Safe Reinforcement Learning Tracking Control based on Tunable Input-to-State Safe Control Barrier Function

Soha Kanso, Mayank Shekhar Jha* and Didier Theilliol

Abstract— **This paper develops a novel off-policy safe Reinforcement Learning (RL) approach for optimal tracking of continuous-time nonlinear systems affine in control. The main contribution consists in the synthesis of an optimal tracker under safety guarantees enabling optimal tracking while satisfying state based safety constraints. To ensure safety during the exploration phase, even in the presence of model uncertainty, control inputs are dynamically adjusted. These adjustments, determined as solutions to a quadratic programming (QP) problem, incorporate tunable input-to-state safe control barrier function (TISSf-CBF) conditions. Additionally, the safety during exploitation (operational phase) of the learned policy is guaranteed by integrating a reciprocal control barrier function (RCBF) into the cost function, leading to an effective trade-off between safety and system performance. Novel mathematically rigorous proofs are developed to guarantee the safety, the stability and the convergence towards optimality. Finally, the effectiveness of the approach is assessed using a simulation example.**

I. INTRODUCTION

The safe tracking problem, a prevalent optimal control problem calls for design of controllers that guide a system to follow a specified trajectory or reference signal while adhering to safety constraints. Reinforcement Learning (RL) has emerged as a powerful learning paradigm for synthesizing optimal controllers in uncertain systems through iterative real-time interactions with the environment [1].

To ensure safety in regulation problems for safety-critical systems, several safe RL approaches have been developed. For example, [2] combines policy-gradient RL with control barrier functions (CBF) to achieve safety, albeit typically requiring nominal models. In [3], a barrier function-based system transformation converts full-state constrained systems into equivalent unconstrained ones, enabling standard control and optimization techniques. Similarly, [4] employs model-based RL with CBF for safe exploration, while [5] leverages a Lyapunov-like barrier function to design a partially model-free safeguarding controller that supports online value function learning. Furthermore, [6] introduces a safe Q-learning algorithm for uncertain systems by framing the task as a constrained optimal control problem using reciprocal CBFs, and [7] presents a barrier-Lyapunov actor-critic framework that integrates CBF and control Lyapunov functions (CLF) with actor-critic RL to ensure both safety and stability. More recently, [8] proposed an end-to-end safe learning approach based on CBF and CLF conditions, which guarantees safety from initialization through exploration and exploitation. Collectively, these approaches illustrate a range of strategies designed to achieve safe RL in systems subject to strict state constraints.

Few notable works have addressed safe tracking in RL. For instance, [9] proposed a safe model-based RL algorithm for collision-free model-reference trajectory tracking of uncertain autonomous vehicles, using a robust CBF condition and Gaussian process regression to estimate model uncertainties. In contrast, our paper ensures safety during exploration despite unknown noise or model uncertainty by integrating input-to-state safe control barrier functions (ISSf-CBF) [10]. Although ISSf-CBFs can be inflexible and lead to conservative data collection, we adopt a generalized version—Tunable ISSf-CBF (TISSf-CBF) [11]—to enable richer exploration while guaranteeing input-to-state safety (ISSf) [12]. Additionally, we introduce an innovative approach that augments system states with tracking errors to address safe tracking in nonlinear systems during both exploration and operational phases.

The contributions of this work are as follows:

- a novel tracking formulation that integrates system states with tracking errors to enhance state tracking;
- safety during exploration is ensured in the presence of unknown probing noise or model uncertainty by satisfying the TISSf-CBF condition, which enables relaxed exploration within the safe set and more conservative actions near its boundaries;
- safety and optimality of the learned operational policy are secured by augmenting the reward function with an RCBF acompanied with rigorous mathematical proofs.

This section is followed by Section 2, which formulates the safe tracking problem. Section 3 introduces a safe policy iteration approach with convergence proofs. Section 4 develops the off-policy algorithm, and Section 5 evaluates its effectiveness on an academic application. Finally, the conclusion highlights the key advances of this work.

Notations. The interior of set $\mathscr{C}$ is denoted as $Int\mathscr{C}$ and $\partial\mathscr{C}$ stands for its boundary. For a differentiable function $V(x)$ and a vector $f(x)$, the notation $L_fV(x)$ corresponds to $\frac{\partial V}{\partial x}f(x)$. The symbol $\otimes$ denotes the Kronecker product. C^1 refers to the class of continuously differentiable functions.

II. SAFE OPTIMAL TRACKING CONTROL PROBLEM

In this section, the formulation of nonlinear optimal tracker is presented. Consider nonlinear systems affine in control

*Corresponding author, Universite de Lorraine, CNRS, CRAN, F-54000 Nancy, France. soha.kanso@univ-lorraine.fr, mayank-shekhar.jha@univ-lorraine.fr, didier.theilliol@univ-lorraine.fr

input in continuous-time

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

where $x \in \mathscr{X} \subseteq \mathbb{R}^n$ represents the state of the system, $u \in \mathscr{U} \subseteq \mathbb{R}^m$ is the control input. $f(.)$: $\mathscr{X} \to \mathbb{R}^n$ and $g(.)$: $\mathscr{X} \to \mathbb{R}^{n\times m}$ are Lipschitz continuous and $f(0) = 0$. The sets $\mathscr{X}$ and $\mathscr{U}$ are compact. $\mathscr{U}$ denotes the set of all admissible inputs that ensure stability of the system. $\mathscr{C} \subseteq \mathscr{X}$ denotes the set of safe states in which these later must evolve to ensure a safe operation. The mathematical definition of $\mathscr{C}$ is as follows

$$\mathscr{C} = \{x| \quad h(x) \geq 0\}, \tag{2}$$

for a smooth (continuously differentiable) function h: $\mathscr{X} \to \mathbb{R}$.

The objective in this work is to design a safe infinite-horizon tracker for the system (1). The controller must force the state $x(t)$ to optimally follow the reference trajectory $x_r(t)$ while adhering to safety boundaries and constraints.

Assumption 1 *[13] The command generator model of the reference trajectory is defined by*

$$\dot{x}_r = z(x_r), \tag{3}$$

where $z(x_r)$ is a Lipschitz continuous function with $z(0) = 0$ and $x_r \in \mathbb{R}^p$ is bounded.

The error is formulated as

$$e_r(t) = C_r x(t) - x_r(t), \tag{4}$$

where $e_r \in \mathbb{R}^p$ and $C_r \in \mathbb{R}^{p\times n}$ refers to the states to be tracked, since the target in this paper is to track specific states. Moreover, all the states are assumed to be measurable. The dynamics of the tracking error can be expressed in terms of the control input u as follows

$$\dot{e}_r = C_r(f(x) + g(x)u) - z(C_r x - e_r). \tag{5}$$

Based on (1) and (5), an augmented system is described in terms of the system states x and the tracking error e_r as follows. Let $X \in \mathscr{C} \times \mathbb{R}^p \subset \mathbb{R}^{n+p}$ be defined as

$$X = \begin{bmatrix} x \\ e_r \end{bmatrix}.$$

Then, the augmented system is represented as follows

$$\dot{X} = \begin{bmatrix} \dot{x} \\ \dot{e}_r \end{bmatrix} = \tilde{F}(X) + \tilde{G}(X)u, \tag{6}$$

with $\tilde{F}(X) = \begin{bmatrix} f(x) \\ C_r f(x) - z(C_r x - e_r) \end{bmatrix}$ and $\tilde{G}(X) = \begin{bmatrix} g(x) \\ C_r g(x) \end{bmatrix}$.

The general reward function, which is often referred to as the stage cost or immediate cost, is usually considered in the following manner for tracking problem

$$r = e_r^T Q e_r + u^T R u, \tag{7}$$

where Q and R are symmetric and positive definite. However, this reward function does not take into account any safety considerations. To this end, in this paper, based on the augmented system (6), a modified reward function that is sensitive to the system safety is introduced as

$$\tilde{r}(X,u) = X^T \tilde{Q} X + u^T R u + B_\vartheta(\rho X), \tag{8}$$

with $\tilde{Q} = \begin{bmatrix} 0 & 0 \\ 0 & Q \end{bmatrix}$, $\rho = \begin{bmatrix} I_{n\times n} & 0_{n\times p} \end{bmatrix} \in \mathbb{R}^{n\times(n+p)}$ and B_ϑ a reciprocal control barrier function (RCBF) defined as follows with $\vartheta > 0$

$$B_\vartheta(\rho X) = -\log\left(\frac{\vartheta h(\rho X)}{\vartheta h(\rho X) + 1}\right). \tag{9}$$

Definition 1 *[14] A function $B : Int\mathscr{C} \to \mathbb{R}$ is a reciprocal control barrier function for the set $\mathscr{C}$ if there exists class κ functions α_1, α_2 and α_3 such that*

$$\frac{1}{\alpha_1(h(x))} \leq B(x) \leq \frac{1}{\alpha_2(h(x))} \tag{10}$$

$$\inf_{u\in\mathscr{U}} [L_f B(x) + L_g B(x) u - \alpha_3(h(x))] \leq 0 \quad \forall x \in Int\mathscr{C}. \tag{11}$$

Definition 2 ***(Control Policy).*** *A control policy is defined as a function from the state space to the control space, typically denoted as $\pi : \mathscr{C} \times \mathbb{R}^p \to \mathscr{U}$. In other words, given a state X, the policy determines a control input by:*

$$u = \pi(X) \tag{12}$$

Based on the safe reward function (8) and the control policy (12), the following discounted performance function is introduced

$$V(X(t)) = \int_t^\infty e^{-\gamma(\tau - t)} \tilde{r}(X(\tau), \pi(X(\tau)))\, \mathrm{d}\tau \tag{13}$$

where $\gamma > 0$ is the discount factor. Using a discounted performance function is crucial in the proposed formulation since, for the tracking problems, the system states follows a trajectory generated by the command generator model. As such, without the discount factor, the performance function becomes infinite because the control input incorporates a feed forward component preventing it from converging to zero as $t \to \infty$ [15].

The objective then becomes to learn a safe and optimal control policy that minimizes (13) while ensuring the forward invariance of the safe set $\mathscr{C}$.

Before addressing the problem of safe tracking, the following definition and assumption are made.

Definition 3 *The set of safe and admissible inputs, denoted as $\mathscr{U}_s$, for the current state x is defined as follows*

$$\mathscr{U}_s = \{u \in \mathscr{U} | x_u \in Int\mathscr{C}\}, \tag{14}$$

where $Int\mathscr{C}$ is the interior of the set defined in (2) and the state of the system evolved by the input u is represented as x_u.

Assumption 2 *There exists a safe feedback control policy u_0: $\mathscr{C} \times \mathbb{R}^p \to \mathscr{U}_s$ that stabilizes the augmented system (6) and the cost defined in (13) is finite.*

The primary objective of this paper is to minimize the value function as defined in (13), which serves as a key measure of performance and effectiveness in achieving safe and optimal tracking control. In the following section, the solution that enables to tackle this objective effectively is developed.

III. SAFE TRACKING PROBLEM

A. Safe HJB for tracking problem

Applying Leibniz's rule to (13), the following safe tracking Bellman equation is defined by

$$\dot{V} = -\tilde{r}(X,u) + \int_t^\infty \frac{\partial}{\partial t} e^{-\gamma(\tau - t)} \tilde{r}\Big(X(\tau), \pi(X(\tau))\Big)\, \mathrm{d}\tau. \tag{15}$$

Since the second term on the right hand side of (15) is equal to $\gamma V(X)$, it gives

$$\dot{V}(X) = -\tilde{r}(X,u) + \gamma V(X). \tag{16}$$

This leads to a safe tracking Lyapunov equation.

Definition 4 *The safe tracking Lyapunov equation (STLE), for nonlinear tracking problem, is defined by*

$$STLE(V,u) = \nabla V^T(\tilde{F}(X)+\tilde{G}(X)u)+\tilde{r}(X,u)-\gamma V(X)=0, \tag{17}$$

with ∇V denotes the gradient of the function V, which can be expressed as $\nabla V = \frac{\partial V}{\partial X}$.

The optimal policy, minimizing (13), is obtained by

$$u^* = \underset{u}{\operatorname{argmin}}[STLE(V,u)] = -\frac{1}{2}R^{-1}\tilde{G}^T(X)\nabla V^*(X) \tag{18}$$

where $V^*(X)$ the optimal cost function defined by

$$V^*(X) \quad = \quad \min_{\pi(.)} \int_t^\infty e^{-\gamma(\tau-t)}\tilde{r}\Big(X(\tau),\pi(X(\tau))\Big)\mathrm{d}\tau. \tag{19}$$

By substituting the optimal control (18) in STLE, (17) becomes the safe tracking Hamilton–Jacobi–Bellman (safe-THJB) equation

$$\begin{aligned} &H_{safe}(V^*(X)) \triangleq \\ &\quad \nabla V^{*T}(X)\tilde{F}(X)+X^T(\tau)\tilde{Q}X(\tau)+B_\vartheta(\rho X)-\gamma V^*(X) \\ &\qquad -\frac{1}{4}\nabla V^{*T}(X)\tilde{G}(X)R^{-1}\tilde{G}^T(X)\nabla V^*(X)=0. \end{aligned} \tag{20}$$

Assuming that there exists an optimal safe control policy, it implies the existence of an optimal safe value function satisfying the safe-THJB equation

$$H_{safe}(V^*(X))=0, \tag{21}$$

where $V^*(X)$ is a safe Lyapunov function (19) for the closed-loop augmented system (6).

Assumption 3 *There exists $V^* \in \mathscr{P}$, where $\mathscr{P}$ is the set of all functions in C^1 that are also positive definite and radially unbounded, such that the safe-THJB equation (21) holds.*

In the next section, policy iteration (PI) algorithm is introduced to solve the optimal safe tracking problem.

B. Safe tracking policy iteration algorithm

Given the analytical challenges to solve the nonlinear safe-HJB equation (21), the problem can be tackled by using the following safe tracking PI algorithm 1.

Algorithm 1 Safe Tracking Policy Iteration Algorithm

Initialization. Initialize u_0 with a safe and admissible policy such as $u_0 \in \mathscr{U}_s$.

Policy Evaluation. Update the value using $STLE(V_i,u_i)$

$$\nabla V_i^T(X)(\tilde{F}(X)+\tilde{G}(X)\pi_i(X))+\tilde{r}(X,\pi_i(X))-\gamma V_i(X)=0 \tag{22}$$

Policy Improvement. The control policy is improved by

$$\pi_{i+1}(X) = -\frac{1}{2}R^{-1}\tilde{G}^T(X)\nabla V_i(X) \tag{23}$$

The algorithm iteratively computes policy evaluation and policy improvement steps until the convergence to the optimal value V^* and its associated optimal policy u^*. The following theorem establishes the convergence property of the proposed safe tracking PI algorithm. To simplify the notation, let $u^*=\pi^*(X)$, $u_i=\pi_i(X)$ and $u_{i+1}=\pi_{i+1}(X)$.

Theorem 1 *Suppose Assumptions 2 and 3 hold, and the solution $V_i(X) \in C^1$ satisfying (22) exists for $i=0,1,\dots$. Then, the following properties hold $\forall i=0,1,\dots$.*

1) $V^*(X) \le V_{i+1}(X) \le V_i(X)\ \forall X \in \mathscr{C}\times\mathbb{R}^p$.
2) *Let $\lim_{i\to\infty} V_i(X_0) = V(X_0)$ and $\lim_{i\to\infty} u_i(X_0) = u(X_0)$, $\forall X_0 \in \mathscr{C}\times\mathbb{R}^p$. Then $X^*=X$ and $u^*=u$, if $V \in C^1$.*
3) *u_i stabilizes the error dynamics.*
4) *u_i is a safe policy, $u_i \in \mathscr{U}_s$.*

Due to space limitations, the proof of Theorem 1 is not included in this paper and will be provided in an extended version. Safe tracking PI algorithm is an effective method that allows to learn an optimal policy while ensuring safety. However, it does present some challenges that need to be addressed. One of the main challenges arises during the stage of data collection, where the state space is explored by adding exploration noise to the policy. While probing noise can provide important information about the system, it can also lead to violation of safety constraints. In the following section, this problem is addressed.

IV. SAFE OFF-POLICY FOR TRACKING PROBLEM

Off-policy methods rely on the incorporation of probing noise for exploration, and introducing such noise during the exploration phase carries inherent risks, particularly when its characteristics are unknown. This uncertainty can potentially lead to exploratory actions that result in undesired or unsafe system states.

A. Safe Exploration

For systems subject to bounded probing noise e_u, consider

$$\dot{X} = \tilde{F}(X)+\tilde{G}(X)u_{noisy}. \tag{24}$$

with $u_{noisy}=u_0+e_u$. The probing noise is assumed to not destabilize the system as denoted in the following assumption.

Assumption 4 *The closed-loop system (24) is input-to-state stable (ISS) when e_u is considered as input.*

The first family of CBFs (RCBFs), introduced in section 2, are unsuitable here since they are undefined and non-differentiable outside the safe set, failing to guarantee the ISSf condition under uncertainty. Therefore, we use TISSf-CBF (derived from Zeroing CBF [14]) to ensure system safety in the presence of unknown exploration noise. TISSf-CBFs also effectively handle model uncertainty by enforcing safety constraints even when the exact system dynamics are not fully known.

Definition 5 *[16] Let $a,b \in \mathbb{R}_{>0}$. A function $\alpha:(-b,a)\to\mathbb{R}$ that is continuous on $(-b,a)$ is said to be extended class-κ, if $\alpha(0)=0$ and $\alpha(r_1)<\alpha(r_2)$ for all $r_1,r_2\in(-b,a)$ satisfying $r_1<r_2$. The function α is said to be extended class-κ_∞, if $a=\infty$, $b=\infty$, $\lim_{r\to\infty}\alpha(r)=\infty$, and $\lim_{r\to-\infty}\alpha(r)=-\infty$.*

Definition 6 *[11] Let $\mathscr{C}\subseteq\mathscr{X}$ be the 0-superlevel set of a function $h:\mathscr{X}\to\mathbb{R}$ that is continuously differentiable on $\mathscr{X}$ with $\frac{\partial h}{\partial x}(x)=0_n$ when $h(x)=0$. The function h is said to be a TISSf-CBF for the system (1) on $\mathscr{C}$ if there exist extended class-κ_∞ function α with α^{-1} continuously differentiable on $\mathbb{R}$ and $\varepsilon:\mathbb{R}\to\mathbb{R}_{>0}$ that is continuously differentiable on*

$\mathbb{R}$ *such that:*

$$L_f h(x) + L_g h(x) u_0 - \frac{1}{\varepsilon(h(x))} L_g h(x) L_g h(x)^T \geq -\alpha(h(x)) \tag{25}$$

for all $x \in \mathscr{X}$*, and*

$$\frac{\partial \varepsilon}{\partial r}(r) \geq 0 \tag{26}$$

for all $r \in \mathscr{X}$.

By satisfying the condition of the TISSf-CBF, the safety of the policy can be assured by marginally modifying the unsafe policy. Thus, the exploration policy u_{noisy} can be adjusted by adding the solution u_{safe} of the following QP problem.

QP Problem: Find the additive control input u_{safe} that satisfies

$$\begin{aligned} &\min_{u_{safe}} \quad \frac{1}{2} u_{safe}^T u_{safe} \\ &L_F h(\rho X) + L_G h(\rho X)(u_0 + u_{safe}) \\ &- \frac{1}{\varepsilon(h(\rho X))} L_G h(\rho X) L_G h(\rho X)^T \geq -\alpha(h(\rho X)) \end{aligned} \tag{27}$$

The solution of the QP problem u_{safe} is crucial for ensuring policy safety within the off-policy algorithm. It allows for data collection not only within the safe set but also near the boundaries, improving the performance of the algorithm. Moreover, it is essential to note that the functions f and g must be explicitly known for the solution of the QP problem which render the approach model-based.

Remark 1 *To differentiate TISSf-CBF from ISSf-CBF, consider the role of* ε*. In ISSf-CBF,* ε *is constant, and small values lead to conservative exploration that may limit data collection near system boundaries. In contrast, TISSf-CBF makes* ε *a function of* $h(.)$*: as* $h(.)$ *approaches zero near boundaries,* ε *decreases to enforce safety, while within the safe set, larger* $h(.)$ *values increase* ε*, promoting broader exploration. This dynamic ensures both safety and effective exploration, as demonstrated in the simulation example.*

B. Safe Learning

The following system is considered subject to probing noise e_u

$$\dot{X} = \tilde{F}(X) + \tilde{G}(X) u_s, \tag{28}$$

with $u_s = u_{noisy} + u_{safe}$. Then, (28) can be expressed as

$$\dot{X} = \tilde{F}(X) + \tilde{G}(X) u_i + \tilde{G}(X) \nu_i \tag{29}$$

with $\nu_i = u_s - u_i$.

From (23), one has

$$\nabla V_i^T(X) \tilde{G}(X) = -2 u_{i+1}^T R. \tag{30}$$

Thus, for all $i \geq 0$, the time derivative of $V_i(X)$ along the solutions of (28) is obtained by

$$\begin{aligned} \dot{V}_i &= \nabla V_i^T(X)(\tilde{F}(X) + \tilde{G}(X) u_i + \tilde{G}(X) \nu_i) \\ &= -\tilde{r}(X, u_i) + \gamma V_i(X) - 2 u_{i+1}^T R \nu_i. \end{aligned} \tag{31}$$

Integrating both sides of (31) over any time interval $[t, t+T]$ yields to

$$\begin{aligned} &V_i(X(t+T)) - V_i(X(t)) \\ &= -\int_t^{t+T} \tilde{r}(X, u_i) - \gamma V_i(X) + 2 u_{i+1}^T R \nu_i \mathrm{d}t. \end{aligned} \tag{32}$$

Considering $\Omega \subseteq \mathbb{R}^{n+p}$ as a compact set, the value function V_i (corresponding to the critic neural network) and the control policy u_{i+1} (corresponding to the actor neural network) are approximated using the representation of basis function:

$$\tilde{V}_i(X) = W_i \tilde{\Phi}(X) \tag{33}$$

$$\tilde{u}_{i+1}(X) = U_i \tilde{\Psi}(X) \tag{34}$$

with $\tilde{\Phi} = [\tilde{\phi}_1, \tilde{\phi}_2 \dots \tilde{\phi}_{N_1}]^T$ and $\tilde{\Psi} = [\tilde{\psi}_1, \tilde{\psi}_2 \dots \tilde{\psi}_{N_2}]^T$, are the vectors of linearly independent smooth basis functions on Ω. $W_i \in \mathbb{R}^{1 \times N_1}$ and $U_i \in \mathbb{R}^{m \times N_2}$ are the matrices weights to be determined.

The weights W_i and U_i can be obtained by solving the following least-squares (LS) equation

$$\tilde{\Theta}_i^N \begin{bmatrix} vec(W_i) \\ vec(U_i^T) \end{bmatrix} = \tilde{E}_i^N \tag{35}$$

for $N > N_1 + m N_2$ and

$$\begin{aligned} \tilde{\Theta}_i^N &= [\tilde{\Theta}_i(t_1), \dots, \tilde{\Theta}_i(t_N)]^T \\ \tilde{E}_i^N &= [\tilde{E}_i(t_1), \dots, \tilde{E}_i(t_N)]^T \end{aligned} \tag{36}$$

where

$$\begin{aligned} \tilde{E}_i(t) = &-I_{\tilde{\Psi}\tilde{\Psi}}(U_{i-1}^T \otimes U_{i-1}^T) \mathrm{vec}(R) \\ &- \int_t^{t+T} (X^T \tilde{Q} X + B_{\vartheta}(\rho X)) \mathrm{d}t \end{aligned} \tag{37}$$

$$\tilde{\Theta}_i(t) = \begin{bmatrix} \left(\tilde{\Phi}(X(t+T)) - \tilde{\Phi}(X(t))\right)^T - \gamma I_{\tilde{\Phi}} \\ 2 I_{u\tilde{\Psi}}(R \otimes I_{N_2}) - 2 I_{\tilde{\Psi}\tilde{\Psi}}(U_{i-1}^T R \otimes I_{N_2}) \end{bmatrix}^T \tag{38}$$

with $I_{\tilde{\Psi}\tilde{\Psi}} = \int_t^{t+T} [\tilde{\Psi}^T(X) \otimes \tilde{\Psi}^T(X)] \mathrm{d}t$, $I_{\tilde{\Phi}} = \int_t^{t+T} (\tilde{\Phi}^T(X) \otimes I_{N_1}) \mathrm{d}t$ and $I_{u\tilde{\Psi}} = \int_t^{t+T} (u_s^T \otimes \tilde{\Psi}^T(X)) \mathrm{d}t$.

The approach proposed encapsulates three major aspects:

- **Safe Exploration**: Probing noise is added to the initial policy to explore the state space and collect rich data. The exploration policy is adjusted by adding the solution of the QP problem (27) in order to assure the safety of the system;
- **Safe Policy Iteration**: After collecting data, the LS equation (35) is solved and the policy evaluation and improvement steps are computed iteratively until the convergence of the critic weights;
- **Safe Operation**: The learned safe and optimal policy is used to generate the control input of the augmented system (6).

Remark 2 *The off-policy algorithm works in a model-free learning framework but relies on a model during exploration to enforce safety constraints. Any model uncertainty can risk safety boundary violations. By assuming unknown probing noise—equivalent to matched disturbances—the approach accommodates model uncertainty, ensuring safe exploration even without an accurate system model.*

To assess effectiveness the algorithm, simulation study over an academic system has been done leading to a thorough evaluation of the capabilities and suitability.

V. SIMULATION STUDY

Consider a nonlinear system described by the following differential equations

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -x_1^3 - 0.5 x_2 + u \end{aligned} \tag{39}$$

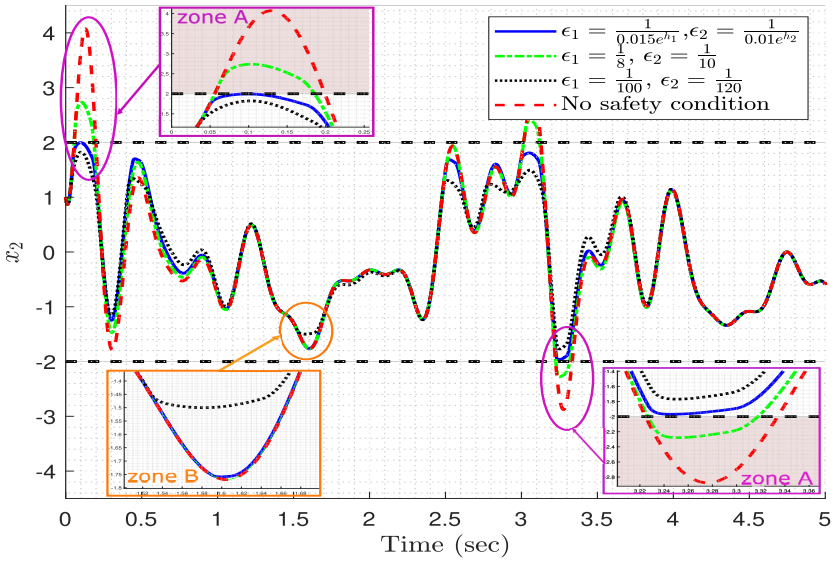

Fig. 1. Trajectory of x_2 during exploration.

A sinusoidal reference trajectory is generated by

$$\dot{x}_r = 0.5\sqrt{5}\cos(\sqrt{5}t) \tag{40}$$

Now, considering that only the states x_1 will be tracked, the error e_r, is defined as

$$e_r = x_1 - x_r \tag{41}$$

The safe set is described by $\mathscr{C} = \{x| \quad -2 < x_2 < 2\}$. The reward function (8) is considered with $Q = 8$, $R = 0.00001$ and $\gamma = 0.1$. The augmented state vector is defined as:

$$X = [X_1 \quad X_2 \quad X_3]^T = [x_1 \quad x_2 \quad e_r]^T$$

The CBF B_ϑ is given by

$$B_\vartheta(\rho X) = B_{1,\vartheta}(\rho X) + B_{2,\vartheta}(\rho X), \tag{42}$$

with

$$B_{1,\vartheta}(\rho X) = -\log\Big(\frac{\vartheta h_1(\rho X)}{\vartheta h_1(\rho X)+1}\Big),$$
$$B_{2,\vartheta}(\rho X) = -\log\Big(\frac{\vartheta h_2(\rho X)}{\vartheta h_2(\rho X)+1}\Big),$$

where $\vartheta = 80$, $\rho = \begin{bmatrix}0 & 1 & 0\end{bmatrix}$, $h_1(\rho X) = -X_2^{min} + X_2$ and $h_2(\rho X) = X_2^{max} - X_2$. From t = 0s to t = 5s, the exploration noise $e_u(t)$ is injected into the initial policy, with $e_u(t)$ being set to

$$e_u(t) = \sum_{k=1}^{12} 10\sin((2k-1)t) \tag{43}$$

The activation functions are considered, respectively, as $\tilde{\Phi}(X) = [X_1^2,\ X_2^2,\ X_3^2,\ X_1X_2,\ X_1X_3,\ X_2X_3,\ X_1^4,\ X_2^4,\ X_3^4,\ X_1^2X_2^2,\ X_1^2X_3^2,\ X_2^2X_3^2,\ X_1^3X_2,\ X_1^3X_3,\ X_2^3X_1,\ X_2^3X_3,\ X_3^3X_1,\ X_3^3X_2\ ]^T$

$\tilde{\Psi}(X) = \begin{bmatrix}X_1, X_2, X_3\end{bmatrix}^T$.

The weights of the critic and actor are trained by finding the solution of (35) for $N = 250$. The input and state data are collected over each interval of $T = 0.02s$. The initial weights of the actor are set to $U_0 = [-2 \quad -16 \quad -60]$.

Based on equation (27), the TISSf-CBF criteria is formulated as

$$\begin{aligned} &L_F h_1(\rho X) + L_G h_1(\rho X)(u_0 + u_{safe}) + \alpha_1(h_1(\rho X)) \\ &\quad - \frac{1}{\varepsilon_1(h_1(\rho X))} L_G h_1(\rho X) L_G h_1(\rho X)^T \geq 0 \\ &L_F h_2(\rho X) + L_G h_2(\rho X)(u_0 + u_{safe}) + \alpha_2(h_2(\rho X)) \\ &\quad - \frac{1}{\varepsilon_2(h_2(\rho X))} L_G h_2(\rho X) L_G h_2(\rho X)^T \geq 0 \end{aligned} \tag{44}$$

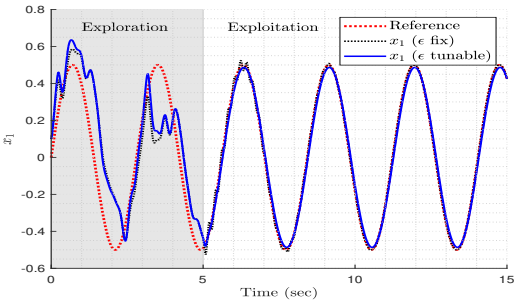

Fig. 2. Trajectory of x_1 during the operational phase under the learned policy.

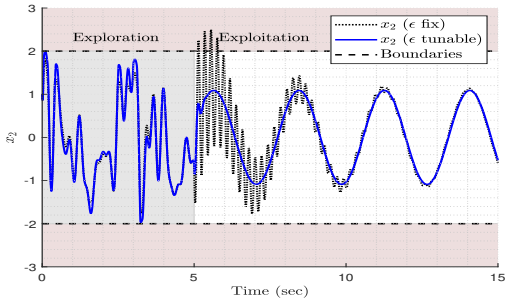

Fig. 3. Trajectory of x_2 during the operational phase under the learned policy.

with $\alpha_1 = 180h_1(\rho X)$, $\alpha_2 = 120h_2(\rho X)$. These values are fixed to ensure that the data are collected from both the safe region and from the vicinity of the safety boundary.

Fig.1 presents the trajectories of the state x_2 during the exploration phase. It displays four curves:

- the blue curve, where TISSf-CBF condition is satisfied for $\varepsilon_1 = \frac{1}{0.015e^{h_1}}$,$\varepsilon_2 = \frac{1}{0.01e^{h_2}}$;
- the green curve, where high constants are assigned to ε_1 and ε_2;
- the black curve where small constants are assigned to ε_1 and ε_2;
- the red curve is the case where no safety guarantees are taken into consideration, thus the exploration policy is not adjusted by the solution of the QP problem.

In this figure (Fig.1), two zones are also highlighted:

- **Zone A** illustrates scenarios where the exploration policy is unsafe when no safety consideration are taken into account. It can be seen that when ε is a function of h, data collection extends beyond the safe set to include the vicinity of its boundaries. In contrast, a high constant value of ε leads to violation of the safe set as x_2 crosses the boundaries of the set. However, with ε set to small values, the exploration is more conservative and data are collected from the safe set but they do not capture information from the vicinity of the boundaries.
- **Zone B** shows the scenario where x_2 is within the safe set and no policy adjustments are required. However, in the case where ε is a small constant, the policy is modified, pushing x_2 further within the safe set.

Figures 2 and 3 illustrate the trajectories of x_1 and x_2 during the exploration (0–5s) and operational phases. During exploration, an input is applied to gather data, after which the trained actor governs the system. As shown in Fig. 2, x_1 successfully tracks the reference, demonstrating the learned policy's efficacy. In Fig. 3, x_2 remains within the safe set during exploitation when the TISSf-CBF condition is met (blue curve), thanks to the boundary-penalizing B_ϑ . In contrast, with a small fixed ε (black curve), x_2 exceeds the safe boundaries, indicating an unsafe policy.

Discussion The exploration phase is critical because data quality directly influences the learned policy. Safety is ensured by collecting informative data both within the safe set and near its boundaries. As shown, the parameter ε governs exploration: a small constant value results in conservative exploration (confined to the safe set), while a large value can cause boundary violations. In TISSf-CBF, where ε is a function of h, near-boundary states (with small h) yield a reduced ε for safe exploration, whereas within the safe set (with large h) ε increases to allow broader exploration.

VI. CONCLUSION

This paper develops a novel approach for safe and optimal tracking control learning using an off-policy RL method. The approach guarantees safety during both exploration and exploitation phases. During exploration, probing noise is introduced to collect diverse, informative data while a QP problem is solved to enforce safety constraints via the TISSf-CBF condition. The safe tracking PI algorithm is iteratively computed to learn a policy that balances safety and optimality. Simulation results demonstrate that the algorithm generates safe policies even in the presence of probing noise, significantly reducing the error between the reference and the state. Although the proposed algorithm shows promising results in ensuring safety for tracking problems, future work will address the challenge of relying on a system model to solve the QP problem.

REFERENCES

[1] F. L. Lewis and D. Vrabie, "Reinforcement learning and adaptive dynamic programming for feedback control," *IEEE Circuits and Systems Magazine*, vol. 9, no. 3, pp. 32–50, 2009.

[2] R. Cheng, G. Orosz, R. M. Murray, and J. W. Burdick, "End-to-end safe reinforcement learning through barrier functions for safety-critical continuous control tasks," in *Proceedings of the AAAI conference on artificial intelligence.*, vol. 33, no. 01, 2019, pp. 3387–3395.

[3] Y. Yang, Y. Yin, W. He, K. G. Vamvoudakis, H. Modares, and D. C. Wunsch, "Safety-aware reinforcement learning framework with an actor-critic-barrier structure," in *2019 American Control Conference (ACC)*. IEEE, 2019, pp. 2352–2358.

[4] M. H. Cohen and C. Belta, "Safe exploration in model-based reinforcement learning using control barrier functions," *Automatica*, vol. 147, p. 110684, 2023.

[5] D. Panagou, D. M. Stipanović, and P. G. Voulgaris, "Distributed coordination control for multi-robot networks using lyapunov-like barrier functions," *IEEE Transactions on Automatic Control*, vol. 61, no. 3, pp. 617–632, 2015.

[6] S. Bandyopadhyay and S. Bhasin, "Safe q-learning for continuous-time linear systems," in *2023 62nd IEEE Conference on Decision and Control (CDC)*. IEEE, 2023, pp. 241–246.

[7] L. Zhao, K. Gatsis, and A. Papachristodoulou, "Stable and safe reinforcement learning via a barrier-lyapunov actor-critic approach," in *2023 62nd IEEE Conference on Decision and Control (CDC)*. IEEE, 2023, pp. 1320–1325.

[8] S. Kanso, M. S. Jha, and D. Theilliol, "Off-policy model-based end-to-end safe reinforcement learning," *International Journal of Robust and Nonlinear Control*, vol. 34, no. 4, pp. 2806–2831, 2024.

[9] Y. Hu, J. Fu, and G. Wen, "Safe reinforcement learning for model-reference trajectory tracking of uncertain autonomous vehicles with model-based acceleration," *IEEE Transactions on Intelligent Vehicles*, 2023.

[10] S. Kolathaya and A. D. Ames, "Input-to-state safety with control barrier functions," *IEEE control systems letters*, vol. 3, no. 1, pp. 108–113, 2018.

[11] A. Alan, A. J. Taylor, C. R. He, G. Orosz, and A. D. Ames, "Safe controller synthesis with tunable input-to-state safe control barrier functions," *IEEE Control Systems Letters*, vol. 6, pp. 908–913, 2021.

[12] M. Z. Romdlony and B. Jayawardhana, "Robustness analysis of systems' safety through a new notion of input-to-state safety," *International Journal of Robust and Nonlinear Control*, vol. 29, no. 7, pp. 2125–2136, 2019.

[13] H. Modares and F. L. Lewis, "Optimal tracking control of nonlinear partially-unknown constrained-input systems using integral reinforcement learning," *Automatica*, vol. 50, no. 7, pp. 1780–1792, 2014.

[14] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 8, pp. 3861–3876, 2016.

[15] B. Kiumarsi and F. L. Lewis, "Actor–critic-based optimal tracking for partially unknown nonlinear discrete-time systems," *IEEE transactions on neural networks and learning systems*, vol. 26, no. 1, pp. 140–151, 2014.

[16] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," in *2019 18th European control conference (ECC)*. IEEE, 2019, pp. 3420–3431.