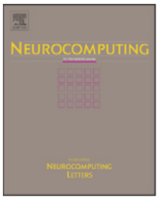
Neurocomputing 110 (2013) 92–100

Contents lists available at SciVerse ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom



Neural-network-based zero-sum game for discrete-time nonlinear systems via iterative adaptive dynamic programming algorithm ☆

Derong Liu *, Hongliang Li, Ding Wang

State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, PR China

ARTICLE INFO

Article history:
Received 3 April 2012
Received in revised form
2 September 2012
Accepted 19 November 2012
Communicated by H. Zhang
Available online 11 January 2013

Keywords:
Adaptive dynamic programming
Approximate dynamic programming
Heuristic dynamic programming
Neural networks
Zero-sum game
H_∞ optimal control

ABSTRACT

In this paper, we solve the zero-sum game problems for discrete-time affine nonlinear systems with known dynamics via iterative adaptive dynamic programming algorithm. First, a greedy heuristic dynamic programming iteration algorithm is developed to solve the zero-sum game problems, which can be used to solve the Hamilton–Jacobi–Isaacs equation associated with H_∞ optimal regulation control problems. The convergence analysis in terms of value function and control policy is provided. To facilitate the implementation of the algorithm, three neural networks are used to approximate the control policy, the disturbance policy, and the value function, respectively. Then, we extend the algorithm to H_∞ optimal tracking control problems through system transformation. Finally, two simulation examples are presented to demonstrate the effectiveness of the proposed scheme.

1. Introduction

During the last decades, adaptive dynamic programming (ADP) [1–5] has received much attention as an intelligent scheme for solving the optimal control problems. ADP algorithms can solve the optimal control problems by an online data-based procedure while the exact knowledge of the system dynamics is not required. Existing ADP approaches can be classified into several main schemes [6]: heuristic dynamic programming (HDP), dual heuristic dynamic programming (DHP), globalized dual heuristic dynamic programming (GDHP), and their action-dependent (AD) versions, ADHDP, ADDHP, and ADGDHP. The optimal state feedback control policy for nonlinear systems can be found by solving the Hamilton–Jacobi–Bellman (HJB) [7] equation, while it reduces to Riccati equation for linear quadratic regulator problem. However, the analytical solution of the HJB equation is difficult to obtain due to its inherently nonlinear nature. Many efforts using ADP have been made to solve the HJB equation [8–11]. Reinforcement learning (RL) [12] is a machine learning method for an agent or controller to learn the optimal control policies based on the observed responses from the environment or system. In recent years, RL has been applied to feedback control [13–17]. The main algorithms of RL, i.e., policy iteration (PI) and value iteration (VI) have been developed to solve the HJB equation of the optimal control problems. PI algorithms contain policy evaluation and policy improvement [18–21], where an initial stabilizing control law is required. VI algorithms solve the optimal control problems without requirement of an initial stabilizing control law [22–29]. However, most of the previous researches on ADP/RL algorithms provide an online or offline approach to the solution of optimal control problems assuming that the system is not affected by disturbance. Since disturbances widely exist in reality and affect the control performance, the ADP algorithm considering the disturbance is the main interest of the present paper.

As a kind of robust optimal control methods, the H_∞ optimal control seeks to not only minimize a cost function but also attenuate a worst-case disturbance [30]. The H_∞ control problem was converted into an L_2-gain optimal control problem [31] using the concept of dissipative system [32]. It relies on solving the Hamilton–Jacobi–Isaacs (HJI) equation which reduces to the game algebraic Riccati equation (GARE). The HJI equation is more difficult to solve than the HJB equation for the nonlinear dynamical systems. Furthermore, the H_∞ control has a strong connection to zero-sum games [33,34], where the controller is a minimizing player and the disturbance is a maximizing player. For Nash equilibrium, any player cannot improve its outcome by

☆ This work was supported in part by the National Natural Science Foundation of China under Grants 60904037, 60921061, and 61034002, in part by Beijing Natural Science Foundation under Grant 4102061, and in part by China Postdoctoral Science Foundation under Grant 201104162.

* Corresponding author. Tel.: +86 10 62557379; fax: +86 10 62650912.
E-mail addresses: derong.liu@ia.ac.cn (D. Liu), hongliang.li@ia.ac.cn (H. Li), ding.wang@ia.ac.cn (D. Wang).

0925-2312/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.neucom.2012.11.021

unilaterally changing its strategy. The Nash equilibrium solution is usually obtained by means of offline iterative computation, and the exact knowledge of the system dynamics is required.

ADP algorithms have been applied to the zero-sum game problems. In [35], an iterative algorithm to solve algebraic Riccati equations (AREs) with an indefinite quadratic term which was replaced by a sequence AREs with a negative semidefinite quadratic term was proposed. In [36], Vrabie et al. presented an ADP algorithm for determining online the Nash equilibrium solution for the two-player zero-sum differential game with linear continuous-time dynamics. In this case, only one of the players is learning and optimizing his behavior strategy while the other is passive and playing based on fixed policies. In [37], integral reinforcement learning was used to solve the same problem. In [38], the authors derived a PI algorithm to solve the stochastic optimal control problem in the presence of both additive and multiplicative noise.

For continuous-time nonlinear systems, Abu-Khalaf et al. [39,40] derived an H_∞ suboptimal state feedback controller for constrained input systems, and the two-player policy iteration scheme generated equations that are easier to solve compared to the original HJI equation of the corresponding constrained input zero-sum game. This method was offline and it contained two iteration loops. In [41], the authors extended the algorithms in [35] to nonlinear control systems including an inter iteration and an outer iteration. An initial stabilizing control law was not required and a local quadratic rate of convergence was given. The algorithm is supposed to have a higher accuracy and numerical stability than existing algorithms to solve HJI equations. In [42], Zhang et al. used four action networks and two critic networks to obtain the saddle point solution of the game, and the full knowledge of the system dynamics was required. If the saddle point did not exist, the mixed optimal control pair was obtained to make the performance index function reach the mixed optimum. The initial stabilizing control pair was required and it contained only one policy iteration loop.

In [43], Toussaint et al. designed a state feedback tracking control law using the H_∞ optimal control theory which produced a locally exponentially stable closed-loop system. In [44], Vamvoudakis et al. presented an online adaptive learning algorithm based on PI to solve the continuous-time two-player zero-sum game with infinite horizon cost for nonlinear systems with known dynamics. The adaptive algorithm was implemented as an actor/critic/disturbance structure that involved simultaneously continuous-time adaptation of critic, actor, and disturbance neural networks (NNs). In [45], Dierks et al. solved the HJI equation online and forward-in-time using a novel single online approximator-based scheme to achieve optimal regulation and tracking control of affine nonlinear continuous-time systems, while the optimal control input and worst disturbance input were calculated using the parameters of the approximator. The approximation errors resulting from NN were considered, the system dynamics were assumed to be known, and an initial stabilizing control was not required.

For discrete-time systems, Al-Tamimi et al. solved online the linear zero-sum game using HDP and DHP [46] and solved the GARE using a model-free Q-learning iterative algorithm [47]. In [48], Kim et al. developed a model-free H_∞ control design algorithm which was expressed in the form of linear matrix inequalities (LMI) for unknown linear discrete-time systems by using Q-learning. In [49], Wei et al. proposed an iterative adaptive critic design algorithm to solve a class of discrete-time two-person zero-sum games for Roesser type 2-D system. In [50], a novel data-based adaptive critic design using output feedback was proposed for discrete-time zero-sum games. In [51], Mehraeen et al. developed an iterative approach to solve offline the approximate HJI equation by using the Taylor series expansion of the value function and derived a sufficient condition for the convergence to saddle point.

In this paper, we solve the zero-sum game problems for discrete-time affine nonlinear systems with known dynamics via iterative ADP algorithm. First, a greedy HDP iteration algorithm is derived to solve the HJI equation associated with H_∞ optimal regulation control problems. The method in [51] have two iterative loops, i.e., the control and disturbance policies are asynchronously updated. In our scheme, only one iterative loop is used, and the initial stabilizing control policy is not required. The convergence in terms of value function and control policy is proved based on the work of [52–54]. To facilitate the implementation of the algorithm, three NNs are used to approximate the control policy, the disturbance policy, and the value function, respectively. Then, we extend this algorithm to H_∞ optimal tracking control problems for discrete-time nonlinear systems. Through system transformation, the tracking control problem is converted into a regulation problem. Finally, two simulation examples are presented to demonstrate the effectiveness of the proposed scheme.

The rest of the paper is organized as follows. Section 2 provides the problem formulation and discrete-time HJI equation for nonlinear systems. In Section 3, we derive a greedy HDP iteration algorithm, give the convergence analysis, and discuss the NN implementation of the iterative ADP algorithm. In Section 4, we solve the optimal tracking control problem for discrete-time nonlinear systems. Section 5 presents two simulation examples to demonstrate the effectiveness of the proposed algorithm and is followed by concluding remarks in Section 6.

2. Problem statement

Consider the discrete-time affine nonlinear dynamical systems described by

$$x_{k+1} = f(x_k) + g(x_k)u_k + h(x_k)w_k, \tag{1}$$

where $x_k \in \Omega \subseteq \mathbb{R}^n$ is the state vector, $u_k = u(x_k) \in \mathbb{R}^m$ is the control input, and $w_k = w(x_k) \in \mathbb{R}^q$ is the disturbance input. $f(x_k) \in \mathbb{R}^n$, $g(x_k) \in \mathbb{R}^{n \times m}$ and $h(x_k) \in \mathbb{R}^{n \times q}$ are smooth and differentiable functions. We assume that the following assumptions hold throughout the paper.

Assumption 1. $f(0) = 0$, and $x_k = 0$ is an equilibrium state of the system.

Assumption 2. $f + gu + hw$ is Lipschitz continuous on a compact set $\Omega \subseteq \mathbb{R}^n$ containing the origin.

Assumption 3. The system (1) is controllable in the sense that there exists a continuous control policy on Ω that asymptotically stabilizes the system.

In this paper, we define the infinite horizon cost function as follows:

$$J(x_0) = \sum_{k=0}^{\infty} \{x_k^T Q x_k + u_k^T R u_k - \gamma^2 w_k^T w_k\} = \sum_{k=0}^{\infty} U(x_k, u_k, w_k), \tag{2}$$

where Q and R are positive definite matrices, γ is a prescribed positive constant. Note that the control policy $u(x_k)$ must not only stabilize the system on Ω but also guarantee that (2) is finite, i.e., the control policy must be admissible [10].

Definition 1 (*Admissible Control Policy*). A control policy $u(x)$ is said to be admissible with respect to (2) on Ω, denoted by $u(x) \in \Psi(\Omega)$, if $u(x)$ is continuous on a compact set $\Omega \subseteq \mathbb{R}^n$, $u(0) = 0$, $u(x)$ stabilizes (1) on Ω and for $\forall x_0 \in \Omega$, $J(x_0)$ is finite.

For the admissible control policy $u(x_k)$ and disturbance policy $w(x_k)$, define the value function as

$$V(x_k,u_k,w_k)=\sum_{i=k}^{\infty}\{x_i^TQx_i+u_i^TRu_i-\gamma^2w_i^Tw_i\}. \tag{3}$$

The Hamilton function can be defined as

$$\begin{aligned}H(x_k,u_k,w_k)=&V(f+gu_k+hw_k,u_k,w_k)-V(x_k,u_k,w_k)\\&+x_k^TQx_k+u_k^TRu_k-\gamma^2w_k^Tw_k.\end{aligned} \tag{4}$$

According to [33,34], this control problem can be referred to a two-player zero-sum differential game, where the infinite-horizon value function is to be minimized by the control policy player $u(x_k)$ and maximized by the disturbance policy player $w(x_k)$. Our goal is to find the feedback saddle point solution (u_k^*,w_k^*) or the Nash equilibrium such that

$$V^*(x_0)=\min_{u_k}\max_{w_k}\{V(x_0,u_k,w_k)\} \tag{5}$$

or $V(x_0,u_k^*,w_k)\le V(x_0,u_k^*,w_k^*)\le V(x_0,u_k,w_k^*)$ for all u_k and w_k. The sufficient condition for the existence of a saddle point is

$$\min_{u_k}\max_{w_k}\{V(x_0,u_k,w_k)\}=\max_{w_k}\min_{u_k}\{V(x_0,u_k,w_k)\}. \tag{6}$$

According to Bellman's optimality principle, the optimal value function $V^*(x_k)$ satisfies the discrete-time HJI equation [33,46]

$$V^*(x_k)=\min_{u_k}\max_{w_k}\{U(x_k,u_k,w_k)+V^*(x_{k+1})\}. \tag{7}$$

The optimal control policy $u^*(x_k)$ and the worst case disturbance $w^*(x_k)$ should satisfy $\partial H(x_k,u_k,w_k)/\partial u_k=0$ and $\partial H(x_k,u_k,w_k)/\partial w_k=0$. Therefore, we obtain

$$u^*(x_k)=-\frac{1}{2}R^{-1}g^T(x_k)\frac{\partial V^*(x_{k+1})}{\partial x_{k+1}} \tag{8}$$

and

$$w^*(x_k)=\frac{1}{2}\gamma^{-2}h^T(x_k)\frac{\partial V^*(x_{k+1})}{\partial x_{k+1}}. \tag{9}$$

Then, the discrete-time HJI equation becomes

$$V^*(x_k)=x_k^TQx_k+u_k^{*T}Ru_k^*-\gamma^2w_k^{*T}w_k^*+V^*(x_{k+1}). \tag{10}$$

This equation reduces to GARE in the zero-sum linear quadratic case. However, in the general nonlinear case, the value function of the optimal control problem cannot be obtained.

For the problem of disturbance attenuation, we need the definition of L_2-gain for discrete-time nonlinear system.

Definition 2. (L_2-gain) The nonlinear system (1) with state feedback control policy u_k and disturbance $w_k\in L_2$ is said to have an L_2-gain less than or equal to γ if

$$\sum_{k=0}^{\infty}\{x_k^TQx_k+u_k^TRu_k\}\le\sum_{k=0}^{\infty}\gamma^2w_k^Tw_k. \tag{11}$$

The disturbance of w_k is locally attenuated by a real value $\gamma>0$ if there exists a neighborhood around the origin such that $\forall w_k\in L_2$ for which the trajectories of the closed-loop system (1) starting from the origin remain in the same neighborhood.

3. Adaptive dynamic programming for zero-sum game

This section consists of three subsections. The greedy HDP iteration algorithm is developed to solve the H_∞ optimal regulation control problems for discrete-time nonlinear systems in the first subsection. The corresponding convergence proof in terms of value function and control policy is presented in the second subsection, and the NN implementation of the algorithm is given in the third subsection.

3.1. Derivation of ADP algorithm for zero-sum game

Since direct solution of the HJI equation for nonlinear systems is computationally intensive, we present a greedy HDP iteration algorithm based on Bellman's principle of optimality.

First, we start with an initial value function V_0 which is not necessarily optimal, and set $\gamma>0$. Then, we find $V_1(x_k)$ by solving (12) with $i=0$

$$\begin{aligned}V_{i+1}(x_k)=&\min_{u_k}\max_{w_k}\{x_k^TQx_k+u_k^TRu_k-\gamma^2w_k^Tw_k\\&+V_i(f(x_k)+g(x_k)u_k+h(x_k)w_k)\}.\end{aligned} \tag{12}$$

The greedy policies $u_i(x_k)$ and $w_i(x_k)$ are updated by

$$u_i(x_k)=-\frac{1}{2}R^{-1}g^T(x_k)\frac{\partial V_i(x_{k+1})}{\partial x_{k+1}} \tag{13}$$

and

$$w_i(x_k)=\frac{1}{2}\gamma^{-2}h^T(x_k)\frac{\partial V_i(x_{k+1})}{\partial x_{k+1}}. \tag{14}$$

Therefore, $V_1(x_k)$ is calculated by (15) with $i=0$

$$\begin{aligned}V_{i+1}(x_k)=&x_k^TQx_k+u_i^T(x_k)Ru_i(x_k)-\gamma^2w_i^T(x_k)w_i(x_k)\\&+V_i(f(x_k)+g(x_k)u_i(x_k)+h(x_k)w_i(x_k)).\end{aligned} \tag{15}$$

After $V_1(x_k)$ is found, we repeat the same value iteration process for $i=1,2,\ldots$. Furthermore, it should be satisfied that $V_i(0)=0$, $\forall i\ge0$. Note that i is the iteration index and k is the time index. As a value iteration algorithm, this iterative ADP algorithm does not require any initial stabilizing controller. In the next subsection we will prove the convergence of the algorithm, i.e., $V_i\to V^*$, $u_i\to u^*$ and $w_i\to w^*$ as $i\to\infty$.

3.2. Convergence analysis of the iterative ADP algorithm for zero-sum game

Theorem 1 (*Monotonicity property*). *Define the control policy sequence $\{u_i\}$ as in (13), the disturbance sequence $\{w_i\}$ as in (14), and the value function sequence $\{V_i\}$ as in (15) with an initial value function V_0. If $V_1(x_k)\ge V_0(x_k)$ holds $\forall x_k$, the value function sequence $\{V_i\}$ is a monotonically non-decreasing sequence, i.e., $V_{i+1}\ge V_i$, $\forall i\ge0$. If $V_1(x_k)\le V_0(x_k)$ holds $\forall x_k$, the value function sequence $\{V_i\}$ is a monotonically non-increasing sequence, i.e., $V_{i+1}\le V_i$, $\forall i\ge0$.*

Proof. If $V_1(x_k)\ge V_0(x_k)$ holds $\forall x_k$, assume that $V_i(x_k)\ge V_{i-1}(x_k)$, $\forall i$ and x_k. Then,

$$\begin{aligned}V_{i+1}(x_k)=&\min_{u_k}\max_{w_k}\{x_k^TQx_k+u_k^TRu_k-\gamma^2w_k^Tw_k\\&+V_i(f(x_k)+g(x_k)u_k+h(x_k)w_k)\}\\\ge&\min_{u_k}\max_{w_k}\{x_k^TQx_k+u_k^TRu_k-\gamma^2w_k^Tw_k\\&+V_{i-1}(f(x_k)+g(x_k)u_k+h(x_k)w_k)\}\\=&V_i(x_k).\end{aligned} \tag{16}$$

Therefore, the value function sequence $\{V_i\}$ is a monotonically non-decreasing sequence by mathematical induction. The other part can be proved in the same way. □

Remark 1. From Theorem 1, we can see that the monotonicity property of the value function V_i is determined by the relationship between V_0 and V_1, i.e., $V_0\ge V_1$ or $V_0\le V_1$, $\forall x_k$. If we set $V_0(\cdot)=0$, we can easily find that $V_1\ge V_0$, i.e., the value function sequence $\{V_i\}$ is a monotonically non-decreasing sequence. Besides, the monotonicity property is still valid if we can find that $V_i\ge V_{i+1}$ or $V_i\le V_{i+1}$ for any x_k and some i.

Next, we will demonstrate the convergence of the iterative ADP algorithm according to the work of [52–54].

Theorem 2 (*Convergence property*). *Suppose the condition $0 \le V^*(f(x)+g(x)u(x)+h(x)w(x)) \le \theta U(x,u,w)$ holds uniformly for some $0<\theta<\infty$ and that $0 \le \alpha V^* \le V_0 \le \beta V^*$, $0 \le \alpha \le 1$ and $1 \le \beta < \infty$. The control policy sequence $\{u_i\}$, the disturbance sequence $\{w_i\}$ and the value function sequence $\{V_i\}$ are iteratively updated by (13)–(15). Then the value function V_i approaches V^* according to the following inequalities*:

$$\left[1+\frac{\alpha-1}{(1+\theta^{-1})^i}\right]V^*(x) \le V_i(x) \le \left[1+\frac{\beta-1}{(1+\theta^{-1})^i}\right]V^*(x). \tag{17}$$

Define $V_\infty(x_k)=\lim_{i\to\infty}V_i(x_k)$, then

$$V_\infty(x_k)=V^*(x_k). \tag{18}$$

Proof. First, we will demonstrate that the system defined in this paper satisfies the conditions of Theorem 2. According to Assumption 2, $f+gu+hw$ is Lipschitz continuous, and the system state cannot jump to infinity by any one step of finite control input, i.e., $f(x)+g(x)u(x)+h(x)w(x)$ is finite. Considering that $V^*(x_k,u,w)$ is finite for any finite state and control, there exists some $0<\theta<\infty$ that makes $0 \le V^*(f(x)+g(x)u(x)+h(x)w(x)) \le \theta U(x,u,w)$ hold uniformly. In addition, for any finite initial value function V_0, there exist α and β such that $0 \le \alpha V^* \le V_0 \le \beta V^*$ is satisfied, where $0 \le \alpha \le 1$ and $1 \le \beta < \infty$.

Next, we will demonstrate the left hand side of the inequality (17) by mathematical induction, i.e.,

$$\left[1+\frac{\alpha-1}{(1+\theta^{-1})^i}\right]V^*(x) \le V_i(x). \tag{19}$$

When $i=1$, since

$$\frac{\alpha-1}{1+\theta}(\theta U(x_k,u(x_k),w(x_k))-V^*(x_{k+1})) \le 0, \quad 0 \le \alpha \le 1 \tag{20}$$

and $\alpha V^* \le V_0$, $\forall x_k$, we have

$$\begin{aligned}
V_1(x_k) &= \min_{u_k}\max_{w_k}\{U(x_k,u_k,w_k)+V_0(x_{k+1})\} \\
&\ge \min_{u_k}\max_{w_k}\{U(x_k,u_k,w_k)+\alpha V^*(x_{k+1})\} \\
&\ge \min_{u_k}\max_{w_k}\left\{\left(1+\theta\frac{\alpha-1}{1+\theta}\right)U(x_k,u_k,w_k)+\left(\alpha-\frac{\alpha-1}{1+\theta}\right)V^*(x_{k+1})\right\} \\
&= \left(1+\frac{\alpha-1}{1+\theta^{-1}}\right)\min_{u_k}\max_{w_k}\{U(x_k,u_k,w_k)+V^*(x_{k+1})\} \\
&= \left(1+\frac{\alpha-1}{1+\theta^{-1}}\right)V^*(x_k).
\end{aligned} \tag{21}$$

Assume that the inequality (19) holds for $i-1$. Then, we have

$$\begin{aligned}
V_i(x_k) &= \min_{u_k}\max_{w_k}\{U(x_k,u_k,w_k)+V_{i-1}(x_{k+1})\} \\
&\ge \min_{u_k}\max_{w_k}\left\{U(x_k,u_k,w_k)+\left[1+\frac{\alpha-1}{(1+\theta^{-1})^{i-1}}\right]V^*(x_{k+1})\right\} \\
&\ge \min_{u_k}\max_{w_k}\left\{\left[1+\frac{(\alpha-1)\theta^i}{(\theta+1)^i}\right]U(x_k,u_k,w_k)\right. \\
&\quad \left.+\left[1+\frac{\alpha-1}{(1+\theta^{-1})^{i-1}}-\frac{(\alpha-1)\theta^{i-1}}{(\theta+1)^i}\right]V^*(x_{k+1})\right\} \\
&= \left[1+\frac{(\alpha-1)\theta^i}{(\theta+1)^i}\right]\min_{u_k}\max_{w_k}\{U(x_k,u_k,w_k)+V^*(x_{k+1})\} \\
&= \left[1+\frac{\alpha-1}{(1+\theta^{-1})^i}\right]V^*(x_k).
\end{aligned} \tag{22}$$

Thus, the left-hand side of inequality (17) is proved and the right-hand side can be shown in the same way.

Lastly, we will demonstrate the convergence of value function as the iteration index i goes to infinity. When $i\to\infty$, for $0<\theta<\infty$, we have

$$\lim_{i\to\infty}\left[1+\frac{\alpha-1}{(1+\theta^{-1})^i}\right]V^*(x_k)=V^*(x_k) \tag{23}$$

and

$$\lim_{i\to\infty}\left[1+\frac{\beta-1}{(1+\theta^{-1})^i}\right]V^*(x_k)=V^*(x_k). \tag{24}$$

Therefore, we can get

$$V_\infty(x_k)=V^*(x_k). \tag{25}$$

The proof is completed. □

Remark 2. From the above demonstration, we see that we can find upper and lower bounds for every iterative value function based on the optimal value function. As the iterative index i increases, the upper bound will exponentially approach the lower bound. When the iterative index i goes to infinity, the upper bound will be nearly equal to the lower bound, which is just the optimal value function. According to the inequality (17), smaller θ will lead to faster convergence speed of the value function. Moreover, it should be mentioned that conditions in Theorem 2 can be satisfied according to Assumptions 1–3, which are some mild assumptions for general control problems.

Specially, when $V_0=0$, we can have $\alpha=0$, $\beta=1$. From the inequality (17), we have

$$\left[1-\frac{1}{(1+\theta^{-1})^i}\right]V^*(x) \le V_i(x) \le V^*(x). \tag{26}$$

According to the results of Theorem 2, we can derive the following theorem.

Theorem 3. *Define the control policy sequence $\{u_i\}$ as in (13), the disturbance sequence $\{w_i\}$ as in (14), and the value function sequence $\{V_i\}$ as in (15). If the system state x_k is controllable, then the control pair (u_i,w_i) converges to (u^*,w^*) as $i\to\infty$.*

Proof. According to Theorem 2, we have proved that $\lim_{i\to\infty}V_i(x_k)=V_\infty(x_k)=V^*(x_k)$, so

$$V_\infty(x_k)=\min_{u_k}\max_{w_k}\{U(x_k,u_k,w_k)+V_\infty(x_{k+1})\}. \tag{27}$$

That is to say the value function sequence $\{V_i\}$ converges to the optimal value function of the discrete-time HJI equation. Considering (8) and (13), (9) and (14), the corresponding control pair (u_i,w_i) converges to the saddle point (u^*,w^*) as $i\to\infty$. □

3.3. NN implementation of the iterative ADP algorithm

The structure diagram of the HDP algorithm is given in Fig. 1. The critic network approximates the value function, the action network approximates the control policy, and the disturbance network approximates the disturbance policy.

We choose the three-layer feed-forward NN as our function approximation scheme. The output of the action network can be formulated as

$$\tilde{u}_i(x_k)=\omega_{a(i)}^T\sigma(v_{a(i)}^T x_k). \tag{28}$$

The target of control input is calculated in (13). The error function of the action network can be defined as

$$e_{a(i)}(x_k)=\tilde{u}_i(x_k)-u_i(x_k). \tag{29}$$

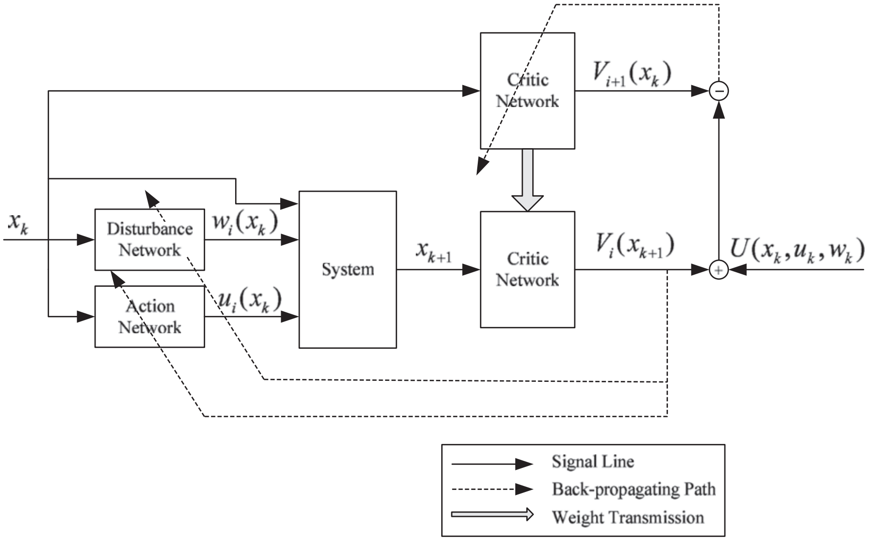


Fig. 1. The structure diagram of HDP algorithm.

The weights of the action network are updated to minimize the following objective function:

$$E_{a(i)}(x_k)=\frac{1}{2}e^T_{a(i)}(x_k)e_{a(i)}(x_k). \tag{30}$$

The output of the disturbance network can be formulated as

$$\tilde{w}_i(x_k)=\omega^T_{d(i)}\sigma(v^T_{d(i)}x_k). \tag{31}$$

The target of disturbance input is calculated in (14). The error function of the disturbance network can be defined as

$$e_{d(i)}(x_k)=\tilde{w}_i(x_k)-w_i(x_k). \tag{32}$$

The weights of the disturbance network are updated to minimize the following objective function:

$$E_{d(i)}(x_k)=\frac{1}{2}e^T_{d(i)}(x_k)e_{d(i)}(x_k). \tag{33}$$

The output of the critic network is denoted as

$$\tilde{V}_{i+1}(x_k)=\omega^T_{c(i+1)}\sigma(v^T_{c(i+1)}x_k). \tag{34}$$

The target value function is given in (15), where $V_i(x_{k+1})=\omega^T_{c(i)}\sigma(v^T_{c(i)}x_{k+1})$. Then, the error function for training critic network is defined as

$$e_{c(i+1)}(x_k)=\tilde{V}_{i+1}(x_k)-V_{i+1}(x_k), \tag{35}$$

and the objective function to be minimized is defined as

$$E_{c(i+1)}(x_k)=\frac{1}{2}e^T_{c(i+1)}(x_k)e_{c(i+1)}(x_k). \tag{36}$$

With these objective functions, many methods like gradient descent and Levenberg–Marquardt (LM) algorithm can be used to tune the weights of NN.

Finally, a summary of this algorithm is given as follows:

1 Initialize the parameters $j^a_{\max}$, $j^d_{\max}$, $j^c_{\max}$, ε_a, ε_d, ε_c, $i_{\max}$, ξ, γ, Q, R, and the weights of NNs.
2 Set the iterative index $i=0$ and $V_0=0$.
3 Choose randomly an array of p state vector $[x^1_k, x^2_k, \ldots, x^p_k]$. Compute the target of the action network $[u_i(x^1_k), u_i(x^2_k), \ldots, u_i(x^p_k)]$ by (13), and train the action network until the given accuracy ε_a or the maximum number of iterations $j^a_{\max}$ is reached. Compute the target of the disturbance network $[w_i(x^1_k), w_i(x^2_k), \ldots, w_i(x^p_k)]$ by (14), and train the disturbance network until the given accuracy ε_d or the maximum number of iterations $j^d_{\max}$ is reached.
4. Compute the output of the action network $[\tilde{u}_i(x^1_k), \tilde{u}_i(x^2_k), \ldots, \tilde{u}_i(x^p_k)]$, the output of the disturbance network $[\tilde{w}_i(x^1_k), \tilde{w}_i(x^2_k), \ldots, \tilde{w}_i(x^p_k)]$, and the output of the critic network $[\tilde{V}_i(x^1_{k+1}), \tilde{V}_i(x^2_{k+1}), \ldots, \tilde{V}_i(x^p_{k+1})]$. Then compute the target of the critic network $[V_{i+1}(x^1_k), V_{i+1}(x^2_k), \ldots, V_{i+1}(x^p_k)]$ by (15). Train the critic network until the given accuracy ε_c or the maximum number of iterations $j^c_{\max}$ is reached.
5. If $i>i_{\max}$ or $\|V_{i+1}(x^s_k)-V_i(x^s_k)\|^2 \le \xi, s=1,2,\ldots,p$, go to step (6); otherwise, set the iterative index $i=i+1$ and go to step (3).
6. The final near optimal control policy is obtained, and stop.

4. Adaptive dynamic programming for H_∞ optimal tracking control

In this section, we will study the H_∞ optimal tracking control problem for nonlinear discrete-time systems based on the method developed in the previous section. The objective for optimal tracking control problem is to design an optimal controller to make the nonlinear system (1) track a reference trajectory x_{dk} in an optimal manner. Except for the assumptions given in Section 2, we assume that $g(x_k)=G$ in (1) is an invertible input transformation matrix, and $h(x_k)=H$ in (1) is a constant matrix. The dynamics of the reference trajectory is defined as

$$x_{dk+1}=f(x_{dk})+Gu_{dk}, \tag{37}$$

where $x_{dk}\in\mathbb{R}^n$, and $f(\cdot)$ is the same as in (1). Based on the work of [17,43], we define the desired control input u_{dk} corresponding to the reference trajectory x_{dk} as

$$u_{dk}=G^{-1}(x_{dk+1}-f(x_{dk})), \tag{38}$$

where G^{-1} is the inverse matrix of input transformation matrix G. Then, we define the state tracking error as

$$e_k=x_k-x_{dk}. \tag{39}$$

By using (1) and (37)–(39), the tracking error dynamics is defined as

$$e_{k+1}=f_{ek}+Gu_{ek}+Hw_{ek}, \tag{40}$$

where $f_{ek}=f(e_k+x_{dk})-f(x_{dk})$, $u_{ek}=u_k-u_{dk}$ is the control input to the new system (40), and $w_{ek}=w_k$. It should be mentioned that e_k=0 is an equilibrium point of (40). In this sense, the nonlinear tracking problem is converted into a regulation problem.

To track the reference trajectory in an optimal manner, our goal is to design an optimal control policy u_{ek} which minimizes the infinite horizon cost function

$$J_e(x_0)=\sum_{k=0}^{\infty}\{e_k^TQ_ee_k+u_{ek}^TR_eu_{ek}-\gamma_e^2w_{ek}^Tw_{ek}\}=\sum_{k=0}^{\infty}U(e_k,u_{ek},w_{ek}), \tag{41}$$

where Q_e and R_e are positive definite matrices, γ_e is a prescribed positive constant. For the admissible control policy u_{ek} and disturbance policy w_{ek}, we define the value function as

$$V_e(e_k,u_{ek},w_{ek})=\sum_{i=k}^{\infty}\{e_k^TQ_ee_k+u_{ek}^TR_eu_{ek}-\gamma_e^2w_{ek}^Tw_{ek}\}. \tag{42}$$

Therefore, we obtain

$$u_{ek}^*=-\frac{1}{2}R_e^{-1}G^T\frac{\partial V^*(e_{k+1})}{\partial e_{k+1}} \tag{43}$$

and

$$w_{ek}^*=\frac{1}{2}\gamma_e^{-2}H^T\frac{\partial V^*(e_{k+1})}{\partial e_{k+1}}. \tag{44}$$

Then, the discrete-time HJI equation becomes

$$V_e^*(e_k)=e_k^TQ_ee_k+u_{ek}^{*T}R_eu_{ek}^*-\gamma_e^2w_{ek}^{*T}w_{ek}^*+V_e^*(e_{k+1}). \tag{45}$$

Thus we can solve the H_∞ optimal tracking problem using the iterative ADP algorithm developed in Section 3.

5. Simulation study

In this section, two simulation examples are provided to illustrate the applicability of the present results.

5.1. Example 1

We use an F-16 aircraft autopilot as the simulation example which is taken from [46]. The discrete-time plant model of this aircraft dynamics is $x_{k+1}=Ax_k+Gu_k+Hw_k$, where

$$A=\begin{bmatrix}0.906488 & 0.0816012 & -0.0005\\ 0.0741349 & 0.90121 & -0.000708383\\ 0 & 0 & 0.132655\end{bmatrix},$$

$$G=\begin{bmatrix}-0.00150808\\ -0.0096\\ 0.867345\end{bmatrix},$$

$$H=\begin{bmatrix}-0.00951892\\ 0.00038373\\ 0\end{bmatrix}.$$

The system states are $x=[x_1\ x_2\ x_3]^T$, where x_1 is the angle of attack, x_2 is the pitch rate, and x_3 is the elevator deflection angle. The operation region of the system is selected as $-1\le x_1\le 1$, $-1\le x_2\le 1$, and $-1\le x_3\le 1$. The weight matrices Q and R are chosen as identity matrices, and the disturbance attenuation is $\gamma=1$. The structures of action network, disturbance network, and critic network are all chosen as 3–5–1. We use LM algorithm to tune the weights of three NNs. After iterating for 200 times, the convergence of the value function at $x_0=[0.5\ 0.5\ 0.5]^T$ is given in Fig. 2. We apply the obtained control policy to the system for 500 time steps. The corresponding state trajectories are given in Fig. 3, and the control input is shown in Fig. 4.

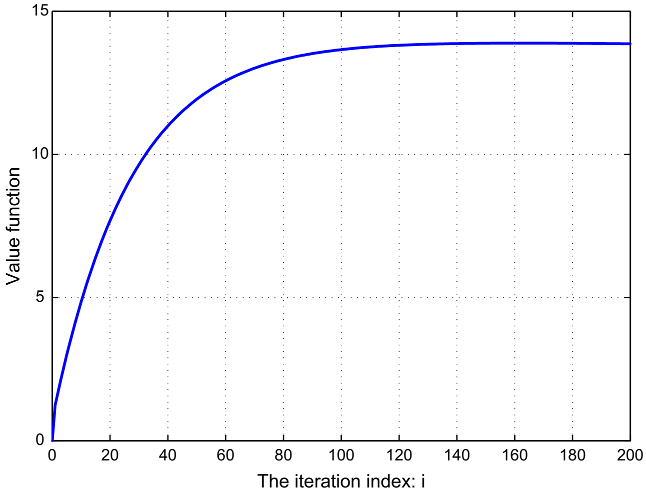


Fig. 2. Convergence process of the value function at $x=[0.5\ 0.50.5]^T$.

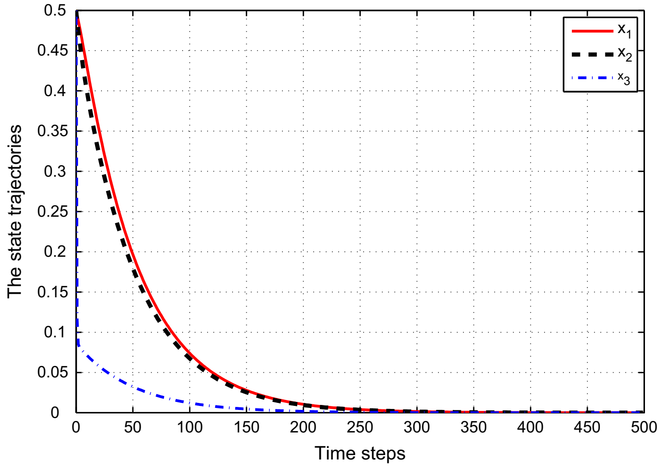


Fig. 3. The state trajectories.

5.2. Example 2

Example 2 is obtained from [24] with some modifications. The discrete-time plant model is $x_{k+1}=f(x_k)+g(x_k)u_k+h(x_k)w_k$, where

$$f(x_k)=\begin{bmatrix}0.2x_{1k}e^{x_{2k}^2}\\ 0.3x_{2k}^3\end{bmatrix},$$

$$g(x_k)=\begin{bmatrix}-0.5 & 0\\ 0 & -0.5\end{bmatrix},$$

$$h(x_k)=\begin{bmatrix}0.1 & 0\\ 0 & 0.1\end{bmatrix}.$$

The system states are $x_k=[x_{1k}x_{2k}]^T$. The operation region of the system is selected as $-1\le x_1\le 1$, $-1\le x_2\le 1$. The weight matrices Q and R are chosen as identity matrices, and the disturbance attenuation is $\gamma=5$. The structures of action network, disturbance network, and critic network are chosen as 2–8–2, 2–8–2, and 2–8–1, respectively. The activation function of hidden layer is chosen as tanh$(\cdot)$ and the activation function of output layer is chosen as linear function. We use LM algorithm to tune the weights of three NNs. After iterating for 10 times, the convergence of the value function at $x_0=[0.5-0.5]^T$ is given in Fig. 5. Then, we apply the obtained nearly optimal control policy to the system for 30 time steps. A disturbance $w_k=[0.5e^{-0.2k}\ 0.5e^{-0.2k}]^T$ is introduced into the system at $k=0$.

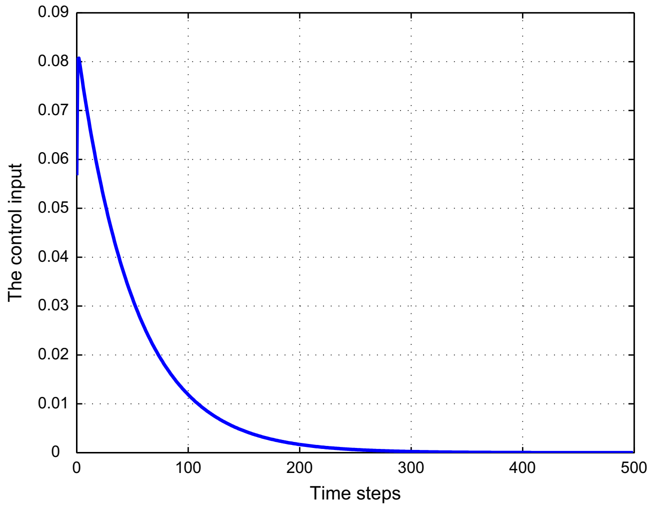


Fig. 4. The control input.

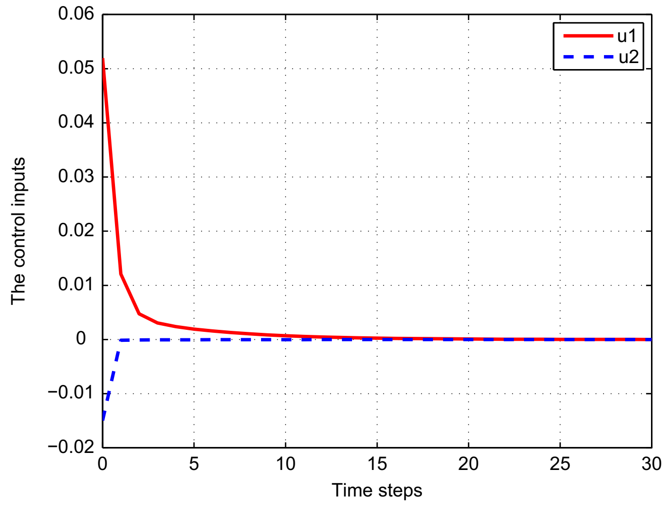


Fig. 7. The control inputs.

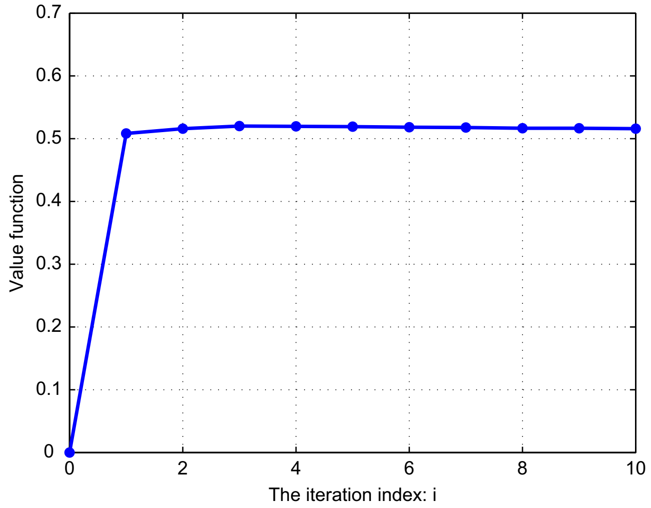


Fig. 5. Convergence process of the value function at $x = [0.5\ -0.5]^T$.

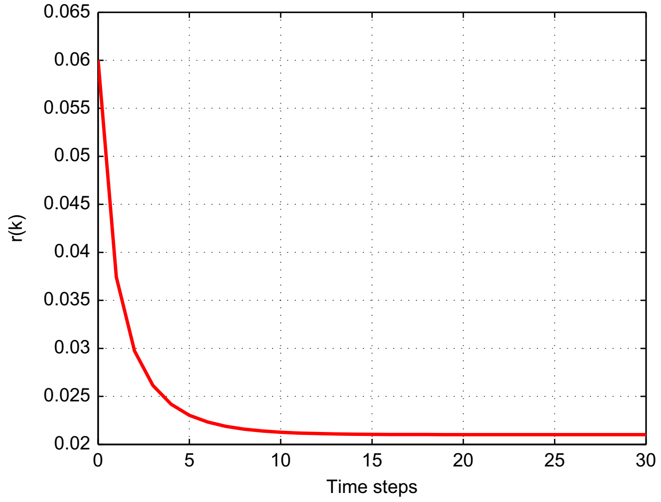


Fig. 8. The performance metric.

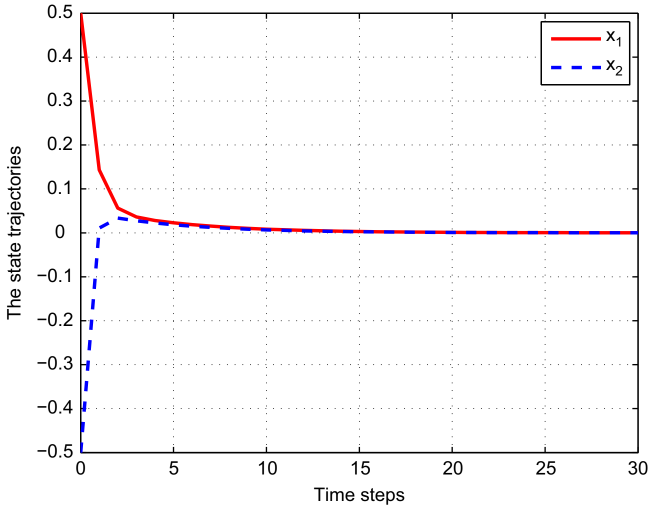


Fig. 6. The state trajectories.

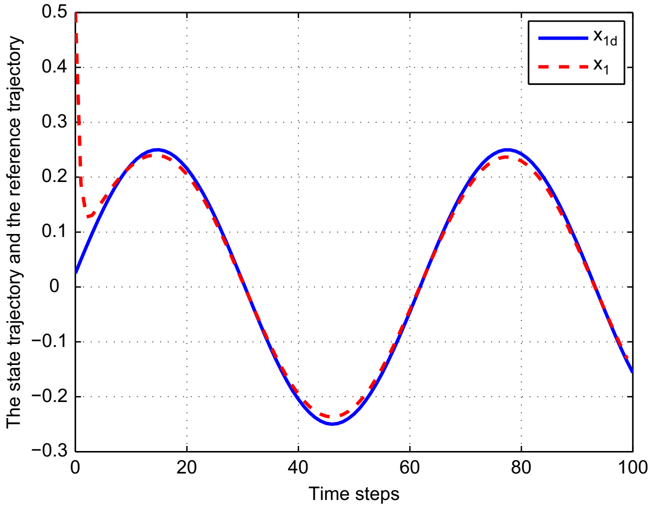


Fig. 9. The state tracking trajectory of x_1.

The corresponding state trajectories are given in Fig. 6, and the control inputs are shown in Fig. 7. To evaluate the performance of the system, a performance metric is defined as [51]

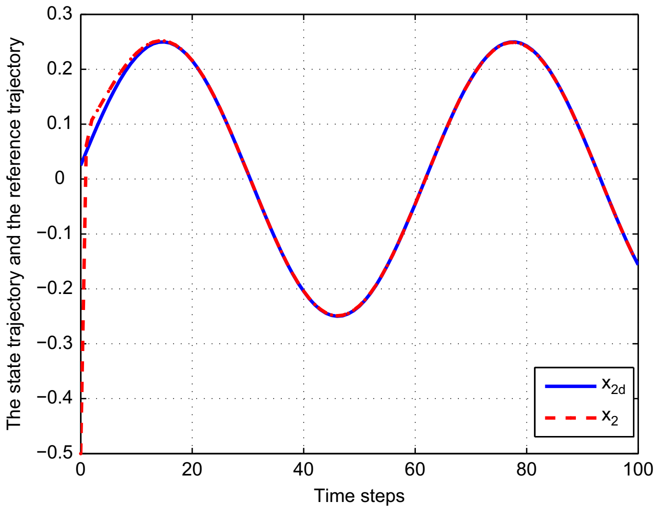


Fig. 10. The state tracking trajectory of x_2.

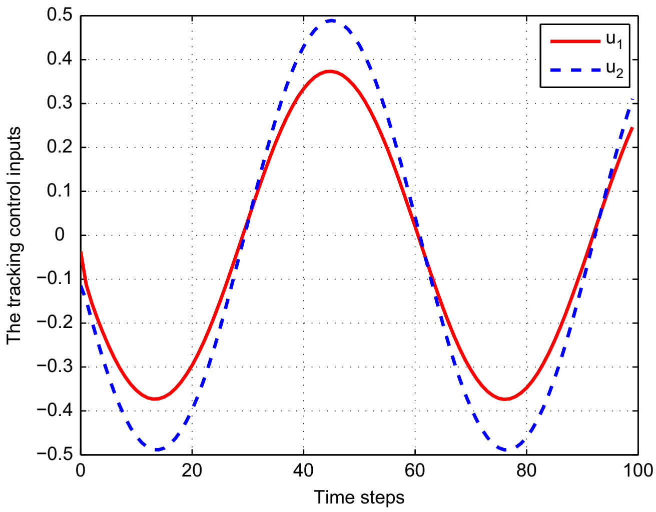


Fig. 11. The tracking control inputs.

$$r(k)=\frac{\sum_{i=0}^{k}(x_i^TQx_i+u_i^TRu_i)}{\sum_{i=0}^{k}\gamma^2 w_i^T w_i}. \tag{46}$$

We can see that $r(k)$ converges to 0.021 from Fig. 8.

It is clear from the simulation results that the iterative ADP algorithm proposed in this paper is very effective in solving H_∞ optimal regulation problem.

To implement the H_∞ optimal tracking control, the parameters are kept constant, and the reference trajectory is selected as

$$x_{dk}=\begin{bmatrix}0.25\sin(0.1k)\\ 0.25\sin(0.1k)\end{bmatrix}.$$

After convergence, we apply the obtained nearly optimal tracking controller to the system for 100 time steps. The disturbance w_k is also introduced into the system at $k=0$. The corresponding state tracking trajectories are given in Figs. 9 and 10, and the tracking control inputs are shown in Fig. 11.

These simulation results verify the excellent performance of the tracking controller developed by the iterative ADP algorithm considering the disturbance.

6. Conclusion

A greedy iterative HDP algorithm is developed in this paper to solve the zero-sum game problems for discrete-time affine nonlinear systems. The convergence analysis in terms of value function and control policy is proved. Three NNs are used to approximate the control policy, the disturbance policy, and the value function, respectively. This algorithm is also extended to H_∞ optimal tracking control problems. The simulation examples confirmed the validity the proposed scheme.

References

[1] P.J. Werbos, Approximate dynamic programming for real-time control and neural modeling, in: D.A. White, D.A. Sofge (Eds.), Handbook of Intelligent Control, Van Nostrand Reinhold, New York, 1992. (Chapter 13).

[2] D.P. Bertsekas, J.N. Tsitsiklis, Neuro-Dynamic Programming, Athena Scientific, Belmont, MA, 1996.

[3] J. Si, A.G. Barto, W.B. Powell, D.C. Wunsch (Eds.), Handbook of Learning and Approximate Dynamic Programming, IEEE Press, Wiley, New York, 2004.

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

[5] F.L. Lewis, D. Vrabie, Reinforcement learning and adaptive dynamic programming for feedback control, IEEE Circ. Syst. Mag. 9 (2009) 32–50.

[6] D.V. Prokhorov, D.C. Wunsch, Adaptive critic designs, IEEE Trans. Neural Networks 8 (1997) 997–1007.

[7] F.L. Lewis, V.L. Syrmos, Optimal Control, Wiley, New York, 1995.

[8] R. Beard, G. Saridis, J. Wen, Galerkin approximations of the generalized Hamilton–Jacobi–Bellman equation, Automatica 33 (1997) 2158–2177.

[9] J.J. Murray, C.J. Cox, G.G. Lendaris, R. Saeks, Adaptive dynamic programming, IEEE Trans. Syst. Man Cybernet. Part C: Appl. Rev. 32 (2002) 140–153.

[10] M. Abu-Khalaf, F.L. Lewis, Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach, Automatica 41 (2005) 779–791.

[11] T. Cheng, F.L. Lewis, M. Abu-Khalaf, A neural network solution for fixed-final time optimal control of nonlinear systems, Automatica 43 (2007) 482–490.

[12] R.S. Sutton, A.G. Barto, Reinforcement Learning: An Introduction, The MIT Press, Cambridge, MA, 1998.

[13] F.L. Lewis, G. Lendaris, D. Liu, Special issue on approximate dynamic programming and reinforcement learning for feedback control, IEEE Trans. Syst. Man Cybernet. Part B: Cybernet. 38 (2008) 896–897.

[14] P. He, S. Jagannathan, Reinforcement learning neural-network-based controller for nonlinear discrete-time systems with input constraints, IEEE Trans. Syst. Man Cybernet. Part B: Cybernet. 37 (2007) 425–436.

[15] Q. Yang, J.B. Vance, S. Jagannathan, Control of nonaffine nonlinear discrete-time systems using reinforcement-learning-based linearly parameterized neural networks, IEEE Trans. Syst. Man Cybernet. Part B: Cybernet. 38 (2008) 994–1001.

[16] P. Shih, B.C. Kaul, S. Jagannathan, J.A. Drallmeier, Reinforcement-learning-based dual-control methodology for complex nonlinear discrete-time systems with application to spark engine EGR operation, IEEE Trans. Neural Networks 19 (2008) 1369–1388.

[17] T. Dierks, S. Jagannathan, Online optimal control of nonlinear discrete-time systems using approximate dynamic programming, J. Control Theory Appl. 9 (2011) 361–369.

[18] R.A. Howard, Dynamic Programming and Markov Processes, MIT Press, Cambridge, MA, 1960.

[19] D. Vrabie, O. Pastravanu, M. Abu-Khalaf, F.L. Lewis, Adaptive optimal control for continuous-time linear systems based on policy iteration, Automatica 45 (2009) 477–484.

[20] D. Vrabie, F.L. Lewis, Neural network approach to continuous-time direct adaptive optimal control for partially unknown nonlinear systems, Neural Networks 22 (2009) 237–246.

[21] K.G. Vamvoudakis, F.L. Lewis, Online actor-critic algorithm to solve the continuous-time infinite horizon optimal control problem, Automatica 46 (2010) 878–888.

[22] A. Al-Tamimi, F.L. Lewis, M. Abu-Khalaf, Discrete-time nonlinear HJB solution using approximate dynamic programming: convergence proof, IEEE Trans. Syst. Man Cybernet. Part B: Cybernet. 38 (2008) 943–949.

[23] T. Dierks, B.T. Thumati, S. Jagannathan, Optimal control of unknown affine nonlinear discrete-time systems using offline-trained neural networks with proof of convergence, Neural Networks 20 (2009) 851–860.

[24] H. Zhang, Q. Wei, Y. Luo, A novel infinite-time optimal tracking control scheme for a class of discrete-time nonlinear systems via the greedy HDP iteration algorithm, IEEE Trans. Syst. Man Cybernet. Part B: Cybernet. 38 (2008) 937–942.

[25] H. Zhang, Y. Luo, D. Liu, Neural-network-based near-optimal control for a class of discrete-time affine nonlinear systems with control constraints, IEEE Trans. Neural Networks 20 (2009) 1490–1503.

[26] F.Y. Wang, N. Jin, D. Liu, Q. Wei, Adaptive dynamic programming for finite-horizon optimal control of discrete-time nonlinear systems with ε-error bound, IEEE Trans. Neural Networks 22 (2011) 24–36.

[27] R. Song, H. Zhang, Y. Luo, Q. Wei, Optimal control laws for time-delay systems with saturating actuators based on heuristic dynamic programming, Neurocomputing 73 (2010) 3020–3027.

[28] X. Zhang, H. Zhang, Q. Sun, Y. Luo, Adaptive dynamic programming-based optimal control of unknown nonaffine discrete-time systems with proof of convergence, Neurocomputing 91 (2012) 48–55.

[29] D. Wang, D. Liu, Q. Wei, Finite-horizon neuro-optimal tracking control for a class of discrete-time nonlinear systems using adaptive dynamic programming approach, Neurocomputing 78 (2012) 14–22.

[30] W. Lin, C.I. Byrnes, H_∞ control of discrete-time nonlinear systems, IEEE Trans. Automat. Control 41 (1996) 494–510.

[31] A.J. van der Shaft, L_2-gain analysis of nonlinear systems and nonlinear state feedback H_∞ control, IEEE Trans. Automat. Control 37 (1992) 770–784.

[32] J.C. Willems, Dissipative dynamical systems. Part 1: general theory, Arch. Ration. Mech. Anal. 45 (1972) 321–351.

[33] T. Basar, P. Bernhard, H_∞ Optimal Control and Related Minimax Design Problems: A Dynamic Game Approach, second ed., Birkhäuser, Boston, 1995.

[34] T. Basar, G.J. Olsder, Dynamic Noncooperative Game Theory, second ed., SIAM, Philadelphia, 1999.

[35] A. Lanzon, Y. Feng, B.D.O. Anderson, M. Rotkowitz, Computing the positive stabilizing solution to algebraic Riccati equations with an indefinite quadratic term via a recursive method, IEEE Trans. Automat. Control 53 (2008) 2280–2291.

[36] D. Varbie, F.L. Lewis, Adaptive dynamic programming algorithm for finding online the equilibrium solution of the two-player zero-sum differential game, in: Proceedings of the International Joint Conference on Neural Networks, Barcelona, Spain, July 2010, pp. 1–8.

[37] D. Varbie, F.L. Lewis, Adaptive dynamic programming for online solution of a zero-sum differential game, J. Control Theory Appl. 9 (2011) 353–360.

[38] Y. Jiang, Z. Jiang, Approximate dynamic programming for optimal stationary control with control-dependent noise, IEEE Trans. Neural Networks 22 (2011) 2392–2398.

[39] M. Abu-Khalaf, F.L. Lewis, J. Huang, Policy iterations and the Hamilton–Jacobi–Isaacs equation for H_∞ state feedback control with input saturation, IEEE Trans. Automat. Control 51 (2006) 1989–1995.

[40] M. Abu-Khalaf, F.L. Lewis, J. Huang, Neurodynamic programming and zero-sum games for constrained control systems, IEEE Trans. Neural Networks 19 (2008) 1243–1252.

[41] Y. Feng, B.D.O. Anderson, M. Rotkowitz, A game theoretic algorithm to compute local stabilizing solutions to HJBI equations in nonlinear H_∞ control, Automatica 45 (2009) 881–888.

[42] H. Zhang, Q. Wei, D. Liu, An iterative adaptive dynamic programming method for solving a class of nonlinear zero-sum differential games, Automatica 47 (2011) 207–214.

[43] G. Toussaint, T. Basar, F. Bullo, H_∞-optimal tracking control techniques for nonlinear underactuated systems, in: Proceedings of the IEEE Conference on Decision and Control, 2000, pp. 2078–2083.

[44] K.G. Vamvoudakis, F.L. Lewis, Online solution of nonlinear two-player zero-sum games using synchronous policy iteration, Int. J. Robust Nonlinear Control 22 (2012) 1460–1483.

[45] T. Dierks, S. Jagannathan, Optimal control of affine nonlinear continuous-time systems using an online Hamilton–Jacobi–Isaacs formulation, in: IEEE Conference on Decision and Control, Atlanta, GA, December 2010, pp. 3048–3053.

[46] A. Al-Tamimi, M. Abu-Khalaf, F.L. Lewis, Adaptive critic designs for discrete-time zero-sum games with application to H_∞ control, IEEE Trans. Syst. Man Cybernet. Part B: Cybernet. 37 (2007) 240–247.

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

[48] K.H. Kim, F.L. Lewis, Model-free H_∞ control design for unknown linear discrete-time systems via Q-learning with LMI, Automatica 46 (2010) 1320–1326.

[49] Q. Wei, H. Zhang, L. Cui, Data-based optimal control for discrete-time zero-sum games of 2-D systems using adaptive critic designs, ACTA Automat. Sin. 35 (2009) 682–692.

[50] L. Cui, H. Zhang, X. Zhang, Y. Luo, Data-based adaptive critic design for discrete-time zero-sum games using output feedback, in: IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning, 2011, pp. 190–195.

[51] S. Mehraeen, T. Dierks, S. Jagannathan, M.L. Crow, Zero-sum two-player game theoretic formulation of affine nonlinear discrete-time systems using neural networks, in: Proceedings of the International Joint Conference on Neural Networks, Barcelona, Spain, July 2010, pp. 1–8.

[52] R.J. Leake, R.W. Liu, Construction of suboptimal control sequences, SIAM J. Control 5 (1967) 54–63.

[53] B. Lincoln, A. Rantzer, Relaxing dynamic programming, IEEE Trans. Automat. Control 51 (2006) 1249–1260.

[54] A. Rantzer, Relaxed dynamic programming in switching systems, IEE Proc. Control Theory 156 (2006) 567–574.

Derong Liu received the PhD degree in electrical engineering from the University of Notre Dame in 1994. He was a Staff Fellow with General Motors Research and Development Center, Warren, MI, from 1993 to 1995. He was an Assistant Professor in the Department of Electrical and Computer Engineering, Stevens Institute of Technology, Hoboken, NJ, from 1995 to 1999. He joined the University of Illinois at Chicago in 1999, and became a Full Professor of electrical and computer engineering and of computer science in 2006. He was selected for the “100 Talents Program” by the Chinese Academy of Sciences in 2008. He has published 10 books. He has been an Associate Editor of several IEEE publications. Currently, he is the Editor-in-Chief of the IEEE Transactions on Neural Networks and Learning Systems, and an Associate Editor of the IEEE Transactions on Control Systems Technology and several other journals including Neurocomputing, International Journal of Neural Systems, Neural Computing and Applications, Soft Computing, Journal of Control Science and Engineering, and Science in China Series F: Information Sciences. He was an elected AdCom member of the IEEE Computational Intelligence Society (2006–2008). He received the Faculty Early Career Development (CAREER) award from the National Science Foundation (1999), the University Scholar Award from University of Illinois (2006–2009), and the Overseas Outstanding Young Scholar Award from the National Natural Science Foundation of China (2008). He is a Fellow of the IEEE.

Hongliang Li received the BS degree in mechanical engineering and automation from Beijing University of Posts and Telecommunications in 2010. He is currently working toward the PhD degree in the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China. His research interests include neural networks, reinforcement learning, adaptive dynamic programming, game theory and multi-agent systems.

Ding Wang received the BS degree in mathematics from Zhengzhou University of Light Industry, Zhengzhou, China, the MS degree in operations research and cybernetics from Northeastern University, Shenyang, China, and the PhD degree in control theory and control engineering from Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2007, 2009, and 2012, respectively. He is currently an assistant professor with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences. His research interests include adaptive dynamic programming, neural networks, and intelligent control.