Off-Policy Risk-Sensitive Reinforcement Learning-Based Constrained Robust Optimal Control

Cong Li, Qingchen Liu, Zhehua Zhou, Martin Buss, Fellow, IEEE, and Fangzhou Liu, Member, IEEE

Abstract—This article proposes an off-policy risk-sensitive reinforcement learning (RL)-based control framework to jointly optimize the task performance and constraint satisfaction in a disturbed environment. The risk-aware value function, constructed using the pseudo control and risk-sensitive input and state penalty terms, is introduced to convert the original constrained robust stabilization problem into an equivalent unconstrained optimal control problem. Then, an off-policy RL algorithm is developed to learn the approximate solution to the risk-aware value function. During the learning process, the associated approximate optimal control policy is able to satisfy both input and state constraints under disturbances. By replaying experience data to the off-policy weight update law of the critic neural network, the weight convergence is guaranteed. Moreover, online and offline algorithms are developed to serve as principled ways to record informative experience data to achieve a sufficient excitation required for the weight convergence. The proofs of system stability and weight convergence are provided. The simulation results reveal the validity of the proposed control framework.

Index Terms—Adaptive dynamic programming (ADP), input saturation, off-policy risk-sensitive reinforcement learning (RL), robust control, state constraint.

I. INTRODUCTION

RECENTLY adaptive dynamic programming (ADP), which emerges as a successful implementation of reinforcement learning (RL) in the control field, has been proposed to approximately solve regulation or tracking problems of continuous or discrete-time nonlinear systems with performance and/or robustness requirements, see [1], [2], and the references therein. The neural network (NN)-based approximate solution to the value function of the Hamilton–Jacobi–Bellman (HJB) equation facilitates the approximate optimal control strategy under an actor–critic learning structure. However, although traditional ADP has been widely adopted to tackle performance and robustness problems, input and state constraint satisfaction during the learning process, which is mainly investigated for safety concerns (e.g., restrictions on torques, joint angels, and angular velocities of robot manipulators), has not yet been efficiently addressed. Violations of any of them could lead to possible serious consequences such as damage to physical components. This motivates us to develop a constraint-satisfying ADP-based control strategy. Furthermore, for practical applications, the constraint-satisfying control strategy should guarantee desired performance even in an uncertain environment.

A. Prior and Related Works

Constrained ADP: In recent ADP-related works [3], [4], by incorporating a suitable nonquadratic functional into the value function, control limits have been addressed under an actor–critic learning structure, which leads to a limited approximate optimal control policy in the form of a bounded hyperbolic tangent function. However, comparing with traditional ADP where a quadratic functional is usually adopted to represent the desired performance regarding control efforts [5], the introduced nonquadratic functional leads to an inevitable performance compromise problem that is ignored in existing related works [3], [4]. In terms of restrictions on system states, most of previous ADP-related works adopt the system transformation technique to deal with state constraints under an actor–critic learning structure [6], [7], [8]. This method seeks for appropriate variable transformations that enable transformed system states to approach to infinity when potential state constraint violation happens. Therefore, approximate optimal control strategies that accomplish bounded transformed system states are constraint-satisfying control policies of the original system. The system transformation technique, nonetheless, is limited to simple constraint forms, e.g., restricted working space in a rectangular form. For certain state constraints, it may be infeasible to find suitable variable transformations to convert the constrained problem into an unconstrained counterpart. Besides, although general state constraints could be tackled by the well-designed penalty functions [9], [10], which become dominant in the optimization process when possible constraint violation happens and, thus, punish potential dangerous behaviors, no strict constraint satisfaction proofs.
are provided. However, in certain cases such as human–robot interaction scenarios, even the violation of safety-related constraints in a small possibility is unacceptable. Moreover, the tradeoff between constraint satisfaction and performance is ignored in [6], [7], [8], [9], and [10]. Methods that partially satisfy requirements of constraint or performance often lack practicability [11].

**Guaranteed Weight Convergence:** For the actor–critic learning structure adopted in the aforementioned ADP-related works, the interplay between actor and critic NNs is likely to cause instability because a wrong step taken by either of an actor or critic learning agent might adversely affect the other and finally destabilize the learning process [3], [9]. To solve this potential instability problem, a single critic learning structure is adopted in [12], [13], [14], and [15] where the approximated value function from the critic NN is directly used to construct the approximate optimal control policy. The prerequisite of using the approximated value function to construct the control strategy is the guaranteed weight convergence to the actual value. However, no proofs have been provided to declare the guaranteed weight convergence to the actual value in the single critic learning structure related works [12], [13], [14], [15]. Conventionally, the weight convergence is checked by the persistence of excitation (PE) condition [16]. Among existing ADP-related works [5], [17], the PE condition has been satisfied by directly adding external noises to inputs to encourage a sufficient exploration of the operation space. However, this method results in a lack of practicability given that the direct incorporation of external noises into control inputs may suffer a degradation of control performance, and a waste of energy, etc. Furthermore, in this method, the choice of explicit noise forms and the time to remove them during the online learning process highly depend on prior knowledge. Hence, for the guaranteed weight convergence, a more practical and easily implemented method to satisfy the PE condition is needed.

**Experience Replay:** Given the fact that the satisfaction of the PE condition implies that available data regarding unknown weights to be learned are rich enough during the learning period [18], a feasible way to get the desired rich data is to reuse past data generated during the learning process. Recently, experience replay (ER) emerges as an effective data-generating mechanism that supports online learning by replaying experience data [19], [20]. In ADP-related works [6], [21], [22], a fixed number of recently recorded transitions is replayed to actor or critic learning agents to achieve the weight convergence, which avoids the necessity of using additional noises to meet the required PE condition. However, for this sequent way of data usage, experience data with different richness levels are used without discrimination, and partial informative data (e.g., data from initial exploring phases) are only used once then abandoned immediately. These characteristics of data usage result in poor sample efficiency and the collected data might not be rich enough for the weight convergence. Such a sample deficiency problem also exists in uniform sampling-based ER techniques where all transitions are replayed at a same frequency regardless of their significance [20]. Unlike replaying experience data in a sequent or a uniform sampling way, prioritized ER (PER) is a more efficient technique that prioritizes data according to certain criteria [23]. Although all the aforementioned ER techniques have shown promise to support learning, their implementations often accompany with prior experience and extensive parameter tuning efforts [6], [21]. To the best of our knowledge, there exists no related works on principled ways to provide rich enough experience data to support the online leaning process.

### B. Contribution

This work focuses on an off-policy risk-sensitive RL-based control framework, as summarized in Fig. 1, for the control task of a nonlinear system under disturbances, input, and state constraints. Based on the pseudo control, risk-sensitive input, and state penalty terms, we transform a generally intractable constrained robust stabilization problem (CRSP) into an optimal control problem (OCP), which is then approximately solved by a single critic learning structure with an off-policy weight update law. The contribution of this article is threefold.

1. We incorporate novel risk-sensitive state penalty terms (RS-SP) into the value function to act as risk criteria during the learning process, which enables us to tackle state constraints in a long time horizon and preserve performance with constraint satisfaction proofs.

2. By exploiting experience data, we design an efficient off-policy critic NN weight update law that guarantees weight convergence without causing undesirable oscillations.

3. The online and offline experience buffer construction algorithms are proposed to provide the required rich enough data for the weight convergence.

The remainder of this article is organized as follows. Section II provides the formulation of a CRSP, the transformation to an OCP, and the transformation equivalence proof. Section III elucidates the approximated solution to the OCP and the off-policy weight update law. Besides, the critic NN
weight convergence proof and the system stability proof are provided. Simulation results shown in Section IV illustrate the effectiveness of the proposed control framework. Finally, Section V concludes this article.

Notations: Throughout this article, $\mathbb{R}$ ($\mathbb{R}^+$) denotes the set of real (positive) numbers; $\mathbb{N}^+$ denotes the set of positive natural numbers; $\mathbb{R}^n$ is the Euclidean space of $n$-dimensional real vector; $\mathbb{R}^{n \times m}$ is the Euclidean space of $n \times m$ real matrices; $I_{m \times n}$ represents the identity matrix with dimension $m \times m$; $\text{Int}(\mathbb{D})$ and $\partial \mathbb{D}$ denote the interior and boundary of the set $\mathbb{D}$, respectively; $\lambda_{\text{min}}(M)$ and $\lambda_{\text{max}}(M)$ are the minimum and maximum eigenvalues of a symmetric matrix $M$, respectively; $\text{diag}(a_1, \ldots, a_n)$ is the $n \times n$ diagonal matrix with the value of main diagonal as $a_1, \ldots, a_n$; The $i$th entry of a vector $x = [x_1, \ldots, x_n]^T \in \mathbb{R}^n$ is denoted by $x_i$, and $\|x\| = \sqrt{\sum_{i=1}^{n} |x_i|^2}$ is the Euclidean norm of the vector $x$; The $i$th entry of a matrix $D \in \mathbb{R}^{n \times m}$ is denoted by $d_{ij}$, and $||D|| = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} |d_{ij}|^2}$ is the Frobenius norm of the matrix $D$. For notational brevity, time-dependence is suppressed without causing ambiguity.

II. Problem Formulation

A. Formulation of Constrained Robust Stabilization Problem

Consider the continuous-time nonlinear dynamical system

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

(1)

where $x \in \mathbb{R}^n$ and $u(x) \in \mathbb{R}^m$ are the state and control inputs. $f(x) : \mathbb{R}^n \rightarrow \mathbb{R}^n$, $g(x) : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times m}$ are the known drift and input dynamics, respectively. $k(x) : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times r}$ represents the known differential system function. $d(x) : \mathbb{R}^n \rightarrow \mathbb{R}^r$ denotes the unknown additive disturbance. The general case that the additive disturbance is unmatched, $k(x) \neq g(x)$ in particular, is considered here. Assuming that $f(0) = 0$ and $d(0) = 0$, which means that the equilibrium point is $x = 0$.

Before proceeding, the following assumptions are provided, which are common in ADP-related works.

Assumption 1 [3]: $f(x) + g(x)u$ is Lipschitz continuous on a set $\Omega \subseteq \mathbb{R}^n$ that contains the origin, and the system is stabilizable on $\Omega$. There exists $g_M \in \mathbb{R}^+$ such that the input dynamics is bounded by $\|g(x)\| \leq g_M$.

Assumption 2 [24]: The unknown additive disturbance $d(x)$ is bounded by a known non-negative function $d_M(x) : \|d(x)\| \leq d_M(x)$, and $d_M(0) = 0$.

Based on the aforementioned settings, we formulate the CRSP as follows.

Problem 1 (CRSP): Given Assumptions 1 and 2, design a control strategy $u(x)$ to stabilize the system (1) to the equilibrium point under the additive disturbance $d(x)$, the input saturation

$$U_j := \{u_j \in \mathbb{R} : |u_j| \leq \beta, j = 1, \ldots, m\}$$

(2)

where $\beta \in \mathbb{R}^+$ is a known saturation bound; and the state constraints

$$X_i := \{x \in \mathbb{R}^n : h_i(x) < 0, i = 1, \ldots, n_c\}$$

(3)

where $X_i$ is a closed and convex set that contains the origin in its interior; $h_i(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ is a known continuous function that relates with the $i$th state constraint; $n_c \in \mathbb{N}^+$ is the number of considered state constraints.

B. Transformation to Optimal Control Problem

Problem 1 consists of three subproblems: 1) disturbance rejection; 2) input saturation; and 3) state constraint. It is nontrivial for ADP to directly deal with these subproblems together [1]. Thus, in this section, with the pseudo-control technique proposed in [24] and [25], reformulated risk-sensitive input penalty terms (RS-IP) based on [3], and our newly designed RS-SP, we first transform the CRSP clarified as Problem 1 into an equivalent OCP. Then, we attempt to solve the subproblems mentioned above simultaneously under an optimization framework.

1) Pseudo Control and Auxiliary System: As illustrated in [24], for a system suffering a matched disturbance, its disturbance-rejection control strategy could be designed by solving its nominal system’s OCP, wherein a value function including the square of the disturbance bound is considered. For the unmatched disturbance $k(x)d(x)$ considered in (1), however, the above robust control design strategy cannot be directly applied. Thus, to address the unmatched $k(x)d(x)$ under an optimization framework as well, it is first decomposed as [25]

$$k(x)d(x) = g(x)\bar{d}(x) + h(x)d(x)$$

(4)

where $\bar{d}(x) := g^T(x)k(x)d(x) : \mathbb{R}^n \rightarrow \mathbb{R}^m$, and $h(x) := (I - g(x)g^T(x))k(x) : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times r}$. Here, $\dagger$ denotes the Moore–Penrose inverse. Then, we introduce the following auxiliary system with a pseudo control $v(x) : \mathbb{R}^n \rightarrow \mathbb{R}^r$:

$$\dot{x} = f(x) + g(x)u(x) + h(x)v(x)$$

(5)

to accomplish that both $g(x)\bar{d}(x)$ and $h(x)d(x)$ are matched disturbances with respect to the range of $g(x)$ and $h(x)$, respectively. Finally, similar to the robust control design strategy proposed in [24], by solving the OCP of the auxiliary system (5) with a value function including the square of the bounds of $\bar{d}(x)$ and $d(x)$, we could address the disturbance-rejection problem of the system (1) under an optimization framework. The corresponding rigorous proof is provided later in Theorem 1 and the following assumption is introduced for the later analysis.

Assumption 3 [25]: The continuous function $h(x)$ is bounded as $\|h(x)\| \leq h_M$; $\bar{d}(x)$ is bounded by a non-negative function $I_M(x) : \|d(x)\| \leq I_M(x)$, and $l_M(0) = 0$.

2) Risk-Sensitive Input and State Penalty Terms: To tackle input and state constraints under an optimization framework, here, we follow the idea of risk-sensitive RL where multiple risk measures, e.g., high moment or conditional value at risk, are used to deal with constraints of Markov decision processes [26]. However, the available risk measures in the risk-sensitive RL field cannot guarantee strict constraint satisfaction and/or not efficient (even inappropriate) to address constraints of continuous-time nonlinear systems. Thus, we propose RS-IP in Definition 1 and RS-SP in Definition 2 as new risk measures during the learning process to enforce strict satisfaction of input and state constraints of continuous-time nonlinear systems.
Definition 1 (RS-IP): A continuous and differential function \( \phi(u) \) is an RS-IP if it has the following properties.
1. A bounded monotonic odd function with \( \phi(0) = 0 \).
2. The first-order partial derivatives of \( \phi(u) \) are bounded.

Here, the RS-IP term is a reformulation of the nonquadratic functional used in [3] and [12] to confront input constraints.

Definition 2 (RS-SP): Given the closed region \( X_i \), \( i = 1, \ldots, n_c \), defined as (3), a continuous scalar function \( S_i(x) : X_i \rightarrow \mathbb{R}, i = 1, \ldots, n_c \), is an RS-SP if the following proprieties hold.
1. \( S_i(0) = 0 \), and \( S_i(x) > 0 \ \forall x \neq 0 \).
2. \( S_i(x) \rightarrow \infty \) if \( x \) approaches \( \partial X_i \).
3. For initial value \( x(0) \in \text{Int}(X_i) \), there exists \( s \in \mathbb{R}^+ \) such that \( S_i(x(t)) \leq s \ \forall t \geq 0 \) along solutions of the dynamics.

Comparing with similar works [9], [10] that use state penalty functions to tackle state constraints but without strict constraint satisfaction proofs, our proposed RS-SP term enables us to provide strict constraint satisfaction proofs in Theorem 1. Here, the novel RS-IP term is inspired by the so-called barrier Lyapunov function [27]. The first point of Definition 2 denotes that \( S_i(x) \) is an effective Lyapunov function candidate, which enables \( S_i(x) \) to serve as part of the Lyapunov function for the system stability proof. The last two points imply that \( \inf_{x \to \partial X} S_i(x) = \infty \) and \( \inf_{x \in \text{Int}(X)} S_i(x) \geq 0 \), which means that \( S_i(x) \) serves as a barrier certificate for an allowable operating region \( X_i \).

3) Optimal Control Problem: Based on the auxiliary system (5) and Definitions 1 and 2, an equivalent OCP of \( x \) is the distance from the state \( L_i \) et al.: OFF-POLICY RISK-SENSITIVE RL-BASED CONSTRAINED ROBUST OPTIMAL CONTROL 2481

Theorem 1. Here, the novel RS-IP term is inspired by the so-called barrier Lyapunov function [27]. The first point of Definition 2 denotes that \( S_i(x) \) is an effective Lyapunov function candidate, which enables \( S_i(x) \) to serve as part of the Lyapunov function for the system stability proof. The last two points imply that \( \inf_{x \to \partial X} S_i(x) = \infty \) and \( \inf_{x \in \text{Int}(X)} S_i(x) \geq 0 \), which means that \( S_i(x) \) serves as a barrier certificate for an allowable operating region \( X_i \).

Problem 2 (OCP): Given Assumptions 1–3, consider the auxiliary system (5), find \( u(x) \) and \( v(x) \) to minimize the value function

\[
V(x(t)) := \int_t^\infty r(x(\tau), u(x(\tau)), v(x(\tau))) d\tau
\]

where the utility function follows \( r(x, u(x), v(x)) := r_d(x) + \rho v^T(x)v(x) + r_c(x, u(x)) \) with \( \rho \in \mathbb{R}^+ \), \( r_d(x) := \rho R_d(x) + \rho^2 \Sigma^R_d(x) \), and \( r_c(x, u(x)) := W(u(x)) + L(x) \).

The input penalty function \( W(u(x)) \) follows:

\[
W(u(x)) := \sum_{j=1}^m 2 \int_0^{\beta_j} \beta_j R_j \phi^{-1} (\beta_j / \beta) \ d\beta_j
\]

where \( \phi(\cdot) \) is the RS-IP term in Definition 1; \( R_j \) is the \( j \)-th diagonal element of a positive-definite diagonal matrix \( R \in \mathbb{R}^{m \times m} \). The state penalty function \( L(x) \) is defined as

\[
L(x) := x^T Q x + \sum_{i=1}^{n_c} k_i S_i(x)
\]

where \( Q \in \mathbb{R}^{n \times n} \) is a positive-definite matrix; \( k_i \) is the risk sensitivity parameter that follows \( k_i = 1/(1 + d_i^2) \), where \( d_i \) is the distance from the state \( x \) to the boundary of \( h_i(x) \); \( S_i(\cdot) \) is the RS-SP term in Definition 2 for \( i \)-th state constraint.

Unlike ADP-related works [3], [9] that incorporate non-quadratic functionals to tackle input saturation but without considering control effort related performance, \( W(u(x)) \) in (7) could take into consideration of requirements for both control limits and control energy expenditures by choosing a suitable matrix \( R \). Much more details are introduced in Section II-C1. The common used risk-neutral quadratic function \( x^T Q x \) [1], [2] (capturing the desired state performance) is augmented with the newly designed weighted RS-SP term \( \sum_{i=1}^{n_c} k_i S_i(x) \) (addressing multiple state constraints) to construct \( L(x) \) in (8), which enables us to consider state-related performance and constraints together. The incorporation of \( S_i(x) \) into \( L(x) \) deteriorates the desired performance represented by \( x^T Q x \). Therefore, we propose the risk sensitivity parameter \( k_i \), which relates with the distance from the constraint boundary, to specify the inevitable tradeoff between the state-related performance and constraint satisfaction during the learning process. Note that this kind of tradeoff is ignored in existing related works [9], [10]. The detailed mechanism of \( L(x) \) is illustrated in Section II-C2.

C. Mechanism of Input and State Penalty Functions

The mechanism of \( W(u(x)) \) and \( L(x) \) to enable the learning process to preserve performance without violating strict input/state constraint satisfaction is detail clarified here.

1) Mechanism of Input Penalty Function \( V(u(x)) \): By Definition 1, the explicit form of the RS-IP term is chosen as \( \phi(\cdot) = \tanh(\cdot) \) [3], [14]. Given the inevitable tradeoff between input-related performance and constraint satisfaction, \( W(u(x)) \) is designed to address input constraints (2) and approximate \( u^T R u \) (a common desired performance criterion for control efforts) simultaneously, where \( R \in \mathbb{R}^{m \times m} \) is a prior-chosen positive-definite matrix reflecting designers’ preferences.

The mechanism of \( W(u(x)) \) to tackle input constraints is clarified from two perspectives, see Fig. 2(a) and (b), respectively. In the first perspective, input constraints are considered in a long time horizon. \( W(u(x)) \) in (7) is an integration of \( \beta R_j \tanh^{-1} (u_j / \beta) \) that is denoted as \( z_1 \) in Fig. 2(a). When any \( u_j, j = 1, \ldots, m \), approaches to the input constraint boundaries \( \pm \beta \), it follows that the value of \( W(u(x)) \) will be infinity. Since the optimization process aims to minimize the value function, the resulting optimal control strategy will be away from \( \pm \beta \); Otherwise, a high value of the value function occurs. From the other perspective, according to the later result in (12), the resulting optimal control strategy based on \( W(u(x)) \) is in a form of \( \tanh(\cdot) \) whose boundness enforces strict satisfaction of input constraints, as shown in Fig. 2(b).

The construction of \( W(u(x)) \) to reflect the desired performance for control energy is shown in Fig. 2(c). Consider \( W_j(u_j) \), the \( j \)-th summand of \( W(u(x)) \). It follows that:

\[
W_j(u_j) := 2 \beta R_j u_j \tanh^{-1} (u_j / \beta) + \beta^2 R_j \log \left( 1 - u_j^2 / \beta^2 \right).
\]

As displayed in Fig. 2(c), by adjusting the value of \( R_j \), \( W_j(u_j) \) approximates the desired control energy criterion \( u_j^T R u_j \) well.

Based on the above discussion, we know that \( W(u(x)) \) in (7) is able to tackle input constraints while preserving performance concerning control energy expenditures.
Mechanism of State Penalty Function $L(x)$: According to Definition 2, when a potential state constraint violation happens, the corresponding RS-SP term will approach to infinity. Since the optimal control strategy aims to minimize the total cost, states will be pushed away from the direction where a high value of the RS-SP based $L(x)$ occurs. Thus, the state constraint violation is avoided. To satisfy Definition 2, we choose $S_i(x) = \log(h_i(x))$ here. Note that the explicit form of $\log(h_i(x))$ is adjusted based on given state constraints, which is exemplified later. For a better explanation of the mechanism of the RS-SP term $S_i(x)$ and the corresponding risk sensitivity parameter $k_i$, we present a four-dimensional system example with the safe regions defined as $X_1 = \{x_1, x_2 \in \mathbb{R} : h_1(x_1, x_2) = x_1^2 + x_2^2 - 1 < 0\} [28]$, $X_2 = \{x_3 \in \mathbb{R} : h_2(x_3) = |x_3| - 2 < 0\}$, and $X_3 = \{x_4 \in \mathbb{R} : h_3(x_4) = |x_4| - 3 < 0\} [6]$. The corresponding RS-SP terms are designed as $S_1(x_1, x_2) = \log(\alpha(x_2)/\alpha(x_2 - x_1^2))$ with $\alpha(x_2) = 1 - x_2^2$, $S_2(x_3) = \log(4/(4 - x_3^2))$, and $S_3(x_4) = \log(9/(9 - x_4^2))$, respectively. As displayed in Fig. 3(a) and (b), these RS-SP terms act as barriers at constraint boundaries and confine states remain in the safe regions. This inherent risk-sensitive property enables us to tackle state constraints under an optimization framework. As long as initial states lie in the safe regions and the value function is always bounded as time evolves, the subsequent state evolution will be restricted to the safe regions. From Fig. 3(c) and (d), we know that the role of $S_i(x)$ will be discouraged by $k_i$ when states are far away from the boundary of $h_i(x)$. Therefore, the state-related performance is maintained when no state constraint violation occurs.

D. HJB Equation for OCP

Aiming at the transformed OCP in Problem 2, for any admissible control policies $u, v \in \Psi(\Omega)$, where $\Psi(\Omega)$ is the admissible control set [3, Definition 1], the associated optimal
value function follows:
\[ V^\pi(x(t)) := \min_{u \in \Psi(\Omega)} \int_0^\infty r(x(\tau), u(x(\tau)), v(x(\tau))) d\tau \] (10)
and the HJB equation satisfies
\[ 0 = \min_{u \in \Psi(\Omega)} \left[ \nabla V^\pi(x) + g(x)u(x) + h(x)v(x) \right. \\
\left. + r(x, u(x), v(x)) \right] \] (11)
where \( \nabla (\cdot) := \partial (\cdot)/\partial x \). Assuming that the minimum on the right side of (11) exits and is unique [5]. Then, the closed forms of the optimal control policies \( u^\pi(x) \) and \( v^\pi(x) \) are obtained as [3]
\[ u^\pi(x) = -\frac{1}{2\beta} \tanh \left( \frac{1}{2\beta} R^{-1} g(x) \nabla V^\pi(x) \right) \] (12)
\[ v^\pi(x) = \frac{1}{2\rho} h^T(x) \nabla V^\pi(x). \] (13)

E. Equivalence Between Problem 1 and Problem 2
Here, we defer a detailed explanation of the method to get the optimal control policies (12) and (13) in Section III, and focus now on the proof of equivalence between Problem 1 in Section II-A and Problem 2 in Section II-B. Comparing with the result provided in [24] that merely considers additive disturbances, as shown in Theorem 1, the additional consideration of input and state constraints further complicates the theoretical analysis.

Theorem 1: Consider the system described by (1) and controlled by the optimal control policy (12). Suppose Assumptions 1–3 hold and the initial states and control inputs lie in the predefined constraint satisfying sets (2) and (3). The optimal control policy (12) guarantees robust stabilization of system (1) without violating input constraint (2) and state constraint (3), if there exists a scalar \( \epsilon_1 \in \mathbb{R}^+ \) such that the following inequality is satisfied:
\[ L(x) > 2\rho v^\pi(x) u^\pi(x) + \epsilon_1. \] (14)

Proof: The detailed proof is clarified in Appendix A.

Remark 1: The inequality (14) serves as one criterion to check the equivalence between Problem 1 and Problem 2. This inequality provides guidelines to debug parameters and helps to check the rightness of the solution. In particular, a feasible parameter set and solution should make the inequality (14) establish. A practical example is provided in Fig. 5(d) of Section IV-B. Note that \( \epsilon_1 \) in (14) is a state-related term according to (38), rather than a constant.

It is proven in Theorem 1 that the CRSP (Problem 1) is equivalent to the OCP (Problem 2) under the inequality (14). Thus, in order to solve the original CRSP, the current task is to obtain the optimal control law (12) focusing on the transformed OCP, which is detail clarified in the next section.

III. APPROXIMATE SOLUTIONS TO OCP
To learn the approximate solution to the OCP, instead of introducing a common actor–critic learning structure used in [3] and [5], here, we adopt a single critic learning structure which enjoys lower computation complexity [12]. Furthermore, departing from traditional methods that directly add additional noises into inputs to meet the PE condition required for the critic NN weight convergence [5], [17], here we exploit experience data to construct the off-policy weight update law to achieve a sufficient excitation required for the critic NN weight convergence. Additionally, an online PER algorithm and an offline experience buffer construction algorithm are proposed as principled ways to provide the required rich enough experience data.

A. Value Function Approximation
According to the Weierstrass high-order approximation theorem [29], there exists a weighting matrix \( W^* \in \mathbb{R}^N \) such that the continuous value function is approximated as
\[ V^\pi(x) = W^*^T \Phi(x) + \epsilon(x) \] (15)
for \( x \in \Omega \) with \( \Omega \) being a compact set, where \( \Phi(x) : \mathbb{R}^n \to \mathbb{R}^N \) is the NN activation function in a polynomial form, and \( \epsilon(x) \in \mathbb{R} \) is the approximation error. Denote \( \nabla \Phi \in \mathbb{R}^{N \times n} \) and \( \nabla \epsilon \in \mathbb{R}^N \) as partial derivatives of \( \Phi(x) \) and \( \epsilon(x) \), respectively. As \( N \to \infty \), both \( \epsilon(x) \) and \( \nabla \epsilon \) converge to zero uniformly. Without loss of generality, the following assumption is given.

Assumption 4 [5]: There exist constants \( b_\epsilon, b_{\Phi} \) such that \( \| \epsilon(x) \| \leq b_\epsilon, \| \nabla \epsilon \| \leq b_{\Phi} \), and \( \| \nabla \Phi \| \leq b_{\Phi} \).

For fixed admissible control policies \( u(x) \) and \( v(x) \), inserting (15) into (11) yields the Lyapunov equation (LE)
\[ W^*^T \nabla \Phi(f(x) + g(x)u(x) + h(x)v(x)) + r(x, u(x), v(x)) = \epsilon_h \] (16)
where the residual error follows \( \epsilon_h := -\nabla \epsilon(x)(f(x) + g(x)u(x) + h(x)v(x)) \in \mathbb{R} \). According to Assumption 1, the system dynamics is Lipschitz. This leads to the bounded residual error, i.e., there exists \( b_{\epsilon_h} \in \mathbb{R}^+ \) such that \( \| \epsilon_h \| \leq b_{\epsilon_h} \).

Unlike the common analysis and derivation process in well-known ADP-related works [1], [2], here, we rewrite the NN parameterized LE (16) into a linear in parameter (LIP) form that follows:
\[ \Theta = -W^*^T Y + \epsilon_h \] (17)
where \( \Theta := r(x, u(x), v(x)) \in \mathbb{R}, \) and \( Y := \nabla \Phi(f(x) + g(x)u(x) + h(x)v(x)) \in \mathbb{R}^N \). Note that both \( \Theta \) and \( Y \) are able to be obtained from real-time data \( x, u(x), \) and \( v(x) \).

Given the LIP form and the measurable \( Y, \Theta \), in (17), from the perspective of adaptive control, we transform the learning of the critic NN weight \( W^* \) into a parameter estimation problem of an LIP system, where \( Y \) and \( W^* \) are treated as the regressor matrix and the unknown parameter vector of an LIP system, respectively. This novel transformation enables us to design a simple weight update law with weight convergence guarantee in Section III-B.

B. Off-Policy Weight Update Law
The ideal critic NN weight \( W^* \) in (17) is approximated by an estimated weight \( \hat{W} \) which satisfies the following relation:
\[ \hat{\Theta} = -\hat{W}^T Y \] (18)
where \( \hat{\Theta} \in \mathbb{R} \) is the estimated utility function. Denoting the weight estimation error as \( \hat{W} := \hat{W} - W^* \in \mathbb{R}^N \). Then, we get
\[
\hat{W} = -\Gamma k_Y \hat{\Theta} - \sum_{l=1}^{P} \Gamma k_Y Y_l \hat{\Theta}_l
\] (20)
where \( \hat{\Theta} := \Theta - \hat{\Theta} = \hat{W}^\top Y + \epsilon_h \).

To achieve \( \hat{W} \rightarrow W^* \) and \( \hat{\Theta} \rightarrow \epsilon_h \), \( \hat{W} \) should be updated to minimize \( E := (1/2)\hat{W}^\top \hat{W} \). Furthermore, in order to guarantee the weight convergence while minimizing \( E \), here, we exploit experience data to support the online learning process. The utilized experience data could achieve the sufficient excitation required for the weight convergence. This depart from related works \([5], [17]\) that incorporate external noises to satisfy the PE condition. Finally, we design a simple yet efficient off-policy weight update law of the critic NN that follows:
\[
\hat{W} = -\Gamma k_Y \hat{\Theta} - \sum_{l=1}^{P} \Gamma k_Y Y_l \hat{\Theta}_l
\] (20)
where \( \hat{\Theta} := \Theta - \hat{\Theta} = \hat{W}^\top Y + \epsilon_h \).

By observing (24), the size of \( \Omega_{\hat{W}} \) relates with the bound of \( \epsilon_{er} \). As \( N \rightarrow \infty \), we know that \( \epsilon_h \rightarrow 0 \) results in \( \epsilon_{er} \rightarrow 0 \). Then, we get \( \hat{V}_{er} \leq -\lambda_{min}(B)\|\hat{W}\|^2 \), i.e., \( \hat{W} \rightarrow 0 \) exponentially as \( t \rightarrow \infty \). Equivalently, it is guaranteed that \( \hat{W} \) converges to \( W^* \). Finally, in conjunction with (12) and (13), the approximate optimal control policies are obtained as
\[
\hat{u}(x) = -\beta \tanh \left( \frac{1}{2\beta} R^{-1} g^\top(x) \nabla \Phi^\top(x) \hat{W} \right)
\] (25)
\[
\hat{v}(x) = -\frac{1}{2\rho} h^\top(x) \nabla \Phi^\top(x) \hat{W}.
\] (26)

In the following part, the main conclusions are provided based on the off-policy weight update law (20) and the approximate optimal control policies (25), (26).

**Theorem 3:** Consider the dynamics (5), the off-policy weight update law of the critic NN in (20), and the control policies (25) and (26). Given Assumptions 1–5, for sufficiently large \( N \), the approximate control policies (25) and (26) stabilize the system (5). Moreover, the critic NN weight learning error \( \hat{W} \) is uniformly ultimately bounded (UUB).

Proof: The detailed proof is referred to in Appendix B. Assumption 5 in Theorem 3 is the prerequisite to ensure that \( \hat{W} \) converges to \( W^* \). The guaranteed weight convergence enables us to directly apply \( \hat{W} \) in (20) to construct the approximate optimal control policies (25) and (26). Assumption 5 is not restrictive and could be easily satisfied by the algorithms proposed in the next section.

**C. Experience Replay With Online/Offline Experience Buffers**

To get rich enough experience data to satisfy Assumption 5, given the sampling deficiency problem of the sequential way of data usage in existing ADP-related works \([6], [21], [22]\) and inspired by the concurrent learning (CL) technique developed for system identification \([19]\), here, we design both online and offline principled methods to provide the sufficient rich experience data. These recorded informative experience data are then replayed to the weight update laws (20), (27) to achieve the required excitation for the guaranteed critic NN weight convergence.

1) **Online PER Algorithm:** Before the estimated weight converges (lines 4 and 5), Algorithm 1 chooses the minimum eigenvalue \( \lambda_{min}(\mathbb{B}) \) as the priority scheme to filter experience data \( Y \) and \( \hat{\Theta} \) recorded into the experience buffers \( \mathbb{B} \) and \( \mathcal{E} \), respectively. Here, the prioritized criterion is different
Algorithm 1 Online PER Algorithm

Input: Iteration index: \( n_r \); Buffer size: \( P \); Threshold: \( \xi \).

Output: Experience buffers: \( \mathcal{B}, \mathcal{E} \).

1: if \( n_r \leq P \) then
2: Record current \( Y, \Theta \) into \( \mathcal{B}, \mathcal{E} \) respectively.
3: else
4: if \( \| W_{n_r} - W_{n_r-1} \| > \xi \) then
5: Record prioritized \( Y, \Theta \) leading to high \( \lambda_{\text{min}}(\mathcal{B}) \).
6: else
7: Record current \( Y, \Theta \) sequentially to update \( \mathcal{B}, \mathcal{E} \).
8: end if
9: end if

from ones used in existing PER algorithms [23]. We prefer experience data accompanied with a larger \( \lambda_{\text{min}}(\mathcal{B}) \) given the facts that: 1) a nonzero \( \lambda_{\text{min}}(\mathcal{B}) \) ensures that \( \text{rank}(\mathcal{B}) = N \) in Assumption 5 holds [19], [31], i.e., the convergence of \( \hat{W} \) to \( W^* \) is guaranteed and 2) according to (23) and (24), a larger \( \lambda_{\text{min}}(\mathcal{B}) \) leads to a faster weight convergence rate and a smaller residual set. Although efficient, the priority scheme \( \lambda_{\text{min}}(\mathcal{B}) \) accompanies with additional computation loads. Thus, once the convergence is achieved (lines 6 and 7), i.e., we have obtained sufficient excitation, we alternate to a low-cost mode where recent data are sequentially recorded. This kind of cyclic replacement way of data usage enjoys the low-cost mode where recent data are sequentially replaced with online counterparts. The simple three-layer NNs adopted in this article provides us with opportunities to revisit the ER technique and investigate principled ways to exploit experience data to support the online learning process. This is difficult in deep RL field because the complexity of deep NNs hinders researchers from understanding the mechanism of the ER technique [20]. Algorithms 1 and 2 are plug-in methods with little extra computations and engineering efforts. We argue that they also provide alternative solutions for the parameter convergence problem existing in the adaptive control field.

Algorithm 2 Offline Experience Buffer Construction Algorithm

Input: \( A = [A, \overline{A}] \) with \( A, \overline{A} \in \mathbb{R}^d \); Mesh size: \( d \in \mathbb{R}^d \), or data point number: \( c \in \mathbb{R}^n \); Empty sets: \( \mathcal{X} \in \mathbb{R}^{n \times d} \).

Output: \( F; \mathcal{G}; \mathcal{H}; \mathcal{K}; \mathcal{R}; P; \).

1: Sampling: \( \mathcal{X}^* = \prod_{i=1}^n (\mathcal{A}_i - \mathcal{A}_i^c) \cap \prod_{i=1}^n \mathcal{E}_i^c \).
2: Data collection: \( F \leftarrow \nabla \Phi^T(\mathcal{X})y(\mathcal{X}); \mathcal{R} \leftarrow r(\mathcal{X}); \mathcal{G} \leftarrow \nabla \Phi^T(\mathcal{X})g(\mathcal{X}); \mathcal{H} \leftarrow \nabla \Phi^T(\mathcal{X})h(\mathcal{X}); \mathcal{K} \leftarrow L(\mathcal{X}) \).

The implementation of using offline recorded experience data to support the online learning process enjoys two advantages: 1) the rank condition in Assumption 5 is easily satisfied and 2) the possible influence of data noises is offset by averaging. It is worth mentioning that the mere exploitation of offline recorded data cannot tackle a dynamic environment well. Thus, during the online operation, the offline experience data recorded into experience buffers \( F, \mathcal{G}, \mathcal{H}, \mathcal{K}, \mathcal{R} \) will be sequentially replaced with online counterparts.

Remark 2: The simple three-layer NNs adopted in this article provides us with opportunities to revisit the ER technique and investigate principled ways to exploit experience data to support the online learning process. This is difficult in deep RL field because the complexity of deep NNs hinders researchers from understanding the mechanism of the ER technique [20]. Algorithms 1 and 2 are plug-in methods with little extra computations and engineering efforts. We argue that they also provide alternative solutions for the parameter convergence problem existing in the adaptive control field.

IV. NUMERICAL SIMULATION

This section provides three simulation examples to validate the effectiveness of our developed off-policy weight update laws (20), (27), the approximate optimal control policy (25), and Algorithms 1 and 2. First, we consider an optimal regulation problem (ORP) of a nonlinear system [5] in Section IV-A. This ORP serves as a benchmark problem to prove that both Algorithm 1-based (20) and Algorithm 2-based (27) enable the estimated critic NN weight to converge to the actual value, which is marginally considered in existing single critic learning structure related works [12], [13]. Then, to show the superiority of our proposed off-policy risk-sensitive RL-based control framework to counter input and state constraints under additive disturbances, a pendulum system [32] is investigated in Section IV-B. Furthermore, a reach task of a robot manipulator [33] is considered in Section IV-C to validate the real-time performance of our proposed control framework.

A. Example 1: ORP of Benchmark Nonlinear System

To validate that based on our proposed off-policy weight update laws (20), (27), and Algorithms 1 and 2, the estimated weight guarantees convergence to the actual optimal value, a benchmark problem [5] is investigated here. Note that only an ORP without considering disturbances nor input/state constraints is investigated here. Otherwise, the actual optimal value of the NN weight is unknown. The

\[ \dot{\hat{W}} = -\Gamma_{k_e} Y \Theta - \frac{1}{P} \sum_{l=1}^P \Gamma_{k_e} Y_l \Theta_l, \quad (27) \]
benchmark continuous-time nonlinear system is given as

$$\dot{x} = f(x) + g(x)u, \quad x \in \mathbb{R}^2$$

where $f(x) := [-x_1 + x_2, -0.5x_1 - 0.5x_2(1 - \cos(2x_1 + 2))]^\top$, $g(x) := [0, \cos(2x_1) + 2]^\top$. The standard quadratic value function follows:

$$V(x) := \int_0^{\infty} x^\top Q x + u^\top Ru \, dt$$

where $Q = I_{2 \times 2}$ and $R = 1$. Following the method in [34], we obtain the optimal value function as $V^* = 0.5x_1^2 + x_2^2$. Thus, the optimal weight follows $W^* = [0.5, 0, 1]^\top$ by choosing the activation function as $\Phi(x) = [x_1^2, x_1x_2, x_2^2]^\top$. Initial values are set as $x(0) = [1, 1]^\top$, $\hat{u}(0) = 0$. For the off-policy weight update law (20) based on Algorithm 1, we choose $P = 5$, $k_c = 1$, $k_e = 1$, $\Gamma = \text{diag}(2, 1.4, 1)$, and $\xi = 10^{-3}$. It is displayed in Fig. 4(a) that after 1 s, the estimated weight converges to

$$\hat{W}_1 = [0.5040, 0.0592, 1.0625]^\top.$$  

Regarding the weight update law (27) under Algorithm 2, we start with constructing offline experience buffers by sampling 10 data points separately for $x_1 \in A_1 = [-2, 2]$, $x_2 \in A_2 = [-4, 4]$. Besides, $P = 100$, $k_e = 1$, $k_c = 1$, and $\Gamma = \text{diag}(5, 0.5, 0.01)$ are chosen during the online operation. As displayed in Fig. 4(b), the estimated weight converges to

$$\hat{W}_2 = [0.50721, -0.0417, 0.9783]^\top.$$  

Thus, it is concluded that the weight update laws (20), (27) under Algorithms 1 and 2 ensure that $\hat{W}$ converges to $W^*$ with a fast speed without incorporating external noises to satisfy the PE condition.

### B. Example 2: CRSP of Pendulum System

This section investigates the CRSP (Problem 1) of the pendulum system used in [32] to validate the effectiveness and efficiency of our proposed control framework shown as Fig. 1. The state-space model of the investigated system follows:

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

where $f(x) := [x_2, -4.9 \sin x_1 - 0.2x_2]^\top$, $g(x) := [0, 0.25]^\top$, $k(x) := [1, -0.2]^\top$, and $d(x) := \omega_1 x_1 \sin(\omega_2 x_2)$. During the simulation, $\omega_1$ and $\omega_2$ are randomly chosen within the interval $[-\sqrt{2}/2, \sqrt{2}/2]$ and $[-2, 2]$, respectively. Thus, $\|d(x)\| \leq \sqrt{2}/2 \|x\|$ and $\|g(x)k(x)d(x)\| \leq 0.4\sqrt{2} \|x\|$ establishes, and Assumptions 2 and 3 are satisfied by choosing $d_M(x) = \sqrt{2}/2 \|x\|$ and $l_M(x) = 0.4\sqrt{2} \|x\|$. Here, the input and state constraints are considered as $U = \{u \in \mathbb{R} : |u| \leq \beta\}$ and $X_1 = \{x_1 \in \mathbb{R} : h_1(x_1) = |x_1| - \alpha_1 < 0\}$, $X_2 = \{x_2 \in \mathbb{R} : h_2(x_2) = |x_2| - \alpha_2 < 0\}$, where $\beta = 1.5$, $\alpha_1 = 2.01$, and $\alpha_2 = 4$. The corresponding auxiliary system follows:

$$\dot{x} = f(x) + g(x)u + h(x)v$$  \hspace{1cm} (28)

where $h(x) = [1, 0]^\top$. Let $\rho = 0.1$, $k_i := 1/(1 + (\alpha_i - |x_i|)^2)$, $i = 1, 2$, $Q = I_{2 \times 2}$, and $R = 1$. For the CRSP of the pendulum, the value function of (28) follows:

$$V_c(x) := \int_0^{\infty} \Phi(x) + \|x\|^2 dt$$

where $\Phi(x) := 2\beta R u \tanh(u/\beta) + \beta^2 R \log(1 - u^2/\beta^2)$.

For comparison, a robust optimization problem (ROP) for the pendulum without considering input and state constraints is also investigated here. Regarding the ROP case, the common quadratic value function given in [24] is recalled here. By setting $Q_o = I_{2 \times 2}$ for the $x^1, Q_o x$ term and $R_o = 1$ for the $u^1, R_o u$ term, the value function of (28) follows:

$$V_{o,c}(x) := \int_0^{\infty} 1.37\|x\|^2 + u^\top u + 0.1v^\top v \, dt.$$  

The activation function for the pendulum is chosen as $\Phi(x) = [x_1^2, x_1x_2, x_2^2, x_1x_1, x_2x_2]^\top$. For the off-policy weight update law (20) based on Algorithm 1, $P = 9$, $k_c = 0.01$, $k_e = 0.001$, $\Gamma = I_{6 \times 6}$, and $\xi = 10^{-3}$ are chosen. The initial values are set as $x(0) = [2, -2], \hat{u}(0) = 0$, $\hat{v}(0) = 0$, and $\hat{W} = [1, 1, 1, 1, 1, 1]^\top$. Note that the initial values of states are purposely set to be near the constraint boundaries to highlight the efficacy of our method. Simulation results for the pendulum system are shown as follows.

The evolution trajectory of the estimated critic NN weight of the CRSP case is shown in Fig. 5(a). It is observed that after 15 s, the convergence result of $\hat{W}$ achieves. As displayed in Fig. 5(b), the control trajectory of the CRSP case illustrates that the RS-IP term enables the satisfaction of input constraints. However, for the ROP case, the input constraint boundary is violated. The phase portrait of states is provided in Fig. 5(c) where the cyan rectangle represents the state constraint boundary. It is observed that the states $x_1, x_2$ asymptotically converge to the equilibrium point and always lie in the state constraint set for the CRSP case. However, for the ROP case, the predefined state constraint is violated. The simulation results displayed in Fig. 5 have verified that the resulting approximate optimal control policy by solving Problem 2 can efficiently deal with Problem 1. To get more insight of the equivalence between Problems 1 and 2, the verification of the inequality (14) is displayed in Fig. 5(d), where the inequality (14) holds after $t = 15$ s.

### C. Example 3: CRSP of Robot Manipulator

To further demonstrate the real-time performance of our proposed control framework, a reach task of a 2-DoF robot manipulator is considered here. In particular, a robot starting from multiple initial positions is driven to reach the desired
point under disturbances, input, and state constraints. The model of the robot manipulator follows [33]:

\[ M(q) \ddot{q} + C(q, \dot{q}) \dot{q} = \tau \]

where \( q := [q_1, q_2]^\top \in \mathbb{R}^2, \) \( c_2 := \cos q_2, \) \( s_2 := \sin q_2, \) \( M(q) := \begin{bmatrix} p_1 + 2p_3 c_2 & p_2 + p_3 c_2 \\ p_2 & p_2 \end{bmatrix} \in \mathbb{R}^{2 \times 2}, \) and
\[
C(q, \dot{q}) := \begin{bmatrix} -p_3 q_2 s_2 & -p_3 (\dot{q}_1 + \dot{q}_2) s_2 \\ p_3 q_1 s_2 & 0 \end{bmatrix} \in \mathbb{R}^{2 \times 2}, \quad p_1 = 3.473 \text{ kg m}^2, \quad p_2 = 0.196 \text{ kg m}^2, \quad p_3 = 0.242 \text{ kg m}^2.
\]

Let \( x := [x_1, x_2, x_3, x_4]^\top = [q_1, q_2, \dot{q}_1, \dot{q}_2]^\top \in \mathbb{R}^4, \) the robot dynamics could be written in the state-space form as (1), where \( f(x) := [x_3, x_4, (M^{-1}(-C)x_3, x_4)]^\top \in \mathbb{R}^4, \) and \( g(x) := [[0, 0]^\top, [0, 0]^\top, (M^{-1})^\top]^\top. \) Besides, with \( k(x) := [(1, 0)^\top, (0, 1)^\top, 0_{2 \times 2}]^\top, \) we assume that the robot suffers a disturbance \( d(x) := [\delta_1 x_1 \sin(x_2), \delta_2 x_2 \cos(x_1)]^\top, \) where \( \delta_1, \delta_2 \in [-1, 1]. \) Given \( \|d(x)\| \leq \|x\|, \) and \( d'(x)k(x)d(x) = 0, \)

Assumptions 2 and 3 are satisfied by setting \( \delta_M(x) = \|x\|, \) and \( \|I_M(x)\| = 0. \) The input constraints are considered as \( \mathbb{U}_j : [u_j \in \mathbb{R} : |u_j| \leq 3], j = 1, 2. \) The state constraint in a rectangular form has been considered in Section IV-B. Thus, to show the effectiveness of the proposed RS-SP term to deal with general state constraints, a circular state constraint \( X_3 := \{x \in \mathbb{R}^2 : h_3(x_1, x_2) = x_1^2 + x_2^2 - 1 < 0\} \) is considered here. The solution procedure for the robot manipulator’s CRSP is similar to Section IV-B, which is omitted here for space limits. To approximate the value function well, we choose the activation function as \( \Phi(x) = [x_1^2, x_2^2, x_3^2, x_4^2, x_1 x_2, x_1 x_3, x_1 x_4, x_2 x_3, x_2 x_4, x_3 x_4]^\top, \) and the parameters for the weight update law (20) are set as \( P = 15, \) \( k_r = 0.2, \) \( k_e = 0.01, \) \( \Gamma = I_{10 \times 10}, \) and \( \xi = 10^{-3}. \)

To fully demonstrate the effectiveness of our method to address state constraints even under input saturation and disturbances, the robot joint trajectories under multiple initial positions are shown in Fig. 6. We observe that the robot is driven to reach the desired point (i.e., zero point) while obeying the predefined state constraints. It is shown that when the states approach to the constraint boundary, they will be driven back to safe states under our proposed method. As displayed in Fig. 7, the weight convergence result achieves at \( t = 12 \) s under our proposed weight update law (20). The aforementioned simulation results based on a 2-DoF robot manipulator validate that the off-policy weight update law (20) and the approximate optimal control policy (25) fulfill real-time requirements for practical applications.

V. CONCLUSION

An off-policy risk-sensitive RL-based control framework is proposed to stabilize a nonlinear system that subjects to additive disturbances, input, and state constraints. The pseudo control and the resulting auxiliary system are first introduced to address additive disturbances under an optimization framework. Then, risk-sensitive input and state penalty terms, incorporated into the value function as optimization criteria, allow us to tackle both input and state constraints in a long time horizon, which helps to avoid abrupt changes of control inputs that are unfavourable for the online learning process. The transformed OCP is approximately solved by a single critic learning structure with our developed off-policy weight update law. The adopted single critic learning structure leads to computation simplicity and eliminates approximation errors.
caused by an actor NN. The exploitation of experience data to guarantee the weight convergence enables the proposed control strategy to be applicable to practical applications. Multiple numerical comparison simulations validate the effectiveness of our developed control framework. The proposed control framework requires the knowledge of a nominal dynamics, which is not always available in practical applications. The future work aims to develop a low-cost model-free control strategy while preserving rigorous stability analysis.

APPENDIX A
PROOF OF THEOREM 1

Proof: 1) Proof of Stability: As for $V^*(x)$ defined as (10), we know that when $x = 0$, $V^*(x) = 0$, and $V^*(x) > 0$ for $\forall x \neq 0$. Thus, it could serve as a Lyapunov function candidate for stability proofs. Taking time derivative of $V^*(x)$ along the system (1) yields

\[
\dot{V}^* = \nabla V^* \cdot (f(x) + g(x)u^*(x) + k(x)d(x)) + \sum_{j=1}^{m} R_j (\tanh (u^*(x)/\beta)) g(x)k(x)d(x).
\]

By setting $\rho = \beta R$, (34), (35), (36), and (37) into (33), we have

\[
\dot{V}^* = -L(x) + 2\rho \nabla^T (v^*)(v^*)(x) + \epsilon_t,
\]

where $\epsilon_t := b_{\epsilon t}$. Based on Assumption 2, 2) Proof of Input and State Constraint Satisfaction: Denote $V^*(0)$ the value of the Lyapunov function candidate $V^*$ at $t = 0$. According to the definition of admissible control policies, $V^*(0)$ is a bounded function. If $L(x) > 2\beta R (v^*)^2 + \epsilon_t$, $V^* < 0$ establishes, which means that $V^*(x) < V^*(0) \forall t$. The boundness of $V^*(t)$ implies that state constraints will not be violated; Otherwise, $V^*(t) \to \infty$ if any state constraint violations happen according to Definition 2. Since the hyperbolic tangent function satisfies $-1 \leq \tanh(x) \leq 1$, the optimal control policy in (12) follows $-\beta \leq u^*(x) \leq \beta$, i.e., inputs are confined into the safety set $\mathcal{S}$. The proof provided here means that the optimal control policy $u^*(x)$ for the system (1) guarantees satisfaction of both constraints in terms of system states and control inputs.

APPENDIX B
PROOF OF THEOREM 3

Proof: Consider the following candidate Lyapunov function:

\[
J := V^*(x) + \frac{1}{2} \bar{W}^\top \Gamma^{-1} \bar{W}.
\]

Taking time derivative of (39) along the system (5) yields

\[
\dot{J} = \dot{L}_V + \dot{L}_W
\]

where $\dot{L}_V := \dot{V}^*(x)$ and $\dot{L}_W := \hat{V}^* \dot{b}_t \Gamma^{-1} \hat{W}^\top$. The first term $\dot{L}_V$ follows:

\[
\dot{L}_V = \nabla V^* \cdot (f(x) + g(x)\dot{u}(x) + h(x)\dot{v}(x)) + \nabla^2 V^* \cdot (\dot{u}(x) + h(x)v(x)) + \nabla^2 V^* \cdot (\dot{v}(x) - u(x)) \nabla^2 V^* \cdot (\dot{v}(x) - v(x)).
\]
According to (30), (31), and (32), (41) is rewritten as
\[
\hat{L}_V = -\mathcal{L}(x) - \nabla \mathcal{V}(u^*(x)) - \rho \nabla v^*(x) \mathcal{T} v^*(x) - \hat{I}_M^2(x) - 2\beta R \tan^{-1}(u^*(x)/\beta) (\hat{u}(x) - u^*(x)) - 2\rho \hat{v}^*(x) (\hat{v}(x) - v^*(x)).
\]
(42)

Besides, we get
\[
-2\beta R \tan^{-1}(u^*(x)/\beta) (\hat{u}(x) - u^*(x)) \\
\leq \beta^2 \| R \tan^{-1}(u^*(x)/\beta) \|^2 + \| \hat{u}(x) - u^*(x) \|^2 \\
\leq \beta^2 \sum_{j=1}^{m} R_j^2 \left( \tan^{-1}(u_j^*(x)/\beta) \right)^2 + \| \hat{u}(x) - u^*(x) \|^2. 
\]
(43)

Based on (34)-(38), the following equation also establishes:
\[
-\nabla \mathcal{V}(u^*(x)) - 2\beta \tan^{-1}(u^*(x)/\beta) (\hat{u}(x) - u^*(x)) \\
\leq \beta^2 \sum_{j=1}^{m} \left( R_j^2 - R_j \right) \left( \tan^{-1}(u_j^*(x)/\beta) \right)^2 + b_e, \\
+ \| \hat{u}(x) - u^*(x) \|^2 \leq \epsilon_3 \| \hat{u}(x) - u^*(x) \|^2. 
\]
(44)

Substituting (44) into (42) yields
\[
\hat{L}_V \leq -\mathcal{L}(x) - \rho \nabla v^*(x) \mathcal{T} v^*(x) - \hat{I}_M^2(x) - 2\rho \nabla v^*(x) + \epsilon_3 \\
+ \| \hat{u}(x) - u^*(x) \|^2 - 2\beta \nabla v^*(x) (\hat{v}(x) - v^*(x)) \\
= -\mathcal{L}(x) - \hat{I}_M^2(x) - \rho \hat{d}_M^2(x) + \epsilon_3 - \rho \hat{v}^*(x) \hat{v}(x) \\
+ \| \hat{u}(x) - u^*(x) \|^2 + \rho \| \hat{v}(x) - v^*(x) \|^2. 
\]
(45)

As for \(\rho \hat{v}^*(x) \hat{v}(x)\) in (45), according to (26)
\[
\rho \hat{v}^*(x) \hat{v}(x) = \frac{1}{4\rho} \hat{W}^\top \nabla \Phi(x) h(x) h^\top(x) \nabla \Phi^\top(x) \hat{W} \\
= \frac{1}{4\rho} (\hat{W}^\top \nabla \Phi(x) h(x) h^\top(x) \nabla \Phi^\top(x)) (\hat{W} - \hat{W}) \\
= \frac{1}{4\rho} \hat{W}^\top \mathcal{H} \hat{W} + \frac{1}{2\rho} \hat{W}^\top \mathcal{H} \hat{W} 
\]
(46)

where \(\mathcal{H} := \nabla \Phi(x) h(x) h^\top(x) \nabla \Phi^\top(x)\).

As for \(\rho \| \hat{v}(x) - v^*(x) \|^2\) in (45), according to (26)
\[
\rho \| \hat{v}(x) - v^*(x) \|^2 = \rho \| \frac{1}{2\rho} h(x) \nabla \Phi^\top(x) \hat{W} \|^2 \\
= \frac{1}{4\rho} \hat{W}^\top \mathcal{H} \hat{W}. 
\]
(47)

For simplicity, denote \(\nabla \Phi^\top(x) \hat{W} = \mathcal{O}(\phi)\) as follows:
\[
\tan(\phi) = \tan(\phi) + \frac{\partial \tan(\phi)}{\partial \phi}(\phi - \phi) + \mathcal{O}(\phi^2) \\
= \tan(\phi) - \frac{1}{2\beta} (I_{m \times m} - \mathcal{D}(\phi)) R^{-1} g^\top(x) \\
\nabla \Phi^\top(x) \hat{W} + \mathcal{O}(\phi^2) 
\]
(48)

where \(\mathcal{D}(\phi) := \text{diag}(\tanh(\phi), \ldots, \tanh(\phi_m))\). \(O((\phi^* - \hat{\phi})^2)\) is a higher-order term of the Taylor series. By following [36, Lemma 1], the higher-order term is bounded as
\[
\| \mathcal{O}(\phi^* - \hat{\phi})^2 \| \leq 2\sqrt{m} + \frac{1}{2} R^{-1} \| g_M b_{\phi x} \| \hat{W}. 
\]
(49)

Using (12), (25), and (48), we get
\[
\hat{u}(x) - u^*(x) = \beta (\tan(\phi^*) - \tan(\phi)) + \epsilon_u^* \\
= -\frac{1}{2} \| I_{m \times m} - \mathcal{D}(\phi) \| R^{-1} g^\top(x) \nabla \Phi^\top(x) \hat{W} + \beta \| \mathcal{O}(\phi^* - \hat{\phi})^2 \| + \epsilon_u^* 
\]
(50)

where \(\epsilon_u^* := \beta \tan(1/2\beta) R^{-1} g^\top(x) (\nabla \Phi^\top(x) W^* + \nabla \phi) - \beta \tan(1/2\beta) R^{-1} g^\top(x) \nabla \Phi^\top(x) W^*\), assuming that it is bounded by \(\| \epsilon_u^* \| \leq \beta_2^*\).

As for \(\| \hat{u}(x) - u^*(x) \|^2\) in (45), since \(\| I_{m \times m} - \mathcal{D}(\phi) \| \leq 2\sqrt{m}\), combining (49) with (50), we get
\[
\| \hat{u}(x) - u^*(x) \|^2 \leq 3\beta^2 \| \mathcal{O}(\phi^* - \hat{\phi})^2 \| + 3 \| \epsilon_u^* \|^2 \\
+ 3\left( -\frac{1}{2} \| I_{m \times m} - \mathcal{D}(\phi) \| R^{-1} g^\top(x) \nabla \Phi^\top(x) \hat{W} \right)^2 \\
\leq 6 \| R^{-1} \| g_M^2 b_{\phi x} \| \hat{W}^2 + 12\beta \| \hat{W} \| + 12\beta \sqrt{m} \| R^{-1} \| g_M b_{\phi x} \| \hat{W}. 
\]
(51)

Substituting (46), (47), and (51) into (45) yields
\[
\hat{L}_V \leq -\mathcal{L}(x) - \hat{I}_M^2(x) - \rho \hat{d}_M^2(x) - \frac{1}{4\rho} \mathcal{H} \mathcal{W} + \epsilon_3 + 6 \| R^{-1} \| g_M^2 b_{\phi x} \| \hat{W} \| \\
+ 12\beta \| \hat{W} \| + 12\beta \sqrt{m} \| R^{-1} \| g_M b_{\phi x} \| \hat{W}. 
\]
(52)

As for the second term \(\hat{L}_W\), based on (20) and (22)
\[
\hat{L}_W \leq -\hat{W}^\top \mathcal{H} \mathcal{W} + \hat{W}^\top \epsilon_r. 
\]
(53)

Finally, as for \(J\), substituting (52) and (53) into (40), and based on the fact that \(\| W^* \| \leq b_{\phi w}, \| \nabla \Phi(x) \| \leq b_{\phi x}, \| h(x) \| \leq h_M\), we get
\[
J \leq -\mathcal{L}(x) - \hat{I}_M^2(x) - \rho \hat{d}_M^2(x) - \frac{1}{4\rho} \mathcal{H} \mathcal{W} + \epsilon_3 + 6 \| R^{-1} \| g_M^2 b_{\phi x} \| \hat{W} \| \\
+ 12\beta \| \hat{W} \| + 12\beta \sqrt{m} \| R^{-1} \| g_M b_{\phi x} \| \hat{W} \| + \epsilon_3 \\
\leq -\mathcal{L}(x) - \hat{I}_M^2(x) - \rho \hat{d}_M^2(x) - \frac{1}{4\rho} \mathcal{H} \mathcal{W} + \epsilon_3 + 6 \| R^{-1} \| g_M^2 b_{\phi x} \| \hat{W} \| \\
+ 12\beta \| \hat{W} \| + 12\beta \sqrt{m} \| R^{-1} \| g_M b_{\phi x} \| \hat{W} \| + \epsilon_3 \\
= -\mathcal{A} - \mathcal{B} \| \hat{W} \| + C \| \hat{W} \| + D 
\]
(54)
where $M := \varepsilon_\rho - (1/2\rho)W^T \mathcal{K}^\rho$, and there exists $b_M := b_{v_\rho} + (1/2\rho)b_2^x h_2^x h_3^x b_{W^\rho} \in \mathbb{R}^+$ such that $\|M\| \leq b_M$; $A := \mathcal{L}(x) + \bar{l}_2^x(x) + \mu \bar{l}_2^x(x) + (1/4\rho)W^T \mathcal{K}^\rho W^\rho$ is positive definite; $B := \min(B) - 6\rho \nu_1^2 \|S^M_{\mathcal{K}}\| + C := 12\beta M^2 \rho \nu_1^2 \|S^M_{\mathcal{K}}\| + b_M$ and $\mathcal{D} := 12m^2 + 3b^2 + \varepsilon$. Let the parameters be chosen such that $B > 0$. Since $A$ is positive definite, the above Lyapunov derivative is negative if

$$\|\tilde{W}\| < \frac{C}{2B} + \sqrt{\frac{C^2}{4B^2} + \frac{D}{B}}.$$  

(55)

Thus, the critic weight learning error converges to the residual set defined as

$$\hat{\Omega}_W := \left\{ \frac{\tilde{W}}{\|\tilde{W}\|} \leq \frac{C}{2B} + \sqrt{\frac{C^2}{4B^2} + \frac{D}{B}} \right\}.$$  

(56)

REFERENCES

[1] D. Liu, S. Xue, B. Zhao, B. Luo, and Q. Wei, “Adaptive dynamic programming for control: A survey and recent advances,” IEEE Trans. Syst., Man, Cybern., Syst., vol. 51, no. 1, pp. 142–160, Jan. 2021.

[2] F. L. Lewis and D. Vrabie, “Reinforcement learning and adaptive dynamic programming for feedback control,” IEEE Circuits Syst. Mag., vol. 9, no. 3, pp. 32–50, 3rd Quart., 2009.

[3] M. Abu-Khalaf and F. L. Lewis, “Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HIBP approach,” Automatica, vol. 41, no. 5, pp. 779–791, 2005.

[4] B. Sun and E.-J. van Kampen, “Reinforcement-learning-based adaptive optimal flight control with output feedback and input constraints,” J. Guid. Control Dyn., vol. 44, no. 9, pp. 1–8, 2021.

[5] K. G. Vamvoudakis and F. L. Lewis, “Online actor–critic algorithm to solve the continuous-time infinite horizon optimal control problem,” Automatica, vol. 46, no. 5, pp. 878–888, 2010.

[6] Y. Yang, D.-W. Ding, H. Xiong, Y. Yin, and D. C. Wunsch, “Online barrier–actor–critic learning for H∞ control with full-state constraints and input saturation,” J. Franklin Inst., vol. 357, no. 6, pp. 3316–3344, 2020.

[7] P. D. Lellis and M. He, “Data-driven approximated optimal control for chemical processes with state and input constraints,” Complexity, to be published.

[8] J. Sun and C. Liu, “Disturbance observer-based robust missile autopilot design with full-state constraints via adaptive dynamic programming,” J. Franklin Inst., vol. 355, no. 5, pp. 2344–2368, 2018.

[9] J. Na, B. Wang, G. Li, S. Zhan, and W. He, “Nonlinear constrained optimal control of wave energy converters with adaptive dynamic programming,” IEEE Trans. Ind. Electron., vol. 66, no. 10, pp. 7904–7915, Oct. 2019.

[10] M. Abu-Khalaf, J. Huang, and F. L. Lewis, Nonlinear H2/H∞ Constrained Feedback Control: A Practical Design Approach Using Neural Networks. London, U.K.: Springer, 2006.

[11] J.-J. E. Slotine and W. Li, Applied Nonlinear Control, vol. 199. Englewood Cliffs, NJ, USA: Prentice-Hall 1991.

[12] X. Yang, Y. Zhou, N. Dong, and Q. Wei, “Adaptive critics for decentralized stabilization of constrained-input nonlinear interconnected systems,” IEEE Trans. Syst., Man, Cybern., Syst., vol. 52, no. 7, pp. 4187–4199, Jul. 2022.

[13] X. Yang and H. He, “Event-triggered robust stabilization of nonlinear input-constrained systems using single network adaptive critic designs,” IEEE Trans. Syst., Man, Cybern., Syst., vol. 50, no. 9, pp. 3145–5157, Sep. 2020.

[14] K. Zhang, R. Su, H. Zhang, and Y. Tian, “Adaptive resilient event-triggered control design of autonomous vehicles with an iterative single critic learning framework,” IEEE Trans. Neural Netw. Learn. Syst., vol. 32, no. 12, pp. 5502–5511, Dec. 2021.

[15] M. Ha, D. Wang, and D. Liu, “Discounted iterative adaptive critic designs with novel stability analysis for tracking control,” IEEE/CAA J. Autom. Sinica, vol. 9, no. 7, pp. 1262–1272, Jul. 2022.

S. Boyd and S. S. Sastry, “Necessary and sufficient conditions for parameter convergence in adaptive control,” Automatica, vol. 22, no. 6, pp. 629–639, 1986.

S. Bhasin, R. Kamalapurkar, M. Johnson, K. G. Vamvoudakis, F. L. Lewis, and W. E. Dixon, “A novel actor–critic–identifier architecture for approximate optimal control of uncertain nonlinear systems,” Automatica, vol. 49, no. 1, pp. 82–92, 2013.

P. A. Ioannou and J. Sun, Robust Adaptive Control, New York, NY, USA: Courier Corp., 2012.

G. V. Chowdhary, “Concurrent learning for convergence in adaptive control without persistence of excitation,” Ph.D. dissertation, Dept. Aerosp. Eng., Georgia Inst. Technol., Atlanta, GA, USA, 2010.

W. Fedus et al., “Revisiting fundamentals of experience replay,” in Proc. Int. Conf. Mach. Learn., 2020, pp. 3061–3071.

R. Kamalapurkar, P. Walters, and W. E. Dixon, “Model-based reinforcement learning for approximate optimal regulation,” in Control of Complex Systems. Amsterdam, The Netherlands: Elsevier, 2016, pp. 247–273.

L. R. G. Carrillo and K. G. Vamvoudakis, “Deep-learning tracking for autonomous flying systems under adversarial inputs,” IEEE Trans. Aerosp. Electron. Syst., vol. 56, no. 2, pp. 1444–1459, Apr. 2020.

T. Schaul, J. Quan, I. Antonoglou, and D. Silver, “Prioritized experience replay,” 2015, arXiv:1511.05952.

F. Lin, R. D. Brandt, and J. Sun, “Robust control of nonlinear systems: Compensating for uncertainty,” Int. J. Control, vol. 56, no. 6, pp. 1453–1459, 1992.

F. Lin and R. D. Brandt, “An optimal control approach to robust control of robot manipulators,” IEEE Trans. Robot. Autom., vol. 14, no. 1, pp. 69–77, Feb. 1998.

Y. Shen, W. Stannat, and K. Obermayer, “Risk-sensitive Markov control processes,” SIAM J. Control Optim., vol. 51, no. 5, pp. 3652–3672, 2013.

K. P. Tee, S. S. Ge, and E. H. Tay, “Barrier Lyapunov functions for the control of output-constrained nonlinear systems,” Automatica, vol. 45, no. 4, pp. 918–927, 2009.

L. Wang, E. A. Theodorou, and M. Egerstedt, “Safe learning of quadrotor dynamics using barrier certificates,” in Proc. IEEE Int. Conf. Robot. Autom. (ICRA), 2018, pp. 2460–2465.

R. Courant and D. Hilbert, Methods of Mathematical Physics, vol. 1. New York, NY, USA: Wiley, 1953.

G. Tao, Adaptive Control Design and Analysis, vol. 37. New York, NY, USA: Wiley, 2003.

S. J. Leon, I. Bica, and T. Hohn, Linear Algebra: With Applications, vol. 6. Upper Saddle River, NJ, USA: Prentice-Hall, 1998.

D. Liu and Q. Wei, “Policy iteration adaptive dynamic programming algorithm for discrete-time nonlinear systems,” IEEE Trans. Neural Netw. Learn. Syst., vol. 25, no. 3, pp. 621–634, Mar. 2014.

C. Li, F. Liu, Y. Wang, and M. Buss, “Concurrent learning-based adaptive control of an uncertain robot manipulator with guaranteed safety and performance,” IEEE Trans. Syst., Man, Cybern., Syst., vol. 52, no. 5, pp. 3299–3313, May 2022.

V. Nevistić and J. A. Primbs, Constrained Nonlinear Optimal Control: A Converse HJB Approach, California Inst. Technol., Pasadena, CA, USA, 1996.

C. Li, Q. Liu, Z. Zhou, M. Buss, and F. Liu, “Off-policy risk-sensitive reinforcement learning based constrained robust optimal control,” 2020, arXiv:2006.05681.

X. Yang, D. Liu, H. Ma, and Y. Xu, “Online approximate solution of HJI equation for unknown constrained-input nonlinear continuous-time systems,” Inf. Sci., vol. 328, pp. 435–454, Jan. 2016.
Qingchen Liu received the Ph.D. degree in system and control from Australian National University, Canberra, ACT, Australia, in 2018. From 2018 to 2019, he worked as a Postdoctoral Research Fellow with Australian National University. From 2019 to 2021, he was an EuroTech Research Fellow with the Chair of Information-Oriented Control, Technical University of Munich, Munich, Germany. He is currently a Research Associate Professor with the Department of Automation, University of Science and Technology of China, Hefei, China. His research interest includes networked systems and multirobotics systems.

Zhehua Zhou received the B.E. degree in mechatronics engineering from Tongji University, Shanghai, China, in 2014, and the M.Sc. degree in electrical and computer engineering from the Department of Electrical and Computer Engineering, Technical University of Munich, Munich, Germany, in 2017, where he is currently pursuing the Ph.D. degree in learning-based control and robotics with the Chair of Automatic Control Engineering, Department of Electrical and Computer Engineering. His research interests include optimal control, learning-based control, and applications to robotics.

Fangzhou Liu (Member, IEEE) received the M.Sc. degree in control theory and engineering from the Harbin Institute of Technology, Harbin, China, in 2014, and the Doktor-Ingenieur degree in electrical engineering from the Technical University of Munich, Munich, Germany, in 2019. He was a Lecturer and a Research Fellow with the Chair of Automatic Control Engineering, Technical University of Munich. He is currently a Full Professor with the School of Astronautics, Harbin Institute of Technology. His current research interests include networked control systems, modeling, analysis, and control on social networks, reinforcement learning and their applications. Prof. Liu has received the Dimitris N. Chorafas Prize and the Promotionspreis der Fakultät für Elektrotechnik und Informationstechnik.

Martin Buss (Fellow, IEEE) received the Diploma Engineering degree in electrical engineering from the Technische Universität Darmstadt, Darmstadt, Germany, in 1990, and the Doctor of Engineering degree in electrical engineering from The University of Tokyo, Tokyo, Japan, in 1994. In 1988, he was a Research Student for one year with the Science University of Tokyo, Tokyo. From 1994 to 1995, he was a Postdoctoral Researcher with the Department of Systems Engineering, Australian National University, Canberra, ACT, Australia. From 1995 to 2000, he was a Senior Research Assistant and a Lecturer with the Chair of Automatic Control Engineering, Department of Electrical Engineering and Information Technology, Technical University of Munich, Munich, Germany. From 2000 to 2003, he was a Full Professor, the Head of the Control Systems Group, and the Deputy Director of the Faculty IV, Electrical Engineering and Computer Science, Institute of Energy and Automation Technology, Technical University of Berlin, Berlin, Germany. Since 2003, he has been a Full Professor (Chair) with the Chair of Automatic Control Engineering, Faculty of Electrical Engineering and Information Technology, Technical University of Munich, where he has been with the Medical Faculty since 2008. His research interests include automatic control, mechatronics, multimodal human system interfaces, optimization, nonlinear, and hybrid discrete-continuous systems. Prof. Buss has been awarded the ERC Advanced Grant SHRINE.