International Journal of Robust and Nonlinear Control

WILEY

RESEARCH ARTICLE

Human–Machine Shared Stabilization Control Based on Safe Adaptive Dynamic Programming With Bounded Rationality

Junkai Tan[1,2] | Jingcheng Wang[3] | Shuangsi Xue[1,2] | Hui Cao[1,2] | Huan Li[1,2] | Zihang Guo[1,2]

[1]School of Electrical Engineering, Xi'an Jiaotong University, Xi'an, China | [2]The Shaanxi Key Laboratory of Smart Grid, and the State Key Laboratory of Electrical Insulation and Power Equipment, Xi'an Jiaotong University, Xi'an, China | [3]Department of Technical Supervision, Power Station Institute of New Energy Technical Supervision, Xi'an, China

Correspondence: Shuangsi Xue (xssxjtu@stu.xjtu.edu.cn)

Received: 24 March 2024 | **Revised:** 6 March 2025 | **Accepted:** 7 March 2025

Funding: This research was supported by the Science and Technology Project of Huaneng Group Headquarters under Grant HNKJ21-H53 and China Postdoctoral Science Foundation (Project ID: 2024M762602).

Keywords: adaptive dynamic programming | barrier function | bounded rationality | human–machine collaboration | shared control

ABSTRACT

This article considers the shared control of bounded rational human behavior with cooperative autonomous machines. For the collaboration of humans and machines, it is crucial to ensure the safety of the interactive process due to the involvement of human beings. First, a barrier-function-based state transformation is developed to ensure full state safety constraints. A level-k thinking framework is exploited to obtain bounded rationality. Every single level-k control policy is approximated by using adaptive dynamic programming. Inspired by the theory of human behavior modeling, a probabilistic distribution based on Softmax is utilized to model human behavior, which imitates the uncertainty of human intelligence in the cooperative game. Through the construction of a shared control framework, the control inputs of humans and machines are blended to achieve stabilization safely and efficiently. Finally, simulations are implemented to test the effectiveness of the proposed cooperation architecture. The result demonstrates that full-state asymmetric constraints and stabilization are guaranteed in commonly safety-critical situations, and the shared control framework ensures the safety of the overall system when one of the participants is not safety-aware.

1 | Introduction

Human–machine fusion decision is a rising topic in the field of safety-critical system control [1–3]. For safety-critical systems, there exist many complex, and asymmetric constraints for the safe operation of the system. These constraints challenge human beings to ensure the safety of the whole system due to the irrationality of human decision-making. By utilizing the human–machine cooperation control scheme, autonomous machine can gather enough information in a relatively short period to manage the crisis quickly and effectively. However, the question of to what extent and when machines are trusted by humans remains to be resolved. In recent years, game-based systems have gained significant interest due to their extensive usage in various fields, including economics, robotics, automated driving, and cyber-physical-systems [4–7]. The objective for both humans and machines in the collaboration is the same, which is to stabilize the whole system efficiently and safely.

Abbreviations: ADP, adaptive dynamic programming; HJI, Hamilton-Jacobi-Isaacs.

https://doi.org/10.1002/rnc.7931

1.1 | Related Work

To ensure the safety of the human–machine collaborative stabilizing control process, one safe adaptive dynamic programming (ADP) approach involves interactions between agents and the environment to learn the optimal controller [8–10]. Safe ADP includes mechanisms designed to guarantee that specific safety constraints are satisfied. Literatures studied multiple ways to improve the performance of safe ADP while ensuring safety. In [11], an adjusted policy iteration framework combined with control barrier function is proposed to deal with both state constraints and input saturation. The authors of [12] introduce a novel structure that combines actor-critic-identifier to identify system dynamics, resulting in enhanced performance in danger detection. A safe-guaranteed controller is proposed to prevent the state trajectory from causing collisions with non-convex boundary limits in [13]. In [14], a barrier-function-based transformation is proposed, which converts safety issues to stabilization problems to meet the full state constraints of a rectangular. In [15–17], the barrier function is integrated into the reward function to penalize the behavior of reaching the boundary. The barrier Lyapunov function-based safe ADP is investigated in [18] and [19], with application to the tracking control of the four-wheel vehicle. In [20], an end-to-end safe learning-based control framework is developed that uses an auxiliary system for approximation. Human beings are often involved in the control process of safety-critical systems; however, due to the vulnerability of human beings, how to ensure safety constraints strictly should be considered [21].

In the interaction process of cooperative human and machine, modeling human performance with irrationality remains an open challenge. In recent years, the theory of cognitive hierarchy has gained prominence to imitate the hierarchical structure of human thinking. In [22–24], learning-based method through solving the non-equilibrium game is developed to learn the cognitive hierarchy online, which achieves stabilization without acquiring system dynamics. The work of [25] proposes a cognition-modeling game architecture incorporating unmanned aircraft into the airspace system. In [26], the bounded-level reasoning structure is proposed to predict the decision-making process of human beings, which is constrained by the limited rationality of their beliefs. To learn the decision process of the human brain, a complementary learning approach is proposed in [27], which simulates different functional areas of the human brain. In [28, 29], the cooperation game of humans and self-driving vehicles is investigated to achieve collaborative human-vehicle decision-making. Additionally, the Softmax function from [30] is utilized to replicate the stochastic distribution of various levels of human behavior, based on the concept of bounded rationality [22, 31]. Bayesian inference is often used to infer the intent of natural humans by deriving probability distributions for different levels of human intelligence [32–35]. Therefore, bounded rationality is a promising approach to modeling irrational human behaviors However, the integration of the irrational human model and autonomous machine in the cooperative game remains an open challenge.

Human–machine cooperation enables the system to work under the guidance of human consciousness while reducing the mental and physical burden of humans in performing work by automating the operation of machines [36, 37]. By calculating the confidence of the human and machine, the shared control architecture allocates autonomy and mixes inputs from the human and machine [38, 39]. Data-driven controller design is a powerful tool for solving human–machine cooperation problems where information of the model is always unavailable. In [40] and [41], the inner–outer loop ADP-based controller is investigated, which improves the performance of the robotic manipulator through the interaction of human guidance. For autonomous driving, human-vehicle collaboration has become an essential research point. Human intervention in automobile-assisted driving is vital. In [42], a robust data-driven controller is developed to improve the reaction performance of the car driver. In [43], a novel human–machine shared control mechanism is developed, which takes the form of non-zero-sum games in a robotic arm. To further investigate more effective interaction process of human intelligence and autonomous machine, it is beneficial to develop a learning-based shared control framework that combines the direct cooperation of human and machine with the ADP technique.

In this article, a cooperative shared control framework that blends the input of bounded rational human and machine is developed, which achieves stabilization control cooperatively. To obtain irrational intelligence, the level-k policy of the cooperative nonzero-sum game is formulated. With the methodology of ADP, the level-k policy and value function are obtained by the approximation of the single critic network. To achieve direct cooperation between the human and machine, the shared control framework is introduced to blend the control inputs.

1.2 | Contributions

The contributions of this article are threefold:

1. A safe human–machine cooperative control game is formulated to achieve stabilization control. The cooperative game is developed to integrate human behavior with autonomous machine inputs. The full state constraints of the human–machine control system are guaranteed by a barrier-function-based system transformation. The optimal control policy of the human–machine cooperative game is obtained by the Nash equilibrium. Compared with other human–machine cooperative control methods [2, 7, 29, 43, 44], the developed method could achieve safe human–machine interaction even one of the participants is not safety-aware, which guarantees the safety of the overall system.
2. A level-k rationality architecture is developed based on the theory of bounded rationality. The bounded rational behavior is approximated via the online single-critic ADP. A probabilistic distribution of different intelligence levels is established, which imitates the time-varying and uncertain property of human being thinking framework. The proposed method could model the human behavior with bounded rationality and uncertainty in the cooperative game, compare with the existing bounded rationality modeling methods [3, 22, 24].

3. The framework of human–machine shared control is proposed, which utilizes the technique of linear arbitration to blend the control inputs of both human and machine. The shared control framework ensures the safety of the overall system when one of the participants is not safety-aware. The authority of human behavior over the shared control system is obtained by calculating the confidence of human intelligence. The proposed shared control system could achieve stabilization control cooperatively and safely in the human–machine cooperative game.

1.3 | Structure

This article is organized as follows. Section 2 illustrates the basic setup for the cooperative human–machine game. Section 3 outlines the barrier-function-based transformation system. Section 4 develops a level-k bounded rationality architecture and utilizes reinforcement learning to obtain the policies of different levels. Section 5 models the probabilistic form of human behavior and develops a shared control framework to blend the cooperative control inputs. Section 6 verifies the effectiveness of the proposed method. Section 7 concludes this article.

1.4 | Notations

The following notations are used in this article: $\mathbb{R}$ stands for the real numbers. $\mathbb{R}^n$ stands for the real n-dimensional vectors. $\mathbb{R}^{m\times n}$ stands for the real $m \times n$ matrices. $\|x\|$ stands for the Euclidean norm of vector x. $\dot{x}(t)$ stands for time derivative of $x(t)$. $\nabla V(s) = \frac{\partial V(s)}{\partial s}$ stands for partial derivative of $V(s)$ with respect to s.

2 | Preliminaries

2.1 | Human–Machine Cooperative Game

To investigate the human–machine cooperative game, we consider the continuous-time nonlinear affine input dynamical system defined as

$$\dot{x} = f(x) + g_h(x)u_h + g_m(x)u_m \tag{1}$$

where $x = \left[x_1 \cdots x_n\right]^{\mathrm{T}} \in \mathbb{R}^n$ is the system state, $u_i \in \mathbb{R}^{o_i}$, for $i = h,\ m$ is the control policy of human and machine, respectively, $f(x) : \mathbb{R}^n \to \mathbb{R}^n$ represent the nonlinear dynamics of the cooperative system. $g_h(x) : \mathbb{R}^n \to \mathbb{R}^{o_h}$ and $g_m(x) : \mathbb{R}^n \to \mathbb{R}^{o_m}$ represent the input matrix gain of human and machine, respectively. Note that the control input dynamics matrices of the human and machine are denoted with two different notations $g_h(x)$ and $g_m(x)$, respectively, for the subsequent analysis. However, the actual control input dynamics matrices of the human and machine could be the same in the real system.

Assumption 1. (Bounded functions [14]). The system given by (1) satisfies:

1. Function $f(x)$ is Lipschitz and bounded, such that $\|f(x)\| \le b_f$.
2. Functions $g_h(x)$ and $g_m(x)$ are bounded, such that $\|g_h(x)\| \le b_{g,h}$ and $\|g_m(x)\| \le b_{g,m}$.

Remark 1. Note that with the property given by Assumption 1, the transformation condition for the following barrier-function-based state transformation could be satisfied. Through the barrier-function-based transformation, the finite state constraints of the original system are mapped to transformed infinite state constraints, so the bounded conditions of the system dynamics are essential for the feasibility of the transformation. With the full-state constraints and continuous nonlinear-affine system setup of this article, where all states are bounded by finite values, the continues system dynamics f, g_h, and g_m could be feasibly bounded with finite states.

Blending the human and machine inputs into a hybrid controller and fusing the affine dynamics of the human and machine, a shared control architecture could be designed. The designed architecture allocates autonomy based on the confidence of the inputs. The dynamics of shared control can be defined as

$$\dot{x} = f(x) + g_{blend}(x)u_{blend} \tag{2}$$

where $g_{blend} = [g_h, g_m] : \mathbb{R}^n \to \mathbb{R}^{(o_h+o_m)}$ is a blending dynamics, u_{blend} is a blending control input.

$$u_{blend} = \beta(u_h, u_m, x) \tag{3}$$

where $\beta(u_h, u_m, x) \in \mathbb{R}^{(o_h+o_m)\times n}$ is a blending paradigm, which combines the human input u_h and machine input u_m.

Remark 2. Note that the blending dynamics g_{blend} is a vector of the human and machine input gain matrix, which is used to blend the control inputs of the human and machine. The blending paradigm β is a function of the human and machine inputs and the system state, which is used to allocate the authority of human behavior in the shared control system. The allocation of the authority of human behavior is determined by the confidence of human intelligence which is calculated by the arbitration function defined in the next subsection.

To achieve the cooperation of the human–machine system, the same performance index for the human and machine is designed as

$$\mathcal{J}_{coop}\left(x_0; u_h, u_m\right) = \frac{1}{2}\int_0^\infty r_{coop}\left(x, u_h, u_m\right) d\tau \tag{4}$$

where $r_{coop}\left(x, u_h, u_m\right) = M(x) + \sum_{j\in\{h,m\}} u_j^T R_{jj} u_j$ is the fully cooperative reward function of human–machine system, $M(x)$ is a quadratic function of state x.

Definition 1. (Nash equilibrium [45]). Input pair $\left(u_h^\star, u_m^\star\right)$ is the Nash equilibrium, if it satisfies:

$$\mathcal{J}_{coop}\left(x_0; u_h^\star, u_m^\star\right) \le \mathcal{J}_{coop}\left(x_0; u_h, u_m^\star\right), \forall u_h \tag{5}$$

$$\mathcal{J}_{coop}\left(x_0; u_h^\star, u_m^\star\right) \le \mathcal{J}_{coop}\left(x_0; u_h^\star, u_m\right), \forall u_m \tag{6}$$

The optimal control policy of the fully cooperative human–machine system is the Nash equilibrium.

2.2 | Human Behavior Modeling and Arbitration

Human behavior is characterized by suboptimality and volatility. According to the literature [30], for behaviors with a probabilistic distribution, the human decision-making mechanism can be represented by a Softmax function as

$$\Pr\left\{u_t = u^k | x\right\} = \frac{e^{-r(x,u^k)}}{\sum_j e^{-r(x,u^j)}} \tag{7}$$

where $r(x, u)$ is the reward function similar to r_{coop} from (4), u^k is the k-th human behavior. The Softmax function is used to model the probabilistic distribution of human behavior, which imitates the uncertainty of human intelligence in the cooperative game, which selects level-k policies with the probability of the Softmax function. As the human–machine system is safety-critical, the confidence of human intelligence should be considered. The confidence of human intelligence is calculated by the confidence function $c(t)$, which is defined as Linear arbitration is the commonly used form of arbitration. The arbitration function of the shared control system increases as confidence in humans grows, with a lower bound of 0 and an upper bound of no more than 1. The specific form of the arbitration function used in this article is given as follows:

$$\alpha = \begin{cases} 0, & c(t) \leqslant \theta_1 \\ \frac{\theta_3}{\theta_2 - \theta_1} \cdot c(t), & \theta_1 < c(t) < \theta_2 \\ \theta_3, & c(t) \geqslant \theta_2 \end{cases} \tag{8}$$

where $c(t)$ is the confidence to human intelligence, and θ_i with $i \in \{1, 2, 3\}$ are the parameters for the arbitration function. The maximum of arbitration is θ_3, determining the maximum share of the human influencing overall decision-making. When there is insufficient confidence in the human, $\alpha = 0$, that is, the human input is not executed. The blue curve in Figure 1 illustrates the arbitration functions above. The arbitration function reflects the authority of humans and machines for the overall system under shared control. When confidence in humans is low, humans do not contribute to the shared control system. When a certain threshold is reached, human behavior begins to play a role and is proportional to the amount of confidence, showing a partially linear relationship. When confidence reaches a certain high level, the shared control system maximizes the allowable level of human behavior and remains. To achieve the objective of stabilizing control of the human–machine system with safety assurance, a cooperative game-based safe shared control framework is developed, as shown in Figure 2. For the human irrationality modeling, the level-k bounded rationality is developed to model the human behavior, and the probabilistic distribution of human intelligence is obtained by the Softmax function. To ensure the safety of the human–machine system, a barrier-function-based transformation is utilized to transform the state constraints into stabilization problems. The arbitration function is used to calculate the confidence of human intelligence and allocate the authority of human behavior in the shared control system.

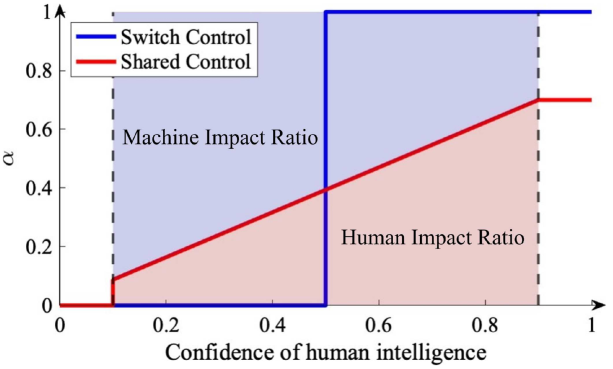


FIGURE 1 | Arbitration function.

3 | Barrier-Function-Based State Transformation

The system state involving human beings should always satisfy the safe constraints. To ensure safety, a barrier-function-based transformation system is given in this section.

3.1 | Barrier Function Transformation

In this subsection, we first consider the problem of state constraints. To simplify the notation of safety limits, the state constraints set is given as $x \in \mathcal{O}$, where $\mathcal{O} = \{x \in \mathbb{R}^n | a \leq Cx + p \leq A\}$, $a = [a_1, \dots, a_l]^T \in \mathbb{R}^l$, $A = [A_1, \dots, A_l]^T \in \mathbb{R}^l$, $p = [p_1, \dots, p_l]^T \in \mathbb{R}^l$ and $C \in \mathbb{R}^{l \times n}$. The problem of the safety-critical game can be formulated as follows.

Problem 1. Consider the nonlinear system (1), and given the cooperative performance index (4), find the Nash equilibrium policies $\left(u_h^\star, u_m^\star\right)$, while satisfying $x \in \mathcal{O}$.

To simplify the analysis procedure without loss of generality, we choose the state constraint in the form of $x_k \in \left(d_k, D_k\right)$, $k = 1, \dots, n$. The lower constraint and upper constraint satisfy $d_k < D_k$ and $\|d_k\| \neq \|D_k\|$, which means the state constraints is asymmetric. To address the state constraint issue, the transformation of the system state using the barrier function is introduced. The safety problem with constraint $x \in \mathcal{O}$ is transformed into a stabilization problem.

Based on the constraint $x_k \in \left(d_k, D_k\right)$, $k = 1, \dots, n$, we select the barrier function in the form of

$$b(x_k; d_k, D_k) = \log\left(\frac{D_k}{d_k} \frac{d_k - x_k}{D_k - x_k}\right) \tag{9}$$

The inverse function of the barrier function $b(x_k; d_k, D_k)$ on the interval (d_k, D_k) is

$$b^{-1}(y_k; d_k, D_k) = \frac{D_k d_k \left(e^{\frac{y_k}{2}} - e^{-\frac{y_k}{2}}\right)}{d_k e^{\frac{y_k}{2}} - D_k e^{-\frac{y_k}{2}}} \tag{10}$$

With the barrier function $b(\cdot)$ and the state $x \in \mathbb{R}^n$ of system (1), the system state transformation can be summarized as

$$s_k = b\left(x_k; d_k, D_k\right) = b_k \tag{11}$$

$$x_k = b^{-1}\left(s_k; d_k, D_k\right) = b_k^{-1}, \ \forall k = 1, \dots, n \tag{12}$$

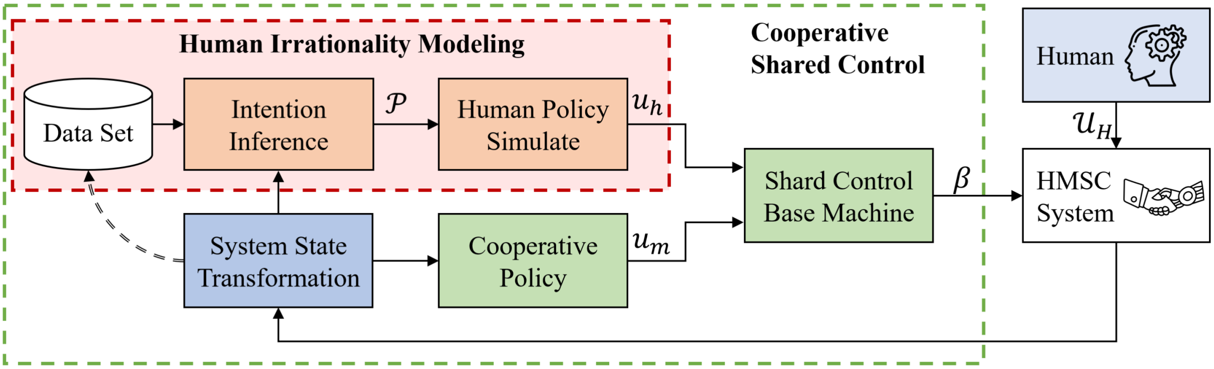


FIGURE 2 | Scheme of the HMSC system.

By utilizing the chain rule, the time derivative of the transformed state s_k is obtained as

$$\frac{ds_k}{dt} = \left(\frac{dx_k}{ds_k}\right)^{-1}\frac{dx_k}{dt} \tag{13}$$

The dynamics of transformed state s_k ($\forall k = 1, \dots, n$) can be expressed as

$$\dot{s}_k = \frac{\dot{x}_k}{\frac{db^{-1}(s_k;d_k,D_k)}{ds_k}} = F_k(s) + G_k^h(s)u_h + G_k^m(s)u_m \tag{14}$$

where $s = \left[s_1, \dots, s_n\right]^{\mathrm{T}}$, $F_k(s_k) = \tau(s_k)\times f\left(\left[b_1^{-1}, \dots, b_n^{-1}\right]^{\mathrm{T}}\right)$, $G_k^h(s) = \tau(s_k)\times g_h\left(\left[b_1^{-1}, \dots, b_n^{-1}\right]^{\mathrm{T}}\right)$ and $G_k^m(s) = \tau(s_k)\times g_m\left(\left[b_1^{-1}, \dots, b_n^{-1}\right]^{\mathrm{T}}\right)$ with $\tau(s_k) = \left(\frac{db^{-1}(s_k;d_k,D_k)}{ds_k}\right)^{-1}$.

Then the transformed system dynamics (14) could be written in the following compact form:

$$\dot{s} = F(s) + G^h(s)u_h + G^m(s)u_m \tag{15}$$

where $F(s) : \mathbb{R}^n \to \mathbb{R}^n$ is the nonlinear dynamics of transformed system. $G^h(s) : \mathbb{R}^n \to \mathbb{R}^{o_h}$ and $G^m(s) : \mathbb{R}^n \to \mathbb{R}^{o_m}$ are the transformed input gain matrix.

Assumption 2. Assuming that as the original Lipschitz and bounded system dynamics $f(x)$, $g_h(x)$, and $g_m(x)$ is transformed into the transformed system dynamics $F(s)$, $G^h(s)$, and $G^m(s)$, the transformed system dynamics $F(s)$ is also Lipschitz

Remark 3. Note that the proposed barrier-function-based state transformation safe ADP method ensures system safety strictly, compared with barrier-penalty method [15, 16] and barrier-lyapunov method [18, 19]. Through the barrier-function-based state transformation, the finite state constraints of the original system are mapped to infinite state constraints of the transformed system. Once the transformed system is stabilized, the original system is also stabilized and the safety constraints are satisfied strictly with the same control input. By designing stabilization controllers for the transformed system, the safety constraints can be strictly satisfied.

3.2 | Nash Equilibrium in Transformed System

Based on the transformed state s and system dynamics, the Nash equilibrium for (15) shall be obtained.

The human–machine cooperative game aims to stabilize the transformed system (15), with minimum resource consumption. The minimization problem can be solved by minimizing the value function as

$$V_{coop}\left(s, u_h, u_m\right) = \int_t^{\infty} r_{coop}\left(s, u_h, u_m\right)d\tau \tag{16}$$

Definition 2. Consider system (15), a pair of policies $u_{h,m} = \left\{u_h, u_m\right\}$ is admissible, if $u_{h,m}$ stabilizes the transformed system (15), and value function V from (16) is finite.

Thus, for the optimality of controlling transformed system (15), an admissible pair of policies $u_{h,m}^* = \left\{u_h^*, u_m^*\right\}$ is the Nash equilibrium, which obtains the optimal value function:

$$V_{coop}^*\left(s, u_h, u_m\right) = \min_{u_h,u_m}\int_t^{\infty} r_{coop}\left(s, u_h, u_m\right)d\tau \tag{17}$$

Define the Hamiltonian function for the transformed human–machine cooperative system as

$$\begin{aligned}\mathcal{H}\left(s, \nabla V, u_h, u_m\right) \triangleq (\nabla V)^{\mathrm{T}}\left[F(s) + G^h(s)u_h + G^m(s)u_m\right] \\ + r_{coop}\left(s, u_h, u_m\right)\end{aligned} \tag{18}$$

where $\nabla V = \frac{\partial V_{coop}}{\partial s}$ is the gradient of the value function.

By differentiating the Hamiltonian function and applying the stationary conditions, in the form of $\frac{\partial \mathcal{H}_i}{\partial r_i} = 0$, we can obtain the optimal controller pair as

$$u_i^{\star}(s) = -\frac{1}{2}R_i^{-1}\left(G^i(s)\right)^{\mathrm{T}}\nabla V^{\star}, \ i = h, m \tag{19}$$

Substituting the optimal controller (19) into the Hamiltonian (18) yields the Hamilton-Jacobi-Isaacs(HJI) equation as

$$\begin{aligned}0 = (\nabla V^*)^{\mathrm{T}}\left(F(s) - \frac{1}{2}\sum_{j\in\{h,m\}} G^j(s)R_j^{-1}(G^j(s))^{\mathrm{T}}\nabla V^*\right) \\ + Q(s) + \frac{1}{4}\sum_{j\in\{h,m\}} (\nabla V^*)^{\mathrm{T}}G^j(s)R_j^{-1}(G^j(s))^{\mathrm{T}}\nabla V^*\end{aligned}$$

Similar to Lemma 1 from [14], given the transformed system (15), we can solve the full-state constraints by finding a pair of Nash Equilibrium policies $u_{h,m} = \{u_h, u_m\}$.

The following section will introduce the cognitive hierarchy to obtain a level-k bounded rationality.

4 | Level-k Bounded Rationality

This section introduces a level-k bounded rationality structure to obtain different levels of human intelligence.

4.1 | Initial Policy (Level-0) and Level-1 Policy

For the cooperative human–machine system, level-0 rationality represents an instinctive reaction, which means players' behaviors are non-cooperative. To prevent the potential stochastic danger, we will obtain human level-0 rationality by solving an optimization problem, which is in the form of minimizing a specific value function as

$$V^0_{u_h}\left(s_0\right) = \min_{u_h^0} \int_0^\infty \left(M(s) + (u_h^0)^{\mathrm{T}} R_h u_h^0\right) \mathrm{d}\tau \tag{20}$$

which is subject to the dynamics of the system $\dot{s} = F(s) + G^h(s)u_h^0$, $M(s)$ is the quadratic function of transformed state s. According to the optimal control theory, the stationary condition for the optimization of value (20) is

$$u_h^0(s) = -\frac{1}{2} R_h^{-1} (G^h(s))^{\mathrm{T}} \nabla V_h^0 \tag{21}$$

where $\nabla V_h^0 = \frac{\partial V^0_{u_h}(s)}{\partial s}$, the value function $V^0_{u_h}$ is known to satisfy the Hamilton-Jacobi-Isaacs(HJI) equation, namely

$$\mathcal{H}\left(s, \nabla V_h^0, u_h^0\right) = \left(\nabla V_h^0\right)^{\mathrm{T}} \left[F(s) + G^h(s)u_h^0\right] + r_{coop}\left(s, u_h^0, 0\right) = 0 \tag{22}$$

Assuming the human always acts the level-0 policy, the level-1 policy of the machine could be solved subsequently, which is the optimal response to the initial level-0 policy.

To acquire the level-1 policy of the machine, an optimization problem is established as follows:

$$V^1_{u_m}\left(s_0\right) = \min_{u_m^1} \int_0^\infty \left(M(s) + (u_m^1)^{\mathrm{T}} R_m u_m^1 + (u_h^0)^{\mathrm{T}} R_h u_h^0\right) \mathrm{d}\tau \tag{23}$$

which is subject to the dynamics of the system $\dot{s} = F(s) + G^h(s)u_h^0 + G^m(s)u_m^1$. The optimal policy for the optimization problem (23) is

$$u_m^1(s) = -\frac{1}{2} R_m^{-1} (G^m(s))^{\mathrm{T}} \nabla V_m^1 \tag{24}$$

where $\nabla V_m^1 = \frac{\partial V^1_{u_m}(s)}{\partial s}$, the value function $V^1_{u_m}$ is known to satisfy the HJI equation, namely $\mathcal{H}\left(s, \nabla V_d^1, u_h^0, u_m^1\right) = 0$.

4.2 | Level-k and Level-$(k+1)$ Policies

An iterative procedure is used to formulate higher-level rational policies for the human and machine, respectively, in which the human and machine have the belief that their partner uses lower-level rationality.

Interacting with the machine which uses level-$(k-1)$ rationality, the human player acquires level-k thinking by solving the minimization of the value function as

$$V^k_{u_h}\left(s_0\right) = \min_{u_h^k} \int_0^\infty \left(M(s) + (u_h^k)^{\mathrm{T}} R_h u_h^k + (u_m^{k-1})^{\mathrm{T}} R_m u_m^{k-1}\right) \mathrm{d}\tau \tag{25}$$

which is subject to the dynamics as

$$\dot{s} = F(s) + G^h(s)u_h^k + G^m(s)u_m^{k-1} \tag{26}$$

The corresponding HJI equation is $\mathcal{H}\left(s, \nabla V_h^k, u_h^k, u_m^{k-1}\right) = 0$. The stationary condition leads to the formulation of the level-k policy as

$$u_h^k(s) = -\frac{1}{2} R_h^{-1} (G^h(s))^{\mathrm{T}} \nabla V_h^k \tag{27}$$

Similarly, the level-$(k+1)$ rationality could be obtained by solving the subsequent minimization of the value function as

$$V^{k+1}_{u_m}\left(s_0\right) = \min_{u_m^{k+1}} \int_0^\infty \left(M(s) + (u_m^{k+1})^{\mathrm{T}} R_m u_m^{k+1} + (u_h^k)^{\mathrm{T}} R_h u_h^k\right) \mathrm{d}\tau \tag{28}$$

which is subject to the dynamic

$$\dot{s} = F(s) + G^h(s)u_h^k + G^m(s)u_m^{k+1} \tag{29}$$

The level-k controller for the system (28) is given by

$$u_m^{k+1}(s) = -\frac{1}{2} R_m^{-1} (G^m(s))^{\mathrm{T}} \nabla V_m^{k+1} \tag{30}$$

The HJI equation satisfies $\mathcal{H}\left(s, \nabla V_d^{k+1}, u_h^k, u_m^{k+1}\right) = 0$.

Theorem 1. *Consider the transformed system (15), given the human and machine bounded level-k and level-$(k+1)$ rationality, respectively, the corresponding value functions are positive definite. If the following conditions hold:*

$$\begin{gathered} u_h^k(0) = 0, \; u_m^{k+1}(0) = 0, \\ \dot{V}^k_{u_h}(\mathrm{s}) < 0, \dot{V}^{k+1}_{u_m}(\mathrm{s}) < 0, \forall \mathrm{s} \neq 0, \\ \mathcal{H}\left(s, \nabla V_h^k, u_h^k, u_m^{k-1}\right) = 0, \forall \mathrm{s}, \\ \mathcal{H}\left(s, \nabla V_d^{k+1}, u_h^k, u_m^{k+1}\right) = 0, \forall \mathrm{s}, \\ \mathcal{H}\left(s, \nabla V_h^k, u_h, u_m^{k-1}\right) \geqslant 0, \forall \mathrm{s}, u_h, \\ \mathcal{H}\left(s, \nabla V_d^{k+1}, u_h^k, u_m\right) \leqslant 0, \forall \mathrm{s}, u_m \end{gathered}$$

Then the bounded rational policy pair $u_{h,m} = \{u_h^k, u_m^{k+1}\}$ stabilizes system (26) and (29) asymptotically. The values of policies (27) and (30) are the minimum as

$$\mathcal{J}_{coop}\left(x_0; u_h^k, u_m^{k-1}\right) = \min_{u_h} \mathcal{J}_{coop}\left(x_0; u_h, u_m^{k-1}\right), \forall x_0 \tag{31}$$

$$\mathcal{J}_{coop}\left(x_0; u_h^k, u_m^{k+1}\right) = \min_{u_m} \mathcal{J}_{coop}\left(x_0; u_h^k, u_m\right), \forall x_0 \tag{32}$$

Proof. Similar to Theorem 2 of article [24], first we obtain the performance index of policy $u_{h,m} = \{u_h, u_m^{k-1}\}$, namely

$$\begin{aligned} J_{coop}(s_0, u_h, u_m^{k-1}) \\ &= \int_0^\infty \left(-\dot{V}_{u_h}^k + \mathcal{H}\left(\tau, \nabla V_h^k, u_h, u_m^{k-1}\right)\right) d\tau \\ &= -\lim_{t\to\infty} V_{u_h}^k(s(t)) + V_{u_h}^k(s_0) + \int_0^\infty \left(\mathcal{H}\left(\tau, \nabla V_h^k, u_h, u_m^{k-1}\right)\right) d\tau \end{aligned}$$

Since the transformed system is stable under input u_h, it can be inferred that $\lim_{t\to\infty} V_{u_h}^k(s(t)) = 0$, which results in

$$\begin{aligned} J_{coop}(s_0, u_h, u_m^{k-1}) \\ &= V_{u_h}^k(s_0) + \int_0^\infty \left(\mathcal{H}\left(\tau, \nabla V_h^k, u_h, u_m^{k-1}\right)\right) d\tau \geq V_{u_h}^k(s_0) \end{aligned}$$

Subsequently, by replacing u with u_h^k, and utilizing the HJI equation $\mathcal{H}\left(s, \nabla V_h^k, u_h^k, u_m^{k-1}\right) = 0$, Equation (31) is obtained. Similarly, the Equation (32) could be concluded. Therefore, policies pair $u_{h,m} = \{u_h^k, u_m^{k+1}\}$ can globally asymptotically stabilize the system (26) and (29), respectively. □

Remark 4. The level-k model inherently captures bounded rationality and potentially irrational behavior through its hierarchical reasoning structure. At level-0, the human exhibits non-strategic behavior that may appear irrational from a game-theoretic perspective. Higher levels represent increasingly sophisticated but still boundedly rational reasoning, as humans typically do not use infinite recursive thinking. This framework allows us to model various degrees of human behavioral sophistication, from simple reactive responses to more strategic thinking, while maintaining computational tractability.

Remark 5. The Nash equilibrium of the human–machine system with level-k bounded rationality is obtained by solving the minimization of the value function in each level of the cognitive hierarchy. By iteratively solving the minimization of the value function, the level-k policy of the human and the level-$(k+1)$ policy of the machine are obtained. The existence and uniqueness of the Nash equilibrium are guaranteed by the iterative procedure of the level-k bounded rationality as level approaches infinity ($k \to \infty$) in Theorem 1. It should be noted that Theorem 1 and its proof could be referred to the existing literature [24, 46].

Remark 6. Level-k-based human behavior modeling together with barrier-function-based transformation can simulate the decision-making of the irrational human in the safety state. The proposed level-k-based human behavior modeling is designed to capture the bounded rationality of human agents in the decision-making process, which integrates the level-k framework with probabilistic distribution to model the uncertainty and bounded rationality of human agents. Through combining the level-k framework with the barrier-function-based transformation, the safe level-k policy is learned under the transformed system dynamics, which means this safe policy is able to maintain the safety constraints of the original system. Although the modeling of the irrational human behavior contains some uncertainties and bounded rationality, every level-k policy is learned under the safety constraints of the transformed system, which ensures that the irrational human behavior can be simulated with certain safety guarantees. Also, the sub-optimality of the level-k policy could be achieved in the cooperative human–machine system. While the level of intelligence k increases to infinity, the optimality of the human–machine system is guaranteed.

4.3 | Online Learning for Bounded Rationality

In this subsection, we use the ADP method to approximate bounded rationality online. Two critic networks are utilized to obtain the value functions V_i, $\forall i \in \{h, m\}$ of the human and machine, respectively. The corresponding policies of the human and machine are denoted as u_i^j for simplification. Up to level-k, we select the single-layer network to approximate the value function as

$$V_i^j = (W_i^j)^T \phi^j(s) + \epsilon_i^j(s) \tag{33}$$

where $W_i \in \mathbb{R}^{p_i}$ represent the ideal neuron weight of the single-layer network and $\phi(x) \in \mathbb{R}^{n \times p_i}$ is the corresponded activation function, p_i is the number of hidden layer neuron, and $\epsilon_i(x)$ is the approximation error of the single-layer network. The gradient for the value function is

$$\nabla V_i^j = (\nabla \phi^j(x))^T W_i^j + (\nabla \epsilon_i^j)^T(s) \tag{34}$$

The estimated value function $\hat{V}_i$ is expressed as

$$\hat{V}_i^j = (\hat{W}_i^j)^T \phi^j(x) \tag{35}$$

where $\hat{W}_i^j \in \mathbb{R}^{p_i}$ is the estimated weight of the single network.

To reduce the computational load, the general form policy u_i^j, $i = h, m$ is approximated by the single neural network

$$u_i^j = -\frac{1}{2} R_i^{-1} (G^i(s)^T ((\nabla \phi_i^j(s))^T W_i^j + (\nabla \epsilon_i^j(s))^T) \tag{36}$$

With the gradient of the value function presented in (34), the actual controller can be expressed as

$$\hat{u}_i^j = -\frac{1}{2} R_i^{-1} G^i(s)^T (\nabla \phi_i^j(s))^T \hat{W}_i^j \tag{37}$$

Based on the estimated value function (35) and control (37), the approximation error of the HJI equation is defined as

$$\begin{aligned} \mathcal{H}\left(s, \nabla \phi_i^j, u_i^j\right) = \sum_{l=h,m} (u_l^j)^T R_i u_l + M(s) \\ + [(W_i^j)^T \nabla \phi_i^j + (\nabla \epsilon_i^j)^T] \left(F + \sum_{l=h,m} G^l u_l^j \right) \end{aligned}$$

For the simplification of notation, denote $\mathcal{H}\left(s, \nabla \phi_i^j, u_i^j\right) = e_{H,i}^j$, $\mathcal{H}\left(s, \nabla \phi_i^j, \hat{u}_i^j\right) = e_i^j$ and $\omega_i^j = \nabla \phi_i (F + G^h u_h^j + G^m u_m^j)$.

To obtain the approximated policy $u_{h,m}^j$, an optimization could be constructed with the approximation error of the HJI equations. First, by combining the historical and instantaneous data, the energy-like objective E_i for ADP optimization is defined as follows:

$$E_i^j = \frac{1}{2}\sum_{k=1}^{M} \frac{(e_{i,k}^j)^2}{\left(1 + (\omega_{i,k}^j)^T \omega_{i,k}^j\right)^2} \tag{38}$$

where $\omega_{i,k}^j, k = 1, \ldots, M$ is the historical data of ω_i^j, $\omega_{i,0}^j$ is the current record of ω_i^j. M is the length of the historical stack. Define $\overline{\omega}_i^j = [\omega_{i,1}^j \ldots \omega_{i,M}^j]$ as the historical data stack.

Therefore, based on the property of the objective function E_i^j, using the gradient descent method, the learning law for the estimated critic network weight $\hat{W}_i$ can be derived as

$$\dot{\hat{W}}_i^j = -a_i^j \frac{\partial E_i^j}{\partial \hat{W}_i^j} = -a_i^j \sum_{k=1}^{M} \frac{\omega_{i,k}^j e_{i,k}^j}{\left(1 + (\omega_{i,k}^j)^T \omega_{i,k}^j\right)^2} \tag{39}$$

where the learning rate of each bounded rational level, denoted as a_i^j, plays a crucial role in determining the convergence speed of network weights W_i.

To prove the stability of the proposed regular ADP controller, the Lyapunov stability analysis is presented. First, the error dynamics of $\tilde{W}_i^j$ is given as

$$\dot{\tilde{W}}_i^j(t) = -a_i^j \sum_{k=1}^{M} \frac{\omega_{i,k}^j}{(\omega_{i,k}^j)^T \omega_{i,k}^j + 1} \left[\frac{(\omega_{i,k}^j)^T \tilde{W}^j + e_{H,i}^{k,j}}{(\omega_{i,k}^j)^T \omega_{i,k}^j + 1} \right] \tag{40}$$

To facilitate the proof of stability, we present the following assumption.

Assumption 3. The following conditions hold:

1. The historical stack $\overline{\omega}_i^j$ satisfies $rank(\overline{\omega}_i^j) = p_i$.
2. The Hamiltonian error $e_{H,i}^j$ is upper bounded by a positive constant $e_{Hmax,i}^j$, such that $e_{H,i}^j \leq e_{Hmax,i}^j$.
3. There exists $\mu_1 \in \mathbb{R}^+$ and $\mu_2 \in \mathbb{R}^+$, such that the following persistent excitation condition holds [47]:

$$\mu_1 I \leq \int_{t_0}^{t_0+\delta} \Psi_i^j(s)\Psi_i^j(s)^T \mathrm{d}s \leq \mu_2 I, \quad \forall t_0 \in \mathbb{R}^+ \tag{41}$$

where $\Psi_i^j(s) = (\omega_{i,k}^j)/((\omega_{i,k}^j)^T \omega_{i,k}^j + 1)^2$, δ is a positive constant, and I is an identity matrix.

Remark 7. Note that this assumption is made to satisfy the persistent excitation (PE) condition for the learning law (39), which is a common requirement for the stability of the learning-based controller. As for the bounded rationality, the PE condition is essential to ensure the convergence of the critic network weights. To satisfy the PE condition, the historical data $\overline{\omega}_i^j$ should be rich enough to provide sufficient information for the learning process, by setting the length of the historical stack M to be a reasonable value and the learning rate a_i^j to be properly selected.

Next, we present the main theorem about stability.

Theorem 2. *Suppose that Assumption 1–3 hold. Consider the critic network W_i^j and the concurrent learning-based update law (39), the critic weights error $\tilde{W}_i^j$ is uniformly ultimately bounded (UUB).*

Proof. Similar to the theorem result of work [14, 48], define the following Lyapunov function:

$$V_i^j(t) = \frac{1}{2a_i^j}(\tilde{W}_i^j)^T \tilde{W}_i^j \tag{42}$$

For each irrational level of human and machine, we have

$$\dot{V}_i^j = -(\tilde{W}_i^j)^T \zeta_i(t) \tilde{W}_i^j + (\tilde{W}_i^j)^T \eta_i \tag{43}$$

where

$$\zeta_i = \sum_{k=1}^{p} \frac{\omega_i \left(\omega_{i,k}^j\right)^{\mathrm{T}}}{\left[1 + \left(\omega_{i,k}^j\right)^{\mathrm{T}} \omega_{i,k}^j\right]^2}, \; \eta_i = \sum_{k=1}^{p} \frac{\omega_{i,k}^j e_{H,i}^{k,j}}{\left[1 + \left(\omega_{i,k}^j\right)^T \omega_{i,k}^j\right]^2}$$

With the assumption that $rank(\overline{\omega}_i^j) = p_i + 1$, we could obtain the following inequality

$$\dot{V}_i^j \leq -\lambda_{\min}(\zeta_i)||\tilde{W}_i^j||^2 + ||\tilde{W}_i^j||\left(\frac{M+1}{2}\right)e_{Hmax,i}^j \tag{44}$$

Lyapunov candidate's differentials $\dot{V}_i$ can be guaranteed to be negative if $\|\tilde{W}_i^j\| \geq \frac{(M+1)e_{Hmax,i}^j}{2\lambda_{\min}(\zeta_i)}$, consequently the error of NN weights is UUB. The proof is completed. □

Remark 8. Theorem 2 demonstrates the stability of the learning-based controller for the level-k bounded rationality. The UUB property of the critic network weights error is guaranteed by the Lyapunov stability analysis. By satisfying the persistent excitation condition and selecting the proper learning rate, the critic network weights will converge to a bounded region. While Theorem 2 establishes UUB rather than strict convergence, this is sufficient for practical implementation. The bounded error means the network weights will stabilize within a small neighborhood of their optimal values. In practice, we can determine the final weights by either: (1) running the learning process for a sufficient time period until the weight updates become negligible or (2) selecting the proper initial NN weights that achieve the approximate optimal solution within the UUB bound. It could also be beneficial to tune the learning rate and exploration parameters to ensure convergence within a desired accuracy. Further details on weight evolution and convergence could be seen in the simulation results, where the network weights stabilize within sub-optimal bounds region after a certain number of iterations, as shown in Figures 5 and 6.

5 | Shared Control of Human–Machine System

In this section, we propose a shared control framework in which the machine cooperates with the human possessing time-varying intelligence to pursue the same goal. To adopt our proposed control framework in the real human–machine scenario, the fixed-level policy throughout the interaction should be refrained, which imposes limitations on the human about their utilization of rationality and ignores the variability of human behavior. Then a human–machine shared control framework is developed, which blends the inputs of the human–machine cooperation.

5.1 | Human Irrationality Modeling and Intent Inference

As mentioned in the previous section, the machine calculates the level-k rational policies by implementing an ADP algorithm that cooperates with human policies. As a result, rather than using a precise level of human behavior, a probabilistic distribution of human policies is utilized to model the stochastic and dynamical effects resulting from human behavioral decision-making.

Problem 2. Given a certain finite number of level-k value functions and strategies, find a level-k probability distribution from simulations based on measured human behavior.

Assumption 4. Assuming that there is no deviation between the actual human behavior and the measured human behavior, the impact of human on the system, which is represented as the control input of human behavior, can be precisely transmitted and observed through channels and sensors.

Remark 9. It should be noted that this assumption is made to ensure the accuracy of the measured human behavior, which is essential for the probabilistic distribution modeling of human policies. The core idea of this assumption is that the machine agent is able to observe and calculate the human agent's control input and its impact on the system by computing r^k from Equation (45), the human agent's input is usually import from certain digital input channels, which could be feasibly measured by analog sensors or other devices and transmitted digitally to the machine agent.

Definition 3. The error of optimism is defined as the difference between the measured human behavior denoted $\mathcal{U}_h(\tau)$, and the human policy of level-k.

$$r^k = \int_{T_{\text{int}}} \left\| \mathcal{U}_h(\tau) + \frac{1}{2} R_h^{-1} (G^h)^T (\nabla \phi_h^j)^T \hat{W}_h^j \right\| \mathrm{d}\tau \tag{45}$$

where $j \in \{1, \ldots, k_m\}$, k_m is the maximum level of human rationality been computed. Remark that (45) is the norm of the measured distance from human approximated policy of each level.

Consider the machine performing at the optimal response, namely the Nash equilibrium $\mathcal{U}_h(t) = -\frac{1}{2} R_h^{-1} (G^h)^T (\nabla \phi_h)^T W_h^*$, $\forall t \geq 0$. According to Theorem 1, it is achievable to train any given level-k to attain convergence with the optimal response strategy of a human, which in the form of $u_h^j(t) = -\frac{1}{2} R_h^{-1} (G^h)^T (\nabla \phi_h^j)^T \hat{W}_h^j$. Moreover, the level-$k$ rationality will approach infinity and ultimately converge to the Nash solution. This implies that the Nash solution represents the limit of the level-k, that is, $\lim_{j \to +\infty} \left\| V_h^j - V_h^\star \right\| = 0$, it provides

$$\begin{aligned} \lim_{j \to +\infty} u_h^j(t) &= \lim_{j \to +\infty} \left(-\frac{1}{2} R_h^{-1} (G^h)^T (\nabla \phi_h^j)^T \hat{W}_h^j \right) \\ &= -\frac{1}{2} R_h^{-1} (G^h)^T (\nabla \phi_h)^T \hat{W}_h^* = \mathcal{U}_h(t) \end{aligned} \tag{46}$$

Consequently, $\lim_{k \to +\infty} r^k = 0$, the following probabilistic distribution model would be established based on the error r^k.

During each interval of interaction T_{int} , the error r^k will be organized in a vector $\mathbf{r}$ of the form $\mathbf{r} = \left[r^1, r^2, \ldots, r^{k_m} \right]$. To formulate the bounded rational human behavior, the softmax function is utilized to transform the error vector $\mathbf{r}$ into a bounded value vector. Then, the policy of the human player could be represented by the distribution as

$$\mathcal{P}(\mathcal{U}_h) = \frac{e^{-r^k}}{\sum_{i=1}^{k_m} e^{-r^i}} \tag{47}$$

Remark 10. The distribution known as "Softmax" is a routine selection for modeling human decision-making [30]. Minor errors indicate greater chances of choosing a suitable bounded rational behavior.

To obtain the safe and stabilizing policies of level-k human–machine cooperation, an online ADP algorithm is formulated in Algorithm 1.

Due to the agility of human consciousness, the intended act of human beings is always powerful and necessary in complex scenarios, such as rapid responses to emergencies safely. However, human is bounded rational, for the limitation of human observation and intelligence. The execution of the machine is predefined, making full use of observed data to perform tasks that achieve localized optimal control.

A typical shared control framework is divided into two main components: (1) intent inference, which obtains confidence in humans by inferring the intentions of human behaviors and (2) arbitration control, which makes decisions about how to fuse human and machine behaviors through the preceding confidence judgments about human intentions. To infer the intent of the irrational human player, confidence in the human decision is established. To provide a judgmental basis for subsequent arbitration of shared control, confidence in human decision-making can be expressed as the difference between the probability of the current belief and the probability of the smallest. Inspired by the article [34], the computation method for confidence is proposed as follows:

$$c(t) = \mathcal{P}(b_h^*) - \min_{b_h \in B \setminus b_h^*} \mathcal{P}(b_h) \tag{48}$$

where b_h^* is the most probable human behavior, $\boldsymbol{B}$ is the set of all possible human behaviors, and $\mathcal{P}(b_h)$ is the probability of human behavior b_h. By calculating the confidence $c(t)$, the shared control system can determine the level of human intelligence and make decisions about how to blend human and machine behaviors with the linear arbitration (8).

5.2 | Arbitration and Shared Control Paradigm

In shared control, the final input to the control system is usually a combination of human behavioral input u_h and machine control input u_m. In this article, the machine and the human share the same goal of stabilizing a nonlinear affine input system, and the reward functions of the two intelligences are defined to be similar in preliminary. Through an arbitration function based on intentional inference, the shared control system allocates the

ALGORITHM 1 | ADP-Based Level-k Human–Machine Cooperation.

```
Require:
Initial state $x_0$
Input gain matrix $R_i$
Learning rate $a_i^j$ for $i \in \{h, m\}$
Maximum level $k_m$
Ensure:
Bounded rational human policy distribution $\mathcal{P}(\mathcal{U}_h)$
Optimal machine policy $\mathcal{U}_m$
for $k = 0, \dots, k_m$ do                                  ▷ Learn level-$k$ policies
    for $i \in \{h, m\}$ do
        Initialize critic weights $\hat{W}_i^k$
        Set cooperator policy $u_{i'}^{k-1}$                 ▷ Previous level policy
        Learn optimal policy $u_i^k$ via (39)
        Update transformed system (15)
    end for
end for
for $k = 0, \dots, k_m$ do                                  ▷ Model human behavior
    Interact with machine policy $u_m^k$
    Calculate probability $\mathcal{P}(\mathcal{U}_h)$ via (47)
end for
Apply modeled human behavior $\mathcal{U}_h$
Learn optimal machine policy $\mathcal{U}_m$ via (39)
Update transformed system (15)
```

autonomy of the human and the machine, the following blending control input is proposed:

$$\Gamma\left(u_h(t), u_m(t)\right) \triangleq (1-\alpha) \cdot u_h(t) + \alpha \cdot u_m(t) \tag{49}$$

where $\alpha \in \left[0, \theta_3\right]$ is the adaptive arbitration function designed in (8), which allocates the authority of the control input between the human and the machine agent.

Because the Jacobian matrix of the human and the machine are not necessarily identical to obtain a shared control input that can be adapted to different dynamics, we reconstruct the above equation (49) to the following vector form:

$$\beta\left(u_h(t), u_m(t)\right) \triangleq [(1-\alpha)u_h(t),\ \alpha u_m(t)]^T \tag{50}$$

With the blending Jacobian matrix $g_{blend} = [g_h, g_m]^T$ from Equation (2), the dynamics of the human–machine system could be rewritten as the product of the blending matrix g_{blend} and the vector-formed shared control input $\beta\left(u_h(t), u_m(t)\right)$:

$$\begin{aligned} \dot{x} &= f(x) + g_{blend}(x)\beta\left(u_h(t), u_m(t)\right), \\ \text{s. t. } & u_m(t) = u_m^k(x) \\ & u_h(t) \sim \mathcal{P}(\mathcal{U}_h = \hat{u}_h^k) \end{aligned} \tag{51}$$

where $u_m^k(x)$ is the level-k machine policy derived from (37), $\mathcal{P}$ is the probabilistic bounded rational human policy calculated by (47). The confidence $c(t)$ is obtained by (48), and the arbitration function α is calculated by (8).

The developed shared control framework is able to consider both impact of human participants and autonomous machines. Through constructing bounded rationality of human thinking, a probabilistic distribution of intent inference is utilized to simulate the irrationality of human. The arbitration function is used to evaluate the confidence of the human and the machine, where the higher the confidence of the human, the lower the confidence of the autonomous machine. By evaluating and combining the input of both human and machine, the system is able to maintain in the safe state space while one participant is safety-awareness and the other is not. These mechanisms ensure that the shared control framework is able to achieve control assignments effectively and safely.

Remark 11. The shared control framework is designed to ensure the safety of the overall system while one participant is not safety-awareness. The shared control framework is able to achieve control assignments effectively and safely by evaluating and combining the input of both human and machine. In this article, the inputs of human is modeled as a probabilistic distribution of level-k rationality, in which the sub-optimality of human behavior is considered and guaranteed by Theorem 1. Then the arbitration function can be seen as a confidence judgment of two cooperative suboptimal agent's behavior (or control inputs). The suboptimality of the shared control system could be guaranteed through a proof similar to Theorem 1 with some minor modifications.

Remark 12. Note that compared with the ADP algorithms [14, 46, 49], the proposed shared control framework that integrates the level-k rationality model to capture the bounded rationality and uncertainty of human agents. Incorporating the direct shared control framework with the level-k rationality model is a novel and effective way to model the human irrational behavior in the shared control system. The proposed shared control framework is able to ensure the safety of the overall system while one participant is not safety-awareness. As for the results comparison with the existing method [14], literature [14] mainly focuses on the safe ADP stabilization control with two equal agents, which is different from the shared control framework proposed in this article. Our work focus on the cooperative control of human–machine system under irrationality of human behavior, which is a new and important research direction in the field of human–machine cooperative control.

Remark 13. In practice, standard system identification methods (e.g., physical-informed model [50] or Koopman operator [51]) can be used to estimate system dynamics from empirical data. Similarly, human behavior modeling can use supervised learning, inverse reinforcement learning [52], or Bayesian-based cognitive modeling [53] to approximate human input under diverse cognitive conditions. Nonetheless, uncertainties from unmodeled dynamics, measurement noise, and variability in human decision-making are unavoidable. Robust or adaptive control methods (e.g., online critic weight updates) can mitigate these uncertainties, preserving stability and performance despite modeling errors. Moreover, interval-based techniques or probabilistic bounds can quantify and manage uncertainty, ensuring safety and reliability in real-world shared control settings.

The shared control framework that blends the human–machine inputs is constructed in this section. In the next section, simulations are carried out for verification.

6 | Simulation Results

To demonstrate the effect of the proposed algorithm, simulations with different settings are implemented. Subsection 6.2 is conducted to show the learning process of the Nash equilibrium policies $\left(u_h^\star, u_m^\star\right)$ from problem 1, while satisfying $x \in \mathcal{O}$. Subsection 6.3 is set to testify the developed shared control framework from Section 5, which is designed to ensure overall system state safety while one participant is not safety-awareness.

6.1 | System Setup

To simplify the subsequent experimental design, and facilitate the comparison between different methods, consider the following nonlinear affine-input system from [14, 51, 54], which is a classical nonlinear system with two states and two inputs:

$$\begin{cases} \dot{x}_1 = x_2, \\ \dot{x}_2 = -x_2 - \frac{x_1}{2} + \frac{x_2(\cos(2x_1)+2)^2}{4} + \frac{x_2(\sin(2x_1)+2)^2}{4} \\ \qquad + (\sin(4x_1^2) + 2)u_h + (\sin(4x_1^2) + 2)(x)u_m \end{cases} \tag{52}$$

where $x(t) \in \mathbb{R}^2$ is the original state, $u_i \in \mathbb{R}, i = h, m$ is the policy of the human and machine player. For the considered nonlinear system (52), we verify that Assumptions 1–6 are satisfied: (1) The system dynamics are continuously differentiable and Lipschitz continuous in the operating region. (2) The system has an equilibrium point at the origin when both inputs are zero. (3) The control coefficients $g_1(x)$ and $g_2(x)$ are bounded and non-zero in the operating region. These properties ensure the theoretical results are applicable to this practical example.

To stabilize the original system (52), the objective of our proposed controller is to guarantee that the state $x(t)$ converges to zero while making sure that the state does not move out of the arbitrary safe boundary, namely $x \in \mathcal{O}$, we give the following exact numerical form expressed as

$$\mathcal{O} = \{(x_1, x_2) | x_i \in \left(a_i, A_i\right), \forall i \in \{1, 2\}\} \tag{53}$$

where $a_1 = -1.3, a_2 = -3.1, A_1 = 0.5$, and $A_2 = 0.5$.

The initial state is selected as $x_0 = [-1, -3]$. The learning rates for the human and machine player are selected as $a_1 = 3, a_2 = 5$, respectively, Using transformation (15), the original state x is transformed into the transformed system state s. Accordingly, the basis function is designed as $\phi_i^j(s) = [s_1^2, s_1 s_2, s_2^2]^T$. and the weights are initialized as $\hat{W}_i^j\left(t_0\right) = [1.5, 1.5, 1.5]^T,\ i = h, m,\ j = 1, \dots, k_m$. The cooperative reward function is defined as

$$r\left(s, u_h, u_m\right) = M(s) + \sum_{j \in \{h,m\}} u_j^T R_j u_j \tag{54}$$

where $M(s) = s^T s$, $R_h = 2I_2$ and $R_m = I_2$. The parameter of the arbitration function is $\theta_1 = 0.1$, $\theta_2 = 0.9$ and $\theta_3 = 0.7$.

6.2 | Example Study 1: Learning Process of Nash Equilibrium Policies

In this simulation, the rationality of the human and machine is up to level-5, which is smart enough to imitate various human behavior. After the learning procedure, the proposed human impact modeling algorithm will be utilized to model a bounded rational human policy map.

The learning process of the optimal machine critic weights is shown in Figure 3, which is obtained by interacting with the modeled human impact in the human–machine cooperative game. The critic network weights of the absolutely rational human are also shown in Figure 4, and they are all eventually convergent. Figure 5 illustrates the critic network weights of human behaviors at different intelligence levels (level-1 to 5). The norm of the critic weights of the human behaviors is shown in Figure 6, in which all the weights are convergent. Note that to save the load of online learning multiple control policies, previous research found that k up to 5 is enough to simulate uncertainty behavior of human-kinds [35, 55], when the level of human rationality is higher than 5, the human behavior is close to the optimal response, which is not necessary to be considered in the shared

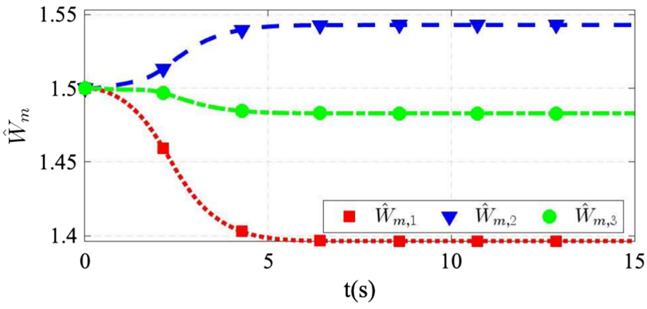


FIGURE 3 | Critic weights of the optimal machine.

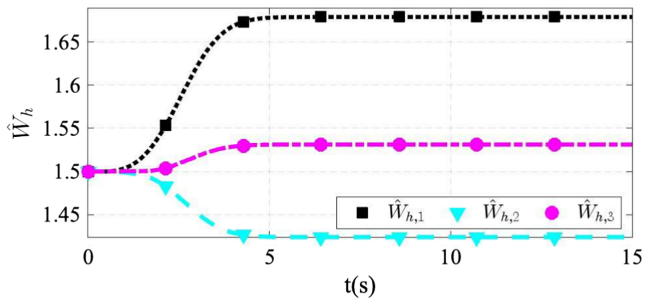


FIGURE 4 | Critic weights of the rational human.

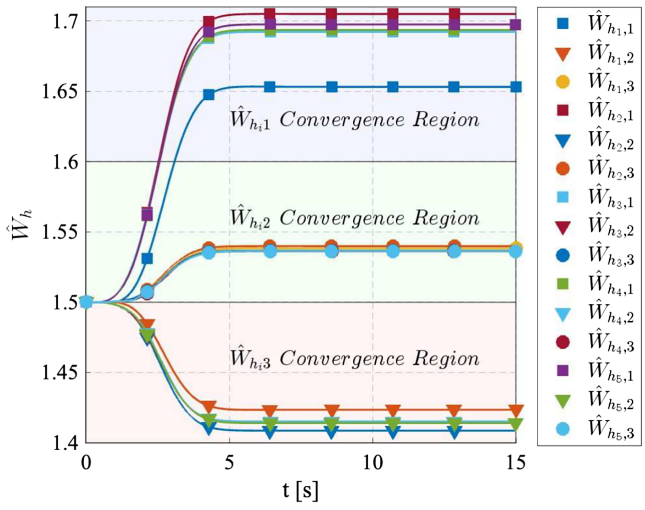


FIGURE 5 | Critic weights of level-k human.

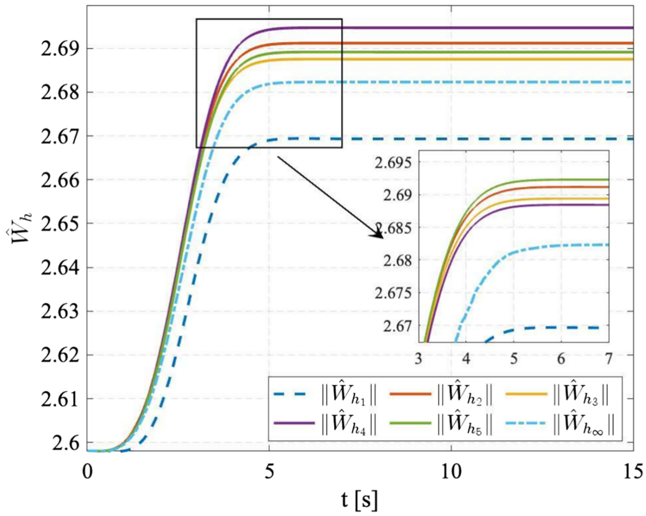


FIGURE 6 | Norm of level-k human critic weights.

control framework. It shows that the weight $\hat{W}_{h_1,1}$ of level-1 converges to 1.65, but the weights $\hat{W}_{h_1,i}$ of level-2 to 5 converge to the interval of 1.68–1.71, which are close to the weight of absolute rationality.

The probabilistic distribution of modeled human behavior is presented in Figure 7, including intelligence from level-1 to level-5. The policy of level-3 has the highest probability, while level-1 has the lowest probability and the other levels have the highest probability. The probability of the other levels is about the same, and such distribution is consistent with the intuition of various human actions.

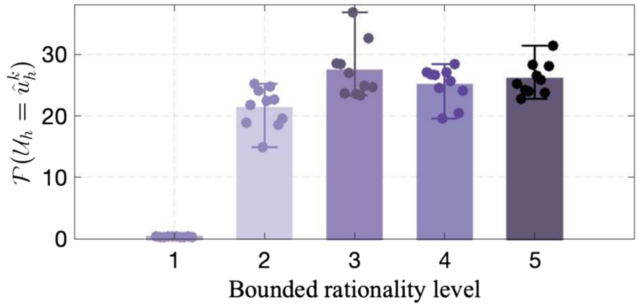


FIGURE 7 | Distribution of modeled human impact.

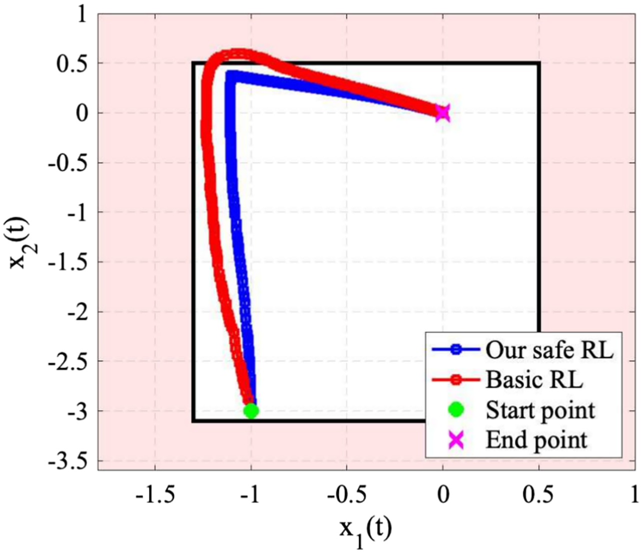


FIGURE 8 | State trajectories comparison.

The main result is presented in Figure 8, where the state trajectories of the transformed system (**Our safe RL approach**) and original non-transformed system (**Basic RL approach**) are presented. The "Basic RL approach" refers to the standard ADP method presented in [54], which provides a nonzero-sum game RL-based baseline for comparison with our framework. The detailed learning parameters and algorithmic setup are consistent with the method described in [54]. The trajectory of the non-transformed system is reaching the safe boundary of rectangle state constraint $\mathcal{O}$, which may cause great damage to the human. With the transformed system, human–machine cooperation finally stabilizes the state without violating the safety constraint.

6.3 | Example Study 2: Shared Control Framework Verification

In the last subsection, we present the simulation result of our safe RL method, which transformed both the human and the machine into a barrier-function-based transformation system to ensure safety. However, in most real-world cases, both the human and the machine cannot ensure they are both safety-aware at the same time, such as the machine is an automated robot unable to sense the safety constraints, whereas the human is a safety-awareness supervised operator, or the human is a unsafe-awareness operator, whereas the machine is a safety-awareness automated robot. It is worth exploring the shared control mechanism that ensures the whole system's safety when one participant is safety-deprived.

This subsection focuses on the simulation verification of the shared control architecture proposed in Section V. To discuss the various safety-critical scenarios and different intelligence levels that exist in human–machine collaboration, experiments are divided into two main categories: (1) safe human and unsafe machine (SHUM), unsafe machine (UM) and (2) unsafe human and safe machine (UHSM), safe machine (SM).

Case 1: Safe Human Unsafe Machine Shared Control.

The cases that involve the unsafe machine are categorized into the following three scenarios: (1) safe human and unsafe machine shared control; (2) unsafe machine alone; and (3) safe human alone. Note that the safe human is aware of the safety constraints, while the unsafe machine is not. For example, human agent is a safety-awareness supervised operator, while the machine agent is an automated robot unable to sense the safety constraints. It should be noted that the UM case is set for the comparison of the SHUM case, which is designed to verify the effectiveness of the shared control framework. In the UM case, the barrier function transformation is not applied to the machine agent, and the machine agent is trained to learn the optimal policy directly. The evolution of state $x = [x_1, x_2]$ with different cooperators is shown in Figure 9. The trajectories of the safe human involved are shown in Figure 9a and b, which do not overstep the limit. However, the trajectory of the system state under the operation of the unsafe machine alone is shown in Figure 9c, which has the state x_2 crossing the upper limit of $A_2 = 0.5$ and the state x_1 almost touching the boundary limit of $a_1 = -1.3$. The detailed evolution of state x of three scenarios is shown in Figure 10. Figure 10a illustrates the evolution of state x under the cooperative shared control of the safe human and the unsafe machine. Figure 10b shows the state trajectory under the action of the safe level-1 human. Figure 10c shows the state trajectory under the action of the unsafe machine, which exceeds the safe state limit.

The comparison of case SHUM and UM is shown in Figures 11 and 12, the initial state is normally distributed on $x'_0 \sim N(x_0, e^2)$. A total of 30 tests were conducted. Figure 11 illustrates the evolution curves of state x_1 under the cooperative shared control of the safe human and the unsafe machine. The blue line is obtained with the unsafe machine acting alone. The green line is obtained under the joint action of the safe human and unsafe machine. The state of unsafe machines almost hits the limit, and their oscillations are larger compared with those of the safe humans involved. The above indications suggest that with the action of unsafe machines, the system is likely to be in an unsafe situation, although faster state convergence can be achieved. However,

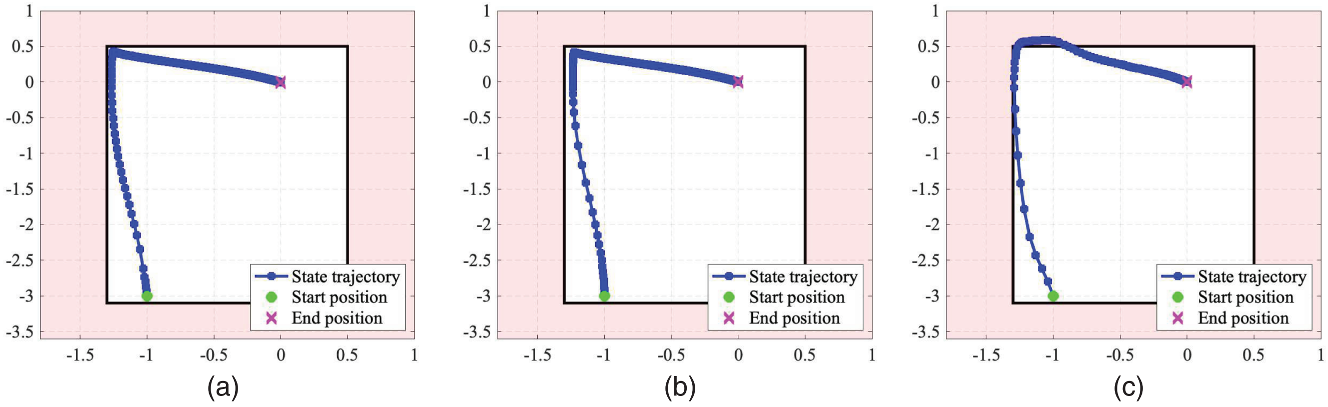


FIGURE 9 | State trajectory of (a) safe human and unsafe machine; (b) safe level-1 human; and (c) unsafe machine.

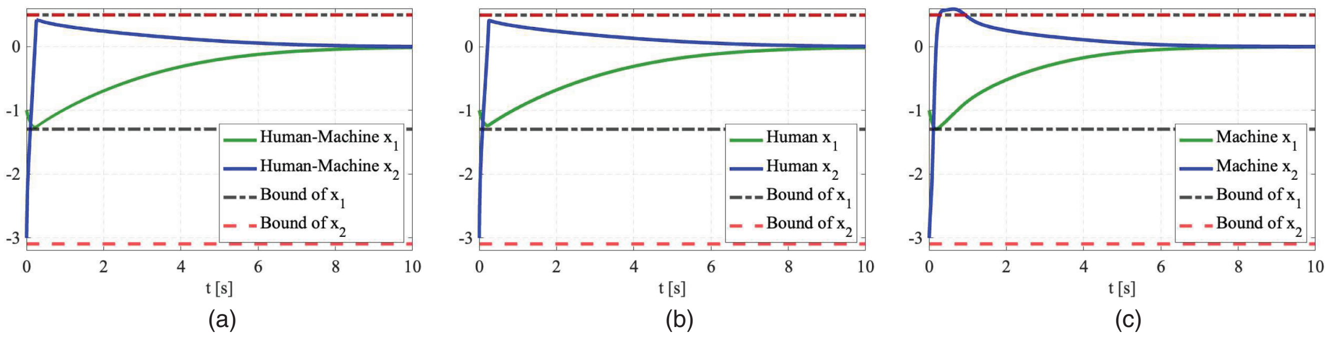


FIGURE 10 | The evolution of state x in (a) safe human and unsafe machine; (b) safe level-1 human; and (c) unsafe machine.

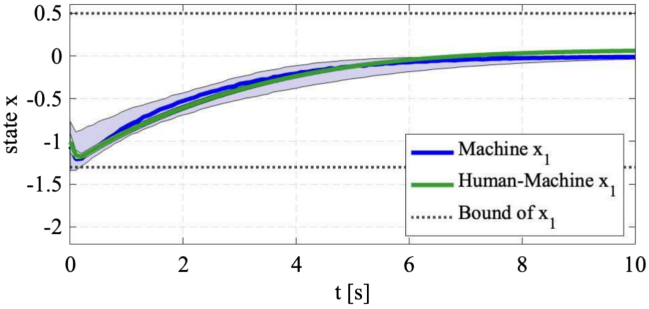


FIGURE 11 | The evolution of state x_1.

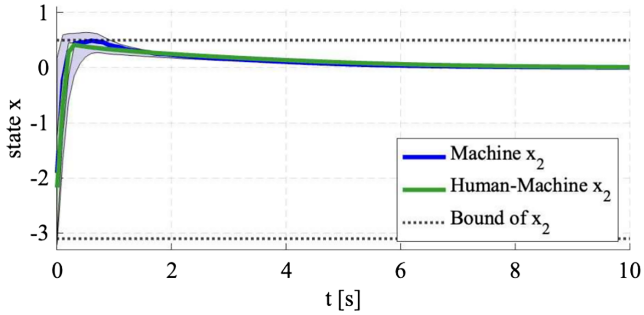


FIGURE 12 | The evolution of state x_2.

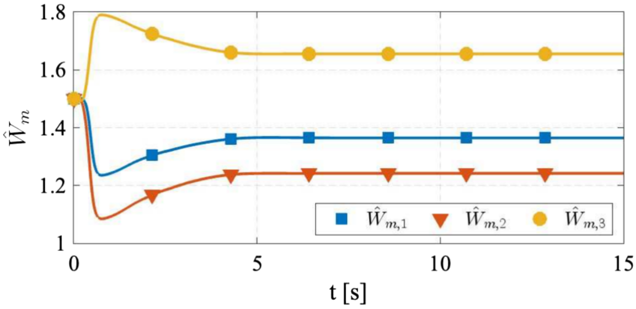


FIGURE 13 | Critic weights of machine in the case of SHUM.

with the concerted action of safe humans, the overall system is in a state far from the safe state boundary. Figure 12 shows how the state x_2 changes for different combinations of safe humans and unsafe machines. The blue curve is for the unsafe machine, which crosses the safe upper limit of state x_2 around $t = 0.25$ s. The green curve involves the safe-aware human and none of them cross any safety limit. In this case, the state under the action of a safe human alone reaches the maximum first, and the state under the act of human–machine cooperation reaches the maximum second. The maximum point of the unsafe machine corresponds to the longest time.

The detailed evolution of the unsafe machine critic network weights in the case of SHUM and UM are illustrated in Figures 13, 14.

Case 2: Unsafe Human Safe Machine Shared Control.

The next simulation is about the interaction between the unsafe human and safe machine. Safe machines aim to secure human–machine collaboration with the help of the safe machine. For example, human agent is a novice unsafe-aware operator, while the machine agent is a safe-aware automation. It should be noted that the unsafe level-1 human case is set for the comparison of the UHSM case, which is designed to verify the effectiveness of the shared control framework. In the unsafe level-1 human case, the barrier function transformation is not applied to the agent, and the agent is trained to learn the optimal policy directly.

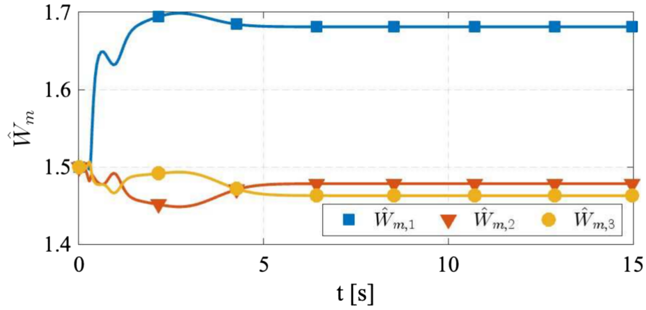
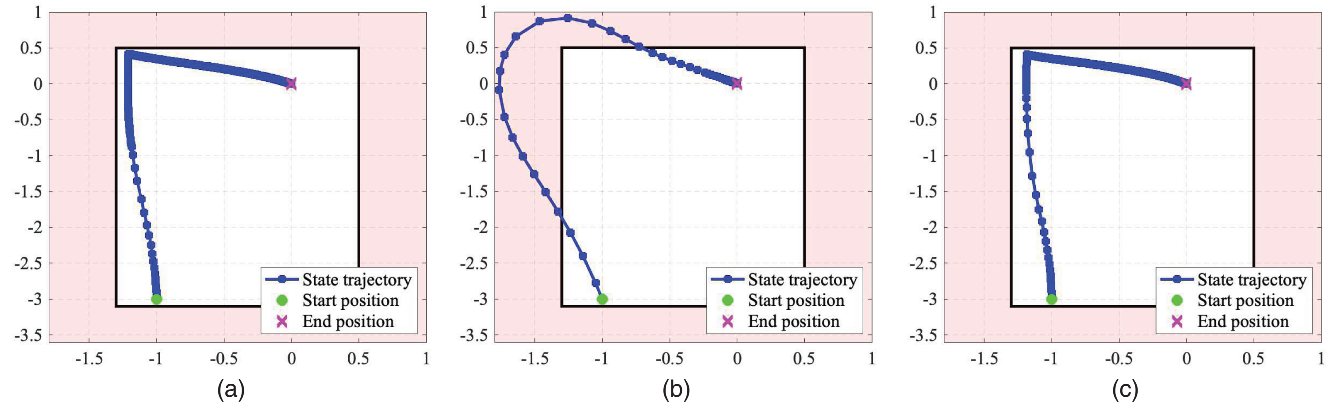


FIGURE 14 | Critic weights of machine in the case of UM.

Figure 15 illustrates the state trajectories for three scenarios with different combinations of the safe machine and unsafe human. Figure 15a and c shows the state trajectories of the unsafe human-safe machine shared control, the safe machine control. Figure 15b shows the state trajectory under the action of the unsafe level-1 human, which greatly exceeds the safe state limit. However, the other two state trajectories with the safe machine involved do not exceed the safe state limit, implying that the collaboration between the safe machine and the unsafe human successfully secures the system. The detailed evolution of state x of three scenarios is shown in Figure 16. Figure 16a illustrates the evolution of state x under the cooperative shared control of the unsafe human and the safe machine. Figure 16b shows the state trajectory under the action of the unsafe level-1 human, which greatly exceeds the safe state limit. Figure 16c shows the state trajectory under the action of the safe machine.

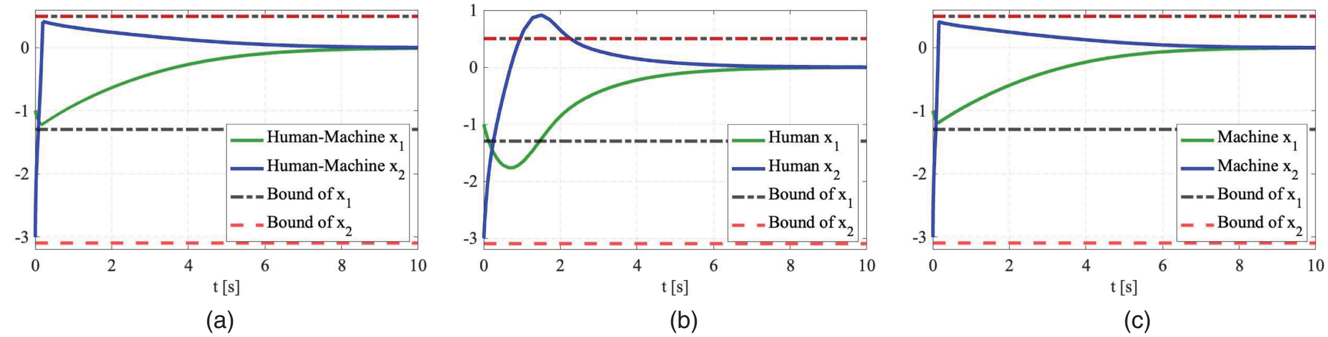
Figures 17 and 18 show the evolution for state x_1 and x_2, respectively. The initial state is normally distributed on $x_0' \sim N(x_0, e^2)$. A total of 30 tests were conducted. The action of the unsafe human generates the state trajectory in blue, and its maximum value exceeds the safety limit. The cooperation between the unsafe human and the safe machine generates the green state trajectory. Through the intervention of the safe machine, the two intelligences achieve safe cooperation and stabilization control of the system. Both the figures show that the state will exceed the safety limit under the action of the unsafe person alone. The state $x = [x_1, x_2]$ is kept within the safety limit with the action of the safe machine. This indicates that the safe machine guarantees the safety of the human–machine shared control. Figures 19 and 20 illustrate the evolution of the critic network weights with the involvement of the safe machine.

In two different experimental cases of subsection 6.3, the developed shared control framework is testified to guarantee the safety of overall cooperative system state. For the extreme conditions, in which one participant like human being is not able to be cautious



(a) (b) (c)

FIGURE 15 | State trajectories of (a) unsafe human and safe machine; (b) unsafe level-1 human; and (c) safe machine.



(a) (b) (c)

FIGURE 16 | State trajectories of (a) unsafe human and safe machine; (b) unsafe level-1 human; and (c) safe machine.

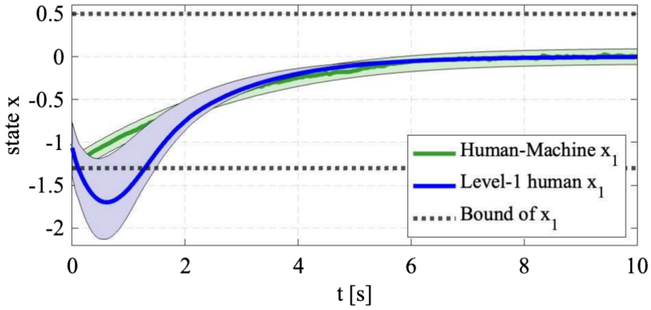


FIGURE 17 | The evolution of state x_1.

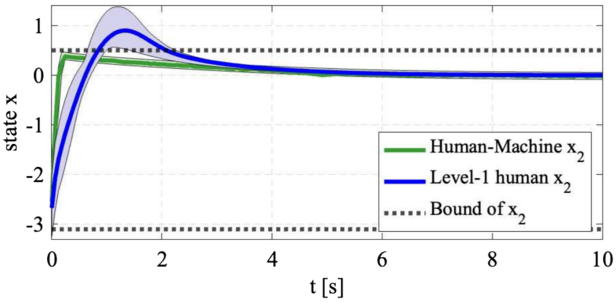


FIGURE 18 | The evolution of state x_2.

enough to conduct assignment safely, the proposed cooperative shared control paradigm is able to reduce the impact of irrational behavior and conduct safe behavior from the other participants.

6.4 | Example Study 3: Cooperative Quadrotor Tracking Control

6.4.1 | Quadrotor System Dynamics and Simulation Setup

The simulation is conducted on a customized 3-DOF quadrotor platform to validate the practical effectiveness of the proposed level-k human–machine shared control framework. The system dynamics configuration is shown in Figure 21, and the system parameters are listed in Table 1. For the 3-DOF hover system dynamics, the state vector is defined as $x = [\phi, \theta, \psi, \dot{\phi}, \dot{\theta}, \dot{\psi}]^\top$ representing the roll-pitch-yaw Euler angles and their rates, the output vector is $y = [\phi, \theta, \psi]^\top$ containing the measured angles, and the control input vector is $u = [V_f, V_b, V_r, V_l]^\top$ consisting of the

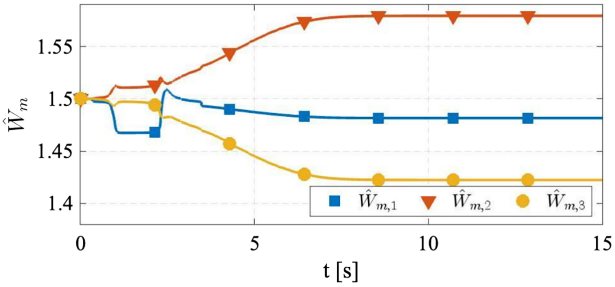


FIGURE 19 | Critic weights of machine in the case of UHSM.

front, back, right, and left motor voltages. The system dynamics can be described by the following state-space model:

$$\begin{aligned}\dot{x} &= Ax + Bu \\ &= Ax + B\left[\alpha u_h + (1-\alpha) u_m\right]\end{aligned} \tag{55}$$

where u_h is the simulated human control input calculated by the proposed level-k human–machine shared control framework, u_m is the machine control input calculated by the ADP-based optimal control method, and $\alpha \in [0,1]$ is the arbitration parameter that dynamically adjusts the human–machine control ratio. and the system matrices A, B are defined as:

$$A = \begin{bmatrix} 0_{3\times3} & I_{3\times3} \\ 0_{3\times3} & 0_{3\times3} \end{bmatrix}, \quad B = \begin{bmatrix} & 0_{3\times4} & & \\ -k_t & -k_t & k_t & k_t \\ \ell k_f & -\ell k_f & 0 & 0 \\ 0 & 0 & \ell k_f & -\ell k_f \end{bmatrix}$$

which satisfies the above Assumptions 1 and 2 for the proposed safe RL framework.

Remark 14 (Quadrotor System Dynamics). Regarding the system dynamics matrices A and B defined above: Matrix A represents the linear state transformation, where the upper-right identity matrix $I_{3\times3}$ captures the natural relationship between angular velocities and angle changes. Matrix B describes the control input mapping, where its structure reflects fundamental quadrotor dynamics: (1) The zero elements in the first three rows indicate that motor inputs affect angles (ϕ, θ, ψ) indirectly through their derivatives, following the principle that forces and torques produce accelerations rather than direct position change. (2) The lower three rows contain the thrust-to-torque conversion coefficients k_t and k_f, which map motor voltages to the resulting roll, pitch, and yaw moments. (3) This decoupled structure deliberately separates attitude dynamics from translational motion, a standard practice in quadrotor control when attitude stabilization is the primary focus. This formulation enables precise attitude control while maintaining mathematical tractability for the safe learning framework.

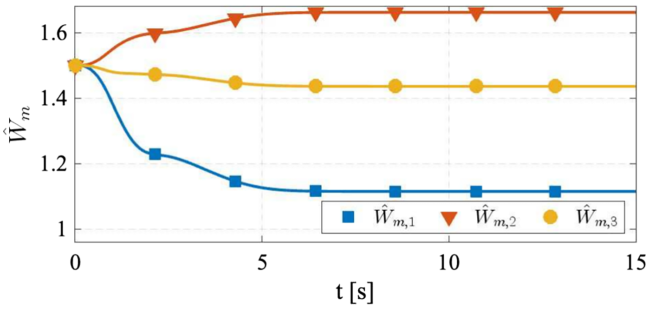


FIGURE 20 | Critic weights of machine in the case of SM.

The desired reference trajectory is designed as a smooth sinusoidal function with varying amplitudes and frequencies:

$$\begin{cases} \phi_d(t) &= A_\phi \sin(\omega_\phi t) \\ \theta_d(t) &= A_\theta \sin(\omega_\theta t) \\ \psi_d(t) &= A_\psi \sin(\omega_\psi t) \end{cases}, \quad \begin{cases} \dot{\phi}_d(t) &= A_\phi \omega_\phi \cos(\omega_\phi t) \\ \dot{\theta}_d(t) &= A_\theta \omega_\theta \cos(\omega_\theta t) \\ \dot{\psi}_d(t) &= A_\psi \omega_\psi \cos(\omega_\psi t) \end{cases}$$

where the trajectory parameters are selected as: $A_\phi = 0.2$ rad, $A_\theta = 0.15$ rad, $A_\psi = 0.3$ rad for amplitudes, and $\omega_\phi = 0.25$ rad/s, $\omega_\theta = 0.25$ rad/s, $\omega_\psi = 0.15$ rad/s for frequencies. To enhance computational efficiency and learning convergence, the following neural network basis functions are adopted:

$$\phi_i^j(s) = \left[s_1^2, s_2^2, s_1 s_2, s_3^2, s_1 s_3, s_2 s_3, s_4^2, s_5^2, s_6^2, s_1 s_4, s_2 s_5, s_3 s_6\right]^\top$$

where $i = h, m$ denotes human/machine agents, $j = 1, \dots, k_m$ represents rationality levels. The network weights are initialized as $\hat{W}_{ci}^j = 10 + \mathcal{N}(0,1)$ to ensure proper exploration, The learning rate is set as $\alpha = 0.001$ for the critic NNs. The optimal value function of sub-optimal levels of human and machine agents are approximated by the critic NNs in (35), while its corresponding control policy is approximated by (37). $s_i(e_i)$ $(i = 1, \dots, 6)$ are the transformed tracking errors of original tracking errors e_i $(i = 1, \dots, 6)$ with the transformation defined as in (14). The safety constraints are set as $a_i = -2$ rad (or rad/s) and $A_i = 2$ rad (or rad/s) for all angles and angular rates. The upper bound

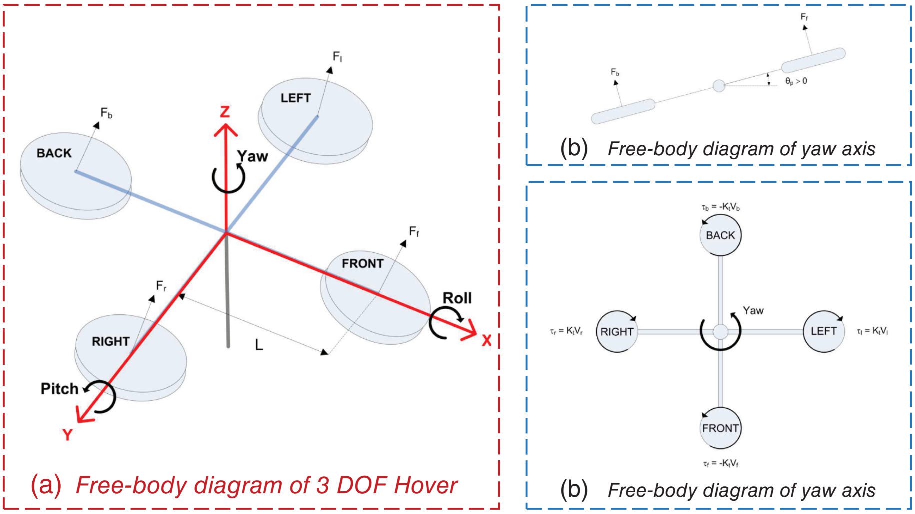


FIGURE 21 | Dynamics of the quadrotor tracking system.

TABLE 1 | Parameters of the quadrotor system.

Basic properties			**Motor characteristics**		
g	Gravitational acceleration	9.81 m/s^2	R_m	Motor armature resistance	0.83 Ω
m_{hover}	Total mass of hover system	2.85 kg	K_{t_m}	Motor torque constant	0.0182 N m A^{-}1
m_{prop}	Mass of propeller assembly	0.7125 kg	J_m	Motor rotor inertia	1.91 × 10^6 kgm^2
ℓ	Arm length (pivot to motor)	0.197 m			
Propulsion parameters			**Inertial properties**		
K_f	Thrust coefficient	0.1188 N V^{-}1	J_y	Yaw inertia	0.1116 kgm^2
K_t	Torque coefficient	0.0036 N m V^{-1}	J_p	Pitch inertia	0.0558 kgm^2
$J_{eq_{prop}}$	Equivalent propeller inertia	0.0279 kgm^2	J_r	Roll inertia	0.0558 kgm^2

of the control input is set as $V_{\max} = 10$ V to ensure the safety of the physical system. The simulation time is set as $T = 25$ s with a sampling time of $\Delta t = 0.001$ s. The level of human rationality is up to level-5, and the learning process is conducted with the proposed shared control framework from Algorithm 1. The control objective is to achieve precise trajectory tracking while maintaining system stability and safety constraints under the human–machine shared control framework. The safety constraints are imposed on both angles and angular rates to protect the physical system.

Remark 15. In the quadrotor case study, the human–machine shared control framework is implemented through a dual-channel control architecture: (1) The human operator provides control inputs u_h which are directly mapped to motor commands $[V_f, V_b, V_r, V_l]$. These inputs exhibit varying degrees of rationality (Level-0 through Level-5) representing different levels of control expertise calculated by the Level-k and Human irrationality modeling method given in Algorithm 1, and serve as real-time cooperative control signals for the machine controller. (2) The machine controller generates control inputs u_m by computing optimal safe controls that satisfy barrier function constraints, actively compensating for potentially unsafe human commands while optimizing system performance through ADP. The final control synthesis follows $u = \alpha u_h + (1 - \alpha)u_m$, where α is dynamically adjusted based on the angular deviation between human and optimal control vectors. Borrow the detailed shared parameter calculation method in [44, 51], this adjustment mechanism is defined as:

$$\alpha = \begin{cases} 0, & \text{if } \eta \geq \frac{2\pi}{3} \\ 1, & \text{if } \eta \leq \frac{\pi}{2} \\ \frac{\eta - \frac{2\pi}{3}}{\frac{\pi}{2} - \frac{2\pi}{3}}, & \text{otherwise} \end{cases} \tag{56}$$

where η denotes the angle between human and machine input vectors. This mechanism ensures smooth transitions between human and machine control while maintaining robust system safety guarantees.

Remark 16 (Constant Rationality Level Assumption). We acknowledge that the current work assumes a constant human rationality level throughout the control process. This simplifying assumption allows us to establish foundational theoretical guarantees while maintaining analytical tractability. However, in real-world applications, human decision-making capabilities may evolve due to factors such as fatigue, learning effects, or varying cognitive load. Extending our framework to accommodate time-varying rationality levels represents an important direction for future research. This could involve developing adaptive mechanisms to estimate and respond to changes in human rationality in real-time, further enhancing the practical applicability of human–machine shared control systems.

Remark 17 (Practical Applicability and Safety Mechanism). The quadrotor shared control system could be improved from our approach and designed as a hierarchical structure to address practical scenarios. In this architecture, human operators provide high-level guidance based on their situational awareness and mission-level decision making, while the machine controller ensures safe and stable execution of these high-level commands. The system integrates both human expertise and autonomous capabilities through an adaptive sharing ratio $\alpha(x, t)$, which dynamically balances between human intent and safety constraints based on the assessed rationality level. This mechanism allows the controller to maintain system safety while maximizing human control authority when appropriate, as demonstrated in Case 1 of Example Study 2. The framework effectively combines human strategic decision-making with automated safeguards against constraint violations, making it particularly suitable for complex mission scenarios requiring both human insight and guaranteed safety properties.

6.4.2 | Simulation Results

The tracking performance of the quadrotor system is demonstrated in Figure 22, where all three Euler angles (ϕ, θ, ψ) closely track their reference trajectories with high precision. The control inputs shown in Figure 23 illustrate the smooth motor voltage adjustments that achieve stable system control while respecting actuator constraints. The tracking error components are further analyzed in Figures 24, 25, which demonstrate small bounded errors in both attitude angles and angular velocities. A comparison of normalized tracking errors between different control methods is provided in Figure 26, where our proposed level-k shared control framework shows superior performance. The evolution of neural network weights for the level-5 human and machine agents is presented in Figure 27, where $\hat{W}_h^5$ and $\hat{W}_m^5$ denote the critic weights of the human and machine

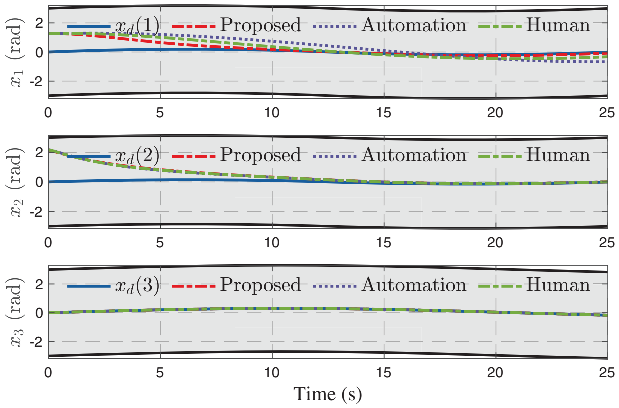


FIGURE 22 | Attitude tracking performance.

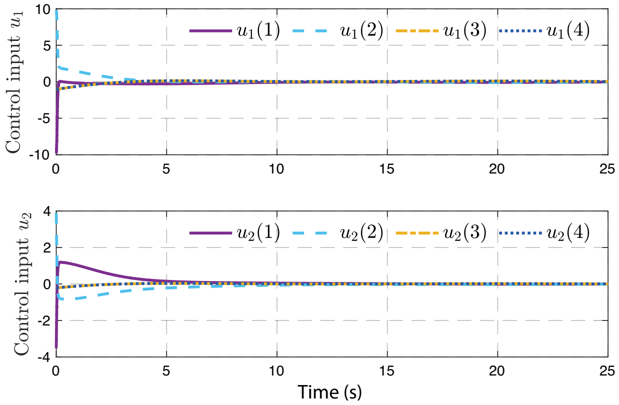


FIGURE 23 | Human input u_1 and machine input u_2.

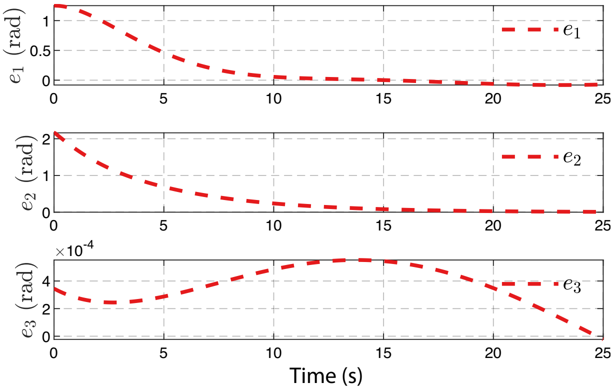


FIGURE 24 | Attitude tracking errors.

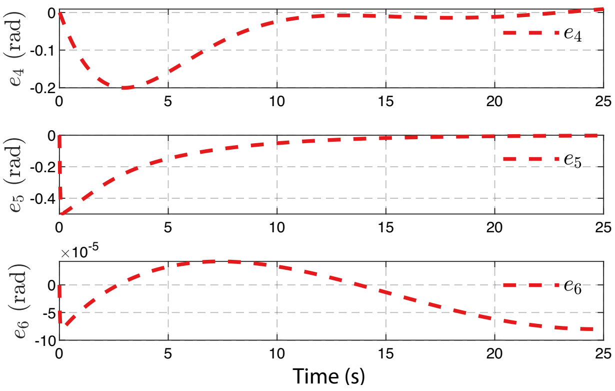


FIGURE 25 | Angular velocity tracking errors.

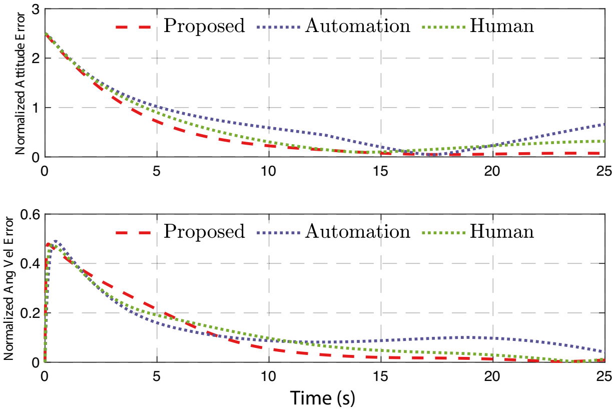


FIGURE 26 | Comparison of normalized tracking errors.

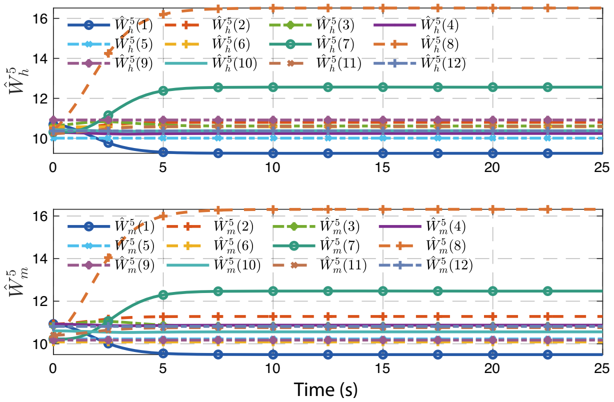


FIGURE 27 | NN weights of level-5 human and machine.

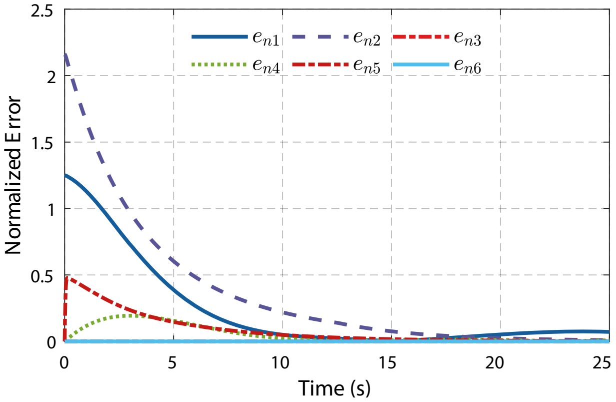


FIGURE 28 | Normalized tracking error evolution.

agents, respectively. The weights are updated iteratively during the learning process, and Figure 27 illustrates the convergence of the learning process. Figure 28 displays the overall normalized tracking error evolution, demonstrating that the proposed human–machine shared control framework effectively reduces tracking errors while maintaining system stability and safety constraints. These results validate that our proposed framework successfully achieves precise trajectory tracking while accommodating both human and machine inputs in a cooperative manner. The bounded tracking errors and smooth control signals indicate effective coordination between the safety-aware machine controller and the bounded rational human operator.

7 | Conclusions

The shared control of bounded rational human behavior with a cooperative machine is investigated in this research. Cooperation between humans and machines is an emerging subject in safety-critical system control, and it is necessary to guarantee human safety. Full state safety limitations are guaranteed by developing a barrier-function-based state transformation. To construct bounded rationality, a level-k thinking framework is developed. The ADP is used to obtain the controller from the level-k framework. A probabilistic distribution based on Softmax is used to model human behavior, simulating the uncertainty of human intelligence in the cooperative game. The control input from both the human and the machine is combined through a shared control framework to stabilize the system safely and effectively. The effectiveness of the proposed architecture is then tested through simulations, which show that not only the full state constraints and stabilization are guaranteed but also the shared control framework ensures system safety even when one of the participants is not safety-aware. Future research may expand our proposed method in different human–machine cooperation scenarios, such as human–quadcopter cooperative tracking control and human–robotic arm cooperative manipulation control.

Acknowledgments

This work was supported by the Science and Technology Project of Huaneng Group Headquarters under Grant HNKJ21-H53, and China Postdoctoral Science Foundation (Project ID: 2024M762602).

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

Data sharing not applicable to this article as no datasets were generated or analyzed during the current study.

References

1. D. P. Losey, C. G. McDonald, E. Battaglia, and M. K. O'Malley, "A Review of Intent Detection, Arbitration, and Communication Aspects of Shared Control for Physical Human–Robot Interaction," *Applied Mechanics Reviews* 70, no. 1 (2018): 010804.

2. P. Ye, X. Wang, W. Zheng, Q. Wei, and F.-Y. Wang, "Parallel Cognition: Hybrid Intelligence for Human-Machine Interaction and Management," *Frontiers of Information Technology & Electronic Engineering* 23, no. 12 (2022): 1765–1779.

3. Y. Yildiz, A. Agogino, and G. Brat, "Predicting Pilot Behavior in Medium-Scale Scenarios Using Game Theory and Reinforcement Learning," *Journal of Guidance, Control, and Dynamics* 37, no. 4 (2014): 1335–1343.

4. C. F. Camerer, T.-H. Ho, and J.-K. Chong, "A Cognitive Hierarchy Model of Games," *Quarterly Journal of Economics* 119, no. 3 (2004): 861–898.

5. Y. Zheng, H. Zhao, J. Zheng, C. He, and Z. Li, "Stackelberg-Game-Oriented Optimal Control for Bounded Constrained Mechanical Systems: A Fuzzy Evidence-Theoretic Approach," *IEEE Transactions on Fuzzy Systems* 30, no. 9 (2022): 3559–3573.

6. 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* (2025): 1–21.

7. J. Li, L. Yao, X. Xu, B. Cheng, and J. Ren, "Deep Reinforcement Learning for Pedestrian Collision Avoidance and Human-Machine Cooperative Driving," *Information Sciences* 532 (2020): 110–124.

8. 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): 1–13.

9. Y. Yang, K. G. Vamvoudakis, H. Modares, Y. Yin, and D. C. Wunsch, "Safe Intermittent Reinforcement Learning With Static and Dynamic Event Generators," *IEEE Transactions on Neural Networks and Learning Systems* 31, no. 12 (2020): 5441–5455.

10. J. Tan, S. Xue, H. Cao, and H. Li, "Nash Equilibrium Solution Based on Safety-Guarding Reinforcement Learning in Nonzero-Sum Game. Proceedings of the 2023 International Conference on Advanced Robotics and Mechatronics (ICARM), 630-635, July," 2023.

11. S. Liu, L. Liu, and Z. Yu, "Safe Reinforcement Learning for Discrete-Time Fully Cooperative Games With Partial State and Control Constraints Using Control Barrier Functions," *Neurocomputing* 517 (2023): 118–132.

12. R. Kamalapurkar, H. Dinh, S. Bhasin, and W. E. Dixon, "Approximate Optimal Trajectory Tracking for Continuous-Time Nonlinear Systems," *Automatica* 51 (2015): 40–48.

13. M. H. Cohen and C. Belta, "Safe Exploration in Model-Based Reinforcement Learning Using Control Barrier Functions," *Automatica* 147 (2023): 110684.

14. Y. Yang, K. G. Vamvoudakis, and H. Modares, "Safe Reinforcement Learning for Dynamical Games," *International Journal of Robust and Nonlinear Control* 30, no. 9 (2020): 3706–3726.

15. Z. Marvi and B. Kiumarsi, "Safe Reinforcement Learning: A Control Barrier Function Optimization Approach," *International Journal of Robust and Nonlinear Control* 31, no. 6 (2021): 1923–1940.

16. N.-M. T. Kokolakis and K. G. Vamvoudakis, "Safety-Aware Pursuit-Evasion Games in Unknown Environments Using Gaussian Processes and Finite-Time Convergent Reinforcement Learning," *IEEE Transactions on Neural Networks and Learning Systems* 35, no. 3 (2024): 3130–3143.

17. J. Tan, S. Xue, H. Li, H. Cao, and D. Li, "Safe Stabilization Control for Interconnected Virtual-Real Systems via Model-Based Reinforcement Learning. Proceedings of the 14th Asian Control Conference (ASCC), 605-610, July," (2024).

18. Y. Zhang, X. Liang, D. Li, et al., "Barrier Lyapunov Function-Based Safe Reinforcement Learning for Autonomous Vehicles With Optimized Backstepping," *IEEE Transactions on Neural Networks and Learning Systems* 35, no. 2 (2024): 2066–2080.

19. Y. Zhang, X. Liang, D. Li, et al., "Adaptive Safe Reinforcement Learning With Full-State Constraints and Constrained Adaptation for Autonomous Vehicles," *IEEE Transactions on Cybernetics* 54, no. 3 (2023): 1907–1920.

20. W. Jin, S. Mou, and G. J. Pappas, "Safe Pontryagin Differentiable Programming," *Advances in Neural Information Processing Systems* 34 (2021): 16034–16050.

21. R. Tian, L. Sun, A. Bajcsy, M. Tomizuka, and A. D. Dragan, "Safety Assurances for Human-Robot Interaction via Confidence-Aware Game-Theoretic Human Models. Proceedings of the 2022 International Conference on Robotics and Automation (ICRA), 11229–11235,May," 2022.

22. A. Kanellopoulos and K. G. Vamvoudakis, "Non-Equilibrium Dynamic Games and Cyber–Physical Security: A Cognitive Hierarchy Approach," *Systems & Control Letters* 125 (2019): 59–66.

23. K. G. Vamvoudakis, F. Fotiadis, A. Kanellopoulos, and N.-M. T. Kokolakis, “Nonequilibrium Dynamical Games: A Control Systems Perspective,” *Annual Reviews in Control* 53 (2022): 6–18.

24. N.-M. T. Kokolakis and K. G. Vamvoudakis, “Bounded Rational Dubins Vehicle Coordination for Target Tracking Using Reinforcement Learning,” *Automatica* 149 (2023): 110732.

25. N. Musavi, D. Onural, K. Gunes, and Y. Yildiz, “Unmanned Aircraft Systems Airspace Integration: A Game Theoretical Framework for Concept Evaluations,” *Journal of Guidance, Control, and Dynamics* 40, no. 1 (2017): 96–109.

26. C. O. Yaldiz and Y. Yildiz, “Driver Modeling Using Continuous Reasoning Levels: A Game Theoretical Approach. Proceedings of the 2022 IEEE 61st Conference on Decision and Control (CDC), 5068–5073, December,” 2022.

27. A. Perrusquía, “Human-Behavior Learning: A New Complementary Learning Perspective for Optimal Decision Making Controllers,” *Neurocomputing* 489 (2022): 157–166.

28. H. Peng, L. Chen, X. Yang, C. Huang, and Z. Hu, “Human-Like Decision Making for Autonomous Driving: A Noncooperative Game Theoretic Approach,” *IEEE Transactions on Intelligent Transportation Systems* 22, no. 4 (2021): 2076–2087.

29. N. Li, I. Kolmanovsky, A. Girard, and Y. Yildiz, “Game Theoretic Modeling of Vehicle Interactions at Unsignalized Intersections and Application to Autonomous Vehicle Control. Proceedings of the 2018 Annual American Control Conference (ACC), 3215-3220, June,” 2018.

30. R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. (MIT Press, 2018).

31. R. Tian, N. Li, I. Kolmanovsky, and A. Girard, “Beating Humans in a Penny-Matching Game by Leveraging Cognitive Hierarchy Theory and Bayesian Learning. Proceedings of the 2020 American Control Conference (ACC), 4652–4657, July,” 2020.

32. M. D. Lee, “How Cognitive Modeling can Benefit From Hierarchical Bayesian Models,” *Journal of Mathematical Psychology* 55, no. 1 (2011): 1–7.

33. S. Jain and B. Argall, “Probabilistic Human Intent Recognition for Shared Autonomy in Assistive Robotics,” *ACM Transactions on Human-Robot Interaction* 9, no. 1 (2020): 1–23.

34. Q. Zhang, Y. Kang, Y.-B. Zhao, P. Li, and S. You, “Traded Control of Human–Machine Systems for Sequential Decision-Making Based on Reinforcement Learning,” *IEEE Transactions on Artificial Intelligence* 3, no. 4 (2022): 553–566.

35. N. Li, D. W. Oyler, M. Zhang, Y. Yildiz, I. Kolmanovsky, and A. R. Girard, “Game Theoretic Modeling of Driver and Vehicle Interactions for Verification and Validation of Autonomous Vehicle Control Systems,” *IEEE Transactions on Control Systems Technology* 26, no. 5 (2018): 1782–1797.

36. M. Selvaggio, M. Cognetti, S. Nikolaidis, S. Ivaldi, and B. Siciliano, “Autonomy in Physical Human-Robot Interaction: A Brief Survey,” *IEEE Robotics and Automation Letters* 6, no. 4 (2021): 7989–7996.

37. W. Wang, X. Na, D. Cao, et al., “Decision-Making in Driver-Automation Shared Control: A Review and Perspectives,” *IEEE/CAA Journal of Automatica Sinica* 7, no. 5 (2020): 1289–1307.

38. A. D. Dragan and S. S. Srinivasa, “A Policy-Blending Formalism for Shared Control,” *International Journal of Robotics Research* 32, no. 7 (2013): 790–805.

39. A. Broad, I. Abraham, T. Murphey, and B. Argall, “Data-Driven Koopman Operators for Model-Based Shared Control of Human–Machine Systems,” *International Journal of Robotics Research* 39, no. 9 (2020): 1178–1195.

40. X. Liu, S. S. Ge, F. Zhao, and X. Mei, “Optimized Impedance Adaptation of Robot Manipulator Interacting With Unknown Environment,” *IEEE Transactions on Control Systems Technology* 29, no. 1 (2021): 411–419.

41. H. Modares, I. Ranatunga, F. L. Lewis, and D. O. Popa, “Optimized Assistive Human–Robot Interaction Using Reinforcement Learning,” *IEEE Transactions on Cybernetics* 46, no. 3 (2016): 655–667.

42. M. Huang, Z.-P. Jiang, and K. Ozbay, “Learning-Based Adaptive Optimal Control for Connected Vehicles in Mixed Traffic: Robustness to Driver Reaction Time,” *IEEE Transactions on Cybernetics* 52, no. 6 (2022): 5267–5277.

43. Y. Li, K. P. Tee, W. L. Chan, R. Yan, Y. Chua, and D. K. Limbu, “Continuous Role Adaptation for Human–Robot Shared Control,” *IEEE Transactions on Robotics* 31, no. 3 (2015): 672–681.

44. J. Tan, S. Xue, Z. Guo, H. Li, H. Cao, and B. Chen, “Data-Driven Optimal Shared Control of Unmanned Aerial Vehicles,” *Neurocomputing* 622 (2025): 129428.

45. X. Cui, J. Chen, B. Wang, and S. Xu, “Off-Policy Algorithm Based Hierarchical Optimal Control for Completely Unknown Dynamic Systems,” *Neurocomputing* 488 (2022): 669–680.

46. Y. Yang, Y. Pan, C.-Z. Xu, and D. C. Wunsch, “Hamiltonian-Driven Adaptive Dynamic Programming With Efficient Experience Replay,” *IEEE Transactions on Neural Networks and Learning Systems* 35, no. 3 (2022): 3278–3290.

47. K. Nguyen, V. T. Dang, D. D. Pham, and P. N. Dao, “Formation Control Scheme With Reinforcement Learning Strategy for a Group of Multiple Surface Vehicles,” *International Journal of Robust and Nonlinear Control* 34, no. 3 (2024): 2252–2279.

48. W. B. Powell, *Approximate Dynamic Programming: Solving the Curses of Dimensionality. Wiley Series in Probability and Statistics*, 2nd ed. (Wiley, 2011).

49. 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* 70, no. 2 (2025): 1304–1311.

50. A. Perrusquía and W. Guo, “Uncovering Reward Goals in Distributed Drone Swarms Using Physics-Informed Multiagent Inverse Reinforcement Learning,” *IEEE Transactions on Cybernetics* 55, no. 1 (2025): 14–23.

51. J. Tan, S. Xue, H. Cao, and S. S. Ge, “Human-AI Interactive Optimized Shared Control,” *Journal of Automation and Intelligence* (2025): S2848855425000024.

52. A. Perrusquía and W. Guo, “Drone’s Objective Inference Using Policy Error Inverse Reinforcement Learning,” *IEEE Transactions on Neural Networks and Learning Systems* 36, no. 1 (2025): 1329–1340.

53. C. L. Baker, J. B. Tenenbaum, and R. R. Saxe, Bayesian Models of Human Action Understanding.

54. K. G. Vamvoudakis and F. L. Lewis, “Multi-Player Non-Zero-Sum Games: Online Adaptive Learning Solution of Coupled Hamilton–Jacobi Equations,” *Automatica* 47, no. 8 (2011): 1556–1569.

55. R. Tian, L. Sun, M. Tomizuka, and D. Isele, “Anytime Game-Theoretic Planning with Active Reasoning About Humans’ Latent States for Human-Centered Robots. Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), 4509–4515, May,” 2021.