Online Adaptive Critic Learning Control of Unknown Dynamics With Application to Deep Submergence Rescue Vehicle

XIAOYANG WANG\(^1\), XIUFEN YE\(^1\), (Senior Member, IEEE), AND WENZHI LIU\(^2\)
\(^1\)College of Automation, Harbin Engineering University, Harbin 150001, China
\(^2\)College of Information and Communication Engineering, Harbin Engineering University, Harbin 150001, China

Corresponding author: Xiufen Ye (yexiufen@hrbeu.edu.cn)

This work was supported in part by the State Key Program of the National Natural Science Foundation of China under Grant 41876100 and Grant 61633004, and in part by the National Key Research and Development Program of China under Grant 2018YFC03010102 and Grant 2017YFC0306001.

**ABSTRACT**  As a powerful tool for nonlinear systems robust controller design, robust adaptive dynamic programming (RADP) methods require initial admissible control and prior knowledge of disturbance to be effective. As the most effective approach to provide robustness to uncertainties, active disturbance attenuation (ADA), was rarely considered in RADP literatures. To combine ADA with RADP, a neural-network identifier was developed initially to approximate the plant dynamics and the imposed external disturbance. System states was extended with the approximated disturbance to establish ADA actor-critic learning. To relax the initial admissible control constraint, a novel auxiliary system were created based on the identifier dynamics. Theoretical analysis and simulations on unstable nonlinear system show that the approximated control law with respect to the auxiliary system and a newly proposed cost function itself could guarantee asymptotic stability of the original system. Simulations and comparison with other model-free control techniques demonstrated the excellent performance and robustness of the proposed method. Applicability of the proposed method was validated by applying it to trajectory tracking control of a deep submergence rescue vehicle.

**INDEX TERMS**  Adaptive dynamic programming, neural networks, robust control, underwater vehicle.

### I. INTRODUCTION
In the past decade, adaptive dynamic programming (ADP), wherein adaptive parameter identification is combined with conventional dynamic programming to solve nonlinear system optimal control problem forward-in-time, has received increasing attention in the research of adaptive and intelligent control studies [1], [2]. For complex nonlinear systems, it is difficult to derive an analytical solution to the Hamilton-Jacobi-Bellman (HJB) equation. Neural networks (NNs) and fuzzy systems are generally incorporated as intelligent components for the value approximation [3], [4]. ADP is a promising approach that provides optimal control solutions for complex tasks and has been applied effectively in robotic manipulations [4]–[6], multi-agent systems [7], [8] and power systems [9], [10]. However, the identifiers respond slowly to the parameter variations of the plant.

Vrabie et al. proposed integral reinforcement learning (IRL) algorithm to determine the solution of the HJB equation of linear [11] and nonlinear [12] systems without requiring knowledge about the state transition dynamics. Modares et al. applied experience replay technique to speed up the convergence of IRL [13]. Li et al. applied the IRL method to solve $H_\infty$ control problem for systems with unknown dynamics [14]. Palanisamy et al. proposed continuous-time $Q$ learning to solve the optimal control problem of systems with completely unknown dynamics [15]. Modares et al. extended the ADP method to optimal tracking control by augmenting the control plant with reference trajectory dynamics [16]. Wen et al. proposed a novel actor-critic RL method for nonlinear system tracking control [17]. Vamvoudakis [18] and Sahoo et al. [19] applied ADP technology for event-triggered control, which significantly improves the efficiency of in-system communications. Yang et al. provided solutions to event-triggering robust control of continuous-time nonlinear systems [20].
The abovementioned ADP methods, however, rarely considered dynamical uncertainties. Real systems are generally subject to parameter variation and external disturbance. To deal with these adverse events, ADP and RL methods can be applied to solve the $H_\infty$ control problem [20], [21], which is formulated as a two-player zero-sum game. The optimal controller minimizes the cost function, whereas a worst-case disturbance generator is used to maximize it. The $H_\infty$ optimal control method aims to find the Nash equilibrium point by solving the Hamilton-Jacobi-Isaacs (HJI) equation. Dierks et al. proposed an on-policy IRL algorithm for solving the HJI equation without requiring the drift dynamics of the control plant [16]. Using a NN-based actor-critic structure, many weights update laws have been proposed to minimize the Bellman error. The on-policy IRL method requires the imposed disturbance to be adjustable online, which is very difficult to implement in practice. Luo et al. proposed an off-policy RL method to solve the $H_\infty$ control problem of nonlinear systems [21]; however, the policy iteration cannot be implemented online. Zhang et al. presented an online algorithm for obtaining the HJI solution for discrete-time nonlinear system control [22]. Wang et al. proposed a model-free method by introducing a system dynamics identifier, in which the control policy is updated online with guaranteed system stability. However, even for off-policy methods, the $H_\infty$ solution of robust control requires that the imposed disturbance to be known, which means that an alternative disturbance observer is necessary for algorithm implementation. Robust adaptive dynamic programming (RADP) methods achieve intelligent robust control using a different perspective. In RADP, the utility function and the optimal control policy for the nominal plant are designed based on nonlinear control theory, such as robust-redesign, back-stepping, and the small-gain theorem [23]–[26]. Problem transfer methods are usually used. Dipak et al. designed a controller to optimize a cost function, which includes penalties on constraint control efforts and a maximum bound on uncertainties [25]. Ding et al. found that weighting the upper bound of uncertainty with a scalar larger than the maximal eigenvalue of a control effort weighting matrix guarantees the uniformly ultimately bounded (UUB) stability of the uncertainty system [26]. Jiang et al. used a model-free method for cost function estimation and added a small gain to the control policy, which has proven robustly optimal [23]. The RADP method has also been applied to the decentralized optimal control of large-scale systems [27] and the output feedback control of interconnected systems [28]. RADP methods aim to guarantee the stability of the uncertain plant by approximating a value function for the nominal plant. In this robust optimal design, only the upper bound of the uncertainty is required. To approximate the robust optimal policy, ADP methods require that the system meets two necessary conditions: the initial admissible control condition and the persistence of excitation condition. These two conditions are generally difficult to satisfy and narrow the extent of ADP application. To solve this problem, Dierks et al. designed a single estimator based control scheme for solving the HJI equation without requiring initial admissible control [29]. Chowdhary et al. introduced a concurrent learning technique to relax the persistence of excitation condition [30]. Yang et al. proposed a robust control strategy for nonlinear systems subjected to unmatched uncertainties [31]; they developed a new critic learning algorithm that relaxes both conditions. To review, the RADP method requires a nominal model and the knowledge about the structure of the uncertainties. Both requirements are difficult to attain in many real applications. Some ADP-based $H_\infty$ methods are completely model-free and are even capable of being applied online. However, an initial admissible control law and disturbance law are necessary to guarantee convergence of the system states in the learning process. It is very difficult to satisfy this requirement when the plant dynamics are completely unknown. Therefore, an online model-free intelligent critic control design method is needed that would remove the requirements of initial admissible control and the persistence of excitation conditions.

In this study, an ADP based model-free robust optimal control scheme was developed for a class of continuous-time non-affine nonlinear systems with unmatched uncertainty. A NN system identifier was developed initially to reconstruct the system dynamical model along with uncertainty effects. Based on the identifier dynamics, an auxiliary system was constructed, and a critic network was employed to approximate a newly designed value function using currently observed system states and the approximated disturbance. A concurrent learning technique was integrated with the critic update law to relax the requirement of the persistence of excitation condition. Here, we show that for a given set of critic weights, the approximated optimal controller of the auxiliary system guarantees the asymptotic stability of the uncertain system. We also show that when applied to an original unknown plant, the optimal control for the auxiliary system achieves optimality for a specified value function. Using Lyapunov’s method, all signals in the closed-loop system are stable in the sense of UUB. The contributions of this paper are summarized as follows:

1) A novel NN-based non-affine dynamics identifier is introduced. In addition to approximating the system dynamics, the external disturbance is also approximated for active disturbance attenuation. The error dynamics of the proposed identifier prove to be asymptotically stable; no prior knowledge about the external disturbance is required.

2) A critic learning based robust control scheme is presented. The designed control system is robust not only to external disturbances, but also to the identifier approximating error and the critic network value prediction error. Thus, the proposed method guarantees system stability during the training process, without requiring an initial admissible controller. The proposed method is completely model-free and can be implemented online for critic learning control.
3) Inspired by the research on ADP-based linear quadratic tracking problems [32], [33], we extended the system states with an approximated external disturbance and introduced an intermedia auxiliary system to generate data for robust critic learning. To the best of our knowledge, this is the first research article that combines robust design and active disturbance attenuation in an ADP-based control scheme.

The remainder of this paper is organized as follows. In Section II, the robust control problem is formulated, and the aim of critic design is introduced. In Section III, the system identifier is presented with stability analysis. In Section IV, we describe the auxiliary system and discuss the associated critic learning rule. Section V illustrates the main results of the new robust critic design with proof of stability and optimality. Simulation results are given in Section VI to illustrate the effectiveness and applicability of the proposed method. Finally, conclusions are drawn in Section VII.

II. PROBLEM FORMULATION

Consider the following nonlinear non-affine system subjected to an unmatched disturbance

\[
\dot{x}(t) = f(x(t), u(t)) + w(t)
\]

where \( x \in \Omega \subseteq \mathbb{R}^n \) is the state vector, and \( u \in \mathbb{R}^m \) is control input vector. \( f(x, u) \in \mathbb{R}^n \) is an unknown continuous differentiable function with respect to \( x \) and \( u \). \( w \in \mathbb{R}^n \) is an uncertain function representing the disturbance effects. \( x_0 = x(0) \) denotes the initial state system.

Assumption 1: System (1) is controllable and function \( f(x, u) \) is locally Lipschitz continuous on the set \( \Omega \). Meanwhile, \( f(0, 0) = 0 \), which indicates that when \( u(t) = 0 \) and \( w(t) = 0 \), \( x = 0 \) is an equilibrium point of system (1).

The goals of this research are summarized below:

(i) Design a dynamics identifier to approximate the unknown system dynamics along with the external disturbance.

(ii) Concerning the existence of identification error and external disturbance, design an adaptive critic control law such that system (1) is asymptotically stable for any intermedia control policy applied during the on-policy training process. (iii) Find a value function such that the controller obtained in (ii) is a solution to the optimal control problem defined as

\[
\min J(x(t), u(t)) = \int_{t}^{\infty} r(x(s), u(s))ds
\]

\[\text{s.t. } \dot{x}(t) = f(x(t), u(t)) + w(t) \]

Based on the work of [36], the robust control problem (2) can be solved by solving the optimal control problem of an auxiliary system. Since the identifier is a certain dynamical system, we construct an auxiliary system based on the identifier dynamics.

III. IDENTIFIER DESIGN WITH STABILITY ANALYSIS

In this section, a three-layer NN is applied to reconstruct the non-affine system dynamics based on the system input-output data. Using the extended state observer theory [38], we designed an observer to estimate the disturbance and uncertainty using only the measurable signals. Let the number of neurons in the hidden layer be denoted by \( l_m \). The dynamics of system (1) can be rewritten as

\[
\dot{x} = \omega_m^T x + \epsilon_m + w \tag{3}
\]

where \( w \in \mathbb{R}^n \) is the external disturbance added to the system, \( \omega_m \in \mathbb{R}^{l_m \times n} \) is the ideal weight between the hidden and output layer, \( \sigma(\cdot) \) is a continuously differentiable and monotonically increasing activation function \( z = v_m^T x \in \mathbb{R}^m \) is the input signal of the hidden layer, where \( v_m \in \mathbb{R}^{(n+m) \times m} \) is the ideal weight between the input and the hidden layer. \( z = [x^T, u^T]^T \in \mathbb{R}^{n+m} \) is the input vector, and \( \epsilon_m \in \mathbb{R}^n \) is the reconstruct error. The differentiable activation function is Lipschitz continuous and for any \( \xi_\alpha, \xi_\beta \in \mathbb{R}^n \), there exists a constant \( \lambda_0 > 0 \) such that the following inequality holds

\[
\| \sigma(\xi_\alpha) - \sigma(\xi_\beta) \| \leq \lambda_0 \| \xi_\alpha - \xi_\beta \| \tag{4}
\]

Note that \( \sigma(\cdot) \) applies the same operation to every element of its input.

For simplicity, the input-hidden weight matrix \( v_m \) is maintained constant and only the hidden-output weight matrix is tunable. We randomly initialize \( v_m \) and keep it static during the training process. The NN identifier dynamics is represented by:

\[
\dot{\hat{x}} = \hat{\omega}_m^T \sigma(\hat{z}) + \hat{w} \tag{5}
\]

where, \( \hat{\omega}_m \) is the current estimation of the ideal hidden-output weight matrix \( \omega_m \), and \( \hat{z} = v_m^T [\hat{x}, u]^T \) denotes the input signal of the hidden layer. Let \( \bar{x} = x - \hat{x} \) and \( \bar{w} = w - \hat{w} \) represents the states and disturbance approximate error respectively. Then \( x \) and \( w \) can be approximated by the following observer

\[
\begin{align*}
\dot{\hat{x}} &= \hat{\omega}_m^T \sigma(\hat{z}) + \hat{w} + \eta_1 \bar{x} \\
\dot{\hat{w}} &= \eta_2 \bar{x}
\end{align*}
\tag{6}
\]

where \( \eta_1 > 0 \) and \( \eta_2 \leq 1 \) are real scalar design parameters. As described in (6), the external disturbance is approximated by integrating the states approximating error and multiplying the designed parameter \( \eta_2 \). The current approximated system states and system inputs are concatenated as the input vector of the NN. Then, the NN output, the approximated disturbance and the approximating error feedback term \( \eta_1 \bar{x} \) are summed together as an approximation of the time derivative of system states. We obtain the approximated system states, disturbance and NN weights by integrating (6) and (10).

Scheme of the NN identifier is displayed in Fig. 2.

Using (3) and (6), the error dynamics of the proposed observer are given by

\[
\begin{align*}
\dot{\hat{x}} &= \omega_m^T (t) \sigma(\hat{z}) + \omega_m^T (t) (\sigma(\hat{z}) - \sigma(\hat{z})) + \bar{w} - \eta_1 \bar{x} + \epsilon_m \tag{7a} \\
\dot{\hat{w}} &= \bar{w} - \eta_2 \bar{x} \tag{7b}
\end{align*}
\]

Note that when the number of nodes in the hidden layer \( l_m \) is large enough, the NN identification error can be arbitrarily small. Moreover, (7a) indicates that \( \epsilon_m \) is closely linked
with $\tilde{x}$, (7b) indicates that $\dot{\tilde{w}}$ is closely linked with $\tilde{w}$. It is reasonable to assume that $\varepsilon_m$ is bounded by a function of $\tilde{x}$; $\tilde{w}$ is bounded by a function of $\tilde{w}$, and $\tilde{\omega}_m$ is bounded by a function of $\tilde{\omega}_m$. We give two assumptions often used in the ADP literature [22], [34], [35], which are helpful for analyzing the stability of the proposed identifier.

Assumption 2: According to (7), the system dynamics identification error $\varepsilon_m$, the time derivative of the external disturbance $\dot{\tilde{w}}$, and the disturbance approximating error $\tilde{\omega}_m$ have the following relationships

$$
\varepsilon_m^T \varepsilon_m \leq \lambda \varepsilon_n \tilde{x}^T \tilde{x}
$$

and

$$
\dot{\tilde{w}}^T \tilde{w} \leq \lambda \tilde{w}^T \tilde{w}
$$

where $\lambda \varepsilon_n > 0$, $\lambda \tilde{w} > 0$ and $\lambda \tilde{\omega}_m > 0$ are constant scalars.

Assumption 3: The ideal weights of NN are bounded such that $\|\tilde{\omega}_m\| \leq \lambda \tilde{w}_n$ and $\|\varepsilon_m\| \leq \lambda \varepsilon_n$, where $\lambda \tilde{w}_n$ and $\lambda \varepsilon_n$ are constant scalars.

Theorem 1: Define a scalar parameter $\dot{\tilde{\omega}}_m = \tilde{\omega}_m^T \tilde{\omega}_m + (1 - \eta_2)\tilde{x}^T \tilde{x}$ and a positive definite matrix $\Lambda = \eta_1 \tilde{x}^T \tilde{x}$.

Using the above inequality and Assumption 2 one gets

$$
\tilde{\omega}_m^T \tilde{\omega}_m \leq \lambda \varepsilon_n \tilde{x}^T \tilde{x}
$$

the system states approximating error dynamics is asymptotically stable, which means $\tilde{x}$ would converge to zero.

Proof: Choose a Lyapunov candidate of the form

$$
L_1 = \lambda L_{11} + L_{21},
$$

where

$$
L_{11}(t) = 2\tilde{x}^T \tilde{x} + \tilde{w}^T \tilde{w}, \quad L_{21}(t) = tr \left( \tilde{\omega}_m \Lambda \tilde{\omega}_m \right)
$$

Taking the derivative of $L_{11}$ along the trajectory of the error dynamics (7), we obtain

$$
\dot{L}_{11} = 2\tilde{x}^T \tilde{x} + 2\tilde{w}^T \tilde{w} + 2\tilde{\omega}_m^T \Omega_1 \tilde{\omega}_m + 2\tilde{\omega}_m^T \Omega_2 \tilde{\omega}_m + 2\tilde{\omega}_m^T \omega_m + 2\tilde{\omega}_m^T \varepsilon_m + 2\tilde{\omega}_m^T \tilde{\epsilon}_m + 2\tilde{\omega}_m^T \varepsilon_m + 2\tilde{\omega}_m^T \tilde{\epsilon}_m
$$

Using the adaptive criterion (10) and trace operation property $tr(AB) = tr(BA) = BA$, considering that the ideal weights are constants, we have:

$$
\dot{L}_{12} = -2\tilde{x}^T \tilde{\omega}_m^T \sigma(z)
$$

Along (14) and the inequalities in Assumption 2 one gets

$$
\dot{L}_1 \leq \mu \tilde{x}^T \tilde{x}
$$

By selecting suitable $\eta_1$ and $\eta_2$ such that $\mu < 0$, one gets $L_1 \leq 0$. Based on the standard Lyapunov extension theorem, the identifier error dynamics is asymptotically stable.

Remark 1: The proposed system dynamics identifier is asymptotically stable with the existence of a disturbance. It is also a high-performance disturbance observer that requires no prior knowledge of the external disturbance. With these advantages, the identifier is very applicable and the proposed control method achieves novel external disturbance attenuation performance using the approximated disturbance.

IV. ADAPTIVE CRITIC CONTROL OF THE AUXILIARY SYSTEM

In this section, based on the identifier dynamics, we present an auxiliary system and the associated HJB equation. A critic NN is used to approximate a solution for the HJB equation.

A. HJB EQUATION FOR THE AUXILIARY SYSTEM

Since the measured system state is a part of the observer’s inputs, we define the auxiliary system as

$$
\begin{align*}
\dot{\hat{x}} &= \hat{\omega}_m^T \sigma(z) + \hat{w} - \eta_1 \tilde{x} + \eta_1 v_1(x) \\
\dot{\hat{w}} &= -\eta_2 \tilde{x} + \eta_2 v_2(x)
\end{align*}
$$

where the auxiliary control inputs $v_1$ and $v_2$ are used to handle the model uncertainty and external disturbance. The augmented system states and the augmented control input are defined as $s = [\tilde{x}^T, \tilde{w}^T]^T$ and $v = [v_1, v_2]^T$ respectively. Thus, (18) can be rewritten as

$$
\dot{s} = \Psi s + f(z) + Bv
$$

where

$$
\begin{align*}
\Psi &= \begin{bmatrix} -\eta_1 I_n & I_n \\ -\eta_2 I_n & 0 \end{bmatrix} \\
B &= \begin{bmatrix} \eta_1 I_n & 0 \\ 0 & \eta_2 I_n \end{bmatrix} \\
f(z) &= \begin{bmatrix} \omega_m^T \sigma(z) \\ 0 \end{bmatrix} \\
v &= [v_1(x), v_2(x)]^T
\end{align*}
$$
Associated with the auxiliary system (19), the value function is described by

\[
V(x) = \int_t^\infty U(x, \dot{x}, u, v, \tau) d\tau
\]

where

\[
U(x, \dot{x}, u, v, \tau) = \Gamma(\dot{x}(\tau), x(\tau)) + Q(\dot{x}(\tau)) + u^T(\tau)u(\tau) + \alpha v^T(\tau)v(\tau)
\]

(21)

With \( \Gamma(x) = \beta_x \dot{x}^T \ddot{x} + \beta_z x^T \ddot{x} + 2 \| \dot{x} \|^2 \), \( \beta_x, \beta_z \) and \( \alpha \) are positive design parameters. The optimal value function is defined as [37]

\[
V^*(\dot{x}, \dot{\hat{w}}) = \min_{u, v \in A(\hat{\Omega})} V^{u,v}(\dot{x}, \dot{\hat{w}})
\]

(22)

where \( A(\Omega) \) is the set of admissible controls defined on \( \Omega \). According to Abu-Khalaf and Lewis [39], the optimal value function satisfies the following Lyapunov function

\[
\nabla_x V^*(s)^T (\Psi s + B v + f(\dot{x}, u)) + U(x, \dot{x}, u, v) = 0
\]

(23)

Define the Hamiltonian with respect to \( V^*(x) \), \( u \) and \( v \) as

\[
H(x, V^*_x, u, v) = \nabla_x V^*(x)^T (\Psi s + B v + f(\dot{x}, u)) + \Gamma(x) + Q(x) + u^T u + \alpha v^T v
\]

(24)

Then the optimal value function can be obtained by solving

\[
\min_{u \in A(\hat{\Omega})} H(x, V^*_x, u, v) = 0
\]

(25)

with \( V^*(0) = 0 \). The corresponding optimal control laws that minimize the Hamiltonian in (22) are given by

\[
u^* = -\frac{1}{2} \hat{g}^T(\ddot{x}) V^*_x
\]

(26a)

\[
v^* = -\frac{1}{2\alpha} B^T V^*_x
\]

(26b)

where \( \hat{g}(\ddot{x}) = \nabla_u \hat{\omega}_m^T \sigma(\ddot{x}) \) is the approximated system input dynamics. Combining the augmented optimal control laws in (26) with the Hamiltonian in (24), the HJB equation can be rewritten as

\[
(V^*_x)^T \hat{\omega}_m^T \sigma(\ddot{x}) + V^*_x W + V^*_x B [\ddot{x}, \dot{x}]^T + \Gamma(x) + Q(x) + \| u^* \|^2 - \alpha \| v^* \|^2 = 0
\]

(27)

\[
\dot{\hat{w}}(t) = v^*_m[\ddot{x}, u^*]^T.
\]

In the following sections, the subscripts of \( \ddot{x} \) and \( \hat{w} \) denote the source of the control input signal.

**Remark 2:** This newly proposed auxiliary system has two main advantages: 1) The auxiliary system constructs the disturbance dynamics model, which removes the requirement of prior knowledge about the disturbance. 2) The auxiliary system has a complete known input dynamics. Thus, the actor network is unnecessary and the inner loop of the policy iteration can be removed to improve the critic learning efficiency.

**B. APPROXIMATE SOLUTION TO THE HJB EQUATION**

It is difficult to compute the analytical solution of the HJB equation (27). In many studies, policy iteration procedures are employed to solve the HJB equation offline. An initial admissible control is generally required to guarantee a bounded value function. In this section, we develop a learning algorithm that updates the critic NN in an online manner. Convergence and stability analysis of the learning rule will be discussed in subsequent sections.

According to the NN universal approximation property, the optimal value function can be represented by a NN as:

\[
V^*(s) = \omega_c^T \sigma_c(s) + \epsilon_c(s)
\]

(28)

where \( \omega_c \in \mathbb{R}^{n_c} \) is the unknown ideal network weights, \( \sigma_c(s) \) represents the input feature vector obtained by performing a specific type of math operation to the input state vector, \( n_c \) is the number of neurons in the hidden layer, and \( \epsilon_c(s) \in \mathbb{R}^n \) is the NN reconstructing error.

Differentiating \( V^*(x) \) with respect to the augmented state, it follows that

\[
V^*_x = \nabla_x \sigma_c^T(s) \omega_c + \nabla_x \epsilon_c(s)
\]

(29)

The corresponding optimal control laws are

\[
u^* = -\frac{1}{2} \hat{g}^T(\ddot{x}) V^*_x
\]

(30a)

\[
v^* = -\frac{1}{2} B^T \nabla_x \sigma_c^T(s) \omega_c + \epsilon_v^*(s)
\]

(30b)

where

\[
\epsilon_v^*(x) = -(1/2) \hat{g}^T(\ddot{x}), 0m \nabla \epsilon_c(s)
\]

\[
\epsilon_v^*(x) = -(1/2) B^T \nabla \epsilon_c(s)
\]

Since the ideal weights are unknown, we introduce a critic network formulated as follows to approximate the value function:

\[
\hat{V}(x) = \hat{\omega}_c^T \sigma(\ddot{x})
\]

(31)

where \( \hat{\omega}_c \) is the approximated NN weights. Meanwhile, the approximated optimal control policy can be formulated as

\[
\hat{u}(s) = -\frac{1}{2} \hat{g}^T(\ddot{x}) \nabla \sigma_c^T(s) \hat{\omega}_c
\]

(32a)

\[
\hat{v}(s) = -\frac{1}{2} B^T \nabla \sigma_c^T(s) \hat{\omega}_c
\]

(32b)

For a control policy \( u \) and \( v \), by applying the NN expression to the Hamiltonian we have:

\[
H(\dot{x}, V^*, u, v) = \omega_c^T \nabla \sigma(\ddot{x}) \times (\Psi s + B v + f(\dot{x}, u)) + \Gamma(\ddot{x}) + Q(\ddot{x}) + u^T u + \alpha v^T v
\]

(33)

According to (33) and the Lyapunov function (23), we have:

\[
H(\dot{s}, V^*, u, v) = - (\nabla \epsilon_c(\ddot{x}))^T (\Psi s + B v + f(\dot{x}, u)) = \epsilon_H
\]

(34)
Replacing $\omega_c$ in (34) with $\hat{\omega}_c$, we derive the approximated Hamiltonian as:

$$\hat{H}(s, \hat{V}, u, v) = \omega_c^T \nabla \sigma_c(\hat{x}) (\Psi s + Bv + f(\hat{x}, u))$$

$$+ \Gamma(\hat{x}) + Q(\hat{x}) + u^T u + \alpha \nabla v$$

(35)

Define the critic approximate error as $e_c = \hat{H}(s, \hat{V}, u, v)$. The relationship between the ground truth and approximate Hamiltonian is given by

$$e_c = -\omega_c^T \nabla \sigma_c(\hat{x}) (\Psi s + Bv + f(\hat{x}, u)) + e_H$$

(36)

The objective of critic learning is to find the critic weights that minimize the square error index $E_c = 0.5 e_c^2$. To learn the optimal control policy online, we propose an on-policy learning algorithm. In the training stage, the approximated control policies are used. Let the critic weights gradient be $\phi = \nabla \sigma_c(s)(\Psi s + Bv + f(\hat{x}, u)) \in \mathbb{R}^l$. Then the critic weights vector can be adjusted by:

$$\dot{\hat{\omega}}_c = -\frac{\alpha_c}{1 + \phi^T \phi} \left( \frac{\partial E_c}{\partial \hat{\omega}_c} - \sum_{j=1}^{m} \frac{\alpha_c \phi_j}{(1 + \phi_j^T \phi_j)^2} e_{cj} \right)$$

(37)

where $\alpha_c$ is the learning rate, $m$ is the length of the memory reserved for concurrent learning, and $(1 + \phi^T \phi)^2$ is the gradient normalization term. The approximating error of the critic network is defined as

$$\hat{\omega}_c = \omega_c - \hat{\omega}_c$$

(38)

Noting that $\dot{\hat{\omega}}_c = -\dot{\hat{\omega}}_c$, based on the critic weights update criteria (37), the error dynamics of the critic weights is given by

$$\dot{e}_c = -\frac{\alpha_c \phi \phi^T}{(1 + \phi^T \phi)^2} + \sum_{j=1}^{m} \frac{\alpha_c \phi_j \phi_j^T}{(1 + \phi_j^T \phi_j)^2} e_{cj}$$

(39)

A signal flowchart of the proposed robust control scheme is depicted in Fig. 1. As shown in the figure, the identifier approximates the system states, the external disturbance and the system dynamics using system input-output data. The Hamiltonian (35) is computed based on the auxiliary system dynamics (6), which is driven by the approximated system states and disturbance. The utility function (21) and the approximated value function gradient (29) are computed online as necessary components for constructing the Hamiltonian. Then, the critic network is updated by (37) using the concurrent learning technique to minimize the value of the obtained Hamiltonian. Finally, with the converged critic weights and the identifier NN weights, the optimal control law can be derived by (32a).
hold for all \( x \in \Omega \), where \( W_k(x) (k = 1, 2, 3) \) are positive-definite functions, then the system is asymptotically stable and convergent with respect to \( x = 0 \).

**Theorem 2:** Consider the auxiliary system (16) and the value function (17). Let Assumptions 1-5 hold. Choose \( \alpha \leq 1/3 \) and \( \Gamma (\hat{x}, \hat{x}) = \beta_1 \hat{x}^{T} \hat{x} + \beta_2 \hat{x}^{T} \hat{x} + (1/\eta_1^2) \hat{w}_M (\hat{x}) \), where

\[
\begin{align*}
\beta_1 &= \frac{4}{\eta_1^2} \left( \frac{2}{\lambda^2} \left( 1 + \frac{2}{\lambda^2} \lambda_2^2 \lambda^2 \right) + \frac{2}{\eta_1^2} \lambda_2^2 \lambda_2^2 \lambda^2 \right) + 1 \\
\beta_2 &= \frac{4}{\eta_1^2} \left( \lambda^2 + \lambda_2^2 + \lambda_2^2 \right) .
\end{align*}
\]

(40)

\( \hat{w}_M (\hat{x}) \) is a function of \( \hat{x} \) such that \( \hat{w}_M (\hat{x}) \geq \hat{w}^T (t) \hat{w}(t) \) for all \( t > 0 \). Then, if the optimal auxiliary control given in (26b) satisfies

\[
\| v^* (x(t)) \|^2 \leq Q (x(t))
\]

(41)

for any critic weights in the set \{ \( \hat{w} \) \}, the control law given by (32a) guarantees the asymptotic stability of the system (1).

**Proof:** The optimal value function given in (22) is a positive-definite function defined on \( \Omega \). For a positive-definite function, there exist two class \( \kappa \) functions \( y_1(\cdot) \) and \( y_2(\cdot) \) such that

\[
y_1 (\| x \|) \leq V^* (x) \leq y_2 (\| x \|)
\]

(42)

for each \( x \in \Omega \), the class \( \kappa \) functions are positive definite. Choose the optimal value function as the Lyapunov function and let \( W_1 (x) = y_1 (\| x \|) \), and \( W_2 (x) = y_2 (\| x \|) \). Differentiating \( V^*(x) \) along the trajectory of the original system, we obtain

\[
\begin{align*}
V^* &= (V^*)^T [\dot{x}, \dot{w}]^T \\
&= V_1 \dot{w} + V_2 w + V_3 \dot{x} + V_4 \dot{x}^T \dot{x} + V_5 \dot{x} \dot{w} + V_6 \dot{x}^T \dot{w} + V_7 \dot{x}^T w + V_8 \dot{w}^T w \\
&= \alpha \dot{w} + \alpha \dot{x} + \alpha \dot{x}^T \dot{x} + \alpha \dot{x}^T w + \alpha \dot{w}^T w \\
&= \beta_1 \dot{x}^{T} \dot{x} + \beta_2 \dot{x}^{T} \dot{x} + (1/\eta_1^2) \dot{w}_M (\dot{x})
\end{align*}
\]

(43)

where \( \dot{w} = \sigma (\dot{x}, \dot{w}) \). From (26) and (27) we have

\[
\begin{align*}
(V^*)^T \alpha \dot{w} (\dot{x}) &= -\Gamma (\dot{x}) - \dot{Q} (x) - \| u \|^2 + \alpha \| v \|^2 \\
&- V_9 B \left[ \dot{x}^{T}, \dot{x}^{T} \right] - V_1 \dot{w}
\end{align*}
\]

(44)

Substituting (44) into (43) and letting \( \dot{\sigma} \) denote \( \sigma (\dot{x}, \dot{w}) \), we have

\[
\begin{align*}
\dot{V}^* &= -\Gamma (\dot{x}, \dot{x}) - Q (x) - \| u \|^2 + \alpha \| v \|^2 - V_1 \dot{w} \left[ \dot{x}^{T}, \dot{x}^{T} \right] \\
&+ V_9 B \left[ \dot{x}^{T}, \dot{x}^{T} \right] - V_9 B \Delta_m \\
&= -\Gamma (\dot{x}, \dot{x}) - Q (x) - \| u \|^2 + \alpha \| v \|^2 \\
&- V_9 B \left[ \dot{x}^{T}, \dot{x}^{T} \right] + V_9 B \Delta_m \\
&= -\Gamma (\dot{x}, \dot{x}) - Q (x) - \| u \|^2 + \alpha \| v \|^2 \\
&+ 2 \alpha v^*_T \dot{w} \left[ \dot{x}^{T}, \dot{x}^{T} \right] - 2 \alpha v^*_T \Delta_m
\end{align*}
\]

(45)

where

\[
\Delta_m = B^{-1} \left[ \sigma^T \dot{w} + \alpha \dot{w} \right] + \beta_1 \dot{x}^{T} \dot{x} + \beta_2 \dot{x}^{T} \dot{x} + (1/\eta_1^2) \dot{w}_M (\dot{x})
\]

Applying Young’s inequalities \( 2a^T b \leq \| a \|^2 + \| b \|^2 \) we have

\[
2 \alpha v^*_T \dot{w} \left[ \dot{x}^{T}, \dot{x}^{T} \right] \leq \alpha \| v^* \|^2 + 2 \alpha \| \dot{x} \|^2
\]

and

\[
-2 \alpha v^*_T \Delta_m \leq \alpha \| v^* \|^2 + \alpha \| \Delta_m \|^2
\]

(46)

According to Assumptions 4 and 5

\[
\| u - u^* \|^2 = \alpha \| \hat{w}^T \hat{w} \|^2 = \beta_1 \dot{x}^{T} \dot{x} + \beta_2 \dot{x}^{T} \dot{x} + (1/\eta_1^2) \| \dot{w}_M \|^2 = \Gamma (\dot{x}, \dot{x})
\]

(49)

Combining (49) with (45), we obtain

\[
\dot{V}^* \leq -3 \alpha \left( Q (x) - \| v \|^2 \right) + (1 + 3 \alpha) \| \dot{w}_M \|^2.
\]

(50)

with \( 0 < \alpha \leq 1/3 \) and condition (41) holds, we have \( \rho > 0 \). Meanwhile, we have \( -3 \alpha Q (x) - \| v \|^2 \leq 0 \) and \( (2 \alpha - 1) \dot{x}^{T} \dot{x} \leq 0 \). Finally, let \( \rho = 3 \alpha - 1 \) one gets

\[
\dot{V}^* \leq -\rho Q (\hat{x})
\]

(51)

Since \( \rho Q (\hat{x}) \geq 0 \), let \( W_3 (x) = \rho Q (\hat{x}) \). Based on (42) and Lemma 1 we can conclude that by applying the approximate control policy, system (1) is asymptotically stable. This completes the proof.

**Remark 3:** Unlike [20] and [40], where identifier and controller are designed independently, the proposed critic control design aims to provide robustness to the identifier approximating error and the critic approximating error. The control system is guaranteed to be asymptotically stable even before the learning processes have converged. Hence the requirement of initial admissible control is relaxed.
Remark 4: Condition (41) cannot be verified directly in [31]. Instead, this condition is verified via numerical simulations. However, observing (32b), \(|v^*|\) can be adjusted to satisfy condition (41) by selecting \(\eta_1\) and \(\eta_2\) of matrix \(B\) in (19). This is a significant advantage of the proposed method.

In what follows, we will show that control law (30a) can minimize a specific value function given by

\[
J(x) = \int_0^\infty \left( \Phi(\dot{x}(\tau)) + u^*(x(\tau)) \right) u^*(x(\tau)) d\tau
\]

(52)

where

\[
\Phi(\dot{x}) = \Gamma(x) + Q(x) - \alpha \|v^* - \dot{x} - \Delta_m\|^2 + \alpha \|\dot{x} + \Delta_m\|^2
\]

Lemma 2: If \(\alpha \leq 0.5\), \(\Phi(\dot{x})\) is a positive definite function.

Proof: Using Young’s inequality, we have

\[
\zeta = \alpha \|v^* - \dot{x} - \Delta_m\|^2 - \alpha \|\dot{x} + \Delta_m\|^2
\leq 2\alpha \|v^*\|^2 + \alpha \|\dot{x} + \Delta_m\|^2
\leq 2\alpha \|v^*\|^2 + 2\alpha \|\dot{x}\|^2 + 2\alpha \|\Delta_m\|^2
\]

(53)

With \(\alpha \leq 0.5\), we have

\[
\zeta \leq \|v^*\|^2 + \|\dot{x}\|^2 + \|\Delta_m\|^2
\]

(54)

and

\[
\Phi(\dot{x}) \geq \Gamma(x) - \|\Delta_m\|^2 + Q(x) - \|v^*\|^2 - \|\dot{x}\|^2
\]

(55)

With (41) and (49), \(\Phi(\dot{x}) \geq 0\). This completes the proof.

Theorem 3: If \(\alpha \leq 0.5\), the controller given in (30a) is the solution to the optimal control problem defined by system (1) and value function (52).

Proof: The Hamiltonian with respect to the newly defined value function \(J(x)\) is formulated as

\[
\dot{H}(x, J, u^*, v^*) = \Phi(\dot{x}) + (u^*)^T u^*
\]

\[
+ J^T \omega_m^T \sigma (\dot{x}) + J^T \epsilon_m
\]

\[
+ J^T \omega_n^T \omega (\dot{x}) + J^T \epsilon_n
\]

(56)

Let \(J(x) = V^*(x)\), and substitute the first two formulas of (44) into (56), it follows that

\[
\dot{H}(x, J, u^*, v^*) = \Phi(\dot{x}) - \Gamma(\dot{x}) - Q(\dot{x}) + \alpha \|v^*\|^2 - 2\alpha v^T \dot{x}
\]

\[
+ V^T (\omega_m^T \sigma (\dot{x}) + \epsilon_m + \tilde{w}) + V^T \omega_n^T \omega (\dot{x})
\]

\[
= \Phi(\dot{x}) - \Gamma(\dot{x}) - Q(\dot{x}) + \alpha \|v^*\|^2
\]

\[
- 2\alpha v^T (\dot{x} + \Delta_m) + \alpha \|\dot{x} + \Delta_m\|^2 - \alpha \|\dot{x} + \Delta_m\|^2
\]

\[
= \Phi(\dot{x}) - \Gamma(\dot{x}) - Q(\dot{x}) + \alpha \|v^* - \dot{x} - \Delta_m\|^2
\]

\[
- \alpha \|\dot{x} + \Delta_m\|^2
\]

\[
= 0
\]

(57)

Therefore, \(J(x) = V^*(x)\) and \(u = u^*\) make up the solution to the HJB equation (56); furthermore the control law given in (23a) is the solution of the optimal control problems (1) and (52). This completes the proof.

B. Stability Analysis

We present the stability analysis results for the proposed on-policy critic learning process via Lyapunov’s method.

Theorem 4: Consider the auxiliary system (5) and the HJB equation. Let Assumption 1-4 hold. The approximate optimal control laws \(\hat{u}\) and \(\hat{v}\) are given by (32a) and (32b), respectively. Let the critic network weights \(\hat{\omega}_c\) be tuned by (37). The state system \(x,\) state approximate error \(\hat{x},\) and the critic weight error \(\hat{\omega}_c\) are UUB, respectively, by

\[
B_\chi = \begin{bmatrix}
\alpha_c e_\tau^2 + \alpha_c \sum_{j=1}^{n_c} e_{c,j}^2 \\
2 (1 - 3\alpha) \sum_{j=1}^{n_c} e_{c,j}^2
\end{bmatrix}

B_{\hat{\omega}_c} = \begin{bmatrix}
\alpha_c \hat{\omega}_c \sum_{j=1}^{n_c} e_{c,j}^2 \\
\alpha_c \hat{\omega}_c \sum_{j=1}^{n_c} e_{c,j}^2
\end{bmatrix}
\]

(58)

Proof: Choose a Lyapunov function candidates composed of three terms, \(L_2 = L_21(t) + L_22(t) + L_23(t)\), where

\[
L_21(t) = V^*(s(t)) , L_22(t) = L_1(t)
\]

\[
L_23(t) = \frac{1}{2} \hat{\omega}_c (t) \hat{\omega}_c (t)
\]

According to the theorems proposed in the Main Results section, the time derivatives of the terms \(L_21\) and \(L_22\) are respectively

\[
\dot{L}_21(t) \leq -\rho Q(x(t))
\]

\[
\dot{L}_22(t) = \dot{L}_1(t) \leq \mu \bar{x} \bar{x}^T \bar{x}
\]

(59)

Consider (39). The time derivative of \(L_23\) is given by

\[
\dot{L}_23(t) = \frac{\hat{\omega}_c^T (t) \dot{\hat{\omega}}_c (t)}{2}
\]

\[
\leq -\hat{\omega}_c \left( \frac{\alpha_c \hat{\omega}_c (t) \sum_{j=1}^{m} \alpha_c \phi^T \phi_j}{(1 + \phi^T \phi_j)^2} + \sum_{j=1}^{m} \frac{\alpha_c \phi^T \phi_j}{(1 + \phi^T \phi_j)^2} \right) \hat{\omega}_c
\]

\[
+ \hat{\omega}_c \left( \frac{\alpha_c \phi^T \phi_j}{(1 + \phi^T \phi_j)^2} \right) \epsilon H + \sum_{j=1}^{n_c} \frac{\alpha_c \phi_{c,j}^T \phi_j}{(1 + \phi^T \phi_j)^2} \epsilon H
\]

(60)

Given that \(Q(x)\) is symmetric, \(\alpha_c > 0\), there exists a positive constant \(q\) such that \(q \bar{x} \bar{x}^T \bar{x} < Q(\bar{x})\); therefore, \(L_23(t) \leq -p q \|x\|^2\). Meanwhile, by using Young’s inequality and noticing that \(\epsilon > 1\), we have:

\[
\dot{L}_23(t) \leq -\lambda_{min} \left( \frac{\phi^T \phi}{(1 + \phi^T \phi)^2} + \sum_{j=1}^{m} \frac{\alpha_c \phi^T \phi_j}{(1 + \phi^T \phi_j)^2} \right) \hat{\omega}_c
\]

\[
\times \left( \alpha_c - \frac{\alpha_c}{2} \|\hat{\omega}_c\|^2 + \frac{\alpha_c}{2} e_{\tau}^2 + \frac{\alpha_c}{2} \sum_{j=1}^{m} e_{c,j}^2 \right)
\]

(61)
Finally, we obtain

\[
\dot{L}_2 \leq \mu \| \tilde{x} \|^2 + \frac{\alpha_c}{2} \tilde{q}^2 + \frac{\alpha_c}{2} \sum_{j=1}^{m} E_{ij}^2 - \rho g \| x \|^2 \\
- \frac{\alpha_c}{2} \lambda_{\text{min}} \left( \frac{\phi_i^T \phi_i}{(1 + \phi_i^T \phi_i)^2} + \frac{\alpha_c \phi_j^T \phi_j}{(1 + \phi_j^T \phi_j)^2} \right) \| \tilde{\omega}_c \|^2
\]

(62)

According to (62), \( \dot{L}_2 (t) < 0 \) if one of the following inequalities holds:

\[
\| x \| > B_x, \| \tilde{x} \| > B_{\tilde{x}}, \| \tilde{\omega}_c \| > B_{\tilde{\omega}_c}
\]

By applying the Lyapunov extension theorem, we can conclude that the estimated state vector \( \tilde{x} \), the model identification error and the critic weights approximate error are UUB, respectively by \( B_x, B_{\tilde{x}} \) and \( B_{\tilde{\omega}_c} \). This completes the proof.

Remark 5: As shown in Theorem 2, the system states are asymptotically stable with any approximate control policy applied; it is not necessary to enlarge parameter \( q \) to restrict the system states within a small bound. The bound of \( \tilde{x} \) and \( \tilde{\omega}_c \) can be adjusted to be arbitrarily small, if we enlarge the design parameters such as \( \alpha_c, \eta_1 \), and \( \eta_2 \).

VI. SIMULATION VERIFICATION

In this section, we first demonstrate the capability of the proposed method in maintaining system stability during the critic learning process. The simulation is established based on an unstable nonlinear system. Then we compared the proposed method with other model-free control schemes to demonstrate control performance of the proposed method. Finally, we applied the proposed method to trajectory tracking control of an autonomous underwater vehicle to verify the applicability of the developed theoretical results.

To study the robustness of the proposed control scheme, we measured the \( L_2 \)-gain defined as:

\[
\rho (t) = \frac{1}{\int_0^t \int_0^\tau \| v (\tau) \|^2 d\tau d\tau} \int_0^t \int_0^\tau \| y (\tau) \|^2 d\tau d\tau
\]

which is an index typically used for measuring system robustness, it represents the sensitivity of observable performance output with respect to disturbance input.

A. NONLINEAR SYSTEM STABILIZING

Consider the following continuous-time nonlinear non-affine system that has been modified based on the system applied in [31] for conducting simulation results

\[
\dot{x} = f (x) + g (x, u) u + k (x) w (x)
\]

(63)

where

\[
f (x) = \begin{bmatrix}
-x_1 + x_2 \\
-0.5x_1 - 0.5x_2 (1 - (\cos (2x_1) + 2)^2)
\end{bmatrix}
\]

\[
g (x, u) = \begin{bmatrix}
0 \\
\cos (2x_1) + 2 \cos (0.2u)
\end{bmatrix}
\]

\[
k (x) = \begin{bmatrix}
x_2 \\
0
\end{bmatrix}
\]

The system states within a small bound. The bound of \( \tilde{x} \) and \( \tilde{\omega}_c \) can be adjusted to be arbitrarily small, if we enlarge the design parameters such as \( \alpha_c, \eta_1 \), and \( \eta_2 \).

\[
x = [x_1, x_2]^T \in \mathbb{R}^2 \text{ is the system states vector, and } u \in \mathbb{R} \text{ is the control input. The initial system state is } x_0 = [1, -0.5]^T.
\]

Here, we used the same disturbance signal \( w(x) \) as in [31], i.e., \( w(x) = \theta_1 x_1 \sin (\theta_2 x_2) \), where \( \theta_i (i = 1, 2) \) are randomly selected within the interval \([-1, 1]\). The key in obtaining robust optimal control of the system (63) is to solve the HJB equation associate with auxiliary system (18) and utility function (64). The designed parameters were selected as \( \eta_1 = 6, \eta_2 = 1, \text{ and } \lambda^{-1} = 0.067 \). The number of neurons in the hidden layer was \( m = 10 \). The concurrent learning memory pool size was chosen as \( m = 16 \). The weighting matrix \( \omega_m \) was initialized a zero matrix; the elements in matrix \( v_m \) were initialized as random numbers in the interval \([0, 1]\). According to (40), we set the utility function parameters as \( \beta_2 = 180, \alpha = 0.2 \). \( \beta_2 \tilde{x}^T \tilde{x} \) and \( Q (\tilde{x}) \) were combined such that the term \( \tilde{x}^T \tilde{x} \) is weighted by 80. The value function to be optimized is given by

\[
V (x) = \int_0^\infty \left( 80 \tilde{x}^T \tilde{x} + 180 \tilde{x}^T \dot{x} + u^T u + 0.2v^T v \right) dt
\]

(64)

The activation function for the critic network was selected as quadratic polynomial components of the augmented system states

\[
\sigma_c = \begin{bmatrix}
x_1^2, x_2^2, w_1^2, w_2^2, x_1 x_2, w_1 w_2, x_1 w_1, x_2 w_2, x_2 w_1, x_1 w_2
\end{bmatrix}^T
\]

With \( n_c = 10 \), the critic weight vector is a 10-dimensional vector written as \( \tilde{\omega}_c = [\tilde{\omega}_{c1}, \tilde{\omega}_{c2}, \ldots, \tilde{\omega}_{c10}] \). It is worth mentioning that the dimension of the activation function was determined by computational simulation and depends more on the implementer’s designing experience than on theoretical analysis. The elements of the initial critic weight vector were all set to zero, which means the initial control policy was \( u = 0 \). As verified in [31], this initial control law cannot stabilize the system; here, it was applied to demonstrate that the proposed method relaxes the initial admissible control condition.

In the experiment, the critic network was updated by computing the Hamiltonian (27) online. It is worth noting
that the auxiliary control $v$ is only involved in the Hamiltonian computation. Actually, it was not applied as the system input. Since the objective of training the critic network is to minimize the Hamiltonian for arbitrary $\hat{x} \in \mathcal{X}$, it is not necessary to have the state samples being conditioned on a specific trajectory. We used system states approximated by the observer as the input of the Hamiltonian. Then, the critic weights were updated and the approximate control policy was applied directly to the plant. We ran the simulation for 20 seconds. The system states trajectory and the state trajectory approximated by the observer are depicted in Fig. 3; notably, the states approximating errors converged. The control output during the learning process is depicted in subplot (a) of Fig. 4.

$Q(x)$ and $||v(x)||^2$ are compared in subplot (b) as the verification condition (41). The converged identifier weight matrices $\omega_m$ and $\nu_m$ are respectively as shown at the bottom of this page. Fig. 6 displays the convergence of the critic weights. The converged critic network weights vector was $\omega_c^T = [0.9848, 0.6003, 0.5402, 0.3299, -0.0794, 0.6677, -0.2765, -0.0555, 0.4303, 0.0272]$. It is worth noting that the critic weights were not updated in the first 16 steps; instead, this period was used to prepare the data for concurrent learning.

We then examined the effects of the parameter uncertainties. To this end, $f(x)$ in system plant (63) was modified as

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

where $\Delta$ is a random number sampled from normal distribution. With the optimal controller obtained from critic learning applied to the system plant, we compared the system state evolution process with and without parameter uncertainties. Simulation results are displayed in Fig. 5. From Fig. 5, parameter uncertainties caused minor effects on the states evolution trajectory. The obtained optimal controller is very robust to parameter uncertainties.

### B. NONLINEAR SYSTEM DISTURBANCE ATTENUATION PERFORMANCE STUDY

In this section, we will respectively apply the newly designed control scheme and two other promising model-free control schemes recognized as the most effective to a nonlinear system.
FIGURE 6. Convergence of the critic network weights.

plant to compare their performance. The $L_2$-gain, which is commonly used as an index of control system disturbance attenuation performance [35], [40], was measured and compared to demonstrate their robustness to disturbance effects. In addition to $L_2$-gain, indexes revealing control performance and learning efficiency are measured and compared to verify the effectiveness of the proposed method.

Consider the following system used in [40] for numerical simulations:

$$
\dot{x} = \begin{bmatrix} -x_1^3 - 2x_2 \\ x_1 + 0.5 \cos x_2 \sin x_3 \\ x_1 + 0.5 \sin x_1 \\ u + \begin{bmatrix} -1 \\ \cos x_2 \end{bmatrix} v \end{bmatrix}
$$

(65)

where $x_1$ and $x_2$ are system states, and $u, v \in \mathbb{R}$ denote the control input and perturbation input signal respectively. We selected the initial system state as $x_0 = [1, -0.5]^T$. The critic network and NN identifier share the same structure in the experiment as that described in Section A. The same external perturbation signal as in [40] was imposed (i.e., $v(t) = 3e^{-t} \cos t, t > 0$). The value function parameters were chosen as $\beta_2 = 180, \alpha = 0.2$, and the learning rate of the critic network was $\alpha_c = 3.5$.

Simulation results are displayed in Fig. 7-11. The approximated / actual system states and the approximated disturbance during the training process are depicted in Fig. 7. System state trajectories with the obtained approximate optimal control are compared in Fig. 9. A comparison of the critic weights convergence process is depicted in Fig. 10. As shown in Fig. 10, the critic weights vector converged to $\omega_c = [2.6658, 0.5470, 0.0395, 0.4896, -0.6941, 0.6320, 0.1468, 0.1737, 0.2040]^T$. The obtained optimal controller feedback gain was $\omega_a = [0.9390, 2.1959, 0.5902]^T$.

Although the critic network structure was more complicated, our method required less than 80 iterations to converge, whereas the method proposed in [40] required five times more iteration steps. The corresponding control output and $\rho(t)$ are depicted in Fig. 11. As shown in Fig. 11, the control system designed by using our method had an $L_2$ gain of 0.585, lower than that obtained in [40], i.e. 1.015. Since the auxiliary system was augmented with the estimated external disturbance, the learned control policy contained a feedforward loop for disturbance attenuation. Thus significantly improved the robustness to disturbance. With the states approximating error and the disturbance approximation terms involved in the HJB equation, the system dynamics become more deterministic and a more accurate critic gradient is obtained. Moreover, the concurrent learning technique reuses the historical data, further improving the learning efficiency.
We then compared our method with the online model-free control method known as model-free adaptive control (MFAC). The MFAC controller was designed based on the full-form dynamic linearization (FFDL) data model. The desired states trajectory was set to constant zeros. A comparison of the states evolution process is displayed in Fig. 8. The parameter setups of the three applied controller are shown in Table 1. We further measured and compared control performance of the three scheme in four aspects: $L_2$ gain; states quadratic cost (SQC) which is the time integration of $x^T(t)Qx(t)$; The learning steps required for the adaptive algorithm to converge (CS) and the states convergent time (SCT). All the measured values are listed in Table 2. From Table 2, robustness and efficiency of our method outperforms the widely applied MFAC scheme and the novel ADP-based $H_\infty$ scheme, which is optimal in terms of zerosum game.

### C. SIMULATION STUDY ON DSRV MOTION CONTROL

In this section, we discussed an experiment in which the proposed ADP method was applied to the depth and pitch control of a deep submergence rescue vehicle (DSRV) [41]. The simulation was carried out based on the Marine Systems Simulator (MSS) [42].

Consider a 5-meter-long DSRV cruising at a constant speed of 8 knots (4.11 m/s). The altitude of the vehicle is adjusted by a horizontal deployed rudder, so the vehicle can move up and down by altering its pitch angle. Fig. 12 displays the coordination system setup and the DSRV motions. The inertia frame (denoted by $I$) was fixed to the ground, with its $x$-axis and $y$-axis pointing to the north and east, respectively, and the $z$-axis points downward. DSRV motion state is described by a five dimensional vector $x = [w, q, x, z, \theta]^T$. $w$ and $q$ represent vehicle heave velocity and pitch angular velocity respectively; $x$ and $z$ represent the advancing distance and diving depth of the vehicle, i.e., the vehicle coordination in the inertial frame; and $\theta$ is the vehicle pitch angle with respect.
TABLE 3. DSRV dynamics parameters.

| Parameters | Value   | Parameters | Value   |
|------------|---------|------------|---------|
| $M_q$      | $-1.573e^{-3}$ | $Z_q$ | $1.300e^{-4}$ |
| $M_w$      | $-1.460e^{-4}$ | $Z_w$ | $3.155e^{-2}$ |
| $M_d$      | $-1.131e^{-2}$ | $Z_d$ | $1.746e^{-2}$ |
| $M_s$      | $1.118e^{-2}$ | $Z_s$ | $4.394e^{-2}$ |
| $M_\delta$ | $-0.1563\times U^2$ | $Z_\delta$ | $2.770e^{-2}$ |
| $M_\theta$ | $-1.280e^{-2}$ | $Z_\theta$ | — |

The dynamics of the DSRV are given by

$$\begin{bmatrix}
\dot{w} \\
\dot{q}
\end{bmatrix} = M \begin{bmatrix}
Z \\
\tau
\end{bmatrix} / \det(M) \\
\begin{bmatrix}
\dot{x} \\
\dot{z}
\end{bmatrix} = J(\theta) \begin{bmatrix}
U_0 \\
w
\end{bmatrix}$$

(66)

where

$$M = \begin{bmatrix}
I_y - M_\dot{q}q & Z_\dot{q} \\
M_\dot{w} & m - Z_\dot{w}
\end{bmatrix}$$

is the inertia matrix comprised of mass, inertia and added mass. The projecting matrix

$$J(\theta) = \begin{bmatrix}
\cos \theta & \sin \theta \\
-\sin \theta & \cos \theta
\end{bmatrix}$$

projects the speed measured in the body frame to the speed in the inertia frame. The heave force $Z$ and pitch torque $\tau$ are given, respectively, by:

$$Z = Z_d q + Z_w w + Z_\delta \delta$$

$$\tau = M_d q + M_w w + M_\theta \theta + M_\delta \delta$$

(67)

All inertia and hydrodynamics parameters are listed in Table 3. To simplify the implementation, we extracted the control system states as

$$x = [\theta - \theta_d, z - z_d]^T$$

(68)

where, $z_d$ and $\theta_d$ are the expected depth and pitch angle, respectively. The objective of this experiment was to control the DSRV to track the desired trajectory in the $x$-$z$ plane, which is generated by

$$[\dot{z}_d, \dot{\theta}_d]^T = f(z_d, \theta_d)$$

(69)

For simplification, we set the desired trajectory of the depth and pitch angle with respect to time as

$$\begin{cases}
\dot{z}_d &= t \times 4.11 \sin(-\theta_d), \\ 
\dot{\theta}_d &= -0.3 (t < 4.11) \\ 
\dot{z}_d &= 5, \\ 
\dot{\theta}_d &= 0 (t \geq 4.11)
\end{cases}$$

(70)

The vehicle states are initially set to zero with the cruise speed $U = 8$ knots. First, we applied the proposed method to learn the optimal control policy while tracking the desired trajectory (70). Fig. 13 displays the actual and approximated depth and pitch angle errors. The DSRV motion trajectory is shown in Fig. 14 (a). The convergence process of the critic weights is depicted in Fig. 15, as shown in the figure, the critic weights converged to $\omega_c = [1.3176, 1.8479, 0.0505, 0.2071, -0.0365, 0.3522, 0.2356, -0.0513, 0.5442, -0.0410]^T$ in less than 60 learning iterations. Then, we applied the obtained optimal controller to the DSRV. Fig. 14 depicts the desired and actual vehicle motion trajectory in the $x$-$z$ plane of the inertia frame. As shown in Fig. 13 and 14, the proposed method guarantees that the trajectory tracking error is asymptotically stable, ultimately reaching and maintaining a cruise depth of 5m. The corresponding rudder command is depicted in Fig. 16. It is worth noting that in real applications,
FIGURE 16. DSRV rudder command by the obtained optimal controller and the controller’s output during the training process.

FIGURE 17. Control performance measurement of learning stage (a) and testing stage (b) in terms of states quadratic cost.

FIGURE 18. Flowchart of the Matlab simulation program.

the rudder angle cannot exceed ±30 degree (0.524 rads). The obtained optimal controller output was limited by this nonlinear saturation constraint. Fig. 17 displays the SQC of the propose method in this study. The obtained optimal controller promoted control performance in terms of SQC by 100 times.

The abovementioned numerical simulations were established using Matlab. We applied Runge-Kutta algorithm (RK) to solve the proposed differential equations. The control scheme was implemented in a callback function which provides time derivative of system states, NN weights and identifier approximated states for RK. The simulation program was executed on a PC with the processor Intel-i7 8500. The average elapsed time for a 20-second simulating process is 3.4s. Flow chart of our implementation is depicted in Fig. 18. System states and adaptive parameters are reshaped and concatenated as a vector $s$ to be transmitted among computational models.

VII. CONCLUSION

Previous work has documented the effectiveness of RADP and $H_{\infty}$ in solving nonlinear system robust optimal control problems. Existing methods were constructed based on the policy iteration framework, and intelligent components were applied for value function approximation. The optimal control problem can be solved forward in time without requiring any knowledge of the system dynamics. However, it is very difficult to guarantee system stability to collect data safely for critic learning. In this study, a novel NN-based identifier was developed to reconstruct an auxiliary system through critic learning based on the auxiliary system and a designed utility function. The obtained control policy is robust to the critic NN and identifier NN weight parameters. The control system is guaranteed to be asymptotically stable before the critic learning converges. This work effectively extended RADP methods by introducing dynamics identifiers and providing robustness to the variation in critic and identifier NN weights. In general, the proposed method achieves online model-free critic learning for nonlinear non-affine systems, without requiring an initial admissible controller, overcoming most of the difficulties encountered in real applications of ADP. The proposed technique is expected to allow robots and autonomous systems to improve their behaviors much as humans do, by learning through safe interactions with the environment.

REFERENCES

[1] D. Wang, H. He, and D. Liu, “Adaptive critic nonlinear robust control: A survey,” IEEE Trans. Cybern., vol. 47, no. 10, pp. 3429–3451, Oct. 2017.
[2] Z. Jiang and Y. Jiang, “Robust adaptive dynamic programming for linear and nonlinear systems: An overview,” Eur. J. Control, vol. 19, no. 5, pp. 417–425, Sep. 2013.
[3] Y.-J. Liu, S. Tong, C. L. P. Chen, and D.-J. Li, “Neural controller design-based adaptive control for nonlinear MIMO systems with unknown hysteresis inputs,” IEEE Trans. Cybern., vol. 46, no. 1, pp. 9–19, Jan. 2016.
[4] L. Liu, Z. Wang, Z. Huang, and H. Zhang, “Adaptive predefined performance control for MIMO systems with unknown direction via generalized fuzzy hyperbolic model,” IEEE Trans. Fuzzy Syst., vol. 25, no. 3, pp. 527–542, Jun. 2017.
[5] W. He, Y. Dong, and C. Sun, “Adaptive neural impedance control of a robotic manipulator with input saturation,” IEEE Trans. Syst., Man, Cybern. Syst., vol. 46, no. 3, pp. 334–344, Mar. 2016.
[6] W. He, Y. Chen, and Z. Yin, “Adaptive neural network control of an uncertain robot with full-state constraints,” IEEE Trans. Cybern., vol. 46, no. 3, pp. 620–629, Mar. 2016.
[7] Y. Wang, L. Cheng, Z.-G. Hou, J. Yu, and M. Tan, “Optimal formation of multirobot systems based on a recurrent neural network,” IEEE Trans. Neural Netw. Learn. Syst., vol. 27, no. 2, pp. 322–333, Feb. 2016.

[8] H. Zhang, J. Zhang, G.-H. Yang, and Y. Luo, “Leader-based optimal coordination control for the consensus problem of multiagent differential games via fuzzy adaptive dynamic programming,” IEEE Trans. Fuzzy Syst., vol. 23, no. 1, pp. 152–163, Feb. 2015.

[9] C. Wang, D. Liu, Q. Wei, D. Zhao, and Z. Xia, “Iterative adaptive dynamic programming approach to power optimal control for smart grid with energy storage devices,” Acta Autom. Sinica, vol. 40, no. 9, pp. 1984–1990, 2014.

[10] Y. Tang, H. He, J. Wen, and J. Liu, “Power system stability control for a wind farm based on adaptive dynamic programming,” IEEE Trans. Smart Grid, vol. 6, no. 1, pp. 166–177, Jan. 2015.

[11] D. Vrabie, O. Pastravanu, M. Abu-Khalaf, and F. L. Lewis, “Adaptive optimal control for continuous-time linear systems based on policy iteration,” Automatica, vol. 45, no. 2, pp. 477–484, Feb. 2009.

[12] D. Vrabie and F. Lewis, “Neural network approach to continuous-time direct adaptive optimal control for partially unknown nonlinear systems,” Neural Netw., vol. 22, no. 3, pp. 237–246, Apr. 2009.

[13] H. Modares, F. L. Lewis, and M.-B. Naghibi-Sistani, “Integral reinforcement learning and experience replay for adaptive optimal control of partially-unknown constrained-input continuous-time systems,” Automatica, vol. 50, no. 1, pp. 193–202, Jan. 2014.

[14] H. Li, D. Liu, and D. Wang, “Integral reinforcement learning for linear continuous-time zero-sum games with completely unknown dynamics,” IEEE Trans. Autom. Sci. Eng., vol. 11, no. 3, pp. 706–714, Jul. 2014.

[15] M. Palanisamy, H. Modares, F. L. Lewis, and M. Auranzeb, “Continuous-time Q-learning for infinite-horizon discounted cost linear quadratic regulator problems,” IEEE Trans. Cybern., vol. 45, no. 2, pp. 165–176, Feb. 2015.

[16] H. Modares and F. L. Lewis, “Optimal tracking control of nonlinear partially-known constrained-input systems using integral reinforcement learning,” Automatica, vol. 50, no. 7, pp. 1780–1792, Jul. 2014.

[17] G. Wen, C. L. P. Chen, S. S. Ge, H. Yang, and X. Liu, “Optimized adaptive nonlinear tracking control using actor-critic reinforcement learning strategy,” IEEE Trans. Ind. Informat., vol. 15, no. 9, pp. 4969–4977, Sep. 2019.

[18] K. G. Vamvoudakis, “Event-triggered optimal adaptive control algorithm for continuous-time nonlinear systems,” IEEE/CIA J. Autom. Sinica, vol. 1, no. 3, pp. 282–293, Jul. 2014.

[19] A. Sahoo, H. Xu, and S. Jagannathan, “Neural network-based event-triggered state feedback control of nonlinear continuous-time systems,” IEEE Trans. Neural Netw. Learn. Syst., vol. 27, no. 3, pp. 497–509, Mar. 2016.

[20] X. Yang and H. He, “Adaptive critic designs for event-triggered robust control of nonlinear systems with mismatched disturbances,” Int. J. Adapt. Control Signal Process., vol. 27, no. 4, pp. 280–301, Apr. 2013.

[21] X. Yang and H. He, “Self-learning robust optimal control for continuous-time nonlinear systems with mismatched disturbances,” Neural Netw., vol. 99, pp. 19–30, Mar. 2018.

[22] H. Modares and F. L. Lewis, “Optimal tracking control of nonlinear partially-unknown constrained-input systems using integral reinforcement learning,” Trans. Neural Netw. Learn. Syst., vol. 29, no. 4, pp. 993–1005, Apr. 2018.

[23] S. Sun, G. Zhao, and Y. Peng, “Adaptive optimal output feedback tracking control for nonlinear discrete-time systems using a combined reinforcement Q-learning and internal model method,” IET Control Theory Appl., vol. 13, no. 18, pp. 3075–3086, Dec. 2019.

[24] T. Dierks, B. T. Thumati, and S. Jagannathan, “Optimal control of unknown affine nonlinear discrete-time systems using offline-trained neural networks with proof of convergence,” Neural Netw., vol. 22, nos. 5–6, pp. 851–860, Jul. 2009.

[25] D. Wang, C. Mu, D. Liu, and H. Ma, “On mixed data and event driven design for adaptive-critic-based nonlinear $H_\infty$ control,” IEEE Trans. Neural Netw. Learn. Syst., vol. 29, no. 4, pp. 993–1005, Apr. 2018.

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

[27] F. L. Lewis, D. Vrabie, and K. G. Vamvoudakis, “Reinforcement learning and feedback control: Using natural decision methods to design optimal adaptive controllers,” IEEE Control Syst., vol. 32, no. 6, pp. 76–105, Dec. 2012.

[28] J. Han, “Extended state observer for a class of uncertain plants,” Chin. Control Decis., vol. 10, no. 1, pp. 85–88, Jun. 1995.

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

[30] D. Wang, H. He, C. Mu, and D. Liu, “Intelligent critic control with disturbance attenuation for affine dynamics including an application to a microgrid system,” IEEE Trans. Ind. Electron., vol. 64, no. 6, pp. 4935–4944, Jun. 2017.

[31] H. Shuping and Y. Zheping, “State-of-art and development trend of deep submarine rescue vehicle (DSRV),” Ship Eng., vol. 4, pp. 1–5, Apr. 2004.

[32] T. Perez, O. N. Smogeli, T. I. Fossen, and A. J. Sørensen, “An overview of the marine systems simulator (MSS): A simulink toolbox for marine control systems,” Model., Identificat. Control, Norwegian Res. Bull., vol. 27, no. 4, pp. 259–275, 2006.
XIUFEN YE (Senior Member, IEEE) was born in 1966. She received the B.S., M.S., and Ph.D. degrees in control theory and control engineering from Harbin Shipbuilding Engineering University (Harbin Engineering University), Harbin, China, in 1987, 1990, and 2003, respectively. She has been a Professor with the College of Automation, Harbin Engineering University, since September 2003. She is the author of more than 180 articles. She holds more than 20 patents. Her current research interests include underwater vehicle intelligent control systems, digital image processing, and object detection and tracking. She was a recipient of the two provincial and ministerial science and technology progress awards. She has served as the Program Committee Chairs for the IEEE ICIA 2010 and the IEEE/ICME CME 2011.

WENZHI LIU was born in 1968. He has been a Professorate Senior Engineer with the College of Information and Communication Engineering, Harbin Engineering University, since April 2019. His current research interests include underwater robotics and other new unmanned system of oceans.

* * *