A Neural-Reinforcement-Learning-based Guaranteed Cost Control for Perturbed Tracking Systems

Xiangnan Zhong, *Member, IEEE,* and Zhen Ni, *Senior Member, IEEE*

***Abstract*—AI-based learning control plays a critical role in the evolution of intelligent control, particularly for complex network systems. Traditional intelligent control methods assume the agent can learn from safe data in the tasks. However, many application scenarios exist perturbations caused by noise and/or malicious attack, which make the received data unreliable and may cause the failure of the learning process. In this paper, we focus on developing an intelligent guaranteed cost control method for nonlinear tracking systems subject to unknown matched and mismatched perturbations. By developing appropriate cost functions for the nominal plants, we transform the robust tracking control problem into a stabilization design for both kinds of perturbations. The explicit proofs are provided to show the equivalence of the transformation for these two situations respectively. Then, the neural-reinforcement-learning-based algorithm with guaranteed cost control is developed to learn the cost functions and optimal control laws adaptively. The designed method can also guarantee the boundedness of a given cost function. Three simulation studies are provided to demonstrate the effectiveness of the proposed method and also validate the theoretical analysis.**

***Impact Statement*—Over the past decades, AI-based learning control has received growing attention. In particular, the reinforcement learning method, which enables the agent to learn in an interactive environment, shows the potential to offer advantages over traditional control methods. It has been recently introduced into the tracking control problem, which is crucial in many applications, including robotics, aerospace, manufacturing, and process control. Despite the advances, reinforcement learning method may face significant challenges when used in the environment that exists perturbations or being attacked. This is because the perturbations can affect the observations or measurements that the agent relies on to learn its control action. The situation will become worse when the perturbations change over time. This limitation makes the learning control hard to be generalized well to changes. Therefore, in this paper, we design an intelligent control method through robust-optimal transformation for perturbed tracking systems to guarantee the control performance.**

***Index Terms*—Reinforcement learning, neural networks, tracking control, matched and mismatched perturbations, and guaranteed cost control.**

NOMENCLATURE

x	State vector
u	Control input
f, h, d	System functions
ε	Unknown perturbations
ξ	Perturbation term
ξ_f	Upper bound of $\|\|\xi\|\|$
r	Reference trajectory
ρ	Tracking error, $\rho = x - r$
f_c	Tracking function
J	Cost function
Λ	Utility function
$\mathcal{M}, \mathcal{R}, \mathcal{P}$	Positive definite symmetric matrices
$J_{\mathcal{T}}$	Modified cost function for the auxiliary nominal plant (8)
$\lambda_{\max}$	Maximal eigenvalue of a matrix
η	Design parameter, $\eta > \lambda_{\max}(\mathcal{R})$
h^+	Moore-Penrose pseudoinverse matrix of h
b	Augment state vector, $b = [\rho^T, r^T]^T$
$\mathcal{F}, \mathcal{G}, \mathcal{Z}_{\mathcal{A}}, \mathcal{Z}_{\mathcal{B}}$	System functions with respect to b
$\mathcal{A}_\xi$	Upper bound of $\|\|\xi\|\|$ with respect to b
Ξ	$\Xi(\cdot) = h^+(\cdot)d(\cdot)$
$\mathcal{V}_h$	Upper bound of $\|\|\Xi(\cdot)\xi(\cdot)\|\|$
$J_{\mathcal{S}}$	Modified cost function for the auxiliary nominal plant (24)
$\Lambda_{\mathcal{S}}$	Utility function with respect to b
$\mathcal{M}_I$	Positive definite symmetric matrix, $\mathcal{M}_I = diag\{\mathcal{M}, 0_{n\times n}\}$
σ	Design parameter, $0 < \sigma < \frac{1}{2}$
$J_{\mathcal{G}}$	Unified cost function for general perturbed tracking system (46)
$r_{\mathcal{S}}, p_{\mathcal{S}}$	Positive definite symmetric matrices, $\mathcal{R} = r_{\mathcal{S}} \cdot r_{\mathcal{S}}^T$, $\mathcal{P} = p_{\mathcal{S}} \cdot p_{\mathcal{S}}^T$
ι	Binary signal
ω_c	Ideal critic network weights for cost function $J_{\mathcal{T}}$ approximation
δ_c	Activation function of critic network for cost function $J_{\mathcal{T}}$ approximation
ϕ_c	Reconstruction error of critic network for cost function $J_{\mathcal{T}}$ approximation
$\hat{\omega}_c$	Estimated critic network weights for cost function $J_{\mathcal{T}}$ approximation
α_c	Learning rate of critic network for cost function $J_{\mathcal{T}}$ approximation
$\omega_{\mathcal{S}}$	Ideal critic network weights for cost function $J_{\mathcal{S}}$ approximation
$\delta_{\mathcal{S}}$	Activation function of critic network for cost function $J_{\mathcal{S}}$ approximation
$\phi_{\mathcal{S}}$	Reconstruction error of critic network for cost function $J_{\mathcal{S}}$ approximation
$\hat{\omega}_{\mathcal{S}}$	Estimated critic network weights for cost function $J_{\mathcal{S}}$ approximation
$\alpha_{\mathcal{S}}$	Learning rate of critic network for cost function $J_{\mathcal{S}}$ approximation

This work was supported in part by the National Science Foundation under Grant 2047010, 2047064, 1947419, and 2205205.

X. Zhong and Z. Ni are with the Department of Electrical Engineering and Computer Science, Florida Atlantic University, Boca Raton, FL 33431, USA. (email: xzhong@fau.edu, zhenni@fau.edu)

I. Introduction

Artificial intelligence (AI) and reinforcement learning methods have attracted great attention in the study of control systems over the past decades. By effectively approximating the solution of Hamilton-Jacobi-Bellman (HJB) equation, reinforcement learning method has been widely recognized as one of the core methodologies to achieve optimal learning-based control [1]–[5]. Extensive studies have been dedicated to advance the evolution of intelligent control in the society [6]–[8]. Recent research efforts, particularly the integration of neural networks and reinforcement learning, have further propelled the field and facilitated more efficient and adaptable control systems. Many research projects have been conducted with this neural-reinforcement-learning-based structure in terms of theoretical analysis [9]–[11], algorithm optimization [12]–[15], architecture development [16], [17] and real-world applications [18]–[20].

This method has also been applied to solve the trajectory tracking problems. For instance, an event-driven neuro-control method was designed in [21] to solve the tracking problem for continuous stirred tack reactor. In [22], an integral reinforcement learning method with neural networks implementation was applied on the optimal tracking control problem for constrained-input systems. By building an augment plant, authors in [23] developed a neuro-optimal tracking control method with value iteration for nonaffine discrete-time systems. In [24], an approximate optimal controller was established for unknown nonlinear tracking systems based on the recurrent neural network model and actor-critic learning algorithm. In these articles, the optimal tracking controller was designed based on the trustable received data. However, if the communication network is vulnerable to noise and/or malicious attack, the learning systems will suffer from uncertainties or perturbations, which make the received data unreliable. Therefore, the reinforcement learning methods can not be applied directly.

Recent studies on data-driven robust control showed the feasibility to stabilize the perturbed systems with corresponding optimal control results [25]–[29]. In [30], the event-based mechanism was considered in the robust-optimal transformation process to save the communication resource. A bounded robust controller was designed in [31] for finite-time-horizon nonlinear systems with uncertainties. In [32], the system with constrained input was studied and the robust adaptive control algorithm was designed with equivalent transformation analysis. A power system application was considered in [33] with actor-critic robust stabilization design by proper transformation. This idea has also been introduced into the robust trajectory tracking control in [34] for the system with general uncertainty. Particularly, the authors designed a self-learning robust tracking control method by integrating the partial derivatives of cost function into the auxiliary cost function for transformation design.

Besides, the guaranteed cost control becomes popular in robust control problems to not only stabilize the system for all admissible perturbations, but also guarantee the boundedness of the given cost function [30], [35]–[37]. In [38], the guaranteed cost tracking control method was developed for continuous-time systems with matched uncertainty. However, mismatched uncertainty or perturbation also exists and is sometimes more general in practice.

Motivated by the above observations and literature studies, this paper develops a neural-reinforcement-learning-based guaranteed cost control method for a class of continuous-time nonlinear tracking systems with matched and mismatched perturbations. The major contributions of this paper can be summarized as follows.

- By developing appropriate auxiliary cost functions and constructing augment systems, we transform the robust tracking control problem into an optimal stabilization design. Therefore, the learning-based robust control can be solved with the help of the transformed optimal control problem. The equivalence of the developed transformation is provided for matched and mismatched perturbation scenarios, respectively. In addition, we also establish the unified solution of learning-based control for both kinds of perturbations. This is important for perturbed tracking systems in general.
- An adaptive reinforcement-learning-based algorithm with guaranteed cost control is developed to asymptotically stabilize the closed-loop design. Comparing with [34], our developed cost function does not include its partial derivatives, which reduces the computational complexity, especially when dealing with large state space problems. Furthermore, we also provide the theoretical analysis for the boundedness guarantee of the given cost function.
- The online learning method is designed with neural networks implementation. Specifically, we implement the developed method with a critic-only structure, which plays a key role in computing the solution of the modified HJB equation efficiently. The learned control law is subsequently applied on the original perturbed system for robust trajectory tracking control. The stability of the developed online learning process is also provided to ensure the tracking performance.

Note that our developed neural-reinforcement-learning-based guaranteed cost control method is an online learning process. Therefore, the neural networks are established and trained based on the real-time data for cost function estimation and control law calculation. In contrast to the traditional optimization-based control methods, which are known for their stability but rely on accurate models and may not handle complex or high-dimensional systems effectively, our developed approach is data-driven. It does not require the explicit system models, which is more flexible and adaptable.

The rest of the paper is organized as follows. In Section II, the tracking control problem with unknown matched and mismatched perturbations are formulated. In Section III, we convert the robust tracking control problem into the corresponding stabilization design with appropriate auxiliary cost function for both kinds of perturbations. Furthermore, the guaranteed cost control algorithm is developed based on the reinforcement learning techniques to stabilize the transformed system in two scenarios respectively. The online learning

method with neural networks implementation is developed in Section IV and the stability analysis of the learning process is also provided. In Section V, three simulation studies are given to demonstrate the effectiveness of the proposed method, including a very unstable problem, i.e., triple-link inverted pendulum balancing system (eight state variables). Finally, Section VI concludes this paper.

II. Tracking Control under Perturbations

Consider a class of nonlinear systems in the nominal condition as follows

$$\dot{x}(t) = f(x(t)) + h(x(t))u(t) \tag{1}$$

where $x(t) \in \mathbb{R}^n$ is the state vector, $u(t) \in \mathbb{R}^m$ is the control input, $f(\cdot) \in \mathbb{R}^n$ and $h(\cdot) \in \mathbb{R}^{n\times m}$ are the drift and input dynamics, respectively, with $f(0) = 0$.

The reference tracking trajectory $r(t) \in \mathbb{R}^n$ is assumed bounded and given as

$$\dot{r}(t) = f_c(r(t)). \tag{2}$$

Define the tracking error as $\rho(t) = x(t) - r(t)$. Based on (1) and (2), the dynamics of $\rho(t)$ is

$$\dot{\rho}(t) = f(x(t)) - f_c(r(t)) + h(x(t))u(t). \tag{3}$$

In this paper, we consider that the data communication network is vulnerable. This means the exchanged data may alter. Most attacks and faults can be modeled by additive inputs on system actuator and/or sensor measurements. Therefore, the general description of the altered system dynamics can be provided as follows

$$\dot{x}(t) = f(x(t)) + h(x(t))u(t) + \varepsilon(x(t)) \tag{4}$$

where $\varepsilon(x(t)) = d(x(t))\xi(x(t))$ is the unknown perturbation caused by attacks or faults. In this paper, we consider two kinds of system perturbations, i.e., matched perturbation ($d(x(t)) = h(x(t))$) and mismatched perturbation ($d(x(t)) \neq h(x(t))$). The perturbation term $\xi(x(t)) \in \mathbb{R}^p$ is upper bounded by a known function $\|\xi(x(t))\| \leq \xi_f(x(t))$ with $\xi_f(0) = 0$. This assumption is reasonable and aligns with the physical limits inherent in the systems.

Therefore, the dynamics of tracking error becomes

$$\dot{\rho}(t) = f(x(t)) - f_c(r(t)) + h(x(t))u(t) + \varepsilon(x(t)). \tag{5}$$

The cost function is given as

$$J(\rho(t)) = \int_t^\infty \Lambda(\rho(\tau), u(\tau))d\tau \tag{6}$$

where $\Lambda(\rho(t), u(t)) = \rho^T(t)\mathcal{M}\rho(t) + u^T(t)\mathcal{R}u(t)$ is the utility function with $\Lambda(0,0) = 0$. Also, we have $\mathcal{M} > 0$ and $\mathcal{R} > 0$ are the symmetric matrices.

Our goal is to develop a data-driven robust controller $u(t)$, such that the closed-loop system (5) with all the admissible perturbations is guaranteed stable, and the cost function (6) is upper bounded by a finite function. Then, the designed controller $u(t)$ is the guaranteed cost control law and the upper bound function is the guaranteed cost function. A neural-reinforcement-learning-based guaranteed cost control method is developed in this paper to achieve this goal. Note that we eliminate the time index t in the following statement for presentation simplification.

III. Adaptive Reinforcement-Learning-based Algorithm Design with Guaranteed Cost Control

In this section, we start with the development of a data-driven guaranteed cost control algorithm for tracking system with matched perturbation. Then, the case with mismatched perturbation is explored and discussed. The theoretical foundation of the developed algorithm is studied for both cases respectively. Furthermore, we also establish the unified solution of data-driven learning-based control for the general perturbed tracking systems.

A. Design with Matched Perturbation

Consider the fact $x = \rho + r$ and we reconstruct the system function with tracking dynamics as

$$\begin{bmatrix}\dot{\rho}\\ \dot{r}\end{bmatrix} = \begin{bmatrix}f(\rho+r) - f_c(r)\\ f_c(r)\end{bmatrix} + \begin{bmatrix}h(\rho+r)\\ 0_{n\times m}\end{bmatrix}u + \begin{bmatrix}\varepsilon(\rho+r)\\ 0_{n\times m}\end{bmatrix} \tag{7}$$

where $\varepsilon(\rho + r) = d(\rho + r)\xi(\rho + r) = h(\rho + r)\xi(\rho + r)$ is the matched perturbation and $\|\xi(\rho + r)\| = \|\xi(x)\| \leq \xi_f(x) \triangleq \xi_f(\rho, r)$. Here, we consider the augment state $\rho + r$ instead of x to show the tracking dynamics. This is an alternative representation of dynamics $[\dot{\rho}, \dot{r}, \dot{x}]$ with $\dot{\rho} = f(x) - f_c(r) + h(x)u + \varepsilon(x)$, $\dot{r} = f_c(r)$, and $\dot{x} = \dot{\rho} + \dot{r}$. The design in (7) simplifies the system function by reducing the state dimensions, which will further facilitate the analysis and reduce computation cost.

The nominal part of system (7) is given as

$$\begin{bmatrix}\dot{\rho}\\ \dot{r}\end{bmatrix} = \begin{bmatrix}f(\rho+r) - f_c(r)\\ f_c(r)\end{bmatrix} + \begin{bmatrix}h(\rho+r)\\ 0_{n\times m}\end{bmatrix}u \tag{8}$$

which is assumed controllable.

It is desired to find the feedback control law u to minimize the modified cost function

$$J_{\mathcal{T}}(\rho) = \int_t^\infty \left\{\eta\xi_f^2(\rho(\tau), r(\tau)) + \Lambda(\rho(\tau), u(\tau))\right\}d\tau \tag{9}$$

where η is the design parameter and $J_{\mathcal{T}}(0) = 0$. Hence, the robust tracking control problem has been converted into an optimal stabilization design with the system dynamics provided in (8) and the cost function modified in (9).

Define the Hamiltonian of the transformed optimal control problem as

$$\begin{aligned}\mathcal{H}(\rho, r, u, \nabla J_{\mathcal{T}}) &= \eta\xi_f^2(\rho, r) + \Lambda(\rho, u)\\ &+ \left(\nabla J_{\mathcal{T}}(\rho)\right)^T\left(f(\rho+r) - f_c(r) + h(\rho+r)u\right)\end{aligned} \tag{10}$$

where $\nabla J_{\mathcal{T}}(\rho) = \partial J_{\mathcal{T}}(\rho)/\partial\rho$ is the partial derivative of $J_{\mathcal{T}}(\rho)$ with respect to ρ.

Based on Bellman's optimality equation, the optimal cost function is the minimal result of (9), which is

$$J_{\mathcal{T}}^*(\rho) = \min_u \int_t^\infty \left\{\eta\xi_f^2(\rho(\tau), r(\tau)) + \Lambda(\rho(\tau), u(\tau))\right\}d\tau. \tag{11}$$

Therefore, the optimal control law can be derived as

$$\begin{aligned}\frac{\partial\mathcal{H}(\rho, r, u, \nabla J_{\mathcal{T}}^*)}{\partial u} = 0 \Rightarrow\\ u^* = -\frac{1}{2}\mathcal{R}^{-1}h^T(\rho + r)\nabla J_{\mathcal{T}}^*(\rho).\end{aligned} \tag{12}$$

Substituting (12) into (10), we have the HJB equation as

$$\mathcal{H}(\rho, r, u^*, \nabla J_{\mathcal{T}}^*) = 0. \tag{13}$$

The following theorem provides the equivalence of this robust-optimal transformation, which shows the designed control law (12) is the result of the orignal robust tracking problem with matched perturbation.

Theorem 1: If $J_{\mathcal{T}}^*(\rho)$ is the solution of HJB equation (13), then the designed optimal control law (12) of system (8) guarantees the closed-loop asymptotic stability of system (7) with matched perturbation.

Proof: Apply the optimal control law u^* to the perturbed system (7). Define $J_{\mathcal{T}}^*(\rho)$ as the Lyapunov function. Then, the first derivative of $J_{\mathcal{T}}^*(\rho)$ is given as

$$\begin{aligned}\dot{J}_{\mathcal{T}}^*(\rho) = &\Big(\nabla J_{\mathcal{T}}^*(\rho)\Big)^T \Big(f(\rho+r) - f_c(r) + h(\rho+r)u^*\Big) \\ &+ \Big(\nabla J_{\mathcal{T}}^*(\rho)\Big)^T \varepsilon(\rho+r). \end{aligned}\tag{14}$$

Considering the HJB equation (13), it follows

$$\begin{aligned}\dot{J}_{\mathcal{T}}^*(\rho) &= -\eta\xi_f^2(\rho,r) - \Lambda(\rho,u^*) + \Big(\nabla J_{\mathcal{T}}^*(\rho)\Big)^T \varepsilon(\rho+r) \\ &= -\eta\xi_f^2(\rho,r) - \rho^T\mathcal{M}\rho - u^{*T}\mathcal{R}u^* \\ &\quad + \Big(\nabla J_{\mathcal{T}}^*(\rho)\Big)^T h(\rho+r)\xi(\rho+r). \end{aligned}\tag{15}$$

Based on (12), we have $(\nabla J_{\mathcal{T}}^*(\rho))^T h(\rho+r) = -2u^{*T}\mathcal{R}$. By adding and subtracting the term $\xi^T(\rho+r)\mathcal{R}\xi(\rho+r)$ on the equation (15), it can be further rewritten as

$$\begin{aligned}\dot{J}_{\mathcal{T}}^*(\rho) &= -\eta\xi_f^2(\rho,r) - \rho^T\mathcal{M}\rho + \xi^T(\rho+r)\mathcal{R}\xi(\rho+r) \\ &\quad - \Big(u^* + \xi(\rho+r)\Big)^T \mathcal{R}\Big(u^* + \xi(\rho+r)\Big) \\ &\le -\rho^T\mathcal{M}\rho - \Big(\eta - \lambda_{\max}(\mathcal{R})\Big)\xi_f^2(\rho,r) \end{aligned}\tag{16}$$

where $\lambda_{\max}(\cdot)$ is the maximal eigenvalue of a matrix. If the design parameter $\eta > \lambda_{\max}(\mathcal{R})$, we have

$$\dot{J}_{\mathcal{T}}^*(\rho) \le -\rho^T\mathcal{M}\rho < 0, \forall \rho \ne 0. \tag{17}$$

Hence, the designed feedback control law (12) can asymptotically stabilize the system (7) with matched perturbation, which completes the proof. ∎

Therefore, the designed control law (12) can guarantee the stability of system (7), i.e., $\rho \to 0$, when $t \to \infty$. Considering the fact $\rho = x - r$, it follows $x \to r$ when $t \to \infty$. This means the designed control law (12) can guarantee the tracking performance of the orignal tracking system (4) with matched perturbation.

Comparing the modified cost function (9) with (6), we can easily obtain

$$J(\rho) \le J_{\mathcal{T}}(\rho) \triangleq \mathcal{B}_{\mathcal{J}}(\rho) \tag{18}$$

which ensures the boundedness of (6). This demonstrates the guaranteed cost control for the tracking system (4) with matched perturbation. In other words, the designed control law (12) is the guaranteed cost control law and the upper bound $\mathcal{B}_{\mathcal{J}}(\rho)$ is the guaranteed cost function. Furthermore, the optimal bound can be derived by $\mathcal{B}_{\mathcal{J}}^*(\rho) = \min_u J_{\mathcal{T}}(\rho)$ and it normally satisfies the condition

$$\min_u \mathcal{H}(\rho, r, u^*, \nabla\mathcal{B}_{\mathcal{J}}^*(\rho)) = 0 \tag{19}$$

where $\nabla\mathcal{B}_{\mathcal{J}}^*(\rho) = \partial\mathcal{B}_{\mathcal{J}}^*(\rho)/\partial\rho$.

B. Design with Mismatched Perturbation

In this section, we develop the data-driven guaranteed cost control method for tracking system with mismatched perturbation, i.e., $d(x) \ne h(x)$. Decompose the perturbation term into two parts as the matched and mismatched elements:

$$\begin{aligned}d(x)\xi(x) &= h(x)h^+(x)d(x)\xi(x) \\ &\quad + \Big(I_n - h(x)h^+(x)\Big)d(x)\xi(x) \end{aligned}\tag{20}$$

where $h^+(x)$ is the Moore-Penrose pseudoinverse matrix of $h(x)$. Hence, the tracking error dynamics can be rewritten as

$$\begin{aligned}\dot{\rho} &= f(x) - f_c(r) + h(x)u + h(x)h^+(x)d(x)\xi(x) \\ &\quad + \Big(I_n - h(x)h^+(x)\Big)d(x)\xi(x). \end{aligned}\tag{21}$$

Define an augment state $b = [\rho^T, r^T]^T \in \mathbb{R}^{2n}$. Since $\rho = x - r$, the dynamics of this augment state is given as

$$\dot{b} = \mathcal{F}(b) + \mathcal{G}(b)u + \mathcal{Z}_{\mathcal{A}}(b)\xi(b) + \mathcal{Z}_{\mathcal{B}}(b)\xi(b) \tag{22}$$

where

$$\begin{aligned}\mathcal{F}(b) &= \begin{bmatrix} f(\rho+r) - f_c(r) \\ f_c(r) \end{bmatrix}, \quad \mathcal{G}(b) = \begin{bmatrix} h(\rho+r) \\ 0_{n\times m} \end{bmatrix}, \\ \mathcal{Z}_{\mathcal{A}}(b) &= \begin{bmatrix} h(\rho+r)h^+(\rho+r)d(\rho+r) \\ 0_{n\times p} \end{bmatrix}, \\ \mathcal{Z}_{\mathcal{B}}(b) &= \begin{bmatrix} \Big(I_n - h(\rho+r)h^+(\rho+r)\Big)d(\rho+r) \\ 0_{n\times p} \end{bmatrix}. \end{aligned}\tag{23}$$

Here, $\mathcal{Z}_{\mathcal{A}}(b)$ and $\mathcal{Z}_{\mathcal{B}}(b)$ represent the matched and mismatched elements of perturbation, respectively.

For the new dynamics (22), the perturbation term $\xi(b)$ is also bounded as $\|\xi(b)\| = \|\xi(x)\| \le \xi_f(x) \triangleq \mathcal{A}_\xi(b)$. Therefore, design the following nominal plant with an auxiliary control variable $v \in \mathbb{R}^p$ as

$$\dot{b} = \mathcal{F}(b) + \mathcal{G}(b)u + \mathcal{Z}_{\mathcal{B}}(b)v. \tag{24}$$

Note that v is not used in the robust control process, but it helps to obtain the feedback control law u in the optimal learning process.

Assume the system (24) is controllable. The objective is to find the optimal control law $[u^{*T}, v^{*T}]^T$ that minimize the following modified cost function

$$\begin{aligned}J_{\mathcal{S}}(b) = \int_t^\infty \Big\{&\mathcal{V}_h^2(b(\tau)) + \sigma\mathcal{A}_\xi^2(b(\tau)) \\ &+ \Lambda_{\mathcal{S}}(b(\tau), u(\tau), v(\tau))\Big\}d\tau \end{aligned}\tag{25}$$

where $0 < \sigma < 1/2$ is the design parameter, $\mathcal{V}_h(b)$ is an upper bound of $\|\Xi(b)\xi(b)\| \le \mathcal{V}_h(b)$ and $\Xi(b) = h^+(\rho+r)d(\rho+r)$. The function $\Lambda_{\mathcal{S}}(b,u,v) = b^T\mathcal{M}_I b + u^T\mathcal{R}u + \sigma v^T\mathcal{P}v$ with $\Lambda_{\mathcal{S}}(0,0,0) = 0$, where $\mathcal{M}_I = diag\{\mathcal{M}, 0_{n\times n}\}$ and $\mathcal{P} > 0$ is a symmetric matrix. In this paper, we consider $\mathcal{R}$ and $\mathcal{P}$ are the

identity matrices with appropriate dimensions in the controller design for tracking system with mismatched perturbation. Hence, it follows $\Lambda_{\mathcal{S}}(b,u,v) = b^T\mathcal{M}_I b + u^T u + \sigma v^T v$. The modified cost function (25) is expected to solve the robust-optimal transformation where $\Lambda_{\mathcal{S}}(b,u,v)$ is designed for the optimal system (24) and the term $\mathcal{V}_h^2(b) + \sigma\mathcal{A}_\xi^2(b)$ reflects the perturbation.

We can further derive the infinitesimal version of (25) as

$$\mathcal{V}_h^2(b) + \sigma\mathcal{A}_\xi^2(b) + \Lambda_{\mathcal{S}}(b,u,v) + \Big(\nabla J_{\mathcal{S}}(b)\Big)^T \cdot \Big(\mathcal{F}(b) + \mathcal{G}(b)u + \mathcal{Z}_{\mathcal{B}}(b)v\Big) = 0 \tag{26}$$

with $J_{\mathcal{S}}(0) = 0$. Thus, the Hamiltonian can be defined as

$$\mathcal{H}_{\mathcal{S}}(b,u,v,\nabla J_{\mathcal{S}}) = \mathcal{V}_h^2(b) + \sigma\mathcal{A}_\xi^2(b) + \Lambda_{\mathcal{S}}(b,u,v) + \Big(\nabla J_{\mathcal{S}}(b)\Big)^T\Big(\mathcal{F}(b) + \mathcal{G}(b)u + \mathcal{Z}_{\mathcal{B}}(b)v\Big) \tag{27}$$

where $\nabla J_{\mathcal{S}}(b) = \partial J_{\mathcal{S}}(b)/\partial b$.

The optimal cost function is given as

$$J_{\mathcal{S}}^*(b) = \min_{u,v}\int_t^\infty \Big\{\mathcal{V}_h^2(b(\tau)) + \sigma\mathcal{A}_\xi^2(b(\tau)) + \Lambda_{\mathcal{S}}(b(\tau),u(\tau),v(\tau))\Big\}d\tau \tag{28}$$

which satisfies the HJB equation

$$\mathcal{H}_{\mathcal{S}}(b,u^*,v^*,\nabla J_{\mathcal{S}}^*) = 0. \tag{29}$$

Then, the optimal control law is provided as

$$\frac{\partial\mathcal{H}_{\mathcal{S}}(b,u,v,\nabla J_{\mathcal{S}}^*)}{\partial u} = 0 \Rightarrow u^* = -\frac{1}{2}\mathcal{G}^T(b)\nabla J_{\mathcal{S}}^*(b), \tag{30}$$

$$\frac{\partial\mathcal{H}_{\mathcal{S}}(b,u,v,\nabla J_{\mathcal{S}}^*)}{\partial v} = 0 \Rightarrow v^* = -\frac{1}{2\sigma}\mathcal{Z}_{\mathcal{B}}^T(b)\nabla J_{\mathcal{S}}^*(b). \tag{31}$$

The following theorem proves that the designed optimal control law (30) can stabilize the perturbed system (22) and, therefore, can ensure the perturbed tracking system (4) follows the trajectory of reference.

Theorem 2: Consider the system (24) with modified cost function (25). Then, the designed feedback control law (30) guarantees the closed-loop asymptotic stability of system (22) with mismatched perturbation.

Proof: Consider $J_{\mathcal{S}}^*(b)$ as a Lyapunov function. Applying $J_{\mathcal{S}}^*(b)$ on the perturbed system (22), we have

$$\begin{aligned}\dot{J}_{\mathcal{S}}^*(b) =& \Big(\nabla J_{\mathcal{S}}^*(b)\Big)^T\Big(\mathcal{F}(b) + \mathcal{G}(b)u^* + \mathcal{Z}_{\mathcal{A}}(b)\xi(b) + \mathcal{Z}_{\mathcal{B}}(b)\xi(b)\Big)\\ =& \Big(\nabla J_{\mathcal{S}}^*(b)\Big)^T\Big(\mathcal{F}(b) + \mathcal{G}(b)u^* + \mathcal{Z}_{\mathcal{B}}(b)v^*\Big) + \Big(\nabla J_{\mathcal{S}}^*(b)\Big)^T\Big(\mathcal{Z}_{\mathcal{A}}(b)\xi(b) + \mathcal{Z}_{\mathcal{B}}(b)\big(\xi(b) - v^*\big)\Big).\end{aligned} \tag{32}$$

Considering the HJB equation (29), it follows

$$\dot{J}_{\mathcal{S}}^*(b) = -\mathcal{V}_h^2(b) - \sigma\mathcal{A}_\xi^2(b) - \Lambda_{\mathcal{S}}(b,u^*,v^*) + \Big(\nabla J_{\mathcal{S}}^*(b)\Big)^T \cdot \mathcal{Z}_{\mathcal{A}}(b)\xi(b) + \Big(\nabla J_{\mathcal{S}}^*(b)\Big)^T\mathcal{Z}_{\mathcal{B}}(b)\big(\xi(b) - v^*\big). \tag{33}$$

Based on (30) and (31), we have

$$\Big(\nabla J_{\mathcal{S}}^*(b)\Big)^T\mathcal{G}(b) = -2u^{*T}, \tag{34}$$

$$\Big(\nabla J_{\mathcal{S}}^*(b)\Big)^T\mathcal{Z}_{\mathcal{B}}(b) = -2\sigma v^{*T}. \tag{35}$$

Since

$$\Big(\nabla J_{\mathcal{S}}^*(b)\Big)^T\mathcal{Z}_{\mathcal{A}}(b) = \Big(\nabla J_{\mathcal{S}}^*(b)\Big)^T\mathcal{G}(b)\Xi(b) \tag{36}$$

we can rewrite (33) as

$$\dot{J}_{\mathcal{S}}^*(b) = -\mathcal{V}_h^2(b) - \sigma\mathcal{A}_\xi^2(b) - b^T\mathcal{M}_I b - u^{*T}u^* - \sigma v^{*T}v^* - 2u^{*T}\Xi(b)\xi(b) - 2\sigma v^{*T}\big(\xi(b) - v^*\big). \tag{37}$$

The mathematical deduction brings

$$-u^{*T}u^* - 2u^{*T}\Xi(b)\xi(b) = -\|u^* + \Xi(b)\xi(b)\|^2 + \|\Xi(b)\xi(b)\|^2 \tag{38}$$

and the inequality

$$-2\sigma v^{*T}\xi(b) \le \sigma\Big(\|v^*\|^2 + \|\xi(b)\|^2\Big). \tag{39}$$

Substituting (38) and (39) into (37), we have

$$\begin{aligned}\dot{J}_{\mathcal{S}}^*(b) \le& -\mathcal{V}_h^2(b) - \sigma\mathcal{A}_\xi^2(b) - \sigma v^{*T}v^* - b^T\mathcal{M}_I b - \|u^* + \Xi(b)\xi(b)\|^2 + \|\Xi(b)\xi(b)\|^2 + \sigma\|v^*\|^2 + \sigma\|\xi(b)\|^2 + 2\sigma v^{*T}v^*\\ \le& -\Big(\mathcal{V}_h^2(b) - \|\Xi(b)\xi(b)\|^2\Big) - b^T\mathcal{M}_I b - \|u^* + \Xi(b)\xi(b)\|^2 + 2\sigma v^{*T}v^*.\end{aligned} \tag{40}$$

It follows,

$$\begin{aligned}\dot{J}_{\mathcal{S}}^*(b) \le& -b^T\mathcal{M}_I b + 2\sigma v^{*T}v^*\\ \le& -(1-2\sigma)b^T\mathcal{M}_I b + 2\sigma\Big(v^{*T}v^* - b^T\mathcal{M}_I b\Big).\end{aligned} \tag{41}$$

Since $0 < \sigma < 1/2$, if $v^{*T}v^* \le b^T\mathcal{M}_I b$, we have $\dot{J}_{\mathcal{S}}^*(b) < 0$, $\forall b \ne 0$. Hence, the designed optimal control law u^* can asymptotically stabilize the perturbed system (22), which means the tracking error ρ is asymptotically stable. This completes the proof. ■

Theorem 2 shows the stability of the tracking error ρ with the designed control law (30). Since $\rho = x - r$, the robust control of tracking system (4) with mismatched perturbation is achieved. Furthermore, we can observe that the modified cost function (25) is an upper bound of (6), i.e.,

$$J(\rho) \le J_{\mathcal{S}}(b) \triangleq \mathcal{B}_{\mathcal{S}}(b). \tag{42}$$

This means the designed method is the guaranteed cost control for tracking system (4) with mismatched perturbation, where u^* in (30) is the guaranteed cost control law and the upper bound $\mathcal{B}_{\mathcal{S}}(b)$ is the guaranteed cost function. Besides, the optimal bound can be obtained as $\mathcal{B}_{\mathcal{S}}^*(b) = \min_{u,v} J_{\mathcal{S}}(b)$. Setting $\nabla\mathcal{B}_{\mathcal{S}}^*(b) = \partial\mathcal{B}_{\mathcal{S}}^*(b)/\partial b(t)$, we have $B_{\mathcal{S}}^*(b)$ satisfies the condition $\min_{u,v}\mathcal{H}(b,u^*,v^*,\nabla\mathcal{B}_{\mathcal{S}}^*(b)) = 0$.

In this paper, we assume $\mathcal{R}$ and $\mathcal{P}$ are the identity matrices in the control design for tracking system with mismatched perturbation. In fact, we can choose proper parameters for the modified cost function when $\mathcal{R}$ and $\mathcal{P}$ are only the positive

and symmetric matrices. Specifically, by setting $\mathcal{R} = r_{\mathcal{S}} \cdot r_{\mathcal{S}}^T$ and $\mathcal{P} = p_{\mathcal{S}} \cdot p_{\mathcal{S}}^T$, we have the cost function revised as

$$J_{\mathcal{S}_0}^*(b) = \min_{u_0, v_0} \int_t^\infty \Big\{ \|r_{\mathcal{S}}\|^2 \mathcal{V}_h^2(b(\tau)) + \sigma \|p_{\mathcal{S}}\|^2 \mathcal{A}_\xi^2(b(\tau)) + \Lambda_{\mathcal{S}}(b(\tau), u(\tau), v(\tau)) \Big\} d\tau. \quad (43)$$

The corresponding feedback control law becomes

$$u_0^* = -\frac{1}{2} \mathcal{R}^{-1} \mathcal{G}^T(b) \nabla J_{\mathcal{S}_0}^*(b), \quad (44)$$

$$v_0^* = -\frac{1}{2\sigma} \mathcal{P}^{-1} \mathcal{Z}_{\mathcal{B}}^T(b) \nabla J_{\mathcal{S}_0}^*(b). \quad (45)$$

where $\nabla J_{\mathcal{S}_0}^*(b) = \partial J_{\mathcal{S}_0}^*(b)/\partial b$. It can be proved that u_0^* is the guaranteed cost control law which can asymptotically stabilize the perturbed system (22) and $J_{\mathcal{S}_0}(b)$ is the guaranteed cost function.

It is worth noting that if both matched and mismatched perturbations exist in the system dynamics, we have

$$\dot{\rho} = f(x) - f_c(r) + h(x)u + h(x)\xi_1(x) + \mathcal{Z}(x)\xi_2(x) \quad (46)$$

where $\xi_1(x)$ and $\xi_2(x)$ are the perturbation terms, and $\mathcal{Z}(x) \neq h(x)$. Therefore, $h(x)\xi_1(x)$ is the matched perturbation and $\mathcal{Z}(x)\xi_2(x)$ is the mismatched one.

We can reformulate (46) as

$$\dot{\rho} = f(x) - f_c(r) + h(x)u + \underbrace{\begin{bmatrix} h(x) & \mathcal{Z}(x) \end{bmatrix}}_{d(x)} \underbrace{\begin{bmatrix} \xi_1(x) \\ \xi_2(x) \end{bmatrix}}_{\xi(x)}. \quad (47)$$

Since $d(x)$ does not consist with the input dynamics $h(x)$, we have reconstructed the function as the system with mismatched perturbations. Then, the cost function (28) and the control law (30) can be used to solve this problem.

Besides, consider the system dynamics with matched perturbation (7) and the corresponding nominal plant (8). If we define the augment state as $b = [\rho^T, r^T]^T$, the nominal plant (8) can be rewritten as $\dot{b} = \mathcal{F}(b) + \mathcal{G}(b)u$ with $\mathcal{F}(b)$ and $\mathcal{G}(b)$ are provided in (23), alongside the cost function (11) and control laws (12) revised as $J_{\mathcal{T}}^*(b) = \min_u \int_t^\infty \{\eta \mathcal{A}_\xi^2(b(\tau)) + b^T \mathcal{M}_1 b + u^T \mathcal{R} u\} d\tau$ and $u^* = -\frac{1}{2}\mathcal{R}^{-1}\mathcal{G}^T(b)\nabla J_{\mathcal{T}}^*(b)$. Comparing the design with the results of system with mismatched perturbation (43) and (44), we can unify the control law for the general perturbed tracking systems as

$$u^* = -\frac{1}{2} \mathcal{R}^{-1} \mathcal{G}^T(b) \nabla J_{\mathcal{G}}^*(b) \quad (48)$$

where $\nabla J_{\mathcal{G}}^*(b) = \partial J_{\mathcal{G}}^*(b)/\partial b$ and $J_{\mathcal{G}}^*(b)$ is the unified cost function provided as

$$J_{\mathcal{G}}^*(b) = \min_{\mathcal{U}} \int_t^\infty \Big\{ \iota \|r_{\mathcal{S}}\|^2 \mathcal{V}_h^2(b(\tau)) + \Big(\iota\sigma \|p_{\mathcal{S}}\|^2 + (1-\iota)\eta \Big) \mathcal{A}_\xi^2(b(\tau)) + \Lambda_{\mathcal{G}}(b(\tau), \mathcal{U}(\tau)) \Big\} d(\tau) \quad (49)$$

in which $\mathcal{U} = [u^T, v^T]^T$ for mismatched perturbation and $\mathcal{U} = u$ for matched perturbation, $\Lambda_{\mathcal{G}}(b, \mathcal{U}) = b^T \mathcal{M}_1 b + u^T \mathcal{R} u + \iota\sigma v^T \mathcal{P} v$, and ι is a binary signal, such that $\iota = 1$ for mismatched perturbation and $\iota = 0$ for matched perturbation.

IV. Proposed Online Learning Method with Stability Analysis

In this section, the online learning method is designed based on neural networks implementation to estimate the solution of the HJB equation. This paper establishes the critic-only structure for both types of perturbed tracking systems, such that the computation cost can be reduced. The stability analysis of the designed learning process is also provided to guarantee that the estimation errors of learned neural network weights are uniformly ultimately bounded (UUB).

A. Adaptive Neural Network Architecture

Consider the controller design with matched perturbation and apply the neural networks to reconstruct the cost function $J_{\mathcal{T}}^*(\rho)$ as

$$J_{\mathcal{T}}^*(\rho) = \omega_c^T \delta_c(\rho) + \phi_c(\rho) \quad (50)$$

where ω_c are the ideal weights for function approximation, $\delta_c(\rho)$ is the activation function, and $\phi_c(\rho)$ is the reconstruction error.

Since the ideal weights ω_c are unknown, we build a critic network with the estimated weights $\hat{\omega}_c$ to approximate the cost function as

$$J_{\mathcal{T}}(\rho) = \hat{\omega}_c^T \delta_c(\rho). \quad (51)$$

The partial derivative of $J_{\mathcal{T}}(\rho)$ is given as $\nabla J_{\mathcal{T}}(\rho) = (\nabla \delta_c(\rho))^T \hat{\omega}_c$. Considering (12), we have

$$u = -\frac{1}{2} \mathcal{R}^{-1} h^T(\rho + r) (\nabla \delta_c(\rho))^T \hat{\omega}_c. \quad (52)$$

This is an estimated version of the feedback control law $u^* = -\frac{1}{2}\mathcal{R}^{-1}h^T(\rho+r)\Big((\nabla\delta_c(\rho))^T\omega_c + \nabla\phi_c(\rho)\Big)$. Now our task becomes to adaptively learn the suitable critic network weights $\hat{\omega}_c$.

Considering (10), the approximate Hamiltonian with the established critic network can be provided as

$$\mathcal{H}(\rho, r, u, \nabla\delta_c^T\hat{\omega}_c) = \eta\xi_f^2(\rho, r) + \rho^T \mathcal{M}\rho - \frac{1}{4}\hat{\omega}_c^T \nabla\delta_c(\rho) h(\rho+r)\mathcal{R}^{-1}h^T(\rho+r) \cdot (\nabla\delta_c(\rho))^T\hat{\omega}_c + \hat{\omega}_c^T\nabla\delta_c(\rho)\Big(f(\rho+r) - f_c(r)\Big). \quad (53)$$

Therefore, we can define the objective error function as $E_c = 0.5e_c^2$ with $e_c = \mathcal{H}(\rho, r, u, \nabla\delta_c^T\hat{\omega}_c)$. Then, the weight adjustment rule of critic network can be derived as

$$\dot{\hat{\omega}}_c = -\alpha_c \frac{1}{(1+\kappa^T\kappa)^2}\left(\frac{\partial E_c}{\partial \hat{\omega}_c}\right) = -\alpha_c \frac{\kappa}{(1+\kappa^T\kappa)^2}\Big(\eta\xi_f^2(\rho, r) + \Lambda(\rho, u) + \kappa^T\hat{\omega}_c\Big) \quad (54)$$

where $\alpha_c > 0$ is the learning rate and $\kappa = \nabla\delta_c(\rho)\big(f(\rho + r) - f_c(r) + h(\rho+r)u\big)$. The term $(1+\kappa^T\kappa)^2$ is for optional normalization. Therefore, the learning rule (54) adaptively updates the weights $\hat{\omega}_c$ and then constructs the feedback control law based on (52). The algorithm of this critic-only learning-based robust controller design for tracking system with matched perturbation is provided in Algorithm 1.

Algorithm 1 Controller Design with Matched Perturbation

Select the learning time $T_{c1} > 0$ and the execution time $T_{c2} > 0$. Set parameters η, $\xi_f(\rho)$, $\mathcal{M}$, $\mathcal{R}$, α_c, and a threshold ν_c.
Begin the learning process
Initialize
for $t = 0 \to T_{c1}$ **do**
  Compute u using (52);
  Take action u and collect ρ from (8);
  Update $J_{\mathcal{T}}(\rho)$ through (51);
  Update $\hat{\omega}_c$ from (54);
  if $\|\Delta\hat{\omega}_c\| < \nu_c$ **then**
    Stop;
  end if
end for
return $\hat{\omega}_c$ and fix it.
Begin the robust control process
for $t = 0 \to T_{c2}$ **do**
  Compute u using (52);
  Take action u and collect x from (4);
end for

Now, we consider the case with mismatched perturbation. Denote the ideal weights as $\omega_{\mathcal{S}}$, the activation function as $\delta_{\mathcal{S}}(b)$ and the reconstruction error as $\phi_{\mathcal{S}}(b)$. Hence, the cost function $J^*_{\mathcal{S}}(b)$ can be provided with a neural network structure as

$$J^*_{\mathcal{S}}(b) = \omega^T_{\mathcal{S}}\delta_{\mathcal{S}}(b) + \phi_{\mathcal{S}}(b). \tag{55}$$

Based on (30), (31) and (55), the optimal control law can be provided as

$$\begin{bmatrix} u^* \\ v^* \end{bmatrix} = \begin{bmatrix} -\frac{1}{2}\mathcal{G}^T(b)\Big(\big(\nabla\delta_{\mathcal{S}}(b)\big)^T\omega_{\mathcal{S}} + \nabla\phi_{\mathcal{S}}(b)\Big) \\ -\frac{1}{2}\mathcal{Z}^T_{\mathcal{B}}(b)\Big(\big(\nabla\delta_{\mathcal{S}}(b)\big)^T\omega_{\mathcal{S}} + \nabla\phi_{\mathcal{S}}(b)\Big) \end{bmatrix}. \tag{56}$$

Since the ideal weights $\omega_{\mathcal{S}}$ are unknown, a critic network with the estimated weights $\hat{\omega}_{\mathcal{S}}$ is established to approximate the cost function as

$$J_{\mathcal{S}}(b) = \hat{\omega}^T_{\mathcal{S}}\delta_{\mathcal{S}}(b) \tag{57}$$

and then the estimated feedback control law can be derived as

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} -\frac{1}{2}\mathcal{G}^T(b)\big(\nabla\delta_{\mathcal{S}}(b)\big)^T\hat{\omega}_{\mathcal{S}} \\ -\frac{1}{2}\mathcal{Z}^T_{\mathcal{B}}(b)\big(\nabla\delta_{\mathcal{S}}(b)\big)^T\hat{\omega}_{\mathcal{S}} \end{bmatrix}. \tag{58}$$

Note that even though we update both u and v in the optimal learning process, only u will be applied on the robust tracking control process.

Based on the neural network design, the approximate Hamiltonian can be built as

$$\begin{aligned}\mathcal{H}_{\mathcal{S}}(b,u,v,\nabla\delta^T_{\mathcal{S}}\hat{\omega}_{\mathcal{S}}) &= \mathcal{V}^2_h(b) + \sigma\mathcal{A}^2_\xi(b) + b^T\mathcal{M}_I b \\ &- \frac{1}{4}\hat{\omega}^T_{\mathcal{S}}\nabla\delta_{\mathcal{S}}(b)\mathcal{G}(b)\mathcal{G}^T(b)(\nabla\delta_{\mathcal{S}}(b))^T\hat{\omega}_{\mathcal{S}} - \frac{1}{4}\hat{\omega}^T_{\mathcal{S}}(b) \\ &\cdot\nabla\delta_{\mathcal{S}}(b)\mathcal{Z}_{\mathcal{B}}(b)\mathcal{Z}^T_{\mathcal{B}}(b)(\nabla\delta_{\mathcal{S}}(b))^T\hat{\omega}_{\mathcal{S}} + \hat{\omega}^T_{\mathcal{S}}\nabla\delta_{\mathcal{S}}(b)\mathcal{F}(b).\end{aligned} \tag{59}$$

Since $\mathcal{H}_{\mathcal{S}}(b,u^*,v^*,\nabla J^*_{\mathcal{S}}) = 0$, we define $e_{\mathcal{S}} = \mathcal{H}_{\mathcal{S}}(b,u,v,\nabla\delta^T_{\mathcal{S}}\hat{\omega}_{\mathcal{S}})$ and the objective error function as $E_{\mathcal{S}} = 0.5e^2_{\mathcal{S}}$. Then, we have the weight adjustment rule derived as

$$\dot{\hat{\omega}}_{\mathcal{S}} = -\alpha_{\mathcal{S}}\Theta\Big(\mathcal{V}^2_h(b) + \sigma\mathcal{A}^2_\xi(b) + \Lambda_{\mathcal{S}}(b,u,v) + \kappa^T_{\mathcal{S}}\hat{\omega}_{\mathcal{S}}\Big) \tag{60}$$

where $\alpha_{\mathcal{S}} > 0$ is the learning rate, $\Theta = \kappa_{\mathcal{S}}/(1+\kappa^T_{\mathcal{S}}\kappa_{\mathcal{S}})^2$, and $\kappa_{\mathcal{S}} = \nabla\delta_{\mathcal{S}}(b)\big(\mathcal{F}(b) + \mathcal{G}(b)u + \mathcal{Z}_{\mathcal{B}}(b)v\big)$. The algorithm of this developed method is provided in Algorithm 2.

Algorithm 2 Controller Design with Mismatched Perturbation

Select the learning time $T_{\mathcal{S}1} > 0$ and the execution time $T_{\mathcal{S}2} > 0$. Set parameters σ, $\mathcal{V}_h(b)$, $\mathcal{A}_\xi$, $\mathcal{M}_I$, $\alpha_{\mathcal{S}}$, and a threshold $\nu_{\mathcal{S}}$.
Begin the learning process
Initialize
for $t = 0 \to T_{\mathcal{S}1}$ **do**
  Compute $[u^T, v^T]^T$ using (58);
  Take action $[u^T, v^T]^T$ and collect b from (24);
  Update $J_{\mathcal{S}}(b)$ through (57);
  Update $\hat{\omega}_{\mathcal{S}}$ from (60);
  if $\|\Delta\hat{\omega}_{\mathcal{S}}\| < \nu_{\mathcal{S}}$ **then**
    Stop;
  end if
end for
return $\hat{\omega}_{\mathcal{S}}$ and fix it.
Begin the robust control process
for $t = 0 \to T_{\mathcal{S}2}$ **do**
  Compute u using the first equation of (58);
  Take action u and collect x from (4);
end for

In this way, we develop the online learning process with critic-only structure. Specifically, the critic network is constructed based on (51) and (57) to estimate the cost function for converted systems, and the control law is determined using (52) and (58). The critic network weights are updated based on (54) and (60). These weight updates result in an enhanced cost function, which, in turn, facilitate the calculation of an improved control law. This iterative process continuously improves both the cost function and the control law, which ultimately achieves optimal control performance.

B. Stability Analysis

The stability analysis of the designed online learning method is discussed. For the case with matched perturbation, consider the nominal system (8) and establish the critic network (51) with weight updating rule (54). Define the Lyapunov function as $L_{mat} = L_{\mathcal{T}}(\rho) + L(\tilde{\omega}_c)$, where $L_{\mathcal{T}}(\rho) = J^*_{\mathcal{T}}(\rho)$ and $L(\tilde{\omega}_c) = \alpha_c^{-1}tr(\tilde{\omega}^T_c\tilde{\omega}_c)$ with $\tilde{\omega}_c = \omega_c - \hat{\omega}_c$ as the weight estimation error. We can obtain $\tilde{\omega}_c$ is UUB. Then, for the system with mismatched perturbation, the nominal plant is established in (24). Design the critic network as (57) with weight updating rule (60). Construct the Lyapunov function as $L_{mis} = L_{\mathcal{S}}(b) + L(\tilde{\omega}_{\mathcal{S}})$, where $\tilde{\omega}_{\mathcal{S}} = \omega_{\mathcal{S}} - \hat{\omega}_{\mathcal{S}}$ is the weight estimation error, $L_{\mathcal{S}}(b) = J^*_{\mathcal{S}}(b)$, and $L(\tilde{\omega}_{\mathcal{S}}) = \alpha_{\mathcal{S}}^{-1}tr(\tilde{\omega}^T_{\mathcal{S}}\tilde{\omega}_{\mathcal{S}})$.

This article has been accepted for publication in IEEE Transactions on Artificial Intelligence. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAI.2023.3346334

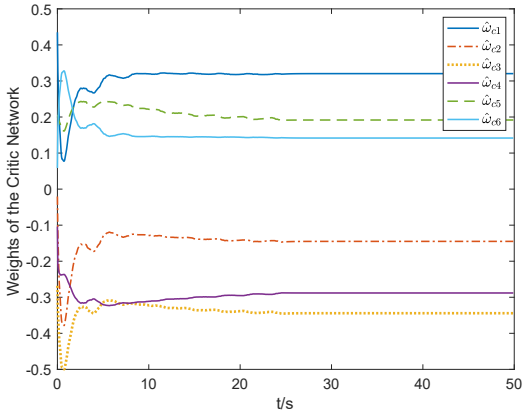

Fig. 1. Convergence process of critic network weights for case study 1.

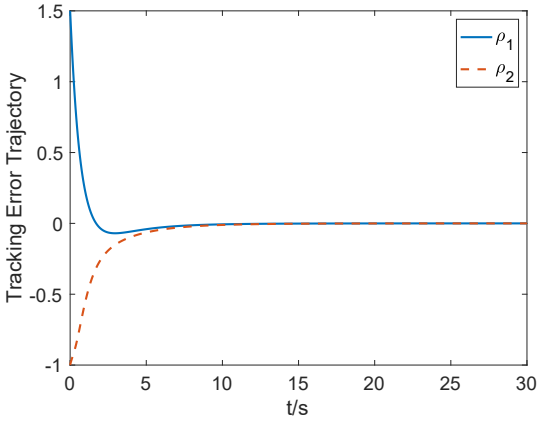

Fig. 3. Tracking error trajectory in robust control process for case study 1.

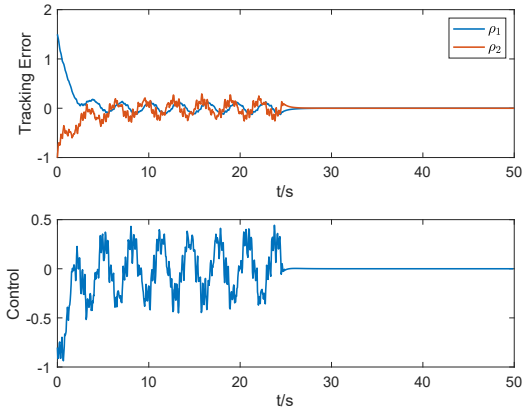

Fig. 2. System response in the critic training process for case study 1.

The UUB of $\tilde{\omega}_S$ can also be obtained. The detailed theoretical study of stability analysis is provided in Appendix.

V. Simulation Studies

Case Study 1: Consider a nonlinear system with the following dynamics

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix} = \begin{bmatrix}x_1^2\sin^2 x_2 + x_2\\ x_1^3 - 1.5x_2\cos x_2\end{bmatrix} + \begin{bmatrix}0\\1\end{bmatrix}u + \begin{bmatrix}0\\1\end{bmatrix}o_{\mathcal{T}} \tag{61}$$

where $x = [x_1, x_2]^T$ is the system state with the initial value as $x(0) = [1, -0.5]^T$, u is the control law and $\xi(x) = o_{\mathcal{T}}$ is the perturbation which is given as

$$o_{\mathcal{T}} = \lambda_1 x_2 \sin(\lambda_2 x_1 x_2) \tag{62}$$

Here, $\lambda_1 \in [-1, 1]$ and $\lambda_2 \in [-100, 100]$ are the unknown parameters. Therefore, the perturbation term is upper bounded by $\|o_{\mathcal{T}}\| \le \|x\| \triangleq \xi_f(x)$. In this case, the term $o_{\mathcal{T}}$ can be considered as the perturbation applied directly on the actuator. We have $h(x) = d(x) = [0, 1]^T$, which means the system (61) contains a matched perturbation.

Assume the reference dynamics are provided as

$$\begin{bmatrix}\dot{r}_1\\ \dot{r}_2\end{bmatrix} = \begin{bmatrix}-r_1\cos^2 r_2 + r_2\\ -0.2r_1 - \sin r_2\end{bmatrix} \tag{63}$$

where $r = [r_1, r_2]^T$ is the reference state with the initial value as $r(0) = [-0.5, 0.5]^T$. Since the tracking error $\rho = x - r$, we have $\rho(0) = [1.5, -1]^T$. To transfer this robust tracking problem into a stabilization design, we develop the modified cost function (9) with the parameters selected as $\eta = 1.5$, $\mathcal{M} = I_2$ and $\mathcal{R} = I$.

Design the nominal plant of the tracking error ρ based on (8). Note that the open loop of the nominal plant is unstable. A critic network is constructed to estimate the cost function and iteratively improve both the cost function and control law. The neuron structure of the critic network is designed as $5 - 6 - 1$, i.e., 5 input neurons, 6 hidden neurons, and 1 output neuron. The initial weights are chosen randomly within $[-0.5, 0.5]$. Set the learning rate of the critic network as $\alpha_c = 0.01$ and select the sampling interval as $0.05s$. To guarantee the persistent excitation condition, we add a small exploratory signal [32] $n(t) = \sin^2(t)\cos(t) + \sin^2(2t)\cos(0.1t) + \sin^2(1.2t)\cos(0.5t) + \sin^5(t) + \sin^2(1.12t) + \cos(2.4t)\sin^3(2.4t)$ to the control $u(t)$ for the first 25s of the learning process.

The learning process spans a total of $50s$ and the convergence evolution of critic network weights between the hidden and output layers is provided in Fig. 1. We can observe that the weights can quickly converge to a stable result $[0.3200, -0.1473, -0.3469, -0.3071, 0.1891, 0.1507]^T$. This verifies the optimal control process of the designed neural-reinforcement-learning-based method. The system response of the critic training process is provided in Fig.2. The oscillation reflects the probing noise to the control signal. After that, we fix the learned weights and build the feedback controller based on (52). We apply the designed controller on the original perturbed tracking system for $30s$ to verify the constructed robust-optimal transformation. In this case study, we select the perturbed parameters as $\lambda_1 = -0.5$ and $\lambda_2 = 100$. Fig. 3 and Fig. 4 show the tracking error and system state trajectories, respectively. We can observe that, under the designed control method, the tracking error can quickly converge to zero. This implies that the control law derived from the transformed stabilization design is the result of the original robust problem and can guide the system towards the

This article has been accepted for publication in IEEE Transactions on Artificial Intelligence. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAI.2023.3346334

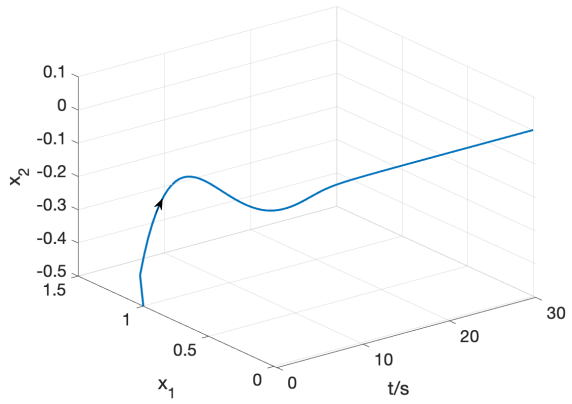

Fig. 4. State trajectory in robust control process for case study 1.

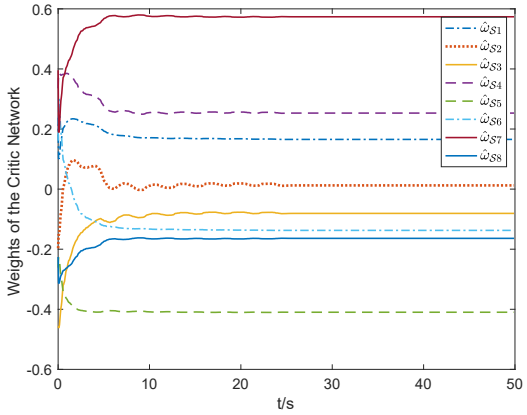

Fig. 5. Convergence process of critic network weights for case study 2.

desired trajectory in the presence of admissible perturbations.

Case Study 2: Now we revise the system (61) to contain an additional perturbation term $o_{\mathcal{S}}$ as

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix}=\begin{bmatrix}x_1^2\sin^2 x_2+x_2\\ x_1^3-1.5x_2\cos x_2\end{bmatrix}+\begin{bmatrix}0\\1\end{bmatrix}u+\begin{bmatrix}1&0\\0&1\end{bmatrix}\begin{bmatrix}o_{\mathcal{S}}\\ o_{\mathcal{T}}\end{bmatrix} \tag{64}$$

where $o_{\mathcal{T}}$ is given in (62) and $o_{\mathcal{S}}$ is provided as

$$o_{\mathcal{S}}=\lambda_3 x_1\cos\Big(\frac{1}{x_2+\lambda_4}\Big) \tag{65}$$

in which $\lambda_3\in[-1,1]$ and $\lambda_4\in[-100,0)\bigcup(0,100]$ are the unknown parameters. The perturbation in this case study can be considered as not only applied on the actuator, but also on the system sensor. Therefore, we have $h(x)\neq d(x)$ in this case study, which means the system contains a mismatched perturbation.

Considering the tracking error $\rho=x-r$ and the augment state $b=[\rho^T,r^T]^T$, we have the augment system dynamics as

$$\dot{b}=\mathcal{F}(b)+\mathcal{G}(b)u+\begin{bmatrix}\lambda_3(b_1+b_3)\cos\Big(\frac{1}{b_2+b_4+\lambda_4}\Big)\\ \lambda_1(b_2+b_4)\sin\Big(\lambda_2(b_1+b_3)(b_2+b_4)\Big)\\ 0\\ 0\end{bmatrix} \tag{66}$$

where

$$\mathcal{F}(b)=\begin{bmatrix}(b_1+b_3)^2\sin^2(b_2+b_4)+(b_2+b_4)+b_3\cos^2 b_4+b_4\\ (b_1+b_3)^3-1.5(b_2+b_4)\cos(b_2+b_4)+0.2b_3-1.1b_4\sin b_4\\ -b_3\cos^2 b_4+b_4\\ -0.2b_3-1.1b_4\sin b_4\end{bmatrix} \tag{67}$$

$$\mathcal{G}(b)=[0,1,0,0]^T \tag{68}$$

The last term in (66) describes the perturbation and based on (22), it can be divided as

$$\mathcal{Z}_{\mathcal{A}}(b)\xi(b)+\mathcal{Z}_{\mathcal{B}}(b)\xi(b) \tag{69}$$

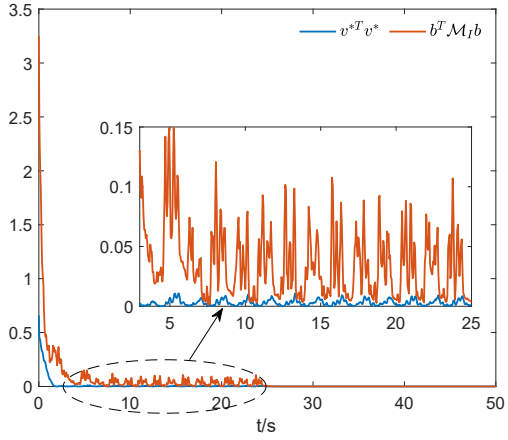

Fig. 6. Stability condition $v^{*T}v^*\leq b^T\mathcal{M}_I b$ verification in the learning process for case study 2.

Since $h^+(\rho+r)=\Big(h^T(\rho+r)h(\rho+r)\Big)^{-1}h^T(\rho+r)=[0,1]$, it follows

$$\mathcal{Z}_{\mathcal{A}}=\begin{bmatrix}0&0\\0&1\\0&0\\0&0\end{bmatrix},\quad \mathcal{Z}_{\mathcal{B}}=\begin{bmatrix}1&0\\0&0\\0&0\\0&0\end{bmatrix},\quad \xi(b)=\begin{bmatrix}o_{\mathcal{S}}\\ o_{\mathcal{T}}\end{bmatrix} \tag{70}$$

Apply the designed guaranteed cost control method to solve this robust tracking problem. We build the nominal plant based on (24) with $\mathcal{F}(b)$, $\mathcal{G}(b)$ and $\mathcal{Z}_{\mathcal{B}}(b)$ given as (67), (68) and (70), respectively. Note that the auxiliary control law $v=[v_1,v_2]^T$ is a two-dimensional variable. But only the first element v_1 is used in the nominal plant considering the design of $\mathcal{Z}_{\mathcal{B}}$. The second element v_2 is used in the cost function to help learn the optimal control law u. Considering the perturbation $\xi(b)$, we have $\|\Xi(b)\xi(b)\|\leq\|\sqrt{(b_1+b_3)^2+(b_2+b_4)^2}\|\triangleq\mathcal{V}_h(b)$ and $\|\xi(b)\|\leq\|\sqrt{(b_1+b_3)^2+(b_2+b_4)^2}\|\triangleq\mathcal{A}_\xi(b)$. By choosing $\sigma=0.4$ and $\mathcal{M}=I_2$, the cost function $J_{\mathcal{S}}(b)$ is designed as

$$J_{\mathcal{S}}(b)=\int_t^\infty\Big\{1.4(b_1(\tau)+b_3(\tau))^2+1.4(b_2(\tau)+b_4(\tau))^2+\Lambda_{\mathcal{S}}(b(\tau),u(\tau),v(\tau))\Big\}d\tau \tag{71}$$

This article has been accepted for publication in IEEE Transactions on Artificial Intelligence. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAI.2023.3346334

Establish the critic network to learn the cost function $J_{\mathcal{S}}(b)$ and help develop the feedback control law u. The initial weights of the critic network are randomly chosen within $[-0.5, 0.5]$. Select the learning rate as $\alpha_{\mathcal{S}} = 0.01$ and the sampling interval as $0.05s$. To test the performance of the developed method, we consider that the perturbation applied on the system jumps among four stages with the parameters selected as follows:
Stage 1: $\lambda_1 = 1$, $\lambda_2 = -100$, $\lambda_3 = -1$, $\lambda_4 = 50$;
Stage 2: $\lambda_1 = -1$, $\lambda_2 = 80$, $\lambda_3 = 1$, $\lambda_4 = -1$;
Stage 3: $\lambda_1 = -1$, $\lambda_2 = -100$, $\lambda_3 = 1$, $\lambda_4 = -100$;
Stage 4: $\lambda_1 = -1$, $\lambda_2 = 100$, $\lambda_3 = -0.5$, $\lambda_4 = -1$;

We conduct the learning process based on the developed neural-reinforcement-learning-based guaranteed cost control method. The small exploratory signal $n(t)$ has also been added to the control for the first 25s to ensure the persistent excitation condition. The evolution of the critic network weights is provided in Fig. 5, which is observed a convergence result to $[0.1654, 0.0124, -0.0808, 0.2531, -0.4098, -0.1374, 0.5736, -0.1639]^T$. The stability condition of the learning process is verified in Fig. 6. We can observe that $v^{*T}v^*$ is consistently smaller than $b^T\mathcal{M}_I b$ which ensures the asymptotic stability of the system. After that, we fix the learned weights $\hat{\omega}_{\mathcal{S}}$ and design the feedback control law based on (58), which is applied on the original perturbed system later. Assume the perturbation starts with Stage 1 and then jumps to Stage 2, 3 and 4 in turn. It lasts 2.5s in Stage 1, 2, and 3, respectively, and then stays in Stage 4. To show the effectiveness of the proposed method, we compare our results with the conventional actor-critic reinforcement learning method, where the controller is designed based on the perturbed formulation directly. The comparisons of the tracking error trajectories under the same initial conditions are provided in Fig. 7. We can observe that our developed robust-optimal transformation design can quickly and effectively minimize the tracking errors. This means the tracking system (64) can accurately follow the reference dynamics in the presence of perturbation, even when the perturbation changes over time. On the other side, the conventional method exhibits higher level of damping in the tracking error trajectory, particularly when transitioning between stages, which leads to system instability. The comparisons of the control laws during this process are provided in Fig. 8. The results indicate that our developed control law based on the transformed design can effectively control the tracking system with admissible mismatched perturbations to achieve expected performance.

Case Study 3 (Triple-link Inverted Pendulum): We have tested our proposed method on a more challenging problem, i.e., triple-link inverted pendulum balancing system (eight state variables). The objective is to maintain stability and balance of the cart and all the pendulums assembly in an inverted position, and also enable the cart to track certain trajectory in the perturbed environment with single control input. This is a very complex and difficult problem due to the highly unstable configuration and non-negligible system nonlinearities. The existence of perturbations and the requirement of trajectory tracking further complicate the balancing process. In this case study, we have successfully implemented our proposed neural-reinforcement-learning-based guaranteed cost control method on this problem.

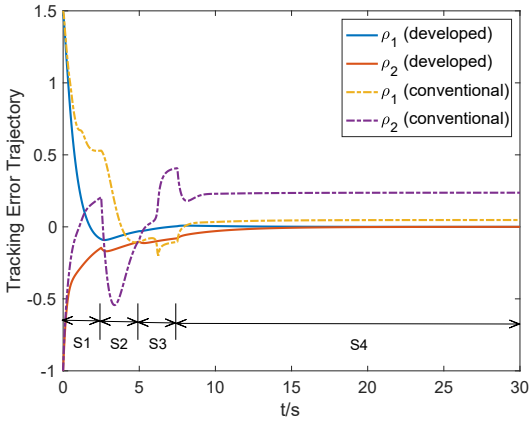

Fig. 7. Tracking error trajectory comparisons of the developed neural-reinforcement-learning-based guaranteed cost control method (developed) and the conventional reinforcement learning method (conventional) for case study 2, where S1: Stage 1, S2: Stage 2, S3: Stage 3, S4: Stage 4.

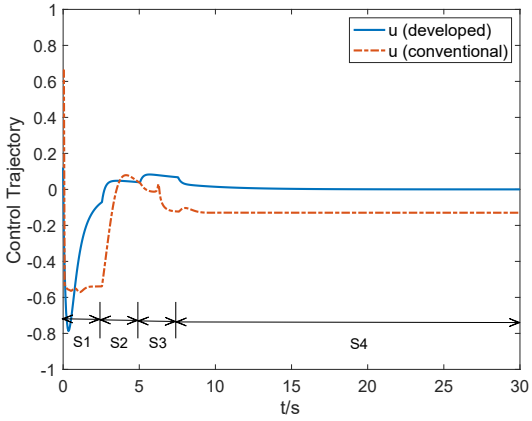

Fig. 8. Control trajectory comparisons of the developed neural-reinforcement-learning-based guaranteed cost control method (developed) and the conventional reinforcement learning method (conventional) for case study 2, where S1: Stage 1, S2: Stage 2, S3: Stage 3, S4: Stage 4.

Specifically, this system includes three pendulum-like links connected in series. These links are equipped with motors that allow them to pivot or rotate around their joints. The entire pendulum mechanism is attached on a cart, which serves as a base and can move horizontally along a track. The system model we considered is the same as that in [39], [40] except the last term becomes

$$L(q,u,\xi) = \begin{bmatrix} K_s u + K_s w - \text{sgn}(s)\mu_x A_{37} \\ -\text{sgn}(\theta_1)\mu_1 A_{38} \\ -\text{sgn}(\theta_2)\mu_1 A_{39} \\ -\text{sgn}(\theta_3)\mu_1 A_{40} \end{bmatrix} \tag{72}$$

where w is the unknown perturbation which is in voltage and converted into force by $K_s = 24.7125N/V$, s is the position of the cart on the track, θ_1 is the vertical angle of the 1st link joint to the cart, θ_2 is vertical angle of the 2nd link joint to the 1st link, θ_3 is the vertical angle of the 3rd link joint to the 2nd link. Note that we use s to represent the cart position

This article has been accepted for publication in IEEE Transactions on Artificial Intelligence. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAI.2023.3346334

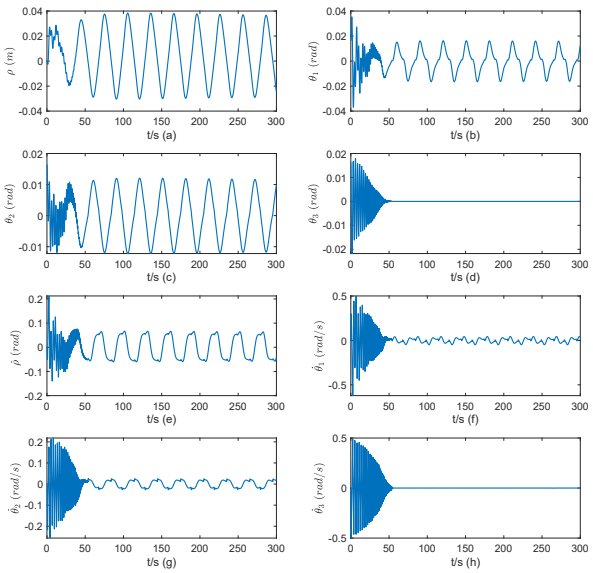

Fig. 9. Typical trajectories of a successful trail on the triple-link inverted pendulum balancing system with perturbation: (a) tracking error of cart position; (b) vertical angle of the 1st link joint to the cart; (c) vertical angle of the 2nd link joint to the 1st link; (d) vertical angle of the 3rd link joint to the 2nd link; (e) velocity error of the cart; (f) angular velocity of the 1st link joint to the cart; (g) angular velocity of the 2nd link joint to the 1st link; (h) angular velocity of the 3rd link joint to the 2nd link.

rather than x in [39], [40] to avoid any conflict with the state variable defined in this paper.

Our goal is to balance the triple-link inverted pendulum system and let the trajectory of the cart follows a sine wave $\dot{r} = \sin(2\pi f t)$ where $f = 0.02$ is the frequency of the wave. We set the following constraints: (1) the cart track should be within $1m$ to both sides from the center point (reference trajectory); (2) each link angle should be within the range of $[-20°, 20°]$ with respect to the vertical axis. For all these two conditions, if either one fails or both fail, we consider the designed controller fails the task.

Therefore, define an eight-dimensional state for the system as $x = [\rho, \theta_1, \theta_2, \theta_3, \dot{\rho}, \dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3]$, where $\rho = s - r$ is the tracking error, $\dot{\rho} = \dot{s} - \dot{r}$ is the velocity error of the cart, $\dot{\theta}_1$ is the angular velocity of the 1st link, $\dot{\theta}_2$ is the angular velocity of the 2nd link, and $\dot{\theta}_3$ is the angular velocity of the 3rd link. The perturbation is assumed as $w = k_1 x_2 \sin(x_1 x_2) + k_2 x_1 \cos(x_3 x_4)$ with the unknown parameters $k_1, k_2 \in [-1, 1]$. Hence, the upper bound is defined as $\xi_f = \|x\|$. We apply the method developed in this paper to solve the problem. The typical robust control trajectories of a successful run for all the state vectors are provided in Fig. 9, i.e., (a) tracking error of cart position; (b)-(d) the joint angle of the 1st, 2nd, and 3rd link of the pendulum, respectively; (e) velocity error of the cart; and (f)-(h) the angular velocity of the 1st, 2nd, and 3rd link of the pendulum, respectively. It is shown that the designed control law can balance the system state in the perturbed environment. Particularly, the tracking error of the cart (Fig. 9 (a)) is controlled within $[-0.04, 0.04]$ which is relatively small comparing to the reference wave magnitude 1. The vertical angles of all the links (Fig. 9 (b)-(d)) are within the admissible ranges to ensure the balance of the system. Fig. 10 compares the cart trajectory and the desired reference in the robust control process. We can observe that with the learned control law, the cart can track the reference dynamics, i.e., sine wave. Therefore, the results indicate that our developed method is effective in this challenging problem, which again demonstrate the validity of the method.

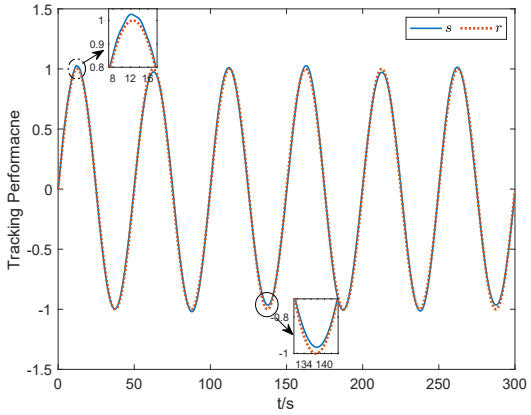

Fig. 10. Comparison of the cart position s and reference trajectory r in the robust control process.

VI. Conclusion and Future Plan

In this paper, we design a data-driven guaranteed cost control method to solve the robust trajectory tracking problem for matched and mismatched perturbed systems. The problem has been converted into the corresponding stabilization design with appropriate cost function. The equivalent analysis of the robust-optimal transformation is provided explicitly for the matched and mismatched cases respectively. The reinforcement learning method is developed with the neural networks implementation to adaptively learn the optimal control law and also guarantee the boundedness of the given cost function. The simulation studies demonstrate the effectiveness and adaptability of the developed method.

Furthermore, we also notice several directions for future work and improvement. For example, it would be useful to expand our results to encompass reference trajectories that not only specify desired state profiles but also prescribe input signals. This expansion will enable us to address a broader range of tracking problems. We also intend to investigate the development of adaptive strategies that can adjust control inputs in response to variations in reference trajectories, which will make our approach more versatile and robust, and ultimately contribute to its broader utility in real-world applications.

Appendix

Stability Analysis of Online Learning Method: For the case with matched perturbation, consider the nominal system (8)

and establish the critic network (51) with weight updating rule (54). Set the estimation error of the critic network weights as $\tilde{\omega}_c = \omega_c - \hat{\omega}_c$ and define the Lyapunov function as

$$L_{mat} = L_{\mathcal{T}}(\rho) + L(\tilde{\omega}_c) \tag{73}$$

where $L_{\mathcal{T}}(\rho) = J^*_{\mathcal{T}}(\rho)$ and $L(\tilde{\omega}_c) = \alpha_c^{-1} tr(\tilde{\omega}_c^T \tilde{\omega}_c)$.

The first derivative of (73) is given as $\dot{L}_{mat} = \dot{L}_{\mathcal{T}}(\rho) + \dot{L}(\tilde{\omega}_c)$. Based on Theorem 1, we have $\dot{L}_{\mathcal{T}}(\rho) \leq -\rho^T \mathcal{M} \rho < 0$ as long as $\eta > \lambda_{\max}(\mathcal{R})$. Therefore, we focus on the second term $\dot{L}(\tilde{\omega}_c)$, which is

$$\dot{L}(\tilde{\omega}_c) = \alpha_c^{-1} tr\Big(\alpha_c \tilde{\omega}_c^T \frac{\kappa}{(1+\kappa^T\kappa)^2}\big(\eta\xi_f^2(\rho,r) + \Lambda(\rho,u) + \kappa^T\hat{\omega}_c\big)\Big). \tag{74}$$

Considering the fact $\hat{\omega}_c = \omega_c - \tilde{\omega}_c$, we have

$$\dot{L}(\tilde{\omega}_c) = \alpha_c^{-1} tr\Big(- \alpha_c \tilde{\omega}_c^T \frac{\kappa\kappa^T}{(1+\kappa^T\kappa)^2}\tilde{\omega}_c + \alpha_c \tilde{\omega}_c^T \cdot \frac{\kappa}{(1+\kappa^T\kappa)^2}\big(\eta\xi_f^2(\rho,r) + \Lambda(\rho,u) + \kappa^T\omega_c\big)\Big). \tag{75}$$

We can further rewrite (75) as

$$\dot{L}(\tilde{\omega}_c) \leq - \left\| \frac{\kappa}{1+\kappa^T\kappa} \right\|^2 \left\|\tilde{\omega}_c\right\|^2 + \frac{\alpha_c}{2} \left\| \frac{\kappa}{1+\kappa^T\kappa} \right\|^2 \left\|\tilde{\omega}_c\right\|^2 + \frac{\left\|\eta\xi_f^2(\rho,r) + \Lambda(\rho,u) + \kappa^T\omega_c\right\|^2}{\alpha_c(1+\kappa^T\kappa)^2}. \tag{76}$$

Define $\Phi_{\mathcal{T}} = \frac{\kappa}{1+\kappa^T\kappa}$ and $\Omega_{\mathcal{T}} = \eta\xi_f^2(\rho,r) + \Lambda(\rho,u) + \kappa^T\omega_c \leq \bar{\Omega}_{\mathcal{T}}$. It follows,

$$\dot{L}(\tilde{\omega}_c) \leq - (1 - \frac{\alpha_c}{2})\left\|\Phi_{\mathcal{T}}\right\|^2 \left\|\tilde{\omega}_c\right\|^2 + \frac{\bar{\Omega}_{\mathcal{T}}}{2\alpha_c}. \tag{77}$$

Therefore, we obtain $\dot{L}(\tilde{\omega}_c) < 0$ as long as the conditions $0 < \alpha_c < 2$ and $\|\tilde{\omega}_c\|^2 > \frac{\bar{\Omega}_{\mathcal{T}}}{\alpha_c(2-\alpha_c)\|\Phi_{\mathcal{T}}\|^2}$ hold. It follows $\dot{L}_{mat} = \dot{L}_{\mathcal{T}}(\rho) + \dot{L}(\tilde{\omega}_c) < 0$, which means the weight estimation error $\tilde{\omega}_c$ is UUB.

Then, for the system with mismatched perturbation, we consider the nominal plant provided in (24). Construct the critic network as (57) with weight updating rule (60). Define the Lyapunov function as

$$L_{mis} = L_{\mathcal{S}}(b) + L(\tilde{\omega}_{\mathcal{S}}) \tag{78}$$

where $\tilde{\omega}_{\mathcal{S}} = \omega_{\mathcal{S}} - \hat{\omega}_{\mathcal{S}}$ is the weight estimation error, $L_{\mathcal{S}}(b) = J^*_{\mathcal{S}}(b)$ and $L(\tilde{\omega}_{\mathcal{S}}) = \alpha_{\mathcal{S}}^{-1} tr(\tilde{\omega}_{\mathcal{S}}^T \tilde{\omega}_{\mathcal{S}})$. Take the first derivative of (78) as $\dot{L}_{mis} = \dot{L}_{\mathcal{S}}(b) + \dot{L}(\tilde{\omega}_{\mathcal{S}})$. According to Theorem 2, we have $\dot{L}_{\mathcal{S}}(b) < 0$ if $v^{*T}v^* \leq b^T\mathcal{M}_I b$. Therefore, it is the term $\dot{L}(\tilde{\omega}_{\mathcal{S}})$ needs to be considered. By setting $\Phi_{\mathcal{S}} = \frac{\kappa_{\mathcal{S}}}{1+\kappa_{\mathcal{S}}^T\kappa_{\mathcal{S}}}$ and $\Omega_{\mathcal{S}} = \mathcal{V}_h^2(b) + \sigma\mathcal{A}_\xi^2(b) + \Lambda_{\mathcal{S}}(b,u,v) + \kappa_{\mathcal{S}}^T\omega_{\mathcal{S}} \leq \bar{\Omega}_{\mathcal{S}}$, we can easily obtain $\dot{L}(\tilde{\omega}_{\mathcal{S}}) < 0$ if $\alpha_{\mathcal{S}} < 2$ and $\|\tilde{\omega}_{\mathcal{S}}\|^2 > \frac{\bar{\Omega}_{\mathcal{S}}}{\alpha_{\mathcal{S}}(2-\alpha_c)\|\Phi_{\mathcal{S}}\|^2}$ hold. Thus, we have $\dot{L}_{min} < 0$. This provides the UUB of the weight estimation error $\tilde{\omega}_{\mathcal{S}}$.

REFERENCES

[1] P. J. Werbos, "Intelligence in the brain: A theory of how it works and how to build it," *Neural Networks*, vol. 22, no. 3, pp. 200–212, 2009.

[2] P. J. Werbos, "Using ADP to understand and replicate brain intelligence: The next level design," in *2007 IEEE International Symposium on Approximate Dynamic Programming and Reinforcement Learning (ADPRL)*, pp. 209–216, 2007.

[3] B. R. Kiran, I. Sobh, V. Talpaert, P. Mannion, A. A. Al Sallab, S. Yogamani, and P. Pérez, "Deep reinforcement learning for autonomous driving: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 6, pp. 4909–4926, 2021.

[4] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, *et al.*, "Mastering the game of go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, 2016.

[5] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel, T. Lillicrap, K. Simonyan, and D. Hassabis, "A general reinforcement learning algorithm that masters chess, shogi, and go through self-play," *Science*, vol. 362, no. 6419, pp. 1140–1144, 2018.

[6] C. J. Watkins and P. Dayan, "Q-learning," *Machine learning*, vol. 8, pp. 279–292, 1992.

[7] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, Feb. 2015.

[8] A. G. Barto, R. S. Sutton, and C. W. Anderson, "Neuronlike adaptive elements that can solve difficult learning control problems," *IEEE transactions on systems, man, and cybernetics*, no. 5, pp. 834–846, 1983.

[9] A. Al-Tamimi, F. L. Lewis, and M. Abu-Khalaf, "Discrete-time nonlinear HJB solution using approximate dynamic programming: convergence proof," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 38, no. 4, pp. 942–949, 2008.

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

[11] F. Liu, J. Sun, J. Si, W. Guo, and S. Mei, "A boundedness result for the direct heuristic dynamic programming," *Neural Networks*, vol. 32, pp. 229–235, 2012.

[12] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, *et al.*, "Mastering the game of go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, Jan. 2016.

[13] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, *et al.*, "Mastering the game of go without human knowledge," *Nature*, vol. 550, no. 7676, pp. 354–359, 2017.

[14] Z. Guo, H. Ren, H. Li, and Q. Zhou, "Adaptive-critic-based event-triggered intelligent cooperative control for a class of second-order constrained multiagent systems," *IEEE Transactions on Artificial Intelligence*, 2023, in press.

[15] D. Xu and F. Fekri, "Improving actor-critic reinforcement learning via hamiltonian monte carlo method," *IEEE Transactions on Artificial Intelligence*, 2023, in press.

[16] D. V. Prokhorov and D. C. Wunsch, "Adaptive critic designs," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 8, no. 5, pp. 997–1007, 1997.

[17] H. He, Z. Ni, and J. Fu, "A three-network architecture for on-line learning and optimization based on adaptive dynamic programming," *Neurocomputing*, vol. 78, pp. 3–13, 2012.

[18] C. Mu, Y. Zhang, H. Jia, and H. He, "Energy-storage-based intelligent frequency control of microgrid with stochastic model uncertainties," *IEEE Transactions on Smart Grid*, vol. 11, no. 2, pp. 1748–1758, 2019.

[19] A. P. Pope, J. S. Ide, D. Mićović, H. Diaz, J. C. Twedt, K. Alcedo, T. T. Walker, D. Rosenbluth, L. Ritholtz, and D. Javorsek, "Hierarchical reinforcement learning for air combat at darpa's alphadogfight trials," *IEEE Transactions on Artificial Intelligence*, 2023, in press.

[20] X. He and C. Lv, "Robotic control in adversarial and sparse reward environments: A robust goal-conditioned reinforcement learning approach," *IEEE Transactions on Artificial Intelligence*, 2023, in press.

[21] X. Yang and Y. Zhou, "Optimal tracking neuro-control of continuous stirred tank reactor systems: A dynamic event-driven approach," *IEEE Transactions on Artificial Intelligence*, 2023, in press.

[22] 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.

[23] D. Wang, M. Ha, and L. Cheng, "Neuro-optimal trajectory tracking with value iteration of discrete-time nonlinear dynamics," *IEEE Transactions on Neural Networks and Learning Systems*, 2023, in press.
[24] H. Zhang, L. Cui, X. Zhang, and Y. Luo, "Data-driven robust approximate optimal tracking control for unknown general nonlinear systems using adaptive dynamic programming method," *IEEE Transactions on Neural Networks*, vol. 22, no. 12, pp. 2226–2236, 2011.
[25] X. Yang and H. He, "Data-driven robust regulation of nonlinear systems with mismatched disturbances," in *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, pp. 1–8, IEEE, 2017.
[26] F. Lin, "An optimal control approach to robust control design," *International Journal of control*, vol. 73, no. 3, pp. 177–186, 2000.
[27] X. Zhong and Z. Ni, "Data-driven reinforcement learning design for multi-agent systems with unknown disturbances," in *2018 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, IEEE, 2018.
[28] X. Zhong, H. He, and D. V. Prokhorov, "Robust controller design of continuous-time nonlinear system using neural network," in *The 2013 International Joint Conference on Neural Networks (IJCNN)*, Aug. 2013.
[29] M. Lin, B. Zhao, and D. Liu, "Event-triggered robust adaptive dynamic programming for multiplayer stackelberg–nash games of uncertain nonlinear systems," *IEEE Transactions on Cybernetics*, 2023, in press.
[30] D. Wang and D. Liu, "Learning and guaranteed cost control with event-based adaptive critic implementation," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 12, pp. 6004–6014, 2018.
[31] D. M. Adhyaru, I. N. Kar, and M. Gopal, "Fixed final time optimal control approach for bounded robust controller design using Hamilton-Jacobi-Bellman solution," *IET Control Theory and Applications*, vol. 3, no. 1, pp. 1183–1195, 2009.
[32] D. Liu, X. Yang, D. Wang, and Q. Wei, "Reinforcement-learning-based robust controller design for continuous-time uncertain nonlinear systems subject to input constraints," *IEEE Transactions on Cybernetics*, vol. 45, no. 7, pp. 1372–1385, 2015.
[33] D. Wang, "Robust policy learning control of nonlinear plants with case studies for a power system application," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 3, pp. 1733–1741, 2019.
[34] D. Wang, L. Cheng, and J. Yan, "Self-learning robust control synthesis and trajectory tracking of uncertain dynamics," *IEEE Transactions on Cybernetics*, vol. 52, no. 1, pp. 278–286, 2020.
[35] S. Chang and T. Peng, "Adaptive guaranteed cost control of systems with uncertain parameters," *IEEE Transactions on Automatic Control*, vol. 17, no. 4, pp. 474–483, 1972.
[36] H.-N. Wu and H.-X. Li, "A Galerkin/neural-network-based design of guaranteed cost control for nonlinear distributed parameter systems," *IEEE Transactions on Neural Networks*, vol. 19, no. 5, pp. 795–807, 2008.
[37] H. Zhang, Y. Liang, H. Su, and C. Liu, "Event-driven guaranteed cost control design for nonlinear systems with actuator faults via reinforcement learning algorithm," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 11, pp. 4135–4150, 2019.
[38] X. Yang, D. Liu, Q. Wei, and D. Wang, "Guaranteed cost neural tracking control for a class of uncertain nonlinear systems using adaptive dynamic programming," *Neurocomputing*, vol. 198, pp. 80–90, 2016.
[39] J. Si and Y.-T. Wang, "Online learning control by association and reinforcement," *IEEE Transactions on Neural Networks*, vol. 12, no. 2, pp. 264–276, 2001.
[40] H. He, Z. Ni, and J. Fu, "A three-network architecture for on-line learning and optimization based on adaptive dynamic programming," *Neurocomputing*, vol. 78, no. 1, pp. 3–13, 2012.

Xiangnan Zhong (Member, IEEE) is currently an Associate Professor with the Department of Electrical Engineering and Computer Science, Florida Atlantic University (FAU), Boca Raton, FL, USA. Her research interests include computational intelligence, reinforcement learning, cyber-physical systems, networked control systems, neural networks, and optimal control.

Prof. Zhong received the National Science Foundation (NSF) Faculty Early Career Development (CAREER) Award in 2021 and the NSF CRII Award in 2019. She was a recipient of the International Neural Network Society (INNS) Aharon Katzir Young Investigator Award in 2021 and the INNS Doctoral Dissertation Award in 2019. She has been serving as an Associate Editor of *IEEE Transactions on Neural Networks and Learning Systems* (TNNLS) since 2021.

Zhen Ni (Senior Member, IEEE) is currently an Associate Professor with the Department of Electrical Engineering and Computer Science (EECS), Florida Atlantic University (FAU), Boca Raton, FL, USA. His research interests mainly include artificial intelligence, and computational methods, and reinforcement learning. Dr. Ni received the prestigious NSF CAREER Award in 2021. He has been an Associate Editor of *IEEE Internet of Things Journal* since 2021, *IEEE Transactions on Neural Networks and Learning Systems* since 2019, and *IEEE Computational Intelligence Magazine* since 2018.