Research Article

Neural Network-Based Intelligent Computing Algorithms for Discrete-Time Optimal Control with the Application to a Cyberphysical Power System

Feng Jiang, Kai Zhang, Jinjing Hu, and Shunjiang Wang

State Grid Liaoning Electric Power Company Limited, Shenyang 110006, China

Correspondence should be addressed to Shunjiang Wang; wangshunjiang@163.com

Received 26 February 2021; Revised 14 March 2021; Accepted 7 April 2021; Published 17 April 2021

Academic Editor: Qiuye Sun

Copyright © 2021 Feng Jiang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Adaptive dynamic programming (ADP), which belongs to the field of computational intelligence, is a powerful tool to address optimal control problems. To overcome the bottleneck of solving Hamilton–Jacobi–Bellman equations, several state-of-the-art ADP approaches are reviewed in this paper. First, two model-based offline iterative ADP methods including policy iteration (PI) and value iteration (VI) are given, and their respective advantages and shortcomings are discussed in detail. Second, the multistep heuristic dynamic programming (HDP) method is introduced, which avoids the requirement of initial admissible control and achieves fast convergence. This method successfully utilizes the advantages of PI and VI and overcomes their drawbacks at the same time. Finally, the discrete-time optimal control strategy is tested on a power system.

1. Introduction

Adaptive dynamic programming (ADP) [1–4], which integrates the advantages of reinforcement learning (RL) [5–8] and adaptive control, has become a powerful tool in solving optimal control problems. With decades of development, ADP has also provided many approaches to solve other control problems, such as robust control [9, 10], optimal control with input constraints [11, 12], optimal tracking control [13, 14], zero-sum games [15], and non-zero-sum games [16]. Furthermore, ADP methods have been widely applied to the real-world systems, such as water-gas shift reaction [17], battery management [18], microgrid systems [19, 20], and Quanser helicopter [21]. These aforementioned papers were all inspired and developed by the basic works of ADP-based optimal control; i.e., optimal control is the core research topic of ADP.

The bottleneck of solving the nonlinear optimal control problems is to obtain the solutions of Hamilton–Jacobi–Bellman (HJB) equations. However, these equations are generally difficult or even impossible to be

solved analytically. To overcome this difficulty, ADP has given several important iterative learning frameworks, such as policy iteration (PI) [2, 22, 23] and value iteration (VI) [24–26]. PI algorithm starts from an initial admissible control policy and then proceeds the policy evaluation step and the policy improvement step successively till convergence. The main advantage of PI is that it ensures all the iterative control policies are admissible and achieves fast convergence. The drawback of PI is also obvious. The requirement of initial admissible control is a strict condition in practice, which seriously limits its applications. Different from PI, VI can start from an arbitrary-positive semidefinite value function, which is an easy-to-realize initial condition. Although the easier initial condition makes VI more practical, it also leads to a longer iteration learning process; that is, VI achieves convergence much slower than PI. Thus, it is desired to develop a new method, which avoids the requirement of initial admissible control and gets convergence faster than the VI algorithm. To realize these purposes, the multistep heuristic dynamic programming (HDP) approach [27] is presented to integrate the merits of PI and VI algorithms and overcome their drawbacks.

This paper reviews the state-of-the-art ADP algorithms for the optimal control of discrete-time (DT) systems. The rest of this paper is arranged as follows. In Section 2, the problem formulation is derived. Three iterative model-based offline learning algorithms along with comprehensive comparisons are presented in Sections 3 and 4. The proposed DT optimal control strategy is tested on a power system in Section 5. Finally, a brief conclusion is drawn in Section 6.

2. Problem Formulation

In this paper, we consider the general nonlinear DT system:

$$x(k+1) = f(x(k)) + g(x(k))u(k), \quad (1)$$

where $x(k) \in \mathbb{R}^n$ represents the system state, $u(k) \in \mathbb{R}^m$ denotes the control input, and $f(x) \in \mathbb{R}^n$ and $g(x) \in \mathbb{R}^{n \times m}$ are the system functions.

The purpose of the optimal control issue is to find out a state feedback control policy $u(x(k))$, which can not only stabilize system (1) but also minimize the following performance index function:

$$J(x(0), u) = \sum_{p=0}^{\infty} r(x(p), u(p)), \quad (2)$$

where $r(x, u) = x^T Q x + u^T R u$. The matrices Q and R determine the performance of system states and control inputs, respectively. Given the admissible control policy $u(x(k))$, the value function can be described by

$$\begin{aligned} V(x(k)) &= \sum_{p=k}^{\infty} r(x(p), u(p)) = r(x(k), u(k)) \\ &+ \sum_{p=k+1}^{\infty} r(x(p), u(p)) \\ &= r(x(k), u(k)) + V(x(k+1)). \end{aligned} \quad (3)$$

According to the definition of optimal control, the optimal value function can be defined by

$$\begin{aligned} V^*(x(k)) &= \min_{u(x)} \sum_{p=k}^{\infty} r(x(p), u(p)) \\ &= \min_{u(x)} \{r(x(k), u(k)) + V^*(x(k+1))\}. \end{aligned} \quad (4)$$

By using the stationarity condition [28], the optimal control policy can be derived as

$$\begin{aligned} u^*(x(k)) &= \arg \min_{u(x)} \{r(x(k), u(k)) + V^*(x(k+1))\} \\ &= \arg \min_{u(x)} \{r(x(k), u(k)) + V^*(f(x(k)) \\ &\quad + g(x(k))u(k))\} \\ &= -\frac{1}{2}R^{-1}g^T(x(k))\nabla V^*(x(k+1)), \end{aligned} \quad (5)$$

where $\nabla V^*(x(k+1)) = \partial V^*(x(k+1))/\partial x(k+1)$.

The key to obtaining the optimal control policy $u^*(x(k))$ is to solve the following DT HJB equation [27]:

$$\begin{aligned} V^*(x(k)) &= r(x(k), u^*(k)) + V^*(x(k+1)) \\ &= \frac{1}{4}\nabla V^{*T}(x(k+1))g(x(k))R^{-1}g^T(x(k))\nabla V^* \\ &\quad (x(k+1)) + x^T(k)Qx(k) + V^*(x(k+1)). \end{aligned} \quad (6)$$

Remark 1. Figure 1 provides the relationship and difference between discrete-time and continuous-time optimal control. The real-world systems generally exist in the continuous-time forms. After mathematical modeling, they are formulated by the continuous-time system models. Through sampling and discretization, the continuous-time system models are converted into the discrete-time ones. Therefore, the associated performance indexes and HJB equations of discrete-time systems are in the discretization forms compared with the continuous-time systems. The key to solving the discrete-time optimal control issue is the discrete-time HJB equation, which is a nonlinear partial difference equation. The existing works regarding continuous-time systems are much more than the ones regarding discrete-time systems. In order to overcome this bottleneck, several ADP learning algorithms along with their neural network (NN) implementations will be introduced.

3. Model-Based PI Algorithm for the Optimal Control Problem of DT Systems

In this section, the model-based PI algorithm along with its NN implementation will be introduced in detail. The model-based PI algorithm [2, 23] is shown in Algorithm 1.

The actor-critic dual-network structure with the gradient-descent updating law is employed to implement Algorithm 1. First, construct the critic NN to approximate the iterative value function:

$$\hat{V}^{l,q}(x(k)) = \hat{W}_c^{l,qT} \phi_c(x(k)), \quad (7)$$

where $\hat{W}_c^{l,q}$ and $\phi_c(x)$ denote the NN weights and NN activation functions of the critic network and q is the iteration index for the following gradient-descent method.

Define the error function for the critic NN:

$$\tilde{V}^{l,q}(x(k)) = \hat{V}^{l,q}(x(k)) - V^l(x(k)), \quad (8)$$

where $V^l(x(k)) = \sum_{h=0}^{H-1} r(x(k+h), u^l(k+h)) + V^l(x(k+H))$. If we select a large enough integer H , then, with the admissible control $u^l(k)$, one has $V^l(x(k+H)) \rightarrow 0$ [2]; that is, $V^l(x(k))$ can be expressed as $V^l(x(k)) = \sum_{h=0}^{H-1} r(x(k+h), u^l(k+h))$.

In order to minimize the error performance $E_c^{l,q}(k) = (\tilde{V}^{l,q}(x(k)))^2/2$, the gradient-descent-based updating law for the critic NN is given by

FIGURE 1: Relationship and difference between discrete-time and continuous-time optimal control.

Step 1: (Initialization)

Let the iteration index $l = 0$.

Select an initial admissible control policy $u^0(x)$.

Choose a small enough computation precision ϵ .

Step 2: (Policy Evaluation)

With $u^l(x)$, compute the iterative value function $V^l(x)$ by

$$V^l(x(k)) = r(x(k), u^l(k)) + V^l(x(k+1)).$$

Step 3: (Policy Improvement)

With $V^l(x)$, update the iterative control policy $u^{l+1}(x)$ by

$$u^{l+1}(x(k)) = -(1/2)R^{-1}g^T(x(k))\nabla V^l(x(k+1)).$$

Step 4: if $\|V^l - V^{l-1}\| \leq \epsilon$, stop and the optimal control policy $u^{l+1}(x)$ is acquired; Else, let $l = l + 1$ and go back to Step 2.

ALGORITHM 1: Model-based PI.

$$\begin{aligned} \hat{W}_c^{l,q+1} &= \hat{W}_c^{l,q} - \beta_c \left[\frac{\partial E_c^{l,q}(k)}{\partial \tilde{V}^{l,q}(x(k))} \frac{\partial \tilde{V}^{l,q}(x(k))}{\partial \tilde{V}^{l,q}(x(k))} \frac{\partial \tilde{V}^{l,q}(x(k))}{\partial \hat{W}_c^{l,q}} \right] \\ &= \hat{W}_c^{l,q} - \beta_c \tilde{V}^{l,q}(x(k)) \phi_c(x(k)), \end{aligned} \quad (9)$$

Similar to the design of critic NN, the actor network, which is used to approximate the iterative control policy, is expressed as

$$\hat{u}^{l,q}(x(k)) = \hat{W}_a^{l,qT} \phi_a(x(k)). \quad (10)$$

The error function for the actor NN is defined as

$$\tilde{u}^{l,q}(x(k)) = \hat{u}^{l,q}(x(k)) - u^l(x(k)), \quad (11)$$

where β_c is the learning rate of the critic NN.

where $u^l(x(k))$ can be attained according to Algorithm 1.

To minimize the error performance $E_a^{l,q}(k) = \tilde{u}^{l,qT}(x(k))\tilde{u}^{l,q}(x(k))/2$, using the chain derivation rule, the updating law for the actor NN is designed by

$$\begin{aligned}\hat{W}_a^{l,q+1} &= \hat{W}_a^{l,q} - \beta_a \left[\frac{\partial E_a^{l,q}(k)}{\partial \tilde{u}^{l,q}(x(k))} \frac{\partial \tilde{u}^{l,q}(x(k))}{\partial \hat{W}_a^{l,q}} \right] \\ &= \hat{W}_a^{l,q} - \beta_a \phi_a(x(k)) \tilde{u}^{l,qT}(x(k)),\end{aligned}\quad (12)$$

where β_a is the learning rate of the actor NN.

Remark 2. Figure 2 displays the NN implementation diagram of PI algorithm. First, NN weights of the actor network should be chosen to generate admissible control. Second, critic and actor networks are updated via the gradient-descent-based learning law to realize policy evaluation and improvement steps, respectively. After iteration, critic and actor networks achieve convergence, where the NN-based approximate optimal control can be obtained. Many stability proofs of the NN implementation procedure have been given in the existing works. Here, we introduce the following rigorous proof to demonstrate the optimality and convergence.

Theorem 1. Let the target iterative value function and control policy be described by $V^l(x(k)) = W_c^{lT} \phi_c(x(k))$ and $u^l(x(k)) = W_a^{lT} \phi_a(x(k))$, respectively. Let the critic and actor NNs be updated via (9) and (12), respectively. If the learning rates β_c and β_a are selected to be appropriately small, then the NN weights $\hat{W}_c^{l,q}$ and $\hat{W}_a^{l,q}$ will asymptotically converge to the ideal values W_c^l and W_a^l , respectively.

Proof. Let $\tilde{W}_c^{l,q} = \hat{W}_c^{l,q} - W_c^l$ and $\tilde{W}_a^{l,q} = \hat{W}_a^{l,q} - W_a^l$. According to (9) and (12), it can be acquired that

$$\begin{aligned}\tilde{W}_c^{l,q+1} &= \tilde{W}_c^{l,q} - \beta_c \tilde{V}^{l,q}(x(k)) \phi_c(x(k)), \\ \tilde{W}_a^{l,q+1} &= \tilde{W}_a^{l,q} - \beta_a \phi_a(x(k)) \tilde{u}^{l,qT}(x(k)),\end{aligned}\quad (13)$$

where $\tilde{V}^{l,q}(x(k)) = \tilde{W}_c^{l,qT} \phi_c(x(k))$ and $\tilde{u}^{l,q}(x(k)) = \tilde{W}_a^{l,qT} \phi_a(x(k))$.

Construct the following Lyapunov function candidate:

$$P(\tilde{W}_c^{l,q}, \tilde{W}_a^{l,q}) = \text{tr} \left\{ \tilde{W}_c^{l,qT} \tilde{W}_c^{l,q} + \tilde{W}_a^{l,qT} \tilde{W}_a^{l,q} \right\}. \quad (14)$$

The difference of the Lyapunov function (14) can be derived as

$$\begin{aligned}\Delta P(\tilde{W}_c^{l,q}, \tilde{W}_a^{l,q}) &= \text{tr} \left\{ \tilde{W}_c^{l,q+1T} \tilde{W}_c^{l,q+1} - \tilde{W}_c^{l,qT} \tilde{W}_c^{l,q} \right\} + \text{tr} \left\{ \tilde{W}_a^{l,q+1T} \tilde{W}_a^{l,q+1} - \tilde{W}_a^{l,qT} \tilde{W}_a^{l,q} \right\} \\ &\leq \beta_c \|\tilde{V}^{l,q}(x(k))\|^2 (\beta_c \|\phi_c(x(k))\|^2 - 2) \\ &\quad + \beta_a \|\tilde{u}^{l,q}(x(k))\|^2 (\beta_a \|\phi_a(x(k))\|^2 - 2).\end{aligned}\quad (15)$$

If the learning rates are selected to satisfy $\beta_c \leq 2/\|\phi_c(x(k))\|^2$ and $\beta_a \leq 2/\|\phi_a(x(k))\|^2$, then one has $\Delta P(\tilde{W}_c^{l,q}, \tilde{W}_a^{l,q}) \leq 0$, which implies the NN weights $\hat{W}_c^{l,q}$ and $\hat{W}_a^{l,q}$ will asymptotically converge to the ideal values.

This completes the proof. $\square$

4. Model-Based VI Algorithm and Multistep HDP Algorithm

With the help of the initial admissible control, the PI algorithm achieves fast convergence. However, the weakness of the PI algorithm is obvious. The PI algorithm requires the initial control policy to be admissible, which is a strict condition. How to find out an initial admissible control policy is still an open problem, which limits the real-world applications of the PI algorithm. To relax the strict condition, the model-based VI algorithm [24–26] is shown in Algorithm 2, where the initial condition becomes much easier.

Remark 3. Different from the PI algorithm, the VI algorithm does not require the initial admissible control, and one only needs to provide a specific initial value function, which makes the VI algorithm more practical in the real-world applications. However, without the help of the initial admissible control, the VI algorithm generally suffers from the low convergence speed. From the aforementioned content, it can be observed that the PI and VI algorithms have their own advantages and disadvantages. The PI algorithm can achieve fast convergence, while it requires an initial admissible control policy. The VI algorithm can start from an easy-to-realize initial condition, while it generally suffers from the low convergence speed. Thus, it is expected to design a new approach, which can make the trade-off between the PI algorithm and the VI algorithm.

That is, it is desired to develop an algorithm, which achieves convergence faster than the VI algorithm and does not require an initial admissible control policy. To realize this goal, the multistep HDP method [27] will be introduced in Algorithm 3.

FIGURE 2: NN implementation diagram of PI algorithm.

Step 1: (Initialization)Let the iteration index $l = 0$.Select an initial value function $V^0(x)$.Choose a small enough computation precision ϵ .**Step 2: (Policy Improvement)**With $V^l(x)$, compute the iterative control policy $u^l(x)$ by $u^l(x(k)) = -(1/2)R^{-1}g^T(x(k))\nabla V^l(x(k+1))$.**Step 3: (Policy Evaluation)**With $u^l(x)$, calculate the iterative value function $V^{l+1}(x)$ by $V^{l+1}(x(k)) = r(x(k), u^l(k)) + V^l(x(k+1))$.**Step 4:** if $\|V^{l+1} - V^l\| \leq \epsilon$, stop and the optimal control policy $u^l(x)$ is acquired; Else, let $l = l + 1$ and go back to Step 2.

ALGORITHM 2: Model-based VI.

Construct the critic and actor NNs to approximate the iterative value function and control policy as follows:

$$\begin{aligned}\hat{V}^l(x(k)) &= \phi_c^T(x(k))\hat{W}_c^l, \\ \hat{u}^l(x(k)) &= \phi_a^T(x(k))\hat{W}_a^l,\end{aligned}\quad (16)$$

where $\hat{W}_c^l$ and $\hat{W}_a^l$ are the NN weights and $\phi_c(x)$ and $\phi_a(x)$ are the associated NN activation functions.

According to Algorithm 3, using the NNs to estimate the solutions will yield the following error:

$$e_c^l = \phi_c^T(x(k))\hat{W}_c^{l+1} - \sum_{p=k}^{k+\alpha-1} r(x(p), \hat{u}^l(p)) - \phi_c^T(x(k+\alpha))\hat{W}_c^l. \quad (17)$$

Let $\theta = \phi_c^T(x(k))$ and $\sigma^l = \sum_{p=k}^{k+\alpha-1} r(x(p), \hat{u}^l(p)) + \phi_c^T(x(k+\alpha))\hat{W}_c^l$. Equation (17) becomes

$$e_c^l = \theta\hat{W}_c^{l+1} - \sigma^l. \quad (18)$$

To minimize $\|e_c^l\|^2$, we employ the least-square method to update $\hat{W}_c^{l+1}$. Collect M different data sets for training, where M is a large enough number. Then, one has $\Pi = [\theta_1^T, \theta_2^T, \dots, \theta_M^T]^T$ and $\Theta^l = [\sigma_1^T, \sigma_2^T, \dots, \sigma_M^T]^T$. The least-square-based updating law for $\hat{W}_c^{l+1}$ is given by

$$\hat{W}_c^{l+1} = (\Pi^T \Pi)^{-1} \Pi^T \Theta^l. \quad (19)$$

To minimize $r(x(k), \hat{u}^l(x(k))) + \hat{V}^l(x(k+1))$, the gradient-descent-based updating law for the actor NN is given by

Let the iteration index $l = 0$.
 Select an initial value function $V^0(x)$.
 Choose a small enough computation precision ϵ .
Step 2: (Policy Improvement)
 With $V^l(x)$, compute the iterative control policy $u^l(x)$ by
 $u^l(x(k)) = -(1/2)R^{-1}g^T(x(k))\nabla V^l(x(k+1))$.
Step 3: (Multistep Policy Evaluation)
 With $u^l(x)$, calculate the iterative value function $V^{l+1}(x)$ by
 $V^{l+1}(x(k)) = \sum_{p=k}^{k+\alpha-1} r(x(p), u^l(p)) + V^l(x(k+\alpha))$.
Step 4: if $\|V^{l+1} - V^l\| \leq \epsilon$, stop and the optimal control policy $u^l(x)$ is acquired;
 Else, let $l = l + 1$ and go back to Step 2.

ALGORITHM 3: Multistep HDP.

$$\begin{aligned}\hat{W}_a^{l,q+1} &= \hat{W}_a^{l,q} - \beta_a \frac{\partial \left(r(x(k), \hat{u}^{l,q}(x(k))) + \hat{V}^l(x(k+1)) \right)}{\partial \hat{u}^{l,q}(x(k))} \frac{\partial \hat{u}^{l,q}(x(k))}{\partial \hat{W}_a^{l,q}} \\ &= \hat{W}_a^{l,q} - \beta_a \phi_a(x(k)) \left[2R\hat{u}^{l,q}(x(k)) + g^T(x(k)) \frac{\partial \phi_c(x(k+1))}{\partial x(k+1)} \hat{W}_c^l \right].\end{aligned}\quad (20)$$

Remark 4. From Table 1 and Figure 3, we can see the performance comparison and relationship among PI algorithm, VI algorithm, and multistep HDP. Due to the existence of initial admissible control, the PI algorithm gets fast convergence. However, the condition of initial admissible control is difficult to realize. Different from the PI algorithm, the initial condition of VI algorithm is easy-to-realize. However, the initial condition may not be admissible, which may lead to the low stability. Multistep HDP follows the initial condition of VI algorithm and develops the multistep policy evaluation step to obtain more history data. Therefore, multistep HDP is easy-to-realize and achieves fast convergence at the same time; that is, multistep HDP successfully combines the advantages of PI and VI algorithms.

5. Application to a Benchmark Power System

The benchmark power system investigated in this paper is illustrated in Figure 4. This power system can be regarded as a microgrid, which is composed of nonpolluting energy (subsystems I and II), load demand sides (subsystem III), and regular generations (subsystem IV). The core control unit is the management center, which maintains the frequency stability against load variations.

5.1. System Model and Application. In Figure 5, first, the real-world power system can be formulated by a state space function via mathematical modeling. After sampling and discretization, the system model can be controlled by computers. Through iterative ADP learning, the approximate optimal control can be obtained. Substituting the approximate optimal control into the system model will yield simulation results. To test the effectiveness of the

proposed DT optimal control strategy, let us consider the following power system [19, 20]:

$$\begin{aligned}\Delta \dot{\xi}_f &= -\frac{1}{T_p} \Delta \xi_f + \frac{\alpha_p}{T_p} \Delta \xi_t, \\ \Delta \dot{\xi}_t &= -\frac{1}{T_t} \Delta \xi_t + \frac{1}{T_t} \Delta \xi_g, \\ \Delta \dot{\xi}_g &= -\frac{1}{\alpha_s T_g} \Delta \xi_f - \frac{1}{T_g} \Delta \xi_g + \frac{1}{T_g} u,\end{aligned}\quad (21)$$

where $\Delta \xi_f$ is the frequency deviation; $\Delta \xi_t$ denotes the turbine power; $\Delta \xi_g$ represents the governor position value; T_t , T_g , and T_p denote the time constants of turbine, governor, and power system, respectively; α_p represents the gain of power system; α_s is the speed regulation coefficient; u denotes the control input; and x is the state variable. Let $x = [\Delta \psi_f, \Delta \psi_t, \Delta \psi_g]^T$, where $x_1 = \Delta \psi_f$, $x_2 = \Delta \psi_t$, and $x_3 = \Delta \psi_g$. Then, the system (21) can be discretized as the form of (1). Set the matrices in the performance index function: $Q = 2I_3$ and $R = 1$.

5.2. Simulation Results. Simulation results are shown in Figure 6. Figure 6(a) implies the system states cannot be stabilized without control. Then, we apply the optimal control strategy into the system. Figure 6(b) indicates the system states can be stabilized after 8 time steps under optimal control. Comparing the trajectories of the system states, the superior control performance of optimal control strategy can be observed. Figure 6(c) shows the 2D plot of convergence trajectory in detail. Figure 6(d) provides the evolution of the control input. The aforementioned simulation results demonstrate the high stability, fast convergence, and low control cost of the DT optimal control strategy.

TABLE 1: Performance comparison among PI algorithm, VI algorithm, and multistep HDP.

Methods and performance	Computation stability	Initial difficulty	Convergence speed
PI algorithm	High	Difficult	High
VI algorithm	Low	Easy	Low
Multistep HDP	Medium	Easy	Medium

FIGURE 3: Relationship and difference among PI algorithm, VI algorithm, and multistep HDP.

FIGURE 4: The benchmark power system considered in this paper.

FIGURE 5: Application to the benchmark power system.

(a)

(b)

FIGURE 6: Continued.

FIGURE 6: Simulation results: (a) system states without control; (b) system states with optimal control; (c) 2D plot of x_1 and x_2 ; (d) control input.

6. Conclusions

In this paper, several state-of-the-art ADP-based methods have been reviewed to address the optimal control problem of DT systems. A comprehensive comparison has been made between PI and VI. A novel multistep HDP method has been introduced to integrate the advantages of PI and VI algorithms with either strict requirement of initial admissible control or longer interaction learning process. The simulation results have demonstrated the effectiveness of our proposed schemes.

Data Availability

Data are available upon request to the corresponding author.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the Science and Technology Foundation of SGCC (Grant no. SGLNDK00DWJS1900036).

References

- [1] Y. Yang, C. Xu, D. Yue, X. Zhong, X. Si, and J. Tan, "Event-triggered ADP control of a class of non-affine continuous-time nonlinear systems using output information," *Neurocomputing*, vol. 378, pp. 304–314, 2020.
- [2] D. Liu, Q. Wei, and P. Yan, "Generalized policy iteration adaptive dynamic programming for discrete-time nonlinear systems," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 45, no. 12, pp. 1577–1591, 2015.
- [3] R. Wang, Q. Sun, W. Hu, Y. Li, D. Ma, and P. Wang, "SoC-based droop coefficients stability region analysis of the battery for stand-alone supply systems with constant power loads," *IEEE Transactions on Power Electronics*, vol. 36, no. 7, pp. 7866–7879, 2021.
- [4] B. Luo, D. Liu, H.-N. Wu, D. Wang, and F. L. Lewis, "Policy gradient adaptive dynamic programming for data-based optimal control," *IEEE Transactions on Cybernetics*, vol. 47, no. 10, pp. 3341–3354, 2017.
- [5] R. Wang, Q. Sun, D. Ma, and Z. Liu, "The small-signal stability analysis of the droop-controlled converter in electromagnetic timescale," *IEEE Transactions on Sustainable Energy*, vol. 10, no. 3, pp. 1459–1469, 2019.
- [6] D. Liu, X. Yang, D. Wang, and Q. Wei, "Reinforcement-learning-based robust controller design for continuous-time uncertain nonlinear systems subject to input constraints," *IEEE Transactions on Cybernetics*, vol. 45, no. 7, pp. 1372–1385, 2015.
- [7] R. Wang, Q. Sun, D. Ma, and X. Hu, "Line impedance co-operative stability region identification method for grid-tied inverters under weak grids," *IEEE Transactions on Smart Grid*, vol. 11, no. 4, pp. 2856–2866, 2020.
- [8] D. Zhao and Y. Zhu, "MEC-a near-optimal online reinforcement learning algorithm for continuous deterministic systems," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 2, pp. 346–356, 2015.
- [9] Q. Zhang, D. Zhao, and D. Wang, "Event-Based robust control for uncertain nonlinear systems using adaptive dynamic programming," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 1, pp. 37–50, 2018.
- [10] Y. Wang, J. Sun, H. He, and C. Sun, "Deterministic policy gradient with integral compensator for robust quadrotor control," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 10, pp. 3713–3725, 2020.
- [11] X. Yang and H. He, "Event-triggered robust stabilization of nonlinear input-constrained systems using single network adaptive critic designs," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 9, pp. 3145–3157, 2020.
- [12] B. Luo, D. Liu, and H.-N. Wu, "Adaptive constrained optimal control design for data-based nonlinear discrete-time systems with critic-only structure," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 6, pp. 2099–2111, 2018.
- [13] W. Zhou, H. Liu, H. He, J. Yi, and T. Li, "Neuro-Optimal tracking control for continuous stirred tank reactor with input constraints," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 8, pp. 4516–4524, 2019.
- [14] H. Zhang, X. Cui, Y. Luo, and H. Jiang, "Finite-horizon H_∞ tracking control for unknown nonlinear systems with saturating actuators," *IEEE Transactions on Neural*

- Networks and Learning Systems*, vol. 29, no. 4, pp. 1200–1212, 2018.
- [15] Q. Wei, D. Liu, Q. Lin, and R. Song, “Adaptive dynamic programming for discrete-time zero-sum games,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 4, pp. 957–969, 2018.
 - [16] R. Song, F. L. Lewis, and Q. Wei, “Off-policy integral reinforcement learning method to solve nonlinear continuous-time multiplayer nonzero-sum games,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 3, pp. 704–713, 2017.
 - [17] Q. Wei and D. Liu, “Data-driven neuro-optimal temperature control of water-gas shift reaction using stable iterative adaptive dynamic programming,” *IEEE Transactions on Industrial Electronics*, vol. 61, no. 11, pp. 6399–6408, 2014.
 - [18] Q. Wei, D. Liu, F. L. Lewis, Y. Liu, and J. Zhang, “Mixed iterative adaptive dynamic programming for optimal battery energy control in smart residential microgrids,” *IEEE Transactions on Industrial Electronics*, vol. 64, no. 5, pp. 4110–4120, 2017.
 - [19] 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 Transactions on Industrial Electronics*, vol. 64, no. 6, pp. 4935–4944, 2017.
 - [20] X. Yang, H. He, and X. Zhong, “Adaptive dynamic programming for robust regulation and its application to power systems,” *IEEE Transactions on Industrial Electronics*, vol. 65, no. 7, pp. 5722–5732, 2018.
 - [21] B. Luo, H.-N. Wu, and T. Huang, “Optimal output regulation for model-free quanser helicopter with multistep q-learning,” *IEEE Transactions on Industrial Electronics*, vol. 65, no. 6, pp. 4953–4961, 2018.
 - [22] D. Wang and X. Zhong, “Advanced policy learning near-optimal regulation,” *IEEE/CAA Journal of Automatica Sinica*, vol. 6, no. 3, pp. 743–749, 2019.
 - [23] Q. Wei, D. Liu, Q. Lin, and R. Song, “Discrete-time optimal control via local policy iteration adaptive dynamic programming,” *IEEE Transactions on Cybernetics*, vol. 47, no. 10, pp. 3367–3379, 2017.
 - [24] Q. Wei, D. Liu, and H. Lin, “Value iteration adaptive dynamic programming for optimal control of discrete-time nonlinear systems,” *IEEE Transactions on Cybernetics*, vol. 46, no. 3, pp. 840–853, 2016.
 - [25] Q. Wei, F. L. Lewis, D. Liu, R. Song, and H. Lin, “Discrete-time local value iteration adaptive dynamic programming: convergence analysis,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 6, pp. 875–891, 2018.
 - [26] Q. Wei, D. Liu, and Q. Lin, “Discrete-time local value iteration adaptive dynamic programming: admissibility and termination analysis,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 11, pp. 2490–2502, 2017.
 - [27] B. Luo, D. Liu, T. Huang, X. Yang, and H. Ma, “Multi-step heuristic dynamic programming for optimal control of nonlinear discrete-time systems,” *Information Sciences*, vol. 411, pp. 66–83, 2017.
 - [28] 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 Transactions on Automation Science and Engineering*, vol. 9, no. 3, pp. 628–634, 2012.