IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

An Intention-Aware Robust Safety Framework for Robot Teleoperation: Unifying Object Interaction and Obstacle Avoidance

Zhitao Gao, Chen Chen, Fangyu Peng, Yukui Zhang, Wenke Zhou, Chengao Jiang, Rong Yan, Xiaowei Tang, and Yu Wang

***Abstract*—Control barrier functions (CBFs) have proven to be effective for obstacle avoidance in robot teleoperation systems. However, for classical CBF, model uncertainties and external disturbances can significantly degrade the robustness of safety control. Moreover, the fixed safety boundary lacks adaptability to dynamic switching on operational intentions. To address these limitations, this paper presents a hierarchical safety teleoperation framework that separates the safety layer from the leader-follower teleoperation layers. On this basis, a virtual proxy is introduced to construct a robust control-affine system decoupled from physical robot uncertainties and external disturbances. Building upon this, we propose an intention-aware adaptive control barrier function (IA-ACBF), which consists of two modules: intention detection and intention quantification. The intention detection module determines the operator's transient intention, which belongs to object interaction or obstacle avoidance. The intention quantification module then maps this to the adaptation of safety boundaries. Finally, the performance of the proposed method is validated through simulations and experiments with the physical robot.**

***Index Terms*—Telerobotics and teleoperation, robot safety, control barrier function, intention recognition.**

I. Introduction

TELEOPERATION is widely used to handle complex interactive tasks [1], [2], where excellent compliance strategies and accurate environmental awareness can significantly improve operational safety [3], [4]. Guided by dynamic operator intentions, teleoperated systems can achieve excellent task adaptability. However, the operator may not always be aware of the obstacles during teleoperation, leading to unexpected collisions and equipment damage. Consequently, it is essential to apply strict safety constraints to obstacle avoidance while adapting to the dynamic operator intentions.

Among the existing obstacle avoidance methods for teleoperation, haptic feedback is a simple but effective method that provides haptic force to the operator when approaching obstacles [5]. The haptic force is commonly associated with the obstacle distance [6], collision risk [7], and approaching velocity [8]. Although haptic feedback can alert the obstacle location while preserving original operator intention, this method lacks strict safety constraints, which limits its ability to completely eliminate collisions. The artificial potential field (APF) method achieves strict obstacle avoidance guarantees by constructing repulsive fields [9], [10]. Conversely, the repulsive fields can distort the original trajectory, such as deviating from the intention trajectory in crowded environments. Recently, control barrier function (CBF) has developed as an effective obstacle avoidance method [11]. Compared to haptic feedback and APF methods, CBF enforces strict safety constraints on obstacle avoidance as well as minimal adjustments to the intention trajectory.

CBF has been successfully applied to relatively simple teleoperation tasks with structured environments. In [12], CBF is used to impose predefined constraints on robot joint angles, velocities, and accelerations to avoid kinematic singularities. Similarly, [13] utilizes a zeroing CBF to impose fixed visibility constraints for surgical robots. In [14], CBF is integrated with haptic feedback to achieve collision-free teleoperation for virtual experiment exploration. Despite the formal guarantees for the safety constraints, these classical CBF based frameworks are established on fully known control affine models. Moreover, these methods using fixed safety boundaries to perform the predefined obstacle avoidance function. Therefore, the following challenges arise when dealing with complex tasks with dynamic operator intention.

The first challenge is to achieve robust safety control under model uncertainties and external disturbances. In complex teleoperation tasks, it is difficult to obtain an accurate dynamic model of the physical robot due to frictions and nonlinearities. Meanwhile, external disturbances can seriously degrade the accuracy of the safety trajectory. Although some studies have enhanced the robustness safety control through uncertainty estimation [15], [16], they introduce additional computations and

Received 3 April 2025; accepted 19 July 2025. Date of publication 1 August 2025; date of current version 29 August 2025. This article was recommended for publication by Associate Editor H. Li and Editor C. Gosselin upon evaluation of the reviewers' comments. This work was supported in part by the National Key R&D Program of China under Grant 2023YFB4705100, and in part by the National Natural Science Foundation of China under Grant 52188102. *(Corresponding author: Chen Chen.)*

Zhitao Gao, Fangyu Peng, Yukui Zhang, Wenke Zhou, Chengao Jiang, Rong Yan, and Xiaowei Tang are with the School of Mechanical Science and Engineering, Huazhong University of Science and Technology, Wuhan 430074, China (e-mail: d201980253@hust.edu.cn; zwm8917@263.net; zhangyukui@hust.edu.cn; 486574429@qq.com; 1114042672@qq.com; yanrong@hust.edu.cn; txwysxf@126.com).

Chen Chen is with the Hubei Key Laboratory of Mechanical Transmission and Manufacturing Engineering, Wuhan University of Science and Technology, Wuhan 430081, China (e-mail: chenchen_1014@foxmail.com).

Yu Wang is with the School of Mechatronics Engineering, Henan University of Science and Technology, Luoyang 471003, China (e-mail: yu_wang97@163.com).

This article has supplementary downloadable material available at https://doi.org/10.1109/LRA.2025.3595023, provided by the authors.

Digital Object Identifier 10.1109/LRA.2025.3595023

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

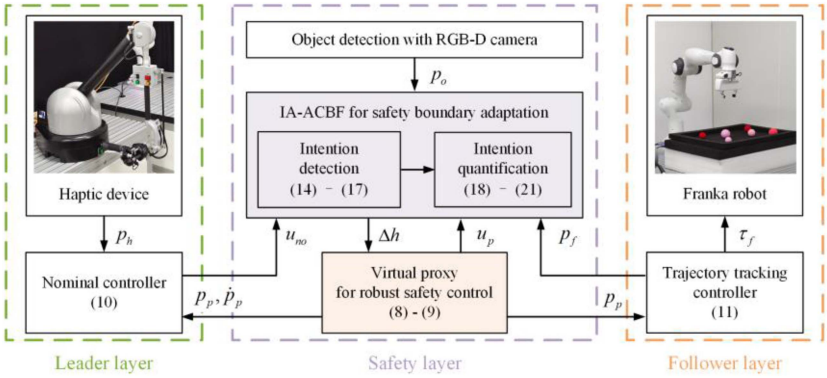

Fig. 1. The proposed hierarchical safety teleoperation framework consists of the leader layer, the safety layer, and the follower layer.

increase control conservatism, which can significantly degrade the transparency of real-time teleoperation.

The second challenge is to adapt the safety boundary to dynamic operator intentions, achieving unified object interaction and obstacle avoidance. Specifically, due to random environmental configurations, the operator may have two opposite potential intentions, "object interaction" or "obstacle avoidance". The fixed safety boundary of the classical CBF lacks the adaptability to such switching intentions, which may even lead to task failure.

To address these challenges, we propose a hierarchical teleoperation framework with virtual proxy based robust safety control, as well as an intention-aware adaptive control barrier function (IA-ACBF) method. Unlike existing ACBF methods that aim to improving solvability or increase conservatism [17], [18], our method focuses on the detecting and quantifying of the operator intentions. This knowledge then guides the adaptation of safety boundaries, unifying object interaction and obstacle avoidance. The main contributions of this paper are as follows:

- A hierarchical safety teleoperation framework is proposed that isolates the safety layer from the leader-follower layer, allowing them to be designed independently.
- A virtual proxy is introduced to construct a model-free control-affine system for safety control, enhancing the robustness to physical robot uncertainties and external disturbances.
- An IA-ACBF method with intention detection and quantification modules is proposed to unify object interaction and obstacle avoidance.

II. Framework Construction

As shown in Fig. 1, the proposed hierarchical safety teleoperation framework (denoted as the hierarchical framework) contains three main components:

1) *Leader layer:* This layer obtains the operator's trajectory via a haptic interface and calculates the nominal control input required for leader-follower synchronization.
2) *Safety layer:* This layer first performs object detection using an RGB-D camera. Then, a virtual proxy is introduced to prevent the safety control from physical robot uncertainties and external disturbances. Finally, an IA-ACBF method is proposed to achieve the safety boundary adaptation through intention detection and quantification modules.
3) *Follower layer:* In this layer, a Franka robot is driven by the trajectory tracking controller to perform tasks along the virtual proxy trajectory.

The working logic of the whole hierarchical framework is briefly described as follows: Firstly, the virtual proxy located in the safety layer constructs a robust control affine system that is independent of model uncertainties and external disturbances. Subsequently, the proposed IA-ACBF method is introduced into the control affine system to achieve intention aware adaptation for safety boundaries. Specifically, the operator transient intentions are first determined by intention detection module, which are divided into "object interaction" or "obstacle avoidance". Next, the transient intentions are further normalized and mapped to the adaptation of safety boundaries by the intention quantification module. Finally, the unifying object interaction and obstacle avoidance is achieved.

III. Virtual Proxy for Robust Safety Control

A. Introduction of Virtual Proxy in Safety Layer

Consider the following control affine system:

$$\dot{x} = f(x) + g(x)u, \tag{1}$$

where $x \in \mathcal{X} \subseteq \mathbb{R}^n$ is the state of the system, $u \in \mathbb{R}^m$ is the control input, $f : \mathcal{X} \to \mathbb{R}^n$ is the drift term, and $g : \mathcal{X} \to \mathbb{R}^{n\times m}$ is the control influence term. Here, a safe set S is defined as the 0-superlevel set of a continuously differentiable function $h : X \to \mathbb{R}$.

$$S = \{x \in \mathcal{X} : h(x) \geq 0\}. \tag{2}$$

The goal of a CBF is to render the safe set forward invariant, that is, $x_0 \in S \Rightarrow x(t) \in S, \forall x_0 \in \mathcal{X}$.

Definition 1 (Control Barrier Function): The function $h(x)$ is a CBF for the system (1) if there exists a class-$\mathcal{K}$ function $\alpha \in \mathcal{K}$ such that $\forall x \in \mathcal{X}$, satisfying [11]:

$$\sup_{u\in\mathbb{R}^m} \dot{h}(x,u) \geq -\alpha(h(x)), \tag{3}$$

$$\dot{h}(x,u) = L_{\mathrm{f}}h(x) + L_{\mathrm{g}}h(x)u, \tag{4}$$

where $\dot{h}$ is the Lie derivative of the h along the system (1).

The safe control input u_s is derived by modifying the nominal controller u_{no} through solving a quadratic program:

$$u_s = \underset{u\in\mathbb{R}^m}{\arg\min}\ \frac{1}{2}\|u-u_{no}\|^2$$
$$\text{s.t.}\dot{h}(x,u) \geq -\alpha(h(x)). \tag{5}$$

On this basis, we analyze the limitations of the classic CBF method, which uses the following physical robot model:

$$M_f(q_f)\ddot{q}_f + C_f(\dot{q}_f,q_f)\dot{q}_f + G_f(q_f) = \tau_f + J^T F_{ext}, \tag{6}$$

where M_f, C_f, G_f are the inertia matrix, Coriolis matrix, and gravity vector of the follower robot. q_f is the joint position vector. τ_f is the control torque, F_{ext} is the external force, and J is the Jacobi.

Rewriting (6) as control-affine form for safety control:

$$\underbrace{\frac{d}{dt}\begin{bmatrix}q_f\\ \dot{q}_f\end{bmatrix}}_{\dot{x}} = \underbrace{\begin{bmatrix}\dot{q}_f\\ M_f^{-1}(-C_f\dot{q}_f - G_f + J^T F_{ext})\end{bmatrix}}_{f(x)} + \underbrace{\begin{bmatrix}0\\ M_f^{-1}\end{bmatrix}}_{g(x)}\underbrace{\tau_c}_{u}. \tag{7}$$

The frictions and nonlinearities cause uncertainties in M_f, C_f, G_f, while F_{ext} introduces external disturbances, which degrades control robustness. Referring to [19], [20], our solution replaces the physical dynamics with a virtual proxy:

$$M_p(p_p)\ddot{p}_p + C_p(\dot{p}_p,p_p)\dot{p}_p + G_p(p_p) = u_p, \tag{8}$$

where M_p, C_p, G_p are the inertia matrix, Coriolis matrix and gravity vector of the proxy, p_p, u_p are the Cartesian position and the control input of the proxy. For simplicity, we set the proxy to a prime point, which reduces (8) to:

$$\underbrace{\frac{d}{dt}\begin{bmatrix}p_p\\ \dot{p}_p\end{bmatrix}}_{\dot{x}} = \underbrace{\begin{bmatrix}\dot{p}_p\\ 0_3\end{bmatrix}}_{f(x)} + \underbrace{\begin{bmatrix}0_{3\times3}\\ M_p^{-1}\end{bmatrix}}_{g(x)}\underbrace{u_p}_{u}. \tag{9}$$

Comparing (9) with (7), the introduction of the virtual proxy makes the $f(x)$ and $g(x)$ independent of the physical robot dynamics. Meanwhile, it also isolates the $f(x)$ from F_{ext} and enhances the robustness to external disturbances.

B. Trajectory Tracking Control in Leader-Follower Layer

After the design of safety layer, we use a nominal PD controller in leader layer to obtain the nominal input required for the proxy to track the operator's reference trajectory:

$$u_{no} = M_p\ddot{p}_p + D_p(\dot{p}_r - \dot{p}_p) + K_p(p_r - p_p), \tag{10}$$

where D_p and K_p are damping and stiffness matrices of the nominal controller, respectively. p_h is the operator trajectory.

Meanwhile, in follower layer, p_p is transformed into the joint space trajectory q_p by inverse kinematics, then a trajectory tracking controller is used to obtain the desired input required for the follower to track the virtual proxy:

$$\tau_f = K_f(q_p - q_f) + D_f\dot{q}_f, \tag{11}$$

where K_f, D_f are proportional and damped gain matrices.

IV. IA-ACBF for Boundary Adaptation

The hierarchical framework with virtual proxy has enhanced the robustness of the safety control. Furthermore, we deal with the problem of fixed safety boundaries for dynamic intention adaptation. To overcome this limitation, we first introduce an ACBF $\tilde{h}$ with a variable boundary set $\tilde{S}$:

$$\tilde{S} = \left\{x \,\middle|\, \tilde{h}(x) \geq 0\right\}, \tag{12}$$

$$\tilde{h}(x) = h(x) + \Delta h, \tag{13}$$

where Δh is the compensation term that dynamically adjusts the safety boundaries. On this basis, we propose the IA-ACBF method, which consists of intention detection and quantification modules. These two modules adjust the Δh, adaptively switch between object interaction and obstacle avoidance within a unified framework.

A. Intention Detection

This module determines the operator's transient intentions through an event-triggered algorithm. The event-triggering conditions are defined as follows:

1) Event 1: The virtual proxy and the follower robot are synchronized.

$$\|p_p - p_f\|_2 \leq p_{ref}. \tag{14}$$

where p_{ref} is a tracking error threshold that can be determined by performing a pre-experiment and referring to the maximum error during the smooth teleoperation. In this paper, p_{ref} is set to 0.02 m.

2) Event 2: System states approach the safety boundary.

$$\tilde{h}(x) \leq h_{ref}. \tag{15}$$

where h_{ref} is a safety boundary threshold. In this paper, h_{ref} is set to 0.0025 m, which is 10% of the initial safety boundary.

3) Event 3: Detection of a sufficiently large control input.

$$abs(u_{no}) \geq u_{ref_1}. \tag{16}$$

where u_{ref_1} is an effective input threshold. u_{ref_1} must be greater than the unexpected control jitter caused by sensor noise to avoid false-active activation. Considering the input noise of the leader device, u_{ref_1} is set to 0.5 N in this paper.

4) Event 4: Effective safety constraints on nominal inputs.

$$u_p/u_{no} \leq u_{ref_2}. \tag{17}$$

where u_{ref_2} is a constraint input threshold. Compared with quickly passing over obstacles, inputs that continuously approach obstacles will be more strongly constrained. The former may be the obstacle avoidance intention, while the latter is the interaction intention. Users can adjust the sensitivity of this intention judgment mechanism through the value of u_{ref_2}. In this paper, u_{ref_2} is set to 0.4.

Remark 1: Since the safety constraints are first imposed directly on the virtual proxy, and then synchronized to the follower robot by tight trajectory tracking. Event 1 ensures that this indirect safety mechanism works and provides the basis for reliable intention detection.

Remark 2: The triggering of Events 2-4 under Event 1 indicates that the control inputs are still valid despite proximity to safety boundaries and historical input constraints, which means that the transient intention of the operator is object interaction. Otherwise, it is obstacle avoidance.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

TABLE I
CONFIGURATIONS OF CASES IN SIMULATION 1

Case	Method	Uncertainty	Disturbance	Virtual proxy
1	Classical CBF	/	/	/
2	Classical CBF	$\Delta m = 1$	$f_{ext} = 1.5\sin(t)$	/
3	Hierarchical framework	$\Delta m = 1$	$f_{ext} = 1.5\sin(t)$	$m_p = \bar{m} = 1$

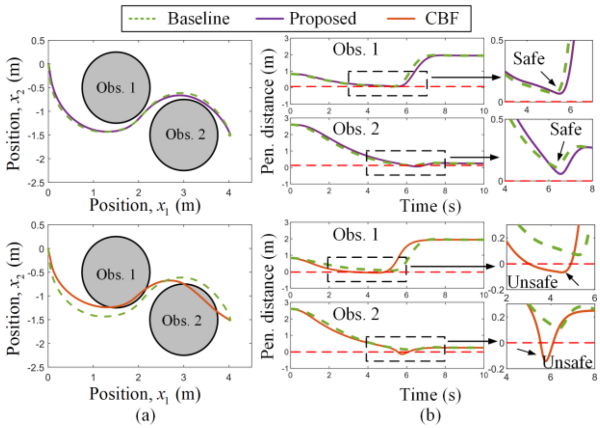

Fig. 2. Numerical results of the simulation 1. (a) 2D trajectory. (b) Comparison of obstacle penetration distances.

B. Intention Quantification

Immediate safety boundary adjustments upon intention detection can cause control instability due to the sensor error and the operational jitter. Therefore, we further quantify the intention through temporal persistence analysis. Referring to time based resource sharing (TBRS) theory [21], human operators exhibit progressive intention reinforcement for a sudden situation as its duration increases. Thus, we proposed the confidence c_t for intention quantification. c_t is a duration-dependent normalized metric in [0, 1], where 0 represents complete distrust and 1 represents full credibility. Based on c_t, the module can be divided into two phases:

1) Confidence Growth Phase: After continuous detection of the object interaction intention, the confidence follows an exponential growth law:

$$c_t = (1 - c_0) \cdot (1 - e^{-rt}) + c_0, \tag{18}$$

where t is the current phase duration, c_0 is the remaining confidence from the previous phase, and r is a growth rate that governs the speed of safety boundary adaptation.

2) Confidence Decay Phase: After continuous detection of the obstacle avoidance intention, the confidence decays exponentially:

$$c_t = c_0 \cdot e^{-Dt}, \tag{19}$$

where D is the decay rate. Note that t should be reset when the phase switches.

The quantified confidence drives safety boundary adaptation through the iterative learning:

$$\Delta h = \begin{cases} \Delta h + k, & \text{if } c_t \geq c_2 \\ \Delta h, & \text{if } c_1 < c_t < c_2 \\ \Delta h - k, & \text{if } c_t \leq c_1 \end{cases}, \tag{20}$$

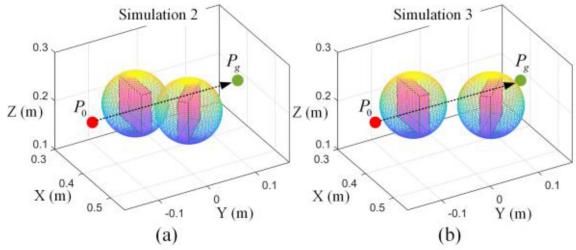

Fig. 3. Two simulation scenarios. (a) Simulation 2: crowded obstacles. (b) Simulation 3: separated obstacles.

Algorithm: IA-ACBF With Intention Detection and Quantification.

Input: Events 1- 4, Δt, k
Variable: Δh, t, c_0, c_t
Initialize: $\Delta h = 0$, $t = 0$, $c_0 = 0$
// Intention detection
if Events 1- 4 are activated
  Transient intention = = Object interaction;
else
  Transient intention = = Obstacle avoidance;
end
// Intention quantification
if (Transient intention = = Object interaction)
  Solve (18), $t = t + \Delta t$;
else
  Solve (19), $t = 0$;
end
// Safety boundary adaptation:
  Update Δh via (20); Apply saturation (21)
Output: Δh

with a saturation function:

$$Sat(\Delta h) = \begin{cases} \Delta h_{\min}, & \text{if } \Delta h \leq \Delta h_{\min} \\ \Delta h_{\max}, & \text{if } \Delta h \geq \Delta h_{\max} \\ \Delta h, & \text{otherwise} \end{cases}, \tag{21}$$

where k is the learning gain, c_1 and $c_2$are user-defined trigger thresholds, $\Delta h_{\max}$ and $\Delta h_{\min}$ are the upper and lower saturations, which prevent the dangerous over-expansion of safety boundaries.

To summarize, the pseudo-code of the IA-ACBF algorithm is shown below:

V. SIMULATION STUDIES

Three simulation studies are carried out to verify the proposed framework, including: robustness verification of the virtual proxy (Simulation 1), and boundary adaptation of the IA-ACBF method (Simulation 2 and 3). The simulation environment is Matlab/Simulink and visualized by Pybullet, and the simulation details can be found in the attached video.

A. Simulation 1: Robustness Test of the Virtual Proxy

This test validates the robustness of the proposed hierarchical framework with virtual proxy using a 2D double-integrator

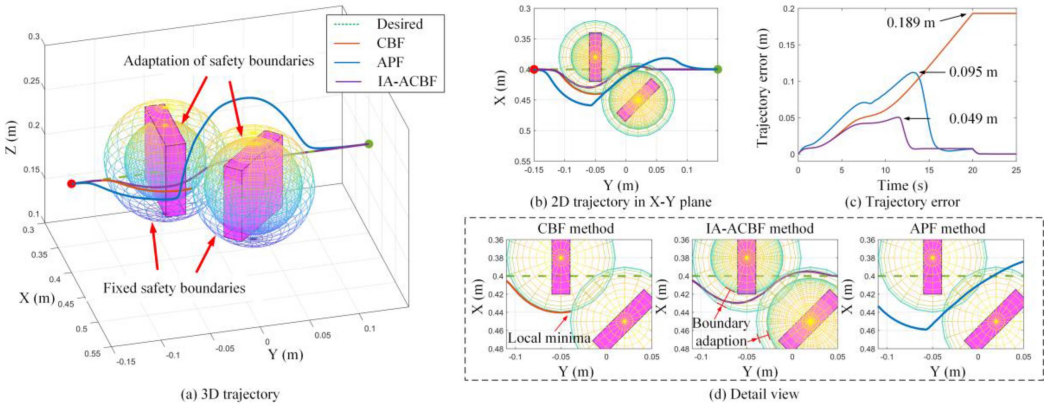

Fig. 4. Trajectory comparison for simulation 2 with obstacles in close proximity. (a) 3D trajectory. (b) X-Y plane projection. (c) Trajectory errors. (d) Detail view of the obstacle area.

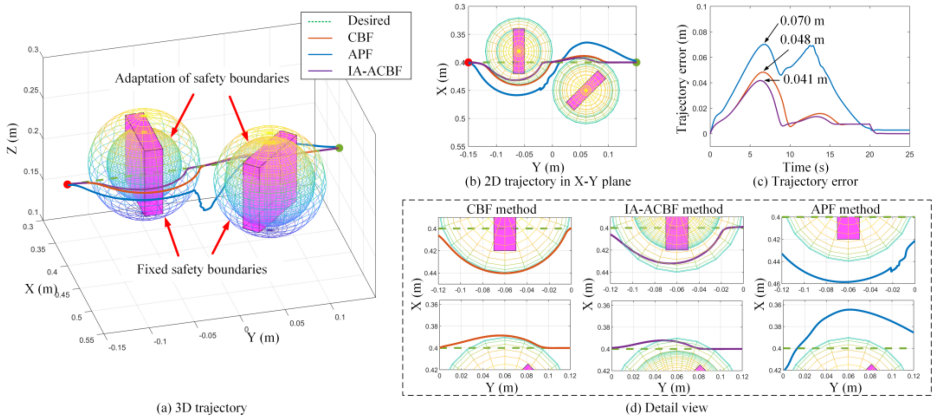

Fig. 5. Trajectory comparison for Simulation 2 with widely separated obstacles. (a) 3D trajectory. (b) X-Y plane projection. (c) Trajectory errors. (d) Detail view of the obstacle area.

system [19]:

$$\ddot{x} = \frac{1}{m} u_{no} + f_{ext}, \tag{22}$$

$$m = \bar{m} + \Delta m, \tag{23}$$

where x, u_{no} and f_{ext} denote robot position, nominal input and disturbances. The real mass m combines nominal mass $\bar{m} = 1$ and the uncertainty Δm. The control objective is to navigate the robot from the initial position x_0 to a desired position x_d while avoiding circular obstacles using CBF:

$$h(x) = (\dot{x} - \dot{x}_c) + (x - x_c) - r, \tag{24}$$

where r is the radius of the obstacle with x_c as its central position.

Three comparative cases are designed with the configurations shown in Table I.

As shown in Fig. 2(a), compared to the classical CBF method, the proposed hierarchical framework has better trajectory robustness, reducing the average error from the baseline trajectory by 37.8% (0.145 m vs. 0.233 m). Fig. 2(b) shows the comparison of obstacle penetration distances, where our method maintains a positive safety margin throughout the entire test, while the classical CBF has two unsafe obstacle penetrations. In summary, the proposed method can enhance the robustness of safety control under uncertainties and disturbances.

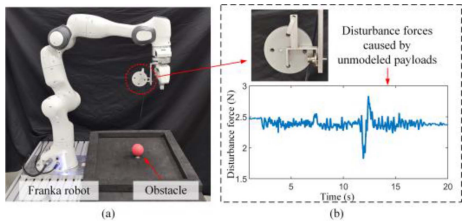

Fig. 6. (a) Real-word robustness experiment setup. (b) Variation of the disturbance force.

B. Simulation 2 and 3: Boundary Adaptation of the IA-ACBF Method

Simulations 2, 3 test the obstacle avoidance performance using the proposed IA-ACBF method. The follower in simulations is a virtual Franka robot whose model parameters are taken from [22]. The operator intention is to perform a linear trajectory from start point $P_0 = [0.4, -0.15, 0.2]$ m to goal point

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

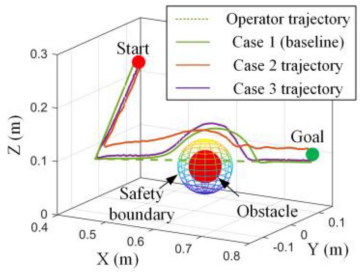

Fig. 7. Trajectory comparison for experiment 1, set 1.

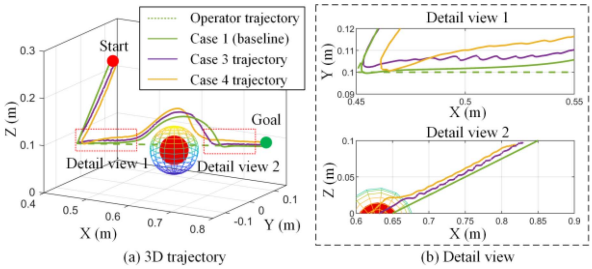

Fig. 8. Trajectory comparison for experiment 1, set 2. (a) Global 3D trajectory of obstacle avoidance. (b) Detail view of the trajectory for the comparison of motion smoothness.

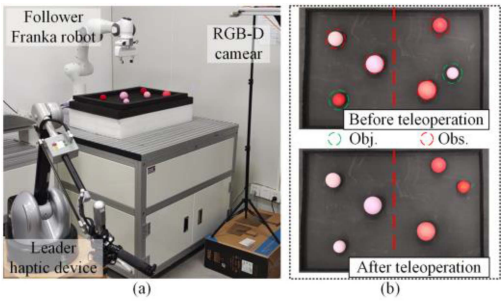

Fig. 9. Robot teleoperation balls classification experiment. (a) Hardware system setup. (b) Task objective: Balls within each half area should be same after teleoperation.

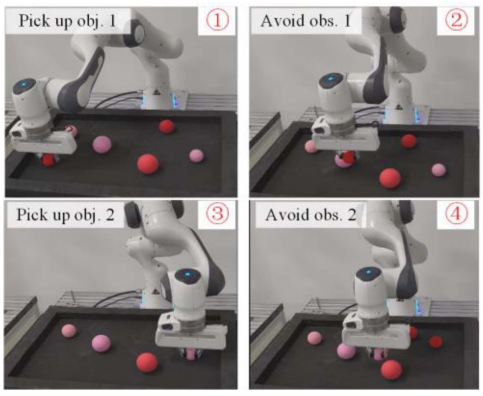

Fig. 10. Experimental snapshots of balls classification task. The details of the procedure can be found in the attached video.

$P_g = [0.4, 0.15, 0.2]$ m. The workspace contains two rectangular obstacles with the size of $0.08 \times 0.02 \times 0.08$ m, which are surrounded by spherical safety boundaries with radius $r = 0.06$ m.

The configuration of the obstacles is shown in Fig. 3, Simulation 2 has crowded obstacles with centers at $o_1 = [0.38, -0.05, 0.2]$ m and $o_2 = [0.45, 0.02, 0.2]$m, while Simulation 3 has separated obstacles with centers at $o_1 = [0.38, -0.06, 0.2]$ m and $o_2 = [0.45, 0.06, 0.2]$ m. Each simulation carries out a comparison of the obstacle avoidance performance using three methods: the classical CBF, the APF and the proposed IA-ACBF methods.

As shown in Fig. 4, for crowded obstacles in Simulation 2, the classical CBF method fails to complete the task due to trapping in local minima. The APF method achieves obstacle avoidance but deviates significantly from the intended trajectory. In contrast, the proposed IA-ACBF method dynamically adapts safety boundaries to accommodate both the operator's straight-line motion intention and obstacle avoidance. By temporarily tightening the safety boundaries, the proposed method successfully navigates through the crowded obstacle region with minimal trajectory error. As shown in Fig. 4(c), the maximum trajectory error of the proposed method is 0.049 m, achieving error reductions of 48.4% compared to the classical APF (0.095 m) and 74.1% compared to the CBF (0.189 m) methods, respectively.

For separated obstacles in Simulation 3, as shown in Fig. 5, while all three methods successfully complete the task, the proposed method has the best obstacle avoidance trajectory performance. Referring to Fig. 5(c), its maximum trajectory error is 0.041 m, which is 14.6% lower compared to the CBF method (0.048 m) and 41.4% lower compared to the APF method (0.070 m).

VI. Experimental Results

We further carry out two real-world experiments to verify the comprehensive performance of the proposed hierarchical framework with the IA-ACBF method. The control system runs on a desktop computer with an AMD 5800x CPU at 4.7 GHz and 32 GB of memory. Considering the single-step solution time (4.54 μm) and control-loop latency (0.76 ms) of the proposed method, the system control period is set to 500 Hz.

A. Experiment 1: Real-World Robustness Test With Unmodelled Payload and Different Proxy Parameters

This experiment is to verify the robustness of the proposed hierarchical framework with the unmodelled payload and different proxy parameters. The experimental setup is shown in Fig. 6(a). The robot needs to follow the operator's trajectory and avoid the obstacle (a red ball with $r = 0.03$ m) in the path. Since an unfixed, unmodeled payload is attached to the end of the robot, unknown disturbance forces will be generated during the robot's movement, which is shown in Fig. 6(b). Subsequently, two sets with four comparative cases are designed with the configurations shown in Table II. Among them, case 1 tests the performance in the most ideal condition (without payload), which serves as a baseline for comparison.

The set 1 consists of case 1, 2, and 3 is used to verify the robustness of the proposed hierarchical framework under unmodeled payload. As shown in Fig. 7, the trajectory of case 1

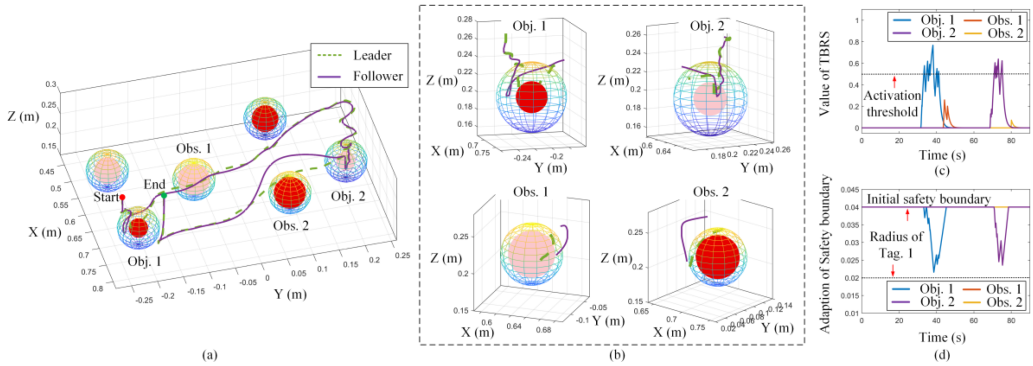

Fig. 11. Performance of the ball classification experiment using the proposed method. (a) 3D trajectory. (b) Detail view of the trajectory arounds the objects and obstacles. (c) TBRS values. (d) Adaptation of the safety boundaries.

TABLE II
CONFIGURATIONS OF CASES IN EXPRIMENT 1

Case	Method	Uncertainty	Proxy parameter
1	Classical CBF (baseline)	Without payload	/
2	Classical CBF	With payload	/
3	Hierarchical framework	With payload	$M_p = 1 * eye(6)$
4	Hierarchical framework	With payload	$M_p = 1 * eye(6)$

is the ideal obstacle avoidance trajectory without payload interference and serves as the comparison baseline. Compared with the case 2 with CBF method, the case 3 with proposed hierarchical framework has better robustness in real-world experiments, reducing the average error from the baseline trajectory by 65.3% (0.0134 m vs. 0.0386 m).

Meanwhile, the set 2 consists of case 1, 3, and 4 is used to quantitatively analyze the influence of proxy parameters on the trajectory tracking performance. As shown in Fig. 8(a), the proxy in case 3 has a smaller inertia parameter and is more tightly constrained by the safety controller. The average error between case 3 and baseline trajectory is 0.0134 m, while the minimum distance from the obstacle boundary is 0.055 m; For comparison, the proxy in case 4 has a larger inertia parameter. The average error between case 4 and baseline trajectory is 0.0238 m, which is 77.6% higher than case 3. Meanwhile, the minimum distance from the proxy to the obstacle decreases to 0.005 m, significantly increasing the risk of collision.

Although increasing inertia parameter of the proxy decreases safety control performance, it also has positive aspects. As shown the local view in Fig. 8(b), the trajectory of case 4 is smoother than that of case 3. In summary, selecting the proxy parameter involves a trade-off between trajectory smoothness and safety constraint performance. To obtain a relatively smooth trajectory while meeting safety constraints, one approach is to increase the safety constraint gain as the proxy inertia parameter increases.

B. Experiment 2: Intention-Aware Safety Control Test With Ball Classification Task

This experiment is used to test the intention-aware adaptive safety control performance of the proposed IA-ACBF method under random environment configurations. The experimental setup is shown in Fig. 9(a), where the leader is the Haption V6DT device and the follower is the physical Franka robot. The positions of the balls are collected by an Intel D435 camera using YOLO_v5 detection algorithm [23]. To account for detection errors and random ball radii ($r \in [0.02, 0.03]$m), we establish relatively conservative spherical initial boundaries around each ball with radius $r_0 = 0.05$ m. As shown in Fig. 9(b), balls of two colors are randomly placed in a rectangular workspace. The task requires that the balls within each half area should be same after teleoperation. This random configuration requires the real-time intention recognition, so that the control framework adjusts safety boundaries for dynamic obstacle avoidance and object interaction.

The experimental snapshots are shown in Fig. 10, the robot first picks up a red ball (object 1) from the lower-left area, avoids a pink ball (obstacle 1), and places it in the upper-right area. Subsequently, the robot picks up the pink ball (object 2) located in the lower-right area, avoids a red ball (obstacle 2), and ultimately places it in the lower-left area.

The global experimental trajectory using the proposed method is shown in Fig. 11(a), where the follower tightly tracks the leader and successfully avoids obstacles. As detailed views shown in Fig. 11(b), for object balls, the robot breaks initial safety boundaries and successfully pick up them. Conversely, for obstacle balls, the robot exhibits robust collision avoidance even when the leader's trajectory transiently penetrates safety boundaries, thereby validating the unified capacity of both obstacle avoidance and object interaction. This unique performance arises from the different activation patterns of the intention detection and quantification modules. As shown in Fig. 11(c), the leader trajectory maintains the proximity intention to object balls for a certain time, which leads to a significant activation of the TBRS, resulting a shrinking of the safety boundaries in Fig. 11(d). In contrast, when the leader trajectory only transiently passes over the obstacle balls, the TBRS value in Fig. 11(c) does not reach the threshold, so that the safety boundaries in Fig. 11(d) remain constant to perform obstacle avoidance.

As a comparison, we also give the experimental trajectory using a classical CBF method, whose limitations become obvious with such a complex teleoperation task. As shown in Fig. 12(a), although the operator attempts to adjust the trajectory to pick up the object ball, a conservative fixed safety boundary makes this impossible. Conversely, when manually shrinking safety boundaries to overcome the object pickup problem, the problem arises instead with the obstacle ball, resulting the failure of obstacle avoidance in Fig. 12(b).

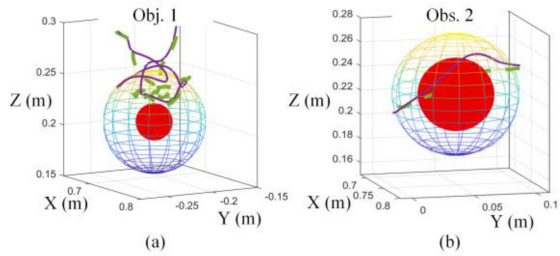

Fig. 12. Performance of the ball classification experiment using the classical CBF method with a fixed safety boundary. (a) Detail view of the trajectory arounds the object 1 with a safety boundary $r_0 = 0.05$ m. (b) Detail view of the trajectory arounds the obstacle 2 by manually shrinking the safety boundary to $r_0 = 0.02$ m.

VII. Conclusion

To deal with the intention-aware robust safety control problem under complex teleoperation tasks, we propose a hierarchical safety teleoperation framework that unifies object interaction and obstacle avoidance. In this framework, the safety layer is decoupled from the leader-follower layers by introducing the virtual proxy. Meanwhile, this design enhances the robustness of the safety control against physical robot uncertainties and external disturbances. On this basis, an IA-ACBF method is proposed to enable real-time intention detection and quantification, achieving dynamic adaptation of safety boundaries. Simulations and the experiment show that the proposed method achieves a trade-off between object interaction and obstacle avoidance. Compared with the classical CBF and APF methods, the proposed method avoids falling into local minima while significantly reducing the error of the obstacle avoidance trajectory.

References

[1] B. Ma et al., "Advances in space robots for on-orbit servicing: A comprehensive review," *Adv. Intell. Syst.*, vol. 5, no. 8, Aug. 2023, Art. no. 2200397, doi: 10.1002/aisy.202200397.

[2] C. Yang et al., "A review of advances in underwater humanoid robots for human-machine cooperation," *Robot. Auton. Syst.*, vol. 179, Mar. 2024, Art. no. 104744, doi: 10.1016/j.robot.2024.104744.

[3] H. Zhou, X. Zhang, J. Liu, and Z. Ju, "An active-passive compliance strategy for robotic plugging and unplugging of rocket electrical connectors," *IEEE/ASME Trans. Mechatron.*, vol. 30, no. 2, pp. 1014–1025, Apr. 2025, doi: 10.1109/TMECH.2024.3410016.

[4] Y. Zhang, W. Liang, M. Yuan, H. He, J. Tan, and Z. Pang, "Monocular visual-inertial and robotic-arm calibration in a unifying framework," *IEEE/CAA J. Automatica Sinica*, vol. 9, no. 1, pp. 146–159, Jan. 2022, doi: 10.1109/JAS.2021.1004290.

[5] D. Lee, A. Franchi, H. I. Son, C. Ha, H. H. Bülthoff, and P. R. Giordano, "Semiautonomous haptic teleoperation control architecture of multiple unmanned aerial vehicles," *IEEE/ASME Trans. Mechatron.*, vol. 18, no. 4, pp. 1334–1345, Aug. 2013, doi: 10.1109/TMECH.2013.2263963.

[6] H. Courtois, N. Aouf, K. Ahiska, and M. Cecotti, "OAST: Obstacle avoidance system for teleoperation of UAVs," *IEEE Trans. Hum.-Mach. Syst.*, vol. 52, no. 2, pp. 157–168, Apr. 2022, doi: 10.1109/THMS.2022.3142107.

[7] S. Peng et al., "Collision risk assessment and operation assistant strategy for teleoperation system," *Appl. Sci.*, vol. 13, no. 7, Apr. 2023, Art. no. 4109, doi: 10.3390/app13074109.

[8] N. Zhong and K. Hauser, "Attentiveness map estimation for haptic teleoperation of mobile robot obstacle avoidance and approach," *IEEE Robot. Automat. Lett.*, vol. 9, no. 3, pp. 2152–2159, Mar. 2024, doi: 10.1109/LRA.2024.3354613.

[9] A. Gottardi, S. Tortora, E. Tosello, and E. Menegatti, "Shared control in robot teleoperation with improved potential fields," *IEEE Trans. Hum.-Mach. Syst.*, vol. 52, no. 3, pp. 410–422, Jun. 2022, doi: 10.1109/THMS.2022.3155716.

[10] W. Qin et al., "A shared control method for teleoperated robot using artificial potential field," *Adv. Intell. Syst.*, vol. 6, no. 10, Oct. 2024, Art. no. 2300814, doi: 10.1002/aisy.202300814.

[11] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *IEEE Trans. Autom. Control*, vol. 62, no. 8, pp. 3861–3876, Aug. 2017, doi: 10.1109/TAC.2016.2638961.

[12] D. Lee, D. Ko, W. K. Chung, and K. Kim, "Quadratic programming-based task scaling for safe and passive robot arm teleoperation," *IEEE/ASME Trans. Mechatron.*, vol. 27, no. 4, pp. 1937–1945, Aug. 2022, doi: 10.1109/TMECH.2022.3175247.

[13] Z. Deng, X. Wei, C. Pan, G. Li, and Y. Hu, "Shared control of tendon-driven continuum robots using visibility-guaranteed optimization for endoscopic surgery," *IEEE Trans. Med. Robot. Bionics*, vol. 6, no. 2, pp. 487–497, May 2024, doi: 10.1109/TMRB.2024.3381371.

[14] D. Zhang, G. Yang, and R. P. Khurshid, "Haptic teleoperation of UAVs through control barrier functions," *IEEE Trans. Haptics*, vol. 13, no. 1, pp. 109–115, Jan./Mar. 2020, doi: 10.1109/TOH.2020.2966485.

[15] A. Alan, T. G. Molnar, E. Daş, A. D. Ames, and G. Orosz, "Disturbance observers for robust safety-critical control with control barrier functions," *IEEE Control Syst. Lett.*, vol. 7, pp. 1123–1128, 2023, doi: 10.1109/LCSYS.2022.3232059.

[16] S. H. Zhang et al., "Feasibility-guaranteed safety-critical control with high-order control barrier function method," *Int. J. Robust Nonlinear Control*, vol. 34, no. 5, pp. 3425–3441, Mar. 2024, doi: 10.1002/rnc.7141.

[17] A. Alan, T. G. Molnar, A. D. Ames, and G. Orosz, "Parameterized barrier functions to guarantee safety under uncertainty," *IEEE Control Syst. Lett.*, vol. 7, pp. 2077–2082, 2023, doi: 10.1109/LCSYS.2023.3285188.

[18] C. Dawson, S. Gao, and C. Fan, "Safe control with learned certificates: A survey of neural Lyapunov, barrier, and contraction methods for robotics and control," *IEEE Trans. Robot.*, vol. 39, no. 3, pp. 1749–1767, Jun. 2023, doi: 10.1109/TRO.2022.3232542.

[19] T. G. Molnar, R. K. Cosner, A. W. Singletary, W. Ubellacker, and A. D. Ames, "Model-free safety-critical control for robotic systems," *IEEE Robot. Automat. Lett.*, vol. 7, no. 2, pp. 944–951, Apr. 2022, doi: 10.1109/LRA.2021.3135569.

[20] A. Singletary et al., "Comparative analysis of control barrier functions and artificial potential fields for obstacle avoidance," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2021, pp. 8129–8136, doi: 10.1109/IROS51168.2021.9636670.

[21] K. Oberauer and S. Lewandowsky, "Modeling working memory: A computational implementation of the time-based resource-sharing theory," *Psychon. Bull. Rev.*, vol. 18, no. 1, pp. 10–45, Feb. 2011, doi: 10.3758/s13423-010-0020-6.

[22] C. Gaz, M. Cognetti, A. Oliva, P. R. Giordano, and A. D. Luca, "Dynamic identification of the Franka Emika Panda robot with retrieval of feasible parameters using penalty-based optimization," *IEEE Robot. Automat. Lett.*, vol. 4, no. 4, pp. 4147–4154, Oct. 2019, doi: 10.1109/LRA.2019.2931248.

[23] J. Redmon et al., "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 779–788, doi: 10.1109/CVPR.2016.91.