Artif Intell Rev (2018) 50:75–91
https://doi.org/10.1007/s10462-017-9603-1

Iterative ADP learning algorithms for discrete-time multi-player games

He Jiang[1] · Huaguang Zhang[1]

Published online: 12 January 2018

Abstract Adaptive dynamic programming (ADP) is an important branch of reinforcement learning to solve various optimal control issues. Most practical nonlinear systems are controlled by more than one controller. Each controller is a player, and to make a tradeoff between cooperation and conflict of these players can be viewed as a game. Multi-player games are divided into two main categories: zero-sum game and non-zero-sum game. To obtain the optimal control policy for each player, one needs to solve Hamilton–Jacobi–Isaacs equations for zero-sum games and a set of coupled Hamilton–Jacobi equations for non-zero-sum games. Unfortunately, these equations are generally difficult or even impossible to be solved analytically. To overcome this bottleneck, two ADP methods, including a modified gradient-descent-based online algorithm and a novel iterative offline learning approach, are proposed in this paper. Furthermore, to implement the proposed methods, we employ single-network structure, which obviously reduces computation burden compared with traditional multiple-network architecture. Simulation results demonstrate the effectiveness of our schemes.

Keywords Adaptive dynamic programming · Approximate dynamic programming · Reinforcement learning · Neural network

1 Introduction

In the past few decades, dynamic programming (DP) was a classical method in solving optimal control problems. Unfortunately, due to the backward-in-time process, DP usually suffers from the curse of dimensionality, which brings enormous quantity of computation and

✉ Huaguang Zhang
hgzhang@ieee.org

He Jiang
jianghescholar@163.com

[1] College of Information Science and Engineering, Northeastern University, Shenyang 110819, Liaoning, People's Republic of China

limits its application. Inspired by the idea of reinforcement learning (RL) (Liu et al. 2015; Zhao and Zhu 2015; Wei et al. 2015), adaptive dynamic programming (ADP) (Wang et al. 2009, 2017; Wei et al. 2014) proposed by Werbos (1977) has become an effective tool to handle various nonlinear optimal control issues, such as optimal tracking control (Zhu et al. 2016; Zhang et al. 2017; Yang et al. 2016), constrained optimal control (Yang et al. 2016; Luo et al. 2015; Zhu et al. 2017), robust optimal control (Wang et al. 2014, 2016a, b) and model-free optimal control (Luo et al. 2016; Song et al. 2015; Zhao et al. 2015). Different from DP, ADP is a forward-in-time approach and obtains the optimal action through the responses from the environment.

ADP can be classified into two main methods: online learning algorithm (Vamvoudakis and Lewis 2010; Zhu and Zhao 2017) and offline learning algorithm (Luo et al. 2015; Al-Tamimi et al. 2008). For the optimal control problems of systems with single input, one needs to solve Hamilton–Jacobi–Bellman (HJB) equations to obtain the optimal solutions. For the linear cases, HJB equations are reduced into well-known Riccati equations, which can be computed directly. However, for the nonlinear cases, the HJB equations become nonlinear partial differential equations, which are difficult or even impossible to be solved analytically due to the nonlinear nature. To overcome this difficulty, several significant online and offline ADP schemes have been reported. There are two mainstream iterative offline learning methods including policy iteration (PI) (Murray et al. 2002; Liu and Wei 2014) and value iteration (VI) (Wei et al. 2016a, b). The VI algorithm was first proposed for nonlinear discrete-time (DT) systems in Al-Tamimi et al. (2008). Afterwards, for the DT version, the PI algorithm along with its convergence proof was presented in Liu and Wei (2014), where how to obtain an initial admissible control policy was also introduced. For continuous-time (CT) optimal control problems, the PI algorithm was designed in Murray et al. (2002), where the requirement of the knowledge of internal system dynamics was relaxed. By utilizing the persistence of excitation (PE) condition, an online actor-critic learning algorithm was proposed in Vamvoudakis and Lewis (2010) instead of offline iteration procedures. In order to solve the optimal control problems without any system dynamic knowledge, neural network (NN) identification technique was employed to reconstruct system dynamics in Liu et al. (2012), where, unfortunately, the identification errors were not considered. By using sampled system data, a direct data-driven learning algorithm, called off-policy, was developed in Luo et al. (2014) instead of prior identification procedure. The aforementioned works all considered the systems with single controller. However, practical systems generally have multiple inputs. The optimal control issues with multiple inputs can be viewed as multi-player games including multi-player non-zero-sum games and two-player zero-sum games. The multi-player games are similar but more complex than the general optimal control problems due to the existence of extra input terms.

Nowadays, the practical systems, such as communication networks and power systems, are generally controlled by more than one controller. Each controller is a player, and the optimal control for the systems with multiple controllers can be viewed as non-zero-sum games for multi-player. The nonlinear multi-player non-zero-sum games rely on solving a set of coupled Hamilton–Jacobi (HJ) equations. However, this is nearly an impossible task before the existence of ADP. Recently, some associated novel schemes have been provided for CT systems. In Vamvoudakis and Lewis (2011), an actor-critic adaptive learning algorithm was proposed to solve the coupled HJ equations in real time. Afterwards, a concurrent learning algorithm was presented in Kamalapurkar et al. (2014), and a novel critic-only method was given in Zhang et al. (2013), where only the critic network was required instead of actor-critic structure. Based on the theoretical framework of Vamvoudakis and

Lewis (2011), Kamalapurkar et al. (2014) and Zhang et al. (2013), NN-based identification technique was combined with ADP methods to solve the CT non-zero-sum games with unknown dynamics in Liu et al. (2014), Johnson et al. (2015), Zhao et al. (2016) and Jiang et al. (2017). Since the identification schemes (Liu et al. 2014; Johnson et al. 2015; Zhao et al. 2016; Jiang et al. 2017) may cause approximation errors, a direct data-driven off-policy learning approach was proposed to address this problem in Song et al. (2017). However, there are still few papers concerning the DT version. Only in Zhang et al. (2016), dual network structure ADP methods were employed to solve the DT non-zero-sum games. Unfortunately, compared with single network structures, the dual network structures may bring extra computation burden. In this paper, a single network control scheme is presented and the corresponding stability analysis is also derived for the first time.

H_∞ control problems can be converted into zero-sum games, optimal solutions of which are the saddle point equilibriums. The key to obtaining the solutions of zero-sum games is to solve the associated Hamilton–Jacobi–Isaacs (HJI) equations (Jiang et al. 2017). However, the HJI equations are difficult to be solved analytically. To address this problem, some ADP based methods have been proposed. In Al-Tamimi et al. (2007a), adaptive critic designs were presented for DT zero-sum games with the application to H_∞ control of F-16 aircraft autopilot design. In Al-Tamimi et al. (2007b), Q-learning technique was utilized to handle the model-free zero-sum issues. Unfortunately, both previous works (Al-Tamimi et al. 2007a, b) were limited to linear systems. With the development of ADP, zero-sum game theoretic formulation of nonlinear systems was provided in Mehraeen et al. (2013), where the requirement of internal dynamics was relaxed by constructing an extra identifier NN. In Liu et al. (2013), three networks, i.e., critic, actor and disturbance networks were employed to solve the nonlinear zero-sum games. Combined with the identification technique, a novel online learning method was proposed for nonlinear zero-sum games with unknown dynamics in Zhang et al. (2014), where the identifier, critic, actor and disturbance NNs were all used. In the recent research work (Wei et al. 2017), the convergence proof of the iterative ADP algorithm for DT zero-sum games was derived. However, three network architecture is still required to implement this iterative algorithm. It is known that utilizing multiple networks results in extra computation burden and high difficulty in the algorithm design. Thus, it is encouraged to reduce the number of NNs. To the best of our knowledge, there are still few studies to investigate the DT zero-sum games by using single network structure, which motivates our research of this paper.

This paper is organized as follows. In Sect. 2, the DT non-zero-sum games are formulated and a novel online learning algorithm is modified based on the gradient descent method. An optimality test and an effectiveness test are both given in the simulation results. In Sect. 3, the DT zero-sum games are formulated and an iterative learning algorithm is implemented via a single network architecture. A linear F-16 aircraft example and a nonlinear Van der Pol's oscillator system example are shown in the simulation results. Finally, a brief conclusion is drawn in Sect. 4.

2 Online learning algorithm for DT multi-player non-zero-sum games

In this section, DT non-zero-sum games are formulated, and a novel online learning algorithm along with the associated NN implementation and stability analysis is presented.

2.1 Problem formulation

Let us consider the following N-player system:

$$x(k+1) = f(x(k)) + \sum_{j=1}^{N} g_j(x(k))u_j(k) \tag{1}$$

where $x \in \mathbb{R}^n$ denotes the state and $u_j \in \mathbb{R}^{m_j}$ with $j = 1, 2, \ldots, N$ represents the player or controller. $f(x) \in \mathbb{R}^n$ and $g_j(x) \in \mathbb{R}^{n\times m_j}$ are the system functions. Assume that $g_j(x)$ is bounded on a compact set, i.e., $\|g_j(x)\| \leq g_{jm}$.

The performance index for each player is given by

$$J_i(x(0), u_i, u_{(-i)}) = \sum_{t=0}^{\infty} s_i(x(t), u_i(t), u_{(-i)}(t)) \tag{2}$$

where $s_i(x, u_i, u_{(-i)}) = x^T Q_i x + u_i^T R_{ii} u_i + \sum\limits_{j=1, j\neq i}^{N} u_j^T R_{ij} u_j$ with the positive definite symmetric matrices $Q_i > 0$, $R_{ii} > 0$ and $R_{ij} > 0$, and $u_{(-i)} = \{u_j : j = 1, 2, \ldots, N, j \neq i\}$. According to the previous investigations (Liu et al. 2012; Sokolov et al. 2015), it is generally required that $s_i(x, u_i, u_{(-i)})$ should be a bounded positive semidefinite function, i.e., $\|s_i\| \leq s_{im}$. That is, u_i and $u_{(-i)}$ should be admissible control policies.

With a set of admissible control policies $\{u_i, u_{(-i)}\}$, the value function can be described by

$$V_i(x(k)) = \sum_{t=k}^{\infty} s_i(x(t), u_i(t), u_{(-i)}(t)) = s_i(x(k), u_i(k), u_{(-i)}(k)) + V_i(x(k+1)). \tag{3}$$

The optimal value function is defined as

$$V_i^*(x(k)) = \min_{u_i} \sum_{t=k}^{\infty} s_i(x(t), u_i(t), u_{(-i)}(t)). \tag{4}$$

According to the stationarity condition (Zhang et al. 2013, 2016), the optimal control policy $u_i^*(x)$, $\forall i$ is derived as

$$u_i^*(k) = -\frac{1}{2} R_{ii}^{-1} g_i^T(x(k)) \nabla V_i^*(x(k+1)) \tag{5}$$

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

A set of control strategies $\{u_i^*, u_{(-i)}^*\}$ is able to form a Nash equilibrium for an N-player non-zero-sum game, if the inequality holds as

$$V_i^* \triangleq V_i(u_i^*, u_{(-i)}^*) \leq V_i(u_i, u_{(-i)}^*), \quad \forall i. \tag{6}$$

If an N-player non-zero-sum game exists, then $V_i^*(x)$ satisfies the following DT coupled HJ equation:

$$V_i^*(x(k)) = s_i(x(k), u_i^*(k), u_{(-i)}^*(k)) + V_i^*(x(k+1)). \tag{7}$$

2.2 NN implementation of the single network algorithm

Since NN is a universal approximator, the value function can be expressed as

$$V_i^*(x(k)) = W_i^T \phi_i(x(k)) + \varepsilon_i(x(k)) \tag{8}$$

where $\phi_i(\cdot) \in \mathbb{R}^{L_i}$ is the activation function vector; $W_i \in \mathbb{R}^{L_i}$ denotes the ideal NN weight with $\|W_i\| \leq W_{im}$, where W_{im} is a positive constant; ε_i is the NN approximation error with $\lim\limits_{L_i \to \infty} \varepsilon_i = 0$ on a compact set. Subsequently, the critic NN is constructed by

$$\hat{V}_i(x(k)) = \hat{W}_i^T(k)\phi_i(x(k)) \tag{9}$$

where $\hat{W}_i(k)$ is the estimation of W_i. The control policy $u_i(k)$ is approximated by

$$\hat{u}_i(k) = -\frac{1}{2} R_{ii}^{-1} g_i^T(x(k)) \nabla \phi_i^T(x(k+1)) \hat{W}_i(k). \tag{10}$$

According to (3), one has

$$s_i(x(k-1), u_i(k-1), u_{(-i)}(k-1)) + V_i(x(k)) - V_i(x(k-1)) = 0. \tag{11}$$

Using the NNs to approximate the value functions and control policies yields the following NN approximation residual error

$$e_i(k) = s_i(k-1) + \hat{W}_i^T(k) \Delta\phi_i(x(k)) \tag{12}$$

where $\Delta\phi_i(x(k)) = \phi_i(x(k)) - \phi_i(x(k-1))$. To minimize the error performance $E_i(k) = \frac{1}{2} e_i^T(k) e_i(k)$, the gradient descent method is frequently utilized as below

$$\begin{aligned} \hat{W}_i(k+1) &= \hat{W}_i(k) - \alpha_i \frac{\partial E_i(k)}{\partial e_i(k)} \frac{\partial e_i(k)}{\partial \hat{W}_i(k)} \\ &= \hat{W}_i(k) - \alpha_i \phi_i(x(k))[s_i(k-1) + \hat{W}_i^T(k)\Delta\phi_i(x(k))]^T \end{aligned} \tag{13}$$

where $0 < \alpha_i < 1$ denotes the NN learning rate.

However, the gradient descent method (13) has some disadvantages. First, it generally needs a slow convergence process to obtain the optimal solutions. Second, the stability analysis of the standard gradient descent algorithm is hard to provide. To overcome these deficiencies, the modified updating law is given by

$$\hat{W}_i(k+1) = \hat{W}_i(k) - \alpha_i \{\phi_i(k)[s_i(k-1) + \hat{W}_i^T(k)\Delta\phi_i(k)]^T + F\hat{W}_i(k)\} \tag{14}$$

where F is a constant parameter to be designed. The schematic diagram of our proposed scheme is shown in Fig. 1.

Remark 1 In this paper, we choose hyperbolic tangent to be the type of the activation function, because the hyperbolic tangent function is naturally bounded, which implies ϕ_i, $\Delta\phi_i$ and $\nabla\phi_i$ are all bounded, i.e., $\|\phi_i\| \leq \phi_{im}$, $\|\Delta\phi_i\| \leq \Delta\phi_{im}$ and $\|\nabla\phi_i\| \leq \nabla\phi_{im}$.

2.3 Stability analysis

Define the NN weight estimation error as $\tilde{W}_i(k) = \hat{W}_i(k) - W_i$. Based on (14), one attains

$$\begin{aligned} \tilde{W}_i(k+1) &= \tilde{W}_i(k) - \alpha_i \{\phi_i(k)[s_i(k-1) + \hat{W}_i^T(k)\Delta\phi_i(k)]^T + F\hat{W}_i(k)\} \\ &= (1 - \alpha_i F - \alpha_i \phi_i(k)\Delta\phi_i^T(k))\tilde{W}_i(k) - (\alpha_i\phi_i(k)\Delta\phi_i^T(k) + \alpha_i F)W_i \\ &\quad - \alpha_i \phi_i(k) s_i^T(k-1). \end{aligned} \tag{15}$$

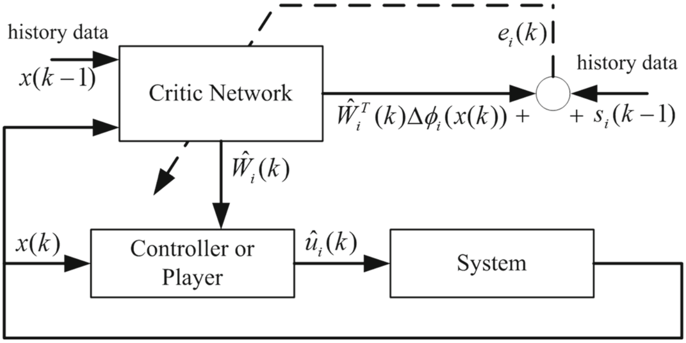

Fig. 1 Schematic diagram of the single network scheme

Theorem 1 *If the NN learning rate α_i is selected small enough, one can set the parameter F such that the NN weight estimation error $\tilde{W}_i(k)$ is uniformly ultimately bounded (UUB).*

Proof Construct the Lyapunov function candidate as

$$L_i(k) = \tilde{W}_i^T(k)\tilde{W}_i(k) \tag{16}$$

which implies the first difference $\Delta L_i(k) \triangleq L_i(k+1) - L_i(k) = \tilde{W}_i^T(k+1)\tilde{W}_i(k+1) - \tilde{W}_i^T(k)\tilde{W}_i(k)$. Let $C_i = (1-\alpha_i F - \alpha_i\phi_i(k)\Delta\phi_i^T(k))$ and $D_i = (\alpha_i\phi_i(k)\Delta\phi_i^T(k)+\alpha_i F)W_i + \alpha_i\phi_i(k)s_i^T(k-1)$ for simplicity. In light of (15), the first difference of (16) can be further expressed as

$$\Delta L_i(k) = \tilde{W}_i^T(k)C_i^2\tilde{W}_i(k) - \tilde{W}_i^T(k)\tilde{W}_i(k) - 2\tilde{W}_i^T(k)C_iD_i + \|D_i\|^2. \tag{17}$$

After a simple arrangement, one gets

$$\begin{aligned}
&\tilde{W}_i^T(k)C_i^2\tilde{W}_i(k) - \tilde{W}_i^T(k)\tilde{W}_i(k)\\
&\quad = -(\alpha_i F + \alpha_i\phi_i(k)\Delta\phi_i^T(k))(2 - \alpha_i F - \alpha_i\phi_i(k)\Delta\phi_i^T(k))\left\|\tilde{W}_i(k)\right\|^2, \qquad (18)\\
&-2\tilde{W}_i^T(k)C_iD_i \le 2\|C_i\|\,\|D_i\|\left\|\tilde{W}_i(k)\right\|\\
&\quad \le 2(\alpha_i F + \alpha_i\phi_{im}\Delta\phi_{im} + 1)(\alpha_i\phi_{im}\Delta\phi_{im}W_{im} + \alpha_i F W_{im} + \alpha_i\phi_{im}s_{im})\left\|\tilde{W}_i(k)\right\|, \qquad (19)
\end{aligned}$$

and

$$\|D_i\|^2 \le (\alpha_i\phi_{im}\Delta\phi_{im}W_{im} + \alpha_i F W_{im} + \alpha_i\phi_{im}s_{im})^2. \tag{20}$$

Let $P_i = (\alpha_i F + \alpha_i\phi_i(k)\Delta\phi_i^T(k))(2 - \alpha_i F - \alpha_i\phi_i(k)\Delta\phi_i^T(k))$, $d_i = 2(\alpha_i F + \alpha_i\phi_{im}\Delta\phi_{im} + 1)(\alpha_i\phi_{im}\Delta\phi_{im}W_{im} + \alpha_i F W_{im} + \alpha_i\phi_{im}s_{im})$ and $c_i = (\alpha_i\phi_{im}\Delta\phi_{im}W_{im} + \alpha_i F W_{im} + \alpha_i\phi_{im}s_{im})^2$. Note that since $\phi_i(k)$ and $\Delta\phi_i(k)$ are both bounded, if the NN learning rate α_i is selected small enough, one can easily choose a positive constant F to guarantee P_i to be positive definite. Then, combining (18), (19) and (20), it can be acquired that

$$\Delta L_i(k) \le -\sigma_{\min}(P_i)\left\|\tilde{W}_i(k)\right\|^2 + d_i\left\|\tilde{W}_i(k)\right\| + c_i \tag{21}$$

where $\sigma_{\min}(P_i)$ represents the minimum value of P_i on the compact set.

Therefore, $\Delta L_i(k) \leq 0$ if

$$\left\|\tilde{W}_i(k)\right\| \geq \frac{d_i}{2\sigma_{\min}(P_i)} + \sqrt{\frac{d_i^2}{4\sigma_{\min}^2(P_i)} + \frac{c_i}{\sigma_{\min}(P_i)}} \triangleq b_i. \tag{22}$$

That is, $\left\|\tilde{W}_i(k)\right\|$ is bounded by b_i. According to the standard Lyapunov extension theorem (Vamvoudakis and Lewis 2011; Zhao et al. 2016), the aforementioned derivation demonstrates that the NN weight estimation error $\tilde{W}_i(k)$ is UUB. The proof is completed. □

Corollary 1 *The error between the obtained control $\hat{u}_i$ and the optimal control u_i^* is bounded, i.e., $\hat{u}_i$ is close to u_i^* within a small approximation error.*

Proof By means of Theorem 1, one attains

$$\begin{aligned} \left\|\hat{u}_i(k) - u_i^*(k)\right\| &= \left\|-\frac{1}{2}R_{ii}^{-1}g_i^T(x(k))\nabla\phi_i^T(x(k+1))\tilde{W}_i(k)\right\| \\ &\leq \frac{1}{2}\left\|R_{ii}^{-1}\right\| g_{im}\nabla\phi_{im}b_i \triangleq B_i \end{aligned} \tag{23}$$

which implies the error between $\hat{u}_i$ and u_i^* is bounded by B_i. This completes the proof. □

2.4 Simulation results

Two simulation examples will be provided to test the optimality and effectiveness of our proposed scheme, respectively.

2.4.1 Optimality test

In order to test the optimality, let $N = 1$, and then the non-zero-sum game issue can be converted to the general optimal control problem. Consider the following linear system:

$$x(k+1) = Ax(k) + Bu(k) \tag{24}$$

with $A = [3, 0; 0, 2]$ and $B = [1; 1]$ and the quadratic function $s(x, u) = p_1 * x_1^2 + p_2 * x_2^2 + p_3 * u^2$, where p_1, p_2 and p_3 are the given constant parameters. The optimal solution can be obtained by solving the well-known Riccati equations, which will result in the optimal control policy $u(k) = Kx(k)$, where K represents the optimal control gain. By using our proposed learning scheme, the approximate optimal NN weight $\hat{W}$ is attained, and then inserting $\hat{W}$ into (10) yields the approximate optimal control policy $\hat{u}(k) = \hat{K}x(k)$, where $\hat{K}$ denotes the approximate optimal control gain. The following Table 1 shows that the obtained approximate optimal control gains are close to the optimal ones, which illustrates the optimality of our scheme.

2.4.2 Effectiveness test

Consider the following nonlinear system:

$$x(k+1) = \begin{bmatrix} -\sin(0.5x_2(k)) \\ -\cos(1.4x_2(k))\sin(0.9x_1(k)) + 2u_1(k) + u_2(k) \end{bmatrix}. \tag{25}$$

The positive definite functions are given by $s_1(x, u_1, u_2) = s_2(x, u_1, u_2) = x_1^2 + x_2^2 + u_1^2 + u_2^2$. Set the NN learning rate $\alpha_i = 0.01$, $\forall i$ and the constant parameter $F = 10$. After using

Table 1 Optimality test

Example number	No. 1	No. 2	No. 3
Parameters	$\begin{cases} p_1 = 1 \\ p_2 = 1 \\ p_3 = 1 \end{cases}$	$\begin{cases} p_1 = 2 \\ p_2 = 2 \\ p_3 = 1 \end{cases}$	$\begin{cases} p_1 = 2 \\ p_2 = 2 \\ p_3 = 3 \end{cases}$
Optimal control gain K	$[-7.0227, 2.7208]$	$[-7.2037, 2.8330]$	$[-6.9341, 2.6659]$
Approximate control gain $\hat{K}$	$[-7.0381, 2.6978]$	$[-7.1996, 2.8117]$	$[-6.8774, 2.6013]$
Approximate error $\left\Vert \hat{K} - K \right\Vert$	0.0277	0.0217	0.0860

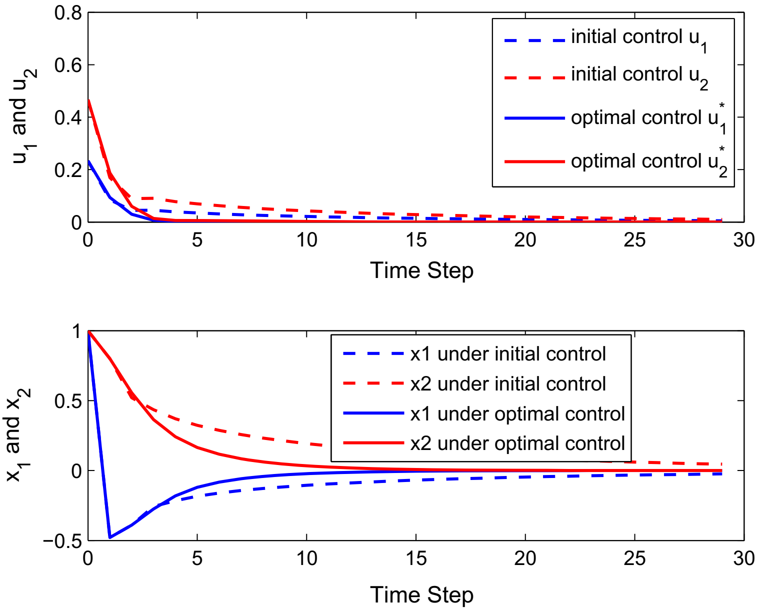

Fig. 2 Trajectories of control inputs and system states

the tuning law (14), we obtain the finally converged NN weights and the simulation results are shown in Fig. 2. Compared with the initial control policies, the obtained optimal control policies use less energy and make the system states converge faster, which demonstrates the nice control performance of our proposed scheme. In Fig. 3, the 3D plot of state trajectories with different initial values is provided.

Remark 2 It is known that online learning algorithms can learn the optimal solutions by using the information generated in real time. This is the main merit of the online schemes. However, compared with the offline learning algorithms, the online schemes have some drawbacks: (1) The "exploration", also called the PE condition, is always required in online learning methods. Unfortunately, how to find out the suitable "exploration" is still an open problem and rarely discussed in the existing works; (2) Without suitable enough initial values, the online learning method may be time-consuming, which limits its applications in real time control; (3) Online schemes just use current data and discard the past. This means the measurable data is utilized only once, which causes low efficiency in data utilization. Thus, in the following section,

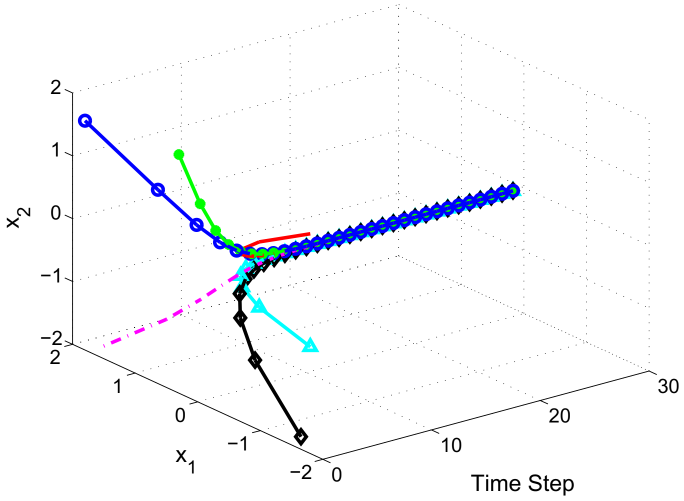

Fig. 3 3D plot of state trajectories with different initial values

an iterative offline learning method will be presented for DT zero-sum games, which can be also applied to non-zero-sum games.

3 Offline learning algorithm for DT two-player zero-sum games

In this section, DT two-player zero-sum games are formulated, and an iterative offline learning algorithm along with NN implementation is provided.

3.1 Problem formulation

Consider the following DT nonlinear system:

$$x(k+1) = f(x(k)) + g(x(k))u(k) + d(x(k))w(k) \tag{26}$$

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

Define the performance index function as

$$J(x(0), u, w) = \sum_{t=0}^{\infty} r(x(t), u(t), w(t)) \tag{27}$$

where $r(x, u, w) = x^T P x + u^T R u - \gamma^2 w^T w$ with the positive definite symmetric matrices P, R and the prescribed value γ denotes an upper bound on the desired L_2 gain disturbance attenuation.

Given admissible control policy $u(x)$ and disturbance policy $w(x)$, the value function is expressed as

$$\begin{aligned} V(x(k)) &= \sum_{t=k}^{\infty} r(x(t), u(t), w(t)) \\ &= r(x(k), u(k), w(k)) + V(x(k+1)). \end{aligned} \tag{28}$$

If a zero-sum game exists, then there must exist a saddle point solution (u^*, w^*) such that

$$V(x(k), u^*, w) \leq V(x(k), u^*, w^*) \leq V(x(k), u, w^*). \tag{29}$$

Let $V^*(x(k)) \triangleq V(x(k), u^*, w^*)$, and $V^*(x(k))$ denotes the optimal value function. If $V^*(x(k))$ exists, then we also have

$$\begin{aligned} V^*(x(k)) &= \min_u \max_w \{r(x(k), u(k), w(k)) + V^*(x(k+1))\} \\ &= \max_w \min_u \{r(x(k), u(k), w(k)) + V^*(x(k+1))\} \end{aligned} \tag{30}$$

which, according to the stationarity condition, implies the forms of the associated optimal policies:

$$u^*(k) = -\frac{1}{2} R^{-1} g^T(x(k)) \frac{\partial V^*(x(k+1))}{\partial x(k+1)}, \tag{31}$$

$$w^*(k) = \frac{1}{2\gamma^2} d^T(x(k)) \frac{\partial V^*(x(k+1))}{\partial x(k+1)}. \tag{32}$$

Substituting u^* and w^* into the value function yields the DT HJI equation:

$$V^*(x(k)) = r(x(k), u^*(k), w^*(k)) + V^*(x(k+1)). \tag{33}$$

However, it is generally difficult or even impossible to obtain the analytical solutions of HJI equations.

Inspired by the previous works (Mehraeen et al. 2013; Liu et al. 2013; Wang et al. 2017a, b) and simultaneous policy update algorithms in Luo et al. (2015) and Jiang et al. (2017), we propose the following iterative learning method for the DT zero-sum games.

```
Algorithm 1 iterative learning method for zero-sum games
Step 1: (Initialization)
Let the iteration index $i = 0$; Select a small enough computation precision $\epsilon$ and
an initial value function $V^{(0)}(x)$ to produce an admissible control policy
$u^{(0)}(k) = -\frac{1}{2}R^{-1}g^T(x(k))\frac{\partial V^{(0)}(x(k+1))}{\partial x(k+1)}$ and a disturbance
policy $w^{(0)}(k) = \frac{1}{2\gamma^2}d^T(x(k))\frac{\partial V^{(0)}(x(k+1))}{\partial x(k+1)}$.
Step 2: (Policy Evaluation)
With $u^{(i)}(x)$ and $w^{(i)}(x)$, compute $V^{(i+1)}(x)$ by
    $V^{(i+1)}(x(k)) = r(x(k), u^{(i)}(k), w^{(i)}(k)) + V^{(i+1)}(x(k+1))$.
Step 3: (Policy Improvement)
Given $V^{(i+1)}(x)$, update $u^{(i+1)}(x)$ and $w^{(i+1)}(x)$ by
    $u^{(i+1)}(k) = -\frac{1}{2}R^{-1}g^T(x(k))\frac{\partial V^{(i+1)}(x(k+1))}{\partial x(k+1)}$,
    $w^{(i+1)}(k) = \frac{1}{2\gamma^2}d^T(x(k))\frac{\partial V^{(i+1)}(x(k+1))}{\partial x(k+1)}$.
Step 4: If $\left\|V^{(i+1)} - V^{(i)}\right\| \leq \epsilon$, stop and the approximate optimal values, i.e.,
$V^{(i+1)}$, $u^{(i+1)}$ and $w^{(i+1)}$ are acquired; Else, let $i = i + 1$ and go back to Step 2.
```

3.2 NN implementation of Algorithm 1

Since NN is a universal approximator, the iterative value function has the following NN representation:

$$V^{(i)}(x) = \theta^T(x)W^{(i)} + \varepsilon^{(i)}(x) \tag{34}$$

where $\theta(x) \in \mathbb{R}^L$ denotes the NN activation function vector; $W^{(i)} \in \mathbb{R}^L$ is the ideal NN weight; $\varepsilon^{(i)}(x)$ represents the NN approximation error with $\lim\limits_{L\to\infty} \varepsilon^{(i)} = 0$ on a compact set. In light of (34), the critic NN is constructed by

$$\hat{V}^{(i)}(x) = \theta^T(x)\hat{W}^{(i)} \tag{35}$$

where $\hat{V}^{(i)}$ and $\hat{W}^{(i)}$ are the estimations of $V^{(i)}$ and $W^{(i)}$, respectively.

By means of Algorithm 1, the estimations of $u^{(i)}$ and $w^{(i)}$, i.e., $\hat{u}^{(i)}$ and $\hat{w}^{(i)}$ are described by

$$\hat{u}^{(i)}(k) = -\frac{1}{2}R^{-1}g^T(x(k))\frac{\partial \hat{V}^{(i)}(x(k+1))}{\partial x(k+1)}, \tag{36}$$

$$\hat{w}^{(i)}(k) = \frac{1}{2\gamma^2}d^T(x(k))\frac{\partial \hat{V}^{(i)}(x(k+1))}{\partial x(k+1)}. \tag{37}$$

According to Algorithm 1, employing critic NN $\hat{V}^{(i)}(x)$ to replace $V^{(i)}(x)$ will yield a NN approximation residual error:

$$\begin{aligned} e^{(i)}(x(k)) &= \hat{V}^{(i+1)}(x(k)) - \hat{V}^{(i+1)}(x(k+1)) - r(x(k), \hat{u}^{(i)}(k), \hat{w}^{(i)}(k)) \\ &= (\theta^T(x(k)) - \theta^T(x(k+1)))\hat{W}^{(i+1)} - r(x(k), \hat{u}^{(i)}(k), \hat{w}^{(i)}(k)) \end{aligned} \tag{38}$$

where $\hat{W}^{(i+1)}$ is an unknown term to be computed at the ith iteration step. For simplicity, let $\delta^{(i)} = \theta^T(x(k)) - \theta^T(x(k+1))$ and $\rho^{(i)} = r(x(k), \hat{u}^{(i)}(k), \hat{w}^{(i)}(k))$. Equation (38) can be rewritten as

$$e^{(i)} = \delta^{(i)}\hat{W}^{(i+1)} - \rho^{(i)}. \tag{39}$$

To compute $\hat{W}^{(i+1)}$ while minimizing the error performance $\left\|e^{(i)}\right\|^2$, we employ the least-square approach, which requires large amounts of measurable system data. Given different data sets, one can construct the database as $\eta^{(i)} = \left[\delta_{[1]}^{(i)T}, \delta_{[2]}^{(i)T}, \ldots, \delta_{[M]}^{(i)T}\right]^T$ and $\zeta^{(i)} = \left[\rho_{[1]}^{(i)T}, \rho_{[2]}^{(i)T}, \ldots, \rho_{[M]}^{(i)T}\right]^T$. Therefore, the solution of $\hat{W}^{(i+1)}$ in the least-square form is

$$\hat{W}^{(i+1)} = (\eta^{(i)T}\eta^{(i)})^{-1}\eta^{(i)T}\zeta^{(i)}. \tag{40}$$

As a result, the aforementioned derivation (35–40) can be summarized as the following Algorithm 2.

```
Algorithm 2 NN-based iterative learning method
Step 1: (Initialization)
    Let the iteration index $i = 0$; Choose a small enough computation precision $\epsilon$;
    Select initial critic NN weights $\hat{W}^{(0)}$ to produce an admissible control policy $\hat{u}^{(0)}$
    and a disturbance policy $\hat{w}^{(0)}$; Collect $M$ system sampling data sets.
Step 2: (Policy Evaluation)
    Calculate $\eta^{(i)}$ and $\zeta^{(i)}$, and then compute critic NN weights $\hat{W}^{(i+1)}$ via (40).
Step 3: (Policy Improvement)
    With $\hat{W}^{(i+1)}$, update the control policy $\hat{u}^{(i+1)}$ and the disturbance
    policy $\hat{w}^{(i+1)}$ through (36) and (37), respectively.
Step 4: If $\left\|\hat{W}^{(i+1)} - \hat{W}^{(i)}\right\| \leq \epsilon$, stop and the approximate optimal
    NN weights $\hat{W}^{(i)}$ are acquired; Else, let $i = i + 1$ and go back to Step 2.
```

Remark 3 For the DT zero-sum games, previous related works generally use traditional actor-disturbance-critic structure to deal with the issues, that is, actor, disturbance and critic NNs are all required to approximate the control policy, disturbance policy and value function, respectively. In addition, one also needs to design a specific NN updating law for each network. If we employ the proposed single-network approach in this paper, only the critic network is required and we just need to design the updating law for the critic NN, which can reduce the computation burden by two-thirds. For the non-zero-sum games, if we utilize previous dual network methods to handle a case with six players, we need six critic networks and six actor networks, i.e., twelve networks in total. By using the critic-only structure proposed in this paper, only six critic networks are enough, which significantly reduces the computation burden by half.

3.3 Simulation results

In this subsection, a linear example and a nonlinear one are given to test the optimality and effectiveness of our proposed approach, respectively.

3.3.1 Linear example

Consider the F-16 aircraft short period dynamics (Al-Tamimi et al. 2007b):

$$x(k+1) = Ax(k) + Bu(k) + Cw(k) \tag{41}$$

where $A = \begin{bmatrix} 0.906488 & 0.0816012 & -0.0005 \\ 0.0741349 & 0.90121 & -0.000708383 \\ 0 & 0 & 0.132655 \end{bmatrix}$, $B = \begin{bmatrix} -0.00150808 \\ -0.0096 \\ 0.867345 \end{bmatrix}$ and $C = \begin{bmatrix} 0.00951892 \\ 0.00038373 \\ 0 \end{bmatrix}$. Let the parameters $P = [1, 0; 0, 1]$, $R = 1$ and $\gamma = 1$. According to Al-Tamimi et al. (2007b), the optimal solutions are $u^*(k) = Kx(k)$ and $w^*(k) = Lx(k)$, where $K = [0.0733,\ 0.0872,\ -0.0661]$ and $L = [0.1476,\ 0.1244,\ 0]$. By using the iterative learning method Algorithm 2, the NN weight $\hat{W}^{(i+1)}$ can be obtained via (40). Substituting $\hat{W}^{(i+1)}$ into (36) and (37) yields the approximate optimal polices $\hat{u}^{(i+1)} = \hat{K}x(k)$ and $\hat{w}^{(i+1)} = \hat{L}x(k)$, where $\hat{K}$ and $\hat{L}$ represent the estimations of K and L, respectively. From Figs. 4 and 5, it can be observed that the NN weights finally converge to the optimal values after a sufficient learning procedure. Figure 6 shows the final convergence of system dynamics.

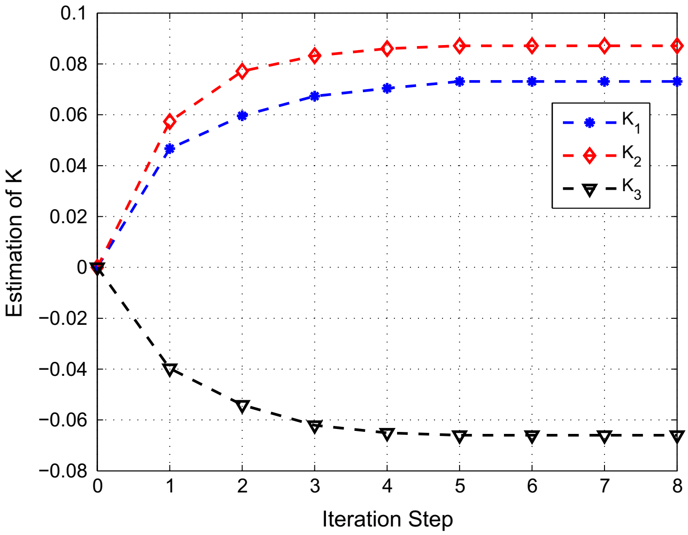

Fig. 4 Estimation of *K*

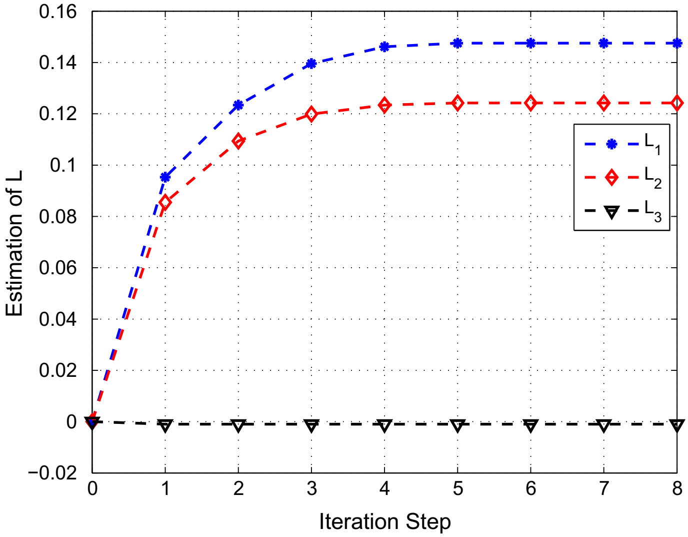

Fig. 5 Estimation of *L*

3.3.2 Nonlinear example

Let us consider the modified Van der Pol's oscillator system:

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix} = \begin{bmatrix} x_2 \\ (1-x_1^2)x_2 - x_1\end{bmatrix} + Bu + Cw \tag{42}$$

where $B = \begin{bmatrix} 3.5 & 0 \\ 0 & 3.5 \end{bmatrix}$ and $C = \begin{bmatrix} 4 & 0 \\ 0 & 3 \end{bmatrix}$.

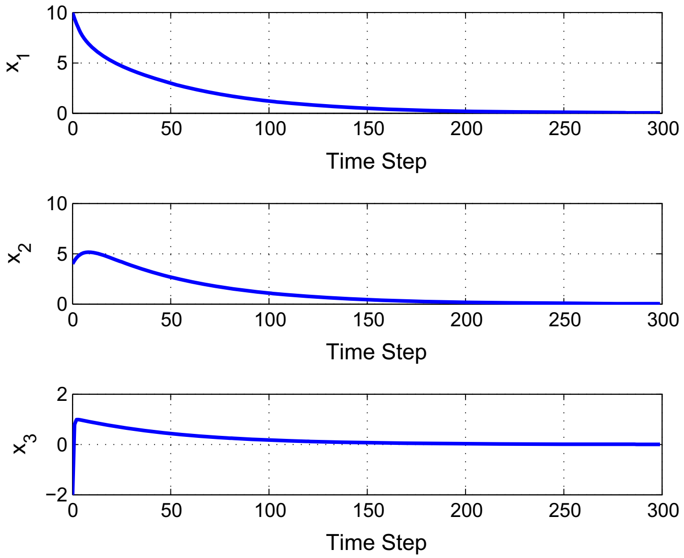

Fig. 6 Trajectories of system states

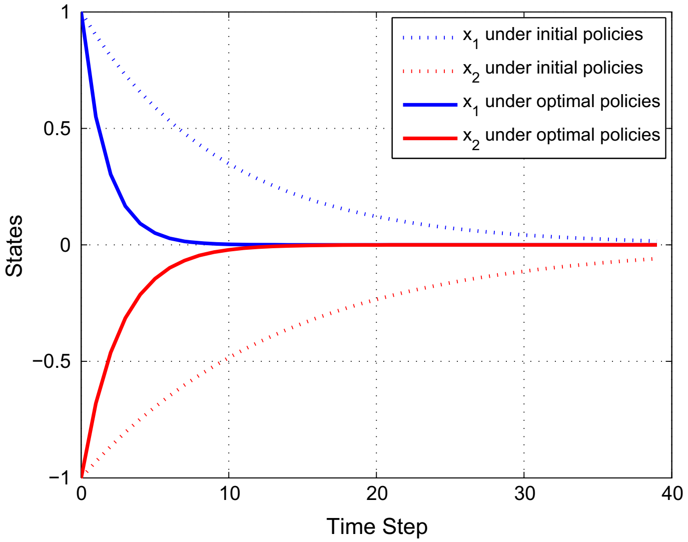

Fig. 7 Trajectories of states under initial policies and optimal policies

With the sampling interval $\Delta t = 0.1s$, discretizing the system (42) yields

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \begin{bmatrix} x_1(k) + \Delta t x_2(k) \\ -\Delta t x_1(k) + (1+\Delta t)x_2(k) - \Delta t x_1^2(k) x_2(k) \end{bmatrix} + \Delta t Bu(k) + \Delta t Cw(k). \tag{43}$$

The positive definite function in the performance index function (27) is selected as $r(x, u, w) = x_1^2 + x_2^2 + u_1^2 + u_2^2 - 5w_1^2 - 5w_2^2$. By employing Algorithm 2, the simu-

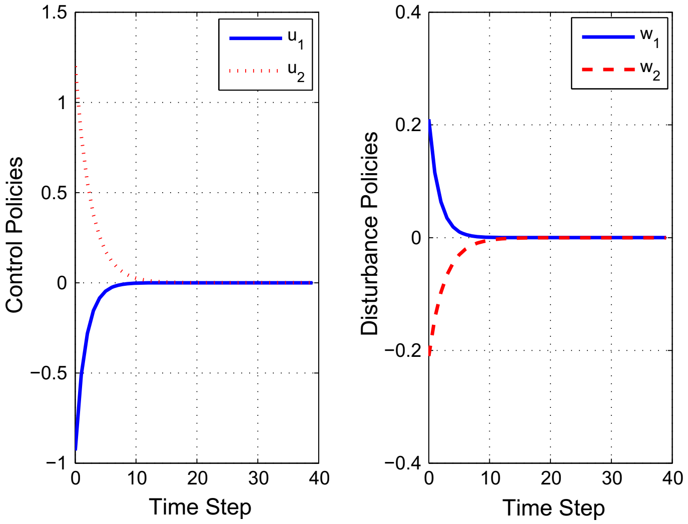

Fig. 8 Trajectories of $u(k)$ and $w(k)$

lation results are obtained in Fig. 7, which shows the states under obtained optimal policies get converged faster than the ones under initial policies. This indicates the control performance with optimality can be achieved through Algorithm 2. Figure 8 shows the trajectories of $u(k)$ and $w(k)$.

4 Conclusion

In this paper, we first review the state-of-the-art ADP works for both multi-player zero-sum game and non-zero-sum game. Then, we present a modified gradient-descent-based online algorithm for DT non-zero-sum games and a novel iterative offline learning approach for DT zero-sum games. The single network architecture is employed to implement the proposed two algorithms. This single network scheme significantly reduces the number of the used NNs and computation burden, and also simplifies the complexity of the algorithm design compared with previous multiple network architecture works. Since the proposed ADP methods have powerful learning ability and adaptivity, it is expected that they can be applied to other decision support systems.

Acknowledgements This work is supported by the National Natural Science Foundation of China (61433004, 61627809, 61621004), and IAPI Fundamental Research Funds 2013ZCX14.

References

Al-Tamimi A, Abu-Khalaf M, Lewis FL (2007) Adaptive critic designs for discrete-time zero-sum games with application to H_∞ control. IEEE Trans Syst Man Cybern B Cybern 37(1):240–247

Al-Tamimi A, Lewis FL, Abu-Khalaf M (2007) Model-free Q-learning designs for linear discrete-time zero-sum games with application to H_∞ control. Automatica 43(3):473–481

Al-Tamimi A, Lewis FL, Abu-Khalaf M (2008) Discrete-time nonlinear HJB solution using approximate dynamic programming: convergence proof. IEEE Trans Syst Man Cybern Part B Cybern 38(4):943–949

Jiang H, Zhang H, Xiao G, Cui X (2017) Data-based approximate optimal control for nonzero-sum games of multi-player systems using adaptive dynamic programming. Neurocomputing 1–8:12. https://doi.org/10.1016/j.neucom.2017.05.086

Jiang H, Zhang H, Luo Y, Cui X (2017) H_∞ control with constrained input for completely unknown nonlinear systems using data-driven reinforcement learning method. Neurocomputing 237:226–234

Johnson M, Kamalapurkar R, Bhasin S, Dixon WE (2015) Approximate N-player nonzero-sum game solution for an uncertain continuous nonlinear system. IEEE Trans Neural Netw Learn Syst 1(3):1645–1658

Kamalapurkar R, Klotz J, Dixon WE (2014) Concurrent learning-based online approximate feedback Nash equilibrium solution of N-player nonzero-sum differential games. IEEE/CAA J Autom Sin 1(3):239–247

Liu D, Wei Q (2014) Policy iteration adaptive dynamic programming algorithm for discrete-time nonlinear systems. IEEE Trans Neural Netw Learn Syst 25(3):621–634

Liu D, Wang D, Zhao D, Wei Q, Jin N (2012) Neural-network-based optimal control for a class of unknown discrete-time nonlinear systems using globalized dual heuristic programming. IEEE Trans Autom Sci Eng 9(3):628–634

Liu F, Sun J, Si J, Guo W, Mei S (2012) A boundedness result for the direct heuristic dynamic programming. Neural Netw 32:229–235

Liu D, Li H, Wang D (2013) Neural-network-based zero-sum game for discrete-time nonlinear systems via iterative adaptive dynamic programming algorithm. Neurocomputing 110:92–100

Liu D, Li H, Wang D (2014) Online synchronous approximate optimal learning algorithm for multi-player non-zero-sum games with unknown dynamics. IEEE Trans Syst Man Cybern Syst 44(8):1015–1027

Liu D, Yang X, Wang D, Wei Q (2015) Reinforcement-learning-based robust controller design for continuous-time uncertain nonlinear systems subject to input constraints. IEEE Trans Cybern 45(7):1372–1385

Luo B, Wu HN, Huang T, Liu D (2014) Data-based approximate policy iteration for affine nonlinear continuous-time optimal control design. Automatica 50(12):3281–3290

Luo B, Wu HN, Huang T, Liu D (2015) Reinforcement learning solution for HJB equation arising in constrained optimal control problem. Neural Netw 71:150–158

Luo B, Wu HN, Huang T (2015) Off-policy reinforcement learning for H_∞ control design. IEEE Trans Cybern 45(1):65–76

Luo B, Liu D, Huang T, Wang D (2016) Model-free optimal tracking control via critic-only Q-learning. IEEE Trans Neural Netw Learn Syst 27(10):2134–2144

Mehraeen S, Dierks T, Jagannathan S (2013) Zero-sum two-player game theoretic formulation of affine nonlinear discrete-time systems using neural networks. IEEE Trans Cybern 43(6):1641–1655

Murray JJ, Cox CJ, Lendaris GG, Saeks R (2002) Adaptive dynamic programming. IEEE Trans Syst Man Cybern Part C Appl Rev 32(2):140–153

Sokolov Y, Kozma R, Werbos L, Werbos P (2015) Complete stability analysis of a heuristic approximate dynamic programming control design. Automatica 59:9–18

Song R, Lewis FL, Wei Q, Zhang H, Jiang ZP, Levine D (2015) Multiple actor-critic structures for continuous-time optimal control using input-output data. IEEE Trans Neural Netw Learn Syst 26(4):851–865

Song R, Lewis FL, Wei Q (2017) Off-policy integral reinforcement learning method to solve nonlinear continuous-time multiplayer nonzero-sum games. IEEE Trans Neural Netw Learn Syst 28(3):704–713

Vamvoudakis KG, Lewis FL (2010) Online actor-critic algorithm to solve the continuous-time infinite horizon optimal control problem. Automatica 46(5):878–888

Vamvoudakis KG, Lewis FL (2011) Multi-player non-zero-sum games: Online adaptive learning solution of coupled Hamilton-Jacobi equations. Automatica 47(8):1556–1569

Wang FY, Zhang H, Liu D (2009) Adaptive dynamic programming: an introduction. IEEE Comput Intell Mag 4(2):39–47

Wang D, Liu D, Li H, Ma H (2014) Neural-network-based robust optimal control design for a class of uncertain nonlinear systems via adaptive dynamic programming. Inf Sci 282:167–179

Wang D, Liu D, Li H, Luo B, Ma H (2016) An approximate optimal control approach for robust stabilization of a class of discrete-time nonlinear systems with uncertainties. IEEE Trans Syst Man Cybern Syst 46(5):713–717

Wang D, Liu D, Zhang Q, Zhao D (2016) Data-based adaptive critic designs for nonlinear robust optimal control with uncertain dynamics. IEEE Trans Syst Man Cybern Syst 46(11):1544–1555

Wang D, He H, Liu D (2017) Adaptive critic nonlinear robust control: a survey. IEEE Trans Cybern 47(10):3429–3451

Wang D, Mu C, Liu D, Ma H (2017) On mixed data and event driven design for adaptive-critic-based nonlinear H_∞ control. IEEE Trans Neural Netw Learn Syst 99:1–13

Wang D, He H, Mu C, Liu D (2017) Intelligent critic control with disturbance attenuation for affine dynamics including an application to a microgrid system. IEEE Trans Ind Electron 64(6):4935–4944

Wei Q, Wang FY, Liu D, Yang X (2014) Finite-approximation-error based discrete-time iterative adaptive dynamic programming. IEEE Trans Cybern 44(12):2820–2833

Wei Q, Liu D, Yang X (2015) Infinite horizon self-learning optimal control of nonaffine discrete-time nonlinear systems. IEEE Trans Neural Netw Learn Syst 26(4):866–879

Wei Q, Liu D, Lin H (2016) Value iteration adaptive dynamic programming for optimal control of discrete-time nonlinear systems. IEEE Trans Cybern 46(3):840–853

Wei Q, Lewis FL, Liu D, Song R, Lin H (2016) Discrete-time local value iteration adaptive dynamic programming: convergence analysis. IEEE Trans Syst Man Cybern Syst 99:1–17

Wei Q, Liu D, Qiao L, Song R (2017) Adaptive dynamic programming for discrete-time zero-sum games. IEEE Trans Neural Netw Learn Syst 99:1–13

Werbos PJ (1977) Advanced forecasting methods for global crisis warning and models of intelligence. Gen Syst Yearb 22(6):25–38

Yang X, Liu D, Wei Q, Wang D (2016) Guaranteed cost neural tracking control for a class of uncertain nonlinear systems using adaptive dynamic programming. Neurocomputing 198:80–90

Yang X, Liu D, Ma H, Xu Y (2016) Online approximate solution of HJI equation for unknown constrained-input nonlinear continuous-time systems. Inf Sci 328:435–454

Zhang H, Cui L, Luo Y (2013) Near-optimal control for nonzero-sum differential games of continuous-time nonlinear systems using single-network ADP. IEEE Trans Cybern 43(1):206–216

Zhang H, Qin C, Jiang B, Luo Y (2014) Online adaptive policy learning algorithm for H_∞ state feedback control of unknown affine nonlinear discrete-time systems. IEEE Trans Cybern 44(12):2706–2718

Zhang H, Jiang H, Luo C, Xiao G (2016) Discrete-time nonzero-sum games for multiplayer using policy iteration-based adaptive dynamic programming algorithms. IEEE Trans Cybern 99:1–10

Zhang H, Cui X, Luo Y, Jiang H (2017) Finite-horizon H_∞ tracking control for unknown nonlinear systems with saturating actuators. IEEE Trans Neural Netw Learn Syst 99:1–13

Zhao D, Zhu Y (2015) MEC—a near-optimal online reinforcement learning algorithm for continuous deterministic systems. IEEE Trans Neural Netw Learn Syst 26(2):346–356

Zhao D, Xia Z, Wang D (2015) Model-free optimal control for affine nonlinear systems with convergence analysis. IEEE Trans Autom Sci Eng 12(4):1461–1468

Zhao D, Zhang Q, Wang D, Zhu Y (2016) Experience replay for optimal control of nonzero-sum game systems with unknown dynamics. IEEE Trans Cybern 46(3):854–865

Zhu Y, Zhao D, Li X (2016) Using reinforcement learning techniques to solve continuous-time non-linear optimal tracking problem without system dynamics. IET Control Theory Appl 10(12):1339–1347

Zhu Y, Zhao D, He H, Ji J (2017) Event-triggered optimal control for partially-unknown constrained-input systems via adaptive dynamic programming. IEEE Trans Ind Electron 64(5):4101–4109

Zhu Y, Zhao D (2017) Comprehensive comparison of online ADP algorithms for continuous-time optimal control. Artif Intell Rev 1-17. https://doi.org/10.1007/s10462-017-9548-4