A Reinforcement Learning-Based Control Approach for Unknown Nonlinear Systems with Persistent Adversarial Inputs

Xiangnan Zhong
Department of Computer & Electrical Engineering and Computer Science
Florida Atlantic University
Boca Raton, FL, USA
xzhong@fau.edu

Haibo He
Department of Electrical, Computer and Biomedical Engineering
University of Rhode Island
Kingston, RI, USA
haibohe@uri.edu

***Abstract*—This paper develops an intelligent control method based on reinforcement learning techniques for unknown nonlinear continuous-time systems in an adversarial environment. The developed method can automatically learn the optimal control input for the system and also predict the worst case adversarial input that one adversary can bring into. Besides, we assume that the agent can only observe partial information of the environment during the learning process. Therefore, a neural network-based observer is developed to adaptively reconstruct the hidden states and dynamics. Then, theoretical analysis is provided to show the stability of the developed intelligent control and the accuracy of the established observer. This method has been applied on a torsional pendulum system and the results demonstrate the effectiveness of the designed approach.**

***Index Terms*—Reinforcement learning, zero-sum games, neural networks, observer, online learning and control.**

I. Introduction

In recent decades, reinforcement learning (RL) and adaptive dynamic programming (ADP) have been studied and adopted in a variety of challenging domains and achieved promising results [1]–[9]. By mirroring the human learning process to explore and interact with the environment, RL and ADP have been widely used in the field of intelligent control and smart systems [10]–[15]. Usually, RL and ADP tackle optimal control problems by estimating the solutions of the Riccati equation for linear systems and the Hamilton-Jacobi-Bellman (HJB) equation for nonlinear systems, respectively [16]–[18]. In [19], an adaptive critic method was developed based on the neural network techniques to achieve the optimal control and also approximate the corresponding performance index. An integral RL (IRL) method was designed in [20] to solve the optimal tracking problems and then the explicit theoretical analysis was also provided to show the stability of the designed method. Besides, robust control problems were considered in [21]–[24]. Various adaptive control schemes were developed based on the RL and ADP methods to deal with the uncertainties in system dynamics. Recently, RL and ADP techniques have been studied in the game theory and multiplayer systems to obtain the optimal decisions for each individual agent [25]–[30]. One of the popular problems is the two-player zero-sum game [31]–[36], where the agents are classified as either the defensive agents to complete the specific missions or the adversarial agents to act in a manner so as to prevent the defensive agents from achieving the goals [37]. Therefore, the adversarial inputs introduce noise which will impact the learning performance. To solve this problem, an iterative ADP-based method was developed in [38] to approximate the solution of the Hamilton-Jacobi-Isaacs (HJI) equation and make the system achieves Nash equilibrium. In [39], a neurodynamic programming method with two-player policy iterations was developed for the zero-sum game which is subject to the constrained control inputs. This problem was also investigated in [40] with two value iterative algorithms for feedback strategies. Both the on-policy and the off-policy RL methods for the stochastic differential games were established in [41], and the promising results were achieved. In addition, the authors in [42] designed a deterministic mixed optimal control scheme for the cases that the saddle point solution does not exist in the zero-sum game. Furthermore, a cooperative RL-based control approach was developed in [43] for the consensus problems of the large networks and complex systems with multiple players in the uncertain, dynamic and adversarial environment.

So far, many of the studies consider zero-sum problems in an environment that the system state is fully observable. However, in many cases, the feedback information can only represent parts of the system state, which makes the learning data imperfect and sometimes unreliable with the existence of the adversarial input. Such problems can be referred as the partially observable processes. Increasing attention has been attracted recently to solve this problem. One popular way is to design a belief state [44]–[46] based on the sufficient statistic of system complete information. However, this design may cause intensive computation burden when trying to obtain such state. The situation becomes worse with the increasing of

This work was supported by the National Science Foundation under Grant ECCS 1947419 and ECCS 1917275.

the state dimensions. Recently, RL and ADP techniques have been introduced into this field and provide the opportunities to adaptively approximate the solutions with the help of iterative algorithms under the partially observable conditions [47]–[51]. Theoretical discussions have been provided to show the relationship between the partially feedback and the hidden state information [52]. The results have demonstrated the feasibility of using the output state to design the control input [53], [54].

Motivated by the above observations and literature studies, this paper designs a RL-based control approach for unknown partially observable systems with persistent adversarial inputs. The major contributions of this paper are as following: First, the problem has been formulated into a two-player zero-sum problem with one defensive agent to minimize the performance index and one adversarial agent to maximize it. Second, a RL-based control method is designed in an input-output setting to obtain the optimal control input and the worst case adversarial input that one adversary can bring into. Moreover, an observer is built to adaptively reconstruct the system dynamics and state variables in an online fashion. Third, the designed method has been implemented based on the neural network techniques. Instead of applying the action and adversarial networks to estimate the control and adversarial inputs in literature, our designed method only require a critic network with the help of established observer to obtain both inputs in an online fashion. The learning process does not require any information of system dynamics. Therefore, this design will significantly reduce the communication cost and computation complexity. Rigorous proofs are also provided to guarantee the stability of the closed-loop control design.

The rest of this paper is organized as follows. In Section II, we formulate the zero-sum problem analyzed in this paper. Section III provides the design of proposed RL-based optimal control method in the input-output setting. Specifically, the Nash equilibrium is studied for the system with theoretical discussions. An observer is then established in this section based on the neural network techniques to reconstruct the system variables and dynamics. The online learning process is developed and a critic network is established to help estimate the optimal control action and worst case adversarial input. The designed closed-loop system is guaranteed stable based on the Lyapunov analysis. In Section IV, numerical experiment results and analysis are presented to demonstrate the effectiveness of the proposed control scheme. Finally, Section V concludes this work.

II. Problem Formulation

A nonlinear continuous-time system with persistent adversarial inputs is considered as following

$$\begin{aligned} \dot{x} &= f(x) + g(x)u + k(x)v \\ y &= Cx \end{aligned} \tag{1}$$

where $x \in \mathbb{R}^n$ is the internal state vector with the initial condition x_0, $u \in \mathbb{R}^m$ is the control input, $v \in \mathbb{R}^l$ is the adversarial input, $f(x)$, $g(x)$, and $k(x)$ are the unknown functions with $f(0) = 0$, $C \in \mathbb{R}^{p*n}$ is the output matrix, and $y \in \mathbb{R}^p$ is the system output. Assume that the system $f+gu+kv$ is Lipschtz continuous on set $\Omega \subseteq \mathbb{R}^n$ containing the origin.

Assumption 1 [50]: The nonlinear continuous-time system described in (1) is controllable and observable. Here, the system output y is considered as the measurable data.

The cost function associated to the system depends on the state x, control input u and adversarial input v, which can be described as,

$$J(x_0, u, v) = \int_0^\infty \underbrace{\left(x^T \Lambda x + u^T R u - \rho^2 v^T v\right)}_{U(x,u,v)} d\tau \tag{2}$$

where $U(x,u,v) = x^T\Lambda x + u^T R u - \rho^2 v^T v$ is the utility function, Λ and R are the positive definite matrices, and ρ is the amount of attenuation from the adversarial input to the defined performance.

In this way, this problem is formulated into a two-player zero-sum game. It is desired to find the saddle point solution (u^*, v^*), so that

$$J(x_0, u^*, v) \le J(x_0, u^*, v^*) \le J(x_0, u, v^*). \tag{3}$$

The optimal performance index can be defined as

$$V^*(x) = \min_u \max_v \int_t^\infty \left(x^T \Lambda x + u^T R u - \rho^2 v^T v\right) d\tau. \tag{4}$$

Comparing with the cost function (2), we obtain

$$V^*(x_0) = \min_u \max_v J(x_0, u, v). \tag{5}$$

Therefore, in this zero-sum game, two players are considered as u and v. Particularly, player u seeks to minimize the performance index, while the other player v seeks to maximize it. Note that, the performance index described in (4) cannot be obtained due to the infeasible access of internal state x. Hence, this paper designs a RL-based control method to solve the problem in an input-output setting. The learning process does not require any information of the system dynamics.

III. RL-Based Control for Nash Equilibrium with Only Input-Output Data

This section includes three parts. First, the Nash equilibrium are investigated for the proposed game. Second, we establish an observer to identify the system dynamics and reconstruct the hidden state. Third, the reinforcement learning scheme is applied along with neural network implementation to estimate the performance index and obtain the control and adversarial inputs. The stability of the proposed closed-loop control design is also discussed by the end of this section.

A. Nash Equilibrium

Assume equation (4) is continuously differentiable. By transforming, we obtain the Hamiltonian function as

$$\begin{aligned} &\mathscr{H}(x, u, v, \frac{\partial V^*}{\partial x}) \\ &\quad = \frac{\partial {V^*}^T(x)}{\partial x}(f(x) + g(x)u + k(x)v) + U(x,u,v). \end{aligned} \tag{6}$$

Therefore, the solution (u^*, v^*) satisfies the first order necessary condition, which is given by the gradient of (6) with respect to u and v, respectively, i.e., $\frac{\partial \mathscr{H}}{\partial u} = 0$ and $\frac{\partial \mathscr{H}}{\partial v} = 0$. Hence, we obtain the optimal control input as,

$$u^* = \arg\min_u \mathscr{H}(x, u, v, \frac{\partial V^*}{\partial x}) = -\frac{1}{2} R^{-1} g^T(x) \frac{\partial V^*(x)}{\partial x}, \tag{7}$$

and the worst case adversarial input as

$$v^* = \arg\max_v \mathscr{H}(x, u, v, \frac{\partial V^*}{\partial x}) = \frac{1}{2\rho^2} k^T(x) \frac{\partial V^*(x)}{\partial x}. \tag{8}$$

By substituting the saddle point solution (7) and (8) into (6), we have the HJI equation

$$\begin{aligned} \mathscr{H}(x, u^*, v^*, \frac{\partial V^*}{\partial x}) &= \frac{\partial V^{*^T}(x)}{\partial x}\Big(f(x) + g(x)u + k(x)v\Big) \\ &\quad + x^T \Lambda x + u^T R u - \rho^2 d^T d \\ &= \frac{\partial V^{*^T}(x)}{\partial x} f(x) - \frac{1}{4} \frac{\partial V^{*^T}(x)}{\partial x} g(x) R^{-1} g^T(x) \frac{\partial V^*(x)}{\partial x} \\ &\quad + \frac{1}{4\rho^2} \frac{\partial V^{*^T}(x)}{\partial x} k(x) k^T(x) \frac{\partial V^*(x)}{\partial x} + x^T \Lambda x = 0 \end{aligned} \tag{9}$$

The following theorem investigates the Nash equilibrium of system (1).

Theorem 1: Consider the nonlinear continuous-time system (1). Let $V(x)$ be a solution of the HJI equation (9) with the optimal control input u^* given as (7) and the worst case adversarial input v^* given as (8), then we have that (i) the system can asymptotically stabilize to the equilibrium point; (ii) the system is in Nash equilibrium with the solution (u^*, v^*).

Proof: Choose the Lyapunov function as

$$L_v = V(x) = \int_t^\infty \big(x^T \Lambda x + u^T R u - \rho^2 v^T v\big) d\tau. \tag{10}$$

The performance index can be represented as quadratic in terms of the system state,

$$V(x) = x^T P x \tag{11}$$

where P is the unique symmetric positive definite matrix that solves the following equation,

$$\mathscr{H}(x, u^*, v^*, x^T P) = 0. \tag{12}$$

Hence, $V(x) = 0$ if and only if $x = 0$. Considering (9), (11), and (12), the time derivative of performance index $V(x)$ for $t \geq 0$ along the closed-loop solution satisfies,

$$\begin{aligned} \dot{V}(x) &= \frac{\partial V^T(x)}{\partial x}\big(f(x) + g(x)u + k(x)v\big) \\ &= -\Big(x^T \Lambda x + u^T R u - \rho^2 d^T d\Big) \end{aligned} \tag{13}$$

Assuming the condition $g(x)R^{-1}g^T(x) > k^T(x)k(x)$ is satisfied, we can upper bound

$$\dot{V}(x) \leq -x^T \Lambda x \tag{14}$$

with $\dot{V}(x) = 0$ if and only if $x = 0$. Therefore, the system (1) is asymptotically stable in the equilibrium point ($x = 0$), which completes the first part of the proof.

Since the system is asymptotically stable at the origin, we can further conclude that $V(x) = 0$ when $t \to \infty$. Hence, the cost function (2) can be rewritten as

$$\begin{aligned} &J(x_0, u, v) \\ &= \int_0^\infty \big(x^T \Lambda x + u^T R u - \rho^2 v^T v\big) d\tau + V(x_0) + \int_0^\infty \dot{V}^* d\tau \\ &= \int_0^\infty \Bigg(x^T \Lambda x + u^T R u - \rho^2 v^T v + \frac{\partial V^{*^T}(x)}{\partial x}\big(f(x) + g(x)u \\ &\qquad + k(x)v\big)\Bigg) d\tau + x_0^T P x_0 \end{aligned} \tag{15}$$

where $V^*(x)$ is the optimal performance index.

Furthermore, considering the control action u^* and the adversarial input v^* given in (7) and (8), respectively, we can further rewrite equation (15) as

$$\begin{aligned} &J(x_0, u, v) \\ &= \int_0^\infty \Bigg(x^T \Lambda x + \frac{\partial V^{*^T}(x)}{\partial x} f(x) + \Big(u^T R u + \frac{\partial V^{*^T}(x)}{\partial x} \\ &\quad \cdot g(x)u\Big) - \rho^2\Big(v^T v - \frac{1}{\rho^2} \frac{\partial V^{*^T}(x)}{\partial x} k(x) v\Big)\Bigg) d\tau + x_0^T P x_0 \\ &= \int_0^\infty \Big((u - u^*)^T R (u - u^*) - \rho^2 (v - v^*)^T (v - v^*)\Big) d\tau \\ &\quad + \int_0^\infty \mathscr{H}(x, u^*, v^*, \frac{\partial V^*}{\partial x}) d\tau + x_0^T P x_0. \end{aligned} \tag{16}$$

Since $\mathscr{H}(x, u^*, v^*, \frac{\partial V^*}{\partial x}) = 0$, it becomes

$$\begin{aligned} &J(x_0, u, v) \\ &= \int_0^\infty \Big((u - u^*)^T R (u - u^*) - \rho^2 (v - v^*)^T (v - v^*)\Big) d\tau \\ &\quad + x_0^T P x_0. \end{aligned} \tag{17}$$

Now, setting $u = u^*$, we obtain

$$J(x_0, u^*, v) = -\rho^2 \int_0^\infty (v - v^*)^T (v - v^*) d\tau + x_0^T P x_0. \tag{18}$$

Setting $v = v^*$, we have

$$J(x_0, u, v^*) = \int_0^\infty (u - u^*)^T R (u - u^*) d\tau + x_0^T P x_0. \tag{19}$$

Finally, setting $u = u^*$ and $v = v^*$, we obtain

$$J(x_0, u^*, v^*) = x_0^T P x_0. \tag{20}$$

Hence, it follows $J(x_0, u^*, v) \leq J(x_0, u^*, v^*) \leq J(x_0, u, v^*)$, and the Nash equilibrium is achieved, which completes the proof. ■

B. Observer Design

This subsection designs an observer based on the neural network techniques to adaptively reconstruct the system internal state and dynamics in an online fashion. Specifically, we rewrite the nonlinear system (1) as

$$\begin{aligned} \dot{x} &= \mathcal{F}x + f'(x) + g(x)u + k(x)v \\ y &= Cx \end{aligned} \tag{21}$$

where $\mathcal{F}$ is a Hurwitz matrix, which is chosen such that $(C, \mathcal{F})$ is observable, and $f'(x) = f(x) - \mathcal{F}x$. Since x is restricted to a compact set of $x \in \mathbb{R}^n$, the unknown nonlinear function $f'(x) + g(x)u + k(x)v$ can be reconstructed by a multilayer neural network with sufficiently large number of hidden layer neurons [55]. Therefore, we design a state network for such nonlinear function

$$f'(x) + g(x)u + k(x)v = \omega_o^{*T}\phi(x, u, v) + \vartheta(x) \tag{22}$$

where $\omega_o^* = [\omega_f^*, \omega_g^*, \omega_k^*]$ is the ideal output weights of the state network, which is bounded by $\|\omega_o^*\| \leq \omega_{oM}$, and $\|\vartheta(x)\| \leq \vartheta_M$ is the bounded neural network approximation error. The activation function $\phi(x, u, v)$ is defined as

$$\phi(x, u, v) = \begin{bmatrix} \phi_f(x) & & \\ & \phi_g(x) & \\ & & \phi_k(x) \end{bmatrix} \times \begin{bmatrix} 1 \\ u \\ v \end{bmatrix} \tag{23}$$

in which $\phi_f(x)$, $\phi_g(x)$, and $\phi_k(x)$ are the bounded polynomial basis functions, and therefore $\|\phi(\cdot)\| \leq \phi_M$.

Since the ideal weights ω_o^* are unknown, we consider the estimates $\hat{\omega}_o$ instead, so that

$$f'(\hat{x}) + g(\hat{x})u + k(\hat{x})v = \hat{\omega}_o^T\phi(\hat{x}, u, v) \tag{24}$$

where $\hat{x}$ is the estimated system state. Define the output of the observer as $\hat{y}$. Then, we have the dynamics of the developed observer as

$$\begin{aligned} \dot{\hat{x}} &= \mathcal{F}\hat{x} + \hat{\omega}_o^T\phi(\hat{x}, u, v) + L(y - \hat{y}) \\ \hat{y} &= C\hat{x} \end{aligned} \tag{25}$$

where $L \in \mathbb{R}^{n\times m}$ is the observer gain which is designed such that $\mathcal{F}_c = \mathcal{F} - LC$ is a Hurwitz matrix. Since $(C, \mathcal{F})$ is observable, the gain L is guaranteed to exist.

Hence, define the objective function for the state network as $E_o = 1/2\tilde{y}^2$, where $\tilde{y} = y - \hat{y}$ is the difference between the real and estimated outputs. Then, the updating law becomes

$$\dot{\hat{\omega}}_o = -\beta_o\frac{\partial E_o}{\partial \hat{\omega}_o} = -\beta_o\Big(\tilde{y}^T C\mathcal{F}_c^{-1}\Big)^T\phi(\hat{x}, u, v) \tag{26}$$

where $\beta_o > 0$ is the learning rate of the state network.

The following theorem is provided for the stability of the established observer and the accuracy of the observation.

Theorem 2: For partially observable nonlinear system given in (1) subject to the adversarial inputs, if the observer is developed in (25) with the updating law in (26), then the observation error $\tilde{x} = x - \hat{x}$ and the weights estimation error $\tilde{\omega}_o = \omega_o^* - \hat{\omega}_o$ are uniformly ultimately bounded (UUB).

Proof: Define the Lyapunov function:

$$L_o = \frac{1}{2}\tilde{x}^T T\tilde{x} + tr\Big(\tilde{\omega}_o^T\tilde{\omega}_o\Big) \tag{27}$$

where T is a positive definite matrix that satisfies

$$(\mathcal{F} - LC)^T T + T(\mathcal{F} - LC) = -M \tag{28}$$

in which M is a symmetric positive definite matrix, and $tr(\cdot)$ denotes the matrix trace.

Considering the equations (21), (22) and (25), we have the observation error as

$$\dot{\tilde{x}} = (\mathcal{F} - LC)\tilde{x} + \omega_o^{*T}\phi(x, u, v) - \hat{\omega}_o^T\phi(\hat{x}, u, v) + \vartheta(x). \tag{29}$$

Define $\Theta = \omega_o^{*T}[\phi(x, u, v) - \phi(\hat{x}, u, v)] + \vartheta(x)$ as the entire approximation error. Then, the observation error (29) can be rewritten as

$$\dot{\tilde{x}} = (\mathcal{F} - LC)\tilde{x} + \tilde{\omega}_o^T\phi(\hat{x}, u, v) + \Theta. \tag{30}$$

Note that the entire approximation error Θ is a bounded term, since ω_o^*, $\phi(\cdot)$ and ϑ are all bounded. This means $\|\Theta\| \leq \xi$ for the positive constant.

Therefore, the first derivative of (27) with respect to the system trajectory becomes

$$\begin{aligned} \dot{L}_o =& \frac{1}{2}\dot{\tilde{x}}^T T\tilde{x} + \frac{1}{2}\tilde{x}^T T\dot{\tilde{x}} + tr\Big(\tilde{\omega}_o^T\dot{\tilde{\omega}}_o\Big) \\ =& \frac{1}{2}\Big((\mathcal{F} - LC)\tilde{x} + \tilde{\omega}_o^T\phi(\hat{x}, u, v) + \Theta\Big)^T T\tilde{x} + \frac{1}{2}\tilde{x}^T T \\ & \cdot\Big((\mathcal{F} - LC)\tilde{x} + \tilde{\omega}_o^T\phi(\hat{x}, u, v) + \Theta\Big) + tr\Big(\tilde{\omega}_o^T\dot{\tilde{\omega}}_o\Big) \\ =& \frac{1}{2}\tilde{x}^T\Big((\mathcal{F} - LC)^T T + T(\mathcal{F} - LC)\Big)\tilde{x} \\ & + \tilde{x}^T T\Big(\tilde{\omega}_o^T\phi(\hat{x}, u, v) + \Theta\Big) \\ & + tr\Big(\tilde{\omega}_o^T\beta_o(\mathcal{F} - LC)^{-T}C^T\tilde{y}\phi(\hat{x}, u, v)\Big). \end{aligned} \tag{31}$$

Considering (28), we have

$$\begin{aligned} \dot{L}_o \leq& -\frac{1}{2}\tilde{x}^T M\tilde{x} + \tilde{x}^T T\Big(\tilde{\omega}_o^T\phi(\hat{x}, u, v) + \Theta\Big) \\ & + tr\Big(\tilde{\omega}_o^T\beta_o(\mathcal{F} - LC)^{-T}C^T C\tilde{x}\phi(\hat{x}, u, v)\Big) \\ \leq& -\frac{1}{2}\lambda_{\min}(M)\|\tilde{x}\|^2 + \|\gamma\|\|\tilde{x}\|\tilde{\omega}_M\phi_M \\ & + \|\tilde{x}\|\|T\|\xi \end{aligned} \tag{32}$$

where $\gamma = \beta_o\Big((\mathcal{F} - LC)^{-T}C^T C\Big) + T$ and $\lambda_{\min}(M)$ is the minimal eigenvalue of M. Hence, $\dot{L}_o < 0$ as long as the following condition is satisfied

$$\|\tilde{x}\| > \frac{2\Gamma}{\lambda_{\min}(M)} \tag{33}$$

where $\Gamma = \|\gamma\|\tilde{\omega}_M\phi_M + \|T\|\xi$. Therefore, based on the Lyapunov method, $\tilde{x}$ and $\tilde{\omega}_o$ are guaranteed to be UUB. This completes the proof. ■

C. Online Learning and Stability Analysis

This subsection develops an online learning control method to estimate the optimal control input u and the worst case adversarial input v without any information of the system dynamics. To achieve this goal, we consider that the objectives of the agent and the adversary are to minimize and maximize the following performance index, respectively,

$$V(x) = \int_t^\infty \Big(x^T \Lambda x + u^T R u - \rho^2 v^T v\Big) d\tau. \tag{34}$$

Therefore, we design a critic network to approximate the performance index (34) as

$$V(x) = \omega_c^{*T} \phi_c(x) + \sigma_c(x) \tag{35}$$

where ω_c^* is the idea critic network weights, $\phi_c(x)$ is the activation function and $\|\sigma_c(x)\| \le \sigma_{cM}$ is the bounded critic network error. Hence, we obtain

$$\frac{\partial V(x)}{\partial x} = \nabla \phi_c^T(x) \omega_c^* + \nabla \sigma_c(x) \tag{36}$$

where $\nabla \phi_c(x) = \partial \phi_c(x)/\partial x$ and $\nabla \sigma_c(x) = \partial \sigma_c(x)/\partial x$.

Since ω_c^* is unknown, we consider the estimated critic network weights $\hat{\omega}_c$ and achieve the corresponding estimated performance index as

$$\hat{V}(\hat{x}) = \hat{\omega}_c^T \phi_c(\hat{x}) \tag{37}$$

where $\hat{x}$ is the estimated state from the designed observer (25). Then, we have

$$\frac{\partial \hat{V}(\hat{x})}{\partial \hat{x}} = \nabla \phi_c^T(\hat{x}) \hat{\omega}_c \tag{38}$$

and the approximate form of Hamiltonian becomes

$$\begin{aligned} \mathscr{H}(\hat{x}, u, v, \frac{\partial \hat{V}(\hat{x})}{\partial \hat{x}}) =& \Big(\nabla \phi_c^T(\hat{x}) \hat{\omega}_c\Big)^T \dot{\hat{x}} + \hat{x}^T \Lambda \hat{x} \\ &+ u^T R u - \rho^2 v^T v \end{aligned} \tag{39}$$

where $\nabla \phi_c(\hat{x}) = \partial \phi_c(\hat{x})/\partial \hat{x}$. Since the optimal Hamiltonian $\mathscr{H}(x, u^*, v^*, \frac{\partial V^*}{\partial x}) = 0$, we define $e_c = \mathscr{H}(\hat{x}, u, v, \frac{\partial \hat{V}(\hat{x})}{\partial \hat{x}})$ as the error function for the critic network.

Define the objective function for the critic network as $E_c = \frac{1}{2} e_c^T e_c$. Hence, we have the updating law as

$$\dot{\hat{\omega}}_c = -\beta_c \frac{\epsilon}{(\epsilon^T \epsilon + 1)^2} \Big(\hat{\omega}_c^T \epsilon + \hat{x}^T \Lambda \hat{x} + u^T R u - \rho^2 v^T v\Big)^T \tag{40}$$

where $\epsilon = \nabla \phi_c(\hat{x}) \dot{\hat{x}}$ and $\beta_c > 0$ is the learning rate of the critic network.

Since $\frac{\partial \hat{V}(\hat{x})}{\partial x}$ is the estimation of $\frac{\partial V^*(x)}{\partial x}$, then substituting (38) into (7) and (8), we obtain the online learning optimal control input and the worst case adversarial input as

$$u = -\frac{1}{2} R^{-1} g^T(\hat{x}) \nabla \phi_c^T(\hat{x}) \hat{\omega}_c, \tag{41}$$

$$v = \frac{1}{2\rho^2} k^T(\hat{x}) \nabla \phi_c^T(\hat{x}) \hat{\omega}_c. \tag{42}$$

The coefficient functions $g(\hat{x})$ and $k(\hat{x})$ can be determined based on the developed observer (25) as

$$g(\hat{x}) = \hat{\omega}_o^T \nabla \phi_u(\hat{x}, u, v) \tag{43}$$

$$k(\hat{x}) = \hat{\omega}_o^T \nabla \phi_v(\hat{x}, u, v) \tag{44}$$

where $\nabla \phi_u(\hat{x}, u, v) = \partial \phi(\hat{x}, u, v)/\partial u$ and $\nabla \phi_v(\hat{x}, u, v) = \partial \phi(\hat{x}, u, v)/\partial v$.

Therefore, instead of applying the action and adversarial networks to estimate the control and adversarial inputs in literature, our designed RL-based optimal control method only requires the critic network with the help of established observer to obtain both inputs in an online fashion. This will significantly reduce the communication cost and computation complexity. In addition, considering (41)-(44), this method does not require any information of the system dynamics in the learning process.

The following theorem will provide the stability of the designed closed-loop control system.

Theorem 3: For the nonlinear continuous-time system (1), the observer is designed in (25) with the state network updating law (26), the critic network is established with the updating law (40), the optimal control input is given by (41) and the worst case adversarial input is provided by (42). Then, all the signals of the closed-loop design are UUB.

Proof: Define the Lyapunov function as

$$\begin{aligned} L_{sys} &= L_v + L_o + L_w \\ &= V(x) + \frac{1}{2} \tilde{x}^T T \tilde{x} + tr\Big(\tilde{\omega}_o^T \tilde{\omega}_o\Big) + \beta_c^{-1} tr\Big(\tilde{\omega}_c^T \tilde{\omega}_c\Big) \end{aligned} \tag{45}$$

where

$$\begin{aligned} &L_v = V(x), \quad L_o = \frac{1}{2} \tilde{x}^T T \tilde{x} + tr\Big(\tilde{\omega}_o^T \tilde{\omega}_o\Big), \\ &L_w = \beta_c^{-1} tr\Big(\tilde{\omega}_c^T \tilde{\omega}_c\Big), \quad \tilde{\omega}_c = \omega_c^* - \hat{\omega}_c. \end{aligned} \tag{46}$$

Consider the first derivative of (45). Based on Theorem 1 and 2, we know $\dot{L}_v \le -x^T \Lambda x$ and $\dot{L}_o < 0$ as long as (33) is satisfied. Therefore, only $\dot{L}_w$ needs to be considered,

$$\dot{L}_w = \beta_c^{-1} tr\Big(\tilde{\omega}_c^T \dot{\tilde{\omega}}_c\Big) \tag{47}$$

where $\dot{\tilde{\omega}}_c$ can be described as

$$\dot{\tilde{\omega}}_c = \beta_c \frac{\epsilon}{(\epsilon^T \epsilon + 1)^2} \Big(\epsilon^T \hat{\omega}_c + \hat{x}^T \Lambda \hat{x} + u^T R u - \rho^2 v^T v\Big). \tag{48}$$

Therefore,

$$\begin{aligned} \dot{L}_w &= \beta_c^{-1} tr\Bigg(\beta_c \tilde{\omega}_c^T \frac{\epsilon}{(\epsilon^T \epsilon + 1)^2} \Big(\epsilon^T \hat{\omega}_c \\ &\qquad + \hat{x}^T \Lambda \hat{x} + u^T R u - \rho^2 v^T v\Big)\Bigg) \\ &= \beta_c^{-1} tr\Bigg(-\beta_c \tilde{\omega}_c^T \frac{\epsilon \epsilon^T}{(\epsilon^T \epsilon + 1)^2} \tilde{\omega}_c + \beta_c \tilde{\omega}_c^T \\ &\qquad \cdot \frac{\epsilon}{(\epsilon^T \epsilon + 1)^2} \Big(\epsilon^T \omega_c^* + \hat{x}^T \Lambda \hat{x} + u^T R u - \rho^2 v^T v\Big)\Bigg). \end{aligned} \tag{49}$$

Define $\Omega_\epsilon = \frac{\epsilon}{\epsilon^T\epsilon+1}$ and $\Phi = \epsilon^T\omega_c^* + \hat{x}^T\Lambda\hat{x} + u^TRu - \rho^2 v^T v \leq \Phi_M$, we can further rewrite (49) as

$$\begin{aligned}\dot{L}_w &\leq -||\Omega_\epsilon||^2||\tilde{\omega}_c||^2 + \frac{1}{2}\left(\beta_c||\Omega_\epsilon||^2||\tilde{\omega}_c||^2 + \frac{||\Phi||^2}{\beta_c(\epsilon^T\epsilon+1)^2}\right) \\ &\leq -(1-\beta_c/2)||\Omega_\epsilon||^2||\tilde{\omega}_c||^2 + \frac{||\Phi_M||^2}{2\beta_c}.\end{aligned} \tag{50}$$

Therefore $\dot{L}_w < 0$, if $\beta_c < 2$ and $||\tilde{\omega}_c|| > \frac{||\Phi_M||}{\sqrt{(2-\beta_c)\beta_c}||\Omega_\epsilon||}$. In this way, the first derivative of (45) becomes

$$\dot{L}_{sys} = \dot{L}_v + \dot{L}_o + \dot{L}_w < 0. \tag{51}$$

This means all signals of the closed-loop system is ensured as UUB, which concludes the proof. ■

IV. Simulation Results

To verify the developed control method, this section provides a torsional pendulum system [56] with the adversarial input whose dynamics can be described as

$$\begin{cases} \dot{\theta} = \alpha \\ J\dot{\alpha} = u + v - Mgl\sin\theta - f_d\dot{\theta} \end{cases} \tag{52}$$

where θ and α are the angle position and the angular velocity of the pendulum, respectively, u and v are the control input and the adversarial input applied on the system, respectively, and other parameters are provided as follows:

$M = 1/3$ kg, is the mass of the pendulum;
$l = 2/3$ m, is the length of the pendulum;
$J = 4/3$ kg $\cdot$ m^2, is the rotary inertia;
$g = 9.8$ m/s^2, is the acceleration of gravity;
$f_d = 0.2$ N $\cdot$ m $\cdot$ s/rad, is the frictional factor.

Here, we define the state vector of the torsional pendulum system as $x = [\theta, \alpha] = [x_1, x_2]$, with the initial state $x_0 = [0.5, -0.5]$. Assume that only the angle position θ can be measured at the output, which is

$$y = x_1 = \theta. \tag{53}$$

Therefore, only x_1 is the measurable feedback which means $C = [1, 0]$ in this example.

The developed RL-based control method is applied to solve the problem. To recover the hidden state from the output, an observer is established based on (25) with the parameters $\mathcal{F} = [0, 1; -1, -2]$ and $L = [1, -1]^T$. The state network of the observer is chosen as a three-layer structure as $4-8-1$ (i.e., four input neurons, eight hidden neurons, and one output neuron). A critic network is also built with the structure as $4-6-1$ to estimate the performance index . The inputs for both observer and critic network are $[x_1, x_2, u, v]$. The initial learning rates of both neural networks are set to be 0.1 and are decreased by 0.05 every five time steps until they reach 0.005 and stay thereafter. The initial weights are chosen randomly within $[-0.5, 0.5]$. Let $\Lambda = I_2$, $R = I$, and $\rho = 5$, where I_n is the identity matrix with n dimensions. The control action and adversarial input are designed based on (41) and (42), respectively.

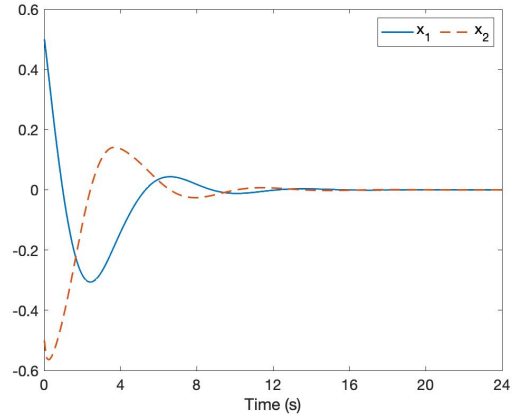

Fig. 1. The trajectories of the system states.

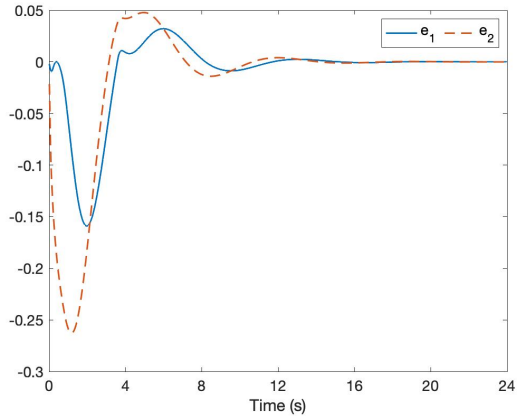

Fig. 2. The observation errors of the designed observer with $e_1 = \tilde{x}_1, e_2 = \tilde{x}_2$.

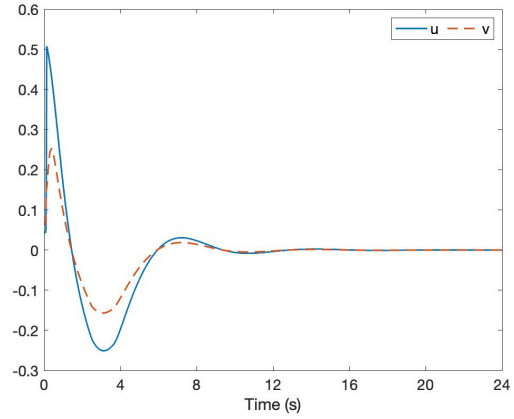

Fig. 3. The trajectories of the control input u and the adversarial input v.

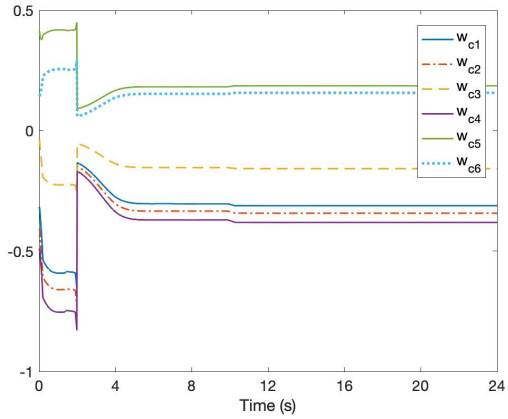

Fig. 4. The critic neural network weights updates.

The results are provided as follows. The trajectories of the system states under the developed control is provided in Fig. 1. We observe that the states can converge to the equilibrium point even in the adversarial environment. Fig. 2 provides the observation errors during the learning process. It is shown that the state errors can quickly decrease to zero and stay thereafter, which means the developed observer can identify the unknown system dynamics from the output feedback. Furthermore, both the control action u and the adversarial input v in the learning process are shown in Fig. 3. In addition, Fig. 4 shows the trajectories of the critic network weights. We can observe that the weights converge after 10s, which means the learning process is optimal.

V. Conclusion

This paper designed a RL-based optimal control method for unknown nonlinear system in an adversarial environment. Since the internal state is unavailable during the learning process, an observer was established to reconstruct the system dynamics from the output feedback. This design could also adaptively derive the control and adversarial coefficient functions and therefore reduce the computation complexity. A critic network was built to estimate the corresponding performance index. The explicit stability analysis of the designed closed-loop system was provided based on the Lyapunov construct. Finally, the numerical experiment showed the efficiency and performance of the developed control method.

References

[1] B. R. Kiran, I. Sobh, V. Talpaert, P. Mannion, A. A. A. Sallab, S. Yogamani, and P. Pérez, "Deep reinforcement learning for autonomous driving: A survey," *arXiv preprint arXiv:2002.00444*, 2020.

[2] D. Kalashnikov, A. Irpan, P. Pastor, J. Ibarz, A. Herzog, E. Jang, D. Quillen, E. Holly, M. Kalakrishnan, V. Vanhoucke, *et al.*, "Qt-opt: Scalable deep reinforcement learning for vision-based robotic manipulation," *arXiv preprint arXiv:1806.10293*, 2018.

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

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

[5] F. L. Lewis, D. Liu, and G. G. Lendaris, "Special issue on adaptive dynamic programming and reinforcement learning in feedback control," *IEEE Transactions on System, Man and Cybernetics, Part B*, vol. 38, no. 4, pp. 896–897, 2008.

[6] P. J. Werbos, "Adp: The key direction for future research in intelligent control and understanding brain intelligence," *IEEE Transactions on Systems Man and Cybernetics Part B-Cybernetics*, vol. 38, no. 4, pp. 898–900, 2008.

[7] G. G. Lendaris, "Higher level application of adp: A next phase for the control field?," *IEEE Transactions on System, Man and Cybernetics, Part B*, vol. 38, no. 4, pp. 901–912, 2008.

[8] D. Liu, Q. Wei, D. Wang, X. Yang, and H. Li, *Adaptive dynamic programming with applications in optimal control*. Springer, 2017.

[9] R. Moghadam and F. L. Lewis, "Output-feedback H_∞ quadratic tracking control of linear systems using reinforcement learning," *International Journal of Adaptive Control and Signal Processing*, vol. 33, no. 2, pp. 300–314, 2019.

[10] A. G. Barto, *Reinforcement Learning: An Introduction*. MIT press, Cambridge, MA, 1998.

[11] R. A. Brooks, "Intelligence without reason," in *Proceedings of the 12th International Joint Conference on Artificial Intelligence*, Sydney, New South Wales, Australia, 1991.

[12] R. Pfeifer and C. Scheier, *Understanding Intelligence*. MIT Press, Cambridge, MA, 1999.

[13] P. F. Christiano, J. Leike, T. Brown, M. Martic, S. Legg, and D. Amodei, "Deep reinforcement learning from human preferences," in *Advances in Neural Information Processing Systems*, 2017.

[14] J. Si, A. G. Barto, W. B. Powell, and D. W. II, eds., *Handbook of Learning and Approximate Dynamic Programming*. Wiley-IEEE, 2004.

[15] F. L. Lewis and D. Liu, eds., *Reinforcement Learning and Approximate Dynamic Programming for Feedback Control*. Wiley-IEEE, 2012.

[16] J. Si, A. G. Barto, W. B. Powell, and D. C. Wunsch, eds., *Handbook of Learning and Approximate Dynamic Programming*. IEEE Press and John Wiley & Sons, 2004.

[17] F. Lewis and D. Liu, eds., *Reinforcement Learning and Approximate Dynamic Programming for Feedback Control*. Wiley, New York, 2013.

[18] H. Zhang, D. Liu, Y. Luo, and D. Wang, *Adaptive Dynamic Programming for Control: Algorithms and Stability* . London: Springer, 2013.

[19] D. V. Prokhorov and D. C. Wunsch, "Adaptive critic designs," *IEEE Trans. on Neural Netw.*, vol. 8, no. 5, pp. 997–1007, 1997.

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

[21] Y. Jiang and Z. Jiang, "Robust adaptive dynamic programming and feedback stabilization of nonlinear systems," *Neural Networks and Learning Systems, IEEE Transactions on*, vol. 25, no. 5, pp. 882–893, 2014.

[22] Y. Fu, J. Fu, and T. Chai, "Robust adaptive dynamic programming of two-player zero-sum games for continuous-time linear systems," *Neural Networks and Learning Systems, IEEE Transactions on*, vol. 26, no. 12, pp. 3314–3319, 2015.

[23] X. Zhong, H. He, and D. V. Prokhorov, "Robust controller design of continuous-time nonlinear system using neural network," in *Proc. Int. Joint Conf. Neural Networks*, pp. 1–8, 2013.

[24] D. Wang and C. Mu, *Adaptive critic control with robust stabilization for uncertain nonlinear systems*. Springer, 2019.

[25] T. Dierks and S. Jagannathan, "Optimal control of affine nonlinear continuous-time systems using an online hamilton-jacobi-isaacs formulation," in *49th IEEE Conference on Decision and Control (CDC)*, pp. 3048–3053, IEEE, 2010.

[26] K. G. Vamvoudakis and F. L. Lewis, "Online solution of nonlinear two-player zero-sum games using synchronous policy iteration," *International Journal of Robust and Nonlinear Control*, vol. 22, no. 13, pp. 1460–1483, 2012.

[27] X. Zhong, H. He, D. Wang, and Z. Ni, "Model-free adaptive control for unknown nonlinear zero-sum differential game," *IEEE transactions on cybernetics*, vol. 48, no. 5, pp. 1633–1646, 2017.

[28] F. L. Lewis, D. Vrabie, and V. L. Syrmos, *Optimal Control*. Wiley, Hoboken, NJ, USA, 2012.

[29] K. G. Vamvoudakis, "Game-theoretic tracking control for actuator attack attenuation in cyber-physical systems," in *2016 International Joint Conference on Neural Networks (IJCNN)*, pp. 4233–4240, IEEE, 2016.

[30] C. Sun and K. G. Vamvoudakis, "Continuous-time safe learning with temporal logic constraints in adversarial environments," in *2020 American Control Conference (ACC)*, pp. 4786–4791, IEEE, 2020.

[31] L. Koçkesen and E. A. Ok, "An introduction to game theory," *University Efe A. Ok New York University July*, vol. 8, 2007.

[32] T. Başar and P. Bernhard, *H_∞ optimal control and related minimax design problems: a dynamic game approach*. Springer Science & Business Media, 2008.

[33] K. Wang, C. Mu, Y. Zhang, and W. Liu, "An approximate control algorithm for zero-sum differential games using adaptive critic technique," in *2018 37th Chinese Control Conference (CCC)*, pp. 2812–2817, IEEE, 2018.

[34] L. Wei and Z. Wu, "Recursive zero-sum stochastic differential game," in *2008 International Conference on Intelligent Computation Technology and Automation (ICICTA)*, vol. 2, pp. 998–1001, IEEE, 2008.

[35] C. Qin, H. Zhang, and Y. Luo, "Model-free adaptive dynamic programming for online optimal solution of the unknown nonlinear zero-sum differential game," in *2014 International Joint Conference on Neural Networks (IJCNN)*, pp. 3815–3820, IEEE, 2014.

[36] K. G. Vamvoudakis and F. R. P. Safaei, "Stochastic zero-sum nash games for uncertain nonlinear markovian jump systems," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pp. 5582–5589, IEEE, 2017.

[37] D. Muniraj, K. G. Vamvoudakis, and M. Farhood, "Enforcing signal temporal logic specifications in multi-agent adversarial environments: A deep q-learning approach," in *2018 IEEE Conference on Decision and Control (CDC)*, pp. 4141–4146, IEEE, 2018.

[38] D. Liu, Y. Huang, D. Wang, and Q. Wei, "Neural-network-observer-based optimal control for unknown nonlinear systems using adaptive dynamic programming," *International Journal of Control*, vol. 86, no. 9, pp. 1554–1566, 2013.

[39] M. Abu-Khalaf, F. L. Lewis, and J. Huang, "Neurodynamic programming and zero-sum games for constrained control systems," *Neural Networks, IEEE Transactions on*, vol. 19, no. 7, pp. 1243–1252, 2008.

[40] D. Liu, H. Li, and D. Wang, "H_∞ control of unknown discrete-time nonlinear systems with control constraints using adaptive dynamic programming," in *The 2012 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–6, IEEE, 2012.

[41] M. Liu, Y. Wan, F. L. Lewis, and V. G. Lopez, "Adaptive optimal control for stochastic multiplayer differential games using on-policy and off-policy reinforcement learning," *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

[42] H. Zhang, Q. Wei, and D. Liu, "An iterative adaptive dynamic programming method for solving a class of nonlinear zero-sum differential games," *Automatica*, vol. 47, no. 1, pp. 207–214, 2011.

[43] K. G. Vamvoudakis and J. P. Hespanha, "Cooperative q-learning for rejection of persistent adversarial inputs in networked linear quadratic systems," *IEEE Transactions on Automatic Control*, vol. 63, no. 4, pp. 1018–1031, 2017.

[44] T. Smith and R. Simmons, "Heuristic search value iteration for pomdps," in *Proceedings of the 20th conference on Uncertainty in artificial intelligence*, pp. 520–527, AUAI Press, 2004.

[45] J. Pineau, G. Gordon, S. Thrun, *et al.*, "Point-based value iteration: An anytime algorithm for pomdps," in *IJCAI*, vol. 3, pp. 1025–1032, 2003.

[46] H. Zhang, "Partially observable markov decision processes: A geometric technique and analysis," *Operations Research*, vol. 58, no. 1, pp. 214–228, 2010.

[47] T. Jaakkola, S. P. Singh, and M. I. Jordan, "Reinforcement learning algorithm for partially observable markov decision problems," vol. 7, p. 345, MIT Press, 1995.

[48] E. Saad, "Reinforcement learning in partially observable markov decision processes using hybrid probabilistic logic programs," *arXiv preprint arXiv:1011.5951*, 2010.

[49] B. Kiumarsi, F. Lewis, M.-B. Naghibi-Sistani, and A. Karimpour, "Optimal tracking control of unknown discrete-time linear systems using input–output measured data," *Cybernetics, IEEE Transactions on*, 2015, in press.

[50] X. Zhong and H. He, "An event-triggered adp control approach for continuous-time system with unknown internal states," *IEEE Transactions on Cybernetics*, vol. 47, no. 3, pp. 683–694, 2016.

[51] X. Zhong, Z. Ni, and H. He, "Event-triggered adaptive dynamic programming for continuous-time nonlinear system using measured input-output data," in *2015 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, IEEE, 2015.

[52] F. L. Lewis and K. G. Vamvoudakis, "Reinforcement learning for partially observable dynamic processes: Adaptive dynamic programming using measured output data," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 41, no. 1, pp. 14–25, 2011.

[53] Z. Ni, H. He, and X. Zhong, *Experimental Studies on Data-Driven Heuristic Dynamic Programming for POMDP*, ch. Frontiers of Intelligent Control and Information Processing. World Scientific Publishing, Singpore.

[54] X. Zhong, Z. Ni, Y. Tang, and H. He, "Data-driven partially observable dynamic processes using adaptive dynamic programming," in *Proc. IEEE Symposium of Adaptive Dynamic Programming and Reinforcement Learning (ADPRL)*, pp. 1–8, IEEE, 2014.

[55] H. A. Talebi, F. Abdollahi, R. V. Patel, and K. Khorasani, *Neural Network-Based State Estimation of Nonlinear Systems*. Springer, New York, 2010.

[56] B. Zhao, D. Liu, and C. Luo, "Reinforcement learning-based optimal stabilization for unknown nonlinear systems subject to inputs with uncertain constraints," *IEEE Transactions on Neural Networks and Learning Systems*, 2019.