Adaptive Critic Nonlinear Robust Control: A Survey

Ding Wang, *Member, IEEE*, Haibo He, *Senior Member, IEEE*, and Derong Liu, *Fellow, IEEE*

***Abstract*—Adaptive dynamic programming (ADP) and reinforcement learning are quite relevant to each other when performing intelligent optimization. They are both regarded as promising methods involving important components of evaluation and improvement, at the background of information technology, such as artificial intelligence, big data, and deep learning. Although great progresses have been achieved and surveyed when addressing nonlinear optimal control problems, the research on robustness of ADP-based control strategies under uncertain environment has not been fully summarized. Hence, this survey reviews the recent main results of adaptive-critic-based robust control design of continuous-time nonlinear systems. The ADP-based nonlinear optimal regulation is reviewed, followed by robust stabilization of nonlinear systems with matched uncertainties, guaranteed cost control design of unmatched plants, and decentralized stabilization of interconnected systems. Additionally, further comprehensive discussions are presented, including event-based robust control design, improvement of the critic learning rule, nonlinear H_∞ control design, and several notes on future perspectives. By applying the ADP-based optimal and robust control methods to a practical power system and an overhead crane plant, two typical examples are provided to verify the effectiveness of theoretical results. Overall, this survey is beneficial to promote the development of adaptive critic control methods with robustness guarantee and the construction of higher level intelligent systems.**

***Index Terms*—Adaptive critic designs, adaptive/approximate dynamic programming (ADP), boundedness, convergence, neural networks, optimal control, reinforcement learning, robust control, stability.**

I. Introduction

ARTIFICIAL intelligence, big data, and deep learning are all hot topics of information technology. The artificial intelligence techniques such as machine learning [1], [2] and deep learning [3]–[5] are extremely helpful for the study of big data [6], [7]. Recently, Google DeepMind developed a program called AlphaGo [8] that has shown performance previously thought to be impossible for at least a decade. Instead of exploring various sequences of moves, AlphaGo learns to make a move by evaluating the strength of its position on the board. This kind of evaluation was ensured to be possible via deep learning capabilities of neural networks [9]–[11]. Due to the excellent properties of adaptivity, advanced input–output mapping, fault tolerance, nonlinearity, and self-learning, neural networks are frequently used for universal function approximation in numerical algorithms. Deep neural networks-based learning has played a vital role in AlphaGo's success [12]. Position evaluation, aimed at approximating the optimal cost function of the game, is the key procedure of AlphaGo. Noticeably, reinforcement learning [13] is an indispensable component of this advanced product.

Manuscript received January 9, 2017; revised May 3, 2017; accepted May 18, 2017. Date of publication July 3, 2017; date of current version September 14, 2017. This work was supported in part by the National Natural Science Foundation of China under Grant 51529701, Grant U1501251, Grant 61533017, Grant 61233001, and Grant 61520106009, in part by the Beijing Natural Science Foundation under Grant 4162065, in part by the U.S. National Science Foundation under Grant ECCS 1053717 and Grant CMMI 1526835, and in part by the Early Career Development Award of SKLMCCS. This paper was recommended by Associate Editor H. Zhang. *(Corresponding author: Haibo He.)*

D. Wang is with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, and also with the School of Computer and Control Engineering, University of Chinese Academy of Sciences, Beijing 100049, China. This research was conducted when D. Wang was a Visiting Scholar at the Department of Electrical, Computer, and Biomedical Engineering, University of Rhode Island, Kingston, RI 02881 USA (e-mail: ding.wang@ia.ac.cn).

H. He is with the Department of Electrical, Computer and Biomedical Engineering, University of Rhode Island, Kingston, RI 02881 USA (e-mail: he@ele.uri.edu).

D. Liu is with the School of Automation, Guangdong University of Technology, Guangzhou 510006, China (e-mail: derongliu@foxmail.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2017.2712188

A. Reinforcement Learning and Adaptive Critic Designs

As an important branch of artificial intelligence and especially machine learning, reinforcement learning tackles modification of actions based on interactions with the environment. The environment comprises everything outside the agent (the learner and the decision-maker) and also interacts with the agent. Reinforcement learning focuses on how an agent ought to take actions in an environment so as to maximize the cumulative reward or minimize the punishment, where the idea of optimization is involved. In fact, people often are interested in mimicking nature and designing automatic control systems that are optimal to effectively achieve required performances without unduely depending on the limited resources. Prescribing a search tracking backward from the final step and employing the principle of optimality thereby finding the optimal policy, dynamic programming is a useful computational technique to solve optimal control problems [14], [15]. However, due to the defect of backward numerical process when coping with the high-dimensional optimization problems, it is computationally untenable to run dynamic programming to obtain the optimal solution (i.e., the well-known "curse of dimensionality" [14]). What is worse, the backward direction of the search process precludes the use of dynamic programming in real-time control.

Reinforcement learning is highly related to dynamic programming technique. Classical dynamic programming algorithms are of limited utility in reinforcement learning because of their dependence on the perfect model and a mass of computational expense. However, dynamic programming provides an essential foundation for understanding reinforcement learning. There is a class of reinforcement learning methods incorporating the actor-critic (or adaptive critic) structure, where an actor component applies an action (or control law) to the environment and a critic component evaluates the value of that action. The combination of actor-critic structure, dynamic programming, and neural networks, results in the adaptive/approximate dynamic programming (ADP) algorithm, invented by Werbos [16]–[18] and Santiago and Werbos [19] for obtaining approximate optimal solutions. The core idea of ADP is the adaptive critic-based optimization and it is regarded as a necessary outlet to achieve truly brain-like intelligence [18], [19].

B. Adaptive Critic Optimal Control Design

Neural networks and fuzzy systems are always regarded as important intelligent complements to practical control engineering. Actually, they are often used as fundamental components of various computational intelligence techniques and the optimization design of complex dynamics based on them is a significant topic of decision and control community [20]–[24]. Linear optimal regulators has been studied by control scientists and engineers for many years. However, it is not an easy task to acquire the analytic solution of the Hamilton–Jacobi–Bellman (HJB) equation for general nonlinear systems. Thus, their optimal feedback design is much too difficult but considerable important. Remarkably, the successive approximation method [25]–[28] and the closely related ADP method are both developed to conquer the difficulty via approximating the HJB solution. In general, ADP is a promising technique to approximate optimal control solutions for complex systems [16]–[19], [25]–[28]. Particularly, it is regarded as an effective strategy to design optimal controllers in online and forward-in-time manners. Among them, the adaptive critic is the basic framework and neural networks are often involved to serve as the function approximator. Employing the ADP method always results in adaptive near-optimal feedback controllers and hence is useful to perform various nonlinear intelligent control applications.

There are several synonyms used for ADP and most of them are closely related to neural networks. They are "adaptive critic designs" [29]–[31], "ADP" [32], [33], "approximate dynamic programming" [18], [27], [34], "neural dynamic programming" [35], [36], "neuro-dynamic programming" [37], "reinforcement learning" [13], [34] including Q-learing [38], and "relaxed dynamic programming" [39], [40]. In the basic framework, there are three components: 1) critic; 2) model; and 3) action. They are usually implemented via neural networks and perform the function of evaluation, prediction, and decision, respectively. Some improved structures are also proposed, such as the goal representation ADP [41]–[44] and fuzzy ADP [44], [45]. In the last two decades, ADP has been promoted extensively when coping with adaptive optimal control of discrete-time systems [46]–[65] and continuous-time systems [66]–[85]. Among them, the iterative ADP algorithm based on value iteration is important to the self-learning optimal control design of discrete-time systems [27], [47], [52], [55], [59], while the policy iteration is significant to the adaptive optimal control design of continuous-time systems [28], [66], [69], [77], [82]. The convergence of these iterative algorithms is a basic issue so that it has been sufficiently studied [27], [28], [47], [52], [54], [55], [57], [59], [62], [64], [66], [69], [73], [76], [77], [79], [82], [83]. For comprehensive survey papers and books of recent developments, please refer to [86]–[99], including various topics in terms of theory, design, analysis, and applications. As emphasized by Lewis *et al.* [87]–[89], the ADP technique is closely related to reinforcement learning when engaging in the research of feedback control. In general, value and policy iterations are fundamental algorithms of reinforcement learning-based ADP in optimal control. It is easy to initialize the value iteration, but one cannot always guarantee stability of iterative control laws during the implementation process. Policy iteration starts with a stabilizing controller, but it is difficult to find the initial admissible control law in many situations. As a result, the generalized version of these two algorithms has received great attention [60], [63], [87]–[89], [96] recently, for integrating their advantages and avoiding the weaknesses.

The rapid development of information technology, especially artificial intelligence, big data, and deep learning, are profoundly affecting our society. Nowadays, the data-driven control design has become a hot topic in the field of control theory and control engineering [100]–[104]. The development of ADP methods greatly promotes the research of data-based optimal control design [46], [54], [55], [76], [78], [80], [81], [93], [105], [106]. A novel iterative neural dynamic programming algorithm was developed in [105] and [106], reflecting a combination of neural dynamic programming technique and the iterative ADP algorithm. The integral reinforcement learning proposed in [107]–[109] provides a new outlet of achieving the model-free optimal regulation. All of these results are beneficial to the development of artificial intelligence and computational intelligence techniques.

C. Adaptive-Critic-Based Nonlinear Robust Control Design

Existing results of ADP methods are mostly obtained under the assumption that there are no dynamical uncertainties in the controlled plants. Nevertheless, practical control systems are always subject to model uncertainties, exogenous disturbances or other changes in their lifetime. They are necessarily considered during the controller design process in order to avoid the deterioration of nominal closed-loop performance. A controller is said to be robust if it works even if the actual system deviates from its nominal model on which the controller design is based. The importance of the robust control problem is evident which has been studied by control scientists for many years (see [110]–[115] and the references therein). In [114] and [115], the robust control problem was handled by using the optimal control approach for the nominal system.[1]

[1] It represents the portion of system without considering the uncertainty during the feedback control design aimed at guaranteeing the desired performance of a dynamic plant containing uncertain elements [113]–[115].

This is a very important result which establishes a connection between the two control topics. However, the detailed procedure is not discussed and it is difficult to cope with general nonlinear systems. Then, an optimal control scheme based on the HJB solution for robust controller design of nonlinear systems was proposed in [116] and [117]. The algorithm was constructed by using the least squares method performed offline while the closed-loop stability analysis was not fully discussed.

Since 2013, there gradually appeared some publications of ADP-based robust control designs [118]–[127]. In general, the problem transformation is conducted to build a close relationship between the robustness and optimality. Moreover, the closed-loop system is always proven to be uniformly ultimately bounded (UUB) that will be defined later. In [118], a policy iteration algorithm was developed to solve the robust control problem of continuous-time nonlinear systems with matched uncertainties and the algorithm was improved in [119]. This method was extended to deal with the robust stabilization of matched nonlinear systems with unknown dynamics [120] and with constrained inputs [121]. Incidentally, it is worth mentioning that a tentative result of ADP-based robust control design of discrete-time nonlinear systems was given in [122]. For improving the learning rule of the critic neural network, the adaptation-oriented near-optimal control problem was revisited and then the robust stabilization of nonlinear systems was studied with further results [123]. Moreover, the robust control method of nonlinear systems with unmatched uncertainties was derived in [124]. The robust control design with matched uncertainties and disturbances was also studied in [125] as an extension of [119]. Note the data-driven approaches are helpful to the ADP-based robust control design since system uncertainties can sometimes be regarded as unknown dynamics. For discussing the optimality of the ADP-based robust controller, a novel data-based robust optimal control method of matched nonlinear systems was constructed [126]. Data-based robust adaptive control for a class of unknown nonlinear systems with constrained-input was studied via integral reinforcement learning [127]. These results guarantee that ADP methods are applicable to a large class of complex nonlinear systems under uncertain environment. Hence, they greatly broadens the application scope of ADP, since many of previous publications do not focus on the robustness of obtained controllers. Subsequently, because of possessing the common speciality of handling system uncertainty, the combination of sliding mode control with ADP provides a new direction to the study of self-learning control design [53], [128]. In [53], the application issue on air-breathing hypersonic vehicle tracking was addressed by employing an innovative combination of sliding mode control and ADP. Then, the sliding mode control method based on ADP was used in [128] to stabilize the closed-loop system with time-varying disturbances and guarantee the nearly optimal performance of the sliding-mode dynamics.

For filling up the gap in most of ADP literature where dynamic uncertainties or unmodeled dynamics were not addressed, an important framework named robust ADP was proposed in [129]–[133] to cope with the nonlinear robust optimal control design from another aspect. An overview of robust ADP method for linear and nonlinear systems was given in [130], outlining the development of robust ADP theory with potential applications in engineering and biology. In [131], a key strategy integrating several tools of modern nonlinear control theory, such as the robust redesign and backstepping techniques as well as the nonlinear small-gain theorem [134], was developed with ADP formulation. After that, the robust ADP method was employed to decentralized optimal control of large-scale systems [132] and output feedback control of interconnected systems [133]. Therein, the applications of robust ADP to power systems were given special attention [129]–[133]. Generally, the robust ADP design cannot only stabilize the original uncertain system, but also achieve optimality in the absence of dynamic uncertainty. It was emphasized that, under the framework of robust ADP, computational designs for robust optimal control can be carried out based only on the online data of the state and input variables [130]. In this sense, the robust ADP method can be regarded as a nonlinear variant of [135], where a computational adaptive optimal control strategy was proposed to iteratively solve the linear algebraic Riccati equation using online information of state and input.

However, as we have seen, most of the previous research only concerns with the robustness of the uncertain system and the optimality of the nominal system [118], [121], [123], [124], [131]. In other words, the direct optimal control design of uncertain nonlinear systems is very difficult. This is because coping with the cost function of the uncertain plant is not an easy task. Therefore, some researchers have paid attention to the study of boundedness of the cost function with respect to the uncertain plant, in addition to optimizing it. The guaranteed cost control strategy [136] possesses the advantage of providing an upper bound on a given cost and therefore the degradation of control performance incurred by system uncertainties can be guaranteed to be less than this bound. When discussing the optimality with respect to the guaranteed cost function, it leads to the optimal guaranteed cost control problem. The guaranteed cost control design is a somewhat mature research topic of control community, but there are some new results with the emerging ADP formulation [137]–[141]. Under the ADP framework, we obtain a novel self-learning optimal guaranteed cost control scheme.

When studying complex dynamical systems, we often partition them into a number of interconnected subsystems for convenience. The combination of these subsystems can be seen as large-scale systems. As one of the effective control schemes for large-scale systems, the decentralized control design has acquired much interest because of its evident advantages, such as easy implementation and low dimensionality [119], [142]–[147]. It is shown that the decentralized stabilization for a class of interconnected nonlinear systems is closely related to the ADP-based robust control design [119], [144]–[147]. In this sense, the self-learning decentralized control scheme can be constructed with ADP formulation. Note that, the robustness issue is also included in the aforementioned guaranteed cost control and decentralized control designs. It will be illustrated that these three control topics are closely connected under the proposed adaptive critic framework.

D. Structure and Notations

Based on the existing results, this paper presents a survey of the adaptive-critic-based robust control design of continuous-time uncertain nonlinear systems. The ADP formulation for nonlinear optimal regulation design is reviewed in Section II. The ADP-based robust stabilization of nonlinear systems with matched uncertainties, guaranteed cost control design of unmatched case, and decentralized control of interconnected case are reviewed in Sections III–V, respectively. After that, further discussions on ADP-based robust control design and some comparison remarks are given in Sections VI and VII, respectively. Some practical applications are performed in Section VIII to verify the effectiveness of the ADP-based robust control methodology. Several notes on future perspectives are included in Section IX and overall conclusions are presented in Section X. Through this survey, it is hoped to further promote the application of ADP-based methods to intelligent control of more general nonlinear systems and the construction of more intelligent control systems.

For consistency and convenience, the following notations will be used throughout the survey. $\mathbb{R}$ represents the set of all real numbers. $\mathbb{R}^n$ is the Euclidean space of all n-dimensional real vectors. $\mathbb{R}^{n\times m}$ is the space of all $n \times m$ real matrices. $\|\cdot\|$ denotes the vector norm of a vector in $\mathbb{R}^n$ or the matrix norm of a matrix in $\mathbb{R}^{n\times m}$. I_n represents the $n \times n$ identity matrix. $\lambda_{\max}(\cdot)$ and $\lambda_{\min}(\cdot)$ stand for the maximal and minimal eigenvalues of a matrix, respectively. Let Ω be a compact subset of $\mathbb{R}^n$, Ω_u be a compact subset of $\mathbb{R}^m$, and $\mathscr{A}(\Omega)$ be the set of admissible control laws (defined in [26], [28], [66], and [77]) on Ω. ρ is the parameter in the utility corresponding to the uncertain term. $\mathcal{L}_2[0, \infty)$ denotes a space of functions where the Lebesgue integral of the element is finite. ϱ is the $\mathcal{L}_2$-gain performance level. i is the symbol of the ith subsystem in an interconnected plant, j is the sampling instant of the event-triggering mechanism, and k is the iteration index of the policy iteration algorithm. $\mathbb{N}^+ = \{i\}_{i=1}^N = \{1, 2, \dots, N\}$ denotes the set of positive integers between 1 and N. $\mathbb{N} = \{0, 1, 2, \dots\}$ stands for the set of all non-negative integers. "T" is used for representing the transpose operation and $\nabla(\cdot) \triangleq \partial(\cdot)/\partial x$ is employed to denote the gradient operator.

II. Review of ADP-Based Continuous-Time Nonlinear Optimal Regulation

In this section, we present a brief review of the continuous-time nonlinear optimal regulation method with neural network implementation. The basic idea of the ADP method for optimal control of continuous-time systems is involved therein.

A. Basic Optimal Control Problem Description

We consider a class of continuous-time nonlinear systems with control-affine inputs given by

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

where $x(t) \in \Omega \subset \mathbb{R}^n$ is the state vector, $u(t) \in \Omega_u \subset \mathbb{R}^m$ is the control vector, and the system functions $f(\cdot)$ and $g(\cdot)$ are differentiable in the arguments satisfying $f(0) = 0$. We let the initial state at $t = 0$ be $x(0) = x_0$ and $x = 0$ be the equilibrium point of the controlled plant. The internal system function $f(x)$ is assumed to be Lipschitz continuous on the set Ω in $\mathbb{R}^n$ which contains the origin. Generally, the nonlinear plant (1) is assumed to be controllable.

In this survey, we consider the undiscounted optimal control problem with infinite horizon cost function. We let

$$U(x(t), u(t)) = Q(x(t)) + u^{\mathsf{T}}(t)Ru(t) \tag{2}$$

denote the utility function,[2] where the scalar function $Q(x) \geq 0$ and the m-dimensional square matrix $R = R^{\mathsf{T}} > 0$, and then define the cost function as

$$J(x(t), u(t)) = \int_t^{\infty} U(x(\tau), u(\tau))\mathrm{d}\tau. \tag{3}$$

For simplicity, the cost $J(x(t), u(t))$ is written as $J(x(t))$ or $J(x)$ in the sequel. What we always concern is the cost function starting from $t = 0$, represented as $J(x(0)) = J(x_0)$.

During optimal control design, we want to derive the optimal feedback control law $u(x)$ to minimize the cost function (3), where $u(x)$ should be admissible.

Definition 1 [26], [28], [66], [77]: A control law $u(x)$ is said to be admissible with respect to (3) on Ω, denoted by $u \in \mathscr{A}(\Omega)$, if $u(x)$ is continuous on Ω, $u(0) = 0$, $u(x)$ stabilizes system (1) on Ω, and $J(x_0, u)$ is finite for all $x_0 \in \Omega$.

For an admissible control law $u(x) \in \mathscr{A}(\Omega)$, if the related cost function (3) is continuously differentiable, then the infinitesimal version is the nonlinear Lyapunov equation

$$0 = U(x, u(x)) + (\nabla J(x))^{\mathsf{T}}[f(x) + g(x)u(x)] \tag{4}$$

with $J(0) = 0$. Define the Hamiltonian of system (1) as

$$H(x, u(x), \nabla J(x)) = U(x, u(x)) + (\nabla J(x))^{\mathsf{T}}[f(x) + g(x)u(x)]. \tag{5}$$

Using Bellman's optimality principle, the optimal cost function $J^*(x)$, specifically defined as

$$J^*(x) = \min_{u\in\mathscr{A}(\Omega)} \int_t^{\infty} U(x(\tau), u(\tau))\mathrm{d}\tau \tag{6}$$

satisfies the so-called continuous-time HJB equation

$$\min_{u\in\mathscr{A}(\Omega)} H\big(x, u(x), \nabla J^*(x)\big) = 0. \tag{7}$$

Based on optimal control theory, the optimal feedback control law is computed by

$$\begin{aligned} u^*(x) &= \arg\min_{u\in\mathscr{A}(\Omega)} H(x, u(x), \nabla J^*(x)) \\ &= -\frac{1}{2}R^{-1}g^{\mathsf{T}}(x)\nabla J^*(x). \end{aligned} \tag{8}$$

[2]The selected state-related utility $Q(x(t))$ is more general than the classical form $x^{\mathsf{T}}(t)Qx(t)$, where $Q = Q^{\mathsf{T}} > 0$. The control-related utility can be chosen as the nonquadratic form [47], [121], [148], [149] instead of the traditionally quadratic one $u^{\mathsf{T}}(t)Ru(t)$ when encountering input constraints.

Algorithm 1 Policy Iteration for Optimal Control Problem

1: Initialization
Let the initial iteration index be $k = 0$ and $J^{(0)}(\cdot) = 0$. Give a small positive number ϵ as the stopping threshold. Start iterating from an initial admissible control law $u^{(0)}$.

2: Policy Evaluation
Using the control law $u^{(k)}(x)$, solve the following nonlinear Lyapunov equation

$$0 = U\big(x, u^{(k)}(x)\big) + \big(\nabla J^{(k+1)}(x)\big)^{\mathsf{T}} \dot{x} \tag{11}$$

with $J^{(k+1)}(0) = 0$, where $\dot{x} = f(x) + g(x)u^{(k)}(x)$.

3: Policy Improvement
Based on $J^{(k+1)}(x)$, update the control law via

$$u^{(k+1)}(x) = -\frac{1}{2}R^{-1}g^{\mathsf{T}}(x)\nabla J^{(k+1)}(x). \tag{12}$$

4: Stopping Criterion
If $|J^{(k+1)}(x) - J^{(k)}(x)| \le \epsilon$, stop and obtain the approximate optimal control law; else, set $k = k+1$ and go back to Step 2.

Using the optimal control expression (8), the HJB equation turns to be the form

$$\begin{aligned} 0 &= U\big(x, u^*(x)\big) + \big(\nabla J^*(x)\big)^{\mathsf{T}}\big[f(x) + g(x)u^*(x)\big] \\ &= H\big(x, u^*(x), \nabla J^*(x)\big), J^*(0) = 0. \end{aligned} \tag{9}$$

We notice that the optimal control law can be derived if the optimal cost function can be obtained, i.e., (9) is solvable. However, that is not the case. Since the continuous-time HJB equation (9) is difficult to deal with in theory, it is not an easy task to obtain the optimal control law (8) for general nonlinear systems. This promotes the investigation of iterative algorithms, such as policy iteration. We first construct two sequences in terms of the cost function $\{J^{(k)}(x)\}$ and the control law $\{u^{(k)}(x)\}$, and then start iterating from an initial admissible controller as follows:

$$u^{(0)}(x) \to J^{(1)}(x) \to u^{(1)}(x) \to J^{(2)}(x) \to \cdots \tag{10}$$

Generally, the policy iteration includes two important iterative steps [13], i.e., policy evaluation based on (4) and policy improvement based on (8), which are shown in Algorithm 1.

Note that the above policy iteration algorithm can finally converge to the optimal cost function and optimal control law, i.e., $J^{(k)}(x) \to J^*(x)$ and $u^{(k)}(x) \to u^*(x)$ as $k \to \infty$. The convergence proof has been given in [28], [82], and related references therein. However, it is still difficult to obtain the exact solution of the Lyapunov equation. This motivates us to develop an approximate strategy to overcome the difficulty [66], [68], [69], [76], [77], [81]–[85], [120], [126], [131], which results in the ADP-based neural control design. Besides, the knowledge of system dynamics $f(x)$ and $g(x)$ is needed to perform the iterative process. Actually, some advanced methods have been proposed to relax this requirement, such as the integral policy iteration algorithm [77], the neural identification scheme [120], and the probing signal method [131], [135]. As discussed in the following sections, great efforts are still being made in this aspect.

B. Neural Control Design With Stability Discussion

As is shown in Section I, several neural networks are often incorporated in adaptive critic designs. Among them, the critic network is regarded as the most fundamental element, even though there may be other elements involved, such as model network [52], [55] and action network [55], [66]. Different configurations reflect distinct objectives of control designers. The single critic structure is often employed to emphasize the simplicity of the design procedure [118], [121].

During the neural network implementation, we take the universal approximation property into consideration and express the optimal cost function $J^*(x)$ on the compact set Ω as

$$J^*(x) = \omega_c^{\mathsf{T}}\sigma_c(x) + \varepsilon_c(x) \tag{13}$$

where $\omega_c \in \mathbb{R}^{l_c}$ is the ideal weight vector, l_c is the number of neurons in the hidden layer, $\sigma_c(x) \in \mathbb{R}^{l_c}$ is the activation function, and $\varepsilon_c(x) \in \mathbb{R}$ is the reconstruction error.[3] Then, the gradient vector of the optimal cost function is

$$\nabla J^*(x) = (\nabla\sigma_c(x))^{\mathsf{T}}\omega_c + \nabla\varepsilon_c(x). \tag{14}$$

Since the ideal weight is unknown, a critic neural network is developed to approximate the optimal cost function as

$$\hat{J}^*(x) = \hat{\omega}_c^{\mathsf{T}}\sigma_c(x) \tag{15}$$

where $\hat{\omega}_c \in \mathbb{R}^{l_c}$ denotes the estimated weight vector. Similarly, we derive the gradient vector as

$$\nabla\hat{J}^*(x) = (\nabla\sigma_c(x))^{\mathsf{T}}\hat{\omega}_c. \tag{16}$$

Note that the specific structure of the critic network is always an experimental choice with engineering experience and intuition after noticing a tradeoff between control accuracy and computational complexity [28]. Actually, selecting the proper neurons for neural networks is more of an art than science [30]. Determining the number of neurons needed for a particular application is still an open problem.

Considering the feedback formulation (8) and the neural network expression (13), the optimal control law can be rewritten as a weight-related form

$$u^*(x) = -\frac{1}{2}R^{-1}g^{\mathsf{T}}(x)\big[(\nabla\sigma_c(x))^{\mathsf{T}}\omega_c + \nabla\varepsilon_c(x)\big]. \tag{17}$$

Using the critic neural network (15), the approximate optimal feedback control function is[4]

$$\hat{u}^*(x) = -\frac{1}{2}R^{-1}g^{\mathsf{T}}(x)(\nabla\sigma_c(x))^{\mathsf{T}}\hat{\omega}_c. \tag{18}$$

Based on the neural network formulation, the approximate Hamiltonian is written as

$$\begin{aligned} &\hat{H}\Big(x, \hat{u}^*(x), \nabla\hat{J}^*(x)\Big) \\ &\quad = U\big(x, \hat{u}^*(x)\big) + \hat{\omega}_c^{\mathsf{T}}\nabla\sigma_c(x)\big[f(x) + g(x)\hat{u}^*(x)\big]. \end{aligned} \tag{19}$$

[3] For most of the general nonlinear cases, the ideal vector ω_c and the ideal scalar ε_c are unknown but they are both bounded.

[4] The control law function is directly computed as a closed-loop expression of the critic weight vector in this single network structure. An additional action network is built when implementing the synchronous policy iteration algorithm [66], [88] to improve the sequential updates [77], [87] in terms of saving computation time and avoiding dynamics knowledge.

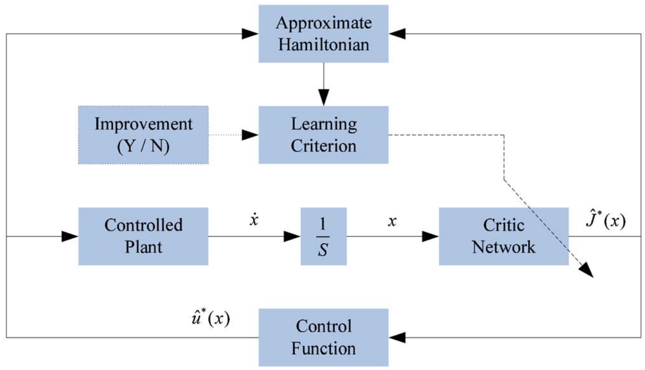

Fig. 1. ADP-based learning process and optimal control design diagram. The solid line represents the signal flow while the dashed line denotes the neural network back-propagating path. The dotted component indicates whether there is an improvement module added to the learning criterion. If it is set to "N," there is no improvement and it is actually the traditional learning rule (21). If it is set to "Y," there will be an improved module (discussed later) during the learning process.

Noticing (9), we define the error as

$$e_c = \hat{H}\Big(x, \hat{u}^*(x), \nabla\hat{J}^*(x)\Big) - H\big(x, u^*(x), \nabla J^*(x)\big) \quad (20)$$

so that the scalar $e_c = \hat{H}(x, \hat{u}^*(x), \nabla\hat{J}^*(x))$. As given in [26], [28], [66], and [82], we define $\partial e_c/\partial\hat{\omega}_c \triangleq \phi \in \mathbb{R}^{l_c}$ and find that the set $\{\phi_1, \phi_2, \ldots, \phi_{l_c}\}$ is linearly independent.

Now, we show how to train the critic network and design the weight vector $\hat{\omega}_c$ to minimize the objective function normally defined as $E_c = (1/2)e_c^2$. Traditionally, based on (19), we can employ the normalized steepest descent algorithm

$$\dot{\hat{\omega}}_c = -\alpha_c \frac{1}{\big(1+\phi^{\mathsf{T}}\phi\big)^2}\left(\frac{\partial E_c}{\partial\hat{\omega}_c}\right) = -\alpha_c \frac{\phi}{\big(1+\phi^{\mathsf{T}}\phi\big)^2} e_c \quad (21)$$

to tune the weight vector, where the constant $\alpha_c > 0$ is the learning rate while the term $(1+\phi^{\mathsf{T}}\phi)^2$ is adopted for normalization. The simple diagram of the ADP-based controller design method is depicted in Fig. 1, where (21) is the basic learning criterion of the neural network.

Defining the error vector between the ideal weight and the estimated value as $\tilde{\omega}_c = \omega_c - \hat{\omega}_c$, we can easily find that $\dot{\tilde{\omega}}_c = -\dot{\hat{\omega}}_c$. Here, let us introduce two new variables $\phi_1 = \phi/(1+\phi^{\mathsf{T}}\phi)$ and $\phi_2 = 1+\phi^{\mathsf{T}}\phi$ with $\phi_1 \in \mathbb{R}^{l_c}$ and $\phi_2 \geq 1$. Then, by using the tuning rule (21), we derive that the critic weight error dynamics can be formulated as

$$\dot{\tilde{\omega}}_c = -\alpha_c\phi_1\phi_1^{\mathsf{T}}\tilde{\omega}_c + \alpha_c\frac{\phi_1}{\phi_2}e_{cH} \quad (22)$$

where the scalar term e_{cH} represents the residual error due to neural network approximation.

In adaptive critic designs, we intend to identify the parameters of the critic network so as to approximate the optimal cost function. As commonly required within the adaptive control field [111], the persistence of excitation assumption is naturally needed during adaptive critic learning. Note that based on [66] and [67], the persistence of excitation condition ensures that $\lambda_{\min}(\phi_1\phi_1^{\mathsf{T}}) > 0$, which is significant to perform the closed-loop stability analysis. The following assumption is commonly used such as in [66], [68], [69], [71], and [81].

Assumption 1: The control matrix $g(x)$ is upper bounded such that $\|g(x)\| \leq \lambda_g$, where λ_g is a positive constant. On the compact set Ω, the terms $\nabla\sigma_c(x)$, $\nabla\varepsilon_c(x)$, and e_{cH} are all upper bounded such that $\|\nabla\sigma_c(x)\| \leq \lambda_\sigma$, $\|\nabla\varepsilon_c(x)\| \leq \lambda_\varepsilon$, and $|e_{cH}| \leq \lambda_e$, where λ_σ, λ_ε, and λ_e are positive constants.

Definition 2 [69], [71], [120]: For a nonlinear system $\dot{x} = f(x(t))$, its solution is said to be UUB, if there exists a compact set $\Omega \subset \mathbb{R}^n$ such that for all $x_0 \in \Omega$, there exist a bound Λ and a time $T(\Lambda, x_0)$ such that $\|x(t) - x_e\| \leq \Lambda$ for all $t \geq t_0 + T$, where x_e is an equilibrium point.

Lemma 1 [66]: For system (1) and the constructed neural network (15), we suppose that Assumption 1 holds. The approximate optimal control law is given by (18) and the critic network is tuned based on (21). Then, the closed-loop system state and the critic weight error are UUB.

The UUB stability actually implies that after a transition period T, the state vector remains within the ball of radius Λ around the equilibrium point. Note that the proof of such UUB stability is performed by employing the well-known Lyapunov approach. Based on Lemma 1, the critic weight error $\tilde{\omega}_c$ is upper bounded by a finite constant. Then, according to (17) and (18), we can find that

$$\big\|u^*(x) - \hat{u}^*(x)\big\| = \frac{1}{2}\big\|R^{-1}g^{\mathsf{T}}(x)\big[(\nabla\sigma_c(x))^{\mathsf{T}}\tilde{\omega}_c + \nabla\varepsilon_c(x)\big]\big\| \quad (23)$$

is also upper bounded. This implies that the near-optimal controller $\hat{u}^*(x)$ can converge to a neighborhood of the optimal value $u^*(x)$ with a finite bound. Besides, this bound can be set adequately small by adjusting the related parameters like the critic learning rate.

It is also worth mentioning that the previous ADP-based optimal regulation method provides the basis for further adaptive critic control designs. Note that the dynamical uncertainties are not included in system (1). Considering the universality of the uncertain phenomenon, it is indeed necessary to extend the ADP-based optimal control design approach to robust stabilization problems and investigate the robustness of ADP-based controllers under uncertain environment.

III. ADP for Nonlinear Robust Control Design With Matched Uncertainties

This section mainly presents the results about ADP-based robust control design for matched uncertain nonlinear systems [116]–[133]. There are several categories of ADP-based robust control strategies, such as the least-square-based problem transformation method [116], [117], adaptive-design-based problem transformation method [118]–[125], data-based problem transformation method [126], [127], the combined sliding mode control method [128], and the robust ADP method [129]–[133]. We will not only exhibit the robustness of the optimal controller with respect to the nominal system but also discuss the optimality of the robust controller. Actually, some of these methods [116], [117], [124], [130]–[133] can be applied to unmatched robust control design.

A. Problem Transformation Method

If dynamical uncertainties are brought into system (1) by various changes during the operation of the controlled plant, we have to pay attention to the robustness of the designed controller. We consider a class of continuous-time nonlinear systems subjected to uncertainties and described by

$$\dot{x}(t) = f(x(t)) + g(x(t))[u(t) + d(x(t))] \quad (24)$$

where the term $g(x)d(x)$ reflects a kind of dynamical uncertainties matched with the control matrix. We assume $d(0) = 0$, so as to keep $x = 0$ as an equilibrium of the controlled plant. It is often assumed that the term $d(x)$ is bounded by a known function $d_M(x)$, i.e., $\|d(x)\| \leq d_M(x)$ with $d_M(0) = 0$.

Considering the uncertain nonlinear system (24), for coping with the robust stabilization problem, we should design a control law $u(x)$, such that the closed-loop state vector is stable with respect to dynamical uncertainties. In this section, by adopting a positive constant ρ and specifying $Q(x) = \rho d_M^2(x)$, we show that the robust control problem can be addressed by designing the optimal controller of the nominal plant (1), where the cost function is still given by (3) and the modified utility is selected as

$$U^R(x(t), u(t)) = \rho d_M^2(x(t)) + u^{\mathsf{T}}(t)Ru(t). \quad (25)$$

Note that in this situation, the optimal control function is kept unchanged even if the modified utility is employed. For system (1) and cost function (3) with modified utility function (25), the Hamiltonian becomes

$$\begin{aligned} H^R(x, u(x), \nabla J(x)) &= \rho d_M^2(x) + u^{\mathsf{T}}(x)Ru(x) \\ &\quad + (\nabla J(x))^{\mathsf{T}}[f(x)+g(x)u(x)]. \end{aligned} \quad (26)$$

Observing the modified utility function (25) and using the optimal control law (8) again, the HJB equation with respect to the modified optimal control problem becomes

$$\begin{aligned} 0 &= \rho d_M^2(x) + \left(\nabla J^*(x)\right)^{\mathsf{T}} f(x) \\ &\quad - \frac{1}{4}\left(\nabla J^*(x)\right)^{\mathsf{T}} g(x)R^{-1}g^{\mathsf{T}}(x)\nabla J^*(x) \\ &= H^R\left(x, u^*(x), \nabla J^*(x)\right), J^*(0) = 0. \end{aligned} \quad (27)$$

We first show the stability of the closed-loop form of the nominal system based on the approximate optimal control law.

Theorem 1 [118]: For the nominal system (1) and cost function (3) with modified utility function (25), the approximate optimal control law obtained by (18) guarantees that the closed-loop system state is UUB.

Then, we show how to guarantee the robust stabilization of the matched uncertain system (24) based on the designed near-optimal control law.

Theorem 2 [123]: For the nominal system (1) and cost function (3) with modified utility function (25), the approximate optimal control obtained by (18) guarantees that the closed-loop form of the uncertain nonlinear plant (24) possesses UUB stability if $\rho > \lambda_{\max}(R)$.

Theorems 1 and 2 exhibit the closed-loop UUB stability of the nominal plant (1) and uncertain plant (24), respectively, when applying the designed near-optimal control law (18). One should pay special attention to the fact that the closed-loop form of the uncertain plant is UUB when using the approximate optimal controller, not the same as the asymptotic stability result when adopting exactly the optimal controller [118]. The proof is performed via the Lyapunov stability theory by regarding $J^*(x)$ as the Lyapunov function candidate.[5]

Next, we discuss the optimality of the robust controller by adding a feedback gain π to the optimal feedback control law (8) of system (1) such that

$$\bar{u}(x) = \pi u^*(x) = -\frac{1}{2}\pi R^{-1}g^{\mathsf{T}}(x)\nabla J^*(x). \quad (28)$$

As is shown in [119], [120], and [126], the feedback control law computed by (28) ensures the closed-loop form of system (1) to be asymptotically stable if $\pi \geq 1/2$. Moreover, there exists a positive number $\pi_1^* \geq 1$, such that when the gain value $\pi > \pi_1^*$, the control law derived by (28) ensures that the closed-loop form of the uncertain system (24) is also asymptotically stable (i.e., achieves robustness).

For system (24), we define a cost function as [119], [120]

$$\bar{J}(x_0) = \int_0^{\infty} \left\{ \bar{Q}(x(\tau)) + \frac{1}{\pi}\bar{u}^{\mathsf{T}}(x(\tau))R\bar{u}(x(\tau)) \right\} \mathrm{d}\tau \quad (29)$$

where the new state-related utility is

$$\begin{aligned} \bar{Q}(x) &= d_M^2(x) - \left(\nabla J^*(x)\right)^{\mathsf{T}} g(x)\bar{d}(x) \\ &\quad + \frac{1}{4}(\pi - 1)\left(\nabla J^*(x)\right)^{\mathsf{T}} g(x)R^{-1}g^{\mathsf{T}}(x)\nabla J^*(x) \end{aligned} \quad (30)$$

and the term $\bar{d}(x)$ therein satisfying $d(x) = R^{1/2}\bar{d}(x)$. By introducing $(1/(\pi - 1))d^{\mathsf{T}}(x)d(x)$ to (30) and considering the condition $\|d(x)\| \leq d_M(x)$, we can obtain the inequality

$$\bar{Q}(x) \geq \frac{\pi - 2}{\pi - 1} d_M^2(x). \quad (31)$$

It is clear that there exists a positive number $\pi_2^* \geq 2$ rendering the function $\bar{Q}(x)$ to be positive definite when $\pi > \pi_2^*$. In this sense, the cost function (29) for the uncertain system (24) is well defined.

Theorem 3 [119], [120]: Considering system (24) and the new cost function (29), there exists a positive number $\pi^* \triangleq \max\{\pi_1^*, \pi_2^*\}$ such that the control law (28) achieves optimality if the feedback gain $\pi > \pi^*$. That is to say, (28) is the robust optimal control law of the uncertain dynamics plus a specified cost function.

Here, we find that the value of the feedback gain π can affect the control performance of the nominal and uncertain systems. To be clear, the relationship between the feedback gain and the controller achievement can be seen in Table I.

According to Theorem 3, we should perform the optimal control design regarding the nominal plant and then attain the robust optimal feedback stabilization of the original system. Therefore, we can employ the ADP method to design the robust optimal controller using actor-critic structure and neural network technique.

[5] According to the definition of the optimal cost function (6), $J^*(x) > 0$ for any $x \neq 0$ and $J^*(x) = 0$ when $x = 0$, which means that $J^*(x)$ is a positive definite function.

TABLE I
RELATIONSHIP BETWEEN THE GAIN AND CONTROLLER ACHIEVEMENT

Feedback Gain	Controller Achievement
$\pi > 1/2$	Stabilizing controller of the nominal system
$\pi = 1$	Optimal controller of the nominal system
$\pi > \pi_1^* \geq 1$	Robust controller of the uncertain system
$\pi > \pi^* \geq 2$	Robust optimal controller of the uncertain system

Algorithm 2 Model-Free Integral Policy Iteration Scheme

1: Initialization
Let the initial iteration index be $k = 0$ and $J^{(0)}(\cdot) = 0$. Give a small positive number ϵ as the stopping threshold. Start iterating from an initial admissible control law $u^{(0)}$.

2: Policy Evaluation and Improvement
Based on the control law $u^{(k)}(x)$, solve $J^{(k+1)}(x)$ and $u^{(k+1)}(x)$ simultaneously from the integral equation

$$\begin{aligned} J^{(k+1)}(x(t+T)) - J^{(k+1)}(x(t)) &= -2\int_t^{t+T} u^{(k+1)\mathsf{T}}(\tau)R\vartheta(\tau)\mathrm{d}\tau \\ &\quad - \int_t^{t+T}\left\{\rho d_M^2(x(\tau)) + u^{(k)\mathsf{T}}(\tau)Ru^{(k)}(\tau)\right\}\mathrm{d}\tau. \end{aligned} \quad (34)$$

3: Stopping Criterion
If $|J^{(k+1)}(x) - J^{(k)}(x)| \leq \epsilon$, stop and obtain the approximate optimal control law; else, set $k = k + 1$ and go back to Step 2.

B. Other ADP-Based Robust Control Methods

To reduce the requirement of the nominal dynamics, the integral policy iteration algorithm [107]–[109] can be employed to develop the ADP-based robust controller [126], [127]. To this end, we should consider the nonlinear system explored by a known bounded probing signal $\vartheta(t)$ given as follows:

$$\dot{x}(t) = f(x(t)) + g(x(t))[u(t) + \vartheta(t)]. \quad (32)$$

The online model-free integral policy iteration scheme is given in Algorithm 2. Different from the Algorithm 1, it iterates from $k = 0$ with the following mode:

$$u^{(0)}(x) \to \left\{J^{(1)}(x), u^{(1)}(x)\right\} \to \left\{J^{(2)}(x), u^{(2)}(x)\right\} \to \cdots \quad (33)$$

Since the terms $f(x)$ and $g(x)$ do not appear in the integral equation (34), it is significant to find that the integral policy iteration can be conducted without using the system dynamics.

In [128], the combined sliding mode controller is designed as $u = u^{\mathrm{a}} + u^{\mathrm{s}}$, where the former part u^{a} is the ADP-based control law used to stabilize the sliding mode dynamics and guarantee a nearly optimal performance while the latter part u^{s} is a discontinuous control action designed to reduce the effect of disturbance and ensure the reachability of the sliding manifold. It incorporates the idea of sliding mode control and extends the results of [118], [120], [121], [124], and [125].

The robust ADP method [129]–[133] can be viewed as an important extension of classical ADP to linear and genuinely nonlinear systems with dynamical uncertainties. The backstepping, robust redesign, and small-gain techniques in modern nonlinear control theory are incorporated into the robust ADP method, such that the system model is input-to-state stable with an arbitrarily small gain [134]. In [131], a class of genuinely nonlinear systems were considered with the form

$$\dot{\varsigma} = \delta_\varsigma(\varsigma, x) \quad (35a)$$

$$\dot{x} = f(x) + g(x)[u + \delta(\varsigma, x)] \quad (35b)$$

where ς is the unmeasurable part of the state, δ_ς and δ are unknown locally Lipschitz functions. The design objective is to find an online control law that stabilizes the uncertain system at the origin. Moreover, in the absence of the dynamic uncertainty (i.e., $\delta = 0$ and the ς-subsystem is absent), the designed control law becomes the optimal controller that minimizes the cost function of the nominal system. Here, the robustness is for the uncertain system while the optimality is discussed with the nominal system. Furthermore, the robust ADP methodology is also extended to nonlinear systems with unmatched dynamic uncertainties [131] and subsequently to large-scale systems [132], [133]. Hence, systematic robust ADP-based online learning algorithms have been proposed to derive stabilizing controllers with appropriate optimality.

At the end of this section, we present the comparison of several ADP-based robust control methods, which is shown in Table II, with the uncertain term and main techniques included.

IV. ADP FOR NONLINEAR GUARANTEED COST CONTROL DESIGN WITH UNMATCHED UNCERTAINTIES

Section III mainly focuses on the ADP-based robust control of nonlinear systems with matched uncertainties, which does not represent the general situation. We should also consider uncertain nonlinear systems with unmatched uncertainties. Though in [116], [117], [124], and [130]–[133], the proposed robust control methods are applicable to nonlinear systems with unmatched uncertainties, only the robustness is discussed, which does not include the cost function with respect to the uncertain plant. In guaranteed cost control design, we not only concern with the robustness, but also pay attention to the boundedness of the corresponding cost function. Based on [137]–[141], we revisit ADP method for nonlinear guaranteed cost control design in this section.

Consider a class of continuous-time uncertain nonlinear dynamical systems given by

$$\dot{x}(t) = f(x(t)) + g(x(t))u(t) + \Delta f(x(t)) \quad (36)$$

where $\Delta f(x(t))$ is the nonlinear perturbation of the corresponding nominal system formed as (1). Before proceeding, we give an assumption to the system uncertainty as used in [137], [138], [150], and [151].

Assumption 2: The dynamical uncertainty $\Delta f(x)$ satisfies

$$\Delta f(x) = G(x)f_G(\varphi(x)) \quad (37a)$$

$$f_G^{\mathsf{T}}(\varphi(x))f_G(\varphi(x)) \leq y^{\mathsf{T}}(\varphi(x))y(\varphi(x)) \quad (37b)$$

where $G(\cdot) \in \mathbb{R}^{n\times r}$ and $\varphi(\cdot)$ with $\varphi(0) = 0$ are known functions showing the architecture of uncertain term, $f_G(\cdot) \in \mathbb{R}^r$ is

TABLE II
COMPARISON OF SEVERAL ADP-BASED ROBUST CONTROL METHODS, INCLUDING THE UNCERTAIN TERM AND THE MAIN TECHNIQUES THEREIN

ADP-Based Control Method	Uncertain Term	Main Techniques
Least-squares-based problem transformation [116], [117]	Matched or unmatched term	Least squares method and neural network approximation
Adaptive-design-based problem transformation [118]–[125]	Matched or unmatched term	Adaptive design, persistence of excitation, critic network
Data-based problem transformation method [126], [127]	Matched term	Integral reinforcement learning and critic network
Combined sliding mode control method [128]	Matched term	Integral sliding mode control, critic, and action network
Robust ADP method [129]–[133]	Matched or unmatched term	Backstepping, robust redesign, and small-gain theorem

an uncertain function satisfying $f_G(0) = 0$, and $y(\cdot) \in \mathbb{R}^r$ is a given function with $y(0) = 0$.

We consider system (36) with cost function defined as in (3) and utility function given by (2). In order to handle the guaranteed cost control design, we should derive a feedback control law $u(x)$ and determine an upper bound function $\Phi(u)$, such that the closed-loop system is robustly stable and meanwhile the related cost function satisfies $J \leq \Phi$. Note that $\Phi(u)$ is called the guaranteed cost function. Only when $\Phi(u)$ is minimized, it becomes the optimal guaranteed cost and is denoted as Φ^*. Besides, the corresponding controller $\breve{u}^*$ is called the optimal guaranteed cost control law. In this sense, we focus on deriving $\Phi^* = \min_u \Phi(u)$ and $\breve{u}^* = \arg\min_u \Phi(u)$.

According to [137] and [138], it has been proven that designing the optimal guaranteed cost controller of system (36) can be transformed into deriving the optimal controller of the nominal system (1) and the guaranteed cost of the uncertain nonlinear dynamics is closely related to the modified cost function of the nominal plant. These facts can be verified from the following lemma, which is derived by rechecking and relaxing the conditions of [152].

Lemma 2 [137], [138]: Assume that there exist a continuously differentiable cost function $V(x)$ satisfying $V(x) > 0$ for all $x \neq 0$ and $V(0) = 0$, a bounded function $\Gamma(x)$ satisfying $\Gamma(x) \geq 0$, as well as a feedback control function $u(x)$ such that

$$(\nabla V(x))^{\mathsf{T}} \Delta f(x) \leq \Gamma(x) \tag{38a}$$
$$U(x,u) + \Gamma(x) + (\nabla V(x))^{\mathsf{T}}(f + gu) = 0. \tag{38b}$$

Then, under the action of the feedback control function $u(x)$, there exists a neighborhood of the origin such that the original uncertain system (36) is asymptotically stable. Moreover

$$J(x(t),u) \leq V(x(t)) = \breve{J}(x(t),u) \tag{39}$$

where $\breve{J}(x(t),u)$ is defined by

$$\breve{J}(x(t),u) = \int_t^{\infty} \big\{U(x(\tau),u(x(\tau))) + \Gamma(x(\tau))\big\}\mathrm{d}\tau \tag{40}$$

as the modified cost function of system (1).

Lemma 2 exhibits the existence of the guaranteed cost function with respect to the uncertain plant (36). Actually, the function $\Gamma(x)$ suitably bounds the term $(\nabla V(x))^{\mathsf{T}}\Delta f(x)$, which is important to design the optimal guaranteed cost controller.

TABLE III
UTILITY FUNCTIONS OF THE DIFFERENT CONTROL TOPICS

Topic	Utility
Normal optimal control	$U(x,u) = Q(x) + u^{\mathsf{T}}Ru$
Matched robust control	$U^{\mathrm{R}}(x,u) = \rho d_M^2(x) + u^{\mathsf{T}}Ru$
Guaranteed cost control	$U^{\mathrm{G}}(x,u) = U(x,u) + \Gamma(x)$

For providing a specific form of $\Gamma(x)$, we define[6]

$$\Gamma(x) = y^{\mathsf{T}}(\varphi(x))y(\varphi(x)) + \frac{1}{4}(\nabla V(x))^{\mathsf{T}}G(x)G^{\mathsf{T}}(x)\nabla V(x) \tag{41}$$

based on [137], [138], and [150]–[152] and find that (38a) is satisfied according to Assumption 2. Moreover, we ought to minimize the upper bound function $\breve{J}(x_0,u)$ regarding u so as to determine the optimal guaranteed cost controller. It also means that, the effort should be put on designing the optimal controller of system (1), where $V(x(t)) = \breve{J}(x(t),u)$ is seen as the cost function and $U^{\mathrm{G}}(x,u) = U(x,u) + \Gamma(x)$ is regarded as the utility function. The comparison of different utility functions of normal optimal control, matched robust control, and guaranteed cost control is given in Table III. Note that the choice of the utility function is not unique. Fox example, in matched robust control design, one can also select $\rho d_M^2(x) + U(x,u)$ as the utility [127]. In many situations, this distinction just reflects the objective and interest of designers.

For system (1) and cost function (40), we can obtain

$$U(x,u) + \Gamma(x) + \big(\nabla\breve{J}(x)\big)^{\mathsf{T}}(f + gu) = 0. \tag{42}$$

Clearly, (42) is formed the same as (38b). Hence, (38b) or (42) is an infinitesimal version of the modified cost function (40) and is nothing but the nonlinear Lyapunov equation. In such situation, we define the Hamiltonian as the following form:

$$H^{\mathrm{G}}(x,u(x),\nabla\breve{J}(x)) = U(x,u(x)) + \Gamma(x) + \big(\nabla\breve{J}(x)\big)^{\mathsf{T}}[f(x) + g(x)u(x)]. \tag{43}$$

The optimal cost function is defined similarly as (6) and the optimal feedback controller is still formed as (8), where the modified HJB equation of this situation becomes

$$0 = U(x,u^*) + (\nabla\breve{J}^*(x))^{\mathsf{T}}\big(f + gu^*\big) + y^{\mathsf{T}}(\varphi(x))y(\varphi(x)) + \frac{1}{4}\big(\nabla\breve{J}^*(x)\big)^{\mathsf{T}}G(x)G^{\mathsf{T}}(x)\nabla\breve{J}^*(x) \tag{44}$$

[6]The form of $\Gamma(x)$ is not unique. One can also introduce an adjustable positive coefficient to build a different bounded function and then define a new utility (and cost function) and subsequently construct the parameterized HJB equation [139].

with $\breve{J}^*(0) = 0$. The following theorem exhibits how to derive the optimal guaranteed cost controller for system (36).

Theorem 4 [137], [138]: Consider the uncertain system (36) with cost function (3) and the corresponding nominal system (1) with cost function (40). Suppose that the modified HJB equation (44) has a continuously differentiable solution $\breve{J}^*(x)$. Then, for any $u \in \mathscr{A}(\Omega)$, the cost function (3) satisfies $J(x_0, u) \leq \Phi(u)$, where

$$\Phi(u) \triangleq \breve{J}^*(x_0) + \int_0^\infty \big(u - u^*\big)^{\mathsf{T}} R\big(u - u^*\big)\mathrm{d}\tau. \tag{45}$$

Furthermore, the optimal guaranteed cost function of the original nonlinear system is $\Phi^* = \Phi(u^*) = \breve{J}^*(x_0)$. Meanwhile, the optimal guaranteed cost control law is just $\breve{u}^* = u^*$.

According to Theorem 4, once the modified HJB equation (44) with respect to system (1) is solved, we can construct the optimal guaranteed cost control strategy of the uncertain plant (36). The ADP-based method can be employed to serve as the important role of solving the modified optimal control problem. Note that the finite-horizon guaranteed cost control [139] and guaranteed cost tracking control [140] are also studied under the framework of ADP.

V. ADP for Nonlinear Decentralized Control Design With Matched Interconnections

In this section, we present how to apply ADP method to large-scale systems by designing the decentralized controller for nonlinear dynamics with matched and bounded interconnections [119], [144]–[147]. This is also closely related to the ADP-based robust control design. Note that in this section, the subscript symbol i denotes the ith subsystem.

Consider a nonlinear system composed of N subsystems with interconnections given by

$$\dot{x}_i(t) = f_i(x_i(t)) + g_i(x_i(t))\big[\bar{u}_i(t) + \mathcal{I}_i(\mathcal{X}(t))\big], i \in \mathbb{N}^+ \tag{46}$$

where $x_i(t) \in \Omega_i \subset \mathbb{R}^{n_i}$ and $\bar{u}_i(t) \in \Omega_{ui} \subset \mathbb{R}^{m_i}$ are the state variable and the control variable of the ith subsystem, respectively, and $\mathcal{X} = [x_1^{\mathsf{T}}, x_2^{\mathsf{T}}, \ldots, x_N^{\mathsf{T}}]^{\mathsf{T}} \in \mathbb{R}^{\mathcal{N}}$ is the overall state with $\mathcal{N} = n_1 + n_2 + \cdots + n_N$. Note that for the subsystem i, $f_i(x_i)$, $g_i(x_i)$, and $g_i(x_i)\mathcal{I}_i(\mathcal{X})$ stand for the nonlinear internal dynamics, the control function matrix, and the interconnected term, respectively. Here, $x_1, x_2, \ldots, x_N$ are called local system states while $\bar{u}_1, \bar{u}_2, \ldots, \bar{u}_N$ are called local control inputs. Let $x_i(0) = x_{i0}$ be the initial state vector with respect to the ith subsystem, $i \in \mathbb{N}^+$.

For the interconnected terms, we assume that

$$\|\mathcal{I}_i(\mathcal{X})\| \leq \sum_{\ell=1}^{N} \beta_{i\ell}\hbar_{i\ell}(x_\ell), i \in \mathbb{N}^+ \tag{47}$$

where $\beta_{i\ell}, i, \ell \in \mathbb{N}^+$ are non-negative constants and $\hbar_{i\ell}(x_\ell), i, \ell \in \mathbb{N}^+$ are positive semidefinite functions. Defining $\hbar_\ell(x_\ell) = \max\{\hbar_{1\ell}(x_\ell), \hbar_{2\ell}(x_\ell), \ldots, \hbar_{N\ell}(x_\ell)\}$, $\ell \in \mathbb{N}^+$, we further obtain the relationship

$$\|\mathcal{I}_i(\mathcal{X})\| \leq \sum_{\ell=1}^{N} \bar{\beta}_{i\ell}\hbar_\ell(x_\ell), i \in \mathbb{N}^+ \tag{48}$$

which satisfies $\bar{\beta}_{i\ell}\hbar_\ell(x_\ell) \geq \beta_{i\ell}\hbar_{i\ell}(x_\ell)$ with $\bar{\beta}_{i\ell}$, $i, \ell \in \mathbb{N}^+$ being non-negative constants. Note that (48) is important to perform adaptive decentralized control design since it relates the interconnection term with a combination of separate terms corresponding to each subsystem.

We focus on finding the decentralized feedback control strategy of system (46). To this end, we should derive N state feedback control laws $\bar{u}_1(x_1), \bar{u}_2(x_2), \ldots, \bar{u}_N(x_N)$, such that the constituted control pair $(\bar{u}_1(x_1), \bar{u}_2(x_2), \ldots, \bar{u}_N(x_N))$ can stabilize system (46). It has been proven in [144], the decentralized control strategy can be developed through tackling the optimal feedback stabilization with respect to N isolated subsystems described by

$$\dot{x}_i(t) = f_i(x_i(t)) + g_i(x_i(t))u_i(t), i \in \mathbb{N}^+. \tag{49}$$

Note that the basic assumptions with respect to the interconnected plant (46) and the isolated plants (49), in terms of equilibrium, differentiability, Lipschitzness, and controllability, can be found by referring to [144]. The designed feedback control $u_i(x_i)$ should be admissible with $u_i \in \mathscr{A}_i(\Omega_i)$, $i \in \mathbb{N}^+$, which is defined similar as $\mathscr{A}(\Omega)$ but considering the subsystem symbol. Letting $\hbar_i(x_i) \leq Q_i(x_i)$ and according to [144], we can derive a set of optimal feedback control laws $u_i^*(x_i)$, $i \in \mathbb{N}^+$ to minimize the local cost functions

$$J_i(x_i(t), u_i(t)) = \int_t^\infty U_i^D(x_i(\tau), u_i(\tau))\mathrm{d}\tau, i \in \mathbb{N}^+ \tag{50}$$

where the utility is

$$U_i^D(x_i(t), u_i(t)) = Q_i^2(x_i(t)) + u_i^{\mathsf{T}}(t)R_i u_i(t) \tag{51}$$

and $R_i = R_i^{\mathsf{T}} > 0$. Note that when starting from $t = 0$, these cost functions becomes $J_i(x_i(0))$, i.e., $J_i(x_{i0})$, $i \in \mathbb{N}^+$. Then, using the symbol of optimal cost functions $J_i^*(x_i)$, that is

$$J_i^*(x_i) = \min_{u_i \in \mathscr{A}_i(\Omega_i)} J_i(x_i, u_i), i \in \mathbb{N}^+ \tag{52}$$

and considering the expression of the optimal control laws

$$u_i^*(x_i) = -\frac{1}{2}R_i^{-1}g_i^{\mathsf{T}}(x_i)\nabla J_i^*(x_i), i \in \mathbb{N}^+ \tag{53}$$

the HJB equations of isolated subsystems are given as

$$\begin{aligned} 0 &= Q_i^2(x_i) + \big(\nabla J_i^*(x_i)\big)^{\mathsf{T}} f_i(x_i) \\ &\quad - \frac{1}{4}\big(\nabla J_i^*(x_i)\big)^{\mathsf{T}} g_i(x_i)R_i^{-1}g_i^{\mathsf{T}}(x_i)\nabla J_i^*(x_i), i \in \mathbb{N}^+ \end{aligned} \tag{54}$$

with $J_i^*(0) = 0$. The main decentralized stabilization result is shown as follows.

Theorem 5 [144]: For isolated subsystems (49) and cost functions (50), the optimal control laws are given by (53). There exist N positive numbers, $\zeta_1, \zeta_2, \ldots, \zeta_N$, such that the state feedback control laws

$$\bar{u}_i(x_i) = \zeta_i u_i^*(x_i) = -\frac{1}{2}\zeta_i R_i^{-1}g_i^{\mathsf{T}}(x_i)\nabla J_i^*(x_i), i \in \mathbb{N}^+ \tag{55}$$

can form a control pair $(\bar{u}_1(x_1), \bar{u}_2(x_2), \ldots, \bar{u}_N(x_N))$, which is the decentralized control scheme of the original interconnected system (46).

TABLE IV
SUMMARY OF ADP METHOD FOR OPTIMAL CONTROL, ROBUST STABILIZATION, GUARANTEED COST CONTROL, AND DECENTRALIZED STABILIZATION

Topic	Plant	Objective
Optimal control (policy iteration)	Nominal systems	Convergence, stability, and optimality of nominal systems
Robust control	Matched or unmatched uncertain systems	Stability and robustness of uncertain systems (sometimes optimality)
Guaranteed cost control	Unmatched uncertain systems	Stability and robustness of uncertain systems (including boundedness)
Decentralized control	Matched interconnected systems	Stability of overall systems (sometimes optimality)

In this circumstance, we point out that for coping with the optimal feedback stabilization, the Hamiltonian of system (49) should be defined as

$$\begin{aligned} &H_i^D(x_i, u_i(x_i), \nabla J_i(x_i)) \\ &\quad = Q_i^2(x_i) + u_i^{\mathsf{T}}(x_i)R_i u_i(x_i) \\ &\qquad + (\nabla J_i(x_i))^{\mathsf{T}}[f_i(x_i) + g_i(x_i)u_i(x_i)],\ i \in \mathbb{N}^+. \end{aligned} \tag{56}$$

Then, we turn to compute the optimal controllers formed as (53) based on the idea of ADP and after that we can construct the decentralized control law. Subsequently, as shown in [145], when the dynamics of isolated subsystems are unknown, the model-free decentralized control scheme of interconnected systems can also be derived. Similar to the robust optimal control design, the decentralized optimal control problem for a class of large-scale systems can be addressed with ADP formulation as well [119].

So far, we have discussed the ADP method for optimal regulation, robust stabilization, guaranteed cost control design, and decentralized stabilization for different kinds of nonlinear plants. A summary can be found in Table IV, describing the important properties, i.e., convergence, stability, optimality, robustness, and boundedness of the four control topics.

Note that in Table IV, the ADP-based robust control cannot always achieve optimality of uncertain systems at the current stage, so it is "sometimes optimality." In addition, the ADP-based guaranteed cost control can also fulfill the boundedness of the guaranteed cost function, so it is "including boundedness." Incidentally, though some expected properties are not pointed out in Table IV, it is not implied that they are unreachable goals. For example, the decentralized control design of interconnected systems with unmatched interconnections is worth performing further study.

VI. ADVANCED TECHNIQUE AND FURTHER DISCUSSION FOR ADP-BASED ROBUST CONTROL DESIGN

In this section, we present an advanced technique for ADP-based nonlinear robust control design to save the communication resource and the further discussion on improving the learning rule of the critic network.

A. Saving the Communication Resource

With the rapid development of network-based systems, more and more control loops are closed through communication mediums. The growing interest in saving the computational load of networked control systems brings an extensive attention to the development of event-triggering mechanism [153], [154]. Using event-driven approaches, the actuators are updated only when certain conditions are satisfied to guarantee the stability performance and control efficiency of the target plants. Hence, it has a good potential to combine event-triggering mechanism with adaptive critic technique, so as to save the computational burden and meanwhile attain intelligent optimization [155]–[162]. A novel optimal adaptive event-triggered method for nonlinear continuous-time systems was proposed based on actor-critic framework and neural network approximation [155]. An event-triggered state feedback neural controller of nonlinear continuous-time systems was designed in [156]. By measuring the input–output data, an event-triggered ADP control approach for continuous-time affine nonlinear systems with unknown internal states was developed in [157]. An event-triggered optimal control method for partially unknown systems with input constraints was proposed based on ADP [159]. Furthermore, by incorporating dynamical uncertainties, the event-based robust control design has also been considered [160]–[162]. Therein, the event-driven adaptive robust control scheme of nonlinear systems with uncertainties via neural dynamic programming was developed. In this part, we focus on discussing how to save the communication resource by using the event-triggered mechanism and aim to establish the event-based adaptive robust control method for nonlinear systems.

Under the framework of event-triggered control mechanism, we define a monotonically increasing sequence of triggering instants $\{s_j\}_{j=0}^{\infty}$, where s_j represents the jth consecutive sampling instant, $j \in \mathbb{N}$. Then, the output of the sampling component is a sequence of sampled state denoted as $\hat{x}_j = x(s_j)$ for all $t \in [s_j, s_{j+1})$. Define the gap function between the current and the sampled states as the event-triggering error

$$e_j(t) = \hat{x}_j - x(t),\ \forall t \in \big[s_j, s_{j+1}\big). \tag{57}$$

During the event-triggered control design, the triggering instants are determined by a triggering condition. An event is triggered when the triggering condition is violated at $t = s_j$. At every triggering instant, the system state is sampled so that the event-triggering error $e_j(t)$ is reset to zero, and then, the feedback control law $u(x(s_j)) = u(\hat{x}_j) \triangleq \mu(\hat{x}_j)$ is updated. Note that the control sequence $\{\mu(\hat{x}_j)\}_{j=0}^{\infty}$ becomes a continuous-time signal by adopting a component of zero-order hold. A diagram of the event-based nonlinear control design under networked environment is depicted in Fig. 2.

Next, we revisit the robust control design of the uncertain system (24) but based on the event-triggering mechanism. The cost function is still defined as (3) and the utility is set as

$$\bar{U}^R(x(t), u(t)) = \rho d_M^2(x(t)) + x^{\mathsf{T}}(t)Qx(t) + u^{\mathsf{T}}(t)Ru(t) \tag{58}$$

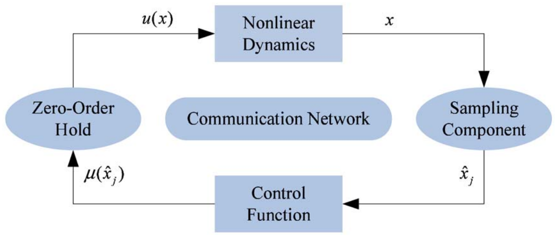

Fig. 2. Structure of the network-based event-triggered control design. The sampling component can be built via the function of a communication channel. It, together with the zero-order hold, forms the main components of time/event transformation.

where $Q = Q^{\mathsf{T}} > 0$. In the time-triggered case, with the new utility (58), the HJB equation can be written as

$$\bar{H}^R\big(x, u^*(x), \nabla J^*(x)\big) = \rho d_M^2(x) + x^{\mathsf{T}} Q x + u^{*\mathsf{T}}(x) R u^*(x) + \big(\nabla J^*(x)\big)^{\mathsf{T}}\big[f(x) + g(x)u^*(x)\big]. \quad (59)$$

Considering the fact that $\hat{x}_j = x(t) + e_j(t)$ and using the control signal $\mu(\hat{x}_j)$, the nominal system (1) becomes a sampled-data version as follows:

$$\dot{x}(t) = f(x(t)) + g(x(t))\mu\big(x(t) + e_j(t)\big), \forall t \in \big[s_j, s_{j+1}\big). \quad (60)$$

With the event-triggering mechanism, the control signal is derived based on the sampled state $\hat{x}_j$ instead of the real state vector $x(t)$. In this circumstance, the time-based optimal feedback control function (8) becomes the event-triggered version given by

$$\mu^*\big(\hat{x}_j\big) = -\frac{1}{2} R^{-1} g^{\mathsf{T}}\big(\hat{x}_j\big) \nabla J^*\big(\hat{x}_j\big) \quad (61)$$

where $\nabla J^*(\hat{x}_j) = (\partial J^*(x)/\partial x)|_{x=\hat{x}_j}$. Then, the HJB equation can be written as

$$\begin{aligned} &H^E\big(x, \mu^*\big(\hat{x}_j\big), \nabla J^*(x)\big) \\ &= \rho d_M^2(x) + x^{\mathsf{T}} Q x + \mu^{*\mathsf{T}}\big(\hat{x}_j\big) R \mu^*\big(\hat{x}_j\big) \\ &\quad + \big(\nabla J^*(x)\big)^{\mathsf{T}}\big[f(x) + g(x)\mu^*\big(\hat{x}_j\big)\big] \\ &= \rho d_M^2(x) + x^{\mathsf{T}} Q x + \big(\nabla J^*(x)\big)^{\mathsf{T}} f(x) \\ &\quad - \frac{1}{2}\big(\nabla J^*(x)\big)^{\mathsf{T}} g(x) R^{-1} g^{\mathsf{T}}\big(\hat{x}_j\big) \nabla J^*\big(\hat{x}_j\big) \\ &\quad + \frac{1}{4}\big(\nabla J^*\big(\hat{x}_j\big)\big)^{\mathsf{T}} g\big(\hat{x}_j\big) R^{-1} g^{\mathsf{T}}\big(\hat{x}_j\big) \nabla J^*\big(\hat{x}_j\big). \end{aligned} \quad (62)$$

It is important to note that (62) is called the event-triggered HJB equation, which in general, is not equal to zero. Instead, the time-triggered HJB equation (59) and the event-triggered HJB equation (62) satisfy the relationship

$$\begin{aligned} &\bar{H}^R(x, u^*(x), \nabla J^*(x)) - H^E\big(x, \mu^*\big(\hat{x}_j\big), \nabla J^*(x)\big) \\ &= -\big[u^*(x) - \mu^*\big(\hat{x}_j\big)\big]^{\mathsf{T}} R \big[u^*(x) - \mu^*\big(\hat{x}_j\big)\big]. \end{aligned} \quad (63)$$

The event-triggered optimal control approach [155] provides the possibility of extending its result to robust control design. Now, we present the main theorem reflecting the transformation of the robust and optimal control problems under event-triggering mechanism. The following assumption is needed.

Assumption 3 [155], [160]: The control law $u(x)$ is Lipschitz continuous with respect to the event-triggering error

$$\begin{aligned} &\|u(x(t)) - u\big(\hat{x}_j\big)\| \\ &= \|u(x(t)) - u(x(t) + e_j(t))\| \le \mathcal{L}_u \|e_j(t)\| \end{aligned} \quad (64)$$

where $\mathcal{L}_u$ a positive real constant.

Theorem 6 [160]: Suppose that Assumption 3 holds. For the uncertain nonlinear system (24), consider its nominal system (1) with cost function (3), utility (58), and the sampled-data system (60). The sampled-data control law is developed by (61) for all $t \in [s_j, s_{j+1})$, $j \in \mathbb{N}$. If the triggering condition is defined as

$$\|e_j(t)\|^2 \le \frac{(1-\eta)\lambda_{\min}(Q)\|x\|^2}{2\|\mathcal{R}\|^2 \mathcal{L}_u^2} - \frac{(2\|\mathcal{R}\|^2 - \rho) d_M^2(x)}{2\|\mathcal{R}\|^2 \mathcal{L}_u^2} \triangleq e_T \quad (65)$$

where the matrix $\mathcal{R}$ satisfies $R = \mathcal{R}^{\mathsf{T}}\mathcal{R}$, e_T is the threshold, and $\eta \in (0, 1)$ is a design parameter of the sample frequency, then, with the event-triggered control law (61), the system (24) achieves robust stabilization.

Note that according to [155], the triggering condition can be given as

$$\|e_j(t)\|^2 \le \frac{(1-\eta)\lambda_{\min}(Q)\|x\|^2}{2\|\mathcal{R}\|^2 \mathcal{L}_u^2} \triangleq \bar{e}_T \quad (66)$$

when studying the event-triggered optimal control problem without considering the uncertain term but using another threshold $\bar{e}_T$. It is also shown that in such event-triggered control design problem, by increasing η to close to 1, one can asymptotically approach the performance of the time-triggered controller (8) [155].

If we perform neural network implementation based on the critic component (15), the event-triggered approximate optimal control law can be formulated as

$$\hat{\mu}^*\big(\hat{x}_j\big) = -\frac{1}{2} R^{-1} g^{\mathsf{T}}\big(\hat{x}_j\big)\big(\nabla \sigma_c\big(\hat{x}_j\big)\big)^{\mathsf{T}} \hat{\omega}_c. \quad (67)$$

Then, a new triggering condition with a different threshold $\hat{e}_T$ can be derived during the adaptive critic control implementation and the UUB stability of the closed-loop system can be analyzed when applying the event-based controller (67) [160].

The ADP-based event-triggered robust control design can be implemented in Algorithm 3. There are two main phases included therein, i.e., the adaptive critic learning and the robust control implementation. Note that the neural learning phase and the robust implementation phase are separated and are performed successively. However, there is a transmission of the weight vector between them. The critic network is first trained to facilitate learning the event-triggered optimal control law of the nominal system. After that, the converged weights are applied to achieve the event-triggered robust stabilization of the original controlled plant with uncertainties.

At last, it is worth mentioning that, using the comparison lemma [154], [163], the minimal intersample time

$$\Delta s_{\min} = \min_{j \in \mathbb{N}} \big\{s_{j+1} - s_j\big\} \quad (68)$$

Algorithm 3 ADP-Based Event-Triggered Robust Control

1: Select an appropriate activation function $\sigma_c(x)$ and initialize the weight vector of the critic neural network.
2: Choose the learning rate α_c and conduct adaptive critic learning by employing the weight updating rule and the triggering condition with threshold $\hat{e}_T$.
3: Keep the converged weight vector unchanged after the online learning process and then go to the robust control implementation.
4: Choose the constant parameter $\mathcal{L}_u$ and perform the robust adaptive critic control design by considering the triggering condition (65) with threshold e_T.
5: Obtain the event-triggered robust control law and then stop the algorithm.

is proven to be lower bounded by a nonzero positive constant [161]. Therefore, the infamous Zeno behavior[7] of the event-triggered robust control design is avoided expectedly.

B. Improving the Critic Learning Rule

The traditional adaptive-critic-based design always depends on the choice of an initial stabilizing controller [58], [66], [120], [131], [132], [157], [160]–[162], which is difficult to obtain in practical control activities and also narrows the application scope of ADP to a certain extent. Generally, we should choose a specified weight vector to create an initial stabilizing control law by the trial-and-error approach and then start the training process. Otherwise, an unstable control may lead to the instability of the closed-loop system. This fact motivates researchers' effort to relax the initial condition [71], [119], [137], [140], [164]–[166], where the interesting idea was from [164]. Therein, a piecewise function is utilized to reduce the proposed initial condition and check the stability, but the theoretical proof is a bit complicated. In this section, we focus on improving the critic learning rule to reduce the initial condition with a simpler manner. To this end, we add a meaningfully reinforced but easily accessible component to the traditional adaptive critic framework, so as to achieve the online optimal regulation and then robust stabilization. An assumption is given here which is the same as [71], [119], [137], [140], and [164]–[166].

Assumption 4: Consider system (1) with cost function (3) and its closed-loop form with the action of the optimal feedback control (17). Let $J_s(x)$ be a continuously differentiable Lyapunov function candidate that satisfies

$$\dot{J}_s(x) = (\nabla J_s(x))^{\mathsf{T}}\big[f(x) + g(x)u^*(x)\big] < 0. \tag{69}$$

There exists a positive definite matrix $\Xi \in \mathbb{R}^{n\times n}$ such that

$$\begin{aligned}&(\nabla J_s(x))^{\mathsf{T}}\big[f(x) + g(x)u^*(x)\big]\\ &\quad= -(\nabla J_s(x))^{\mathsf{T}}\,\Xi\nabla J_s(x) \le -\lambda_{\min}(\Xi)\|\nabla J_s(x)\|^2. \end{aligned}\tag{70}$$

Note that during the implementation process, $J_s(x)$ can be obtained by suitably selecting a polynomial with respect to the state vector, such as the form $J_s(x) = (1/2)x^{\mathsf{T}}x$.

[7]The minimal intersample time might be zero which causes the accumulation of interexecution times [153], [154].

When applying the approximate optimal control (18) to the controlled plant, we should certainly exclude the case that the closed-loop system is unstable, that is

$$(\nabla J_s(x))^{\mathsf{T}}\big[f(x) + g(x)\hat{u}^*(x)\big] > 0. \tag{71}$$

Hence, we utilize an additional term to improve the training process by adjusting $\dot{J}_s(x)$ along the negative gradient direction with respect to $\hat{\omega}_c$, which is

$$\dot{\hat{\omega}}_c^{\mathrm{s}} = -\alpha_s \frac{\partial\big[(\nabla J_s(x))^{\mathsf{T}}\big(f(x) + g(x)\hat{u}^*(x)\big)\big]}{\partial \hat{\omega}_c} \tag{72}$$

where $\alpha_s > 0$ is the adjusting rate of the additional stabilizing term. This parameter affects the extent of the criterion improvement and can be determined by control practitioners according to their design objectives. Therefore, the improved critic learning rule is developed by [123], [167], [168]

$$\dot{\hat{\omega}}_c^{\mathrm{I}} = -\alpha_c \frac{\phi}{\big(1 + \phi^{\mathsf{T}}\phi\big)^2} e_c + \dot{\hat{\omega}}_c^{\mathrm{s}}. \tag{73}$$

The learning rule (73) reflects an efficient improvement to the traditional criteria, such as those used in [66], [71], [119], [120], [137], [140], [157], [160]–[162], and [164]–[166]. It highlights the elimination of the original stabilizing control law. As a result, the weight vector of the critic network can be simply initialized as zero when we implement the adaptive neural control algorithm. Using Assumption 4, the closed-loop stability with the improved learning rule can also be analyzed.

VII. Comparison Remarks Between ADP-Based Robust Control and H_∞ Control Designs

As is shown in previous sections, the wide existence of uncertain parameters or disturbances of the dynamical plant always leads to the necessity of designing robust controllers. There exists a class of H_∞ control methods [169], which focuses on constructing the worst-case control law for specified plants including additive disturbances or dynamical uncertainties. From the point of minimax optimization, the H_∞ control problem can be formulated as a two-player zero-sum differential game. In order to obtain a controller that minimizes the cost function in the worst-case disturbance, it incorporates the requirement of finding the Nash equilibrium solution corresponding to the Hamilton–Jacobi–Isaacs (HJI) equation. However, it is intractable to acquire the analytic solution for general nonlinear systems. This issue is similar to the difficulty of solving the HJB equation in nonlinear optimal regulation design discussed in the previous sections. Hence, using the idea of ADP, iterative methods have been developed to solve the H_∞ control problems. Similar to the adaptive critic optimal regulation, this is known as the adaptive-critic-based H_∞ control design (see [170]–[180] and the related references therein).

Consider a class of continuous-time affine nonlinear systems with external perturbations described by

$$\dot{x}(t) = f(x(t)) + g(x(t))u(t) + h(x(t))v(t) \tag{74a}$$

$$z(t) = \mathcal{Q}(x(t)) \tag{74b}$$

where $v(t) \in \mathbb{R}^q$ is the perturbation vector with $v(t) \in \mathcal{L}_2[0, \infty)$, $z(t) \in \mathbb{R}^p$ is the objective output, and $h(\cdot)$ is differentiable in its argument.

In nonlinear H_∞ control design, we need to find a feedback control law $u(x)$ such that the closed-loop dynamics is asymptotically stable and has $\mathcal{L}_2$-gain no larger than ϱ, that is

$$\int_0^\infty \left[\|\mathcal{Q}(x(\tau))\|^2 + u^\mathsf{T}(\tau)Ru(\tau)\right]\mathrm{d}\tau \le \varrho^2 \int_0^\infty v^\mathsf{T}(\tau)Pv(\tau)\mathrm{d}\tau \tag{75}$$

where $\|\mathcal{Q}(x)\|^2 = x^\mathsf{T}(t)Qx(t)$ and Q, R, and P are symmetric positive definite matrices with appropriate dimensions. If the condition (75) is satisfied, the closed-loop system is said to have $\mathcal{L}_2$-gain no larger than ϱ. Note that the solution of H_∞ control problem is the saddle point of zero-sum game theory and is denoted as a pair of laws (u^*, v^*), where u^* and v^* are called the optimal control and the worst-case disturbance, respectively.

Based on [170]–[180], we generally let the utility be

$$\mathcal{U}(x(t), u(t), v(t)) = x^\mathsf{T}(t)Qx(t) + u^\mathsf{T}(t)Ru(t) - v^\mathsf{T}(t)Pv(t) \tag{76}$$

and define the infinite horizon cost function as

$$\mathcal{J}(x(t), u, v) = \int_t^\infty \mathcal{U}(x(\tau), u(\tau), v(\tau))\mathrm{d}\tau. \tag{77}$$

The design goal is to find the feedback saddle point solution (u^*, v^*), such that the Nash condition

$$\mathcal{J}^*(x_0) = \min_u \max_v \mathcal{J}(x_0, u, v) = \max_v \min_u \mathcal{J}(x_0, u, v) \tag{78}$$

holds, where the asterisked symbol $\mathcal{J}^*(x_0)$ represents the optimal cost. For an admissible control $u \in \mathscr{A}(\Omega)$, if the related cost function (77) is continuously differentiable, then its infinitesimal version is the nonlinear Lyapunov equation

$$0 = \mathcal{U}(x, u, v) + (\nabla\mathcal{J}(x))^\mathsf{T}[f(x) + g(x)u + h(x)v] \tag{79}$$

with $\mathcal{J}(0) = 0$. Define the Hamiltonian of system (74a) as

$$\mathcal{H}(x, u, v, \nabla\mathcal{J}(x)) = \mathcal{U}(x, u, v) + (\nabla\mathcal{J}(x))^\mathsf{T}[f + gu + hv]. \tag{80}$$

According to Bellman's optimality principle, the optimal cost function $\mathcal{J}^*(x)$ guarantees the so-called HJI equation

$$\min_u \max_v \mathcal{H}(x, u, v, \nabla\mathcal{J}^*(x)) = 0. \tag{81}$$

The saddle point solution (u^*, v^*) satisfies the stationary condition [177], which can be used to obtain the optimal control law and the worst-case disturbance law as follows:

$$u^*(x) = -\frac{1}{2}R^{-1}g^\mathsf{T}(x)\nabla\mathcal{J}^*(x) \tag{82a}$$

$$v^*(x) = \frac{1}{2\varrho^2}P^{-1}h^\mathsf{T}(x)\nabla\mathcal{J}^*(x). \tag{82b}$$

Considering the two formulas in (82), the HJI equation becomes the form

$$\begin{aligned} 0 &= \mathcal{H}\big(x, u^*, v^*, \nabla\mathcal{J}^*(x)\big) \\ &= x^\mathsf{T}Qx + \big(\nabla\mathcal{J}^*(x)\big)^\mathsf{T}f(x) \\ &\quad - \frac{1}{4}\big(\nabla\mathcal{J}^*(x)\big)^\mathsf{T}g(x)R^{-1}g^\mathsf{T}(x)\nabla\mathcal{J}^*(x) \\ &\quad + \frac{1}{4\varrho^2}\big(\nabla\mathcal{J}^*(x)\big)^\mathsf{T}h(x)P^{-1}h^\mathsf{T}(x)\nabla\mathcal{J}^*(x) \end{aligned} \tag{83}$$

with $\mathcal{J}^*(0) = 0$. Note that the HJI equation (83) is difficult to solve in theory. This inspires us to devise an approximate control strategy to overcome the difficulty by virtue of ADP.

Incorporating the critic neural network, the approximate values of the control and disturbance laws are

$$\hat{u}(x) = -\frac{1}{2}R^{-1}g^\mathsf{T}(x)(\nabla\sigma_c(x))^\mathsf{T}\hat{\omega}_c \tag{84a}$$

$$\hat{v}(x) = \frac{1}{2\varrho^2}P^{-1}h^\mathsf{T}(x)(\nabla\sigma_c(x))^\mathsf{T}\hat{\omega}_c. \tag{84b}$$

The closed-loop system is also proven to be UUB with the approximate control (84a) and disturbance law (84b). Recently, there are also some results of event-triggered H_∞ control based on ADP [179], [180]. Therein, the H_∞ control problem for continuous-time affine nonlinear systems was investigated with network-based event-triggering formulation.

The ADP-based robust control and ADP-based H_∞ control methods are both developed to cope with the external perturbations. Basically, both of them concern with the uncertainties or disturbances and guarantee the robustness of the controlled plants. However, there are also some apparent differences between them, listed as follows.

1) The design objective of ADP-based robust control and ADP-based H_∞ control is not totally the same. Achieving robust stability is the single task of the robust control design while attaining certain $\mathcal{L}_2$-gain performance level is the additional objective of the H_∞ control design. It means that, the H_∞ control scheme is established with the purpose of disturbance attenuation.
2) The cost functions, or specifically the utilities are defined differently. In ADP-based robust control design, we define a modified utility in terms of the state variable reflecting the bound of the uncertainty and the control variable. However, in ADP-based H_∞ control design, what we give is an utility composed of the state, control, and disturbance variables. As a result, the ADP method is employed to solve the modified HJB equation in the former while it is adopted to cope with the HJI equation in the latter.
3) The feedback controller in the ADP-based robust control design is not the same as the H_∞ control design. In the robust control framework, the uncertain term is not incorporated to the expression of the feedback controller. However, the H_∞ method contains another law called disturbance except the control law, which also should be formulated during the design process.

Though there exist great distinctions, it is certainly convinced that the ADP method is applicable to both robust control and H_∞ control problems. The involvement of ADP

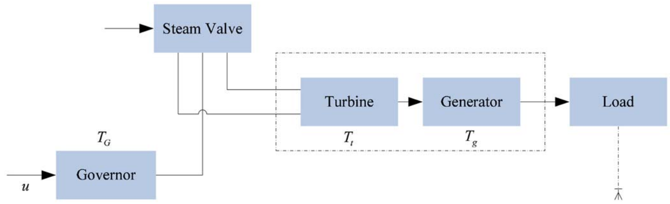

Fig. 3. Block diagram of the proposed power system.

to robust and H_∞ designs bring the adaptive and self-learning properties into the traditional control methods. Hence, with ADP formulation, the adaptive, learning, and intelligent systems are widely constructed under uncertain environment. All of these research demonstrates the necessity and significance of adaptive-critic-based nonlinear robust control designs.

VIII. Applications

There are many successful applications with ADP-based control design. Among them, complex industrial systems, such as power systems [30], [44], [67], [84], [129]–[133], [158], [181]–[186], mechanical systems [32], [36], [41], [44], [71], [123], [125], [131], [187], [188], and intelligent transportation systems [189], [190] are the most common application areas. In particular, with adaptive-critic-based robust control methods, there are some direct applications in fields such as interceptor-target engagement [125], jet engine [131], power systems [130]–[133], and so on. In this section, we first take a practical power system to perform event-triggered optimal regulation and then apply the ADP-based optimal control scheme to achieve robust stabilization of an overhead crane, thereby demonstrating the applicability of theoretical results.

A. Power System Application

Smart grids including various load changes and multiple renewable generations have been acquiring intensive attention in recent years [191]–[194]. In modern power systems, many kinds of distributed and renewable energies have been considered to integrate into micro-grids. However, the imbalance between load consumptions and power generations may result in the frequency deviation, especially for micro-grids. Hence, the frequency stability of micro-grids has been a significant topic to the development of modern power systems. In this example, we consider a power system described in Fig. 3, which is composed of a turbine generator, a system load, and an automatic generation control [67]. Let ξ_f, ξ_g, and ξ_G be the incremental change of the frequency deviation, the generator output, and the governor value position, respectively, while let the control input u represent the incremental speed change of positive deviation. If we define $x = [\xi_f, \xi_g, \xi_G]^{\mathsf{T}} \in \mathbb{R}^3$ as the state vector, where $x_1 = \xi_f$, $x_2 = \xi_g$, and $x_3 = \xi_G$, then the state-space description of the power system can be written as

$$\dot{x} = \begin{bmatrix} -\dfrac{1}{T_G} & 0 & -\dfrac{1}{F_r T_G} \\ \dfrac{K_t}{T_t} & -\dfrac{1}{T_t} & 0 \\ 0 & \dfrac{K_g}{T_g} & -\dfrac{1}{T_g} \end{bmatrix} x + \begin{bmatrix} \dfrac{1}{T_G} \\ 0 \\ 0 \end{bmatrix} u \tag{85}$$

where the related parameters are described in Table V.

TABLE V
Parameters of the Proposed Power System

Symbol	Meaning
T_G	Time constant of the governor
T_t	Time constant of the turbine model
T_g	Time constant of the generator model
F_r	Feedback regulation constant
K_t	Gain constant of the turbine model
K_g	Gain constant of the generator model

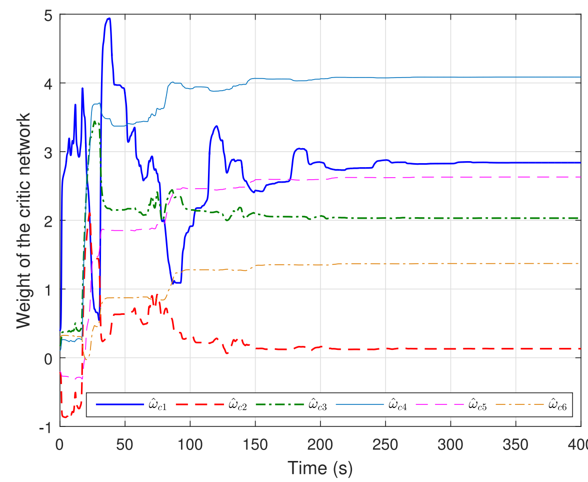

Fig. 4. Convergence of the weight vector of the neural network.

In this simulation, we select the parameters as $T_G = 5$, $T_t = 10$, $T_g = 10$, $F_r = 0.5$, $K_t = 1$, and $K_g = 1$. The cost function is defined as (3) with utility being chosen as $U(x, u) = 1.5x^{\mathsf{T}}x + u^{\mathsf{T}}u$. The initial state vector is set as $x_0 = [0.2, -0.2, 0.1]^{\mathsf{T}}$. The critic network is constructed to approximate the optimal cost function as

$$\hat{J}^*(x) = \hat{\omega}_{c1}x_1^2 + \hat{\omega}_{c2}x_1x_2 + \hat{\omega}_{c3}x_1x_3 + \hat{\omega}_{c4}x_2^2 + \hat{\omega}_{c5}x_2x_3 + \hat{\omega}_{c6}x_3^2. \tag{86}$$

Note that $\sigma_c(x) = [x_1^2, x_1x_2, x_1x_3, x_2^2, x_2x_3, x_3^2]^{\mathsf{T}}$ and $\hat{\omega}_c = [\hat{\omega}_{c1}, \hat{\omega}_{c2}, \hat{\omega}_{c3}, \hat{\omega}_{c4}, \hat{\omega}_{c5}, \hat{\omega}_{c6}]^{\mathsf{T}}$ are the activation function and the estimated weight vector of the neural network, respectively. Choosing $\eta = 0.5$ and adding a probing noise for persistence of excitation, we perform the simulation with the sampling time being set as 0.1 s. After the critic learning stage with $\alpha_c = 1.8$, the weight vector converges to $[2.8392, 0.1313, 2.0318, 4.0852, 2.6303, 1.3715]^{\mathsf{T}}$, showing in Fig. 4. It is found that the convergence has occurred at $t = 350$ s and after that we remove the probing signal. The time-based controller uses 4000 samples of state while the event-based control law only requires 1571 samples, showing in Fig. 5, which reduces the controller updates by 60.73% during the learning session.

At last, by applying the approximate optimal controller to system (85) for $t = 100$s, we obtain the 3-D view of the state trajectory shown in Fig. 6. This substantiates the effectiveness of the ADP-based event-triggered optimal state feedback control strategy.

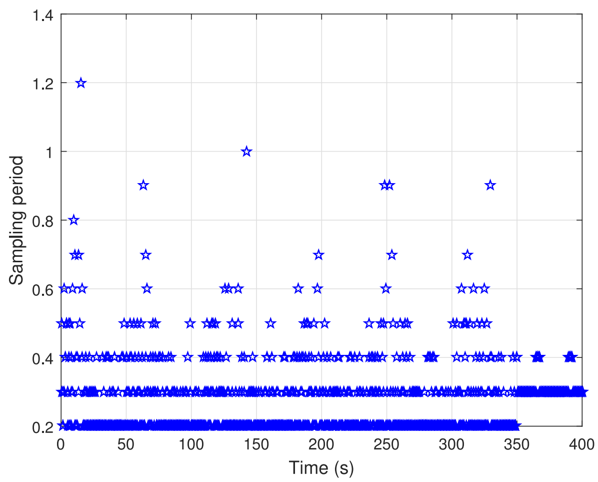

Fig. 5. Sampling period in the learning process.

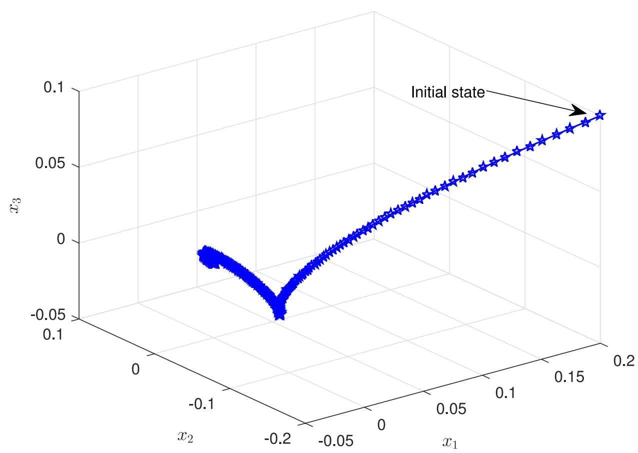

Fig. 6. 3-D view of the state trajectory.

TABLE VI
PARAMETERS OF THE PROPOSED OVERHEAD CRANE PLANT

Symbol	Meaning
M_t	Total mass of the trolley
M_l	Mass of the load
L_r	Length of the rope
g_a	Gravitational acceleration
θ	Swing angle of the load with the vertical line
χ	Trolley position with respect to the origin
f_t	Force applied to the trolley

B. Overhead Crane System Application

The overhead traveling cranes, which transport loads from one place to another and play an important role in industry, incorporate complex nonlinearities and difficult control design tasks [195], [196]. In general, an overhead crane plant consists of a trolley, a load, and a rope. The simple structure of a typical overhead crane is shown in Fig. 7. The parameters used in the whole plant are given in Table VI. Note that here, the stiffness and mass of the rope are not considered and the load is seen as a point mass, which are reasonable if a multiwire rope is adopted in practice [195]. Based on [195] and [196], the dynamical model of the overhead crane can be formulated as

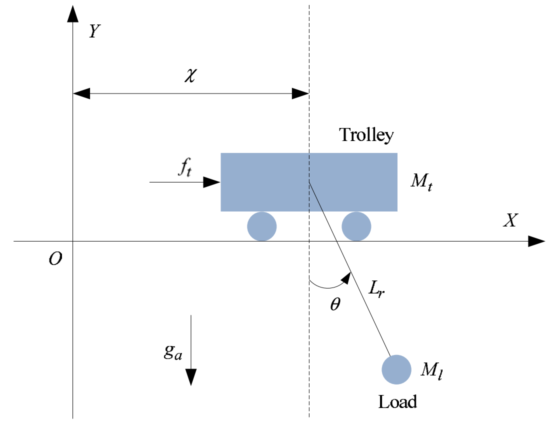

Fig. 7. Diagram of the proposed overhand crane.

$$(M_t + M_l)\ddot{\chi} + M_l L_r\big[\ddot{\theta}\cos(\theta) - \dot{\theta}^2\sin(\theta)\big] = f_t \tag{87a}$$

$$\ddot{\chi}\cos(\theta) + L_r\ddot{\theta} + g_a\sin(\theta) = 0. \tag{87b}$$

For obtaining the state space description of plant (87), we define $u = f_t$ as the control input and $x = [x_1, x_2, x_3, x_4]^{\mathsf{T}}$ as the system state, where $x_1 = \chi$, x_2 is the trolley velocity, $x_3 = \theta$, and x_4 is the angular velocity of the load. Then, the dynamics (87) can be rewritten as (88), shown at the bottom of this page. Clearly, it is a nonlinear system with 4-D state variable and 1-D control variable.

For the simulation purpose, we set $M_t = 1.2$ kg, $M_l = 0.8$ kg, and $L_r = 0.5$ m and choose $g_a = 9.8$ m/s^2 and then make a modification to the plant (88) by introducing an uncertain term $d(x) = 2\varpi x_1\sin(x_2^2 x_3)\cos(x_3 x_4^2)$ with $\varpi \in [-0.5, 0.5]$, so as to help to evaluate the robustness of the controlled plant. Then, we find that the bounded function can be selected as $d_M(x) = \|x\|$ and the modified utility function can be written as $U^R(x,u) = 2\|x\|^2 + u^{\mathsf{T}}u$. In this example, for using the ADP method, the optimal cost function is approximated by

$$\begin{aligned}\hat{J}^*(x) = {} & \hat{\omega}_{c1}x_1^2 + \hat{\omega}_{c2}x_1x_2 + \hat{\omega}_{c3}x_1x_3 \\ & + \hat{\omega}_{c4}x_1x_4 + \hat{\omega}_{c5}x_2^2 + \hat{\omega}_{c6}x_2x_3 \\ & + \hat{\omega}_{c7}x_2x_4 + \hat{\omega}_{c8}x_3^2 + \hat{\omega}_{c9}x_3x_4 + \hat{\omega}_{c10}x_4^2.\end{aligned} \tag{89}$$

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\\ \dot{x}_4\end{bmatrix} = \begin{bmatrix} x_2 \\ \dfrac{M_l L_r x_4^2\sin(x_3) + M_l g_a\sin(x_3)\cos(x_3)}{M_t + M_l\sin^2(x_3)} \\ x_4 \\ -\dfrac{(M_t+M_l)g_a\sin(x_3) + M_l L_r x_4^2\sin(x_3)\cos(x_3)}{\big(M_t + M_l\sin^2(x_3)\big)L_r}\end{bmatrix} + \begin{bmatrix} 0 \\ \dfrac{1}{M_t + M_l\sin^2(x_3)} \\ 0 \\ -\dfrac{\cos(x_3)}{\big(M_t + M_l\sin^2(x_3)\big)L_r}\end{bmatrix}u \tag{88}$$

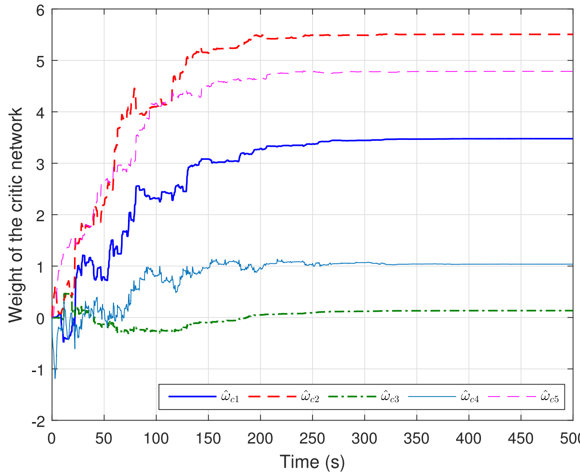

Fig. 8. Convergence of the weight vector (part I).

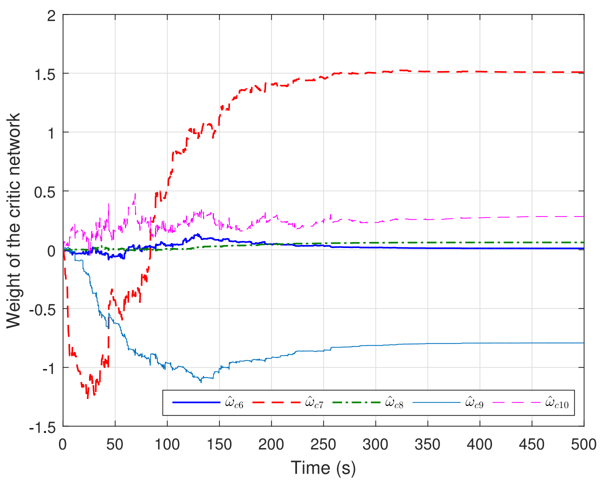

Fig. 9. Convergence of the weight vector (part II).

We choose the initial state vector as $x_0 = [2.5, 0, 0, 0]^{\mathsf{T}}$, select the learning rate parameters as $\alpha_c = 2.9$ and $\alpha_s = 0.1$, and then employ a probing noise for guaranteeing the persistence of excitation condition. By performing a sufficient learning stage, the weight vector of the critic network converges to [3.4771, 5.5064, 0.1333, 1.0377, 4.7870, 0.0116, 1.5098, 0.0620, −0.7923, 0.2821]$^{\mathsf{T}}$, as illustrated in Figs. 8 and 9. In this simulation, we find that the convergence has occurred at $t = 450$s and then we remove the probing signal.

Finally, the performance of robust stabilization is checked by selecting $\varpi = 0.5$ and applying the derived control law to the uncertain system for $t = 25$ s. The system and control trajectories are depicted in Fig. 10. Clearly, under the action of the developed controller, the state vector is driven to zero as time goes on, which validates the good robustness property with respect to the dynamical uncertainty.

IX. Several Notes on Future Perspectives

Although there are many excellent results in terms of ADP-based intelligent control design at present, further studies on various fundamental issues are still needed, such as convergence of the iterative algorithm, stability of the controlled system, as well as optimality and robustness of the feedback controller. For instance, the stability and robustness of finite horizon optimal control [59], [62] and discounted optimal control [82], [197], [198] are important to improve the existing ADP-related control design when combining with advanced learning mechanisms and iterative algorithms. From the view of convergence and optimality, the generalized value iteration and policy iteration for discrete-time systems [60], [63], [96] as well as the generalized policy iteration and value iteration for continuous-time systems [84], [88], [96] should be given special attention. They are all advanced iteration algorithms compared with traditional opinions that value iteration is for discrete-time systems [27], [47], [52], [55], [59] while policy iteration is for continuous-time systems [28], [66], [69], [77], [82]. Besides, avoiding the weaknesses of neural network approximation and achieving global optimal stabilization [83] are worth further study as well. Establishing the uniqueness of HJB solution and studying the convergence of value and policy iterations with abstract dynamic programming [199], [200] are also interesting and important. Actually, there are many future study topics indicated in the previous surveys [86], [87], [92], [93]. Greater efforts should be put to establish perfect methodology for ADP-related research in theory. Meanwhile, more and more practical applications of ADP and reinforcement learning with significant economic impact are of great demand.

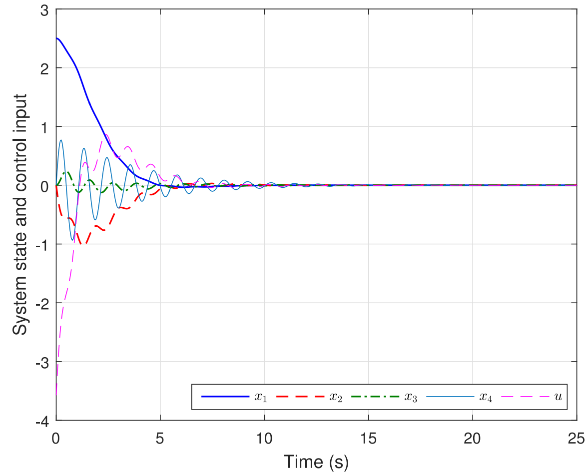

Fig. 10. State and control trajectories.

As is known, most of the techniques in reinforcement learning can be viewed as attempts to achieve much the same effect as dynamic programming, with less computation and without assuming a perfect model of the environment. The ADP method is also developed for performing optimization of complex systems with unknown and uncertain dynamics. A common and significant aspect of ADP and reinforcement learning is the model-free design property. Hence, it is extremely necessary to use effectively the data information to establish more advanced data-driven control approaches. The parallel/computational control method [12], [100], iterative neural dynamic programming algorithm [105], [106], the integral reinforcement learning technique [107]–[109], and the concurrent learning algorithm [180] are all of meaningful attempts. When considering the uncertainty and robustness, the robust optimal control strategy with efficient data-driven component is indeed called for further study.

How to combine data-based approach [46], [54], [55], [76], [78], [80], [81], [93], [105]–[109] with event-triggered mechanism [155]–[162] to conduct the mixed data/event driven control [201], [202] also should be considered. With this new formulation and by virtue of the discussion on robustness [116]–[133], an effective robust optimal control methodology of complex nonlinear systems with dynamical uncertainties can be developed, which reduces the requirement of the dynamical model and saves the communication resource simultaneously. Thus, it is beneficial to study the mixed data/event driven control design for complex nonlinear systems. In addition, when extending the existing results to multiagent systems, distributed cooperative optimization [203], [204] can be attained. The communication factor is always considered in distributed control design, which may be quite useful to network-based systems. Consequently, the distributed design together with the previously discussed decentralized control design involving the idea of ADP may be another promising direction for dealing with intelligent control of complex systems, especially under uncertain environment.

However, it is far from enough, since practical processes often contain big data resources and complicated situations. This is becoming more and more apparent along with the trend of emerging high technologies, such as artificial intelligence, big data, cloud computing, cyber-physical systems, deep learning, and knowledge automation [12], [100]. Particularly, deep reinforcement learning is able to output control signal directly based on input images, which incorporates both advantages of the perception of deep learning and the decision making of reinforcement learning [1], [3], [4], [6], [12]. This mechanism makes the artificial intelligence much close to human thinking modes. Combining deep learning with ADP and reinforcement learning will benefit us to construct more intelligent systems and accomplish higher level brain-intelligence.

X. Conclusion

This survey reviews the main results of adaptive-critic-based (or ADP-based) robust control of nonlinear continuous-time systems. In summary, the ADP-based robust stabilization of nonlinear systems with matched uncertainties, nonlinear guaranteed cost control design of unmatched case, nonlinear decentralized control design of interconnected case, and further discussions on event-based robust control design, improvement of the critic learning rule, nonlinear H_∞ control design, as well as several future perspectives are included. It is a comprehensive survey of ADP-based robust control in terms of motivation, method, analysis, design, and application.

Repeatedly, the idea of ADP is proposed to achieve optimal decision and control of complex systems with uncertain and unknown dynamics in an online manner. As Werbos [205]–[208] pointed out, ADP may be the only approach that can achieve truly brain-like intelligence. More and more evidence has accumulated, suggesting that optimality is an organizing principle for understanding brain intelligence [206]–[208]. There has been a hot interest in brain research around the world in recent years. We certainly hope ADP can make considerable contributions to brain research in general and to brain-like intelligence in particular. Continuing efforts are still being made in the quest for finding solutions to dynamic programming problems with manageable amount of computation and communication as well as inclusive guarantee of stability, convergence, optimality, and robustness. Consequently, the research on robust adaptive critic control design will certainly attain greater progress in the future.

References

[1] I. Arel, D. C. Rose, and T. P. Karnowski, "Deep machine learning—A new frontier in artificial intelligence research," *IEEE Comput. Intell. Mag.*, vol. 5, no. 4, pp. 13–18, Nov. 2010.

[2] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[3] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[4] V. Mnih *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, pp. 529–533, Feb. 2015.

[5] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, Jan. 2015.

[6] J. Qiu, Q. Wu, G. Ding, Y. Xu, and S. Feng, "A survey of machine learning for big data processing," *EURASIP J. Adv. Signal Process.*, vol. 2016, doi: 10.1186/s13634-016-0355-x.

[7] X.-W. Chen and X. Lin, "Big data deep learning: Challenges and perspectives," *IEEE Access*, vol. 2, pp. 514–525, 2014.

[8] D. Silver *et al.*, "Mastering the game of go with deep neural networks and tree search," *Nature*, vol. 529, pp. 484–489, Jan. 2016.

[9] S. Haykin, *Neural Networks: A Comprehensive Foundation*. Upper Saddle River, NJ, USA: Prentice-Hall, 1999.

[10] S. Jagannathan, *Neural Network Control of Nonlinear Discrete-Time Systems*. Boca Raton, FL, USA: CRC Press, 2006.

[11] W. Yu, *Recent Advances in Intelligent Control Systems*. London, U.K.: Springer-Verlag, 2009.

[12] F.-Y. Wang *et al.*, "Where does AlphaGo go: From church-turing thesis to AlphaGo thesis and beyond," *IEEE/CAA J. Autom. Sinica*, vol. 3, no. 2, pp. 113–120, Apr. 2016.

[13] R. S. Sutton and A. G. Barto, *Reinforcement Learning—An Introduction*. Cambridge, MA, USA: MIT Press, 1998.

[14] R. E. Bellman, *Dynamic Programming*. Princeton, NJ, USA: Princeton Univ. Press, 1957.

[15] F. L. Lewis, D. Vrabie, and V. L. Syrmos, *Optimal Control*, 3rd ed. New York, NY, USA: Wiley, 2012.

[16] P. J. Werbos, "Beyond regression: New tools for prediction and analysis in the behavioural sciences," Ph.D. dissertation, Harvard Univ., Cambridge, MA, USA, 1974.

[17] P. J. Werbos, "Advanced forecasting methods for global crisis warning and models of intelligence," *Gener. Syst. Yearbook*, vol. 22, no. 12, pp. 25–38, 1977.

[18] P. J. Werbos, "Approximate dynamic programming for real-time control and neural modeling," in *Handbook of Intelligent Control: Neural, Fuzzy, and Adaptive Approaches*, D. A. White and D. A. Sofge, Eds. New York, NY, USA: Van Nostrand Reinhold, 1992, ch. 13.

[19] R. A. Santiago and P. J. Werbos, "New progress towards truly brain-like intelligent control," in *Proc. World Congr. Neural Netw.*, San Diego, CA, USA, Jun. 1994, pp. 27–33.

[20] J. Wang, X. Xu, D. Liu, Z. Sun, and Q. Chen, "Self-learning cruise control using kernel-based least squares policy iteration," *IEEE Trans. Control Syst. Technol.*, vol. 22, no. 3, pp. 1078–1087, May 2014.

[21] 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.

[22] 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.

[23] B. Xu, "Robust adaptive neural control of flexible hypersonic flight vehicle with dead-zone input nonlinearity," *Nonlin. Dyn.*, vol. 80, no. 3, pp. 1509–1520, 2015.
[24] 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.
[25] G. N. Saridis and F. Y. Wang, "Suboptimal control for nonlinear stochastic systems," *Control Theory Adv. Technol.*, vol. 10, no. 4, pp. 847–871, 1994.
[26] R. W. Beard, G. N. Saridis, and J. T. Wen, "Galerkin approximations of the generalized Hamilton–Jacobi–Bellman equation," *Automatica*, vol. 33, no. 12, pp. 2159–2177, Dec. 1997.
[27] A. Al-Tamimi, F. L. Lewis, and M. Abu-Khalaf, "Discrete-time nonlinear HJB solution using approximate dynamic programming: Convergence proof," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 38, no. 4, pp. 943–949, Aug. 2008.
[28] 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.
[29] D. V. Prokhorov and D. C. Wunsch, "Adaptive critic designs," *IEEE Trans. Neural Netw.*, vol. 8, no. 5, pp. 997–1007, Sep. 1997.
[30] R. Padhi, N. Unnikrishnan, X. Wang, and S. N. Balakrishnan, "A single network adaptive critic (SNAC) architecture for optimal control synthesis for a class of nonlinear systems," *Neural Netw.*, vol. 19, no. 10, pp. 1648–1660, Dec. 2006.
[31] T. Hanselmann, L. Noakes, and A. Zaknich, "Continuous-time adaptive critics," *IEEE Trans. Neural Netw.*, vol. 18, no. 3, pp. 631–647, May 2007.
[32] J. J. Murray, C. J. Cox, G. G. Lendaris, and R. Saeks, "Adaptive dynamic programming," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 32, no. 2, pp. 140–153, May 2002.
[33] G. G. Lendaris, "A retrospective on adaptive dynamic programming for control," in *Proc. Int. Joint Conf. Neural Netw.*, Atlanta, GA, USA, Jun. 2009, pp. 1750–1757.
[34] J. Si, A. G. Barto, W. B. Powell, and D. C. Wunsch, Eds., *Handbook of Learning and Approximate Dynamic Programming*. New York, NY, USA: IEEE Press, 2004.
[35] D. Liu and H. Zhang, "A neural dynamic programming approach for learning control of failure avoidance problems," *Int. J. Intell. Control Syst.*, vol. 10, no. 1, pp. 21–32, 2005.
[36] J. Si and Y.-T. Wang, "Online learning control by association and reinforcement," *IEEE Trans. Neural Netw.*, vol. 12, no. 2, pp. 264–276, Mar. 2001.
[37] D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-Dynamic Programming*. Belmont, MA, USA: Athena Sci., 1996.
[38] C. Watkins and P. Dayan, "Q-learning," *Mach. Learn.*, vol. 8, nos. 3–4, pp. 279–292, 1992.
[39] B. Lincoln and A. Rantzer, "Relaxing dynamic programming," *IEEE Trans. Autom. Control*, vol. 51, no. 8, pp. 1249–1260, Aug. 2006.
[40] A. Rantzer, "Relaxed dynamic programming in switching systems," *IEE Proc. Control Theory Appl.*, vol. 153, no. 5, pp. 567–574, Sep. 2006.
[41] H. He, Z. Ni, and J. Fu, "A three-network architecture for on-line learning and optimization based on adaptive dynamic programming," *Neurocomputing*, vol. 78, no. 1, pp. 3–13, 2012.
[42] Z. Ni, H. He, D. Zhao, X. Xu, and D. V. Prokhorov, "GrDHP: A general utility function representation for dual heuristic dynamic programming," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 3, pp. 614–627, Mar. 2015.
[43] X. Zhong, Z. Ni, and H. He, "A theoretical foundation of goal representation heuristic dynamic programming," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 12, pp. 2513–2525, Dec. 2016.
[44] Y. Tang *et al.*, "Fuzzy-based goal representation adaptive dynamic programming," *IEEE Trans. Fuzzy Syst.*, vol. 24, no. 5, pp. 1159–1175, Oct. 2016.
[45] 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.
[46] 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, 2009.
[47] H. Zhang, Y. Luo, and D. Liu, "Neural-network-based near-optimal control for a class of discrete-time affine nonlinear systems with control constraints," *IEEE Trans. Neural Netw.*, vol. 20, no. 9, pp. 1490–1503, Sep. 2009.
[48] J. Fu, H. He, and X. Zhou, "Adaptive learning and control for MIMO system based on adaptive dynamic programming," *IEEE Trans. Neural Netw.*, vol. 22, no. 7, pp. 1133–1148, Jul. 2011.
[49] X. Xu, Z. Hou, C. Lian, and H. He, "Online learning control using adaptive critic designs with sparse kernel machines," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 5, pp. 762–775, May 2013.
[50] A. Heydari and S. N. Balakrishnan, "Finite-horizon control-constrained nonlinear optimal control using single network adaptive critics," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 1, pp. 145–157, Jan. 2013.
[51] A. Heydari, "Revisiting approximate dynamic programming and its convergence," *IEEE Trans. Cybern.*, vol. 44, no. 12, pp. 2733–2743, Dec. 2014.
[52] C. Mu, C. Sun, A. Song, and H. Yu, "Iterative GDHP-based approximate optimal tracking control for a class of discrete-time nonlinear systems," *Neurocomputing*, vol. 214, pp. 775–784, Nov. 2016.
[53] C. Mu, Z. Ni, C. Sun, and H. He, "Air-breathing hypersonic vehicle tracking control based on adaptive dynamic programming," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 3, pp. 584–598, Mar. 2017.
[54] C. Mu, D. Wang, and H. He, "Data-driven finite-horizon approximate optimal control for discrete-time nonlinear systems using iterative HDP approach," *IEEE Trans. Cybern.*, to be published.
[55] D. Wang, D. Liu, Q. Wei, D. Zhao, and N. Jin, "Optimal control of unknown nonaffine nonlinear discrete-time systems based on adaptive dynamic programming," *Automatica*, vol. 48, no. 8, pp. 1825–1832, Aug. 2012.
[56] D. Liu, D. Wang, D. Zhao, Q. Wei, and N. Jin, "Neural-network-based optimal control for a class of unknown discrete-time nonlinear systems using globalized dual heuristic programming," *IEEE Trans. Autom. Sci. Eng.*, vol. 9, no. 3, pp. 628–634, Jul. 2012.
[57] D. Zhao, Z. Xia, and D. Wang, "Model-free optimal control for affine nonlinear systems with convergence analysis," *IEEE Trans. Autom. Sci. Eng.*, vol. 12, no. 4, pp. 1461–1468, Oct. 2015.
[58] Q. Zhao, H. Xu, and S. Jagannathan, "Neural network-based finite-horizon optimal control of uncertain affine nonlinear discrete-time systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 3, pp. 486–499, Mar. 2015.
[59] F.-Y. Wang, N. Jin, D. Liu, and Q. Wei, "Adaptive dynamic programming for finite-horizon optimal control of discrete-time nonlinear systems with ε-error bound," *IEEE Trans. Neural Netw.*, vol. 22, no. 1, pp. 24–36, Jan. 2011.
[60] Q. Wei, D. Liu, and X. Yang, "Infinite horizon self-learning optimal control of nonaffine discrete-time nonlinear systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 4, pp. 866–879, Apr. 2015.
[61] Q. Wei and D. Liu, "A novel policy iteration based deterministic Q-learning for discrete-time nonlinear systems," *Sci. China Inf. Sci.*, vol. 58, no. 12, pp. 1–15, 2015.
[62] R. Song, W. Xiao, Q. Wei, and C. Sun, "Neural-network-based approach to finite-time optimal control for a class of unknown nonlinear systems," *Soft Comput.*, vol. 18, no. 8, pp. 1645–1653, 2014.
[63] D. Liu, H. Li, and D. Wang, "Error bounds of adaptive dynamic programming algorithms for solving undiscounted optimal control problems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 6, pp. 1323–1334, Jun. 2015.
[64] L. Liu, Z. Wang, and H. Zhang, "Adaptive fault-tolerant tracking control for MIMO discrete-time systems via reinforcement learning algorithm with less learning parameters," *IEEE Trans. Autom. Sci. Eng.*, vol. 14, no. 1, pp. 299–313, Jan. 2017.
[65] B. Xu, C. Yang, and Z. Shi, "Reinforcement learning output feedback NN control using deterministic learning technique," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 3, pp. 635–641, Mar. 2014.
[66] K. G. Vamvoudakis and F. L. Lewis, "Online actor–critic algorithm to solve the continuous-time infinite horizon optimal control problem," *Automatica*, vol. 46, no. 5, pp. 878–888, May 2010.
[67] K. G. Vamvoudakis, M. F. Miranda, and J. P. Hespanha, "Asymptotically stable adaptive–optimal control algorithm with saturating actuators and relaxed persistence of excitation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 11, pp. 2386–2398, Nov. 2016.
[68] S. Bhasin *et al.*, "A novel actor–critic–identifier architecture for approximate optimal control of uncertain nonlinear systems," *Automatica*, vol. 49, no. 1, pp. 82–92, 2013.

[69] H. Modares, F. L. Lewis, and M. B. Naghibi-Sistani, "Adaptive optimal control of unknown constrained-input systems using policy iteration and neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 10, pp. 1513–1525, Oct. 2013.

[70] 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.

[71] D. Nodland, H. Zargarzadeh, and S. Jagannathan, "Neural network-based optimal adaptive output feedback control of a helicopter UAV," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 7, pp. 1061–1073, Jul. 2013.

[72] H. Zargarzadeh, T. Dierks, and S. Jagannathan, "Optimal control of nonlinear continuous-time systems in strict-feedback form," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 10, pp. 2535–2549, Oct. 2015.

[73] R. Kamalapurkar, P. Walters, and W. E. Dixon, "Model-based reinforcement learning for approximate optimal regulation," *Automatica*, vol. 64, pp. 94–104, Feb. 2016.

[74] J. Na and G. Herrmann, "Online adaptive approximate optimal tracking control with simplified dual approximation structure for continuous-time unknown nonlinear systems," *IEEE/CAA J. Autom. Sinica*, vol. 1, no. 4, pp. 412–422, Oct. 2014.

[75] Y. Lv, J. Na, Q. Yang, X. Wu, and Y. Guo, "Online adaptive optimal control for continuous-time nonlinear systems with completely unknown dynamics," *Int. J. Control*, vol. 89, no. 1, pp. 99–112, 2016.

[76] B. Luo, H.-N. Wu, T. Huang, and D. Liu, "Data-based approximate policy iteration for affine nonlinear continuous-time optimal control design," *Automatica*, vol. 50, no. 12, pp. 3281–3290, Dec. 2014.

[77] D. Vrabie and F. L. 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, 2009.

[78] H. Zhang, L. Cui, X. Zhang, and Y. Luo, "Data-driven robust approximate optimal tracking control for unknown general nonlinear systems using adaptive dynamic programming method," *IEEE Trans. Neural Netw.*, vol. 22, no. 12, pp. 2226–2236, Dec. 2011.

[79] R. Song, F. L. Lewis, Q. Wei, and H. Zhang, "Off-policy actor-critic structure for optimal control of unknown systems with disturbances," *IEEE Trans. Cybern.*, vol. 46, no. 5, pp. 1041–1050, May 2016.

[80] Y. Zhu, D. Zhao, and X. Li, "Iterative adaptive dynamic programming for solving unknown nonlinear zero-sum game based on online data," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 3, pp. 714–725, Mar. 2017.

[81] G. Xiao, H. Zhang, Y. Luo, and H. Jiang, "Data-driven optimal tracking control for a class of affine non-linear continuous-time systems with completely unknown dynamics," *IET Control Theory Appl.*, vol. 10, no. 6, pp. 700–710, Apr. 2016.

[82] D. Liu, X. Yang, and H. Li, "Adaptive optimal control for a class of continuous-time affine nonlinear systems with unknown internal dynamics," *Neural Comput. Appl.*, vol. 23, no. 7, pp. 1843–1850, 2013.

[83] Y. Jiang and Z.-P. Jiang, "Global adaptive dynamic programming for continuous-time nonlinear systems," *IEEE Trans. Autom. Control*, vol. 60, no. 11, pp. 2917–2929, Nov. 2015.

[84] T. Bian and Z.-P. Jiang, "Value iteration and adaptive dynamic programming for data-driven adaptive optimal control design," *Automatica*, vol. 71, pp. 348–360, Sep. 2016.

[85] W. Gao and Z.-P. Jiang, "Adaptive dynamic programming and adaptive optimal output regulation of linear systems," *IEEE Trans. Autom. Control*, vol. 61, no. 12, pp. 4164–4169, Dec. 2016.

[86] F.-Y. Wang, H. Zhang, and D. Liu, "Adaptive dynamic programming: An introduction," *IEEE Comput. Intell. Mag.*, vol. 4, no. 2, pp. 39–47, May 2009.

[87] F. L. Lewis and D. Vrabie, "Reinforcement learning and adaptive dynamic programming for feedback control," *IEEE Circuits Syst. Mag.*, vol. 9, no. 3, pp. 32–50, Jul. 2009.

[88] 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. Mag.*, vol. 32, no. 6, pp. 76–105, Dec. 2012.

[89] F. L. Lewis and D. Liu, *Reinforcement Learning and Approximate Dynamic Programming for Feedback Control*. Hoboken, NJ, USA: Wiley, 2012.

[90] D. Vrabie, K. G. Vamvoudakis, and F. L. Lewis, *Optimal Adaptive Control and Differential Games by Reinforcement Learning Principles*. London, U.K.: IET, 2013.

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

[92] H.-G. Zhang, X. Zhang, Y.-H. Luo, and J. Yang, "An overview of research on adaptive dynamic programming," *Acta Autom. Sinica*, vol. 39, no. 4, pp. 303–311, Apr. 2013.

[93] D.-R. Liu, H.-L. Li, and D. Wang, "Data-based self-learning optimal control: Research progress and prospects," *Acta Autom. Sinica*, vol. 39, no. 11, pp. 1858–1870, Nov. 2013.

[94] L. Busoniu, R. Babuska, B. D. Schutter, and D. Ernst, *Reinforcement Learning and Dynamic Programming Using Function Approximators*. Boca Raton, FL, USA: CRC Press, 2010.

[95] D. Liu, "Approximate dynamic programming for self-learning control," *Acta Autom. Sinica*, vol. 31, no. 1, pp. 13–18, Jan. 2005.

[96] D. Liu, Q. Wei, D. Wang, X. Yang, and H. Li, *Adaptive Dynamic Programming With Applications in Optimal Control*. London, U.K.: Springer, 2017.

[97] D. Liu and D. Wang, "Adaptive dynamic programming and reinforcement learning," in *UNESCO Encyclopedia of Life Support Systems*, H. Ichibushi Ed. New York, NY, USA: EOLSS, 2012, ch. 4.

[98] Y. Sokolov, R. Kozma, L. D. Werbos, and P. J. Werbos, "Complete stability analysis of a heuristic approximate dynamic programming control design," *Automatica*, vol. 59, pp. 9–18, Sep. 2015.

[99] D.-R. Liu, H.-L. Li, and D. Wang, "Feature selection and feature learning for high-dimensional batch reinforcement learning: A survey," *Int. J. Autom. Comput.*, vol. 12, no. 3, pp. 229–242, 2015.

[100] F.-Y. Wang, "Parallel control: A method for data-driven and computational control," *Acta Autom. Sinica*, vol. 39, no. 4, pp. 293–302, 2013.

[101] Z. Wang and D. Liu, "A data-based state feedback control method for a class of nonlinear systems," *IEEE Trans. Ind. Informat.*, vol. 9, no. 4, pp. 2284–2292, Nov. 2013.

[102] Z.-S. Hou and Z. Wang, "From model-based control to data-driven control: Survey, classification and perspective," *Inf. Sci.*, vol. 235, pp. 3–35, Jun. 2013.

[103] Z. Hou and S. Jin, "Data-driven model-free adaptive control for a class of MIMO nonlinear discrete-time systems," *IEEE Trans. Neural Netw.*, vol. 22, no. 12, pp. 2173–2188, Dec. 2011.

[104] P. Yan, D. Liu, D. Wang, and H. Ma, "Data-driven controller design for general MIMO nonlinear systems via virtual reference feedback tuning and neural networks," *Neurocomputing*, vol. 171, pp. 815–825, Jan. 2016.

[105] C. Mu, D. Wang, and H. He, "Novel iterative neural dynamic programming for data-based approximate optimal control design," *Automatica*, vol. 81, pp. 240–252, Jul. 2017.

[106] D. Wang, C.-X. Mu, and D.-R. Liu, "Data-driven nonlinear near-optimal regulation based on iterative neural dynamic programming," *Acta Autom. Sinica*, vol. 43, no. 3, pp. 366–375, 2017.

[107] J. Y. Lee, J. B. Park, and Y. H. Choi, "Integral Q-learning and explorized policy iteration for adaptive optimal control of continuous-time linear systems," *Automatica*, vol. 48, no. 11, pp. 2850–2859, 2012.

[108] J. Y. Lee, J. B. Park, and Y. H. Choi, "On integral generalized policy iteration for continuous-time linear quadratic regulations," *Automatica*, vol. 50, no. 2, pp. 475–489, 2014.

[109] J. Y. Lee, J. B. Park, and Y. H. Choi, "Integral reinforcement learning for continuous-time input-affine nonlinear systems with simultaneous invariant explorations," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 5, pp. 916–932, May 2015.

[110] E. Lavretsky and K. A. Wise, *Robust and Adaptive Control With Aerospace Applications*. London, U.K.: Springer-Verlag, 2013.

[111] M. Krstic, P. V. Kokotovic, and I. Kanellakopoulos, *Nonlinear and Adaptive Control Design*. New York, NY, USA: Wiley, 1995.

[112] F. L. Lewis, S. Jagannathan, and A. Yesildirek, *Neural Network Control of Robot Manipulators and Nonlinear Systems*. London, U.K.: Taylor & Francis, 1999.

[113] M. J. Corless and G. Leitmann, "Continuous state feedback guaranteeing uniform ultimate boundedness for uncertain dynamic systems," *IEEE Trans. Autom. Control*, vol. AC-26, no. 5, pp. 1139–1144, Oct. 1981.

[114] F. Lin, *Robust Control Design: An Optimal Control Approach*. Chichester, U.K.: Wiley, 2007.

[115] F. Lin, R. D. Brand, and J. Sun, "Robust control of nonlinear systems: Compensating for uncertainty," *Int. J. Control*, vol. 56, no. 6, pp. 1453–1459, 1992.

[116] D. M. Adhyaru, I. N. Kar, and M. Gopal, "Fixed final time optimal control approach for bounded robust controller design using Hamilton–Jacobi–Bellman solution," *IET Control Theory Appl.*, vol. 3, no. 9, pp. 1183–1195, Sep. 2009.

[117] D. M. Adhyaru, I. N. Kar, and M. Gopal, "Bounded robust control of nonlinear systems using neural network–based HJB solution," *Neural Comput. Appl.*, vol. 20, no. 1, pp. 91–103, 2011.

[118] D. Wang, D. Liu, and H. Li, "Policy iteration algorithm for online design of robust control for a class of continuous-time nonlinear systems," *IEEE Trans. Autom. Sci. Eng.*, vol. 11, no. 2, pp. 627–632, Apr. 2014.

[119] D. Wang, D. Liu, H. Li, and H. Ma, "Neural-network-based robust optimal control design for a class of uncertain nonlinear systems via adaptive dynamic programming," *Inf. Sci.*, vol. 282, pp. 167–179, Oct. 2014.

[120] D. Wang, D. Liu, Q. Zhang, and D. Zhao, "Data-based adaptive critic designs for nonlinear robust optimal control with uncertain dynamics," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 11, pp. 1544–1555, Nov. 2016.

[121] D. Liu, X. Yang, D. Wang, and Q. Wei, "Reinforcement-learning-based robust controller design for continuous-time uncertain nonlinear systems subject to input constraints," *IEEE Trans. Cybern.*, vol. 45, no. 7, pp. 1372–1385, Jul. 2015.

[122] D. Wang, D. Liu, H. Li, B. Luo, and H. Ma, "An approximate optimal control approach for robust stabilization of a class of discrete-time nonlinear systems with uncertainties," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 5, pp. 713–717, May 2016.

[123] D. Wang, "Adaptation-oriented near-optimal control and robust synthesis of an overhead crane system," in *Proc. Int. Conf. Neural Inf. Process.*, Guangzhou, China, Nov. 2017.

[124] 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 Netw.*, Dallas, TX, USA, Aug. 2013, pp. 1–8.

[125] J. Sun, C. Liu, and Q. Ye, "Robust differential game guidance laws design for uncertain interceptor-target engagement via adaptive dynamic programming," *Int. J. Control*, vol. 90, no. 5, pp. 990–1004, 2017.

[126] D. Wang, C. Li, D. Liu, and C. Mu, "Data-based robust optimal control of continuous-time affine nonlinear systems with matched uncertainties," *Inf. Sci.*, vol. 366, pp. 121–133, Oct. 2016.

[127] X. Yang, D. Liu, B. Luo, and C. Li, "Data-based robust adaptive control for a class of unknown nonlinear constrained-input systems via integral reinforcement learning," *Inf. Sci.*, vol. 369, pp. 731–747, Nov. 2016.

[128] Q.-Y. Fan and G.-H. Yang, "Adaptive actor–critic design-based integral sliding-mode control for partially unknown nonlinear systems with input disturbances," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 1, pp. 165–177, Jan. 2016.

[129] Y. Jiang and Z.-P. Jiang, "Robust adaptive dynamic programming for large-scale systems with an application to multimachine power systems," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 59, no. 10, pp. 693–697, Oct. 2012.

[130] Z.-P. 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.

[131] Y. Jiang and Z.-P. Jiang, "Robust adaptive dynamic programming and feedback stabilization of nonlinear systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 5, pp. 882–893, May 2014.

[132] T. Bian, Y. Jiang, and Z.-P. Jiang, "Decentralized adaptive optimal control of large-scale systems with application to power systems," *IEEE Trans. Ind. Electron.*, vol. 62, no. 4, pp. 2439–2447, Apr. 2015.

[133] W. Gao, Y. Jiang, Z.-P. Jiang, and T. Chai, "Output-feedback adaptive optimal control of interconnected systems based on robust adaptive dynamic programming," *Automatica*, vol. 72, pp. 37–45, Oct. 2016.

[134] Z.-P. Jiang, A. R. Teel, and L. Praly, "Small-gain theorem for ISS systems and applications," *Math. Control Signals Syst.*, vol. 7, no. 2, pp. 95–120, 1994.

[135] Y. Jiang and Z.-P. Jiang, "Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics," *Automatica*, vol. 48, no. 10, pp. 2699–2704, 2012.

[136] S. S. L. Chang and T. K. C. Peng, "Adaptive guaranteed cost control of systems with uncertain parameters," *IEEE Trans. Autom. Control*, vol. 17, no. 4, pp. 474–483, Aug. 1972.

[137] D. Liu, D. Wang, F.-Y. Wang, H. Li, and X. Yang, "Neural-network-based online HJB solution for optimal robust guaranteed cost control of continuous-time uncertain nonlinear systems," *IEEE Trans. Cybern.*, vol. 44, no. 12, pp. 2834–2847, Dec. 2014.

[138] D. Wang, D. Liu, H. Li, and H. Ma, "Adaptive dynamic programming for infinite horizon optimal robust guaranteed cost control of a class of uncertain nonlinear systems," in *Proc. Amer. Control Conf.*, Chicago, IL, USA, Jul. 2015, pp. 2900–2905.

[139] H.-N. Wu, M.-M. Li, and L. Guo, "Finite-horizon approximate optimal guaranteed cost control of uncertain nonlinear systems with application to mars entry guidance," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 7, pp. 1456–1467, Jul. 2015.

[140] X. Yang, D. Liu, Q. Wei, and D. Wang, "Guaranteed cost neural tracking control for a class of uncertain nonlinear systems using adaptive dynamic programming," *Neurocomputing*, vol. 198, pp. 80–90, Jul. 2016.

[141] C. Mu and D. Wang, "Neural-network-based adaptive guaranteed cost control of nonlinear dynamical systems with matched uncertainties," *Neurocomputing*, vol. 245, pp. 46–54, Jul. 2017.

[142] A. Saberi, "On optimality of decentralized control for a class of nonlinear interconnected systems," *Automatica*, vol. 24, no. 1, pp. 101–104, 1988.

[143] D. D. Siljak, *Decentralized Control of Complex Systems*. Boston, MA, USA: Academic Press, 2012.

[144] D. Liu, D. Wang, and H. Li, "Decentralized stabilization for a class of continuous-time nonlinear interconnected systems using online learning optimal control approach," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 2, pp. 418–428, Feb. 2014.

[145] D. Liu, C. Li, H. Li, D. Wang, and H. Ma, "Neural-network-based decentralized control of continuous-time nonlinear interconnected systems with unknown dynamics," *Neurocomputing*, vol. 165, pp. 90–98, Oct. 2015.

[146] C. Mu, C. Sun, D. Wang, A. Song, and C. Qian, "Decentralized adaptive optimal stabilization of nonlinear systems with matched interconnections," *Soft Comput.*, doi: 10.1007/s00500-017-2526-6.

[147] D. Wang, H. He, B. Zhao, and D. Liu, "Adaptive near-optimal controllers for non-linear decentralised feedback stabilisation problems," *IET Control Theory Appl.*, vol. 11, no. 6, pp. 799–806, Apr. 2017.

[148] S. E. Lyshevski, "Nonlinear discrete-time systems: Constrained optimization and application of nonquadratic costs," in *Proc. Amer. Control Conf.*, Philadelphia, PA, USA, Jun. 1998, pp. 3699–3703.

[149] H. Zhang, C. Qin, and Y. Luo, "Neural-network-based constrained optimal control scheme for discrete-time switched nonlinear system using dual heuristic programming," *IEEE Trans. Autom. Sci. Eng.*, vol. 11, no. 3, pp. 839–849, Jul. 2014.

[150] W. M. Haddad, V. Chellaboina, J. L. Fausz, and A. Leonessa, "Optimal non-linear robust control for non-linear uncertain systems," *Int. J. Control*, vol. 73, no. 4, pp. 329–342, 2000.

[151] W. M. Haddad and V. Chellaboina, *Nonlinear Dynamical Systems and Control: A Lyapunov-Based Approach*. Princeton, NJ, USA: Princeton Univ. Press, 2008.

[152] W. M. Haddad, V.-S. Chellaboina, and J. L. Fausz, "Robust nonlinear feedback control for uncertain linear systems with nonquadratic performance criteria," *Syst. Control Lett.*, vol. 33, no. 5, pp. 327–338, 1998.

[153] P. Tabuada, "Event-triggered real-time scheduling of stabilizing control tasks," *IEEE Trans. Autom. Control*, vol. 52, no. 9, pp. 1680–1685, Sep. 2007.

[154] P. Tallapragada and N. Chopra, "On event triggered tracking for nonlinear systems," *IEEE Trans. Autom. Control*, vol. 58, no. 9, pp. 2343–2348, Sep. 2013.

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

[156] 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.

[157] X. Zhong and H. He, "An event-triggered ADP control approach for continuous-time system with unknown internal states," *IEEE Trans. Cybern.*, vol. 47, no. 3, pp. 683–694, Mar. 2017.

[158] L. Dong, Y. Tang, H. He, and C. Sun, "An event-triggered approach for load frequency control with supplementary ADP," *IEEE Trans. Power Syst.*, vol. 32, no. 1, pp. 581–589, Jan. 2017.

[159] Y. Zhu, D. Zhao, H. He, and J. Ji, "Event-triggered optimal control for partially unknown constrained-input systems via adaptive dynamic programming," *IEEE Trans. Ind. Electron.*, vol. 64, no. 5, pp. 4101–4109, May 2017.

[160] D. Wang, C. Mu, H. He, and D. Liu, "Adaptive-critic-based event-driven nonlinear robust state feedback," in *Proc. IEEE Conf. Decis. Control*, Las Vegas, NV, USA, Dec. 2016, pp. 5813–5818.

[161] D. Wang, C. Mu, H. He, and D. Liu, "Event-driven adaptive robust control of nonlinear systems with uncertainties through NDP strategy," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published, doi: 10.1109/TSMC.2016.2592682.

[162] Q. Zhang, D. Zhao, and D. Wang, "Event-based robust control for uncertain nonlinear systems using adaptive dynamic programming," *IEEE Trans. Neural Netw. Learn. Syst.*, to be published, doi: 10.1109/TNNLS.2016.2614002.

[163] H. K. Khalil, *Nonlinear Systems*. 3rd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2002.

[164] T. Dierks and S. Jagannathan, "Optimal control of affine nonlinear continuous-time systems," in *Proc. Amer. Control Conf.*, Baltimore, MD, USA, Jun. 2010, pp. 1568–1573.

[165] H. Zhang, L. Cui, and Y. Luo, "Near-optimal control for nonzero-sum differential games of continuous-time nonlinear systems using single-network ADP," *IEEE Trans. Cybern.*, vol. 43, no. 1, pp. 206–216, Feb. 2013.

[166] X. Yang, D. Liu, H. Ma, and Y. Xu, "Online approximate solution of HJI equation for unknown constrained-input nonlinear continuous-time systems," *Inf. Sci.*, vol. 328, pp. 435–454, Jan. 2016.

[167] D. Wang and C. Mu, "Developing nonlinear adaptive optimal regulators through an improved neural learning mechanism," *Sci. China Inf. Sci.*, vol. 60, no. 5, pp. 058201:1–058201:3, 2017.

[168] D. Wang and C. Mu, "A novel neural optimal control framework with nonlinear dynamics: Closed-loop stability and simulation verification," *Neurocomputing*, to be published.

[169] T. Basar and P. Bernhard, *H_∞-Optimal Control and Related Minimax Design Problems: A Dynamic Game Approach*, 2nd ed. Boston, MA, USA: Birkhäuser, 2008.

[170] 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.

[171] M. Abu-Khalaf, F. L. Lewis, and J. Huang, "Policy iterations on the Hamilton–Jacobi–Isaacs equation for H_∞ state feedback control with input saturation," *IEEE Trans. Autom. Control*, vol. 51, no. 12, pp. 1989–1995, Dec. 2006.

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

[173] H. Modares, F. L. Lewis, and M. B. N. Sistani, "Online solution of nonquadratic two-player zero-sum games arising in the H_∞ control of constrained input systems," *Int. J. Adapt. Control Signal Process.*, vol. 28, nos. 3–5, pp. 232–254, 2014.

[174] B. Luo, H.-N. Wu, and T. Huang, "Off-policy reinforcement learning for H_∞ control design," *IEEE Trans. Cybern.*, vol. 45, no. 1, pp. 65–76, Jan. 2015.

[175] B. Luo, T. Huang, H.-N. Wu, and X. Yang, "Data-driven H_∞ control for nonlinear distributed parameter systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 11, pp. 2949–2961, Nov. 2015.

[176] C. Qin, H. Zhang, and Y. Luo, "Mode-free H_∞ control design for unknown continuous-time linear system by using adaptive dynamic programming," *Asian J. Control*, vol. 18, no. 2, pp. 609–618, 2016.

[177] H. Zhang, C. Qin, B. Jiang, and Y. Luo, "Online adaptive policy learning algorithm for H_∞ state feedback control of unknown affine nonlinear discrete-time systems," *IEEE Trans. Cybern.*, vol. 44, no. 12, pp. 2706–2718, Dec. 2014.

[178] C. Qin, H. Zhang, Y. Wang, and Y. Luo, "Neural network-based online H_∞ control for discrete-time affine nonlinear system using adaptive dynamic programming," *Neurocomputing*, vol. 198, pp. 91–99, Jul. 2016.

[179] D. Wang, C. Mu, Q. Zhang, and D. Liu, "Event-based input-constrained nonlinear H_∞ state feedback with adaptive critic and neural implementation," *Neurocomputing*, vol. 214, pp. 848–856, Nov. 2016.

[180] Q. Zhang, D. Zhao, and Y. Zhu, "Event-triggered H_∞ control for continuous-time nonlinear system via concurrent learning," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published, doi: 10.1109/TSMC.2016.2531680.

[181] P. J. Werbos, "Computational intelligence for the smart grid-history, challenges, and opportunities," *IEEE Comput. Intell. Mag.*, vol. 6, no. 3, pp. 14–21, Aug. 2011.

[182] J. Liang, G. K. Venayagamoorthy, and R. G. Harley, "Wide-area measurement based dynamic stochastic optimal power flow control for smart grids with high variability and uncertainty," *IEEE Trans. Smart Grid*, vol. 3, no. 1, pp. 59–69, Mar. 2012.

[183] C. Wang, D. R. Liu, Q. L. Wei, D. B. Zhao, and Z. C. 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.

[184] 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.

[185] J. Yan, H. He, X. Zhong, and Y. Tang, "Q-learning-based vulnerability analysis of smart grid against sequential topology attacks," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 1, pp. 200–210, Jan. 2017.

[186] 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.

[187] Q. Wei and D. Liu, "Data-driven neuro-optimal temperature control of water-gas shift reaction using stable iterative adaptive dynamic programming," *IEEE Trans. Ind. Electron.*, vol. 61, no. 11, pp. 6399–6408, Nov. 2014.

[188] Q. Wei and D. Liu, "Adaptive dynamic programming for optimal tracking control of unknown nonlinear systems with application to coal gasification," *IEEE Trans. Autom. Sci. Eng.*, vol. 11, no. 4, pp. 1020–1036, Oct. 2014.

[189] D. Zhao, Y. Dai, and Z. Zhang, "Computational intelligence in urban traffic signal control: A survey," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 42, no. 4, pp. 485–494, Jul. 2012.

[190] D. B. Zhao, D. R. Liu, and J. Q. Yi, "An overview on the adaptive dynamic programming based urban city traffic signal optimal control," *Acta Autom. Sinica*, vol. 35, no. 6, pp. 676–681, 2009.

[191] C. Mu, Y. Tang, and H. He, "Observer-based sliding mode frequency control for micro-grid with photovoltaic energy integration," in *Proc. IEEE Power Energy Soc. Gen. Meeting*, Boston, MA, USA, Jul. 2016, pp. 1–5.

[192] C. Mu, Y. Tang, and H. He, "Improved sliding mode design for load frequency control of power system integrated an adaptive learning strategy," *IEEE Trans. Ind. Electron.*, to be published, doi: 10.1109/TIE.2017.2694396.

[193] S. K. Pandey, S. R. Mohanty, and N. Kishor, "A literature survey on load–frequency control for conventional and distribution generation power systems," *Renew. Sustain. Energy Rev.*, vol. 25, pp. 318–334, Sep. 2013.

[194] K. P. S. Parmar, S. Majhi, and D. P. Kothari, "Load frequency control of a realistic power system with multi-source power generation," *Int. J. Elect. Power Energy Syst.*, vol. 42, no. 1, pp. 426–433, 2012.

[195] M.-S. Park, D. Chwa, and S.-K. Hong, "Antisway tracking control of overhead cranes with system uncertainty and actuator nonlinearity using an adaptive fuzzy sliding-mode control," *IEEE Trans. Ind. Electron.*, vol. 55, no. 11, pp. 3972–3984, Nov. 2008.

[196] D. Qian, J. Yi, and D. Zhao, "Control of overhead crane systems by combining sliding mode with fuzzy regulator," in *Proc. World Congr. Int. Federation Autom. Control*, Milan, Italy, Aug. 2011, pp. 9320–9325.

[197] H. Modares and F. L. Lewis, "Linear quadratic tracking control of partially-unknown continuous-time systems using reinforcement learning," *IEEE Trans. Autom. Control*, vol. 59, no. 11, pp. 3051–3056, Nov. 2014.

[198] D. Wang, H. He, X. Zhong, and D. Liu, "Event-driven nonlinear discounted optimal regulation involving a power system application," *IEEE Trans. Ind. Electron.*, to be published, doi: 10.1109/TIE.2017.2698377.

[199] D. P. Bertsekas, *Abstract Dynamic Programming*. Belmont, MA, USA: Athena Sci., 2013.

[200] D. P. Bertsekas, "Value and policy iterations in optimal control and adaptive dynamic programming," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 3, pp. 500–509, Mar. 2017.

[201] D. Wang, C. Mu, D. Liu, and H. Ma, "On mixed data and event driven design for adaptive-critic-based nonlinear H_∞ control," *IEEE Trans. Neural Netw. Learn. Syst.*, to be published, doi: 10.1109/TNNLS.2016.2642128.

[202] K. G. Vamvoudakis and H. Ferraz, "Event-triggered H-infinity control for unknown continuous-time linear systems using Q-learning," in *Proc. IEEE Conf. Decis. Control*, Las Vegas, NV, USA, Dec. 2016, pp. 1376–1381.

[203] H. Zhang, T. Feng, G.-H. Yang, and H. Liang, "Distributed cooperative optimal control for multiagent systems on directed graphs: An inverse optimal approach," *IEEE Trans. Cybern.*, vol. 45, no. 7, pp. 1315–1326, Jul. 2015.

[204] H. Ma *et al.*, "Neural-network-based distributed adaptive robust control for a class of nonlinear multiagent systems with time delays and external noises," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 6, pp. 750–758, Jun. 2016.

[205] P. J. Werbos, "Building and understanding adaptive systems: A statistical/numerical approach to factory automation and brain research," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-17, no. 1, pp. 7–20, Jan. 1987.

[206] P. J. Werbos, "Using ADP to understand and replicate brain intelligence: The next level design," in *Proc. IEEE Symp. Approx. Dyn. Program. Reinforcement Learn.*, Honolulu, HI, USA, Apr. 2007, pp. 209–216.

[207] P. J. Werbos, "ADP: The key direction for future research in intelligent control and understanding brain intelligence," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 38, no. 4, pp. 898–900, Aug. 2008.

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

Ding Wang (M'15) received the B.S. degree in mathematics from the Zhengzhou University of Light Industry, Zhengzhou, China, in 2007, the M.S. degree in operations research and cybernetics from Northeastern University, Shenyang, China, in 2009, and the Ph.D. degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2012.

He was a Visiting Scholar with the Department of Electrical, Computer, and Biomedical Engineering, University of Rhode Island, Kingston, RI, USA, from 2015 to 2017. He is currently an Associate Professor with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences. He has published over 100 journal and conference papers, and coauthored two monographs. His current research interests include adaptive and learning systems, computational intelligence, and intelligent control.

Dr. Wang was a recipient of the Excellent Doctoral Dissertation Award of the Chinese Academy of Sciences in 2013, and a nomination of the Excellent Doctoral Dissertation Award of the Chinese Association of Automation in 2014. He was selected for the Young Elite Scientists Sponsorship Program by the China Association for Science and Technology in 2017. He serves as an Associate Editor for the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS and *Neurocomputing*. He is the Publications Chair of the 24th International Conference on Neural Information Processing in 2017.

Haibo He (SM'11) received the B.S. and M.S. degrees in electrical engineering from the Huazhong University of Science and Technology, Wuhan, China, in 1999 and 2002, respectively, and the Ph.D. degree in electrical engineering from Ohio University, Athens, OH, USA, in 2006.

From 2006 to 2009, he was an Assistant Professor with the Department of Electrical and Computer Engineering, Stevens Institute of Technology, Hoboken, NJ, USA. He is currently the Robert Haas Endowed Chair Professor with the Department of Electrical, Computer, and Biomedical Engineering, University of Rhode Island, Kingston, RI, USA. He has published one sole-author research book (Wiley), edited one book (Wiley—IEEE) and six conference proceedings (Springer), and authored and co-authored over 250 peer-reviewed journal and conference papers. His current research interests include adaptive dynamic programming, computational intelligence, machine learning and data mining, and various applications.

Dr. He was a recipient of the IEEE International Conference on Communications Best Paper Award in 2014, the IEEE Computational Intelligence Society Outstanding Early Career Award in 2014, the National Science Foundation CAREER Award in 2011, and the Providence Business News "Rising Star Innovator Award" in 2011. He is currently the Editor-in-Chief of the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS.

Derong Liu (S'91–M'94–SM'96–F'05) received the Ph.D. degree in electrical engineering from the University of Notre Dame, Notre Dame, IN, USA, in 1994.

He was a Staff Fellow with General Motors Research and Development Center, Detroit, MI, USA, from 1993 to 1995. He was an Assistant Professor with the Department of Electrical and Computer Engineering, Stevens Institute of Technology, Hoboken, NJ, USA, from 1995 to 1999. He joined the University of Illinois at Chicago, Chicago, IL, USA, in 1999, and became a Full Professor of Electrical and Computer Engineering and Computer Science in 2006. He served as the Associate Director of the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China. He has published 18 books.

Dr. Liu was a recipient of the Faculty Early Career Development Award from the National Science Foundation in 1999, the University Scholar Award from the University of Illinois, from 2006 to 2009, the Overseas Outstanding Young Scholar Award from the National Natural Science Foundation of China, in 2008, and the Outstanding Achievement Award from the Asia Pacific Neural Network Assembly in 2014. He was selected for the 100 Talents Program by the Chinese Academy of Sciences in 2008. He is the Editor-in-Chief of *Artificial Intelligence Review* (Springer). He was the Editor-in-Chief of the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS from 2010 to 2015. He is a fellow of the International Neural Network Society and the International Association of Pattern Recognition.