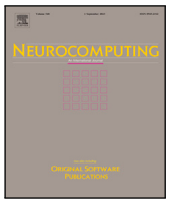
Neurocomputing 622 (2025) 129428

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom



Data-driven optimal shared control of unmanned aerial vehicles

Junkai Tan [a,b], Shuangsi Xue [a,b,c,*], Zihang Guo [a,b], Huan Li [a,b], Hui Cao [a,b], Badong Chen [c]

[a] School of Electrical Engineering, Xi'an Jiaotong University, Xi'an, 710049, China
[b] The State Key Laboratory of Electrical Insulation and Power Equipment, School of Electrical Engineering, Xi'an Jiaotong University, Xi'an, 710049, China
[c] The National Key Laboratory of Human–Machine Hybrid Augmented Intelligence, National Engineering Research Center for Visual Information and Applications, and the Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an, 710049, China

ARTICLE INFO

Communicated by D. Wang

Keywords:
Shared control
Reinforcement learning
Optimal control
Koopman operator
Unmanned aerial vehicle
Approximate dynamic programming

ABSTRACT

Cooperation between humans and autonomy is a critical topic of unmanned aerial vehicle (UAV) control. How to co-pilot the UAV with human operator to achieve optimal performance presents a significant challenge. In this paper, we propose a novel data-driven optimal shared control method for UAV using the Koopman operators to predict the nonlinear dynamics of the UAVs. An original shared control mechanism is established to allocate the relationship between optimal and human control inputs. The model of the system is learned from human maneuver data via the Koopman operator approach, and the optimal controller is approximated online using reinforcement learning techniques. The Lyapunov theory analyzes the stability of the proposed method. Compared with offline RL methods, the proposed method can learn the optimal controller online without a precise UAV dynamics model from human maneuver data. The effectiveness of the proposed method is demonstrated by numerical and Human-in-the-loop (HiTL) simulation.

1. Introduction

For the unmanned aerial vehicle (UAV) autonomous controller design, the precise dynamics model of the UAV are often necessitated. Nonetheless, the exact mathematical dynamics of the UAV are challenging to quantify due to intricate nonlinearity and uncertainty. Methods such as system identification [1,2], dynamic mode decomposition [3,4], and Koopman operator approach [5,6] are investigated to identify the unknown system dynamics. To obtain the drifted nonlinear dynamics of UAV, adaptive system identification approach is explored to facilitate the design of model-based controllers [7,8]. Dynamic mode decomposition is a data-driven technique that linearizes the nonlinear dynamics of the UAV and forecasts the system's future state [9,10]. For intricate nonlinear systems such as autonomous vehicles and power systems, the Koopman operator approach is employed to extend original state-space into a high-dimensional Hilbert space, which predicts the nonlinear dynamics [11,12]. The accuracy of measured system dynamics significantly impacts the efficacy of autonomy control. To design effective controllers for UAV, it is critical to estimate the unknown UAV dynamics accurately.

The delegation of authority between human operators and autonomous controllers is an urgent concern in the domain of human–machine collaboration. Effectively and smoothly co-piloting UAV alongside human operators to achieve optimal performance remains challenging. A substantial quantity of research has been conducted to improve the collaboration between human operators and autonomous controllers, including switch control methods [13], shared control methods [14,15], and hierarchical control methods [16–18]. However, the effectiveness and smoothness of the shared control of UAV are difficult to achieve. The direct shared control method is the most common method to allocate the relationship between the autonomous controller and human operator. The shared control parameter is set by the human operator's intention and the autonomy's decision [19,20], which is difficult to determine and may cause the UAV to be unstable and unsafe due to the sudden switch of control authority. The indirect shared control method is another method to allocate the relationship between the autonomous controller and human operator, in which the automation obtains the human operator's intention and adjusts the control input accordingly [21–23]. The input from humans is not directly used to control the UAV, but to adjust the control input of the autonomy. The hierarchical control method is a multi-level structure containing high-level decision-making and low-level control [24,25]. In the hierarchical control method, the human operator is always responsible for high-level decision-making which provides a high-level goal, while the autonomy is responsible for low-level control which generates the control input to achieve the high-level goal [26–28]. Cooperative control of complex nonlinear systems such as UAV demands a smooth and efficient control mechanism to establish the interaction between

* Corresponding author.
E-mail address: xssxjtu@xjtu.edu.cn (S. Xue).

https://doi.org/10.1016/j.neucom.2025.129428
Received 24 October 2024; Received in revised form 6 December 2024; Accepted 8 January 2025
Available online 14 January 2025

the autonomous controller and the human operator.

Exploitation or integration of human operator experience presents additional challenge for the shared control of UAV. The experience of human operator collected from their maneuvers is a valuable asset for the controller design. However, the utilization or assimilation of human operator experience is hard to achieve. Methods such as reinforcement learning (RL) [29–31], imitation learning [32,33], and inverse reinforcement learning [34–36] are employed to derive the optimal controller from the practical experiences of human operator. Inverse learning algorithms are investigated to learn the inverse optimal controller from the human operator's experience [37,38]. The imitation learning method learns a similar controller from the human operator's demonstration [39]. Approximate dynamic programming (ADP) is a learning-based control method that approximates the optimal value and policy functions from interactions in the RL manner [40–42]. Integral reinforcement learning is studied to cooperatively learn to control UAV with expert pilot's demonstration data [43]. Other methods such as model predictive control [44], and optimized backstepping control [45] are investigated to exploit the human operator's experience. Those methods require a large amount of human operator experience to learn optimal controller, which is inefficient and impractical for complex tasks of highly nonlinear UAV systems. To exploit human operator experience effectively and efficiently, data-driven control method is a promising way for learning and co-piloting UAV with human operators to achieve optimal performance.

Motivated by the aforementioned challenges of human-UAV cooperative control, a novel data-driven optimal shared control method is proposed in this paper. The proposed method employs Koopman operators to forecast the nonlinear dynamics of UAV based on the maneuver data of human operator. The optimal controller is approximated online utilizing model-based RL techniques with forecasted UAV dynamics, and an experience replay stack is used to archive historical operation data. An efficient and smooth shared control mechanism is established to allocate the relationship between the autonomous optimal and human control inputs. Human-in-the-loop (HiTL) simulations are conducted to evaluate the performance of the proposed method. The contributions of this paper are summarized as follows:

1. An innovative data-driven optimal shared control method is proposed for the cooperative control of UAV. With employing the data-driven approaches of Koopman operators and RL control, the proposed method could learn the optimal controller online from human maneuver data without a precise UAV dynamics model, which can copilot UAV with human operators while achieving optimal performance. The proposed method does not require precise UAV dynamics and large amount of maneuver data for the optimal controller learning, compared with optimal control methods [29,30,34,38,45,46].
2. We propose a smooth shared control mechanism to allocate the relationship between autonomous optimal control input and human control input, which judges the cooperative intention of human operators and integrates both human and autonomous control inputs according to an adaptive parameter. This shared control mechanism could achieve better and smoother performance compared with authority allocation mechanisms in [6, 47].
3. The optimal controller is approximated online utilizing RL approach that exploits human operator experience, while the model of the UAV system is learned from human maneuver data via the Koopman approach. Our method is able to learn optimal controller online efficiently using data from human maneuvers, in contrast to offline training methods [12,13]. The performance of the controller is evaluated by HiTL simulations.

The rest of this paper is organized as follows. In Section 2, we present the UAV attitude dynamics model and the Koopman operator.

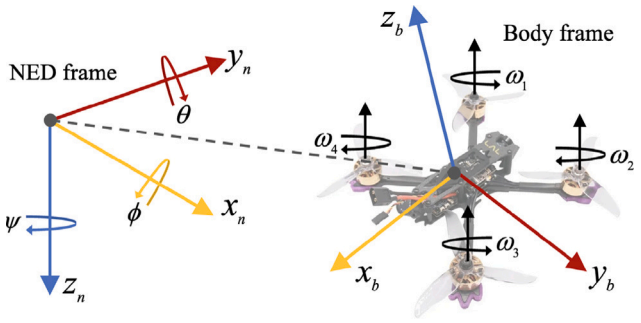


Fig. 1. The configuration of the UAV.

In Section 3, the problem of optimal shared control of UAV is formulated. Section 4 provides the main result of data-driven optimal shared control of UAV. In Section 5, we provide the UAV simulation of the proposed method. Finally, Section 6 concludes the paper.

Notation: The following notation will be used throughout the paper: $\mathcal{R}$, $\mathcal{R}^n$, $\mathcal{R}^{m\times n}$ denote the real number set, n-dimensional real space, and $m\times n$ real matrix, respectively, ∇ denotes the gradient operator, † denotes the Moore–Penrose pseudo inverse, $\|\cdot\|$ denotes the Euclidean norm.

2. Preliminaries and system description

2.1. Attitude dynamics model of UAV

The classical configuration of the body frame and North-East-Down (NED) frame of UAV is shown in Fig. 1. The attitude dynamics model of UAV could be formulated as an Euler–Lagrange equation in the form of:

$$M\ddot{\Phi} = -C(\Phi,\dot{\Phi})\dot{\Phi} + \mathcal{T} + W(\Phi,\dot{\Phi}) \tag{1}$$

where $M = \mathrm{diag}\left(\left[J_\phi, J_\theta, J_\psi\right]\right) \in \mathcal{R}^{3\times3}$ denotes the inertial matrix. $\Phi = [\phi,\theta,\psi]^\top \in \mathcal{R}^{3\times1}$ indicates the angle collection of roll, pitch, and yaw Euler angles, which are bounded by $\phi \in \left[-\frac{\pi}{2},\frac{\pi}{2}\right]$, $\theta \in \left[-\frac{\pi}{2},\frac{\pi}{2}\right]$, $\psi \in [-\pi,\pi]$. $C(\Phi,\dot{\Phi}) : \mathcal{R}^{6\times1} \to \mathcal{R}^{3\times3}$ denotes the coupled Coriolis term. $\mathcal{T} = \left[\gamma_\phi,\gamma_\theta,\gamma_\psi\right]^\top \in \mathcal{R}^{3\times1}$ is the input torque generated by $\gamma_\phi = \alpha_l\alpha_w u_\phi$, $\gamma_\theta = \alpha_l\alpha_w u_\theta$, and $\gamma_\psi = \alpha_\gamma u_\psi$, where α_l is the distance from the center of UAV mass to each rotor, α_w is the thrust factor of the UAV, and α_γ is the drag factor of the UAV, u_ϕ, u_θ, u_ψ are the control inputs generated by $u_\phi = \omega_1^2-\omega_3^2$, $u_\theta = \omega_2^2-\omega_4^2$ and $u_\psi = \omega_1^2+\omega_3^2-\omega_2^2-\omega_4^2$, where ω_j $(j = 1,\ldots,4)$ is the speed of UAV's jth rotor. $W(\Phi,\dot{\Phi}) = \left[w_1,w_2,w_3\right]^\top : \mathcal{R}^{6\times1} \to \mathcal{R}^{3\times1}$ is the uncertain disturbance which vanishes when the UAV is in a stable state. Define $B_\gamma = M^{-1}\times\mathrm{diag}\left(\left[\alpha_l\alpha_w,\alpha_l\alpha_w,\alpha_\gamma\right]\right) \in \mathcal{R}^{3\times3}$ as the gain matrix of control input. The Lagrange-formed attitude dynamics model (1) could be reformulated as:

$$\ddot{\Phi} = -M^{-1}(C(\Phi,\dot{\Phi})\dot{\Phi}) + B_\gamma U + \Omega(\Phi,\dot{\Phi}) \tag{2}$$

where $u = \left[u_\phi,u_\theta,u_\psi\right]^\top$ is the overall control input of the UAV, $\Omega(\Phi,\dot{\Phi})$ is the transformed uncertain disturbance derived by $\Omega(\Phi,\dot{\Phi}) = M^{-1}\times W(\Phi,\dot{\Phi})$. Define $x = \left[\Phi^\top,\dot{\Phi}^\top\right]^\top \in \mathcal{R}^{6\times1}$ as the system state of the dynamics system (2). Then the attitude dynamics model (2) could be written in a general nonlinear affine-input form:

$$\dot{x} = f(x) + g(x)u + d(x) \tag{3}$$

where the drift dynamics matrix $f(x)$, control input matrix $g(x)$ and uncertain disturbance matrix $d(x)$ are defined as

$$f = \begin{bmatrix} 0_{3\times3} & I_{3\times3} \\ 0_{3\times3} & -M^{-1}C \end{bmatrix} x, g = \begin{bmatrix} 0_{3\times3} \\ B_\gamma \end{bmatrix}, d = \begin{bmatrix} 0_{3\times1} \\ \Omega \end{bmatrix}.$$

Consider the following dynamics of desired trajectory $x_d = \left[\Phi_d^\top,\dot{\Phi}_d^\top\right]^\top \in \mathcal{R}^{6\times1}$ as $\dot{x}_d = f_d(x_d)$, where $f_d : \mathcal{R}^{6\times1} \to \mathcal{R}^{6\times1}$ is the drift

dynamics matrix of the desired trajectory. Define the tracking error of the desired trajectory as $e = x - x_d \in \mathcal{R}^{6\times1}$. Subtracting the nonlinear-formed system dynamics (3) and the dynamics of the desired trajectory, the following tracking error dynamics could be obtained:

$$\dot{e} = \dot{x} - \dot{x}_d = \left[f(x) - f_d(x_d)\right] + g(x)u + d(x) \tag{4}$$

Accordingly, the nonlinear-formed system dynamics (3), the tracking error dynamics (4), and the dynamics of the desired trajectory could be augmented to the following dynamics:

$$\dot{X} = F(X) + G(X)U + D(X) \tag{5}$$

where $X = [e^\top, x_d^\top]^\top \in \mathcal{R}^{12\times1}$ is the augmented state, $U = \left[u^\top, 0_{1\times3}\right]^\top \in \mathcal{R}^{6\times1}$ is the augmented control input, and the augmented dynamics matrices are defined as:

$$F(X) = \begin{bmatrix} f(e+x_d) - f_d(x_d) \\ f_d(x_d) \end{bmatrix}$$

$$G(X) = \begin{bmatrix} g(e+x_d) & 0_{6\times3} \\ 0_{6\times3} & 0_{6\times3} \end{bmatrix},\ D(X) = \begin{bmatrix} d(e+x_d) \\ 0_{6\times1} \end{bmatrix}$$

To facilitate the succeeding analysis and controller design with the augmented dynamics (5), the following assumption is made for the UAV dynamics model.

Assumption 1. The following conditions are assumed to be satisfied for the augment dynamics (5):

1. The drift dynamics matrix $f(x)$ and control input matrix $g(x)$ are Lipschitz continuous with respect to x.
2. The uncertain disturbance matrix $D(X)$ is bounded by a known function $L_D(X)$, i.e., $\|D(X)\| \le L_D(X)$ with $L_D(0) = 0$.

It can be seen that the augmented dynamics (5) is a complex nonlinear affine-input system with uncertain disturbances. To copilot the UAVs with human operators to achieve optimal performance, it is essential to design an optimal controller to track the desired trajectory. However, the optimal controller is difficult to obtain due to the complex nonlinear dynamics and uncertain disturbances of the UAVs. In the following subsection, we will introduce the Koopman operator approach to predict the nonlinear dynamics of the UAVs.

2.2. Koopman operator

For an uncontrolled nonlinear dynamic system evolving on the state-space $\mathcal{R}^n$, the state-space representation of the system is given by:

$$x_{t+1} = \mathcal{F}(x_t) \tag{6}$$

where $\mathcal{F} : \mathcal{R}^n \to \mathcal{R}^n$ is the nonlinear drift dynamics matrix of the system. To predict the future state of the system, the Koopman operator approach is utilized to predict the nonlinear dynamics of the UAVs, which extends the state space to a high-dimensional Hilbert space. The Koopman operator $\mathcal{K} : \mathcal{M} \to \mathcal{M}$ is defined as:

$$(\mathcal{K}\Psi)\left(x_{t+1}\right) = \Psi\left(\mathcal{F}\left(x_t\right)\right) \tag{7}$$

where $\Psi : \mathcal{R}^n \to \mathcal{M}$ is the transfer function, which transforms the original state-space to the Hilbert space of Koopman operator. The above Koopman operator is defined under the setting of an uncontrolled nonlinear dynamics system. However, for the controlled nonlinear dynamical system, the Koopman operator should be extended to the controlled setting. Define the extended system state as $\chi_t = \left[x_t^\top, u_t^\top\right]^\top$, then the extended system dynamics could be written as:

$$\chi_{t+1} = \overline{\mathcal{F}}(\chi_t) := \begin{bmatrix} \mathcal{F}(x, u(0)) \\ \mathcal{S}u \end{bmatrix} \tag{8}$$

where $\mathcal{S}$ is the left shift operator, satisfies $(\mathcal{S}u)_i = u_{i+1}$, u_i indicates the ith sequential element of control inputs. The Koopman operator $\mathcal{K} : \mathcal{M} \to \mathcal{M}$ for the extended system dynamics could be redefined as:

$$(\mathcal{K}\Theta)(\chi) = \Theta(\overline{\mathcal{F}}(\chi)) \tag{9}$$

where $\Theta : \mathcal{R}^{n+\ell} \to \mathcal{R}$ is the transfer function, which transforms the extended state-space to the Hilbert space of Koopman operator. The above Koopman operator is defined under the setting of a controlled nonlinear dynamical system. This Koopman operator approach is utilized to predict the nonlinear dynamics of the UAVs, which extends the state space to a high-dimensional Hilbert space. In the next section, we will introduce the optimal shared control of UAVs based on the Koopman operator approach.

3. Problem formulation: Optimal shared control of UAV

3.1. Extended dynamic mode decomposition

Precise mathematical dynamics of the UAVs are difficult to obtain due to the complex nonlinear dynamics and uncertain disturbances. To predict the nonlinear dynamics of the UAVs, the extended dynamic mode decomposition (EDMD) algorithm is utilized to approximate the Koopman operator $\mathcal{K}$. EDMD is a data-driven method that approximate the Koopman operator in the following form:

$$\Theta\left(\chi_{t+1}\right) = \mathcal{K}^\top\Theta\left(\chi_t\right) + \epsilon\left(\chi_t\right) \tag{10}$$

where $\Theta : \mathcal{R}^{n+\ell} \to \mathcal{R}$ is the transfer function defined in the last section, $\epsilon\left(\chi_t\right)$ is the approximation error of the Koopman operator. Collect the data $\left(\chi_j, \chi_{j+1}\right), j = 1, \dots, N_\mathcal{K}$ from the system dynamics $\chi_{j+1} = \mathcal{F}\left(\chi_j\right)$, To approximate the Koopman operator $\mathcal{K}$, we seek matrix K which minimizes the following objective function:

$$E_\mathcal{K} = \sum_{j=1}^{N_\mathcal{K}} \left\|\epsilon\left(\chi_j\right)\right\|^2 = \sum_{j=1}^{N_\mathcal{K}} \left\|\Theta\left(\chi_{j+1}\right) - K^\top\Theta\left(\chi_j\right)\right\|^2 \tag{11}$$

where $\Theta(\chi_j) = \left[\Theta_1(\chi_j), \dots, \Theta_{N_\Theta}(\chi_j)\right]^\top$ is a vector of basis transfer functions $\Theta_i : \mathcal{R}^{n+\ell} \to \mathcal{R}$, $i = 1, \dots, N_\Theta$. Given a set of data collected from the system dynamics, $X_\mathcal{K} = \left[\Theta(\chi_1), \dots, \Theta(\chi_{(N_\mathcal{K}-1)})\right]$ is the transformed state data, $Y_\mathcal{K} = \left[\Theta(\chi_2), \dots, \Theta(\chi_{N_\mathcal{K}})\right]$ denotes the transformed output state data, $U_\mathcal{K} = \left[\Theta(u_1), \dots, \Theta(u_{N_\mathcal{K}})\right]$ indicates the transformed control input data, in which relation $\chi_{j+1} = \overline{\mathcal{F}}\left(\chi_i, u_i\right)$ is satisfied. With the collected data set $X_\mathcal{K}$, $Y_\mathcal{K}$ and $U_\mathcal{K}$, Linearized system dynamics matrix $A \in \mathcal{R}^{n\times n}$ and $B \in \mathcal{R}^{n\times m}$ could be obtained as the solution to the minimization of (11), which is given by:

$$[A, B] = Y_\mathcal{K}\left[X_\mathcal{K}, U_\mathcal{K}\right]^\dagger \tag{12}$$

where † is the operator of Moore–Penrose pseudo inverse. With the linearized system dynamics matrix A and B, the estimated drift dynamics matrix and estimated control input matrix for the augmented system (5) could be obtained as:

$$\hat{F} = \begin{bmatrix} A\times(e+x_d) - f_d(x_d) \\ f_d(x_d) \end{bmatrix},\ \hat{G} = \begin{bmatrix} B & 0_{6\times3} \\ 0_{6\times3} & 0_{6\times3} \end{bmatrix} \tag{13}$$

The nonlinear dynamics of the UAVs are predicted by the Koopman operator approach. To achieve the optimal shared control of UAVs, the optimal control problem is formulated in the next subsection.

3.2. Formulation of optimal control

In the last subsection, the system dynamics is obtained using Koopman operator and EDMD algorithm. To design the optimal shared controller for the UAV, the problem of optimal control should be formulated. First, the following quadratic cost function can be defined:

$$J(X, U) = \int_{t_o}^{\infty} (r(X(\tau), U(\tau)))\, d\tau \tag{14}$$

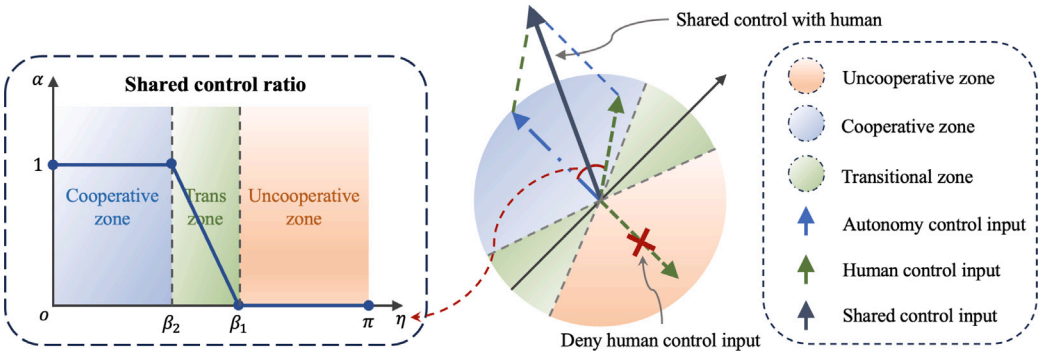


Fig. 2. The mechanism of shared control.

where the saturated control input $U(t)$ satisfies $-\mu \le U(t) \le \mu$, in which μ is the symmetric constraint for the control input, which is set as the maximum control input of the UAV, and the instantaneous reward function $r(X(\tau), U(\tau))$ is given as:

$$r(X,U) = X^\top QX + \Xi(U) \tag{15}$$

where $Q \in \mathbb{R}^{n\times n}$ is positive definite state penalty matrices for the state X. $\Xi(U)$ is the penalty of control input:

$$\Xi(U) = 2R\int_0^U \left(\mu \tanh^{-1}\left(\frac{\zeta_U}{\mu}\right)\right) \mathrm{d}\zeta_U. \tag{16}$$

where $R \in \mathbb{R}^{n\times n}$ denotes the positive definite control input penalty matrix. ζ_U is an integral variable. To develop the optimal controller for the UAV, it is essential to evaluate the optimal value function $J^*(X)$ and the optimal control input $U^*(X)$. The optimal value function $J^*(X)$ is given as:

$$J^*(X) = \min_{U(\tau)\in\Omega_U} \int_t^\infty (r(X(\tau), U(\tau)))\, d\tau \tag{17}$$

where $\Omega_U \in \mathbb{R}^{m\times 1}$ is the admissible set of control input. To obtain the optimal value function (17), we introduce the Hamilton function for the optimal control problem:

$$H(X,U,\nabla J^*) = X^\top QX + \Xi(U) + (\nabla J^*)^\top (F + GU + D) \tag{18}$$

where $\nabla J^* = \frac{\partial J^*}{\partial X}$ is the gradient of optimal value function. Following the extreme condition of the value function (17) and the Hamilton function (18), the optimal control input could be derived as:

$$U^*(X) = -\mu \tanh\left(\frac{R^{-1}G^\top}{2\mu}\left(\nabla J^*(X)\right)^\top\right) \tag{19}$$

Combining the optimal control input (19) with the Hamilton function (18), the HJB equation is obtained as:

$$0 = X^\top QX + \Xi(U^*) + (\nabla J^*)^\top (F + GU^*) \tag{20}$$

The optimal value function (17) and the corresponding saturated optimal control input (19) could be derived by solving the HJB equation (20). Now the problem of optimal control of the UAV is formulated. However, solving the HJB equation (20) is still a complex and challenging problem due to its nonlinearity and high dimensionality. The next section will introduce a novel shared control mechanism that collects and allocates control inputs from the human operator and the optimal controller of autonomy.

Remark 1. Control inputs of UAV systems are inherently constrained by physical limitations in practical applications. These constraints are naturally incorporated into the cost function (17) and penalty function (19) through well-defined saturation bounds. The choice of hyperbolic tangent function *tanh* in (19) is motivated by its smoothness and natural boundedness properties as discussed in [2]. Such bounds effectively prevent control saturation while ensuring system stability. The specific value of the saturation bound μ is determined directly from the physical constraints of the UAV system. Both human and autonomy control inputs are subject to identical constraints in the shared control mechanism, maintaining consistency with the constraints in the optimal control formulation.

3.3. Shared control allocation

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

$$\mathcal{U} = U^* + \alpha U_h \tag{21}$$

where $\alpha \in [0,1]$ is the shared control parameter.

To achieve the optimal shared control of the UAV, methods such as Maxwell's Demon Algorithm (MDA) from [6,47] are studied to set parameter α 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 smooth and may cause the UAV system to be unstable. In this paper, a novel shared control 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:

$$\alpha = \begin{cases} 0, & \text{if } \eta \ge \beta_1 \\ 1, & \text{if } \eta \le \beta_2 \\ \frac{\eta-\beta_1}{\beta_2-\beta_1}, & \text{otherwise} \end{cases} \tag{22}$$

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 shared control 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, when angle η is greater than β_1, the shared control parameter α is set to 0, the UAVs are controlled by the autonomy. When angle η is less than β_2, the shared control parameter α is set to 1, the UAVs 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 α is set to the ratio of angle η to the threshold values β_1 and β_2. Compared with MDA methods in [6,47], the proposed shared control mechanism is able to allocate the relationship of control input smoothly and effectively, which judges the intention of human operator and autonomy. The smoothness 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.

Remark 2. The shared control mechanism ensures optimality through blending of optimal and human control inputs. When $\alpha = 0$, only the optimal control input is applied, achieving global optimality. When $\alpha = 1$, the system is driven by human control input alone, which is generally suboptimal. For $0 < \alpha < 1$, the blended shared control provides suboptimal performance. However, when $\alpha = 1$ and $\eta = 0$, meaning the human input aligns with the optimal input direction, the shared control achieves optimality despite full human control authority.

4. Main results: Data-driven optimal shared control

In this section, the design of the actor-critic is presented to solve the optimal shared control problem of the UAVs. 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. Approximation of value function via actor-critic

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

$$J^*(X) = W_c^\top \varphi_c(X) + \varepsilon_c(X) \tag{23}$$

where $W_c \in \mathbb{R}^{n_{\varphi_c} \times 1}$ is the weights of critic NN, ε_c and ε_a 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:

$$U^*(X) = -\mu \tanh\left(R^{-1}\hat{G}^\top\left(\nabla\varphi_a^\top(X)W_a + \nabla\varepsilon_a^\top\right)/(2\mu)\right) \tag{24}$$

where $W_a \in \mathbb{R}^{n_{\varphi_a} \times 1}$ are the weights of the actor NNs. In the practice, the ideal weights W_c and W_a are unknown, estimated weights are utilized to approximate the optimal value functions and the control inputs:

$$\hat{J}(X) = \hat{W}_c^\top \varphi_c(X) \tag{25}$$

$$\hat{U}(X) = -\mu \tanh\left(R^{-1}\hat{G}^\top \hat{W}_a^\top \varphi_a(X)/(2\mu)\right) \tag{26}$$

where $\hat{W}_c \in \mathbb{R}^{n_\varphi \times 1}$ are the estimated weights of the critic NN. $\hat{W}_a$ are the estimated weights of the actor NN.

According to the proposed shared control mechanism (21) in the last section, the shared control input is obtained by blending the optimal control input and human control input:

$$\hat{U}^\curlyvee = \hat{U} + \alpha U_h \tag{27}$$

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

$$\begin{aligned}\delta(X, \hat{W}_c, \hat{U}, U_h) =& \nabla\hat{J}^\top\left(\hat{F} + \hat{G}\hat{U}^\curlyvee + D\right) + r(X, \hat{U}^\curlyvee) \\ =& \hat{W}_c^\top \nabla\varphi_c\left(\hat{F} + \hat{G}(\hat{U} + \alpha U_h) + D\right) \\ &+ X^\top Q X + \Xi(\hat{U} + \alpha U_h)\end{aligned} \tag{28}$$

where δ is the shared control Bellman error. The shared control Bellman error is utilized to train actor-critic NNs to approximate the 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 $\{\hat{U}^\curlyvee(t), \delta(t), \{\hat{U}^{\curlyvee j}(t), \delta^j(t)\}_{j=1}^N\}$ is collected without extrapolation but stored as a stack, where $\{\hat{U}^{\curlyvee j}(t), \delta^j(t)\}$ is the jth historical stored data collection. The weights of actor-critic NNs are learned by minimizing a defined squared loss function $E = \delta^\top\delta + \sum_{k=1}^N \delta^{k\top}\delta^k$, motivated by literature [48]. Accordingly, a concurrent learning-based gradient descent update law is utilized to update the weights of the critic NN:

$$\dot{\hat{W}}_c = -\frac{k_{c1}\delta\sigma}{\left(\sigma^\top\sigma + 1\right)^2} - \frac{k_{c2}}{N}\sum_{k=1}^N \frac{\delta^k\sigma^k}{\left((\sigma^k)^\top\sigma^k + 1\right)^2} \tag{29}$$

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

$$\dot{\hat{W}}_a = \mathrm{Proj}\left(-k_a F_a\left(\hat{W}_a - \hat{W}_c\right)\right) \tag{30}$$

where $k_a > 0$ is the learning rates of actor NN. $F_a \in \mathbb{R}^{n_\varphi \times n_\varphi}$ is positive definite matrices for the updating of actor NN. Proj$(\cdot)$ 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. The detailed algorithm is shown in Algorithm 1.

The detailed architecture of the Koopman operator-based RL algorithm is shown in Fig. 3. The human control input is collected from the human operator and stored in the experience replay stack. The experience replay stack contains the historical data of the human control input, the optimal control input, the state, and the Bellman error. These historical data are utilized to predict the dynamics of the UAVs

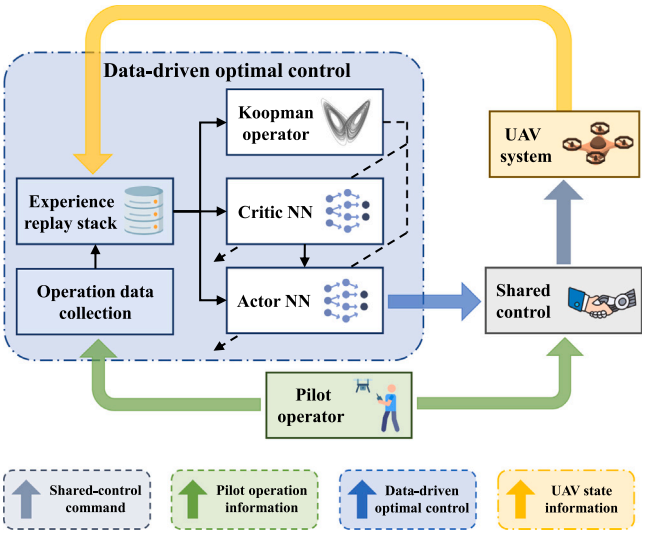


Fig. 3. The proposed data-driven optimal shared control algorithm structure.

Algorithm 1 Data-driven Optimal Shared Control of UAVs

```
Initialize actor-critic weights Ŵ_c, Ŵ_a, learning rates k_ci (i = 1,2), k_a, and projection matrices F.
Initialize experience replay stack {U^, δ, {U^j, δ^j}_{j=1}^N} and Koopman data set {Y_K, X_K, U_K}.
while t < T_end do
    Collect human control input U_h and system state X.
    if UAV model is unknown then
        Compute transfer function Θ([X^⊤, U_h^⊤]^⊤).
        Update Koopman data set with Θ.
        Estimate dynamics matrices A, B using (12).
        Calculate F̂ and Ĝ using (13).
    end if
    Estimate optimal control Û(X) using (26).
    Compute and apply shared control input U^ using (27).
    Calculate Bellman error δ(X, Ŵ_c, Û, U_h) using (28).
    Update experience replay stack with U^ and δ.
    Update actor-critic weights Ŵ_c and Ŵ_a using (29) and (30).
end while
```

and train the actor-critic NNs. The online trained actor-critic NNs could approximate the optimal value functions and the control inputs. Then the shared control input is obtained by the shared control mechanism (21), which blends the optimal control input and the human control input. The shared control input is applied to the UAVs to achieve optimal shared control. Next, the stability analysis of the closed-loop system is presented.

4.3. 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 estimated errors are proved to be ultimate uniform bounded (UUB) under the proposed data-driven optimal shared control scheme. First, three assumptions are given here for the proof.

Assumption 2. The following assumptions are given for the stability analysis:

1. On a tight set $X \in \chi \in \mathbb{R}^n$, both $F(X)$ and $G(X)$ are Lipschitz continuous with $F(0) = 0$, and $G(X)$ satisfied bounded condition $\|G(X)\| \le G_H$ for all $X \in \chi$.
2. Cost matrix Q and R are bounded, such that $\underline{\lambda}_Q \le \|Q\| \le \bar{\lambda}_Q$, $\underline{\lambda}_R \le \|R\| \le \bar{\lambda}_R$, where constants $\underline{\lambda}_Q, \underline{\lambda}_R \ge 0$ and $\bar{\lambda}_q, \bar{\lambda}_R > 0$.

Assumption 3. Assuming that the following parameters and operators are bounded: $\|\hat{W}_c\| \le W_{Hi}$, $\|\sigma(X)\| \le \sigma_{Hi}$, $\|\nabla\sigma(X)\| \le \sigma_{D,Hi}$, $\|\varphi(X)\| \le \varphi_{Hi}$, $\|\nabla\varphi(X)\| \le \varphi_{D,Hi}$, $\|\varepsilon(X)\| \le \varepsilon_{Hi}$, $\|\nabla\varepsilon(X)\| \le \varepsilon_{D,Hi}$,

Assumption 4 (*Persistent Excitation Condition [49,50]*)**.** Assuming that the online collected and extrapolated data set for the weights update law satisfies the following excitation condition:

$$\vartheta_1 I_{\mathcal{L}} \leqslant \int_t^{t+T} \left(\sigma(\tau)\sigma(\tau)^\top/\rho(\tau)\right) \mathrm{d}\tau,$$

$$\vartheta_2 I_{\mathcal{L}} \leqslant \inf_{t\in\mathbf{R}_{t\ge t_0}} \left(\sum_{k=1}^{N} \sigma^k(t)\sigma^k(t)^\top/\rho^k(t)\right)/N,$$

$$\vartheta_3 I_{\mathcal{L}} \leqslant \int_t^{t+T} \left(\frac{1}{N}\sum_{k=1}^{N} \sigma^k(\tau)\sigma^k(\tau)^\top/\rho^k(\tau)\right) \mathrm{d}\tau$$

where $\rho = \left(\sigma^\top\sigma + 1\right)^2$, $\rho^k = \left(\sigma^{k\top}\sigma^k + 1\right)^2$, and at least one of the non-negative constants ϑ_1, ϑ_2, ϑ_3 is positive.

Based on the design of input (26), it could be obtained that:

$$\left\|U^*(X) - \hat{U}(X)\right\|^2 \le \Sigma\tilde{W}_a^\top\tilde{W}_a + \Pi_u \tag{31}$$

where Σ is a upper bound related with φ_H, $\varphi_{D,H}$, σ_H and $\sigma_{D,H}$, Π_u is a upper bound related to $\varepsilon_{D,H}$. The Hamiltonian error δ, or Bellman error, is abbreviated in the following form:

$$\delta = -\sigma^\top\tilde{W}_c + \frac{1}{4}\tilde{W}_a G_\sigma \tilde{W}_a + \Delta(X) + \xi_H, \tag{32}$$

$$\delta^k = -(\sigma^k)^\top\tilde{W}_c + \frac{1}{4}\tilde{W}_a G_\sigma^k \tilde{W}_a + \Delta^k(X), \tag{33}$$

where the $G_\sigma = \nabla\varphi_a^\top G \mathcal{A}^T R^{-1}\mathcal{A} G^\top \nabla\varphi_a^\top$, $G_\sigma^k = G_\sigma(X^K)$, and $\Delta, \Delta^k : \mathbb{R}^n \to \mathbb{R}$ are uniformly bounded on χ, $\overline{\|\Delta\|}$ and $\overline{\|\Delta^k\|}$ decrease as $\overline{\|\nabla\varepsilon\|}$ and $\overline{\|\nabla W\|}$ decrease. The stability analysis of closed-loop system state and network weight estimation errors is given in the following theoretical result.

Theorem 1. *Considering the augmented system dynamics (5) and the proposed data-driven optimal shared control scheme, Assumptions 2, 3 and 4 are satisfied, The actor-critic NNs are updated by the adaptive update law (29) and (30). The control input is estimated by (26). Then the closed-loop system states X and weights errors $[\tilde{W}_c^\top, \tilde{W}_a^\top]^\top$ will be UUB provided that:*

$$\|\mathcal{Z}\| \ge \left(Y_{\mathrm{res}}/\lambda_{\min}(\mathcal{M})\right)^{\frac{1}{2}} \tag{34}$$

where $\mathcal{Z} = \left[X^\top, \tilde{W}_c^\top, \tilde{W}_a^\top\right]^\top$.

Table 1
Parameters of the UAV system and the update law.

Initial parameters	$X_0 = 0.03[\mathbf{1}_3, \mathbf{0}_3]$, $W_{c0} = 0.15(\mathbf{1}_9 + rand(9))$ $W_{a0} = 0.15(\mathbf{1}_9 + rand(9))$, $\mu_{\mathrm{sat}} = 0.5$
UAV parameters	$\gamma_\phi = 0.0211\,\mathrm{kg\ m^2}$, $\gamma_\theta = 0.0219\,\mathrm{kg\ m^2}$ $\gamma_\psi = 0.0366\,\mathrm{kg\ m^2}$, $B_\gamma = \mathrm{diag}\,([41, 41, 110])$
Update parameters	$R = I_2$, $Q = I_6$, $k_{c1} = 2$, $k_{c2} = 1$ $k_a = 1$, $F_a = I_6$

Proof. Based on the Lyapunov stability theory, we construct the following Lyapunov function: $\mathcal{V}(\mathcal{Z}) = J^* + \frac{1}{2}\tilde{W}_c^\top\tilde{W}_c + \frac{1}{2}\tilde{W}_a^\top\tilde{W}_a$. To facilitate the analysis, the ith element of shared control input $\hat{U}^*$ can be rewritten in a compact form: $\hat{U}^*(i) = \hat{U}(i) + \alpha_i U_h(i) \approx \mathcal{A}_i\hat{U}$, where $\mathcal{A}_i \in [1, 2]$ is a coefficient associated with the shared control parameter α_i. Subsequently, the shared control input can be expressed as: $\hat{U}^* \approx \mathcal{A}\hat{U}$ with diagonal matrix $\mathcal{A} = \mathrm{diag}\left(\mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_m\right)$ containing the coefficients $\mathcal{A}_i$. Taking the time derivative of the Lyapunov function $\mathcal{V}$, we obtain:

$$\dot{\mathcal{V}} = \nabla J^*\left(F + G\mathcal{A}U^* + D\right) + \tilde{W}_c^\top\dot{\hat{W}}_c^\top + \tilde{W}_a^\top\dot{\hat{W}}_a^\top \tag{35}$$

Substituting the $(\nabla J^*)^\top F(X)$ term from (32) and (33) into (35), and employing the Bellman errors from (32) and (33), the time derivative of Lyapunov function can be expressed as:

$$\begin{aligned}\dot{\mathcal{V}} =& -X^\top QX - \Xi(\mathcal{A}U^*) + \tilde{W}_a^\top\left(-k_a F_a\left(\tilde{W}_a - \hat{W}_c\right)\right)\\ &- \tilde{W}_c^\top\left(-k_{c1}\frac{\sigma}{\rho}\left(-\sigma^\top\tilde{W}_c + \frac{1}{4}\tilde{W}_a^\top G_\sigma\tilde{W}_a + \Delta\right)\right)\\ &- \tilde{W}_c^\top\left(-\frac{k_{c2}}{N}\sum_{k=1}^{N}\frac{\sigma^k}{\rho^k}\frac{1}{4}\tilde{W}_a^\top G_\sigma^k\tilde{W}_a\right)\\ &- \tilde{W}_c^\top\left(-\frac{k_{c2}}{N}\sum_{k=1}^{N}\frac{\sigma^k}{\rho^k}\left(-(\sigma^k)^\top\tilde{W}_a + \Delta^k\right)\right)\end{aligned} \tag{36}$$

Substitute inequality (31), then employing Young's inequality and Assumptions 2–4, the derivative can be rewritten as:

$$\dot{\mathcal{V}} \le -\mathcal{Z}^\top\mathcal{M}\mathcal{Z} + Y_{\mathrm{res}}$$

where $\mathcal{M} = \left[m_1\ 0\ 0; 0\ m_2\ 0; 0\ m_3\ m_4\right]$ is a positive definite matrix, $m_1 = \underline{\lambda}_Q$, $m_2 = \frac{1}{2}k_{c1}\sigma\sigma^\mathrm{T} + \frac{1}{2}k_{c2}\vartheta_2 I_{\mathcal{L}}$, $m_3 = -F_a I_{\mathcal{L}}$, $m_4 = F_a I_{\mathcal{L}} - \bar{\lambda}_R\mathcal{A}\Sigma I_{\mathcal{L}}$, and Y_{res} is a residual defined as:

$$\begin{aligned}Y_{\mathrm{res}} =& \frac{1}{2}k_{c1}\left(\frac{1}{4}\tilde{W}_a^\top G_\sigma\tilde{W}_a + \xi_H + \Delta\right)^2 + \bar{\lambda}_R\Pi_u\\ &+ \frac{1}{2}k_{c2}\left(\frac{1}{4}\tilde{W}_a^\top G_{\sigma,k}\tilde{W}_a + \Delta^k\right)^2\end{aligned}$$

Therefore, with the selection of an appropriate positive definite matrix $\mathcal{M}$, which could be satisfied with proper selection of k_{ci}, k_a, F_a, and the initial weights of the actor-critic NNs, the closed-loop system state X and the neural network weight estimation errors $\left[\tilde{W}_c^\top, \tilde{W}_a^\top\right]^\top$ are guaranteed to be ultimately uniformly bounded when condition (34) is satisfied. This completes the stability proof. □

5. Simulation verification

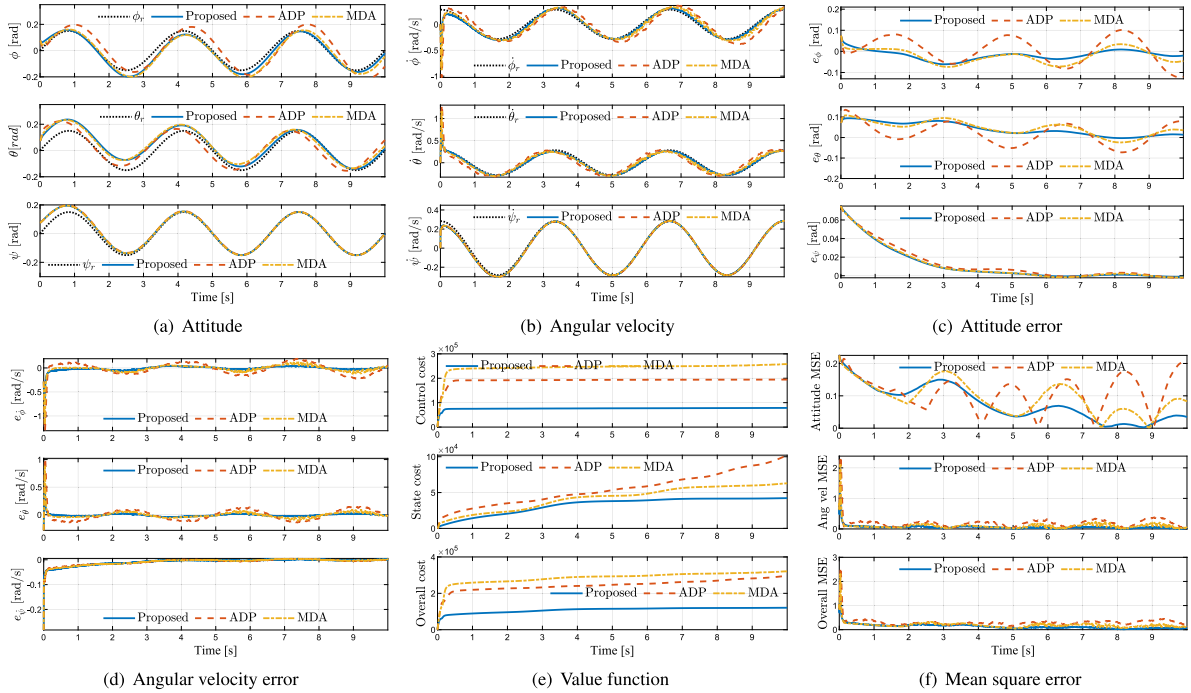
5.1. Example 1: numerical simulations

Experimental setup

The system dynamics of the UAV is set by dynamics (5). To compare and evaluate the performance of proposed method and methods from [6,30], a numerical simulation of UAV shared control is conducted. To simplify the comparison, the human control input is constructed as a PD controller as $U_h = -K_p X - K_d\dot{X}$, in which $K_p = 3$, $K_d = 0.5$ are the proportional and derivative gains inspired by [22,51]. The simulator is implemented in MATLAB R23b Simulink on a PC with an Intel Core i3-12100F CPU (@3.3 GHz) and 24 GB RAM. The solver

Table 2
Controller performance comparison results.

Method	ϕ RMSE	θ RMSE	ψ RMSE	$\dot{\phi}$ RMSE	$\dot{\theta}$ RMSE	$\dot{\psi}$ RMSE	Att RMSE	Ang RMSE	All RMSE
Proposed	**26.6856%** ↓	**48.5170%** ↓	**19.0512%** ↓	**24.5528%** ↓	**15.0650%** ↓	**7.0001%**	**31.5294%** ↓	**15.5393%** ↓	**19.0512%** ↓
ADP [30]	42.8268%	43.9461%	43.7003%	69.9931%	54.8944%	6.9509% ↓	56.7201%	51.0785%	20.6817%
MDA [6]	36.7381%	26.3966%	28.6679%	41.8387%	30.2750%	7.0761%	36.9340%	53.6926%	19.5877%
Method	T_1 state cost	T_2 state cost	state cost	T_1 ctrl cost	T_2 ctrl cost	Ctrl cost	T_1 cost	T_2 cost	All cost
Proposed	**1.4209**$\times10^4$ ↓	**2.8020**$\times10^4$ ↓	**4.2229**$\times10^4$ ↓	**7.4963**$\times10^4$ ↓	**3.5239**$\times10^3$	**7.8487**$\times10^4$ ↓	**8.9172**$\times10^4$ ↓	**3.1544**$\times10^4$ ↓	**1.2072**$\times10^5$ ↓
ADP [30]	2.7739$\times10^4$	7.4434$\times10^4$	1.0217$\times10^5$	1.9162$\times10^5$	2.9818$\times10^3$ ↓	1.9460$\times10^5$	2.1936$\times10^5$	7.7416$\times10^4$	2.9678$\times10^5$
MDA [6]	2.5872$\times10^5$	4.4234$\times10^4$	6.2952$\times10^4$	2.4000$\times10^5$	1.7915$\times10^4$	2.5791$\times10^5$	2.5872$\times10^5$	6.2149$\times10^4$	3.2087$\times10^5$



(a) Attitude (b) Angular velocity (c) Attitude error

(d) Angular velocity error (e) Value function (f) Mean square error

Fig. 4. Simulation results of the UAV system and the controller performance comparison.

of the ODEs is the Fourth-order Runge–Kutta method with a fixed step size of $T = 0.001$ s. The simulation time is set as $t_{\text{end}} = 10$ s. The basis functions of the critic NNs are designed as:

$$\varphi_c = \varphi_a = \left[X(1)^2, X(1)X(4), X(4)^2, X(2)^2, X(2)X(5), X(5)^2, X(3)^2, X(3)X(6), X(6)^2\right]$$

The parameters of the initial condition, UAV system, and update law are shown in Table 1. A sinusoidal desired trajectory is set as: $X_d = \left[A_\phi \sin(\omega t), A_\theta \cos(\omega t), A_\psi \sin(\omega t)\right]^\top$, where $A_\phi = A_\theta = A_\psi = 0.1$, $\omega = 0.5$. To evaluate the performance of proposed method, three methods are compared in simulations:

- **Proposed**: Proposed data-driven optimal shared control.
- **ADP**: Adaptive dynamic programming method in [30].
- **MDA**: Model-based shared control method in [6].

According to the proposed method, the dynamics model of the UAV system is predicted and linearized as A and B by (12). The predicted linear system dynamics matrix A and B are computed. With predicted linear system dynamics matrix A and B, the estimated drift dynamics matrix $\hat{F}$ and control input matrix $\hat{G}$ could be computed by (13). The shared control input is calculated by (27) and applied to the UAV system.

Simulation results

The simulation results are shown in Figs. 4–6. The state of the UAV is shown in Figs. 4(a)–4(b), which indicates that the proposed

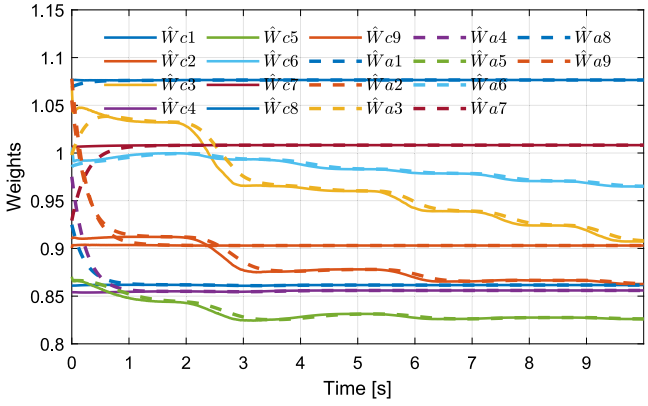


Fig. 5. Simulation results of actor-critic NNs weights.

method tracks desired trajectory precisely and responds quickly. The attitude tracking error and angular velocity tracking error are shown in Figs. 4(c)–4(d), in which the proposed method achieves smaller tracking errors and better tracking performance. The cost comparison results are shown in Fig. 4(e), where the proposed method achieves the lowest cost of control input, state, and overall cost. In Fig. 4(f), the mean squared error (MSE) of the tracking control process is shown, which indicates that the proposed method has a smaller MSE and better

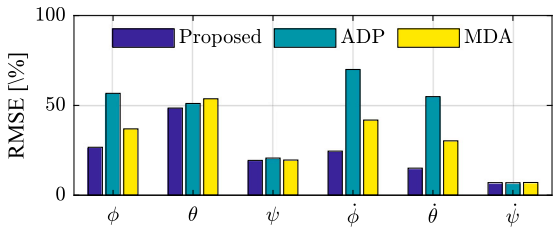


Fig. 6. RMSE of the UAV system states.

tracking performance. The weights of the actor-critic NNs are shown in Fig. 5, the weights are updated online and converge to the stable values. The performance comparison results are shown in Table 2, which compares the cost and root mean square error (RMSE) of the attitude and angular velocity. The results indicate that the proposed method achieves the best performance in terms of cost and RMSE. The detailed RMSE of the UAV system is shown in Fig. 6. It shows that the proposed method has the smallest RMSE and best tracking performance.

5.2. Example 2: Human-in-the-loop simulations

Experimental setup

In this subsection, human-in-the-loop UAV simulations are conducted to further verify the effectiveness of the proposed data-driven optimal shared control scheme. To give more straightforward feedback on the UAV system's performance, we choose to control the UAV to fly through two circles in the simulation. The dynamics of the UAV system is simplified and transferred as a position control system based on the original dynamics system (5). Assume that the attitude angles are small enough that $\sin\phi \approx \phi$, $\cos\phi \approx 1$, $\sin\theta \approx \theta$, $\cos\theta \approx 1$. The simplified dynamics system is given by:

$$\begin{cases} \dot{p}_{\mathrm{i}} = v_{\mathrm{i}}, \ i \in \{x, y, z\} \\ \dot{v}_{\mathrm{x}} = -g(\phi_d \sin\psi + \theta_d \cos\psi), \\ \dot{v}_{\mathrm{y}} = -g(-\phi_d \cos\psi + \theta_d \sin\psi), \\ \dot{v}_{\mathrm{z}} = g - f/m, \end{cases} \tag{37}$$

where p_{i} is the position of the UAV in the i-axis, v_{i} is the velocity of the UAV in the i-axis, $p = [p_x, p_y, p_z]^\top$, $v = [v_x, v_y, v_z]^\top$. $\boldsymbol{\Theta} = [\phi, \theta]^\top$ is the desired attitude of the UAV, take $U_h = \boldsymbol{\Theta}$ as the input signals of a human operator. Consider the desired trajectory $p_d = [p_{x_d}, p_{y_d}, p_{z_d}]^\top$ generated from $\dot{p}_d = [v_{x_d}, v_{y_d}, v_{z_d}]^\top$. The state of UAV tracking dynamics for example 2 is defined as $X = [p - p_d, v - v_d]^\top$. For the cooperation of the human operator and the UAV, we choose the Rflysim simulation environment [52] for the human-in-the-loop (HiTL) simulation. The experiment setup is shown in Fig. 7, in which the human operator observes the UAV's attitude and position on the screen, and controls the UAV by two joysticks of Logitech F310 gamepad. The setup of the experiment is shown in Fig. 7(a), where the human operator observes the UAV's state on the screen through the interface of the simulation environment. The left joystick is used to control the roll angle of the UAV as shown in Fig. 7(b), and the right joystick is used to control the pitch and yaw angles of the UAV as shown in Fig. 7(c). Joysticks have a resolution of $1/256$ and a value range of $[-0.5, 0.5]$. The detailed scheme of example 2 is shown in Fig. 8, in which the controller is calculated by Eq. (26), and the shared control input is calculated by Eq. (27). It should be noted that although this example is conducted in simulation, the proposed method can be easily extended to the real-world scenario by rapid prototyping validation capabilities of Rflysim [52].

In the HiTL simulation, two circles are set as the targets for the UAV to fly through. The center of the circles are [250 m, 75 m, 100 m] and [500 m, −75 m, 100 m] with a radius of 10 m, UAV is set to fly through the two circles in sequential order $T_{c1} = 30$ s, $T_{c2} = 60$ s. Initial weights of the actor-critic NNs are set as $W_{c0} = W_{a0} = 2 \times \mathbf{1}_4$. The state penalty matrix is set as $Q = \mathrm{diag}([10000, 10000, 10000, 0.1, 0.1, 0.1])$, the control penalty matrix is set as $R = 10000 \times \mathbf{1}_3$. The history stack size is set as $N = 30$. The learning rate is set as $\alpha = 0.001$. The simulator is implemented in MATLAB R23b Simulink and Rflysim on the same PC as in example 1. The simulation time is set as $t_{\mathrm{end}} = 60$ s. The solver of the ODEs is the Fourth-order Runge–Kutta method with a fixed step size of $T = 0.001$ s. The basis of NNs is selected as

$$\varphi_c = \varphi_a = \left[X(1)X(4), X(2)X(5), X(1)^3X(4), X(2)^3X(5)\right] \tag{38}$$

To illustrate the performance of the proposed method in the HiTL simulation, two methods are compared in the simulation:

- **Proposed**: Proposed data-driven optimal shared control.
- **MDA**: Model-based shared control method in [6,53].
- **Human operator**: Human operator direct control.

Note that the human operator in the 'Proposed' and 'MDA' methods is the same person. The intention and interaction of the human operator are the same in the simulation.

Simulation results

The simulation results are shown in Figs. 9–14. The trajectories of the UAV in the HiTL simulation are shown in Fig. 9, where three methods are compared, all three methods can make the UAV fly through the two circles, but the proposed method achieves the shortest tracking trajectory. The tracking performance of the UAV is shown in Fig. 10, in which the proposed method achieves the smallest tracking error and best tracking performance. The control inputs of the human operator and the shared control are shown in Fig. 11, which shows that the shared control input is blended by the human operator's input and the optimal control input. The shared control input is not only much smoother than the human operator's input, but also blends the human operator's intention and the automation's intention at the same time effectively. The detailed state of the UAV is shown in Fig. 12, where the attitude and angular velocity of the UAV are shown. The cost comparison results are shown in Fig. 13, where the proposed method achieves the lowest cost of control inputs and overall cost. The weights of the actor-critic NNs are shown in Fig. 14, which shows that the weights are updated online and converge to stable values.

To evaluate the performance of the proposed method in the HiTL simulation, four quantitative metrics are introduced to evaluate the performance of different approaches as follows:

1. Position smoothness index (PSI)
$$\mathrm{PSI} = \int_T \left\| d^3 p / d\tau^3 \right\|^2 d\tau \tag{39}$$
which is mean-square third-order derivative of position.
2. Attitude smoothness index (ASI)
$$\mathrm{ASI} = \int_T \left\| d^3 \Theta / d\tau^3 \right\|^2 d\tau \tag{40}$$
which is mean-square third-order derivative of attitude.
3. Accumulated tracking error (ATE)
$$\mathrm{ATE} = \int_T \left\| p - p_d \right\|^2 d\tau \tag{41}$$
where p is the position of the UAV and p_d is the desired position of the UAV.
4. Accumulated Control energy
$$\mathrm{ACE} = \int_T \left\| U \right\|^2 d\tau \tag{42}$$
where U is the control input imposed on the UAV system.

The performance comparison results are shown in Table 3. The results indicate that the proposed method achieves the best performance in terms of PSI, ATE, and CE. Note that the ASI of the proposed method is not the best, which is because the human operator's input is

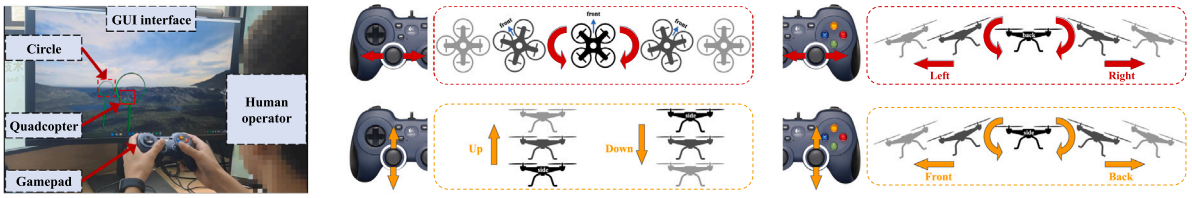


Fig. 7. (a) Scenario of experiment. (b) Left joystick's input functions. (c) Right joystick's input functions.

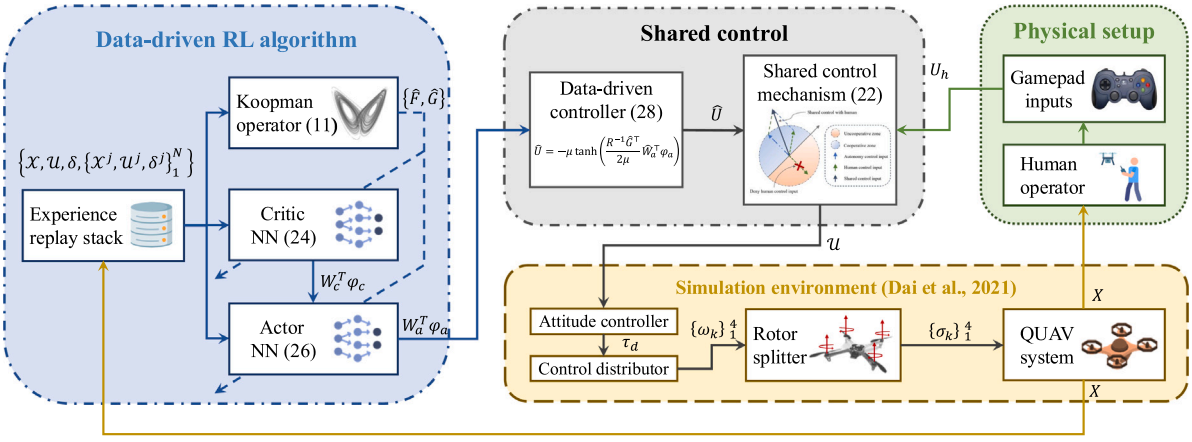
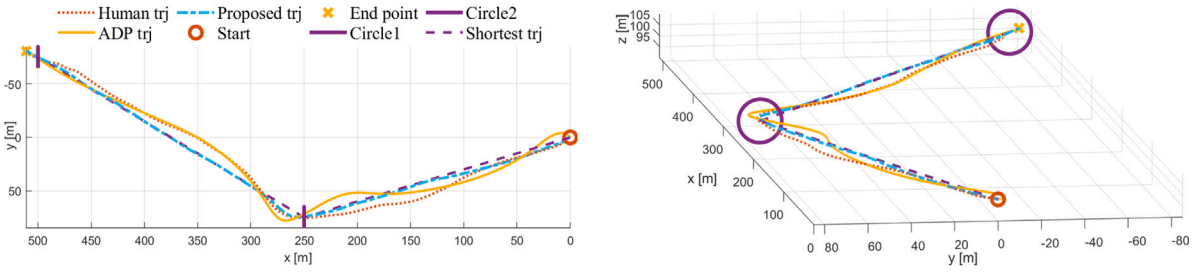


Fig. 8. Scheme of the HiTL simulation example 2.



(a) Overlook of the trajectories. (b) Human operator's perspective of the trajectories.

Fig. 9. Trajectories of the UAV system in the HiTL simulation.

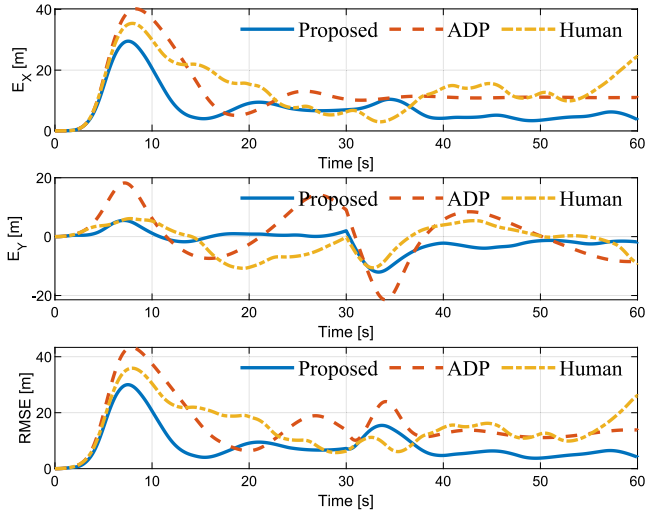


Fig. 10. Tracking error of the UAV.

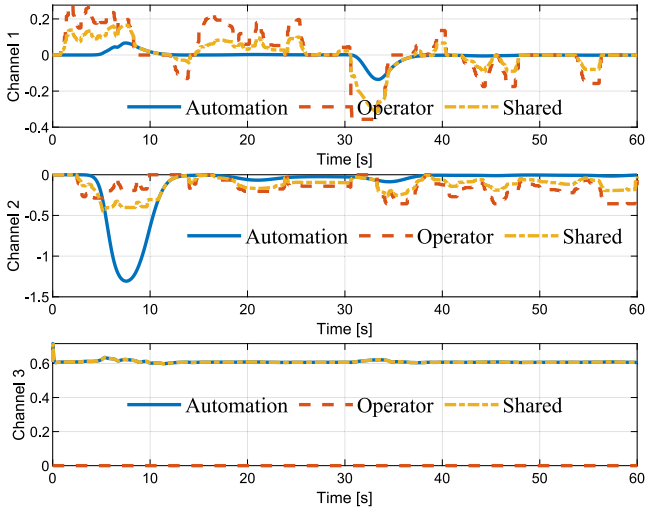


Fig. 11. Input of human operator and shared control.

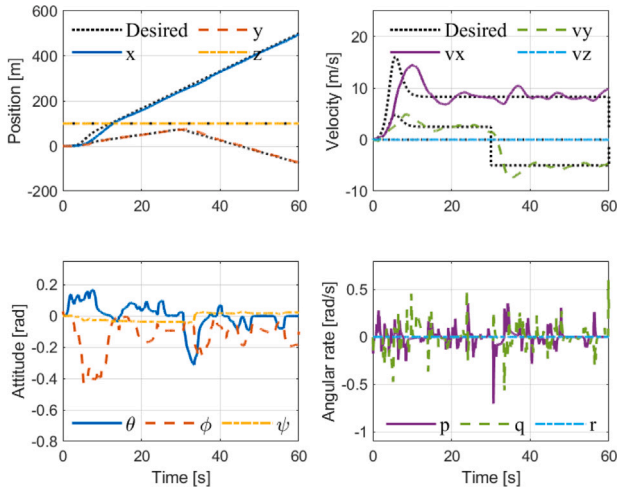


Fig. 12. State of the UAV.

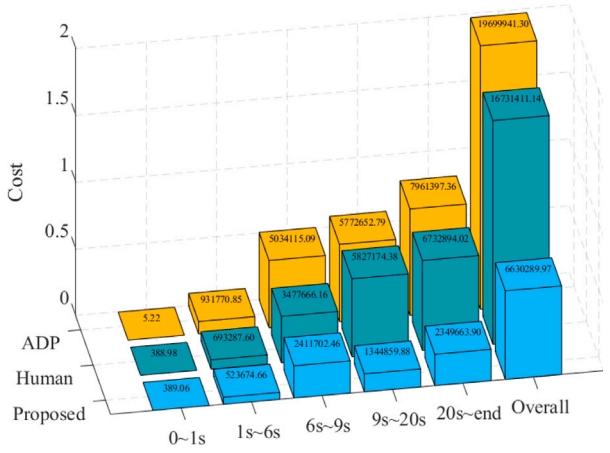


Fig. 13. Cost comparison results of the HiTL simulation.

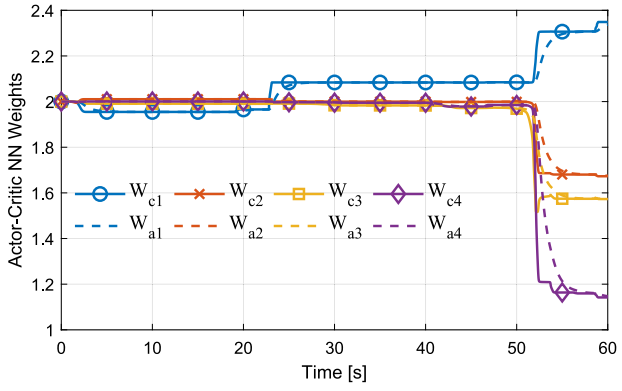


Fig. 14. Weights of the actor-critic NNs.

Table 3
Performance comparison results of the HiTL simulation.

Method	PSI	ASI	ATE	ACE
Proposed	**57.7396** ↓	**39.1561**	**514.7209** ↓	**40.7349** ↓
ADP only	66.9467	5.5052 ↓	945.9328	46.2972
Human only	98.5432	127.5812	918.9737	47.0130

very vibrating and the shared control input is blended by the human operator's input and the optimal control input. However, compared with the case of 'human only', ASI of the proposed method is much better, which indicates that the shared control mechanism can smooth the human operator's input effectively.

Table 4
Proportion of cooperation in the HiTL simulation.

Index	Roll Control	Pitch Control	All Control
Cooperative time (%)	54.04	80.38	90.05
Cost of Human Control	7.52×10^3	1.50×10^4	2.25×10^4
Cost of Shared Control	2.89×10^3	1.16×10^4	1.45×10^4
Control Cost Saving (%)	61.52	22.68	35.65

The proportion of cooperation in the HiTL simulation is shown in Table 4, which shows that the proposed method can save the cost of human control effectively. The simulation results show that the proposed data-driven optimal shared control scheme can achieve better tracking performance and HiTL control performance for the UAV system.

6. Conclusion

In this paper, a data-driven optimal shared control scheme is proposed for UAV control. The Koopman-based RL algorithm is designed to approximate the optimal control input without the knowledge of the system dynamics. A novel shared control mechanism is designed to blend the human control input and the optimal control input, in which the intention of the human operator is judged and integrated into the shared control input smoothly and effectively. The value function and optimal control input are approximated online utilizing actor-critic NNs, which exploits and integrates human operator experience from collected maneuvers data. The stability analysis of the closed-loop system is presented based on the Lyapunov stability theory. Numerical and HiTL simulation results demonstrate that the proposed method achieves more effective and smooth tracking performance for UAV control. The future work will concentrate on the real-world practical application of the proposed method, such as the implementation of the proposed method on a real pilot-UAV system.

CRediT authorship contribution statement

Junkai Tan: Writing – original draft, Methodology, Conceptualization. **Shuangsi Xue:** Resources, Methodology. **Zihang Guo:** Visualization, Methodology. **Huan Li:** Investigation, Conceptualization. **Hui Cao:** Supervision, Project administration. **Badong Chen:** Resources, Project administration.

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.

Acknowledgment

This research is supported by the National Natural Science Foundation of China under Grant No. 62436005 and 62311540022, and China Postdoctoral Science Foundation China (Project ID: 2024M762602).

Data availability

The authors do not have permission to share data.

References

[1] A. Perrusquía, W. Guo, Closed-loop output error approaches for drone's physics informed trajectory inference, IEEE Trans. Autom. Control 68 (12) (2023) 7824–7831, http://dx.doi.org/10.1109/TAC.2023.3247461.

[2] 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 Trans. Robot. 36 (2) (2020) 414–430, http://dx.doi.org/10.1109/TRO.2019.2955321.

[3] J.A. Rosenfeld, B.P. Russo, R. Kamalapurkar, T.T. Johnson, The occupation kernel method for nonlinear system identification, SIAM J. Control Optim. 62 (3) (2024) 1643–1668, http://dx.doi.org/10.1137/19M127029X.

[4] J.A. Rosenfeld, R. Kamalapurkar, L.F. Gruss, T.T. Johnson, Dynamic mode decomposition for continuous time systems with the Liouville operator, J. Nonlinear Sci. 32 (1) (2022) 5, http://dx.doi.org/10.1007/s00332-021-09746-w.

[5] M.O. Williams, I.G. Kevrekidis, C.W. Rowley, A data–driven approximation of the koopman operator: Extending dynamic mode decomposition, J. Nonlinear Sci. 25 (6) (2015) 1307–1346, http://dx.doi.org/10.1007/s00332-015-9258-5.

[6] A. Broad, I. Abraham, T. Murphey, B. Argall, Data-driven koopman operators for model-based shared control of human–machine systems, Int. J. Robot. Res. 39 (9) (2020) 1178–1195, http://dx.doi.org/10.1177/0278364920921935.

[7] M.F. Pairan, S.S. Shamsudin, M.F. Zulkafli, Neural network based system identification for quadcopter dynamic modelling: A review, J. Adv. Mech. Eng. Appl. 1 (2) (2020) 20–33.

[8] I. Sa, P. Corke, System identification, estimation and control for a cost effective open-source quadcopter, in: 2012 IEEE International Conference on Robotics and Automation, 2012, pp. 2202–2209, http://dx.doi.org/10.1109/ICRA.2012.6224896.

[9] B.S. Guevara, L.F. Recalde, J. Varela-Aldás, D.C. Gandolfo, J.M. Toibero, Quadcopters control using online dynamic mode decomposition, IFAC-PapersOnLine 56 (3) (2023) 589–594, http://dx.doi.org/10.1016/j.ifacol.2023.12.088.

[10] L. Chen, J. Zhang, Q. Quan, Quadcopter attitude control with vibration reduction by additive-state-decomposition dynamic inversion design with a notch filter, Nonlinear Dynam. 111 (9) (2023) 8313–8327, http://dx.doi.org/10.1007/s11071-023-08272-6.

[11] W. Guo, S. Zhao, H. Cao, B. Yi, X. Song, Koopman operator-based driver-vehicle dynamic model for shared control systems, Appl. Math. Model. 114 (2023) 423–446, http://dx.doi.org/10.1016/j.apm.2022.10.014.

[12] V.S. Donge, B. Lian, F.L. Lewis, A. Davoudi, Data-efficient reinforcement learning for complex nonlinear systems, IEEE Trans. Cybern. (2023) 1–12, http://dx.doi.org/10.1109/TCYB.2023.3324601.

[13] W. Jin, D. Kulić, J.F.-S. Lin, S. Mou, S. Hirche, Inverse optimal control for multiphase cost functions, IEEE Trans. Robot. 35 (6) (2019) 1387–1398, http://dx.doi.org/10.1109/TRO.2019.2926388.

[14] X. Xing, W. Li, S. Yuan, Y. Li, Fuzzy logic-based arbitration for shared control in continuous human–robot collaboration, IEEE Trans. Fuzzy Syst. 32 (7) (2024) 3979–3991, http://dx.doi.org/10.1109/TFUZZ.2024.3386822.

[15] Q. Zhang, Y. Kang, Y.-B. Zhao, P. Li, S. You, Traded control of human–machine systems for sequential decision-making based on reinforcement learning, IEEE Trans. Artif. Intell. 3 (4) (2022) 553–566, http://dx.doi.org/10.1109/TAI.2021.3127857.

[16] H. Zhang, Y. Zhang, X. Zhao, Event-triggered adaptive dynamic programming for hierarchical sliding-mode surface-based optimal control of switched nonlinear systems, IEEE Trans. Autom. Sci. Eng. 21 (3) (2024) 4851–4863, http://dx.doi.org/10.1109/TASE.2023.3303359.

[17] T. An, X. Zhu, B. Ma, H. Jiang, B. Dong, Hierarchical approximate optimal interaction control of human-centered modular robot manipulator systems: A stackelberg differential game-based approach, Neurocomputing 585 (2024) 127573, http://dx.doi.org/10.1016/j.neucom.2024.127573.

[18] T. An, X. Zhu, M. Zhu, B. Ma, B. Dong, Fuzzy logic nonzero-sum game-based distributed approximated optimal control of modular robot manipulators with human-robot collaboration, Neurocomputing 543 (2023) 126276, http://dx.doi.org/10.1016/j.neucom.2023.126276.

[19] E. Eraslan, Y. Yildiz, A.M. Annaswamy, Shared control between pilots and autopilots: An illustration of a cyberphysical human system, IEEE Control Syst. 40 (6) (2020) 77–97, http://dx.doi.org/10.1109/MCS.2020.3019721.

[20] X. Liu, S.S. Ge, F. Zhao, X. Mei, Optimized interaction control for robot manipulator interacting with flexible environment, IEEE/ASME Trans. Mechatronics 26 (6) (2021) 2888–2898, http://dx.doi.org/10.1109/TMECH.2020.3047919.

[21] B. Barros Carlos, A. Franchi, G. Oriolo, Towards safe human-quadrotor interaction: Mixed-initiative control via real-time NMPC, IEEE Robot. Autom. Lett. 6 (4) (2021) 7611–7618, http://dx.doi.org/10.1109/LRA.2021.3096502.

[22] 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 Trans. Cybern. (2024) 1–11, http://dx.doi.org/10.1109/TCYB.2024.3402353.

[23] M. Li, J. Qin, Z. Wang, Q. Liu, Y. Shi, Y. Wang, Optimal motion planning under Dynamic Risk Region for safe human–robot cooperation, IEEE/ASME Trans. Mechatronics (2024) 1–11, http://dx.doi.org/10.1109/TMECH.2024.3408810.

[24] Y. Sun, J. Hu, Z. Peng, B.K. Ghosh, Hierarchical critic learning optimal control for lower limb exoskeleton robots with prescribed constraints, Internat. J. Robust Nonlinear Control 34 (3) (2024) 2162–2183, http://dx.doi.org/10.1002/rnc.7075.

[25] W. Zhu, M. Hayashibe, A hierarchical deep reinforcement learning framework with high efficiency and generalization for fast and safe navigation, IEEE Trans. Ind. Electron. 70 (5) (2023) 4962–4971, http://dx.doi.org/10.1109/TIE.2022.3190850.

[26] R. Chai, H. Niu, J. Carrasco, F. Arvin, H. Yin, B. Lennox, Design and experimental validation of deep reinforcement learning-based fast trajectory planning and control for mobile robot in unknown environment, IEEE Trans. Neural Netw. Learn. Syst. 35 (4) (2024) 5778–5792, http://dx.doi.org/10.1109/TNNLS.2022.3209154.

[27] J. Li, Z. Li, X. Li, Y. Feng, Y. Hu, B. Xu, Skill learning strategy based on dynamic motion primitives for human–robot cooperative manipulation, IEEE Trans. Cogn. Dev. Syst. 13 (1) (2021) 105–117, http://dx.doi.org/10.1109/TCDS.2020.3021762.

[28] Z. Jin, A. Liu, W.-A. Zhang, L. Yu, C.-Y. Su, A learning based hierarchical control framework for human–robot collaboration, IEEE Trans. Autom. Sci. Eng. 20 (1) (2023) 506–517, http://dx.doi.org/10.1109/TASE.2022.3161993.

[29] A. Perrusquía, W. Yu, Identification and optimal control of nonlinear systems using recurrent neural networks and reinforcement learning: An overview, Neurocomputing 438 (2021) 145–154, http://dx.doi.org/10.1016/j.neucom.2021.01.096.

[30] Q. Ma, P. Jin, F.L. Lewis, Guaranteed cost attitude tracking control for uncertain quadrotor unmanned aerial vehicle under safety constraints, IEEE/CAA J. Autom. Sin. 11 (6) (2024) 1447–1457, http://dx.doi.org/10.1109/JAS.2024.124317.

[31] D. Wang, N. Gao, D. Liu, J. Li, F.L. Lewis, Recent progress in reinforcement learning and adaptive dynamic programming for advanced control applications, IEEE/CAA J. Autom. Sin. 11 (1) (2024) 18–36, http://dx.doi.org/10.1109/JAS.2023.123843.

[32] H. Wen, W. Fu, W. Chen, J. Huan, C. Li, X. Duan, Imitation learning and teleoperation shared control with unit tangent fuzzy movement primitives, IEEE Trans. Fuzzy Syst. (2024) 1–15, http://dx.doi.org/10.1109/TFUZZ.2024.3443713.

[33] B. Lian, W. Xue, F.L. Lewis, Heterogeneous multi-player imitation learning, Control. Theory Technol. 21 (3) (2023) 281–291, http://dx.doi.org/10.1007/s11768-023-00171-w.

[34] J. Town, Z. Morrison, R. Kamalapurkar, Pilot performance modeling via observer-based inverse reinforcement learning, IEEE Trans. Control Syst. Technol. (2024) 1–8, http://dx.doi.org/10.1109/TCST.2024.3410128.

[35] A. Perrusquía, W. Guo, Drone's objective inference using policy error inverse reinforcement learning, IEEE Trans. Neural Netw. Learn. Syst. (2024) 1–12, http://dx.doi.org/10.1109/TNNLS.2023.3333551.

[36] R. Self, M. Abudia, S.N. Mahmud, R. Kamalapurkar, Model-based inverse reinforcement learning for deterministic systems, Automatica 140 (2022) 110242, http://dx.doi.org/10.1016/j.automatica.2022.110242.

[37] W. Jin, T.D. Murphey, D. Kulić, N. Ezer, S. Mou, Learning from sparse demonstrations, IEEE Trans. Robot. 39 (1) (2023) 645–664, http://dx.doi.org/10.1109/TRO.2022.3191592.

[38] B. Lian, Y. Kartal, F.L. Lewis, D.G. Mikulski, G.R. Hudas, Y. Wan, A. Davoudi, Anomaly detection and correction of optimizing autonomous systems with inverse reinforcement learning, IEEE Trans. Cybern. 53 (7) (2023) 4555–4566, http://dx.doi.org/10.1109/TCYB.2022.3213526.

[39] W. Xue, B. Lian, Y. Kartal, J. Fan, T. Chai, F.L. Lewis, Model-free inverse H-infinity control for imitation learning, IEEE Trans. Autom. Sci. Eng. (2024) 1–12, http://dx.doi.org/10.1109/TASE.2024.3427657.

[40] Y. Li, K.P. Tee, R. Yan, W.L. Chan, Y. Wu, A framework of human–robot coordination based on game theory and policy iteration, IEEE Trans. Robot. 32 (6) (2016) 1408–1418, http://dx.doi.org/10.1109/TRO.2016.2597322.

[41] H. Yang, Q. Hu, H. Dong, X. Zhao, D. Li, Optimized data-driven prescribed performance attitude control for actuator saturated spacecraft, IEEE/ASME Trans. Mechatronics 28 (3) (2023) 1616–1626, http://dx.doi.org/10.1109/TMECH.2022.3230993.

[42] D. Wang, J. Wang, L. Hu, L. Zhang, Novel parallel formulation for iterative reinforcement learning control, IEEE Trans. Syst., Man, Cybern. Syst. 54 (10) (2024) 6320–6331, http://dx.doi.org/10.1109/TSMC.2024.3428482.

[43] H. Modares, I. Ranatunga, F.L. Lewis, D.O. Popa, Optimized assistive human–robot interaction using reinforcement learning, IEEE Trans. Cybern. 46 (3) (2016) 655–667, http://dx.doi.org/10.1109/TCYB.2015.2412554.

[44] H. Hu, D. Isele, S. Bae, J.F. Fisac, Active uncertainty reduction for safe and efficient interaction planning: A shielding-aware dual control approach, Int. J. Robot. Res. (2023) 02783649231215371, http://dx.doi.org/10.1177/02783649231215371.

[45] G. Wen, W. Hao, W. Feng, K. Gao, Optimized backstepping tracking control using reinforcement learning for quadrotor unmanned aerial vehicle system, IEEE Trans. Syst., Man, Cybernetics: Syst. 52 (8) (2022) 5004–5015, http://dx.doi.org/10.1109/TSMC.2021.3112688.
[46] L. Chen, F. Hao, Optimal tracking control for unknown nonlinear systems with uncertain input saturation: A dynamic event-triggered ADP algorithm, Neurocomputing 564 (2024) 126964, http://dx.doi.org/10.1016/j.neucom.2023.126964.
[47] E. Tzorakoleftherakis, T.D. Murphey, Controllers as filters: Noise-driven swing-up control based on maxwell's demon, in: 2015 54th IEEE Conference on Decision and Control, CDC, 2015, pp. 4368–4374, http://dx.doi.org/10.1109/CDC.2015.7402901.
[48] D. Wang, M. Zhao, M. Ha, J. Qiao, Stability and admissibility analysis for zero-sum games under general value iteration formulation, IEEE Trans. Neural Netw. Learn. Syst. 34 (11) (2023) 8707–8718, http://dx.doi.org/10.1109/TNNLS.2022.3152268.
[49] R. Ji, J. Ma, D. Li, S.S. Ge, Finite-time adaptive output feedback control for MIMO nonlinear systems with actuator faults and saturations, IEEE Trans. Fuzzy Syst. 29 (8) (2021) 2256–2270, http://dx.doi.org/10.1109/TFUZZ.2020.2996709.
[50] X. You, S. Dian, R. Guo, S. Li, Exponential stability analysis for discrete-time quaternion-valued neural networks with leakage delay and discrete time-varying delays, Neurocomputing 430 (2021) 71–81, http://dx.doi.org/10.1016/j.neucom.2020.12.021.
[51] B.R. Gaines, Linear and nonlinear models of the human controller, Int. J. Man-Mach. Stud. 1 (4) (1969) 333–360, http://dx.doi.org/10.1016/S0020-7373(69)80001-5.
[52] X. Dai, C. Ke, Q. Quan, K.-Y. Cai, RFlySim: Automatic test platform for UAV autopilot systems with FPGA-based hardware-in-the-loop simulations, Aerosp. Sci. Technol. 114 (2021) 106727, http://dx.doi.org/10.1016/j.ast.2021.106727.
[53] J. Tan, S. Xue, H. Cao, S.S. Ge, Human-AI interactive optimized shared control, J. Autom. Intell. (2025) http://dx.doi.org/10.1016/j.jai.2025.01.001.

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.

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.

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.

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

Her current research interest includes consensus control of multi-agent systems and sliding-mode control.

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.

Badong Chen received the Ph.D. degree in Computer Science and Technology from Tsinghua University, Beijing, China, in 2008. He is currently a professor with the Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an, China. His research interests are in signal processing, machine learning, artificial intelligence and robotics. He has authored or coauthored over 200 articles in various journals and conference proceedings (with 15000+ citations in Google Scholar), and has won the 2022 Outstanding Paper Award of IEEE Transactions on Cognitive and Developmental Systems. Dr. Chen serves as a Member of the Machine Learning for Signal Processing Technical Committee of the IEEE Signal Processing Society, and serves (or has served) as an Associate Editor for several international journals including IEEE Transactions on Neural Networks and Learning Systems, IEEE Transactions on Cognitive and Developmental Systems, IEEE Transactions on Circuits and Systems for Video Technology, Neural Networks and Journal of The Franklin Institute. He has served as a PC or SPC Member for prestigious conferences including UAI, IJCAI and AAAI, and served as a General Co-Chair of 2022 IEEE International Workshop on Machine Learning for Signal Processing.