control of UAV.

The balance between UAV autonomy and human operator intervention presents a significant challenge in UAV control [16, 17]. Human operators supervise UAV operations and provide control inputs, while UAV autonomy is designed to track desired trajectories and operate efficiently [18, 19]. Recent studies have focused on the shared control of UAV with human operators to enhance UAV performance and alleviate the workload of human operators [20, 21]. The shared control of UAV is characterized by the allocation of the relationship between the control inputs of autonomy and human operators, wherein these inputs are incorporated to facilitate UAV cooperative control tasks. Two primary methods exist for the shared control of UAV with human operators: the direct shared control method [22,23], which involves the direct combination of control inputs from both autonomy and human operators; and the arbitration shared control method, which integrates control inputs from autonomous systems and human operators through an arbitration mechanism, such as fuzzy logic-based arbitration [24,25], and learning-based arbitration [26,27], in which the human input will be transmitted to the autonomous system for processing, resulting in a comprehensive control input derived from the autonomy. However, the direct shared control method may lead to in-

*E-mail: huicao@mail.xjtu.edu.cn

stability in the UAV system, as it fails to account for the interplay and relationship between the autonomy control input and the inputs of human operators [28,29]. Also, the arbitration shared control method may lead to inefficiencies in the UAV system, attributed to an inadequate and non-continuous arbitration mechanism. Achieving optimal shared control of UAV with human operators requires the design of a cooperative control method that accounts for the interplay between the autonomy's control input and that of human operators.

Optimal shared control of UAV in online settings presents major difficulties, where calculation and implementation of optimal shared control between UAV and human operators demand massive computational resources and communication bandwidth, which may lead to delays in control input and an overall decrease in UAV performance [30, 31]. Conventional optimal control approaches for UAV primarily rely on offline optimal control methods, wherein the optimal control is computed offline and then executed online [32,33]. However, offline optimal control of UAV might encounter difficulties with complex and emergent scenarios, in which frequent human operator intervention is necessitated [34,35], Consequently, the initially calculated optimal control inputs may not remain optimal and require online updates. To obtain optimal shared control input of UAV with human operators, online reinforcement learning (RL) and adaptive dynamic programming (ADP) methods have been investigated to obtain the optimal controller for UAV [36–38]. Event-triggered mechanisms have been widely studied to lessen the online computational demands of optimal shared control in UAV systems, updating the control input merely when the event-triggered condition is met [39–42]. Dynamic event-triggering rule has been investigated to enhance triggering performance and decrease triggering frequency by adjusting the triggering threshold adaptively based on system states [43, 44]. Therefore, it is essential to develop a computationally and communicatively efficient method for calculating the optimal shared controller to control UAVs optimally and efficiently.

Motivated by the aforementioned challenges, this paper proposes a cooperative game-based optimal shared control method for UAV. The method formulates a cooperative non-zero sum game between human operators and UAV, where the optimal shared control is achieved by solving the Nash equilibrium of the game. The optimal control policy is approximated through an actor-critic algorithm that learns from historical UAV operation data. A novel continuous and efficient shared mechanism is established to allocate control authority between optimal and human control inputs. To reduce the computational burden of online controller approximation, a dynamic event-triggered mechanism is proposed that updates the shared control input only when specific triggering conditions are met. The performance of the proposed method is evaluated through extensive numerical simulations. Results demonstrate that the proposed method achieves efficient and continuous optimal shared control of UAV with human operators compared to existing shared control approaches. The main contributions of this paper are summarized as follows:

(1) A cooperative game-based control method is proposed for optimal shared control of UAV. The method formulates a cooperative non-zero sum game between human operators and UAV, achieving optimal shared control by solving the Nash equilibrium of the game. A novel shared mechanism is established to allocate the relationship between optimal and human control inputs continuously and efficiently, improving upon traditional shared control methods [20,23,45,46].
(2) The optimal shared control of UAV is approximated using RL methods, learning optimal control policies from historical UAV operation data. The Nash equilibrium of the cooperative game is achieved through an actor-critic algorithm, demonstrating up to 79.23% improvement in UAV performance compared to traditional optimal control methods [22,25,47–49].
(3) A dynamic event-triggered mechanism is developed to reduce the online computational burden of optimal shared control in UAV systems. This mechanism updates the shared control input only when triggering conditions are satisfied, while avoiding Zeno behavior in the shared control input. The proposed method demonstrates a 75.33% improvement in computational efficiency compared to existing adaptive UAV control methods [37,41,50].

The rest of this paper is organized as follows. In Section I, preliminaries and the system description of UAV are introduced. In Section II, the problem of optimal shared control of UAV is formulated. Section III provides the main result of cooperative game-based optimal shared control of UAV. Section IV presents the stability analysis of the proposed method. In Section V, we provide numerical simulations to verify the effectiveness of proposed method. Finally, Section VI concludes the paper.

2. Preliminaries: System Description of UAV

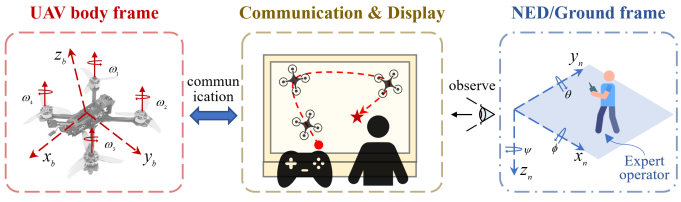


Fig. 1. The configuration of the UAV.

The human-UAV cooperation system is illustrated in Fig. 1, which consists of an expert operator and a UAV. The frame of the UAV is defined as the body frame with the origin at the center of mass of the UAV. The body frame is defined as x_b pointing forward, y_b pointing to the right, and z_b pointing up. The expert operator at the ground is responsible for controlling the UAV to achieve the desired trajectory, in which the expert operator observes the state

of the UAV and sends the control input to the UAV in the north-east-down (NED) frame. It should be noted that the human operator is responsible for supervising the operation of the UAV and sending the control input to the UAV, and the frame of the human operator is defined as the gournd frame with the origin at the position of the human operator, which is the same as the NED frame. To facilitate the cooperative control design of the UAV with the human operator, we consider constructing the dynamics model of the UAV in the earth-fixed NED frame. Define the state vector of the UAV as $\Omega = [x, y, z, \dot{x}, \dot{y}, \dot{z}, \phi, \theta, \psi, \dot{\phi}, \dot{\theta}, \dot{\psi}]^\top$, where x, y, and z are the position of the UAV in the earth-fixed NED frame; $\dot{x}, \dot{y}$, and $\dot{z}$ are the corresponding velocities of the UAV; ϕ, θ, ψ, $\dot{\phi}$, $\dot{\theta}$, and $\dot{\psi}$ denote the angular position and angular velocity of roll, pitch, and yaw, respectively. Consider the final control input implemented to the UAV consists of both the control input of the autonomy and the human operator, the control input of the autonomy is defined as $\mathcal{U} = [\dot{x}_d, \dot{y}_d, \dot{z}_d, \dot{\psi}_d]^\top$, in which $\dot{x}_d$, $\dot{y}_d$, $\dot{z}_d$. To facilitate the modeling of the UAV dynamics, we assume that the angles ϕ, θ, and ψ are small enough to apply the small angle approximation. Based on the UAV model presented in literature [15] , the dynamics model of the UAV in the NED frame is described by

$$\begin{aligned} &\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = -\frac{k_t}{m}\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} + \frac{\mathcal{R}_b^{ned}}{m}\begin{bmatrix} 0 \\ 0 \\ -F \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \\ &\text{s.t.} \quad \mathcal{R}_b^{ned} = \begin{bmatrix} 1 & \phi\theta - \psi & \theta + \phi\psi \\ \psi & \phi\theta\psi + 1 & \theta\psi - \phi \\ -\theta & \phi & 1 \end{bmatrix} \end{aligned} \tag{1}$$

where m denotes the mass of the UAV, g represents the gravitational acceleration, k_t denotes the coefficient of aerodynamic drag, F represents the thrust force generated by the UAV, and $\mathcal{R}_b^{ned}$ denotes the rotation matrix that transforms coordinates from the body frame to the earth-fixed NED frame. The rotational dynamics of the UAV is described by the Euler's equation from literature [51] given by

$$\begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} \frac{I_{yy} - I_{zz}}{I_{xx}}\dot{\theta}\dot{\psi} \\ \frac{I_{zz} - I_{xx}}{I_{yy}}\dot{\phi}\dot{\psi} \\ \frac{I_{xx} - I_{yy}}{I_{zz}}\dot{\theta}\dot{\phi} \end{bmatrix} + \begin{bmatrix} \frac{l}{I_{xx}}\tau_1 \\ \frac{l}{I_{yy}}\tau_2 \\ \frac{l}{I_{zz}}\tau_3 \end{bmatrix} \tag{2}$$

where I_{xx}, I_{yy}, and I_{zz} are the inertia of the UAV about the x, y, and z axes, respectively. l is the distance from the center of mass to the rotor. Following the approach in [52] , the control inputs are defined as the desired velocities in x, y, z directions and yaw rate, while the torques τ_i $(i = 1, 2, 3)$ applied to the UAV and the thrust force F are given by

$$\begin{cases} \tau_1 = -h_{\phi_1}\dot{\phi} + h_{\phi_2}(\phi_d - \phi) \\ \tau_2 = -h_{\theta_1} + h_{\theta_2}(\theta_d - \theta) \\ \tau_3 = h_{\psi_1}\left(\dot{\psi}_d - \dot{\psi}\right) \\ F = mg + mh_{z_1}(\dot{z} - \dot{z}_d) \end{cases} \tag{3}$$

where $h_{\phi_2}, h_{\theta_2}, h_{z_1}, h_{\phi_1}, h_{\theta_1}$, and h_{ψ_1} are control gains of the autopilot. Inspired by the autopilot command presented in [53] , the autopilot inputs are the desired angles ϕ_d and θ_d, which are typically expressed as inverse tangent functions. Under the small angle assumption for ϕ, θ, and ψ, the desired angles ϕ_d and θ_d can be efficiently approximated using linear approximation from [54] . Consequently, the approximate desired angles ϕ_d and θ_d are given by

$$\begin{cases} \theta_d = \dfrac{\pi\left(h_{y_1}(\dot{y}_d - \dot{y})\psi + h_{x_1}(\dot{x}_d - \dot{x})\right)}{4g + 4h_{z_1}(\dot{z}_d - \dot{z})}, \\ \phi_d = \dfrac{\pi\left(h_{x_1}(\dot{x}_d - \dot{x})\psi - h_{y_1}(\dot{y}_d - \dot{y})\right)}{4g + 4h_{z_1}(\dot{z}_d - \dot{z})} \end{cases} \tag{4}$$

where $h_{x_1}, h_{y_1}, h_{z_1}, h_{\phi_2}, h_{\theta_2}, h_{\phi_1}, h_{\theta_1}$, and h_{ψ_1} are control gains of the autopilot.

Remark 2.1. Under the small-angle assumption for ϕ, θ, and ψ, a linearized UAV model is derived for simplicity and ease of analysis based on [15,51,52]. However, this approach may be inadequate for capturing UAV behavior under high-speed flight or aggressive maneuvers, and it can become invalid when no hover point exists (e.g., trajectory tracking). For practical applications, the linearized approach should be carefully verified through simulations or experiments, as it may not hold for aggressive maneuvers or large attitude variations. The initial conditions of the UAV and historical operational data is referred by the actual UAV data from [15]. The model parameters are inspired by the Pixhawk 4 autopilot system. The above practical considerations are essential for the design and implementation of the proposed method.

To formulate the cooperative game of the UAV and human operator, the control input matrix G is assumed to be the same for both UAV and human operator, which means the input channel of the UAV and human operator is the same. The control inputs implemented by the autonomation and human operator are denoted as $\mathcal{U}_a$ and $\mathcal{U}_h$, respectively, and the control input of the UAV is defined as a function of both human and automation inputs, written as $\mathcal{U}(t, \Omega) = \mathcal{U}(\mathcal{U}_a, \mathcal{U}_h)$. Firstly, we define the input $\mathcal{U}$ as the sum of the control input of the autonomy and the control input of the human operator, in which the dynamics model of the UAV could be written in the form of

$$\dot{\Omega} = F(\Omega) + G(\Omega)\mathcal{U} = F(\Omega) + \sum_{i=a,h} G_i(\Omega)\mathcal{U}_i \tag{5}$$

where the dynamics matrix $F(\Omega)$ and control input matrixes $G(\Omega)$, $G_a(\Omega)$ and $G_h(\Omega)$ are defined as

$$F(\Omega) = \begin{bmatrix} 0_{3\times3} & I_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & K_1 & M_1 & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & I_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & K_2 & M_2 & M_3 \end{bmatrix}\Omega, \; G_i = \begin{bmatrix} 0_{3\times4} \\ L_1 \\ 0_{3\times4} \\ L_2 \end{bmatrix} \tag{6}$$

where the matrices K_1, M_1, K_2, M_2, M_3, L_1, and L_2 are

defined as

$$K_1 = \begin{bmatrix} \frac{-k_t}{m} & 0 & 0 \\ 0 & \frac{-k_t}{m} & 0 \\ 0 & 0 & -h_{z_1} - \frac{k_t}{m} \end{bmatrix}, \quad M_1 = \begin{bmatrix} 0 & -g & 0 \\ g & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$K_2 = \begin{bmatrix} 0 & \frac{\pi l h_{\phi_2} h_{y_1}}{4 g I_{xx}} & 0 \\ \frac{-\pi l h_{\theta_2} h_{x_1}}{4 g I_{yy}} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \; M_2 = \begin{bmatrix} \frac{-l h_{\phi_2}}{I_{xx}} & 0 & 0 \\ 0 & \frac{-l h_{\theta_2}}{I_{yy}} & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$M_3 = \begin{bmatrix} \frac{-l h_{\phi_1}}{I_{xx}} & 0 & 0 \\ 0 & \frac{-l h_{\theta_1}}{I_{yy}} & 0 \\ 0 & 0 & \frac{-l h_{\psi_1}}{I_{zz}} \end{bmatrix}, \quad L_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & h_{z_1} & 0 \end{bmatrix},$$

$$L_2 = \begin{bmatrix} 0 & \frac{-\pi l h_{\phi_2} h_{y_1}}{4 g I_{xx}} & 0 & 0 \\ \frac{\pi l h_{\theta_2} h_{x_1}}{4 g I_{yy}} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{l h_{\psi_1}}{I_{zz}} \end{bmatrix}$$

Assumption 2.2. As one of the cooperative game players, the observed state of the human operator to the UAV is $\hat{\Omega}_h$, the action of the human operator is $\hat{\mathcal{U}}_h$. Assume the following conditions hold for the observation and action of the human operator, the communication and interaction between the human operator and the UAV:

(1) Assuming the observation of the human operator to the UAV is accurate and reliable, which means $\hat{\Omega}_h(t) \approx \Omega(t)$. It should be noted that the observation of the human operator to the UAV is not perfect, and there may be some errors in the observation. However, the observing error is small enough to be ignored in the following analysis.
(2) The ground station and UAV maintain a reliable, low-latency communication link that enables real-time communication between the human operator and the UAV. The computation and communication delay between the human operator and the UAV is negligible, which means the action of the human operator is implemented to the UAV immediately as $\hat{\mathcal{U}}_h(t) \approx \mathcal{U}_h(t)$.

To copilot the UAV with human operators to achieve optimal performance, it is essential to design an optimal controller to track the desired trajectory.

3. Problem Formulation: Optimal Shared Control of UAV

3.1. *Formulation of optimal control*

Table 1. Variables and their physical meanings in the UAV system

Variable	Physical Meaning
Position and Motion Variables:	
x, y, z	Position coordinates in NED frame (m)
$\dot{x}, \dot{y}, \dot{z}$	Linear velocities in NED frame (m/s)
ϕ, θ, ψ	Euler angles: roll, pitch, yaw (rad)
p, q, r	Angular rates about body axes (rad/s)
$\dot{\phi}, \dot{\theta}, \dot{\psi}$	Euler angle rates about world axes (rad/s)
$\dot{p}, \dot{q}, \dot{r}$	Angular accelerations about body axes (rad/s^2)
Physical Parameters:	
m	Total mass of UAV (kg)
g	Gravitational acceleration (m/s^2)
I_{xx}, I_{yy}, I_{zz}	Principal moments of inertia (kg·m^2)
k_t	Aerodynamic drag coefficient
l	Distance from center of mass to rotor (m)
Forces and Moments:	
F	Total thrust force (N)
τ_1, τ_2, τ_3	Control moments about body axes (N·m)
$\mathcal{R}_b^{ned}$	Rotation matrix from body to NED frame
Control Variables:	
$\mathcal{U}_a$	Control input from autonomous system
$\mathcal{U}_h$	Control input from human operator
$\mathcal{U}$	Combined control input of UAV
β	Shared control allocation parameter
μ_k	Symmetric saturation bound for control inputs
State and Performance Variables:	
Ω	Complete state vector of UAV system
Ω_e	Tracking error of system states
Q_i	State penalty matrices for tracking error
R_{ik}	Control input weighting matrices
Λ_{ik}	Control input penalty functions

To design the optimal shared controller for the UAV, the problem of optimal control should be formulated. First, the following quadratic cost function for player i can be defined:

$$\mathcal{V}_i(\Omega_e, \mathcal{U}_a, \mathcal{U}_h) = \int_0^\infty r_i(e(\tau), \mathcal{U}_a(\tau), \mathcal{U}_h(\tau))\, d\tau, \; \forall i \in \{a, h\} \tag{7}$$

where $\Omega_e = \Omega - \Omega_d$ is the tracking error of system states, and the instantaneous reward function $r_i(\Omega_e, \mathcal{U}_a, \mathcal{U}_h)$ of the i-th player in the cooperative non-zero sum game is defined as:

$$r_i(\Omega_e, \mathcal{U}_a, \mathcal{U}_h) = \Omega_e^\top Q_i \Omega_e + \sum_{k=a,h} \Lambda_{ik}(\mathcal{U}_k), \; \forall i \in \{a, h\} \tag{8}$$

where $Q_i \in \mathbb{R}^{n \times n}, (i, k = a, h)$ is positive definite state penalty matrices of tracking error Ω_e for player i. To con-

strain the control inputs of both the UAV and human operator, and inspired by the work in [55,56] , $\Lambda_{ik}(\mathcal{U}_k)$ is defined as penalty of the control input $\mathcal{U}_k$ to the i-th player in the form of inverse hyperbolic tangent integral:

$$\Lambda_{ik}(\mathcal{U}_k) = 2\mu_k R_{ik} \int_0^{\mathcal{U}_k} \tanh^{-1}(\gamma_{\mathcal{U}}/\mu_k)\, \mathrm{d}\gamma_{\mathcal{U}} \tag{9}$$

where $i,k = a,h$, $\mu_k \in \mathbb{R}^{m\times 1}$ is the symmetric saturation bound for player k's control input satisfying $-\mu_k \leq \mathcal{U}_k \leq \mu_k$, $R_{ik} \in \mathbb{R}^{m\times m}$ is the positive definite weighting matrix that player i assigns to player k's control input $\mathcal{U}_k$. γ_U is an integral variable. To develop the optimal controller for the UAV, it is essential to evaluate the optimal value function $\mathcal{V}_i^*(\Omega_e)$ and the optimal control input $\mathcal{U}_a^*(\Omega_e)$. To facilitate the succeeding analysis and controller design with the dynamics (5), the following assumption and definition are made for the non-zero sum game of the UAV and human.

Assumption 3.1. [57,58] Assume the following conditions are satisfied for the augment dynamics (5):

(1) On a compact set $\Omega \in \chi \in \mathbb{R}^n$, both $F(\Omega)$ and $G_i(\Omega)$ are Lipschitz continuous with $F(0) = 0$, and $G_i(\Omega)$ satisfied bounded condition $\|G_i(\Omega)\| \leq G_{Hi}$ for all $\Omega \in \chi$.
(2) Cost matrix Q_i and R_{ij} are bounded, such that $\underline{\lambda}_{Q_i} \leq \|Q_i\| \leq \bar{\lambda}_{Q_i}$, $\underline{\lambda}_{R_{ij}} \leq \|R_{ij}\| \leq \bar{\lambda}_{R_{ij}}$, where constants $\underline{\lambda}_{Q_i}, \underline{\lambda}_{R_{ij}} \geq 0$ and $\bar{\lambda}_{Q_i}, \bar{\lambda}_{R_{ij}} > 0$.

Definition 3.2. Consider the two-player non-zero sum game between the UAV and human operator given in 5, given a set of control input $\{\mathcal{U}_a^*, \mathcal{U}_h^*\}$, a Nash equilibrium is achieved if the following conditions are satisfied:

$$\begin{aligned} \mathcal{V}_a^*(\Omega_e) &= \mathcal{V}_a(\Omega_e, \mathcal{U}_a^*, \mathcal{U}_h^*) \leq \mathcal{V}_a(\Omega_e, \mathcal{U}_a, \mathcal{U}_h^*) \\ \mathcal{V}_h^*(\Omega_e) &= \mathcal{V}_h(\Omega_e, \mathcal{U}_a^*, \mathcal{U}_h^*) \leq \mathcal{V}_h(\Omega_e, \mathcal{U}_a^*, \mathcal{U}_h) \end{aligned} \tag{10}$$

The optimal value function $\mathcal{V}_i^*(\Omega_e)$ for player i is given as:

$$\mathcal{V}_i^*(\Omega_e) = \min_{\mathcal{U}_i(\tau)\in\Omega_U} \int_t^\infty r_i(\Omega_e(\tau), \mathcal{U}_a(\tau), \mathcal{U}_h(\tau)) d\tau \tag{11}$$

where $\Omega_U \in \mathbb{R}^{m\times 1}$ is the admissible set of control input. To obtain the optimal value function (11), we introduce the Hamilton function for the optimal control problem:

$$\begin{aligned} H_i(\Omega_e, \mathcal{U}_a, \mathcal{U}_h, \nabla\mathcal{V}_i^*) =& {\Omega_e}^\top Q_i \Omega_e + \Lambda_{ia}(\mathcal{U}_a) + \Lambda_{ih}(\mathcal{U}_h) \\ &+ \nabla{\mathcal{V}_i^*}^\top (F + G_a\mathcal{U}_a + G_h\mathcal{U}_h) \end{aligned} \tag{12}$$

where $\nabla\mathcal{V}_i^* = \frac{\partial\mathcal{V}^*}{\partial\Omega_e}, (i = a,h)$ is the gradient of optimal value function. Following the extreme condition of the value function (11) and the Hamilton function (12), the optimal control input for player i could be derived as:

$$\begin{aligned} \mathcal{U}_i^*(\Omega_e) &= \underset{\mathcal{U}_i(\tau)\in\Omega_U}{\operatorname{argmin}} H_i \\ &= -\mu_i \tanh\left(\frac{R_{ii}^{-1} G_i^\top}{2\mu_i} \nabla\mathcal{V}_i^*\right), \ \forall i \in \{a,h\} \end{aligned} \tag{13}$$

Combining the optimal control input (13) with the Hamilton function (12), the HJB equation is obtained as:

$$\begin{aligned} 0 =& {\Omega_e}^\top Q_i \Omega_e + \Lambda_{ia}(\mathcal{U}_a^*) + \Lambda_{ih}(\mathcal{U}_h^*) \\ &+ (\nabla\mathcal{V}_i^*)^\top (F + G_a\mathcal{U}_a^* + G_h\mathcal{U}_h^*), \ \forall i \in \{a,h\} \end{aligned} \tag{14}$$

The optimal value function (11) and the corresponding saturated optimal control input (13) could be derived by solving the HJB equation (14). Now the problem of optimal control of the UAV is formulated. However, solving the HJB equation (14) is still a complex and challenging problem due to its nonlinearity and high dimensionality. The next section will introduce a novel shared mechanism that collects and allocates control inputs from the human operator and the optimal controller of autonomy.

3.2. *Shared control allocation*

In this subsection, to achieve the closed-loop optimal shared control of UAV, a novel shared mechanism is established, which allocates the relationship between optimal and human control inputs. Given the human control input $\mathcal{U}_h$ and the control input $\mathcal{U}_a$ produced by the optimal controller of autonomy, the shared control input $\mathcal{U}$ is defined as:

$$\mathcal{U} = \mathcal{U}_a + \beta\mathcal{U}_h \tag{15}$$

where $\beta \in [0,1]$ is the shared control parameter. To achieve the optimal shared control of the UAV, methods such as Maxwell's Demon Algorithm (MDA) from [23,45] are studied to set parameter β by judging if the human control input is in the same direction as the optimal control input. However, the MDA is a method similar to the switch control method, which is not continuous and may cause the UAV system to be unstable. In this paper, a novel shared mechanism is proposed to allocate the relationship between the optimal control input and human control input. The ratio of the optimal control input and human control input is defined as:

$$\beta = \begin{cases} 0, & \text{if } \eta \geq \beta_1 \\ 1, & \text{if } \eta \leq \beta_2 \\ \frac{\eta-\beta_1}{\beta_2-\beta_1}, & \text{otherwise} \end{cases} \tag{16}$$

where η is the angle between the vector of optimal control input and the vector of human control input. β_1 and β_2 are the threshold values. In this paper, we choose $\beta_1 = 2\pi/3$ and $\beta_2 = \pi/2$. The control allocation mechanism ensures continuity of the control signal $\mathcal{U}(t)$, preventing sudden jumps in actuation commands. While not necessarily differentiable at switching points where $\beta(t)$ changes, this continuity property is sufficient to maintain system stability and performance, as demonstrated in our experimental results. The continuity of $\mathcal{U}(t)$ prevents abrupt changes in UAV commands, while the potential non-differentiability at switching points has minimal impact on actual system behavior due to the natural mechanical filtering of the UAV dynamics.

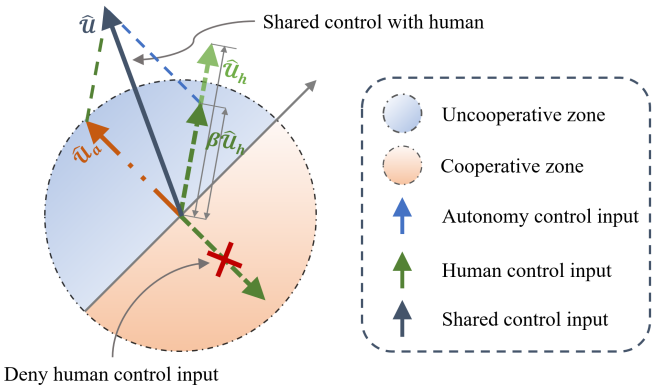


Fig. 2. The mechanism of shared control

The shared mechanism is illustrated in Fig. 2, where the blue slash-dot vector is the optimal control input of the autonomy, the green dotted vector indicates the human control input. The angle between the optimal control input and the human control input is denoted as η. The overall shared control input is constructed by blending the optimal control input and the human control input.

Remark 3.3. when angle η is greater than β_1, the shared control parameter β is set to 0, the UAV are controlled by autonomy. When angle η is less than β_2, the shared control parameter β is set to 1, the UAV are controlled by both the autonomy and the human operator fully. When angle η is between β_1 and β_2, the shared control parameter β is set to the ratio of angle η to the threshold values β_1 and β_2. Compared with existing MDA methods [23,45], the proposed shared mechanism is able to allocate the relationship of control input continuously and effectively, which judges the intention of human operator and autonomy. The continuousness of the control input is guaranteed by the setting of intermediate transition zones.

To learn from the human operator's maneuver data and achieve optimal shared control of the UAV, the optimal controller is approximated by the actor-critic algorithm using historical pilot operation data in the next section.

4. Main results: Cooperative Game-based Optimal Shared Control of UAV

In this section, the design of the actor-critic is presented to solve the optimal shared control problem of the UAV. First, the optimal value function and the optimal control policy are reconstructed using the actor-critic algorithm. With the reconstructed optimal value function and control policy, the bellman error is established. By minimizing the bellman error, the actor-critic neural networks (NNs) are trained to obtain the optimal value function and the optimal control policy.

4.1. *Actor-critic design*

For the approximation of the optimal value function, a structure of actor-critic NNs is developed. The optimal value function for player i is reconstructed by:

$$\mathcal{V}_i^*(\Omega_e) = \mathcal{W}_{ci}^\top \varphi_{ci}(\Omega_e) + \varepsilon_{ci}(\Omega_e),\ \forall i \in \{a,h\} \tag{17}$$

where $\mathcal{W}_{ci} \in \mathbb{R}^{n_{\varphi_{ci}} \times 1}$ is the weights of critic NN, ε_{ci} and ε_{ai} are the construction errors of the actor-critic NNs. To obtain the optimal control input, the actor NNs are utilized to approximate the optimal control policy:

$$\mathcal{U}_i^*(\Omega_e) = -\mu_i \tanh\left(\frac{R_{ii}^{-1} G_i^\top}{2\mu_i}\left(\nabla\varphi_{ai}^\top \mathcal{W}_{ai} + \nabla\varepsilon_{ai}^\top\right)\right), \forall i \in \{a,h\} \tag{18}$$

where $\mathcal{W}_{ai} \in \mathbb{R}^{n_{\varphi_{ai}} \times 1}$ are the weights of the actor NNs. In the practice, the ideal weights $\mathcal{W}_{ci}$ and $\mathcal{W}_{ai}$ are unknown, estimated weights are utilized to approximate the optimal value functions and the control inputs:

$$\hat{\mathcal{V}}_i(\Omega_e) = \hat{\mathcal{W}}_{ci}^\top \varphi_{ci}(\Omega_e),\ \forall i \in \{a,h\} \tag{19}$$

$$\hat{\mathcal{U}}_i(\Omega_e) = -\mu_i \tanh\left(\frac{R_{ii}^{-1} G_i^\top}{2\mu_i} \hat{\mathcal{W}}_{ai}^\top \varphi_{ai}\right),\ \forall i \in \{a,h\} \tag{20}$$

where $\hat{\mathcal{W}}_{ci} \in \mathbb{R}^{n_\varphi \times 1}$ are the estimated weights of the critic NN for the leader and the follower. $\hat{\mathcal{W}}_{ai}$ are the estimated weights of the actor NN. According to the proposed shared mechanism (15) in the last section, the shared control input is obtained by blending the optimal control input and human control input:

$$\hat{\mathcal{U}} = \hat{\mathcal{U}}_a + \beta \hat{\mathcal{U}}_h \tag{21}$$

where $\hat{\mathcal{U}}$ is the shared optimal control input implementing to the UAV, $\hat{\mathcal{U}}_a$ and $\hat{\mathcal{U}}_h$ are the estimated control inputs of the autonomy and the human operator derived from (20), the shared control parameter β is set by the shared mechanism (16).

By inserting the obtained shard control input into the Hamilton function, the shared Bellman error is obtained:

$$\begin{aligned}\delta_i(\Omega_e, \hat{\mathcal{W}}_{ci}, \hat{\mathcal{U}}) =& \nabla\hat{\mathcal{V}}_i^\top \left(F + G\hat{\mathcal{U}}\right) + r(\Omega_e, \hat{\mathcal{U}}) \\ =& \hat{\mathcal{W}}_{ci}^\top \nabla\varphi_{ci}\left(F + G(\hat{\mathcal{U}}_a + \beta\hat{\mathcal{U}}_h)\right) + {\Omega_e}^\top Q_i \Omega_e \\ &+ \Lambda_{ia}(\hat{\mathcal{U}}_a) + \Lambda_{ih}(\beta\hat{\mathcal{U}}_h),\ \forall i \in \{a,h\}\end{aligned} \tag{22}$$

where δ_i is the shared Bellman error. The shared Bellman error is utilized to train actor-critic NNs and approximate optimal value functions and control inputs.

4.2. *Online value function approximation*

In this subsection, weights of actor-critic NNs are updated online by minimizing the Bellman error. The historical stack data set $\{\hat{\mathcal{U}}(t), \delta_i(t), \{\hat{\mathcal{U}}^j(t), \delta_i^j(t)\}_{j=1}^N\}$ is collected without extrapolation but stored as a stack, where $\{\hat{\mathcal{U}}^j(t), \delta_i^j(t)\}$ is the jth historical stored data collection. The weights of actor-critic NNs are learned by minimizing

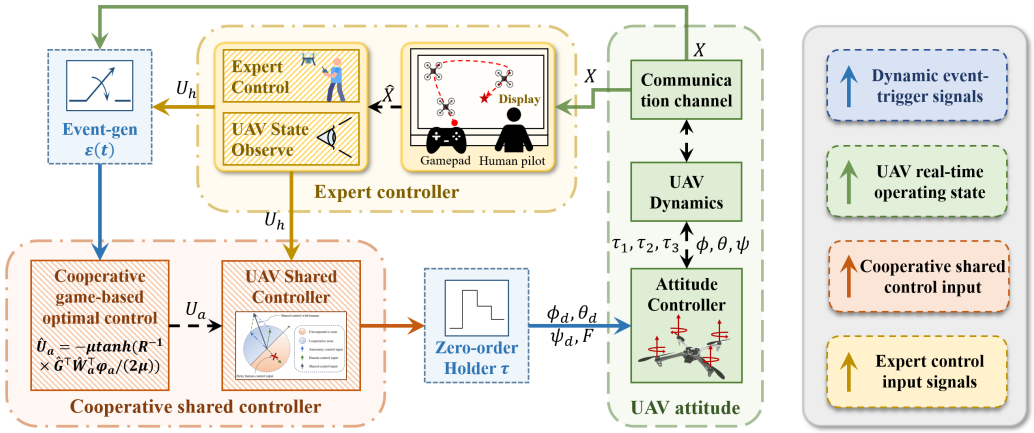


Fig. 3. The proposed cooperative optimal shared control algorithm structure.

a defined squared loss function $E = \delta_i^\top \delta_i + \sum_{k=1}^{N} \delta_i^{k\top} \delta_i^k$. Accordingly, a concurrent gradient descent update law is utilized to update the weights of the critic NN:

$$\dot{\hat{\mathcal{W}}}_{ci} = -\frac{k_{ci1}\delta_i \zeta_i}{\left(\zeta_i^\top \zeta_i + 1\right)^2} - \frac{k_{ci2}}{N} \sum_{k=1}^{N} \frac{\delta_i^k \zeta_i^k}{\left(\zeta_i^{k\top} \zeta_i^k + 1\right)^2}, \forall i \in \{a, h\} \tag{23}$$

where $k_{cij} > 0$ $(i = a, h, j = 1, 2)$ are the learning rates of critic NN. The regression vectors $\zeta_i = \nabla \varphi_{ci}^\top (\Omega_e)(F + G\hat{\mathcal{U}})$, $\zeta_i^k = \nabla \varphi_{ci}^\top ({\Omega_e}^k)(F + G\hat{\mathcal{U}}^k)$, where ${\Omega_e}^k$ is the kth historical data sample. For the actor NN, the weights are updated by a gradient projection update law:

$$\dot{\hat{\mathcal{W}}}_{ai} = \Gamma \left\{ -k_{ai} \mathcal{F}_{ai} \left(\hat{\mathcal{W}}_{ai} - \hat{\mathcal{W}}_{ci} \right) \right\}, \ \forall i \in \{a, h\} \tag{24}$$

where $k_{ai} > 0$ is the learning rates of actor NN. $\mathcal{F}_{ai} \in \mathbb{R}^{n_\varphi \times n_\varphi}$ is positive definite matrices for the updating of actor NN. Γ is a projection operator to ensure the actor NN weights are bounded. Then the online learning of the optimal value function and control input are achieved by actor-critic NNs.During the cooperation task of the UAV and human operator, large amounts of historical data are collected and stored in the experience replay stack, the computational load of controller approximation and communication burden are the key challenges for the online implementation of the proposed algorithm. To solve this problem, a computation and communication-efficient dynamic event-triggering rule is proposed in the next section.

4.3. *Dynamic event-triggering rule*

To reduce the computational load of the controller approximation and the communication burden of the human-UAV system, a dynamic event-triggering rule is proposed to trigger the controller approximation and the communication of the human-UAV system. First, a dynamic variable η is defined to store the information of the triggering event, which is calculated and updated utilizing the following dynamic equation:

$$\begin{aligned} \dot{\eta} &= -\lambda\eta + ((1-\theta)\lambda_{\min}(Q_a)\|\Omega_e\|^2 - \frac{\mathcal{G}_M^2 \mathcal{K}_\zeta}{2} \left\|\hat{\mathcal{W}}_{ai}\right\|^2 \|e_j\|^2) \\ &= -\lambda\eta + \Lambda\left(\Omega_e, e_j\right) \end{aligned} \tag{25}$$

where $e_j = \Omega_e - \Omega_{\mathcal{K}}$ is the error between the current state and the last triggered state. The initial condition of the dynamic variable satisfies $\eta(0) \geq 0$. $\lambda > 0$ is the decay rate of the dynamic variable, which is the key parameter to control the triggering event. $\theta \in [0, 1]$ is the threshold value of the triggering event, which is an adjustable parameter to control the triggering event. $\lambda_{\min}(Q_a)$ is the minimum eigenvalue of automation state penalty matrix Q_a. $\mathcal{G}_M$ is the norm of automation control input matrix G_a. $\mathcal{K}_\zeta$ is a positive upper bounded parameter, which is defined as

$$\mathcal{K}_\zeta = \left(\mathcal{G}_\varphi^2 \mathcal{G}_M^2 + \mathcal{G}_g^2 \varphi_{dM}^2\right) \left\|R_{aa}^{-1}\right\|^2 \tag{26}$$

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 (25) and its dynamic equation, the next triggering time is obtained by the following dynamic event-triggering rule:

$$\begin{aligned} \tau_j = &\inf\left\{ \left[t \in \mathbf{R}_0^+ \,|\, t > \tau_{j-1} \right] \cap [\eta(t) + \alpha \right. \\ &\left. \times \left((1-\theta)\lambda_{\min}(Q_a)\|\Omega_e\|^2 - \frac{\mathcal{G}_M^2 \mathcal{K}_\zeta}{2} \left\|\hat{\mathcal{W}}_{ai}\right\|^2 \|e_j\|^2 \leq 0 \right] \right\} \\ = &\inf\left\{ \left[t \in \mathbf{R}_0^+ \,|\, t > \tau_{j-1} \right] \cap [\eta(t) + \alpha\Lambda\left(\Omega_e, e_j\right) \leq 0] \right\} \end{aligned} \tag{27}$$

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:

$$\hat{\mathcal{U}}(t) = \begin{cases} \hat{\mathcal{U}}_a(\tau_j) + \beta(\tau_j)\hat{\mathcal{U}}_h(\tau_j), & \text{if } t \geq \tau_j \\ \hat{\mathcal{U}}_a(\tau_{j-1}) + \beta(\tau_{j-1})\hat{\mathcal{U}}_h(\tau_{j-1}), & \text{otherwise} \end{cases} \tag{28}$$

where τ_j is the jth triggering time caculated by the dynamic event-triggering rule (27). The proposed dynamic event-triggering rule is able to reduce the computational load of the controller approximation and the communication burden of the human-UAV system. Then the proposed cooperative game-based optimal shared control UAV algorithm is able to achieve the optimal shared control of the UAV with the human operator.

Algorithm 4.1. Cooperative game-based optimal shared control UAV algorithm

1: **Initialize parameters**:
- Weights of actor-critic NNs $\hat{\mathcal{W}}_{ci}$, $\hat{\mathcal{W}}_{ai}$.
- Triggering parameters α, λ, θ, η.
- Historical stack $\{\hat{\mathcal{U}}(t), \delta_i(t), \{\hat{\mathcal{U}}^j(t), \delta_i^j(t)\}_{j=1}^N\}$.

2: **while** $t < T_{end}$ **do**
3: **Collect**: human control input $\mathcal{U}_h$, UAV state Ω
4: **if** Triggering condition (27) satisfied **then**
5: Compute optimal strategies:
- Optimal value function $\hat{\mathcal{V}}_i$ via (11)
- Optimal control input $\hat{\mathcal{U}}_a$ via (13)
- Optimal shared control $\hat{\mathcal{U}}(\tau_j)$ via (21)

6: Evaluate Bellman errors from (22):
- Human bellman error $\delta_1(\Omega_e, \hat{\mathcal{W}}_{c1}, \hat{\mathcal{U}})$ via (22)
- Autonomy bellman error $\delta_2(\Omega_e, \hat{\mathcal{W}}_{c2}, \hat{\mathcal{U}})$ via (22)

7: Update experience replay buffers:
- Human: $[\hat{\mathcal{U}}, \delta_1, [\hat{\mathcal{U}}^j, \delta_1^j]_{j=1}^N]$
- Autonomy: $[\hat{\mathcal{U}}, \delta_2, [\hat{\mathcal{U}}^j, \delta_2^j]_{j=1}^N]$

8: Update actor-critic weights:
- Critic weights via (23)
- Actor weights via (24)

9: **end if**
10: Apply control $\hat{\mathcal{U}}(\tau_j)$ to UAV system
11: Update dynamic variable η by (25)
12: **end while**

The detailed algorithm is shown in Algorithm 4.1. The detailed architecture of the proposed cooperative game-based optimal shared control UAV algorithm is shown in Fig. 3. The proposed algorithm is able to formulate a non-zero sum game between the UAV and human operator, in which the control input of the expert is collected and shared with the autonomy, and the autonomy formulates the cooperative game to obtain the optimal control input, then an innovative shared mechanism (15) is proposed to allocate the relationship between the optimal control input and human control input continuously and efficiently. The actor-critic NNs are trained to approximate the optimal value function and the optimal control input by minimizing the shared Bellman error. In the next section, the stability analysis of the closed-loop system and the theoretical analysis of the dynamic event-triggering rule are presented.

5. Theoretical Analysis

5.1. *Stability analysis of the closed-loop system*

In this subsection, with the help of the Lyapunov stability theory, the closed-loop system states and the actor-critic NN errors are proved to be ultimate uniform bounded (UUB) under the proposed cooperative optimal shared control scheme. First, two assumptions are given for the proof.

Assumption 5.1. [43, 59] Assuming that the following parameters and operators are bounded: $\|\hat{\mathcal{W}}_{ci}\| \leq \mathcal{W}_{Hi}$, $\|\zeta_i(\Omega_e)\| \leq \zeta_{Hij}$, $\|\nabla\zeta_i(\Omega_e)\| \leq \zeta_{D,Hij}$, $\|\varphi(\Omega_e)\| \leq \varphi_{Hi}$, $\|\nabla\varphi(\Omega_e)\| \leq \varphi_{D,Hi}$, $\|\varepsilon(\Omega_e)\| \leq \varepsilon_{Hi}$, $\|\nabla\varepsilon(\Omega_e)\| \leq \varepsilon_{D,Hi}$,

Assumption 5.2. [60] Consider the online collected data and extrapolated dataset for updating the weights. The following persistent excitation conditions are satisfied:

$$\begin{cases} \int_t^{t+T} \frac{\zeta_i(\tau)\zeta_i(\tau)^\top}{(\zeta_i(\tau)^\top\zeta_i(\tau)+1)^2}\mathrm{d}\tau \geqslant \phi_{1,i} I_{\mathcal{K}} \\ \inf_{t\in\mathbf{R}_{t\geq t_0}} \sum_{k=1}^N \frac{\zeta_i^k(t)\zeta_i^k(t)^\top}{N\left(\zeta_i^k(t)^\top\zeta_i^k(t)+1\right)^2} \geqslant \phi_{2,i} I_{\mathcal{K}} \end{cases} \tag{29}$$

where $\zeta_i(\tau)$ and $\zeta_i^k(t)$ are the regression vectors, $I_{\mathcal{K}}$ is an identity matrix with appropriate dimensions, and at least one of the non-negative constants $\phi_{1,i}$, $\phi_{2,i}$ is strictly positive.

Based on the design of input (20), it could be obtained that:

$$\left\|\mathcal{U}_i^*(\Omega_e) - \hat{\mathcal{U}}_i(\Omega_e)\right\|^2 \leq \zeta_i \tilde{W}_{ai}^\top \tilde{W}_{ai} + \Pi_{ui} \tag{30}$$

where $\tilde{W}_{*i} = \hat{\mathcal{W}}_{*i} - \mathcal{W}_{*i}$ is the weights estimation error of NNs, ζ is a upper bound related with φ_H, $\varphi_{D,H}$, ζ_{Hi} and $\zeta_{D,Hi}$, Π_u is a upper bound related to $\varepsilon_{D,H}$. The Bellman error δ_i is abbreviated as:

$$\begin{aligned} \delta_i =& -\zeta_i^\top \tilde{W}_{ci} + \frac{1}{4}\tilde{W}_{ai} G_{\zeta_i} \tilde{W}_{ai} + \frac{1}{4}\tilde{W}_{aj} G_{\zeta_j} \tilde{W}_{aj} \\ &+ \Delta_i(\Omega_e) + \xi_{Hi}, \end{aligned} \tag{31}$$

$$\begin{aligned} \delta_i^k =& -(\zeta_i^k)^\top \tilde{W}_{ci} + \frac{1}{4}\tilde{W}_{ai} G_{\zeta_i}^k \tilde{W}_{ai} + \frac{1}{4}\tilde{W}_{aj} G_{\zeta_j}^k \tilde{W}_{aj} \\ &+ \Delta_i^k(\Omega_e), \end{aligned} \tag{32}$$

where $i, j = a, h$, $j \neq i$, $G_{\zeta_h} = \nabla\varphi_{ah}^\top G_h R_{hh}^{-1} G_h^\top \nabla\varphi_{ah}^\top$, $G_{\zeta_a} = \beta^2 \nabla\varphi_{aa}^\top G_a R_{aa}^{-1} G_a^\top \nabla\varphi_{aa}^\top$, $G_{\zeta_i}^k = G_{\zeta_i}(\Omega_e{}^K)$, and $\Delta_i, \Delta_i^k : \mathbb{R}^n \to \mathbb{R}$ are uniformly bounded on χ, $\overline{\|\Delta_i\|}$ and $\overline{\left\|\Delta_i^k\right\|}$ decrease as $\overline{\|\nabla\varepsilon_i\|}$ and $\overline{\|\nabla W_i\|}$ decrease. The stability analysis of closed-loop system state and network weight estimation errors is given in the following theoretical result.

Theorem 5.3. Consider the augmented system dynamics (5) under the proposed cooperative optimal shared control scheme. Let Assumptions 3.1, 5.1 and 5.2 be satisfied. The actor-critic NNs weights are updated according to the adaptive laws (23) and (24), and the control input is approximated via (20). Then the closed-loop system states Ω and the network weight estimation errors $[\tilde{W}_{ci}^\top, \tilde{W}_{ai}^\top]^\top$ will be ultimately uniformly bounded (UUB), provided that:

$$\|\Xi\| \geq \sqrt{\frac{\Upsilon_{\text{res}}}{\lambda_{\min}(\mathcal{M})}} \tag{33}$$

where $\Xi = \left[\Omega^\top, \tilde{W}_{c1}^\top, \tilde{W}_{a1}^\top, \tilde{W}_{c2}^\top, \tilde{W}_{a2}^\top\right]^\top$ denotes the augmented state vector, Υ_{res} is a positive constant related to the system parameters and the learning rates, $\lambda_{\min}(\mathcal{M})$ is the minimum eigenvalue of the system matrix $\mathcal{M}$.

Proof. Let us analyze the stability based on Lyapunov stability theory. Consider the following Lyapunov function candidate:

$$\mathcal{V}(\Xi) = \sum_{\substack{i,j=a,h \\ j\neq i}} \left\{ \mathcal{V}_i^* + \frac{1}{2}\tilde{W}_{ci}^\top \tilde{W}_{ci} + \frac{1}{2}\tilde{W}_{ai}^\top \tilde{W}_{ai} \right\} \tag{34}$$

Taking the time derivative of the Lyapunov function along the trajectories of system (5), with the optimal value functions (11) and optimal control inputs (13), yields:

$$\dot{\mathcal{V}} = \sum_{\substack{i,j=a,h \\ j\neq i}} \left\{ \nabla \mathcal{V}_i^* \left(F + G_i \mathcal{U}_i + G_k \mathcal{U}_k\right) + \tilde{W}_{ci}^\top \dot{\hat{W}}_{ci}^\top + \tilde{W}_{ai}^\top \dot{\hat{W}}_{ai}^\top \right\} \tag{35}$$

Substituting the $(\nabla \mathcal{V}_i^*)^\top F(\Omega)$ term from (31) and (32) into (35), and employing the Bellman errors from (31) and (32), the derivative can be rewritten as:

$$\begin{aligned}
\dot{\mathcal{V}} = \sum_{i=a,h} \Bigg\{ & + \tilde{W}_{ai}^\top \left[-k_{ai} \mathcal{F}_{ai} \left(\hat{\mathcal{W}}_{ai} - \hat{\mathcal{W}}_{ci} \right) \right] - \Omega^\top Q_i \Omega \\
& - \tilde{W}_{ci}^\top \left[-k_{ci1} \frac{\zeta_i}{\rho_i} \left(-\zeta_i^\top \tilde{W}_{ci} + \Delta_i + \xi_{Hi} \right) \right] - \sum_{j=a,h} \Lambda_{ij}(\mathcal{U}_j) \\
& - \tilde{W}_{ci}^\top \left[-k_{ci1} \frac{\zeta_i}{\rho_i} \left(\frac{1}{4} \tilde{W}_{ai}^\top G_{\zeta_i} \tilde{W}_{ai} + \frac{1}{4} \tilde{W}_{aj}^\top G_{\zeta_j} \tilde{W}_{aj} \right) \right] \\
& - \tilde{W}_{ci}^\top \left[-\frac{k_{ci2}}{N} \sum_{k=1}^{N} \frac{\zeta_i^k}{\rho_i^k} \left(\frac{1}{4} \tilde{W}_{ai}^\top G_{\zeta_i}^k \tilde{W}_{ai} + \frac{1}{4} \tilde{W}_{aj}^\top G_{\zeta_j}^k \tilde{W}_{aj} \right) \right] \\
& - \tilde{W}_{ci}^\top \left[-\frac{k_{ci2}}{N} \sum_{k=1}^{N} \frac{\zeta_i^k}{\rho_i^k} \left(-(\zeta_i^k)^\top \tilde{W}_{ai} + \Delta^k \right) \right] \Bigg\}
\end{aligned} \tag{36}$$

Substitute inequality (30), then employing Young's inequality [28] and assumptions 3.1-5.2, the derivative can be rewritten as:

$$\dot{\mathcal{V}} \leq -\Xi^\top \mathcal{M} \Xi + \Upsilon_{\text{res}}$$

where

$$\mathcal{M} = \begin{bmatrix} m_1 & 0 & 0 & 0 & 0 \\ 0 & m_2 & 0 & 0 & 0 \\ 0 & m_3 & m_4 & 0 & 0 \\ 0 & 0 & 0 & m_5 & 0 \\ 0 & 0 & 0 & m_6 & m_7 \end{bmatrix}$$

is a positive definite matrix, and $m_1 = \underline{\lambda}_{Q_a} + \underline{\lambda}_{Q_h}$, $m_2 = \frac{1}{2}k_{c11}\zeta_1\zeta_1^{\mathrm{T}} + \frac{1}{2}k_{ci2}\phi_2 I_{\mathcal{K}}$, $m_3 = -\mathcal{F}_{a1} I_{\mathcal{K}}$, $m_4 = \mathcal{F}_{a1} I_{\mathcal{K}} - \bar{\lambda}_{R_{aa}} \zeta_u I_{\mathcal{K}}$, $m_5 = \frac{1}{2}k_{c21}\zeta_2\zeta_2^{\mathrm{T}} + \frac{1}{2}k_{ci2}\phi_2 I_{\mathcal{K}}$, $m_6 = \frac{1}{2}k_{c21}\zeta_2\zeta_2^{\mathrm{T}} + \frac{1}{2}k_{ci2}\phi_2 I_{\mathcal{K}}$, $m_7 = \mathcal{F}_{a2} I_{\mathcal{K}} - \bar{\lambda}_{R_{hh}} \zeta_u I_{\mathcal{K}}$, and Υ_{res} is a residual defined as:

$$\begin{aligned}
\Upsilon_{\text{res}} = \sum_{\substack{i,j=a,h \\ j\neq i}} \Bigg\{ & \frac{k_{ci1}}{2} \left[\frac{1}{4} \tilde{W}_{ai}^\top G_{\zeta_i} \tilde{W}_{ai} + \frac{1}{4} \tilde{W}_{aj}^\top G_{\zeta_j} \tilde{W}_{aj} + \Delta_i \right]^2 \\
& + \frac{k_{ci2}}{2} \left[\frac{1}{4} \tilde{W}_{ai}^\top G_{\zeta_{i,k}} \tilde{W}_{ai} + \frac{1}{4} \tilde{W}_{aj}^\top G_{\zeta_{j,k}} \tilde{W}_{aj} + \Delta^k \right]^2 \\
& + \bar{\lambda}_{R_{ii}} \Pi_{ui} + \zeta_j \tilde{W}_{aj}^\top \tilde{W}_{aj} \Bigg\}.
\end{aligned}$$

The stability analysis follows similar approaches as in recent ADP literature [37, 43, 48, 55, 56, 61], but extends to the cooperative game framework. Therefore, when the parameters and initial conditions are properly chosen, an appropriate positive definite matrix $\mathcal{M}$ could be chosen to ensure that the derivative of the Lyapunov function $\dot{\mathcal{V}}$ becomes negative definite. This ensures that the closed-loop system state Ω and the network weight estimation errors $[\tilde{W}_{c1}^\top, \tilde{W}_{a1}^\top, \tilde{W}_{c2}^\top, \tilde{W}_{a2}^\top]^\top$ are ultimately uniformly bounded (UUB) when the condition (33) is satisfied. This result confirms the stability and convergence of the proposed cooperative optimal shared control scheme. This completes the stability proof for the proposed cooperative optimal shared control scheme. □

Remark 5.4. This proof satisfies the classical Lyapunov stability conditions where $\mathcal{V}(\Xi)$ is positive definite, $\dot{\mathcal{V}}$ is negative definite when $\|\Xi\| > \sqrt{\Upsilon_{\text{res}}/\lambda_{\min}(\mathcal{M})}$, and the residual term Υ_{res} provides an ultimate bound. While the current residual bound is sufficient to prove UUB stability, the bound could be improved by: (i) utilizing tighter inequalities beyond Young's inequality; (ii) exploiting the cooperative game structure to reduce conservatism; and (iii) considering additional cross-coupling terms in the Lyapunov function.

5.2. *Theoretical Results of the Dynamic Event-Triggering Rule*

In this subsection, the theoretical results of the dynamic event-triggering rule are presented.

Theorem 5.5. Consider the proposed dynamic event-triggering rule (27). Under the proposed control scheme,

the Zeno behavior is avoided and there exists a positive minimum inter-triggering interval given by:

$$\Delta t_{\min} = \frac{\mathcal{K}}{\mathcal{G}(\mathcal{K}+1)}$$

where $\mathcal{K} = \sqrt{\frac{2(1-\theta)\lambda_{\min}(Q_a)}{\mathcal{G}_M^2 \mathcal{K}_\zeta \|\hat{\mathcal{W}}_{ai}\|^2}}$ and $\mathcal{G} = \frac{\mathcal{G}_M^2}{2\|R_{aa}\|}\varphi_{dM}\left\|\hat{\mathcal{W}}_{ai}\right\| + \mathcal{G}_f$.

Proof. Based on the dynamics of variable η in (25), the triggering event occurs when $\{\eta(t) + \alpha\Lambda(\Omega_e, e_j) \leq 0\}$. Therefore, the triggering condition (27) can be rewritten as:

$$(1-\theta)\lambda_{\min}(Q_a)\|\Omega_e\|^2 \geq \frac{\mathcal{G}_M^2 \mathcal{K}_\zeta}{2}\left\|\hat{\mathcal{W}}_{ai}\right\|^2 \|e_j\|^2 \tag{37}$$

From the controller design (13), the control input $\hat{\mathcal{U}}_i(\Omega_e)$ is bounded by:

$$\left\|\hat{\mathcal{U}}_i(\Omega_e)\right\| \leq \|\mu_i\| \tag{38}$$

Using the control input bound (38), the triggering condition (37) leads to:

$$\|\dot{\Omega}_e\| \leq \mathcal{G}_f\|\Omega_e\| + \mathcal{G}_M\|\mu_i\|\,\|\Omega_e + e_j\| \tag{39}$$

Let $\mathcal{H}(t) = \|e_j/\Omega\|$ denote the ratio between error and state norms. For any $t \in [\tau_j, \tau_{j+1})$, taking the time derivative of $\mathcal{H}$ yields:

$$\dot{\mathcal{H}} = \frac{\mathrm{d}}{\mathrm{d}t}\frac{\sqrt{e_j^\top e_j}}{\sqrt{\Omega_e^\top \Omega_e}} \leq \frac{\|\dot{\Omega}_e\|\|e_j\|}{\|\Omega_e\|\|\Omega_e\|} + \frac{\|\dot{\Omega}_e\|}{\|\Omega_e\|} \leq \mathcal{G}(1+\mathcal{H})^2$$

When $\dot{\mathcal{H}} = \mathcal{G}(1+\mathcal{H})^2$, the growth rate of $\mathcal{H}$ reaches its maximum. Given $\mathcal{H}(0) = 0$, solving the triggering condition (37) yields $\mathcal{H}(\tau) = \tau\mathcal{G}/(1-\tau\mathcal{G})$. Therefore, the minimum inter-triggering interval is $\Delta t_{\min} = \frac{\mathcal{K}}{\mathcal{G}(\mathcal{K}+1)}$, which proves the absence of Zeno behavior in the system. □

Table 2. Parameters of the UAV and update law.

Parameter	Value
Initial:	$\Omega_0 = [-2.1975, 0.7529, -0.4799, 0.1028, -0.0079, -0.0572, -0.0013, 0.0270, 0.5237, -0.0448, 0.0315, -0.0584]^\top$, $\mu_a = \mu_h = 1.0$, $\mathcal{W}_{c1} = \mathcal{W}_{c2} = \mathbf{1}_{12} + rand(12)$, $\mathcal{W}_{a1} = \mathcal{W}_{a2} = \mathbf{1}_{12} + rand(12)$,
DET:	$\lambda = 15$, $\theta = 0.2$, $\mathcal{G}_M = 300$, $\mathcal{K}_\zeta = 0.1$, $\eta(0) = 0.3$, $\alpha = 0.1$,
ADP:	$R_{aa} = R_{hh} = R_{ah} = R_{ha} = 50diag([3,1,0.5,1])$, $Q_a = Q_h = diag([10,10,10,1,1,1,1,1,1,1,1,0])$, $k_{ca1} = k_{ch1} = 0.001$, $k_{ca2} = k_{ch2} = 0.1$, $k_{aa1} = k_{ah1} = 0.01$, $\mathcal{F}_a = \mathcal{F}_h = 3$,
UAV:	$m = 0.579902\text{kg}$, $g = 9.81 \cdot \text{m/s}^2$, $I_{xx} = 0.002261\text{kg} \cdot \text{m}^2$, $I_{yy} = 0.002824\text{kg} \cdot \text{m}^2$, $I_{zz} = 0.002097\text{kg} \cdot \text{m}^2$
Model:	$h_{x_1} = -5.25$, $h_{y_1} = -5.25$, $h_{z_1} = 3$, $k_t = 0.01$ $h_{\phi_2} = 3.50$, $h_{\theta_2} = 3.50$, $h_{\psi_2} = 0.35$, $h_{\phi_1} = 0.40$, $h_{\theta_1} = 0.40$, $h_{\psi_1} = 0.10$,

6. Numerical Simulations

In this subsection, the proposed cooperative game-based shared optimal control algorithm is verified by numerical simulations, where the dynamics of the UAV system is given by (1). The UAV system is controlled by the proposed cooperative game-based shared optimal control algorithm, where the shared control input is calculated by (21). To simulate the control input, the actual control effect of the human operator is implemented by a LQR autopilot controller, in which the simulated signals of the human operator are given the control input solved by the LQR algorithm with the cost function $\mathcal{V} = \Omega^\top Q_{hh}\Omega + \mathcal{U}_h^\top R_{hh}\mathcal{U}_h$. The detailed parameters of the UAV and update law are shown in Table 2. The simulator for this example is implemented in Simulink MATLAB R2023b on a Windows PC with an Intel Core i3-12100 CPU with 4 cores and 24 GB RAM. The time step of the simulation is set to $\Delta t = 10^{-5}$ seconds, and simulation time is set to $T_{end} = 30$ seconds. The desired position of the UAV is set as $p_d = [0, 0, -1.5]^\top$, which means the desired state of the UAV is set as $\Omega_d = [0, 0, -1.5, 0, 0, 0, 0, 0, 0, 0, 0, 0]^\top$.

The historical data used for training the RL-based optimal shared controller could be found in [62], which is collected from a XILO Phreakstyle Freestyle frame equipped with a Pixhawk 4 flight controller in an indoor OptiTrack motion capture lab, where sensor data from IMU and control inputs are recorded at 40 Hz and fused using a Kalman filter for reliable state estimation [15].

To evaluate the performance of the proposed algo-

rithm, the following three methods are compared:

(1) **Proposed**: The proposed cooperative game-based shared optimal control algorithm.
(2) **NZS**: Non-zero sum-based adaptive dynamic programming control method in literature [61] .
(3) **MDA**: Maxwell's demon algorithm-based shared control method in literature [23] .

The detailed update parameters for the 'NZS' and 'MDA' methods are set to the same as our proposed method, which is reasonable and fair for the comparison. The control inputs of the 'NZS' could be represented by the following equation:

$$\mathcal{U}_{NZS} = \hat{\mathcal{U}}_a + \hat{\mathcal{U}}_h \tag{40}$$

where $\hat{\mathcal{U}}_a$ and $\hat{\mathcal{U}}_h$ are the approximated control inputs of the automation and human operator, respectively. It should be noted that both of them are approximated by the actor NNs in the 'NZS' method, which could not be achieved in real-world applications due to the involvement of the human operator. The control inputs of the 'MDA' could be represented by the following equation:

$$\begin{aligned} \mathcal{U}_{MDA} &= \hat{\mathcal{U}}_a + \beta_{MDA}\mathcal{U}_h \\ \beta_{MDA} &= \begin{cases} 1, & \text{if } \mathcal{U}_a \cdot \mathcal{U}_h \geq 0 \\ 0, & \text{otherwise} \end{cases} \end{aligned} \tag{41}$$

where $\hat{\mathcal{U}}_a$ is the approximated control input of the automation, which is approximated by the actor NNs in the 'MDA' method, and $\mathcal{U}_h$ is the control input of the human operator, which is calculated by the LQR autopilot controller. The β_{MDA} is the cooperation factor in the 'MDA' method, which is determined by the sign of the dot product of the control inputs of the automation and human operator, in other words, if the inputs of the automation and human operator are in the same direction, the β_{MDA} is set to 1, otherwise, the β_{MDA} is set to 0.

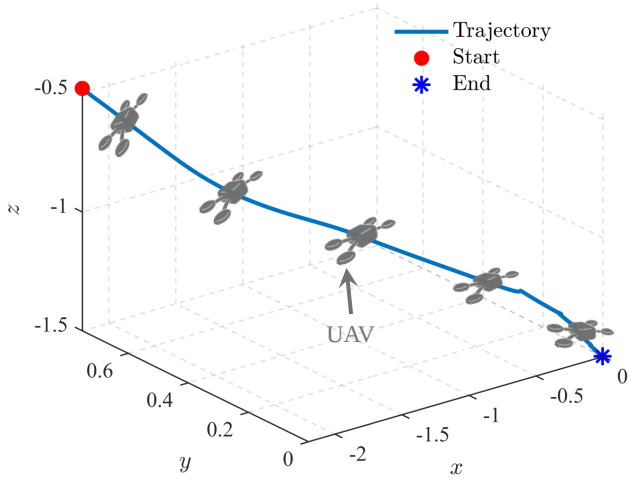


Fig. 4. 3-dimensional trajectory of the UAV system.

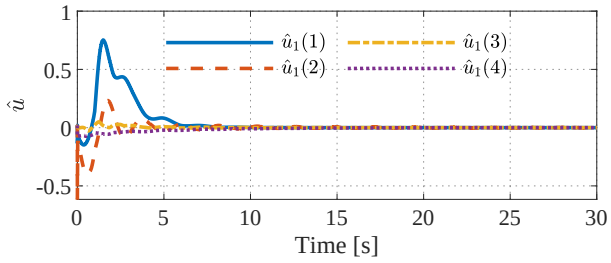


Fig. 5. Simulation results of human control input.

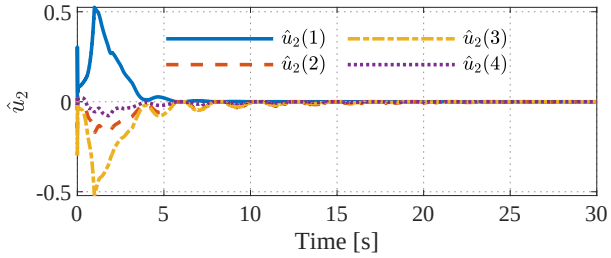


Fig. 6. Simulation results of automation control input.

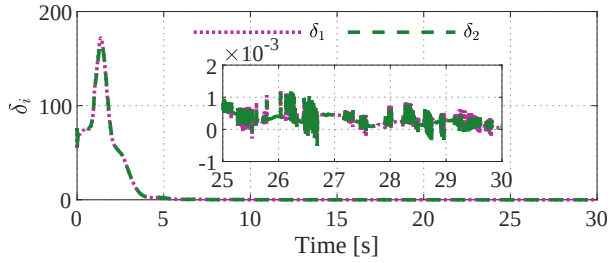


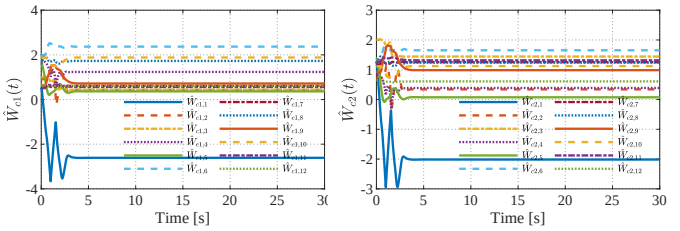
Fig. 7. Simulation results of shared Bellman errors.

6.1. *Basic stabilization performance*

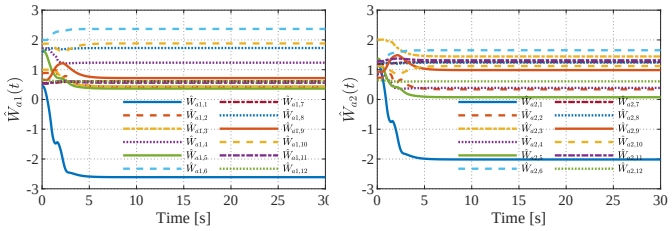
The simulation results of the proposed cooperative game-based shared optimal control algorithm are shown in Fig. 4-9. The 3D trajectory of the UAV system is shown in Fig. 4, where the UAV achieves position stabilization continuously and efficiently under the proposed algorithm with only limited overshoot. The control inputs approximated by the actor NNs are shown in Fig. 5 and Fig. 6. The control inputs remain bounded by the saturation value $\mu_a = \mu_h = 1.0$, and demonstrate smooth and continuous behavior under the shared mechanism with minimal chattering. The Bellman errors of the cooperative non-zero sum game are shown in Fig. 7, where both errors demonstrate fast convergence to small values (less than 0.1) under the proposed algorithm.

The weights of the actor-critic NNs are shown in Fig. 6.1 and Fig. 8. Fig. 6.1 shows the weights of the critic NNs and Fig. 8 shows the weights of the actor NNs. All weights demonstrate stable convergence behavior and remain ultimately uniformly bounded under the proposed algorithm. The UAV system states are shown in Fig. 9 (a)-(d), where the UAV achieves precise position stabilization with satisfactory transient performance under the proposed algorithm. The final control input of the human operator is

simulated using an LQR autopilot controller to emulate realistic human control behavior.



(a)Critic NN 1 weights (b)Critic NN 2 weights



(c)Actor NN 1 weights (d)Actor NN 2 weights

Fig. 8. Simulation results of actor-critic NN weights.

The dynamic event-triggering performance is illustrated in Fig. 9(e) and Fig. 9(f). The triggering mechanism achieves a mean triggering period of $\Delta t_{\min} = 0.0040$ seconds with a total of $N_{\text{trigger}} = 7401$ triggering events. The minimum inter-event time remains strictly positive, confirming the absence of Zeno behavior. Compared to time-triggered methods [40] with $\Delta t_{\min} = 0.001$ seconds, the proposed dynamic event-triggering rule reduces the number of triggering events by 75.33% while maintaining control performance. This demonstrates the effectiveness of our approach in balancing control performance and computational efficiency. The simulation results validate that the proposed cooperative game-based optimal shared control algorithm can achieve precise position stabilization with reduced computational load and event triggers while ensuring continuous human-UAV cooperation.

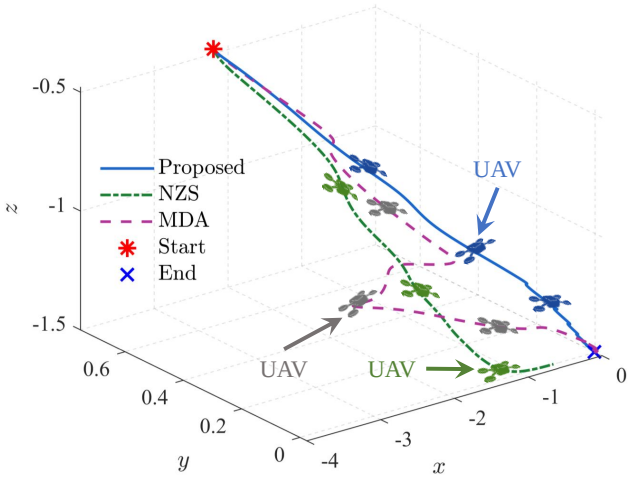


Fig. 10. Comparison results of the UAV trajectory.

6.2. *Comparison of the proposed method with existing methods*

The simulation results of the comparison between the proposed cooperative game-based shared optimal control algorithm and the existing 'NZS' and 'MDA' methods are shown in Fig. 10-13. The 3-dimensional trajectory comparison of the UAV system is shown in Fig. 10, where the proposed cooperative game-based shared optimal control algorithm achieves a better performance in controlling the UAV system trajectory than the existing MDA method. The 'MDA' method stabilizes the UAV system in a twisted trajectory, which is not smooth enough for the UAV system control and will may cause the UAV system to be unstable in real-world applications. The 'NZS' method achieves a smoother trajectory than the 'MDA' method, but it doesn't achieve UUB of the UAV system position state x under the same control parameters setting, also, compared with the 'NZS' method, the proposed cooperative game-based shared optimal control algorithm achieves a smoother and faster performance. The main result of the comparison is shown in Fig. 11-12, where the proposed cooperative game-based shared optimal control algorithm outperforms the existing MDA method in controlling states of x, z, V_x, V_y, ϕ, θ, ψ, $\dot{\phi}$, $\dot{\theta}$, and $\dot{\psi}$.

The detailed comparison of the control input is shown in Fig. 13(a)-(d), where the proposed cooperative game-based shared optimal control algorithm achieves smoother control input than the existing MDA method. It should be noted that the 'NZS' method is the perfect non-zero sum control case, which is not practical in real-world applications due to the involvement of the human operator. Compared with the 'NZS' method, the proposed control algorithm achieves a similar performance in controlling the UAV system, and in the meantime, the 'NZS' method doesn't achieve UUB of the UAV system position state x under the same ADP parameters setting. which is a critical requirement for the UAV system control and will cause the UAV system to be unstable in real-world applications. The comparison results of the Bellman errors are shown in Fig. 13(e)-(f), where the Bellman of the proposed method is more stable and converges to a smaller value than the 'NZS' and 'MDA' methods.

Five quantitative metrics are introduced to evaluate the performance of different approaches as follows:

(1) State smoothness index (SSI)

$$\text{SSI} = \int_T \left\| \frac{d^3 X}{d\tau^3} \right\|^2 d\tau \tag{42}$$

which is the mean-square third-order derivative of UAV states.

(2) Attitude tracking error (ATE)

$$\text{ATE} = \int_T \|\Theta - \Theta_d\|^2 \, d\tau \tag{43}$$

where Θ is the angle of the UAV and Θ_d is the desired angle of the UAV.

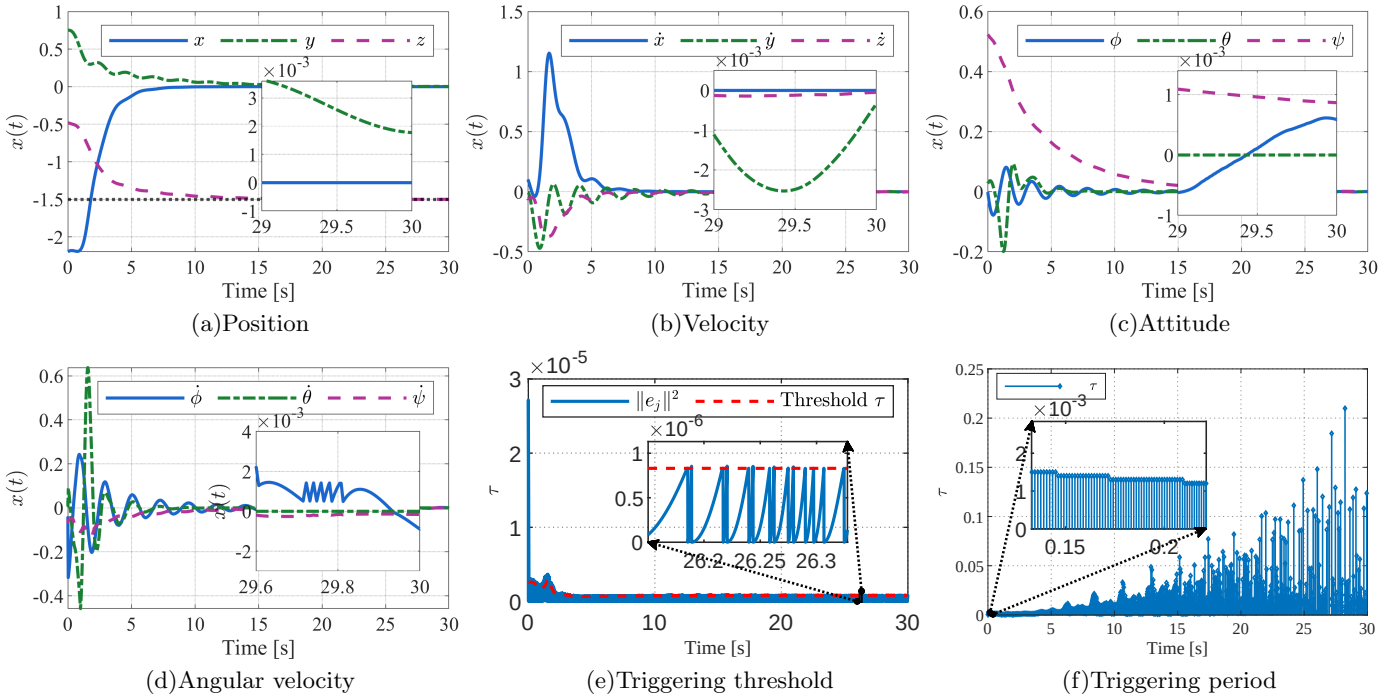


Fig. 9. Simulation results: (a)-(d) UAV system states; (e)-(f) Triggering mechanism.

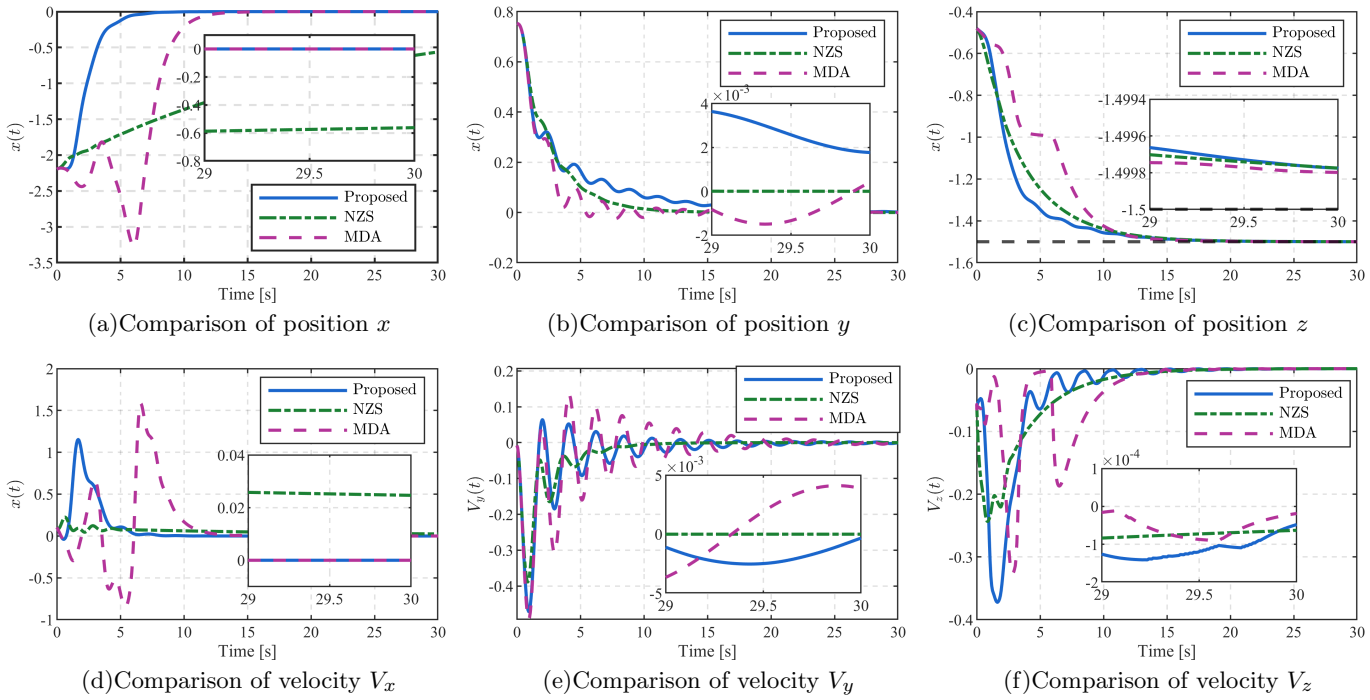


Fig. 11. Comparison results of position, and velocity.

(3) Position tracking error (PTE)

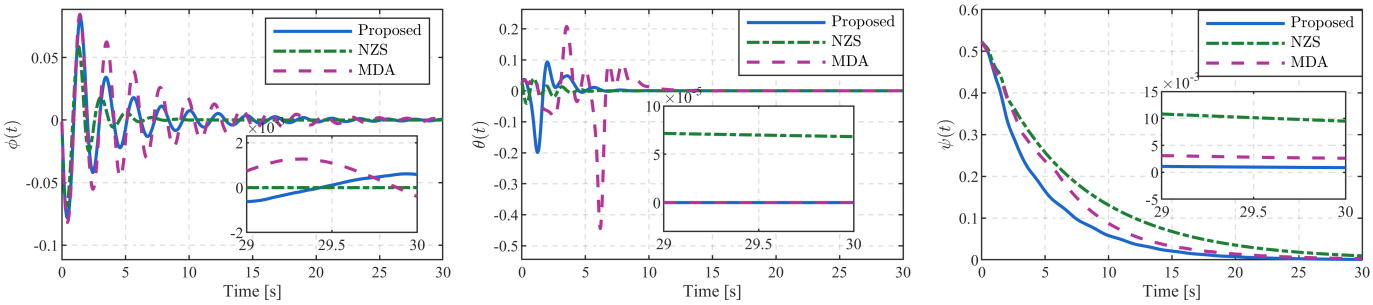
$$\mathrm{PTE} = \int_T \|p - p_d\|^2 \, d\tau \tag{44}$$

where p is the angle of the UAV and p_d is the desired position of the UAV.

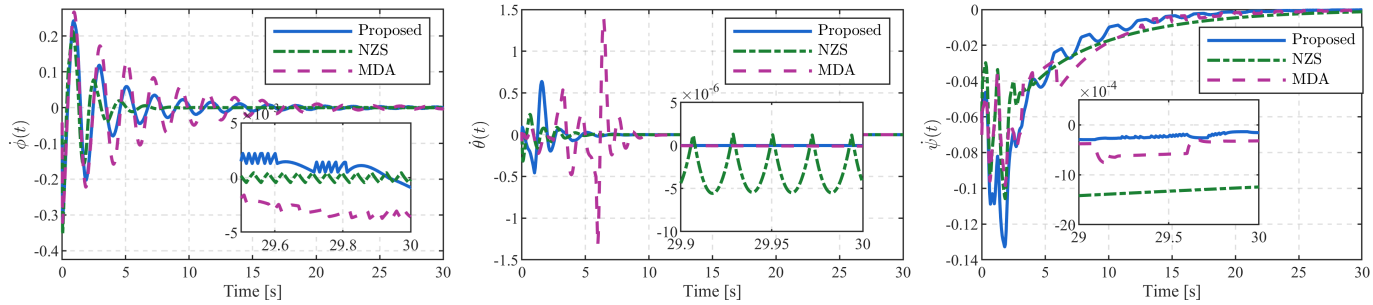
(4) Accumulated Control energy (ACE)

$$\mathrm{ACE} = \int_T \left\|\hat{\mathcal{U}}\right\|^2 d\tau \tag{45}$$

where $\hat{\mathcal{U}}$ is the control input imposed on the UAV.

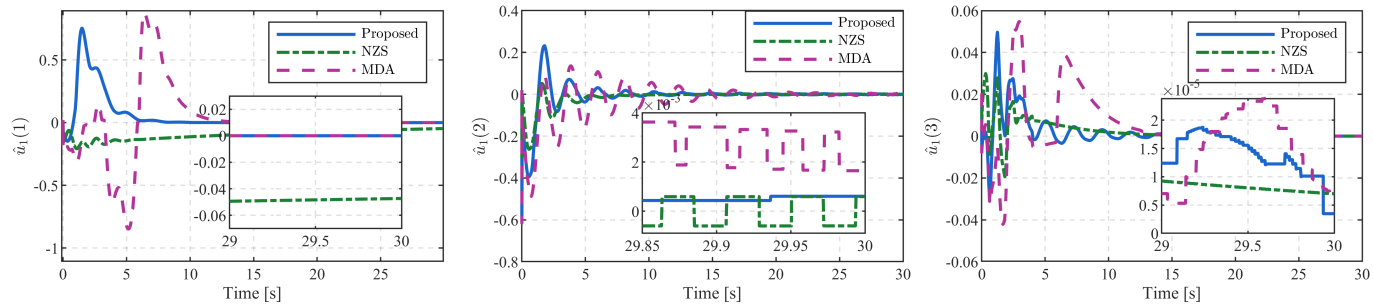


(a)Comparison of attitude ϕ (b)Comparison of attitude θ (c)Comparison of attitude ψ

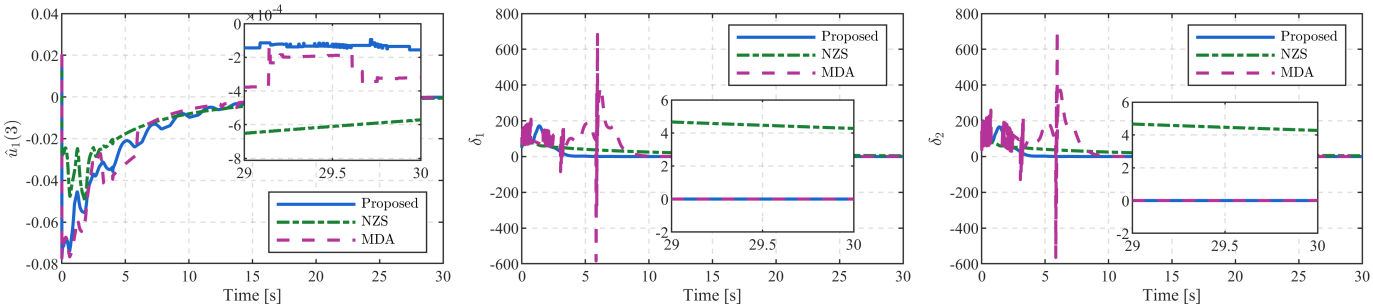


(d)Comparison of ang velocity $\dot{\phi}$ (e)Comparison of ang velocity $\dot{\theta}$ (f)Comparison of ang velocity $\dot{\psi}$

Fig. 12. Comparison results of attitude, and angular velocity.



(a)Comparison of input 1 (b)Comparison of input 2 (c)Comparison of input 3



(d)Comparison of input 4 (e)Comparison of Delta 1 (f)Comparison of Delta 2

Fig. 13. Comparison results of the control input and delta.

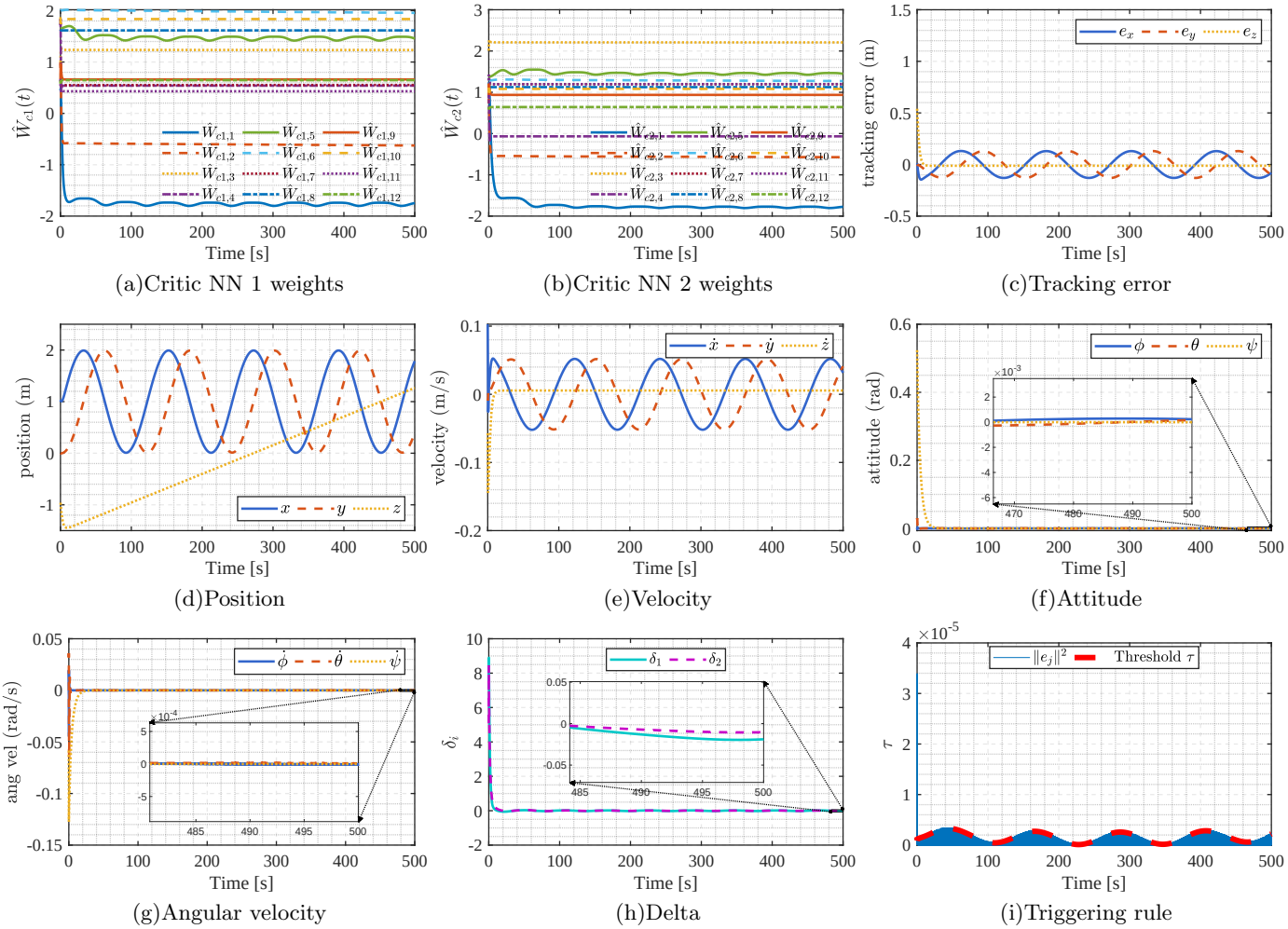
(5) Overall accumulated cost (OAC)

$$\text{OAC} = PTE + ATE \tag{46}$$

which is the sum of the PTE and ATE.

Table 3. Comparison results of example 2.

Method	ATE	PTE	SSI	ACE	OAC
Proposed	**2.4347** ↓	**7.4813** ↓	10.1115	**1.9123** ↓	**11.828** ↓
NZS	3.7580	36.0246	**4.9888**	3.8669	42.8494
Shared	3.2714	20.6823	33.1272	4.5526	28.5062



(a)Critic NN 1 weights (b)Critic NN 2 weights (c)Tracking error

(d)Position (e)Velocity (f)Attitude

(g)Angular velocity (h)Delta (i)Triggering rule

Fig. 14. Tracking performance of the proposed method.

The detailed comparison of the performance is shown in Table 3, where the proposed cooperative game-based shared optimal control algorithm achieves the best performance in terms of the ATE, PTE, ACE, and OAC metrics. Compared with the existing MDA method, the proposed cooperative control algorithm improves the ATE matrices by 25.58%, the PTE matrices by 63.83%, the SSI matrices by 69.48%, the ACE matrices by 57.99%, and the OAC matrices by 58.51%. Compared with the existing NZS method, the proposed cooperative control algorithm improves the ATE matrices by 35.22%, the PTE matrices by 79.23%, the ACE matrices by 50.55%, and the OAC matrices by 58.51%. It should be noted that although the 'NZS' method achieves a better performance in terms of the SSI metric than the proposed method, the 'NZS' method doesn't achieve stabilizing within the set time T_{end} under the same ADP parameters setting, and the proposed method achieves a better performance in terms of the SSI metric than the existing MDA method, which means the proposed controller is much more smooth and efficient in controlling the UAV system than the MDA method.

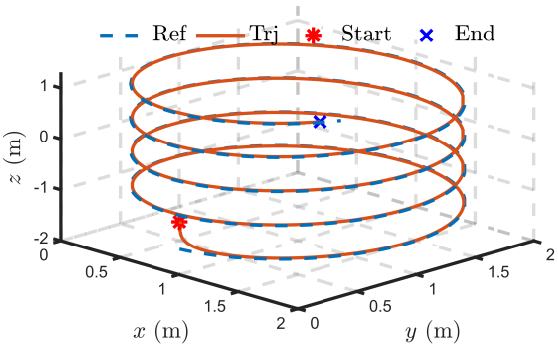


Fig. 15. 3-dimensional trajectory of the UAV system.

6.3. *Tracking performance*

An tracking control simulation is conducted to verify the tracking performance of the proposed cooperative game-based shared optimal control algorithm. In this example, the desired trajectory of the UAV is set as a circular path

with fixed height:

$$\begin{aligned} x_d &= 2\cos(0.4t) \quad (\text{m}) \\ y_d &= 2\sin(0.4t) \quad (\text{m}) \\ z_d &= -1.5 + 0.005t \quad (\text{m}) \end{aligned} \tag{47}$$

The simulation results of tracking performance are shown in Fig. 14-15. To demonstrate the tracking capability, both step responses and circular trajectory tracking scenarios are examined. The tracking error shown in Fig. 14(c) demonstrates fast convergence and remains within 0.2m during steady-state tracking. The critic NN weights (Fig. 14(a)-(b)) exhibit stable convergence within 5 seconds, while the system states (Fig. 14(d)-(f)) maintain bounded trajectories throughout the 30-second simulation.

The dynamic event-triggering mechanism (Fig. 14(f)) effectively reduces computational load while preserving control performance. The 3D trajectory tracking result in Fig. 15 shows the UAV successfully following the desired circular path with a maximum position error of 0.18m. The tracking result confirms the effectiveness of the proposed cooperative shared control scheme in coordinating human-UAV interaction while maintaining precise trajectory tracking. The simulation validates that the proposed method can achieve efficient trajectory tracking while maintaining computational efficiency through event-triggered control and continuous human-automation cooperation.

7. Conclusion

In this paper, a cooperative game-based shared optimal control algorithm is proposed for the UAV system control, where an optimal shared control input is derived by the cooperative game-based shared optimal control algorithm. The proposed method establishes a non-zero sum game between the human operator and the shared control input, where the human operator and the shared control input are obtained by the actor-critic neural networks. An innovative shared mechanism is proposed to achieve cooperation between the human operator and the shared control input continuously and efficiently. The proposed method is verified by numerical simulations, where the proposed optimal shared control algorithm could achieve up to 79.23% improvement compared with the existing MDA methods, and the computational complexity is reduced by 75.33% compared with the existing time-triggered methods. The future work will focus on the real-world implementation of the proposed cooperative game-based shared optimal control algorithm.

References

[1] R. Zhang, L. Dou, B. Xin, C. Chen, F. Deng and J. Chen, A Review on the Truck and Drone Cooperative Delivery Problem, *Unmanned Systems* **12** (September 2024) 823–847.

[2] A. P. Dani, I. Salehi, G. Rotithor, D. Trombetta and H. Ravichandar, Human-in-the-Loop Robot Control for Human-Robot Collaboration: Human Intention Estimation and Safe Trajectory Tracking Control for Collaborative Tasks, *IEEE Control Systems Magazine* **40** (December 2020) 29–56.

[3] A. Garg and S. S. Jha, Deep deterministic policy gradient based multi-UAV control for moving convoy tracking, *Engineering Applications of Artificial Intelligence* **126** (November 2023) p. 107099.

[4] D. Panagou, D. M. Stipanovic and P. G. Voulgaris, Distributed Coordination Control for Multi-Robot Networks Using Lyapunov-Like Barrier Functions, *IEEE Transactions on Automatic Control* **61** (March 2016) 617–632.

[5] A. Franchi, C. Secchi, M. Ryll, H. H. Bulthoff and P. R. Giordano, Shared Control : Balancing Autonomy and Human Assistance with a Group of Quadrotor UAVs, *IEEE Robotics & Automation Magazine* **19** (September 2012) 57–68.

[6] A. allam, A. Nemra and M. Tadjine, Cooperative Guidance of a Multi-quadrotors System with Switching and Reduced Network Information Exchange: Application to Uncooperative Target Entrapping, *Unmanned Systems* **12** (July 2024) 799–818.

[7] N. Hamdadou, H. Bouadi, N. Hebablia and M. Yazid, Stochastic Estimator Based Adaptive Sliding Mode Control for a Quadrotor in Rainy Flight Conditions, *Unmanned Systems* **12** (January 2024) 133–148.

[8] B. Wang and Y. Zhang, Adaptive Sliding Mode Fault-Tolerant Control for an Unmanned Aerial Vehicle, *Unmanned Systems* **05** (October 2017) 209–221.

[9] A. B. Farjadian, A. M. Annaswamy and D. Woods, Bumpless Reengagement Using Shared Control between Human Pilot and Adaptive Autopilot, *IFAC-PapersOnLine* **50** (July 2017) 5343–5348.

[10] J. Tan, S. Xue, H. Cao and S. S. Ge, Human–AI interactive optimized shared control, *Journal of Automation and Intelligence* (January 2025) p. S2949855425000024.

[11] A. Perrusquía, W. Guo, B. Fraser and Z. Wei, Uncovering drone intentions using control physics informed machine learning, *Communications Engineering* **3** (February 2024) p. 36.

[12] X. Dai, C. Ke, Q. Quan and K.-Y. Cai, RFlySim: Automatic test platform for UAV autopilot systems with FPGA-based hardware-in-the-loop simulations, *Aerospace Science and Technology* **114** (July 2021) p. 106727.

[13] E. Eraslan, Y. Yildiz and A. M. Annaswamy, Shared Control Between Pilots and Autopilots: An Illustration of a Cyberphysical Human System, *IEEE Control Systems* **40** (December 2020) 77–97.

[14] A. Perrusquia, W. Guo, B. Fraser and Z. Wei, Uncovering drone intentions using control physics informed machine learning, preprint, In Review (July 2023).

[15] J. Town, Z. Morrison and R. Kamalapurkar, Pilot Performance Modeling via Observer-Based Inverse Rein-

forcement Learning, *IEEE Transactions on Control Systems Technology* (2024) 1–8.

[16] M. Marcano, S. Díaz, J. Pérez and E. Irigoyen, A Review of Shared Control for Automated Vehicles: Theory and Applications, *IEEE Transactions on Human-Machine Systems* **50** (December 2020) 475–491.

[17] A. B. Farjadian, B. Thomsen, A. M. Annaswamy and D. D. Woods, Resilient Flight Control: An Architecture for Human Supervision of Automation, *IEEE Transactions on Control Systems Technology* **29** (January 2021) 29–42.

[18] H. Wen, W. Fu, W. Chen, J. Huan, C. Li and X. Duan, Imitation Learning and Teleoperation Shared Control With Unit Tangent Fuzzy Movement Primitives, *IEEE Transactions on Fuzzy Systems* (2024) 1–15.

[19] J. Tan, J. Wang, S. Xue, H. Cao, H. Li and Z. Guo, Human–Machine Shared Stabilization Control Based on Safe Adaptive Dynamic Programming With Bounded Rationality, *International Journal of Robust and Nonlinear Control* (March 2025) p. rnc.7931.

[20] B. Barros Carlos, A. Franchi and G. Oriolo, Towards Safe Human-Quadrotor Interaction: Mixed-Initiative Control via Real-Time NMPC, *IEEE Robotics and Automation Letters* **6** (October 2021) 7611–7618.

[21] H. Virani, D. Liu and D. Vincenzi, The Effects of Rewards on Autonomous Unmanned Aerial Vehicle (UAV) Operations Using Reinforcement Learning, *Unmanned Systems* **09** (October 2021) 349–360.

[22] M. Li, J. Qin, Q. Ma, Y. Shi and W. X. Zheng, Master-Slave Safe Cooperative Tracking via Game and Learning Based Shared Control, *IEEE Transactions on Automatic Control* (2024) 1–8.

[23] A. Broad, I. Abraham, T. Murphey and B. Argall, Data-driven Koopman operators for model-based shared control of human–machine systems, *The International Journal of Robotics Research* **39** (August 2020) 1178–1195.

[24] Y. Yang, H. Jiang, C. Hua and J. Li, Practical Preassigned Fixed-Time Fuzzy Control for Teleoperation System Under Scheduled Shared-Control Framework, *IEEE Transactions on Fuzzy Systems* **32** (February 2024) 470–482.

[25] X. Xing, W. Li, S. Yuan and Y. Li, Fuzzy Logic-Based Arbitration for Shared Control in Continuous Human–Robot Collaboration, *IEEE Transactions on Fuzzy Systems* **32** (July 2024) 3979–3991.

[26] X. Yi, H. Liu, Y. Wang, H. Duan and K. P. Valavanis, Safe Reinforcement Learning-Based Visual Servoing Control for Quadrotors Tracking Unknown Ground Vehicles, *IEEE Transactions on Intelligent Vehicles* (2024) 1–11.

[27] V. S. Donge, B. Lian, F. L. Lewis and A. Davoudi, Data-Efficient Reinforcement Learning for Complex Nonlinear Systems, *IEEE Transactions on Cybernetics* (2023) 1–12.

[28] Y. Yang, Z. Ding, R. Wang, H. Modares and D. C. Wunsch, Data-Driven Human-Robot Interaction Without Velocity Measurement Using Off-Policy Reinforcement Learning, *IEEE/CAA Journal of Automatica Sinica* **9** (January 2022) 47–63.

[29] M. Li, J. Qin, Z. Wang, Q. Liu, Y. Shi and Y. Wang, Optimal Motion Planning Under Dynamic Risk Region for Safe Human–Robot Cooperation, *IEEE/ASME Transactions on Mechatronics* (2024) 1–11.

[30] W. Lu, H. Liu and Q. Gao, Data-Driven Cooperative Multi-Task Assignment in Heterogeneous Multi-Agent Pursuit-Evade Game, *Unmanned Systems* **12** (May 2024) 523–533.

[31] S. S. K. S. Narayanan, D. Tellez-Castro, S. Sutavani and U. Vaidya, SE(3) Koopman-MPC: Data-driven Learning and Control of Quadrotor UAVs, *IFAC-PapersOnLine* **56** (January 2023) 607–612.

[32] B. Lian, Y. Kartal, F. L. Lewis, D. G. Mikulski, G. R. Hudas, Y. Wan and A. Davoudi, Anomaly Detection and Correction of Optimizing Autonomous Systems With Inverse Reinforcement Learning, *IEEE Transactions on Cybernetics* **53** (July 2023) 4555–4566.

[33] H.-N. Wu and M. Wang, Human-in-the-Loop Behavior Modeling via an Integral Concurrent Adaptive Inverse Reinforcement Learning, *IEEE Transactions on Neural Networks and Learning Systems* (2023) 1–12.

[34] K. Xie, Adaptive optimal output regulation of unknown linear continuous-time systems by dynamic output feedback and value iteration, *Control Engineering Practice* (2023).

[35] K. Xie, X. Yu and W. Lan, Optimal output regulation for unknown continuous-time linear systems by internal model and adaptive dynamic programming, *Automatica* **146** (December 2022) p. 110564.

[36] S. Zhao, J. Wang, H. Xu and B. Wang, ADP-Based Attitude-Tracking Control with Prescribed Performance for Hypersonic Vehicles, *IEEE Transactions on Aerospace and Electronic Systems* (2023) 1–14.

[37] Q. Ma, P. Jin and F. L. Lewis, Guaranteed Cost Attitude Tracking Control for Uncertain Quadrotor Unmanned Aerial Vehicle Under Safety Constraints, *IEEE/CAA Journal of Automatica Sinica* **11** (June 2024) 1447–1457.

[38] J. Tan, S. Xue, H. Li, Z. Guo, H. Cao and D. Li, Prescribed Performance Robust Approximate Optimal Tracking Control Via Stackelberg Game, *IEEE Transactions on Automation Science and Engineering* (2025) 1–1.

[39] S. Xue, B. Luo, D. Liu and Y. Gao, Event-Triggered ADP for Tracking Control of Partially Unknown Constrained Uncertain Systems, *IEEE Transactions on Cybernetics* **52** (September 2022) 9001–9012.

[40] L. Dong, Y. Tang, H. He and C. Sun, An Event-Triggered Approach for Load Frequency Control With Supplementary ADP, *IEEE Transactions on Power Systems* **32** (January 2017) 581–589.

[41] Y. Wu, M. Chen, H. Li and M. Chadli, Event-Triggered Distributed Intelligent Learning Control of Six-Rotor UAVs Under FDI Attacks, *IEEE Transactions on Artificial Intelligence* **5** (July 2024) 3299–

3312.
[42] H. Zhang, Y. Zhang and X. Zhao, Event-Triggered Adaptive Dynamic Programming for Hierarchical Sliding-Mode Surface-Based Optimal Control of Switched Nonlinear Systems, *IEEE Transactions on Automation Science and Engineering* **21** (July 2024) 4851–4863.
[43] C. Mu, K. Wang and T. Qiu, Dynamic Event-Triggering Neural Learning Control for Partially Unknown Nonlinear Systems, *IEEE Transactions on Cybernetics* **52** (April 2022) 2200–2213.
[44] J. Tan, S. Xue, Q. Guan, K. Qu and H. Cao, Finite-time safe reinforcement learning control of multi-player nonzero-sum game for quadcopter systems, *Information Sciences* **712** (September 2025) p. 122117.
[45] E. Tzorakoleftherakis and T. D. Murphey, Controllers as filters: Noise-driven swing-up control based on Maxwell's demon, *2015 54th IEEE Conference on Decision and Control (CDC)*, (December 2015), pp. 4368–4374.
[46] J. Tan, S. Xue, Z. Guo, H. Li, H. Cao and B. Chen, Data-driven optimal shared control of unmanned aerial vehicles, *Neurocomputing* **622** (March 2025) p. 129428.
[47] Y. Li, G. Carboni, F. Gonzalez, D. Campolo and E. Burdet, Differential game theory for versatile physical human–robot interaction, *Nature Machine Intelligence* **1** (January 2019) 36–43.
[48] 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* **50** (January 2014) 193–202.
[49] J. Tan, S. Xue, H. Li, H. Cao and D. Li, Safe Stabilization Control for Interconnected Virtual-Real Systems via Model-based Reinforcement Learning, *2024 14th Asian Control Conference (ASCC)*, (July 2024), pp. 605–610.
[50] J. Tan, S. Xue, Q. Guan, T. Niu, H. Cao and B. Chen, Unmanned aerial-ground vehicle finite-time docking control via pursuit-evasion games, *Nonlinear Dynamics* (March 2025).
[51] S. Bouabdallah, A. Noth and R. Siegwart, PID vs LQ control techniques applied to an indoor micro quadrotor, *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, **3** (September 2004), pp. 2451–2456 vol.3.
[52] S. Bouabdallah and R. Siegwart, Full control of a quadrotor, *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, (October 2007), pp. 153–158.
[53] B. Xu and K. Sreenath, Safe Teleoperation of Dynamic UAVs Through Control Barrier Functions, *2018 IEEE International Conference on Robotics and Automation (ICRA)*, (IEEE, Brisbane, QLD, May 2018), pp. 7848–7855.
[54] S. Rajan, S. Wang, R. Inkol and A. Joyal, Efficient approximations for the arctangent function, *IEEE Signal Processing Magazine* **23** (May 2006) 108–111.
[55] H. Modares and F. L. Lewis, Optimal tracking control of nonlinear partially-unknown constrained-input systems using integral reinforcement learning, *Automatica* **50** (July 2014) 1780–1792.
[56] M. Abu-Khalaf and F. L. Lewis, Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach, *Automatica* **41** (May 2005) 779–791.
[57] L. Cui, B. Pang and Z.-P. Jiang, Learning-Based Adaptive Optimal Control of Linear Time-Delay Systems: A Policy Iteration Approach, *IEEE Transactions on Automatic Control* **69** (January 2024) 629–636.
[58] J. Tan, S. Xue, H. Cao and H. Li, Safe Human-Machine Cooperative Game with Level-k Rationality Modeled Human Impact, *2023 IEEE International Conference on Development and Learning (ICDL)*, (November 2023), pp. 188–193.
[59] P. Deptula, H.-Y. Chen, R. A. Licitra, J. A. Rosenfeld and W. E. Dixon, Approximate Optimal Motion Planning to Avoid Unknown Moving Avoidance Regions, *IEEE Transactions on Robotics* **36** (April 2020) 414–430.
[60] M. H. Cohen and C. Belta, Safe exploration in model-based reinforcement learning using control barrier functions, *Automatica* **147** (January 2023) p. 110684.
[61] R. Kamalapurkar, J. R. Klotz, P. Walters and W. E. Dixon, Model-Based Reinforcement Learning in Differential Graphical Games, *IEEE Transactions on Control of Network Systems* **5** (March 2018) 423–433.
[62] J. Town and R. Kamalapurkar, Pilot Performance mod-
eling via observer-based inverse reinforcement learning, *https://codeocean.com/capsule/7831037/tree/v1* (April 2024).

Shuangsi Xue received the B.E. degree in electrical engineering and automation from Hunan University, Changsha, China, in 2014, and the M.E. and Ph.D. degrees in electrical engineering from Xian Jiaotong University, Xian, China, in 2018 and 2023, respectively.

He is currently an Assistant Professor at the School of Electrical Engineering, Xian Jiaotong University. His current research interest includes adaptive control and data-driven control of networked systems.

Junkai Tan received the B.E. degree in electrical engineering at the School of Electrical Engineering in Xi'an Jiaotong University, Xi'an, China. He is currently working toward the M.E. degree in electrical engineering at the School of Electrical Engineering, Xi'an Jiaotong University.

His current research interest includes adaptive dynamic programming and inverse reinforcement learning.

Zihang Guo received the B.E. degree in electrical engineering at the School of Electrical Engineering in Xi'an Jiaotong University, Xi'an, China. He is currently working toward the M.E. degree in electrical engineering at the School of Electrical Engineering, Xi'an Jiaotong University.

His current research interest includes neural network and sliding mode-based path planning and tracking methods.

Qingshu Guan received the B.S. degree in automation from North China Electric Power University, Beijing, China, in 2020, and the M.S. degree from the School of Electric Engineering, Xi'an Jiaotong University, Xi'an, China, in 2023. He is currently pursuing the Ph.D. degree with the School of Electric Engineering, Xi'an Jiaotong University, Xi'an, China.

His current interests include deep reinforcement learning and strategy optimization.

Kai Qu received his B.S. degree in Electrical Engineering and Automation from Xidian University (XDU), Xi'an, China. He then earned both his M.E. and Ph.D. degrees in Electrical Engineering from Xi'an Jiaotong University (XJTU), Xi'an, China, in 2020 and 2024, respectively.

He is currently an Assistant Professor at the School of Electrical Engineering, Xian Jiaotong University. His research interests include time series analysis, machine learning, and their applications.

Hui Cao received the B.E., M.E., and Ph.D. degrees in electrical engineering from Xi'an Jiaotong University, Xi'an, China, in 2000, 2004, and 2009, respectively.

He is a Professor at the School of Electrical Engineering, Xi'an Jiaotong University. He was a Postdoctoral Research Fellow at the Department of Electrical and Computer Engineering, National University of Singapore, Singapore, from 2014 to 2015. He has authored or coauthored over 30 scientific and technical papers in recent years. His current research interest includes knowledge representation and discovery. Dr. Cao was a recipient of the Second Prize of National Technical Invention Award.