Nonlinear Dyn (2023) 111:16093–16110
https://doi.org/10.1007/s11071-023-08688-0

ORIGINAL PAPER

Reinforcement learning-based optimal control of unknown constrained-input nonlinear systems using simulated experience

Hamed Jabbari Asl · **Eiji Uchibe**

Received: 17 December 2022 / Accepted: 24 June 2023 / Published online: 19 July 2023

Abstract Reinforcement learning (RL) provides a way to approximately solve optimal control problems. Furthermore, online solutions to such problems require a method that guarantees convergence to the optimal policy while also ensuring stability during the learning process. In this study, we develop an online RL-based optimal control framework for input-constrained nonlinear systems. Its design includes two new model identifiers that learn a system's drift dynamics: a slow identifier used to simulate experience that supports the convergence of optimal problem solutions and a fast identifier that keeps the system stable during the learning phase. This approach is a critic-only design, in which a new fast estimation law is developed for a critic network. A Lyapunov-based analysis shows that the estimated control policy converges to the optimal one. Moreover, simulation studies demonstrate the effectiveness of our developed control scheme.

Keywords Optimal control · Reinforcement learning · Input constraints · Uncertainty

H. J. Asl (✉) · E. Uchibe
Department of Brain Robot Interface, ATR Computational Neuroscience Laboratories, 2-2-2 Hikaridai, Seikacho, Soraku-gun, Kyoto 619-0288, Japan
e-mail: hjabbari@atr.jp

1 Introduction

Optimal control is a demanding design objective of control systems, in which a cost function quantifies the system's desired behavior with the goal of developing a control policy that minimizes the cost function under the constraints of the system dynamics. Solutions to the optimal control problem of dynamic systems generally aim to solve the underlying Hamilton-Jacobi-Bellman (HJB) equation. For nonlinear systems, an HJB equation is a nonlinear partial differential equation whose analytical solution is very difficult or even impossible to derive. Dynamic programming is the classic method used to solve HJB equations; however, since this is a backward approach, only offline solutions are possible, and it is also computationally expensive for complex systems.

Owing to the similarity of reinforcement learning (RL) and optimal control, RL has been widely implemented to solve optimal control problems. In fact, in RL, an agent interacts with its environment and improves its behavior based on the observed reward, where a complete exploration of the environment leads to the agent's optimal behavior. Such performance resembles solutions to optimal control problems in the sense that solving HJB equations gives an evaluation of the control policy, and then the controller is updated based on this evaluation to achieve optimal performance. RL-based approaches that solve optimal control problems are also called adaptive or approximate

dynamic programming (ADP), in which mainly neural networks (NNs) are used to approximately solve dynamic programming in a forward-in-time manner, thus reducing the computational complexity.

RL-based adaptive optimal controllers were first developed for discrete-time systems due to the iterative nature of the ADP approach [1,2]. Extensions of RL-based controllers to continuous-time systems were investigated by several researchers [3–5]. For continuous-time systems, the problem involves several challenges, including guaranteeing convergence to optimal control policies as well as stability during the training process, dealing with the system's model uncertainties, and ensuring that the algorithm is online. For online solutions, Vamvoudakis and Lewis [5] developed an RL-based policy iteration (PI) approach adopting the actor-critic structure, where actor and critic neural networks are simultaneously tuned. Such a method requires a model of the system. One online actor-critic approach developed for linear systems [6], called integral RL, does not require a system's internal dynamics but rather sequentially updates the critic and actor neural networks. A previously proposed PI algorithm [7] deals with unknown system models by incorporating a model identifier in the design.

Although the main goal of RL-based controllers is to approximately solve the optimal control problem, previous works [5,7] achieved this goal by making the system states satisfy a persistence of excitation (PE) condition, which is analogous to the exploration concept in RL. Generally, it is difficult to satisfy the PE condition in practice and impossible to guarantee it in advance. A common practice in designing RL-based controllers that satisfy the PE condition is to add a probing signal to the controller [3,5,7,8]. However, this solution is problematic because the proper design of probing noise requires much trial-and-error development: The probing signal can affect the system's stability, and when to remove the signal from the controller remains unclear. Although one approach [9] utilized the system's recorded values to solve the online optimal control problem without a PE, other results [10,11] showed that simulated experience using the system model can be more effective than recorded experience. Another approach [11] utilized a model identifier that estimates the system dynamics over the entire operating domain to simulate experience. However, the model identifiers that learn the system model over the whole task space are usually slow, especially for complex systems, and thus the system's stability during the learning phase cannot be ensured.

A further issue, which is critical from a practical point of view, is the amplitude limitation on control inputs. Previous designs [5,7,11,12] failed to guarantee that the control commands remain within an admissible range, which may degrade the system performance or even result in instability. Some bounded-input PI algorithms [13–16] require a dynamic model of the system. Other approaches [17,18] guarantee convergence to the optimal problem solution by adding a probing signal, whose effect was not clarified in the stability analysis. The bounded controllers in other works [9,19] used recorded data to deal with satisfying the PE condition; however, this approach still needs to add a finite amount of probing noise as input to the system.

In this study, we designed an RL-based approach for optimal control of input-constrained uncertain nonlinear systems. Our main motivation was to apply the simulated experience concept [11] in developing an RL controller for bounded-input systems. To achieve this goal, two novel model identifiers are incorporated in the design: a slow identifier that learns the system model over the entire operating domain of the system, which is used for experience simulation, and a fast and precise identifier that learns the model along the system trajectory, which keeps the system stable until the slow identifier provides sufficient information to solve the optimal control problem. Unlike most RL approaches that use separate networks for critics and actors as a way to maintain stability, our design is a critic-only approach, which is facilitated by exploiting a bounded feature in the components of the cost function. The update law of the critic network includes a new bounded control term that increases the learning rate. We conducted a Lyapunov-based analysis to show that the closed-loop system is uniformly ultimately bounded (UUB) and that the UUB convergence of the control policy to the optimal policy is guaranteed. Simulation studies demonstrate the effectiveness of our developed control scheme.

The main contributions of this study are as follows.

1. A new online RL method is developed for nonlinear systems that takes into account input constraints. Existing works addressing the same problem require an exploration signal [9,13,15,18], which is difficult to design for unknown systems while ensuring that it does not threaten stability.

2. New model identifiers are developed that guarantee fast convergence and high precision. The identifiers are used to apply the concept of experience simulation in the solution of the RL problem for bounded-input systems.
3. Unlike many online RL approaches [5,7,11,15], our design is a critic-only approach. In addition, a non-standard bounded adaptation term is designed for the critic network, which is shown to significantly increase the rate of convergence.

2 Optimal control problem and system definition

We studied a continuous-time nonlinear system defined as

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

where $\mathbf{x} \in \mathbb{R}^n$ is the system state vector, $\mathbf{f}(\mathbf{x}) \in \mathbb{R}^n$ is the unknown drift dynamics of the system, $\mathbf{g}(\mathbf{x}) \in \mathbb{R}^{n\times m}$ is the input dynamics, which is assumed to be known in this paper, and $\mathbf{u} = [u_1, \cdots, u_m]^\top \in \mathbb{R}^m$ is the control input, which is saturated such that $|u_i| \leq \lambda$ for $i = 1, \cdots, m$, where λ is a known upper constraint of the inputs. Throughout this paper, any variable denoted with a time-dependent argument, such as $\mathbf{x}(t)$, is used interchangeably with the corresponding variable without the argument, such as $\mathbf{x}$.

Assumption 1 The functions $\mathbf{f}(\mathbf{x})$ and $\mathbf{g}(\mathbf{x})$ are second-order differentiable. In addition, $\mathbf{g}(\mathbf{x})$ fulfills the following condition:

$$\|\mathbf{g}(\mathbf{x})\| \leq b_g$$

for some $b_g \in \mathbb{R}^+$ [7,10].

Remark 1 The boundedness of the input dynamics $\mathbf{g}(\mathbf{x})$ that is noted in Assumption 1 is satisfied in many practical systems, including robotic systems [20] and aircraft systems [21].

For $\mathbf{u} \in \Omega_a$, where Ω_a is the set of admissible policies [5], the following performance/value function is introduced [18]:

$$V(\mathbf{x}(t)) \triangleq \int_t^\infty [Q(\mathbf{x}(\tau)) + U(\mathbf{u}(\tau))]\mathrm{d}\tau, \tag{2}$$

where $Q(\mathbf{x}) \in \mathbb{R}$ is a general continuous positive-definite function that penalizes states, and the positive-definite integrand function is defined as

$$U(\mathbf{u}) \triangleq 2\int_{\mathbf{0}}^{\mathbf{u}} (\lambda \tanh^{-1}(\mathbf{v}/\lambda))^\top \mathbf{R}\mathrm{d}\mathbf{v}, \tag{3}$$

where $\mathbf{v} \in \mathbb{R}^m$ is an auxiliary variable and $\mathbf{R} = \mathrm{diag}(\bar{r}_1, \cdots, \bar{r}_m) \in \mathbb{R}^{m\times m}$ is a positive-definite diagonal matrix. This nonquadratic function, which penalizes inputs, is widely adopted in the literature and practical systems to deal with input constraints [4,13,18].

The objective of the optimal controller is to minimize the performance function defined in Eq. (2). To develop this controller, we first differentiated V along the system trajectories (Eq. 1) using Leibniz's rule to achieve the following Bellman equation:

$$\dot{V}(\mathbf{x}) = -Q(\mathbf{x}) - U(\mathbf{u}).$$

Therefore, we can define the following Hamiltonian function:

$$H(\mathbf{x}, \mathbf{u}, \nabla V) \triangleq Q(\mathbf{x}) + U(\mathbf{u}) + \nabla V(\mathbf{x})\left(\mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u}(\mathbf{x})\right), \tag{4}$$

where $\nabla V(\mathbf{x}) \triangleq \partial V(\mathbf{x})/\partial \mathbf{x} \in \mathbb{R}^{1\times n}$ is the gradient of $V(\mathbf{x})$.

Assume $V^*(\mathbf{x})$ is the optimal cost (value) function and that $H(\mathbf{x}, \mathbf{u}, \nabla V^*)$ is the corresponding Hamiltonian. Consequently, the optimal controller can be obtained by employing the stationary condition on the Hamiltonian; in other words, by taking the derivative of the Hamiltonian with respect to $\mathbf{u}$ and setting the obtained equation to zero. Then the optimal controller can be written as [13]

$$\begin{aligned}\mathbf{u}^*(\mathbf{x}) &= \arg\min_{\mathbf{u}\in\Omega_a} [H(\mathbf{x}, \mathbf{u}, \nabla V^*)] \\ &= -\lambda \tanh(\mathbf{D}^*),\end{aligned} \tag{5}$$

where $\mathbf{D}^* \triangleq \frac{1}{2\lambda}\mathbf{R}^{-1}\mathbf{g}(\mathbf{x})^\top \nabla V^*(\mathbf{x})^\top \in \mathbb{R}^n$. Substituting Eq. (5) into Eq. (3) and solving it gives [13]

$$U(\mathbf{u}^*) \triangleq \lambda^2 \bar{\mathbf{R}} \ln(\mathbf{1} - \tanh^2(\mathbf{D}^*)) + \lambda \nabla V^* \mathbf{g} \tanh(\mathbf{D}^*), \tag{6}$$

where $\bar{\mathbf{R}} = [\bar{r}_1, \cdots \bar{r}_m] \in \mathbb{R}^{1\times n}$ and $\mathbf{1} \in \mathbb{R}^n$ is a vector whose elements all equal 1. The optimal performance function and the associated optimal controller satisfy the following Hamilton-Jacobi-Bellman (HJB) equation:

$$H(\mathbf{x}, \mathbf{u}^*, \nabla V^*) = Q(\mathbf{x}) + U(\mathbf{u}^*) + \nabla V^*(\mathbf{f} + \mathbf{g}\mathbf{u}^*) = 0. \tag{7}$$

The optimal controller in Eq. (5) can be obtained by solving the HJB equation (Eq. 7) for the optimal value

V^* and using it in Eq. (5). However, it is generally very difficult, if not impossible, to solve this equation for nonlinear systems. The uncertainty in the dynamic model also increases the technical difficulties.

It was previously shown [13] that the above-mentioned optimal problem can be solved through offline policy iteration in a reinforcement learning-based method by estimating the value function $V(\mathbf{x})$ for a given control policy $\mathbf{u}(\mathbf{x})$ from the Hamiltonian function (or Bellman equation) (Eq. 4) and updating the policy with the estimated value function and the structured form of the controller in Eq. (5). Thus, the algorithm converges to the solution of the HJB equation. An online policy iteration solution was proposed [5] that directly estimated the optimal value function $V^*(\mathbf{x})$ to solve the HJB equation expressed in Eq. (7).

This paper considers an online policy iteration-based method to solve the optimal regulation problem, and the function approximation property of neural networks (NN) estimates both optimal value function V^* and unknown dynamic model $\mathbf{f}$, where the estimations are denoted by $\hat{V}$ and $\hat{\mathbf{f}}$, respectively. Using the estimated values, the approximate HJB equation can be written as

$$H(\mathbf{x}, \hat{\mathbf{x}}, \hat{\mathbf{u}}, \nabla\hat{V}) = Q(\mathbf{x}) + U(\hat{\mathbf{u}}) + \nabla\hat{V}(\hat{\mathbf{f}} + \mathbf{g}\hat{\mathbf{u}}), \quad (8)$$

where $\hat{\mathbf{x}}$ is the estimate of $\mathbf{x}$, and

$$U(\hat{\mathbf{u}}) \triangleq \lambda^2\bar{\mathbf{R}}\ln(\mathbf{1} - \tanh^2(\hat{\mathbf{D}})) + \lambda\nabla\hat{V}\mathbf{g}\tanh(\hat{\mathbf{D}}), \quad (9)$$

in which $\hat{\mathbf{D}} \in \mathbb{R}^n$ is the estimate of $\mathbf{D}^*$, and $\hat{\mathbf{u}} \triangleq -\lambda\tanh(\hat{\mathbf{D}})$.

Bellman residual error δ_B, defined as the error between the actual and approximate HJB equations, is given by

$$\begin{aligned}\delta_B &\triangleq H(\mathbf{x}, \hat{\mathbf{x}}, \hat{\mathbf{u}}, \nabla\hat{V}) - H(\mathbf{x}, \mathbf{u}^*, \nabla V^*) \\ &= H(\mathbf{x}, \hat{\mathbf{x}}, \hat{\mathbf{u}}, \nabla\hat{V}). \end{aligned} \quad (10)$$

The Bellman error is measurable and mainly used to design the tuning rule of the neural network of the value function estimator (critic network). Such a tuning approach requires the system states to satisfy the PE condition in order to guarantee convergence. However, it is difficult in practice to guarantee PE in online estimation because the system needs to visit many points in the state space. Adding probing noise is one possible solution to this problem [5,7,18]. The selection of probing noise requires careful attention because no analytical approach is able to compute probing noise that provides PE to nonlinear systems and, moreover, such added noise may result in instability. Although previous works [9,22] have used the recorded past visited values of the Bellman error to solve the online optimal control problem without PE, the results of other work [11] show that the simulated experience of the Bellman error based on a system model can be more effective than using experienced data.

According to Eqs. (10) and (4), in applying a simulated experience, a uniform estimation of function $\mathbf{f}$ is required over the entire operating domain, which allows us to estimate the value of the Bellman error in unexplored points. Online techniques for estimating $\mathbf{f}$ over the whole task space are generally slow because they need to visit and collect sufficiently rich data. Therefore, using only this type of model identifier in the Bellman error might deteriorate the system performance and cause instability. Therefore, we use both a fast identifier that rapidly estimates the value of $\mathbf{f}$ along the system trajectory and a slow identifier that estimates $\mathbf{f}$ over a large space to simulate experience at the unvisited points. These estimates are denoted as $\hat{\mathbf{f}}$ and $\hat{\mathbf{f}}_s$, and they are used to predict the Bellman error along the trajectory and at unexplored points, respectively. These Bellman errors are used together to estimate the optimal value function. A certain amount of time is required before $\hat{\mathbf{f}}_s$ can become accurate enough to simulate experience, so during this period, the estimated Bellman error using $\hat{\mathbf{f}}$ provides information for stabilizing the system.

In some existing works [18,23], a model identifier is designed to estimate the input dynamics $\mathbf{g}(\mathbf{x})$ and drift dynamics $\mathbf{f}(\mathbf{x})$. However, estimating them separately requires rich information, which is difficult to collect in optimal regulation problems where state and control trajectories quickly converge to zero. Therefore, similar to previous works [7,10], in this study, we assume that the function $\mathbf{g}(\mathbf{x})$ is known and thus only estimate the drift dynamics.

In the following sections, we first present the identifiers that estimate unknown drift dynamics $\mathbf{f}(\mathbf{x})$, and then we present an approach to estimate the optimal value function. The critic-only optimal policy can then be extracted from the estimated value function. A block diagram of the proposed control system is shown in Fig. 1.

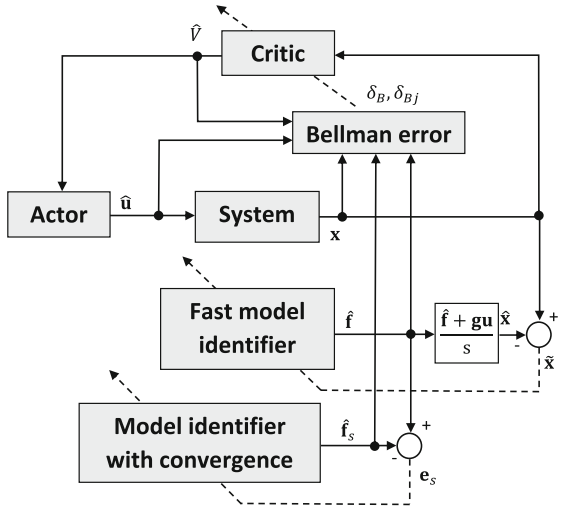

Fig. 1 Block diagram of proposed RL-based optimal controller

3 Identifier design

3.1 Fast model identifier along system trajectories

Our objective here is to design a fast and precise estimator for the drift dynamics $\mathbf{f}(\mathbf{x})$ of the system (Eq. 1). To achieve this, the function approximation property of NNs is used in conjunction with a robust integral of the sign of error (RISE) term to gain continuous estimation with zero error. The prescribed performance feature is also added to satisfy the desired convergence rate.

The function $\mathbf{f}(\mathbf{x})$ can be represented by a neural network over compact set Ω_N:

$$\mathbf{f}(\mathbf{x}) = \mathbf{W}_f^\top \boldsymbol{\sigma}_f(\mathbf{z}) + \boldsymbol{\epsilon}_f, \tag{11}$$

where $\mathbf{W}_f \in \mathbb{R}^{\mathcal{L}_f \times n}$ is a bounded constant ideal weight matrix, $\mathcal{L}_f$ denotes the number of neurons, $\boldsymbol{\sigma}_f(\cdot) \in \mathbb{R}^{\mathcal{L}_f}$ is the activation matrix with vector $\mathbf{z}$ as its input, and $\boldsymbol{\epsilon}_f \in \mathbb{R}^n$ denotes the functional reconstruction error. Activation function $\boldsymbol{\sigma}_f$, error $\boldsymbol{\epsilon}_f$, and their time derivatives are assumed to be bounded [18]. Substituting Eq. (11) into Eq. (1), we obtain $\dot{\mathbf{x}} = \mathbf{W}_f^\top \boldsymbol{\sigma}_f + \boldsymbol{\epsilon}_f + \mathbf{g}(\mathbf{x})\mathbf{u}$.

The identifier state is denoted by $\hat{\mathbf{x}} \in \mathbb{R}^n$, and identification error $\tilde{\mathbf{x}} \in \mathbb{R}^n$ is defined as $\tilde{\mathbf{x}} \triangleq \mathbf{x} - \hat{\mathbf{x}}$. The objective is to design an identifier such that error $\tilde{\mathbf{x}}$ and its time derivative converge to zero. Accordingly, the identifier can be used as an estimator of the system model. To gain a high-performance identifier, we constrain error signal $\tilde{\mathbf{x}} = [\tilde{x}_1, \cdots, \tilde{x}_n]^\top$ to converge with some desired performance conditions. According to Assumption 1, imposing a convergence rate on $\tilde{\mathbf{x}}$ also imposes a convergence rate on the estimation of unknown function $\mathbf{f}(\mathbf{x})$.

To constrain $\tilde{\mathbf{x}}$, a smoothly decreasing performance function $\mu(t) \in \mathbb{R}^+$ is first introduced as [24]

$$\mu(t) \triangleq (\mu_0 - \mu_\infty)\, \mathrm{e}^{-\kappa_\mu t} + \mu_\infty, \tag{12}$$

where $\kappa_\mu > 0$ and $\mu_0 > \mu_\infty > 0$ are design parameters. Then, provided the error signal satisfies the condition

$$-\bar{\delta}_i \mu(t) < \tilde{x}_i(t) < \bar{\delta}_i \mu(t), \tag{13}$$

where $\bar{\delta}_i \in \mathbb{R}^+$ are positive prescribed constants for $i = 1, \cdots, n$, the convergence rate of $\tilde{x}_i(t)$ is guaranteed to be faster than that of function $\mu(t)$. It can be seen from Eqs. (12) and (13) that $-\bar{\delta}_i \mu_0$ and $\bar{\delta}_i \mu_0$ respectively signify the lower bound of the undershoot and the upper bound of the overshoot of $\tilde{x}_i(t)$. In addition, the lower bound of the convergence rate of error is given by κ_μ.

The following transformation is assumed to transform the constrained error $\tilde{\mathbf{x}}$ into an unconstrained one:

$$\tilde{x}_i(t) = \mu(t) T_i(\varsigma_i), \tag{14}$$

where $\varsigma_i \in \mathbb{R}$ denotes the transformed unconstrained error, and $T_i(\varsigma_i) \in \mathbb{R}$ is a strictly increasing function satisfying the following criteria:

$$-\bar{\delta}_i < T_i(\varsigma_i) < \bar{\delta}_i, \quad \lim_{\varsigma_i \to -\infty} T_i(\varsigma_i) = -\bar{\delta}_i, \tag{15}$$

$$\lim_{\varsigma_i \to +\infty} T_i(\varsigma_i) = \bar{\delta}_i, \tag{16}$$

which is defined as [24]

$$T_i(\varsigma_i) \triangleq \frac{\bar{\delta}_i \mathrm{e}^{\varsigma_i} - \bar{\delta}_i \mathrm{e}^{-\varsigma_i}}{\mathrm{e}^{\varsigma_i} + \mathrm{e}^{-\varsigma_i}}. \tag{17}$$

Considering the properties of $T_i(\varsigma_i)$ and the fact that $\mu(t) > 0$, we conclude that inverse transformation $\varsigma_i(t) = T_i^{-1}[\tilde{x}_i(t)/\mu(t)]$ is properly defined provided that ς_i is bounded. Consequently, to achieve the prescribed performance condition (Eq. 13), we only need to guarantee the boundedness of ς_i.

Using Eq. (17), ς_i can be written as

$$\varsigma_i(t) = \frac{1}{2} \ln \frac{\bar{\delta}_i + \tilde{x}_i(t)/\mu(t)}{\bar{\delta}_i - \tilde{x}_i(t)/\mu(t)}. \tag{18}$$

Then the time derivative of ς_i can be obtained as $\dot{\varsigma}_i = r_i(\dot{\tilde{x}}_i - \frac{\dot{\mu}}{\mu}\tilde{x}_i)$, where $r_i \triangleq \phi_i/\mu$ with $\phi_i \in \mathbb{R}$ is defined as $\phi_i \triangleq \frac{1}{2}(\frac{1}{\bar{\delta}_i + \tilde{x}_i/\mu} + \frac{1}{\bar{\delta}_i - \tilde{x}_i/\mu})$. Considering the properties

of the transformation, it can be concluded that $r_i > 0$ provided that Eq. (13) is satisfied. For an n-dimensional system, the transformed error vector can be defined as $\boldsymbol{\varsigma} \triangleq [\varsigma_1 \cdots \varsigma_n]^\top \in \mathbb{R}^n$, whose time derivative can be written as follows:

$$\dot{\boldsymbol{\varsigma}} = \boldsymbol{\Upsilon}\left(\dot{\tilde{\mathbf{x}}} - \ell\tilde{\mathbf{x}}\right), \tag{19}$$

where $\ell \triangleq \dot{\mu}/\mu \in \mathbb{R}$, and diagonal matrix $\boldsymbol{\Upsilon} \in \mathbb{R}^{n\times n}$ can be defined as $\boldsymbol{\Upsilon} \triangleq \text{diag}\{r_1, \cdots, r_n\}$.

The following identifier is proposed to approximate the system given by Eq. (1):

$$\dot{\hat{\mathbf{x}}} \triangleq \hat{\mathbf{W}}_f^\top \hat{\boldsymbol{\sigma}}_f + \mathbf{g}(\mathbf{x})\mathbf{u} - \ell\tilde{\mathbf{x}} + \boldsymbol{\Upsilon}^{-1}\boldsymbol{\nu}, \tag{20}$$

where $\hat{\mathbf{W}}_f \in \mathbb{R}^{\mathcal{L}_f\times n}$ is the estimated weight matrix, $\hat{\boldsymbol{\sigma}}_f \triangleq \boldsymbol{\sigma}_f(\hat{\mathbf{x}})$, and $\boldsymbol{\nu} \in \mathbb{R}^n$ is the RISE term defined as [25]

$$\boldsymbol{\nu} \triangleq k_f\boldsymbol{\varsigma} - k_f\boldsymbol{\varsigma}(0) + \int_0^t [\alpha k_f\boldsymbol{\varsigma}(\bar{t}) + \beta_1 \text{sgn}(\boldsymbol{\varsigma}(\bar{t}))]\text{d}t,$$

in which $k_f, \alpha, \beta_1 \in \mathbb{R}^+$ are constant design parameters. Having $\dot{\hat{\mathbf{x}}}$ through Eq. (20), the estimate of drift dynamics $\mathbf{f}$ in Eq. (1) can be given as

$$\hat{\mathbf{f}} \triangleq \dot{\hat{\mathbf{x}}} - \mathbf{g}(\mathbf{x})\mathbf{u}. \tag{21}$$

The first term in the right-hand side of Eq. (20) is a neural network-based estimate of function $\mathbf{f}(\mathbf{x})$. The second component is a feedforward term that utilizes known information in the system dynamics (Eq. 1). The term $-\ell\tilde{\mathbf{x}}$ is included in Eq. (20) to cancel out the effect of its counterpart in Eq. (19), and the last component in Eq. (20) includes the RISE term featured by the prescribed performance property. This term applies feedback information and compensates all residual estimation errors such that the prescribed performance condition (Eq. 13) is fulfilled and the estimation error converges to zero.

Now, utilizing Eqs. (1) and (20), the identification error dynamics can be developed as

$$\dot{\tilde{\mathbf{x}}} = \mathbf{W}_f^\top \boldsymbol{\sigma}_f - \hat{\mathbf{W}}_f^\top \hat{\boldsymbol{\sigma}}_f + \boldsymbol{\epsilon}_f + \ell\tilde{\mathbf{x}} - \boldsymbol{\Upsilon}^{-1}\boldsymbol{\nu}. \tag{22}$$

Owing to the definition of $\boldsymbol{\Upsilon}$ and the induced norm for matrices, when the condition set in Eq. (13) is satisfied, we can consider a lower bound for the norm of $\boldsymbol{\Upsilon}$ as $\|\boldsymbol{\Upsilon}\| \geq \phi_m/\mu$, where ϕ_m denotes the minimum value of ϕ_i for $i = 1, \cdots, n$. Considering this and substituting Eq. (22) into Eq. (19), $\dot{\boldsymbol{\varsigma}}$ can be obtained as

$$\dot{\boldsymbol{\varsigma}} = -k_f\boldsymbol{\varsigma} - \int_0^t [\alpha k_f\boldsymbol{\varsigma}(\bar{t}) + \beta_1 \text{sgn}(\boldsymbol{\varsigma}(\bar{t}))]\text{d}t + \boldsymbol{\Upsilon}\tilde{\mathbf{W}}_f^\top \boldsymbol{\sigma}_f + \mathbf{S} + \boldsymbol{\epsilon}_0, \tag{23}$$

where $\boldsymbol{\epsilon}_0 \triangleq \boldsymbol{\Upsilon}_0\boldsymbol{\epsilon}_f$, $\mathbf{S} \triangleq \boldsymbol{\Upsilon}\hat{\mathbf{W}}_f^\top\tilde{\boldsymbol{\sigma}}_f + \boldsymbol{\Upsilon}_1\boldsymbol{\epsilon}_f$, $\tilde{\boldsymbol{\sigma}}_f \triangleq \boldsymbol{\sigma}_f - \hat{\boldsymbol{\sigma}}_f$, $\boldsymbol{\Upsilon}_0 \triangleq \phi_m/\mu\mathbf{I}_n \in \mathbb{R}^{n\times n}$, $\mathbf{I}_n$ denotes the identity matrix of dimension n, and $\boldsymbol{\Upsilon}_1 \triangleq \boldsymbol{\Upsilon} - \boldsymbol{\Upsilon}_0$. Now auxiliary variable $\mathbf{e}_\varsigma \in \mathbb{R}^n$ is defined as

$$\mathbf{e}_\varsigma \triangleq \dot{\boldsymbol{\varsigma}} + \alpha\boldsymbol{\varsigma}. \tag{24}$$

Taking the time derivative of (23), $\dot{\mathbf{e}}_\varsigma$ can be obtained as

$$\dot{\mathbf{e}}_\varsigma = -k_f\mathbf{e}_\varsigma - \beta_1 \text{sgn}(\boldsymbol{\varsigma}(t)) - \boldsymbol{\varsigma} + \tilde{\mathbf{N}} + \mathbf{N}_B + \dot{\boldsymbol{\epsilon}}_0, \tag{25}$$

where $\tilde{\mathbf{N}}, \mathbf{N}_B \in \mathbb{R}^n$ are defined as

$$\tilde{\mathbf{N}} \triangleq \dot{\boldsymbol{\Upsilon}}\tilde{\mathbf{W}}_f^\top\boldsymbol{\sigma}_f - \boldsymbol{\Upsilon}\dot{\hat{\mathbf{W}}}_f^\top\boldsymbol{\sigma}_f + \boldsymbol{\Upsilon}_1\tilde{\mathbf{W}}_f^\top\dot{\boldsymbol{\sigma}}_f + \boldsymbol{\varsigma} + \dot{\mathbf{S}} + \alpha\dot{\boldsymbol{\varsigma}},$$

and $\mathbf{N}_B \triangleq \boldsymbol{\Upsilon}_0\tilde{\mathbf{W}}_f^\top\dot{\boldsymbol{\sigma}}_f$. Considering Assumption 1, the results from Appendix A in a previous work [26] can be invoked to obtain the following upper bound for $\tilde{\mathbf{N}}$:

$$\|\tilde{\mathbf{N}}\| \leq \rho(\|\boldsymbol{\omega}\|)\|\boldsymbol{\omega}\|, \tag{26}$$

where $\boldsymbol{\omega} \in \mathbb{R}^{2n}$ is defined as $\boldsymbol{\omega} \triangleq [\boldsymbol{\varsigma}^\top \ \mathbf{e}_\varsigma^\top]^\top$ and $\rho(\|\boldsymbol{\omega}\|) \in \mathbb{R}^+$ is a positive globally invertible nondecreasing function. Considering the definitions of $\boldsymbol{\epsilon}_0$ and $\mathbf{N}_B$ and the properties of their components, the following bounds can also be stated:

$$\|\dot{\boldsymbol{\epsilon}}_0\| \leq \zeta_1, \quad \|\ddot{\boldsymbol{\epsilon}}_0\| \leq \zeta_2, \tag{27}$$

$$\|\mathbf{N}_B\| \leq \zeta_3, \quad \|\dot{\mathbf{N}}_B\| \leq \zeta_4 + \zeta_5\|\boldsymbol{\varsigma}\|, \tag{28}$$

where $\zeta_1, \zeta_2, \zeta_3, \zeta_4, \zeta_5 \in \mathbb{R}^+$ are positive constants.

Theorem 1 *For the system defined by Eq. (1) and the adaptation rule for the NN weights given by*

$$\dot{\hat{\mathbf{W}}}_f \triangleq \mathit{Proj}(\gamma\dot{\boldsymbol{\sigma}}_f\boldsymbol{\varsigma}^\top\boldsymbol{\Upsilon}_0), \tag{29}$$

in which $\gamma \in \mathbb{R}^+$ is a positive adaptation gain and $\mathit{Proj}(\cdot)$ denotes the projection operator, provided Assumption 1 holds, the initial conditions satisfy Eq. (13), gain k_f is selected as sufficiently large based on the initial conditions, and the following conditions are satisfied:

$$\beta_1 > \max(\zeta_1 + \zeta_3, \zeta_1 + \frac{\zeta_3 + \zeta_4}{\alpha}), \tag{30}$$

$$\beta_2 > \zeta_5, \ \textit{and} \ \beta_2 < \alpha \tag{31}$$

in which β_2 is defined in Eq. (49), the identifier given in Eq. (20) guarantees the asymptotic identification of the state and its derivative, in the sense that $\|\tilde{\mathbf{x}}\|, \|\dot{\tilde{\mathbf{x}}}\| \to 0$ as $t \to \infty$.

Proof See the proof in Appendix A. □

Remark 2 The neural network used for estimating drift dynamics $\mathbf{f}$ employs the error between $\mathbf{x}$ and $\hat{\mathbf{x}}$ and thus is decoupled from the value function estimator introduced in Sect. 4.

3.2 Model identifier satisfying convergence

The identifier developed in the previous section assures the fast convergence of $\hat{\mathbf{f}}$ to the true values along the system trajectories. However, the convergence of NN weights to true values is not guaranteed. This means that we cannot use $\hat{\mathbf{f}}$ to estimate drift dynamics $\mathbf{f}$ at the unvisited points. It is known from the literature that the convergence of these weights requires a PE condition, which is often difficult to achieve in practice and nearly impossible to verify online. To relax the PE condition while estimating the drift dynamics over the entire operating domain, we follow the experience replay approach [27] that uses recorded input-output data to improve the efficiency of information utilization. Although this method does not require a restrictive PE condition, the recorded data must still be rich enough to estimate the true weights; consequently, the estimation of function $\mathbf{f}$ by this method is much slower than using the method in the previous section, despite providing an opportunity to explore the Bellman error at the unvisited points and relaxing the PE condition required for convergence of the critic weights.

Similar to the previous section, it is assumed that function $\mathbf{f}(\mathbf{x})$ can be represented by an NN as $\mathbf{f} = \mathbf{W}_s^\top \boldsymbol{\sigma}_s + \boldsymbol{\epsilon}_s$, where $\mathbf{W}_s \in \mathbb{R}^{\mathcal{L}_s \times n}$ is an unknown weight matrix, $\boldsymbol{\sigma}_s \in \mathbb{R}^{\mathcal{L}_s}$ is the activation function vector, $\boldsymbol{\epsilon}_s \in \mathbb{R}^n$ is the estimation error, $\mathcal{L}_s$ is the number of neurons, and the following bounds are satisfied:

$$\|\boldsymbol{\epsilon}_s\| \leq b_{\epsilon s}, \quad \|\boldsymbol{\sigma}_s\| \leq b_{\sigma s}, \quad \|\nabla \boldsymbol{\sigma}_s\| \leq b_{\sigma s x},$$

where $b_{\epsilon s}, b_{\sigma s}, b_{\sigma s x} \in \mathbb{R}^+$ are positive constants. Hence, an estimate of $\mathbf{f}$ can be represented as $\hat{\mathbf{f}}_s \triangleq \hat{\mathbf{W}}_s^\top \boldsymbol{\sigma}_s$, where $\hat{\mathbf{W}}_s \in \mathbb{R}^{\mathcal{L}_s \times n}$ is an estimate of $\mathbf{W}_s$. Having an exact estimate of function $\mathbf{f}$ along the system trajectory, obtained through the RISE-based approach in the previous section, function estimation error $\mathbf{e}_s \in \mathbb{R}^n$ is considered to be[1]

$$\begin{aligned}\mathbf{e}_s &\triangleq \mathbf{f} - \hat{\mathbf{f}}_s \\ &= \tilde{\mathbf{W}}_s^\top \boldsymbol{\sigma}_s + \boldsymbol{\epsilon}_s,\end{aligned} \tag{32}$$

where $\tilde{\mathbf{W}}_s \triangleq \mathbf{W}_s - \hat{\mathbf{W}}_s \in \mathbb{R}^{\mathcal{L}_s \times n}$. Now assume that a history stack contains M recorded pairs of $(\mathbf{f}_{sj}, \boldsymbol{\sigma}_{sj})$ for $j = 1 \cdots M$, where subscript j denotes the jth sample in all variables. The function estimation error for each pair of recorded data based on the current estimated weight can be written as

$$\begin{aligned}\mathbf{e}_{sj} &= \mathbf{f}_{sj} - \hat{\mathbf{W}}_s^\top \boldsymbol{\sigma}_{sj} \\ &= \tilde{\mathbf{W}}_s^\top \boldsymbol{\sigma}_{sj} + \boldsymbol{\epsilon}_{sj}.\end{aligned}$$

Assumption 2 The recorded data in the history stack contain sufficient linearly independent elements $\boldsymbol{\sigma}_{sj}$ for $j = 1 \cdots M$ such that

$$0 < c_{\sigma_1} \triangleq \lambda_{\min}\left(\sum_{j=1}^{M} \boldsymbol{\sigma}_{sj} \boldsymbol{\sigma}_{sj}^\top\right), \tag{33}$$

where $\lambda_{\min}$ denotes the minimum eigenvalue. This condition is satisfied when $\operatorname{rank}(\sum_{j=1}^{M} \boldsymbol{\sigma}_{sj} \boldsymbol{\sigma}_{sj}^\top) = \mathcal{L}_s$, which can be easily verified online [10].

Assumption 2 requires the system states to be excited over a finite time interval. This is a less restrictive condition than the PE condition in traditional estimation methods. The PE condition needs the excited states to be present throughout an infinite time period, and it is very difficult or even impossible to verify online. It has been shown that condition (33) can be fulfilled in the RL control of nonlinear systems [11]. It has also been shown [10,28] that one can use an *a priori* available history stack to satisfy Eq. (33).

The objective is to design an estimation law for $\hat{\mathbf{W}}_s$ that guarantees that the estimated weight matrix converges as closely as possible to the true weight matrix $\mathbf{W}_s$. According to the subsequent convergence analysis, the following estimation law is designed:

$$\dot{\hat{\mathbf{W}}}_s \triangleq \boldsymbol{\Gamma}_s \boldsymbol{\sigma}_s \mathbf{e}_s^\top + \gamma_{s1} \boldsymbol{\Gamma}_s \boldsymbol{\Sigma} + \gamma_{s2} \boldsymbol{\Gamma}_s \frac{\boldsymbol{\Sigma}}{\|\boldsymbol{\Sigma}\| + \varepsilon_s}, \tag{34}$$

[1] The accuracy of $\hat{\mathbf{f}}(\mathbf{x})$ can be monitored through the error signal $\tilde{\mathbf{x}}$, and its output can be used in Eq. (32) instead of $\mathbf{f}$ when the error becomes negligible.

where $\gamma_{s1}, \gamma_{s2} \in \mathbb{R}^+$ are positive constants, ε_s is a small positive value, $\boldsymbol{\Sigma} \in \mathbb{R}^{\mathcal{L}_s \times n}$ is defined as $\boldsymbol{\Sigma} \triangleq \sum_{j=1}^{M} \boldsymbol{\sigma}_{sj} \mathbf{e}_{sj}^\top$, and variable least-squares gain matrix $\boldsymbol{\Gamma}_s \in \mathbb{R}^{\mathcal{L}_s \times \mathcal{L}_s}$ is defined as

$$\dot{\boldsymbol{\Gamma}}_s \triangleq \begin{cases} \rho_s \boldsymbol{\Gamma}_s - \boldsymbol{\Gamma}_s (\boldsymbol{\sigma}_s \boldsymbol{\sigma}_s^\top) \boldsymbol{\Gamma}_s, & \text{if } \|\boldsymbol{\Gamma}_s\| \le \bar{\Gamma}_s \\ \mathbf{0}, & \text{otherwise} \end{cases},$$

where $\bar{\Gamma}_s \in \mathbb{R}^+$ is a saturation constant, $\rho_s \in \mathbb{R}^+$ is a forgetting factor, and $\boldsymbol{\Gamma}_s(0)$ is a symmetric positive-definite matrix satisfying $\|\boldsymbol{\Gamma}_s(0)\| \le \bar{\Gamma}_s$. Using (32), (34) and the definition of $\tilde{\mathbf{W}}_s$, the dynamics of $\tilde{\mathbf{W}}_s$ can be written as

$$\begin{aligned} \dot{\tilde{\mathbf{W}}}_s \triangleq & -\boldsymbol{\Gamma}_s \boldsymbol{\sigma}_s (\boldsymbol{\sigma}_s^\top \tilde{\mathbf{W}}_s + \boldsymbol{\epsilon}_s^\top) \\ & - \gamma_{s1} \boldsymbol{\Gamma}_s \sum_{j=1}^{M} (\boldsymbol{\sigma}_{sj} \boldsymbol{\sigma}_{sj}^\top \tilde{\mathbf{W}}_s + \boldsymbol{\sigma}_{sj} \boldsymbol{\epsilon}_{sj}^\top) \\ & - \gamma_{s2} \boldsymbol{\Gamma}_s \frac{\sum_{j=1}^{M} (\boldsymbol{\sigma}_{sj} \boldsymbol{\sigma}_{sj}^\top \tilde{\mathbf{W}}_s + \boldsymbol{\sigma}_{sj} \boldsymbol{\epsilon}_{sj}^\top)}{\|\boldsymbol{\Sigma}\| + \varepsilon_s}. \end{aligned} \tag{35}$$

Remark 3 The last term in Eq. (34) was inspired by previous work [29], where a non-smooth form of it, i.e., $\boldsymbol{\Sigma}/\|\boldsymbol{\Sigma}\|$, is used in the estimation law; moreover, this work showed that, in the absence of function reconstruction error $\boldsymbol{\epsilon}_s$, a finite-time convergence could be achieved. The use of this type of bounded control term (and its variations, e.g., [30]) is a common practice in the control of uncertain systems and improves stability and performance without employing a high-gain control term that may degrade system performance in the presence of noise [30]. Although the last term in Eq. (34) does not guarantee finite-time convergence due to the embedded smoothening term ε_s and the existence of the function estimation error, simulation studies show that this bounded term significantly improves the convergence rate.

The convergence of $\hat{\mathbf{W}}_s$ to the true value can be analyzed by considering the following Lyapunov function:

$$L_{W_s} \triangleq \frac{1}{2} \mathrm{tr}(\tilde{\mathbf{W}}_s^\top \boldsymbol{\Gamma}_s^{-1} \tilde{\mathbf{W}}_s), \tag{36}$$

whose time derivative using (35) can be obtained as

$$\begin{aligned} \dot{L}_{W_s} = & -\mathrm{tr}(\tilde{\mathbf{W}}_s^\top \boldsymbol{\sigma}_s \boldsymbol{\sigma}_s^\top \tilde{\mathbf{W}}_s) - \mathrm{tr}(\tilde{\mathbf{W}}_s^\top \boldsymbol{\sigma}_s \boldsymbol{\epsilon}_s^\top) \\ & - \mathrm{tr}(\gamma_{s1} \sum_{j=1}^{M} \tilde{\mathbf{W}}_s^\top \boldsymbol{\sigma}_{sj} \boldsymbol{\sigma}_{sj}^\top \tilde{\mathbf{W}}_s) \\ & - \mathrm{tr}(\gamma_{s1} \sum_{j=1}^{M} \tilde{\mathbf{W}}_s^\top \boldsymbol{\sigma}_{sj} \boldsymbol{\epsilon}_{sj}^\top) \\ & - \mathrm{tr}(\frac{\gamma_{s2} \sum_{j=1}^{M} \tilde{\mathbf{W}}_s^\top \boldsymbol{\sigma}_{sj} \boldsymbol{\sigma}_{sj}^\top \tilde{\mathbf{W}}_s}{\|\boldsymbol{\Sigma}\| + \varepsilon_s}) \\ & - \mathrm{tr}(\frac{\gamma_{s2} \sum_{j=1}^{M} \tilde{\mathbf{W}}_s^\top \boldsymbol{\sigma}_{sj} \boldsymbol{\epsilon}_{sj}^\top}{\|\boldsymbol{\Sigma}\| + \varepsilon_s}) \\ & - \frac{1}{2} \mathrm{tr}\left(\tilde{\mathbf{W}}_s^\top \boldsymbol{\Gamma}_s^{-1} (\rho_s \boldsymbol{\Gamma}_s - \boldsymbol{\Gamma}_s \boldsymbol{\sigma}_s \boldsymbol{\sigma}_s^\top \boldsymbol{\Gamma}_s) \boldsymbol{\Gamma}_s^{-1} \tilde{\mathbf{W}}_s \right). \end{aligned}$$

By defining $c_{\sigma_2} \triangleq \lambda_{\max}(\sum_{j=1}^{M} \boldsymbol{\sigma}_{sj} \boldsymbol{\sigma}_{sj}^\top)$, where $\lambda_{\max}$ denotes the maximum eigenvalue, we have

$$-\mathrm{tr}(\frac{\gamma_{s2} \sum_{j=1}^{M} \tilde{\mathbf{W}}_s^\top \boldsymbol{\sigma}_{sj} \boldsymbol{\sigma}_{sj}^\top \tilde{\mathbf{W}}_s}{\|\boldsymbol{\Sigma}\| + \varepsilon_s}) \le -\gamma_{s2} \vartheta_s \|\tilde{\mathbf{W}}_s\|_F,$$

where $\vartheta_s > 0$ is defined as

$$\vartheta_s \triangleq 1 - \frac{(c_{\sigma_2} - c_{\sigma_1}) \|\tilde{\mathbf{W}}_s\|_F + M b_{\epsilon s} b_{\sigma s} + \varepsilon_s}{c_{\sigma_2} \|\tilde{\mathbf{W}}_s\|_F + M b_{\epsilon s} b_{\sigma s} + \varepsilon_s}.$$

Furthermore, assuming that ε_s is selected such that it satisfies $(\|\boldsymbol{\Sigma}\| + \varepsilon_s) > \sum_{j=1}^{M} \sup_{j=1 \cdots M} (\|\boldsymbol{\sigma}_{sj} \boldsymbol{\epsilon}_{sj}^\top\|_F)$, we have

$$-\mathrm{tr}(\frac{\gamma_{s2} \sum_{j=1}^{M} \tilde{\mathbf{W}}_s^\top \boldsymbol{\sigma}_{sj} \boldsymbol{\epsilon}_{sj}^\top}{\|\boldsymbol{\Sigma}\| + \varepsilon_s}) \le \gamma_{s2} \varpi_s \|\tilde{\mathbf{W}}_s\|_F$$

for some $\varpi_s < 1$ (note that $\|\boldsymbol{\Sigma}\| < \|\sum_{j=1}^{M} \boldsymbol{\sigma}_{sj} \boldsymbol{\epsilon}_{sj}^\top\|$ happens only if $\|\sum_{j=1}^{M} \boldsymbol{\sigma}_{sj} \boldsymbol{\sigma}_{sj}^\top \tilde{\mathbf{W}}_s\| < 2\|\sum_{j=1}^{M} \boldsymbol{\sigma}_{sj} \boldsymbol{\epsilon}_{sj}^\top\|$, which means for small values of $\tilde{\mathbf{W}}_s$). Therefore, considering Eq. (33), an upper bound of $\dot{L}_{W_s}$ can be written as

$$\dot{L}_{W_s} \le -\gamma_{s1} c_{\sigma_1} \|\tilde{\mathbf{W}}_s\|_F^2 + \epsilon_w \|\tilde{\mathbf{W}}_s\|_F, \tag{37}$$

where $\epsilon_w \triangleq (1 + \gamma_{s1} M) b_{\epsilon s} b_{\sigma s} - \gamma_{s2} (\vartheta_s - \varpi_s)$, and $\|\cdot\|_F$ is the Frobenius norm. Then, from Eqs. (36) and (37), we can conclude that $\tilde{\mathbf{W}}_s$ converges to a neighborhood of zero.

Remark 4 For large values of $\|\boldsymbol{\Sigma}\|$, since ε_s is bounded, the term $(\vartheta_s - \varpi_s)$ will be a positive value that results in a smaller ϵ_w, and thus, according to (37), a faster convergence. In addition, one can consider γ_{s2} a strictly increasing saturated function of $\|\boldsymbol{\Sigma}\|$ to reduce the effect of the last term in (34) for the small values of $\|\boldsymbol{\Sigma}\|$ (indicating the estimation error).

Remark 5 In contrast to the PE condition, the rank condition in Eq. (33) can be easily verified online. An algorithm for the selection of data points based on maximizing singular value was given previously [27]. The

history stack can be updated to improve the estimation performance by replacing old data with new data if they result in larger c_{σ_1}.

4 Optimal value function approximation

The solution of the optimal control problem, according to the closed-form equation of the optimal controller (Eq. 5), requires the optimal value function $V^*(\mathbf{x})$. Consequently, in this section, we present an approach to estimate this function, which can then be used to derive the estimated optimal controller. Following the standard RL techniques [4,13], the Bellman error δ_B can be employed as an indirect performance metric of the quality of the estimate of the value function. According to Eqs. (8) and (10), the Bellman error depends on the dynamics of the system. Therefore, using the estimate of the drift dynamics through the model identifiers proposed in the previous section, this section presents an online method to estimate the optimal value function using the Bellman error. Considering the function approximation property of NNs, optimal value function V^* can be represented as

$$V^*(\mathbf{x}) = \mathbf{W}_c^\top \boldsymbol{\sigma}_c(\mathbf{x}) + \epsilon_c(\mathbf{x}), \tag{38}$$

where $\mathbf{W}_c \in \mathbb{R}^{\mathcal{L}_c}$ is the constant ideal weight vector, $\boldsymbol{\sigma}_c \in \mathbb{R}^{\mathcal{L}_c}$ is the basis function or activation function vector satisfying $\boldsymbol{\sigma}_c(\mathbf{0}) = \mathbf{0}$ and $\nabla\boldsymbol{\sigma}(\mathbf{0}) \triangleq \partial\boldsymbol{\sigma}_c(\mathbf{0})/\partial\mathbf{x} = \mathbf{0}$, ϵ_c is the functional reconstruction error, and $\mathcal{L}_c$ is the number of neurons. Error ϵ_c and its gradient $\nabla\epsilon_c$ are bounded over the compact set Ω_N as [31]

$$|\epsilon_c| \leq b_{\epsilon c}, \quad \|\nabla\epsilon_c\| \leq b_{\epsilon cx}$$

for some positive $b_{\epsilon c}, b_{\epsilon cx} \in \mathbb{R}^+$, and the following bounds are considered for the activation function [5,8]:

$$\|\boldsymbol{\sigma}_c\| \leq b_{\sigma c}, \quad \|\nabla\boldsymbol{\sigma}_c\| \leq b_{\sigma cx},$$

where $b_{\sigma c}, b_{\sigma cx} \in \mathbb{R}^+$ are positive constants.

Using (38), the gradient of V^* can be written as

$$\nabla V^*(\mathbf{x}) = \mathbf{W}_c^\top \nabla\boldsymbol{\sigma}_c(\mathbf{x}) + \nabla\epsilon_c(\mathbf{x}). \tag{39}$$

Substituting Eq. (39) into Eq. (7), the HJB equation can be written as

$$Q(\mathbf{x}) + U(\mathbf{u}^*) + \mathbf{W}_c^\top \nabla\boldsymbol{\sigma}_c \left(\mathbf{f} + \mathbf{g}\mathbf{u}^*\right) = \epsilon_{hjb}, \tag{40}$$

where $\epsilon_{hjb} \triangleq -\nabla\epsilon_c(\mathbf{f}+\mathbf{g}\mathbf{u}^*)$, which is a bounded term [7,18].

Since ideal weight vector $\mathbf{W}_c$ is not available, the optimal value function is estimated through the following critic neural network:

$$\hat{V}(\mathbf{x}) = \hat{\mathbf{W}}_c^\top \boldsymbol{\sigma}_c(\mathbf{x}), \tag{41}$$

where $\hat{\mathbf{W}}_c \in \mathbb{R}^{\mathcal{L}_c}$ is the estimation of $\mathbf{W}_c$. Using Eq. (41), we obtain $\nabla\hat{V} = \hat{\mathbf{W}}_c^\top \nabla\boldsymbol{\sigma}_c$. Therefore, considering Eq. (5), the optimal controller (actor) can be estimated as

$$\hat{\mathbf{u}} = -\lambda \tanh(\hat{\mathbf{D}}), \tag{42}$$

where $\hat{\mathbf{D}} \triangleq \frac{1}{2\lambda}\mathbf{R}^{-1}\mathbf{g}^\top \nabla\boldsymbol{\sigma}_c^\top \hat{\mathbf{W}}_c$. Accordingly, the solution to the optimal control problem is converted to finding an adaptation rule for $\hat{\mathbf{W}}_c$ such that a proper estimation is guaranteed for the optimal value function.

In reinforcement learning, the online update law for $\hat{\mathbf{W}}_c$ is developed using the Bellman error. To guarantee that $\hat{\mathbf{W}}_c$ converges to a true value, sufficient exploration of the state space is required. In this paper, we follow a previous approach [11] that utilizes a system's model to simulate the Bellman error at any desired unexplored point. Since the estimate of drift dynamics $\mathbf{f}$ is available at any desired point $\mathbf{x}_j$ through $\hat{\mathbf{f}}_s$, the Bellman error can be evaluated at such points as follows:

$$\begin{aligned}\delta_{Bj} \triangleq\ & Q(\mathbf{x}_j) + U(\hat{\mathbf{u}}(\mathbf{x}_j, \hat{\mathbf{W}}_c)) \\ & + \nabla\hat{V}(\mathbf{x}_j)(\hat{\mathbf{f}}_s(\mathbf{x}_j) + \mathbf{g}\hat{\mathbf{u}}(\mathbf{x}_j, \hat{\mathbf{W}}_c)).\end{aligned}$$

Based on the above definition, the online update law for the critic weight vector is given as

$$\begin{aligned}\dot{\hat{\mathbf{W}}}_c \triangleq\ & -\frac{\alpha_c \boldsymbol{\beta}}{(1+\boldsymbol{\beta}^\top\boldsymbol{\beta})^2}\delta_B - \alpha_c\gamma_{c1}\boldsymbol{\Xi} \\ & -\alpha_c\gamma_{c2}\frac{\boldsymbol{\Xi}}{\|\boldsymbol{\Xi}\| + \varepsilon_c},\end{aligned} \tag{43}$$

where $\alpha_c, \gamma_{c1}, \gamma_{c2}, \varepsilon_c \in \mathbb{R}^+$ are positive design gains, $\boldsymbol{\Xi} \in \mathbb{R}^{\mathcal{L}_c}$ is defined as

$$\boldsymbol{\Xi} \triangleq \sum_{j=1}^{N} \frac{\boldsymbol{\beta}_j}{N(1+\boldsymbol{\beta}_j^\top\boldsymbol{\beta}_j)^2}\delta_{Bj},$$

and vector variables $\boldsymbol{\beta}, \boldsymbol{\beta}_j \in \mathbb{R}^{\mathcal{L}_c}$ are defined as

$$\begin{aligned}\boldsymbol{\beta} &\triangleq \nabla\boldsymbol{\sigma}_c(\hat{\mathbf{f}} - \lambda\mathbf{g}\tanh(\kappa\hat{\mathbf{D}})), \\ \boldsymbol{\beta}_j &\triangleq \nabla\boldsymbol{\sigma}_c(\mathbf{x}_j)(\hat{\mathbf{f}}_s(\mathbf{x}_j) - \lambda\mathbf{g}(\mathbf{x}_j)\tanh(\kappa\hat{\mathbf{D}}(\mathbf{x}_j))),\end{aligned} \tag{44}$$

where $\kappa \in \mathbb{R}^+$ is a positive constant. The last term in Eq. (43) resembles the last component in Eq. (34),

which is a bounded control term that increases the convergence rate without employing a high-gain estimation law (see *Remark* 3 and *Remark* 4). Vector $\boldsymbol{\beta}_j$ is obtained from the simulated experience at points $\mathbf{x}_j$ for $j = 1, \cdots, N$, satisfying the following condition.

Assumption 3 There exists a finite set of points $\mathbf{x}_j$ for $j = 1, \cdots, N$, such that

$$0 < c_{\beta_1} \triangleq \frac{1}{N}\left(\inf_{t\in\mathbb{R}_{\geq t_0}}\left(\lambda_{\min}\left(\sum_{j=1}^{N}\frac{\boldsymbol{\beta}_j\boldsymbol{\beta}_j^\top}{(1+\boldsymbol{\beta}_j^\top\boldsymbol{\beta}_j)^2}\right)\right)\right). \tag{45}$$

Assumption 3 can be satisfied when $\text{rank}(\sum_{j=1}^{N}\boldsymbol{\beta}_j\boldsymbol{\beta}_j^\top) = \mathcal{L}_c$, and it can be easily verified online. Since the vectors $\boldsymbol{\beta}_j$ are obtained from unvisited points, a sufficient number of points can be selected to fulfill the condition (45) [10].

Now, in the following theorem, we present the stability and convergence results of the given online policy iteration algorithm.

Theorem 2 *For the system of Eq. (1), consider the controller in Eq. (42), the critic weight update law in Eq. (43), and the model identifier* $\hat{\mathbf{f}}_s$ *with its weight update law given in Eq. (34). Provided Assumptions (1)-(3), and the following sufficient gain conditions are satisfied:*

$$\gamma_{s1}c_{\sigma_1} > \frac{\eta_1^2\varrho_1}{4} + \frac{\eta_3^2}{4}, \qquad \gamma_{c1}c_{\beta_1} > \frac{\varrho_1}{\eta_1^2} + \frac{\eta_3^2}{4}, \tag{46}$$

where b_β, ϱ_1, η_1, η_2, *and* η_3 *are positive constants defined subsequently in the proof, then system state* $\mathbf{x}$ *and weight estimation errors* $\tilde{\mathbf{W}}_c$ *and* $\tilde{\mathbf{W}}_s$ *are UUB, which means UUB convergence of* $\hat{\mathbf{u}}$ *to* $\mathbf{u}^*$.

Proof See the proof in Appendix C. □

Remark 6 Due to the existence of the fast model identifier, a similar stability analysis as presented in Appendix C can be made to show that the closed-loop system remains stable before the condition of Eq. (45) is satisfied; however, the convergence of $\hat{\mathbf{W}}_c$ to the optimal weight cannot be guaranteed.

Remark 7 Although we addressed the effect of the estimation error of the slow model identifier as an unbounded term in the stability analysis, it can be assumed to be *a priori* bounded provided we use only the identifier's outputs when the estimation error is smaller than a certain predefined value.

5 Simulation results

In this section, we present the results of simulation studies that evaluated the performance of our proposed control scheme. Since there are no known solutions to optimal control problems for bounded-input nonlinear systems, in the first two studies, to show that our proposed method converges to optimal solutions, we selected an actuator bound that is large enough to verify that the commands do not violate the bound. Input constraints were considered in the second study, where we evaluated the effectiveness of the proposed algorithm and compared the results with those of two available solutions in the literature for a bounded-input system.

5.1 Systems without actuator saturation

In this simulation study, the model of the nonlinear system of Eq. (1) was considered to be [7]

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} -x_1 + x_2 \\ -0.5x_1 - 0.5x_2\left(1 - (\cos(2x_1) + 2)^2\right) \end{bmatrix},$$
$$\mathbf{g}(\mathbf{x}) = \begin{bmatrix} 0 \\ \cos(2x_1) + 2 \end{bmatrix},$$

which satisfies Assumption 1. The state vector $\mathbf{x} = [x_1,\ x_2]^\top$ was initialized at $\mathbf{x}(0) = [-1,\ -1]^\top$. The design parameters of the fast model identifier (21) were selected as $k_f = 8$, $\alpha = 1$, $\beta_1 = 0.2$, $\kappa_\mu = 2$, $\mu_0 = 4$, $\mu_\infty = 0.3$, and $\bar{\delta}_i = 1$, and sigmoid functions were considered for the basis of the NN with $\mathcal{L}_f = 6$. For the slow model identifier, the design parameters were selected as $\gamma_{s1} = \gamma_{s2} = 3$, $\varepsilon_s = 0.001$, $\boldsymbol{\Gamma}_s(0) = 3\mathbf{I}_{\mathcal{L}_s}$, and $\rho_s = 0.1$, and the basis was considered to be

$$\begin{aligned}\boldsymbol{\sigma}_s = [&x_1,\ x_2,\ \sin(x_1),\ \cos(x_1),\ \sin(x_2),\ \cos(x_2),\\ &\sin(x_1)^2,\ \cos(x_1)^2,\ \sin(x_2)^2,\ \cos(x_2)^2,\ x_1\cos(x_2),\\ &x_1\cos(x_2)^2,\ x_2\cos(x_1),\ x_2\cos(x_1^2)]^\top.\end{aligned}$$

Here, all of the initial weights were set to 0.5. We used an algorithm from a previous study [27] to record the data in the history stack for this identifier, and Assumption 2 was satisfied around $t = 1.2$ s. The optimal control problem was defined by considering $Q(\mathbf{x}) = \mathbf{x}^\top\mathbf{x}$ and $R = 1$. The design parameters were selected as $\alpha_c = 200$, $\gamma_{c1} = \gamma_{c2} = 1.5$, $\varepsilon_c = 0.0005$, $\lambda = 3$, and $\kappa = 3$. The basis was selected as $\boldsymbol{\sigma}_c =$

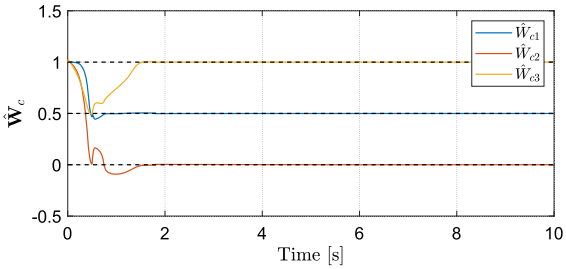

Fig. 2 Estimated value function weights

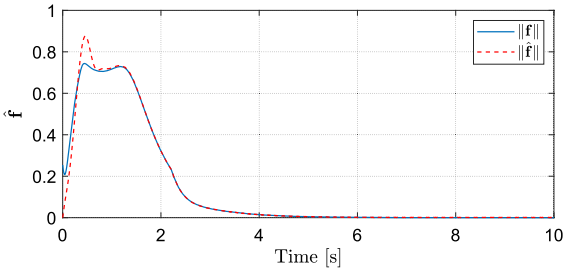

Fig. 3 Norm of unknown function $\mathbf{f}$ estimated by fast identifier

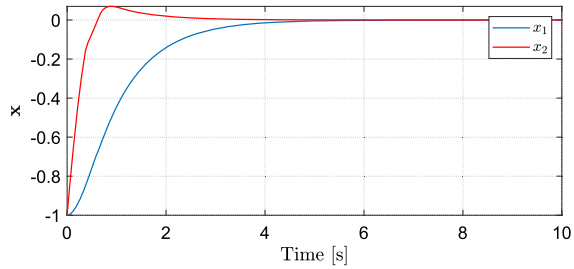

Fig. 4 State trajectory during online learning

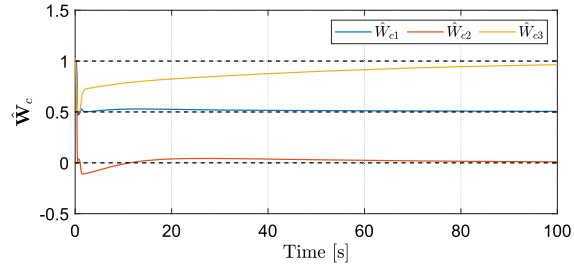

Fig. 5 Time evolution of estimated value function weights without using last term in Eq. (43)

$[x_1^2,\ x_1x_2,\ x_2^2]^\top$, the initial weights were set to 1, and the data for the simulation of the Bellman error were selected from a 5×5 grid around the trajectory such that Assumption 3 was satisfied.

Based on an analytical solution, the ideal weights are $\mathbf{W}_c = [0.5,\ 0,\ 1]^\top$ [7]. The estimated values in this simulation converged to $\hat{\mathbf{W}}_c = [0.499,\ 0.00,\ 1.00]^\top$ after 10 s, which shows the effectiveness of the proposed algorithm. The trajectories of the estimated weights are shown in Fig. 2. The trajectories of the norm of $\hat{\mathbf{f}}$ estimated by the fast model identifier and its true value are shown in Fig. 3. The trajectories of the system's states are depicted in Fig. 4. In comparison to results reported in the literature for the same nonlinear system, e.g., [7], our results show faster convergence of the estimated value function and smoother trajectories of the states. This is because these approaches require the application of an exploration input signal and forcing the system to visit many points in the state space to satisfy the PE condition; however, our approach, similar to an earlier study [11], simulates those points and gains smoother real trajectories. Also note that the design of that study [11] required an additional estimation network for the actor.

To demonstrate the effectiveness of the last term of the update law (Eq. 43) in increasing the convergence rate, the above simulation was repeated by removing this term from the update law. The estimated weight vector after 10 s was $\hat{\mathbf{W}}_c = [0.526,\ -0.011,\ 0.782]^\top$, and it took about 100 s until the weights nearly converged to the optimal values. The time evolution of the estimated weights in this condition is shown in Fig. 5. We observed a similar level of effectiveness for the last term of the adaptation law (Eq. 34) used for the model identification.

In another study, we considered a four-order linear dynamic system $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$ that describes a mechanical system consisting of a mass attached to a spring and a damper [32]. Spring constants, damping coefficients, and the mass of the system are selected to obtain the following results in the state and input matrices:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -6 & -1.7 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -3 & -0.2 \end{bmatrix}, \qquad \mathbf{B} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}. \tag{47}$$

The state vector was initialized at $\mathbf{x}(0) = [-1,\ -1,\ -1,\ -1]^\top$. The design parameters for the fast model identifier were selected to be the same as those used in the previous simulation. The state vector was considered to be the basis for the slow model identifier. The parameters of this identifier were selected as $\gamma_{s1} = \gamma_{s2} = 1$, and all other parameters and initial values of the weights were the same as those in the previous simulation. The design

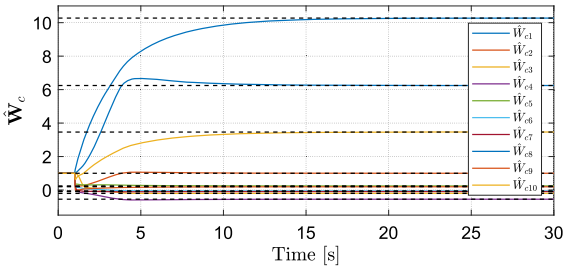

Fig. 6 Time evolution of estimated value function weights for the linear system described by Eq. (47)

parameters for the estimation of the value function were selected as $\alpha_c = 2$, $\gamma_{c1} = 3$, $\gamma_{c2} = 1$, $\varepsilon_c = 0.001$, $\lambda = 10$, and $\kappa = 3$. The basis vector was $\boldsymbol{\sigma}_c = [x_1^2,\ 2x_1x_2,\ 2x_1x_3,\ 2x_1x_4,\ 2x_2^2,\ 2x_2x_3,\ 2x_2x_4,\ 2x_3^2,\ 2x_3x_4,\ 2x_4^2]^\top$. The initial weights were set to 1, and the data for the simulation of the Bellman error were selected from a $5 \times 5 \times 5 \times 5$ grid around the trajectory.

For the cost function defined by $\mathbf{Q}(\mathbf{x}) = \mathbf{x}^\top\mathbf{x}$ and $U(\mathbf{u}) = u^2$, by solving the algebraic Riccati equation, the ideal weight vector was obtained as $\mathbf{W}_c = [6.245,\ 1.000,\ -0.203,\ -0.548,\ 0.245,\ -0.054,\ -0.093,\ 10.274,\ 0.196,\ 3.462]^\top$, and the estimated one after 30 s was $\hat{\mathbf{W}}_c = [6.245,\ 0.999,\ -0.194,\ -0.548,\ 0.246,\ -0.051,\ -0.101,\ 10.274,\ 0.197,\ 3.461]^\top$. These results confirm the convergence of the proposed method to the optimal control solution. The time evolution of the weights is shown in Fig. 6.

5.2 Nonlinear system with actuator saturation

The nonlinear system considered in this study was defined by the following terms [13,18]:

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} x_1 + x_2 - x_1(x_1^2 + x_2^2) \\ -x_1 + x_2 - x_2(x_1^2 + x_2^2) \end{bmatrix}, \qquad \mathbf{g}(\mathbf{x}) = \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$

where the state vector was initialized at $\mathbf{x}(0) = [1,\ -1]^\top$. The control input was assumed to be limited to $|u| \leq 1$, and the nonquadratic cost function (Eq. 2) was defined by considering $Q(\mathbf{x}) = \mathbf{x}^\top\mathbf{x}$ and $R = 1$. The design parameters of the fast model identifier were identical to those of the previous simulation example. For the slow model identifier, the design parameters were selected as $\gamma_{s1} = \gamma_{s2} = 1$, $\varepsilon_s = 0.1$, $\boldsymbol{\Gamma}_s(0) = 8\mathbf{I}_{\mathcal{L}_s}$, and $\rho_s = 0.5$, and the basis was considered to be $\boldsymbol{\sigma}_s = [x_1,\ x_2,\ x_1x_1^2,\ x_1x_2^2,\ x_2x_1^2,\ x_2x_2^2]^\top$. Again, all of the initial weights were set to 0.5. Here, $\alpha_c = 35$, $\gamma_{c1} = \gamma_{c2} = 4.5$, $\varepsilon_c = 0.001$, and $\lambda = 1$ were selected as design parameters for the value function estimation. The data for the simulation of the Bellman error were selected from a 5×5 grid around the trajectory, and the basis was selected as [18]

$$\begin{aligned}\boldsymbol{\sigma}_c =&[x_1^2,\ x_2^2,\ x_1x_2,\ x_1^4,\ x_2^4,\ x_1^3x_2,\ x_1^2x_2^2,\ x_1x_2^3,\ x_1^6,\\ &x_2^6,\ x_1^5x_2,\ x_1^4x_2^2,\ x_1^3x_2^3,\ x_1^2x_2^4,\ x_1x_2^5,\ x_1^8,\ x_2^8,\\ &x_1^7x_2,\ x_1^6x_2^2,\ x_1^5x_2^3,\ x_1^4x_2^4,\ x_1^3x_2^5,\ x_1^2x_2^6,\ x_1x_2^7]^\top,\end{aligned}$$

as were the initial critic weights [18]

$$\begin{aligned}\hat{\mathbf{W}}_c(0) =&[2.92,\ 1.84,\ 0.96,\ 1.96,\ 1.39,\ -1.07,\ -1.42,\\ &-1.00,\ -1.57,\ -0.88,\ 1.14,\ 0.76,\ 3.69,\ 3.02,\\ &2.66,\ -0.35,\ 1.22,\ -1.30,\ -3.35,\ 0.12,\ 2.12,\\ &-0.01,\ 2.49,\ 2.13]^\top.\end{aligned}$$

The time evolution of the estimation of weights is shown in Fig. 7. According to the estimated values and (42), the estimated controller after 50 s is given by

$$\begin{aligned}\hat{\mathbf{u}} =& -\tanh(4.5x_1 + 3.95x_2 + 12.5x_2^3 - 2.15x_1^3\\ &+ 9.7x_1^2x_2 + 17.11x_1x_2^2 + 0.69x_2^5 + 0.31x1^5 + 1.1x_1^4x_2\\ &+ 6.69x_1^3x_2^2 + 7.86x_1^2x_2^3 + 9.07x_1x_2^4 + 6.25x_2^7 - 0.67x_1^7\\ &- 3.35x_1^6x_2 + 0.24x_1^5x_2^2 + 4.3x_1^4x_2^3 + 0.02x_1^3x_2^4\\ &+ 7.92x_1^2x_2^5 + 1.06x_1x_2^6).\end{aligned}$$

The trajectories of the states obtained by this estimated controller are shown in Fig. 8. The figure also shows the trajectories of the states obtained by the estimated controllers given in previous works [18] (obtained after 250 s) and [13] (obtained offline). The evolution of the control efforts obtained by the three methods is shown in Fig. 9. The cost values measured within 20 s were 2.77 for our method, while those for the previous methods were 2.84 [18] and 5.46 [13]. These results demonstrate that the proposed approach yields a superior performance for an estimated optimal controller. Notably, our method achieves this result without using an exploratory signal in contrast to the earlier works [18], [13].

6 Conclusion

This study addressed online RL-based solutions to the optimal regulation problem of unknown continuous-

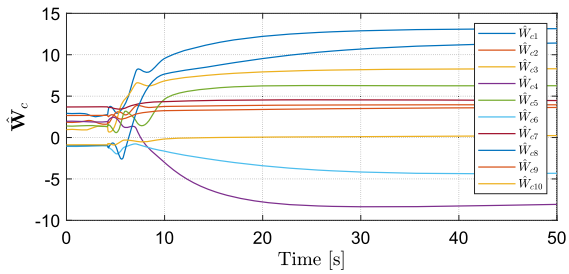

Fig. 7 Evolution of a small number of estimated value function weights

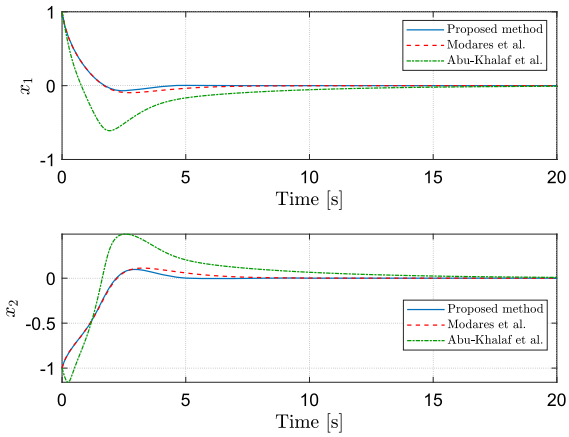

Fig. 8 State trajectories obtained by estimated optimal controllers

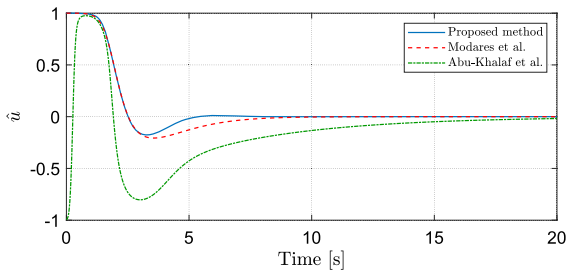

Fig. 9 Evolution of control efforts obtained by three methods

time nonlinear systems. The designed method uses the simulated experience concept and incorporates actuator bounds in its control design. Two model identifiers were developed whose outputs were used to guarantee the convergence of the estimated controller to the optimal one and to keep the system stable while learning the optimal solution. The proposed estimation law for the critic network satisfies fast convergence without employing a large estimation gain. A stability analysis shows the controller's UUB convergence to the optimal one, and the simulation results demonstrate the effectiveness of the developed technique.

Declarations

- This work is supported by the Innovative Science and Technology Initiative for Security, Grant Number JPJ004596, and ATLA, Japan. The study is partially based on results obtained from project JPNP20006, commissioned by the New Energy and Industrial Technology Development Organization (NEDO). This work was partially supported by JSPS KAKENHI Grant Number JP21H03527.
- The datasets generated and/or analyzed during the current study are available from the corresponding author upon reasonable request.
- The authors declare that there is no conflict of interest to disclose.
- All authors contributed to the study's conception and design.

Funding This work is based on results obtained from project JPNP20006, commissioned by the New Energy and Industrial Technology Development Organization (NEDO). This work was partially supported by JSPS KAKENHI Grant Number JP21H03527.

Data availability Enquiries about data availability should be directed to the authors.

Declarations

Conflicts of interest The authors have not disclosed any competing interests.

Appendices

A Proof of *Theorem* 1

Consider $\mathcal{D} \subseteq \mathbb{R}^{2n+2}$ a domain containing $\mathbf{y} = 0$, where $\mathbf{y} \in \mathbb{R}^{2n+2}$ is defined as $\mathbf{y} \triangleq [\boldsymbol{\omega}^\top \sqrt{P} \sqrt{Q_f}]^\top$, in which $Q_f(t) \in \mathbb{R}^+$ is defined as $Q_f(t) \triangleq \frac{\alpha}{2\gamma}\mathrm{tr}(\tilde{\mathbf{W}}_f^\top \tilde{\mathbf{W}}_f)$ with $\mathrm{tr}(\cdot)$ denoting the trace of a matrix. Here, function $P(t) \in \mathbb{R}$ is given by

$$P(t) \triangleq \beta_1 \sum_{i=1}^{n} |\varsigma_i(0)| - \boldsymbol{\varsigma}(0)^\top (\dot{\boldsymbol{\epsilon}}_f(0) + \mathbf{N}_B(0)) - \int_0^t \mathcal{K}(\bar{t})\mathrm{d}\bar{t}, \tag{48}$$

where function $\mathcal{K}(t) \in \mathbb{R}$ is defined as

$$\mathcal{K}(t) \triangleq \mathbf{e}_\varsigma^\top (\dot{\boldsymbol{\epsilon}}_0 - \beta_1 \mathrm{sgn}(\boldsymbol{\varsigma})) + \dot{\boldsymbol{\varsigma}}^\top \mathbf{N}_B - \beta_2 \|\boldsymbol{\varsigma}\|^2, \quad (49)$$

in which $\beta_2 \in \mathbb{R}^+$ is a positive constant. Provided the conditions of Eqs. (30) and (31) are satisfied, it can be shown that $P(t) \geq 0$; see the proof in Appendix B. Now consider the continuously differentiable positive-definite function as follows:

$$\mathcal{V} = \frac{1}{2}\boldsymbol{\varsigma}^\top \boldsymbol{\varsigma} + \frac{1}{2}\mathbf{e}_\varsigma^\top \mathbf{e}_\varsigma + P + Q_f. \quad (50)$$

It can be concluded that

$$\psi_1(\mathbf{y}) \leq \mathcal{V}(\mathbf{y}) \leq \psi_2(\mathbf{y}), \quad (51)$$

where positive-definite strictly increasing functions $\psi_1, \psi_2 \in \mathbb{R}^+$ are defined as $\psi_1 \triangleq 0.5\|\mathbf{y}\|^2$ and $\psi_2 \triangleq \|\mathbf{y}\|^2$. Using Eqs. (24), (25), and the time derivative of Eq. (48), the time derivative of $\mathcal{V}$ can be developed as

$$\begin{aligned}\dot{\mathcal{V}} = &-(\alpha - \beta_2)\boldsymbol{\varsigma}^\top \boldsymbol{\varsigma} - k_f \mathbf{e}_\varsigma^\top \mathbf{e}_\varsigma + \mathbf{e}_\varsigma^\top \tilde{\mathbf{N}} + \mathbf{e}_\varsigma^\top \mathbf{N}_B \\ &- \dot{\boldsymbol{\varsigma}}^\top \mathbf{N}_B - \frac{\alpha}{\gamma}\mathrm{tr}(\tilde{\mathbf{W}}_f^\top \dot{\hat{\mathbf{W}}}_f).\end{aligned}$$

Therefore, using Eq. (24) and the definition of $\mathbf{N}_B$, and knowing that $\mathbf{a}^\top \mathbf{b} = \mathrm{tr}(\mathbf{b}\mathbf{a}^\top), \forall \mathbf{a}, \mathbf{b} \in \mathbb{R}^n$, we can write

$$\begin{aligned}\dot{\mathcal{V}} = &-(\alpha - \beta_2)\boldsymbol{\varsigma}^\top \boldsymbol{\varsigma} - k_f \mathbf{e}_\varsigma^\top \mathbf{e}_\varsigma + \mathbf{e}_\varsigma^\top \tilde{\mathbf{N}} \\ &- \frac{\alpha}{\gamma}\mathrm{tr}(\tilde{\mathbf{W}}_f^\top \dot{\hat{\mathbf{W}}}_f - \gamma \tilde{\mathbf{W}}_f^\top \dot{\boldsymbol{\sigma}}_f \boldsymbol{\varsigma}^\top \boldsymbol{\Upsilon}_0).\end{aligned}$$

Considering the adaptation law (Eq. 29) and using the properties of the projection operator, we have

$$-\frac{\alpha}{\gamma}\mathrm{tr}(\tilde{\mathbf{W}}_f^\top \dot{\hat{\mathbf{W}}}_f - \gamma \tilde{\mathbf{W}}_f^\top \dot{\boldsymbol{\sigma}}_f \boldsymbol{\varsigma}^\top \boldsymbol{\Upsilon}_0) \leq 0. \quad (52)$$

Therefore, using Eqs. (26) and (52), an upper bound for $\dot{\mathcal{V}}$ can be written as

$$\dot{\mathcal{V}} \leq -(\alpha - \beta_2)\|\boldsymbol{\varsigma}\|^2 - k_f \|\mathbf{e}_\varsigma\|^2 + \rho(\|\boldsymbol{\omega}\|)\|\boldsymbol{\omega}\|\|\mathbf{e}_\varsigma\|.$$

By splitting k_f into adjustable positive gains $k_{f1}, k_{f2} \in \mathbb{R}^+$ as $k_f = k_{f1} + k_{f2}$, we can further bound $\dot{\mathcal{V}}$ as follows if the condition in Eq. (31) is satisfied:

$$\dot{\mathcal{V}} \leq -\beta_3 \|\boldsymbol{\omega}\|^2 - k_{f2}\|\mathbf{e}_\varsigma\|^2 + \rho(\|\boldsymbol{\omega}\|)\|\boldsymbol{\omega}\|\|\mathbf{e}_\varsigma\|, \quad (53)$$

where β_3 is defined as $\beta_3 \triangleq \min\{(\alpha - \beta_2), k_{f1}\}$. Completing the squares for the last two terms of Eq. (53), we obtain $\dot{\mathcal{V}} \leq -(\beta_3 - \rho^2(\|\boldsymbol{\omega}\|)/4k_{f2})\|\boldsymbol{\omega}\|^2$. Therefore, we have $\dot{\mathcal{V}} \leq c\|\boldsymbol{\omega}\|^2$ for some positive constant $c \in \mathbb{R}^+$ in the following domain:

$$\mathcal{D} \triangleq \left\{ \mathbf{y} \in \mathbb{R}^{2n+2} \mid \|\mathbf{y}\| \leq \rho^{-1}(2\sqrt{\beta_3 k_{f2}}) \right\},$$

which, considering Eq. (51), indicates that $\mathcal{V} \in \mathcal{L}_\infty$ in $\mathcal{D}$. Therefore, from Eq. (50), we have $\boldsymbol{\varsigma}$, $\mathbf{e}_\varsigma$, P, Q_f, and hence $\tilde{\mathbf{W}}_f \in \mathcal{L}_\infty$ in $\mathcal{D}$. Since $\boldsymbol{\varsigma}$ is bounded, we can conclude that the prescribed performance condition is satisfied and $\tilde{\mathbf{x}} \in \mathcal{L}_\infty$ in $\mathcal{D}$. To analyze the convergence of the signals, we need to show that $\boldsymbol{\omega}$ is uniformly bounded. From Eq. (24), we can see that $\dot{\boldsymbol{\varsigma}} \in \mathcal{L}_\infty$ in $\mathcal{D}$. Therefore, since $\boldsymbol{\Upsilon}$ is bounded, from Eq. (19) we have $\dot{\tilde{\mathbf{x}}} \in \mathcal{L}_\infty$ in $\mathcal{D}$. Consequently, from Eq. (22), we conclude that $\boldsymbol{\nu} \in \mathcal{L}_\infty$ in $\mathcal{D}$. Since $\dot{\tilde{\mathbf{x}}}$ is bounded, from the definition of $\boldsymbol{\Upsilon}$, we also see that $\dot{\boldsymbol{\Upsilon}} \in \mathcal{L}_\infty$ in $\mathcal{D}$. From Eq. (29), $\dot{\hat{\mathbf{W}}}_f \in \mathcal{L}_\infty$ in $\mathcal{D}$, and we also have $\dot{\hat{\boldsymbol{\sigma}}}_f \in \mathcal{L}_\infty$. From these results and Eq. (25), we can see that $\dot{\mathbf{e}}_\varsigma \in \mathcal{L}_\infty$ in $\mathcal{D}$. Then, since $\dot{\boldsymbol{\varsigma}}, \dot{\mathbf{e}}_\varsigma \in \mathcal{L}_\infty$, we conclude that $\dot{\boldsymbol{\omega}} \in \mathcal{L}_\infty$ in $\mathcal{D}$, which indicates that $\boldsymbol{\omega}$ is uniformly continuous in $\mathcal{D}$.

Now consider region $\mathcal{S} \subset \mathcal{D}$ as $\mathcal{S} \triangleq \{\mathbf{y} \subset \mathcal{D} \mid \psi_2(\mathbf{y}) < \frac{1}{2}(\rho^{-1}(2\sqrt{\beta_3 k_{f2}}))^2\}$. Based on the above results, Theorem 8.4 of an earlier work [33] can be used to conclude that $\|\boldsymbol{\omega}\| \to 0$ as time goes to infinity for all $\mathbf{y}(0) \in \mathcal{S}$. According to Eq. (17), $T_i \to 0$ as $\varsigma_i \to 0$, and from Eq. (14), $\|\tilde{\mathbf{x}}\| \to 0$. Therefore, from Eqs. (19) and (24), we conclude that $\|\dot{\tilde{\mathbf{x}}}\| \to 0$ as $t \to 0$. The convergence of $\dot{\tilde{\mathbf{x}}}$ to zero indicates that $\hat{\mathbf{f}}$, given by Eq. (21), converges to $\mathbf{f}$.

B Proof of $P(t) \geq 0$

Here we show that $P(t) \geq 0$. The proof follows the same steps as in a prior study [34]. Integrating both sides of Eq. (49), we have

$$\begin{aligned}\int_0^t \mathcal{K}(\bar{t})\mathrm{d}\bar{t} = &\int_0^t \mathbf{e}_\varsigma^\top (\dot{\boldsymbol{\epsilon}}_0 - \beta_1 \mathrm{sgn}(\boldsymbol{\varsigma}))\,\mathrm{d}\bar{t} + \int_0^t \dot{\boldsymbol{\varsigma}}^\top \mathbf{N}_B \mathrm{d}\bar{t} \\ &- \int_0^t \beta_2 \|\boldsymbol{\varsigma}\|^2 \mathrm{d}\bar{t}.\end{aligned}$$

Using Eq. (24), we have

$$\begin{aligned}\int_0^t \mathcal{K}(\bar{t})\mathrm{d}\bar{t} = &\int_0^t \frac{\mathrm{d}\boldsymbol{\varsigma}^\top}{\mathrm{d}\bar{t}}(\dot{\boldsymbol{\epsilon}}_0 + \mathbf{N}_B)\mathrm{d}\bar{t} - \int_0^t \frac{\mathrm{d}\boldsymbol{\varsigma}^\top}{\mathrm{d}\bar{t}}\beta_1 \mathrm{sgn}(\boldsymbol{\varsigma})\mathrm{d}\bar{t} \\ &+ \int_0^t \alpha \boldsymbol{\varsigma}^\top (\dot{\boldsymbol{\epsilon}}_0 - \beta_1 \mathrm{sgn}(\boldsymbol{\varsigma}))\,\mathrm{d}\bar{t} - \int_0^t \beta_2 \|\boldsymbol{\varsigma}\|^2 \mathrm{d}\bar{t}.\end{aligned} \quad (54)$$

Integrating the first integral in Eq. (54) by parts yields

$$\int_0^t \mathcal{K}(\bar{t})\mathrm{d}\bar{t} = \boldsymbol{\varsigma}^\top (\dot{\boldsymbol{\epsilon}}_0 + \mathbf{N}_B)\Big|_0^t - \int_0^t \boldsymbol{\varsigma}^\top \frac{\mathrm{d}(\dot{\boldsymbol{\epsilon}}_0 + \mathbf{N}_B)}{\mathrm{d}\bar{t}}\mathrm{d}\bar{t}$$

$$
\begin{aligned}
&- \int_0^t \frac{\mathrm{d}\boldsymbol{\varsigma}^\top}{\mathrm{d}\bar{t}} \beta_1 \mathrm{sgn}(\boldsymbol{\varsigma}) \mathrm{d}\bar{t} - \int_0^t \beta_2 \|\boldsymbol{\varsigma}\|^2 \mathrm{d}\bar{t} \\
&+ \int_0^t \alpha \boldsymbol{\varsigma}^\top \left(\dot{\boldsymbol{\epsilon}}_0 - \beta_1 \mathrm{sgn}(\boldsymbol{\varsigma})\right) \mathrm{d}\bar{t} \\
&= \boldsymbol{\varsigma}^\top (\dot{\boldsymbol{\epsilon}}_0 + \mathbf{N}_B) - \boldsymbol{\varsigma}(0)^\top (\dot{\boldsymbol{\epsilon}}_0(0) + \mathbf{N}_B(0)) \\
&+ \int_0^t \alpha \boldsymbol{\varsigma}^\top \left(\dot{\boldsymbol{\epsilon}}_0 - \frac{1}{\alpha} \frac{\mathrm{d}(\dot{\boldsymbol{\epsilon}}_0 + \mathbf{N}_B)}{\mathrm{d}\bar{t}} - \beta_1 \mathrm{sgn}(\boldsymbol{\varsigma}) \right) \mathrm{d}\bar{t} \\
&- \beta_1 \sum_{i=1}^n |\varsigma_i| + \beta_1 \sum_{i=1}^n |\varsigma_i(0)| - \int_0^t \beta_2 \|\boldsymbol{\varsigma}\|^2 \mathrm{d}\bar{t}.
\end{aligned}
$$

Knowing that $\sum_{i=1}^n |\varsigma_i| \geq \|\boldsymbol{\varsigma}\|$, and using Eqs. (27) and (28), we can write the following inequality:

$$
\begin{aligned}
\int_0^t \mathcal{K}(\bar{t}) \mathrm{d}\bar{t} &\leq (\zeta_1 + \zeta_3 - \beta_1) \|\boldsymbol{\varsigma}\| + \beta_1 \sum_{i=1}^n |\varsigma_i(0)| \\
&+ \int_0^t \alpha (\zeta_1 + \frac{\zeta_2 + \zeta_4}{\alpha} - \beta_1) \|\boldsymbol{\varsigma}\| \mathrm{d}\bar{t} \\
&+ \int_0^t (\zeta_5 - \beta_2) \|\boldsymbol{\varsigma}\|^2 \mathrm{d}\bar{t} - \boldsymbol{\varsigma}(0)^\top (\dot{\boldsymbol{\epsilon}}_0(0) + \mathbf{N}_B(0)).
\end{aligned}
$$

Therefore, if the conditions in Eqs. (30) and (31) are satisfied, we have

$$
\int_0^t \mathcal{K}(\bar{t}) \mathrm{d}\bar{t} \leq \beta_1 \sum_{i=1}^n |\varsigma_i(0)| - \boldsymbol{\varsigma}(0)^\top (\dot{\boldsymbol{\epsilon}}_0(0) + \mathbf{N}_B(0)),
$$

which indicates that $P(t) \geq 0$.

C Proof of *Theorem 2*

Consider the following Lyapunov function:

$$
L \triangleq V^*(\mathbf{x}) + L_{W_c} + L_{W_s}, \tag{55}
$$

where L_{W_s} is defined in Eq. (36), and $L_{W_c} \triangleq \frac{1}{2\alpha_c} \tilde{\mathbf{W}}_c^\top \tilde{\mathbf{W}}_c$, in which $\tilde{\mathbf{W}}_c \triangleq \mathbf{W}_c - \hat{\mathbf{W}}_c$. Using Eqs. (1), (5), and (42), we have

$$
\begin{aligned}
\dot{V}^* &= \frac{\partial V^*}{\partial \mathbf{x}} \dot{\mathbf{x}} = \nabla V^* (\mathbf{f} + \mathbf{g}\hat{\mathbf{u}}) \\
&= \nabla V^* (\mathbf{f} + \mathbf{g}\mathbf{u}^*) + \lambda \nabla V^* \mathbf{g} (\tanh(\mathbf{D}^*) - \tanh(\hat{\mathbf{D}})).
\end{aligned}
$$

Therefore, using Eqs. (7) and (38), $\dot{V}^*$ can be written as

$$
\begin{aligned}
\dot{V}^* = &- Q(\mathbf{x}) - U(\mathbf{u}^*) \\
&+ \lambda (\mathbf{W}_c^\top \nabla \boldsymbol{\sigma}_c + \nabla \epsilon_c^\top) \mathbf{g} (\tanh(\mathbf{D}^*) - \tanh(\hat{\mathbf{D}})).
\end{aligned} \tag{56}
$$

Defining $\tilde{\mathbf{u}} \triangleq \tanh(\mathbf{D}^*) - \tanh(\hat{\mathbf{D}})$, and knowing that

$$
\begin{aligned}
&U(\mathbf{u}^*) > 0, \\
&\lambda \nabla \epsilon_c^\top \mathbf{g} \tilde{\mathbf{u}} \leq \lambda \|\nabla \epsilon_c\| \|\mathbf{g}\| \|\tilde{\mathbf{u}}\| \leq 2\lambda b_{\epsilon c x} b_g, \\
&\lambda \mathbf{W}_c^\top \nabla \boldsymbol{\sigma}_c \mathbf{g} \tilde{\mathbf{u}} \leq \lambda \|\mathbf{W}_c\| \|\nabla \boldsymbol{\sigma}_c\| \|\mathbf{g}\| \|\tilde{\mathbf{u}}\| \\
&\qquad\qquad \leq 2\lambda \|\mathbf{W}_c\| b_{\sigma c x} b_g,
\end{aligned}
$$

and $Q(\mathbf{x}) > q_{\min} \mathbf{x}^\top \mathbf{x}$ for some $q_{\min} \in \mathbb{R}^+$, an upper bound can be written for $\dot{V}^*$ as

$$
\dot{V}^* \leq -q_{\min} \|\mathbf{x}\|^2 + \kappa_1, \tag{57}
$$

where $\kappa_1 \triangleq 2\lambda \|\mathbf{W}_c\| b_{\sigma c x} b_g + 2\lambda b_{\epsilon c x} b_g$.

To develop the time derivative of the second term of the right-hand side of Eq. (55), we first write the Bellman error (Eq. 10) as follows by substituting the gradient of Eq. (41) into Eq. (8):

$$
\delta_B = Q(\mathbf{x}) + U(\hat{\mathbf{u}}) + \hat{\mathbf{W}}_c^\top \nabla \boldsymbol{\sigma}_c \left(\hat{\mathbf{f}} + \mathbf{g}\hat{\mathbf{u}} \right). \tag{58}
$$

From Eq. (40), we have

$$
Q(\mathbf{x}) = -U(\mathbf{u}^*) - \mathbf{W}_c^\top \nabla \boldsymbol{\sigma}_c \left(\mathbf{f} + \mathbf{g}\mathbf{u}^* \right) + \epsilon_{hjb}.
$$

Therefore, substituting this expression into Eq. (58), we obtain

$$
\begin{aligned}
\delta_B = &U(\hat{\mathbf{u}}) - U(\mathbf{u}^*) - \mathbf{W}_c^\top \nabla \boldsymbol{\sigma}_c (\mathbf{f} + \mathbf{g}\mathbf{u}^*) \\
&+ \hat{\mathbf{W}}_c^\top \nabla \boldsymbol{\sigma}_c \left(\hat{\mathbf{f}} + \mathbf{g}\hat{\mathbf{u}} \right) + \epsilon_{hjb}.
\end{aligned} \tag{59}
$$

Considering the definition of $U(\mathbf{u}^*)$ and $U(\hat{\mathbf{u}})$, given respectively in Eqs. (6) and (9), and knowing that $\ln(\mathbf{1} - \tanh^2(\mathbf{D}^*)) = \ln(\mathbf{4}) - 2\mathbf{D}^* \mathrm{sgn}(\mathbf{D}^*) + \boldsymbol{\epsilon}_{D^*}$ and $\ln(\mathbf{1} - \tanh^2(\hat{\mathbf{D}})) = \ln(\mathbf{4}) - 2\hat{\mathbf{D}} \mathrm{sgn}(\hat{\mathbf{D}}) + \boldsymbol{\epsilon}_D$ for some bounded $\boldsymbol{\epsilon}_{D^*}$ and $\boldsymbol{\epsilon}_D$ [18], where $\mathrm{sgn}(\cdot)$ denotes the *signum* function, we develop the following expression:

$$
\begin{aligned}
U(\hat{\mathbf{u}}) - U(\mathbf{u}^*) = &2\lambda^2 \bar{\mathbf{R}} \left(\mathbf{D}^* \mathrm{sgn}(\mathbf{D}^*) - \hat{\mathbf{D}} \mathrm{sgn}(\hat{\mathbf{D}}) \right) \\
&- \hat{\mathbf{W}}_c^\top \nabla \boldsymbol{\sigma}_c \mathbf{g}\hat{\mathbf{u}} + \mathbf{W}_c^\top \nabla \boldsymbol{\sigma}_c \mathbf{g}\mathbf{u}^* \\
&+ \nabla \epsilon_c \mathbf{g}\mathbf{u}^* + \lambda^2 \bar{\mathbf{R}} (\boldsymbol{\epsilon}_D - \boldsymbol{\epsilon}_{D^*}).
\end{aligned} \tag{60}
$$

The *signum* function can be approximated by a tanh function with the following relation quantifying the approximation error [35]:

$$
0 \leq x \mathrm{sgn}(x) - x \tanh(\kappa x) \leq \frac{1}{0.2785\kappa}.
$$

Therefore, the expression in Eq. (60) can be written as

$$
U(\hat{\mathbf{u}}) - U(\mathbf{u}^*) = 2\lambda^2 \bar{\mathbf{R}} \left(\mathbf{D}^* \tanh(\kappa \mathbf{D}^*) - \hat{\mathbf{D}} \tanh(\kappa \hat{\mathbf{D}}) \right)
$$

$$+ 2\lambda^2\bar{\mathbf{R}}(\boldsymbol{\kappa}_{D^*} - \boldsymbol{\kappa}_D) - \hat{\mathbf{W}}_c^\top \nabla\boldsymbol{\sigma}_c \mathbf{g}\hat{\mathbf{u}} + \mathbf{W}_c^\top \nabla\boldsymbol{\sigma}_c \mathbf{g}\mathbf{u}^* + \nabla\epsilon_c \mathbf{g}\mathbf{u}^* + \lambda^2\bar{\mathbf{R}}(\boldsymbol{\epsilon}_D - \boldsymbol{\epsilon}_{D^*}), \tag{61}$$

where $\boldsymbol{\kappa}_{D^*}$ and $\boldsymbol{\kappa}_D$ denote the approximation errors. Then, considering the definitions of $\mathbf{D}^*$ and $\hat{\mathbf{D}}$, and adding and subtracting $\lambda\mathbf{W}_c^\top \nabla\boldsymbol{\sigma}_c \mathbf{g}\tanh(\kappa\hat{\mathbf{D}})$ to the right-hand side of Eq. (61), we have

$$\begin{aligned} U(\hat{\mathbf{u}}) - U(\mathbf{u}^*) &= \lambda\tilde{\mathbf{W}}_c^\top \nabla\boldsymbol{\sigma}_c \mathbf{g}\tanh(\kappa\hat{\mathbf{D}}) \\ &+ 2\lambda^2\bar{\mathbf{R}}(\boldsymbol{\kappa}_{D^*} - \boldsymbol{\kappa}_D) - \hat{\mathbf{W}}_c^\top \nabla\boldsymbol{\sigma}_c \mathbf{g}\hat{\mathbf{u}} + \mathbf{W}_c^\top \nabla\boldsymbol{\sigma}_c \mathbf{g}\mathbf{u}^* \\ &+ \nabla\epsilon_c \mathbf{g}\mathbf{u}^* + \lambda^2\bar{\mathbf{R}}(\boldsymbol{\epsilon}_D - \boldsymbol{\epsilon}_{D^*}) \\ &+ \lambda\mathbf{W}_c^\top \nabla\boldsymbol{\sigma}_c \mathbf{g}\left(\tanh(\kappa\mathbf{D}^*) - \tanh(\kappa\hat{\mathbf{D}})\right). \end{aligned} \tag{62}$$

Substituting Eq. (62) into Eq. (59) and doing certain manipulations, we obtain

$$\delta_B = -\boldsymbol{\beta}^\top \tilde{\mathbf{W}}_c + \epsilon_\delta, \tag{63}$$

where $\boldsymbol{\beta} \in \mathbb{R}^{\mathcal{L}_c}$ is defined in Eq. (44) and $\epsilon_\delta \in \mathbb{R}$ is given by

$$\begin{aligned} \epsilon_\delta \triangleq\ & 2\lambda^2\bar{\mathbf{R}}(\boldsymbol{\kappa}_{D^*} - \boldsymbol{\kappa}_D) + \lambda^2\bar{\mathbf{R}}(\boldsymbol{\epsilon}_D - \boldsymbol{\epsilon}_{D^*}) + \nabla\epsilon_c \mathbf{g}\mathbf{u}^* \\ &+ \epsilon_{hjb} + \lambda\mathbf{W}_c^\top \nabla\boldsymbol{\sigma}_c \mathbf{g}\left(\tanh(\kappa\mathbf{D}^*) - \tanh(\kappa\hat{\mathbf{D}})\right) \\ &- \mathbf{W}_c^\top \nabla\boldsymbol{\sigma}_c \tilde{\mathbf{f}}, \end{aligned}$$

where all of its elements are bounded terms, and thus an upper bound can be considered for it as $|\epsilon_\delta| \leq \bar{\epsilon}_\delta$. Also note that approximation errors $\boldsymbol{\epsilon}_{D^*}$, $\boldsymbol{\epsilon}_D$, $\boldsymbol{\kappa}_{D^*}$, and $\boldsymbol{\kappa}_D$ converge to zero as $\mathbf{x}$ goes to zero. Following the same steps, the unmeasurable form of simulated Bellman error δ_{Bj} can be obtained as

$$\delta_{Bj} = -\boldsymbol{\beta}_j^\top \tilde{\mathbf{W}}_c - \mathbf{W}_c^\top \nabla\boldsymbol{\sigma}_{cj} \tilde{\mathbf{f}}_{sj} + \epsilon_{\delta j}, \tag{64}$$

where $\tilde{\mathbf{f}}_{sj} \triangleq \mathbf{f}_j - \hat{\mathbf{f}}_{sj} = \mathbf{e}_{sj}$, subscript j indicates the jth sample of the variables, and $\epsilon_{\delta j} \triangleq 2\lambda^2\bar{\mathbf{R}}(\boldsymbol{\kappa}_{D^*} - \boldsymbol{\kappa}_{Dj}) + \lambda^2\bar{\mathbf{R}}(\boldsymbol{\epsilon}_{Dj} - \boldsymbol{\epsilon}_{D^*}) + \lambda\mathbf{W}_c^\top \nabla\boldsymbol{\sigma}_c \mathbf{g}(\tanh(\kappa\mathbf{D}^*) - \tanh(\kappa\hat{\mathbf{D}}(\mathbf{x}_j))) - \nabla\epsilon_c \mathbf{f}_j$, for which an upper constant bound can be considered.

Then, using Eqs. (63) and (64) and the adaptation rule (Eq. 43), the time derivative of L_{W_c} can be written as

$$\begin{aligned} \dot{L}_{W_c} =\ & -\tilde{\mathbf{W}}_c^\top \bar{\boldsymbol{\beta}}\bar{\boldsymbol{\beta}}^\top \tilde{\mathbf{W}}_c + \tilde{\mathbf{W}}_c^\top \frac{\bar{\boldsymbol{\beta}}}{m_s}\epsilon_\delta \\ &- \frac{\gamma_{c1}}{N}\sum_{j=1}^{N} \tilde{\mathbf{W}}_c^\top \bar{\boldsymbol{\beta}}_j \bar{\boldsymbol{\beta}}_j^\top \tilde{\mathbf{W}}_c \\ &- \frac{\gamma_{c1}}{N}\sum_{j=1}^{N} \tilde{\mathbf{W}}_c^\top \frac{\bar{\boldsymbol{\beta}}_j}{m_{sj}} \mathbf{W}_c^\top \nabla\boldsymbol{\sigma}_{cj} \tilde{\mathbf{f}}_{sj} + \frac{\gamma_{c1}}{N}\sum_{j=1}^{N} \tilde{\mathbf{W}}_c^\top \frac{\bar{\boldsymbol{\beta}}_j}{m_{sj}}\epsilon_{\delta j} \\ &- \frac{\gamma_{c2}\sum_{j=1}^{N} \tilde{\mathbf{W}}_c^\top \bar{\boldsymbol{\beta}}_j \bar{\boldsymbol{\beta}}_j^\top \tilde{\mathbf{W}}_c}{N(\|\boldsymbol{\Xi}\| + \varepsilon_c)} + \frac{\gamma_{c2}\sum_{j=1}^{N} \tilde{\mathbf{W}}_c^\top \bar{\boldsymbol{\beta}}_j \epsilon_{\delta j}/m_{sj}}{N(\|\boldsymbol{\Xi}\| + \varepsilon_c)} \\ &- \frac{\gamma_{c2}\sum_{j=1}^{N} \tilde{\mathbf{W}}_c^\top \bar{\boldsymbol{\beta}}_j \mathbf{W}_c^\top \nabla\boldsymbol{\sigma}_{cj} \tilde{\mathbf{f}}_{sj}/m_{sj}}{N(\|\boldsymbol{\Xi}\| + \varepsilon_c)}, \end{aligned}$$

where $\bar{\boldsymbol{\beta}} \triangleq \boldsymbol{\beta}/(1 + \boldsymbol{\beta}^\top\boldsymbol{\beta})$ and $m_s \triangleq 1 + \boldsymbol{\beta}^\top\boldsymbol{\beta}$ that satisfy the following inequality:

$$\|\frac{\bar{\boldsymbol{\beta}}}{m_s}\| \leq b_\beta < 1,$$

in which b_β is a positive constant. We have

$$-\frac{\gamma_{c2}\sum_{j=1}^{N} \tilde{\mathbf{W}}_c^\top \bar{\boldsymbol{\beta}}_j \bar{\boldsymbol{\beta}}_j^\top \tilde{\mathbf{W}}_c}{N(\|\boldsymbol{\Xi}\| + \varepsilon_c)} \leq -\gamma_{c2}\vartheta_c\|\tilde{\mathbf{W}}_c\|,$$

where $\vartheta_c > 0$ is defined as

$$\vartheta_c \triangleq 1 - \frac{(c_{\beta_2} - c_{\beta_1})\|\tilde{\mathbf{W}}_c\| + \hbar_1 + \hbar_2 + \varepsilon_c}{c_{\beta_2}\|\tilde{\mathbf{W}}_c\| + \hbar_1 + \hbar_2 + \varepsilon_c},$$

in which $c_{\beta_2} \triangleq \frac{1}{N}(\sup_{t\in\mathbb{R}_{\geq t_0}}(\lambda_{\max}(\sum_{j=1}^{N}\frac{\boldsymbol{\beta}_j\boldsymbol{\beta}_j^\top}{(1+\boldsymbol{\beta}_j^\top\boldsymbol{\beta}_j)^2})))$, and $\hbar_1 \triangleq \bar{b}_\beta b_{\sigma cx} b_{\sigma sx}\|\boldsymbol{W}_c\|\|\tilde{\mathbf{W}}_s\|_F$, $\hbar_2 \triangleq \bar{b}_\beta b_{\sigma cx}\bar{\epsilon}_{sj}\|\boldsymbol{W}_c\| + \bar{b}_\beta\bar{\epsilon}_{\delta j}$, in which $\bar{b}_\beta \triangleq \sup_{j=1\cdots N}(\|\bar{\boldsymbol{\beta}}_j/m_{sj}\|)$, $\bar{\epsilon}_{sj} \triangleq \sup_{j=1\cdots N}(\|\boldsymbol{\epsilon}_{sj}\|)$ and $\bar{\epsilon}_{\delta j} \triangleq \sup_{j=1\cdots N}(|\epsilon_{\delta j}|)$. Also, assuming that $\varepsilon_c > \sup_{j=1\cdots N}(\|\tilde{\mathbf{f}}_{sj}\|) + \bar{\epsilon}_\delta$, the following inequality can be developed:

$$\begin{aligned} &-\frac{\gamma_{c2}\sum_{j=1}^{N} \tilde{\mathbf{W}}_c^\top \bar{\boldsymbol{\beta}}_j \mathbf{W}_c^\top \nabla\boldsymbol{\sigma}_{cj} \tilde{\mathbf{f}}_{sj}/m_{sj}}{N(\|\boldsymbol{\Xi}\| + \varepsilon_c)} \\ &\quad + \frac{\gamma_{c2}\sum_{j=1}^{N} \tilde{\mathbf{W}}_c^\top \bar{\boldsymbol{\beta}}_j \epsilon_{\delta j}/m_{sj}}{N(\|\boldsymbol{\Xi}\| + \varepsilon_c)} \leq \\ &\gamma_{c2}\bar{b}_\beta b_{\sigma cx}\varpi_{c1}\|\mathbf{W}_c\|\|\tilde{\mathbf{W}}_c\| + \gamma_{c2}\bar{b}_\beta\varpi_{c2}\|\tilde{\mathbf{W}}_c\| \end{aligned}$$

for some $\varpi_{c1}, \varpi_{c2} < 1$. Then, defining the following positive constants:

$$\begin{aligned} \varrho_1 &\triangleq \gamma_{c1}\bar{b}_\beta b_{\sigma cx} b_{\sigma sx}\|\boldsymbol{W}_c\|, \\ \varrho_2 &\triangleq \gamma_{c1}\bar{b}_\beta b_{\sigma cx}\bar{\epsilon}_{sj}\|\boldsymbol{W}_c\| + \gamma_{c2}\bar{b}_\beta b_{\sigma cx}\varpi_{c1}\|\boldsymbol{W}_c\| \\ &\quad + \gamma_{c1}\bar{b}_\beta\bar{\epsilon}_{\delta j} + \gamma_{c2}\bar{b}_\beta\varpi_{c2} + b_\beta\bar{\epsilon}_\delta, \end{aligned}$$

and considering Eq. (45), the following upper bound can be written for $\dot{L}_{W_c}$:

$$\dot{L}_{W_c} \leq -\gamma_{c1}c_{\beta_1}\|\tilde{\mathbf{W}}_c\|^2 + \varrho_1\|\tilde{\mathbf{W}}_c\|\|\tilde{\mathbf{W}}_s\|_F$$

$$+ (\varrho_2 - \gamma_{c2}\vartheta_c)\|\tilde{\mathbf{W}}_c\|. \tag{65}$$

Therefore, using Eqs. (37), (57), and (65) and Young's inequality, an upper bound for $\dot{L}$ can be written as

$$\begin{aligned}\dot{L} \leq & - q_{\min}\|\mathbf{x}\|^2 - (\gamma_{c1}c_{\beta_1} - \frac{\varrho_1}{\eta_1^2} - \frac{\eta_2^2}{4})\|\tilde{\mathbf{W}}_c\|^2 \\ & - (\gamma_{s1}c_{\sigma_1} - \frac{\eta_1^2\varrho_1}{4} - \frac{\eta_3^2}{4})\|\tilde{\mathbf{W}}_s\|_F^2 \\ & + \kappa_1 + \frac{(\varrho_2 - \gamma_{c2}\vartheta_c)^2}{\eta_2^2} + \frac{\epsilon_w^2}{\eta_3^2},\end{aligned} \tag{66}$$

where $\eta_1, \eta_2, \eta_3 \in \mathbb{R}^+$ are adjustable constants. Therefore, considering Eq. (66), whenever the gain conditions in Eq. (46) are satisfied, we conclude that $\mathbf{x}$, $\tilde{\mathbf{W}}_c$, and $\tilde{\mathbf{W}}_s$ are uniformly ultimately bounded.

References

1. SN Balakrishnan and Victor Biega: Adaptive-critic-based neural networks for aircraft optimal control. J. Guid. Control Dyn. **19**(4), 893–898 (1996)
2. He, P. and Jagannathan, S.: Reinforcement learning neural-network-based controller for nonlinear discrete-time systems with input constraints. *IEEE Trans. Syst. Man Cybern. Part B Cybern.* 37(2):425–436 (2007)
3. T Dierks, and Sarangapani, Jagannathan: Optimal control of affine nonlinear continuous-time systems. In *Proceedings of the 2010 American Control Conference*. pp. 1568–1573 (2010)
4. Doya, Kenji: Reinforcement learning in continuous time and space. Neural Comput. **12**(1), 219–245 (2000)
5. Vamvoudakis, K.G., Lewis, F.L.: Online actor-critic algorithm to solve the continuous-time infinite horizon optimal control problem. Automatica **46**(5), 878–888 (2010)
6. Vrabie, D., Pastravanu, O., Abu-Khalaf, M., Lewis, F.L.: Adaptive optimal control or continuous-time linear systems based on policy iteration. Automatica **45**(2), 477–484 (2009)
7. Bhasin, S., Kamalapurkar, R., Johnson, M., Vamvoudakis, K.G., Lewis, F.L., Dixon, W.E.: A novel actor critic identifier architecture for approximate optimal control of uncertain nonlinear systems. Automatica **49**(1), 82–92 (2013)
8. Modares, H., Lewis, F.L.: Optimal tracking control of nonlinear partially-unknown constrained-input systems using integral reinforcement learning. Automatica **50**(7), 1780–1792 (2014)
9. Modares, H., Lewis, F.L., Naghibi-Sistani, M.-B.: Integral reinforcement learning and experience replay for adaptive optimal control of partially-unknown constrained-input continuous-time systems. Automatica **50**(1), 193–202 (2014)
10. Kamalapurkar, R., Andrews, L., Walters, P., Dixon, W.E.: Model-based reinforcement learning for infinite-horizon approximate optimal tracking. IEEE Trans. Neural Netw. Learn. Syst. **28**(3), 753–758 (2016)
11. Kamalapurkar, R., Walters, P., and Dixon, W.,: Concurrent learning-based approximate optimal regulation. In *52nd IEEE Conference on Decision and Control*, pp. 6256–6261 (2013)
12. Zhao, Bo., Liu, Derong, Alippi, Cesare: Sliding-mode surface-based approximate optimal control for uncertain nonlinear systems with asymptotically stable critic structure. IEEE Trans. Cybern. **51**(6), 2858–2869 (2020)
13. Abu-Khalaf, M., Lewis, F.L.: Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach. Automatica **41**(5), 779–791 (2005)
14. Guo, Xinxin, Yan, Weisheng, Cui, Rongxin: Integral reinforcement learning-based adaptive nn control for continuous-time nonlinear mimo systems with unknown control directions. IEEE Trans. Syst. Man Cybern. Syst. **50**(11), 4068–4077 (2019)
15. Modares, H., Lewis, F.L., Naghibiistani, M.-B.: Online solution of nonquadratic two-player zero-sum games arising in the Hs control of constrained input systems. Int. J. Adap. Control Signal Process. **28**(35), 232–254 (2014)
16. Yang, Y., Vamvoudakis, K.G., Modares, H., Yin, Y., Wunsch, D.C.: Safe intermittent reinforcement learning with static and dynamic event generators. IEEE Trans. Neural Netw. Learn. Syst. **31**(12), 5441–5455 (2020)
17. Mishra, A., and Ghosh, S.: Variable gain gradient descent-based reinforcement learning for robust optimal tracking control of uncertain nonlinear system with input constraints. *Nonlinear Dyn.* pp. 2195—2214 (2022)
18. Modares, H., Lewis, F.L., Naghibi-Sistani, M.-B.: Adaptive optimal control of unknown constrained-input systems using policy iteration and neural networks. IEEE Trans. Neural Netw. Learn. Syst. **24**(10), 1513–1525 (2013)
19. Huo, Y., Wang, D., Qiao, J., and Li, M.: Adaptive critic design for nonlinear multi-player zero-sum games with unknown dynamics and control constraints. *Nonlinear Dyn.* pp. 1–13 (2023)
20. Jean-Jacques E, Slotine, WL. et al: *Applied nonlinear control*, volume 199. Prentice hall Englewood Cliffs, NJ, (1991)
21. Sastry, S.: Nonlinear Systems: Analysis, Stability, and Control, vol. 10. Springer Science and Business Media, Berlin (2013)
22. Dong H., Zhao X., and Luo B.: Optimal tracking control for uncertain nonlinear systems with prescribed performance via critic-only ADP. *IEEE Trans. Syst. Man Cybern. Syst.* (2020)
23. Lv, Yongfeng, Ren, Xuemei, Na, Jing: Online optimal solutions for multi-player nonzero-sum game with completely unknown dynamics. Neurocomputing **283**, 87–97 (2018)
24. Wang, Wei, Wen, Changyun: Adaptive actuator failure compensation control of uncertain nonlinear systems with guaranteed transient performance. Automatica **46**(12), 2082–2091 (2010)
25. Xian, B., Dawson, D.M., de Queiroz, M.S., Chen, J.: A continuous asymptotic tracking control strategy for uncertain nonlinear systems. IEEE Trans. Autom. Control **49**(7), 1206–1211 (2004)
26. Marcio S, De Queiroz, Jun, Hu, Darren M, Dawson, Timothy, Burg, and Sreenivasa R, Donepudi: Adaptive posi-

tion/force control of robot manipulators without velocity measurements: Theory and experimentation. *IEEE Trans. Syst. Man Cybern Part B* 27(5):796–809 (1997)
27. Chowdhary, G. and Johnson, E.: Concurrent learning for convergence in adaptive control without persistency of excitation. In *49th IEEE Conference on Decision and Control* p. 3674–3679 (2010)
28. Girish, V.: Chowdhary and Eric N, Johnson: Theory and flight-test validation of a concurrent-learning adaptive controller. J. Guid Control Dyn. **34**(2), 592–607 (2011)
29. Vahidi-Moghaddam, Amin, Mazouchi, Majid, Modares, Hamidreza: Memory-augmented system identification with finite-time convergence. IEEE Control Syst. Lett. **5**(2), 571–576 (2020)
30. Spong, M.W.: On the robust control of robot manipulators. IEEE Trans. Autom. Control **37**(11), 1782–1786 (1992)
31. Hornik, Kurt, Stinchcombe, Maxwell, White, Halbert: Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks. Neural Netw. **3**(5), 551–560 (1990)
32. Edwin, K.P., Chong, E.K., Zak, S.H.: An Introduction to Optimization **75**, 514 (2013)
33. Khalil, H.K.: *Noninear Systems*. Prentice-Hall. New Jersey, 3rd edn (1996)
34. Patre, P.: Lyapunov-based robust and adaptive control of nonlinear systems using a novel feedback structure. University of Florida, Florida (2009)
35. Marios M, Polycarpou and Petros A, Ioannou: A robust adaptive nonlinear control design. In *1993 American Control Conference* pp. 1365–1369 (1993)