Generalized Policy Iteration Adaptive Dynamic Programming Algorithm for Optimal Tracking Control of a Class of Nonlinear Systems

Qiao Lin[1], Qinglai Wei[1], Derong Liu[2]

1. The State Key Laboratory of Management and Control for Complex Systems Institute of Automation,
Chinese Academy of Sciences, Beijing 100190, China
E-mail: linqiao2014@ia.ac.cn; qinglai.wei@ia.ac.cn

2. School of Automation and Electrical Engineering, University of Science and Technology Beijing,
Beijing 100083, China
E-mail: derong@ustb.edu.cn

Abstract: This paper deals with optimal tracking control problems for a class of discrete-time nonlinear systems using a generalized policy iteration adaptive dynamic programming (ADP) algorithm. First, by system transformation, the optimal tracking control problem is transformed into an optimal regulation problem. Then the generalized policy iteration ADP algorithm is employed to obtain the optimal tracking controller with convergence and optimality analysis. The developed algorithm uses the idea of two iteration procedures to obtain the iterative tracking control laws and the iterative value functions. Three neural networks, including model network, critic network and action network, are used to implement the developed algorithm. At last, an simulation example is given to demonstrate the effectiveness of the developed method.

Key Words: adaptive dynamic programming; affine nonlinear systems; discrete-time; generalized policy iteration; neural network; tracking control.

1 INTRODUCTION

The optimal tracking control problems have always been the key focus of control field in recent decades. Gilbert and Ha [2] used feedback linearization technique to design traditional tracking controller, which is only effective in the neighborhood of the equilibrium point. To avoid this shortcoming, an effective brain-like method called adaptive/approximate dynamic programming (ADP) was proposed [3]. The ADP algorithm can overcome the curse of dimensionality and solve the Hamilton-Jacobi-Bellman (HJB) equation forward-in-time. Therefore more and more attention has been paid to the ADP algorithm [3]− [10]. According to [4], ADP approaches were classified into four schemes: heuristic dynamic programming (HDP), dual heuristic programming (DHP), action dependent HDP (ADHDP), also known as Q-learning, and action dependent DHP (ADDHP). Policy and value iteration algorithms are primary tools in ADP to obtain the solution of the HJB equation indirectly. Zhang et al. [7] and Wang et al. [6] used the value iteration algorithm to solve optimal tracking control problems for nonlinear systems. Bhasin et al. [9] proposed an online actor-critic-identifier architecture to approximate the optimal control law for uncertain nonlinear systems by policy iteration algorithms. Abu-Khalaf and Lewis [8] proposed a policy iteration algorithm for continuous-time nonlinear systems with control constraints.

However, Sutton and Barto [5] pointed out that almost all reinforcement learning and ADP methods could be described as generalized policy iteration algorithm. So it's important to investigate the generalized policy iteration algorithm for the development of ADP. Compared with other iteration algorithms, the present generalized policy iteration ADP algorithm, which is a general idea of interacting policy and value iteration ADP algorithm, has two iteration procedures, including i-iteration and j-iteration. Furthermore, as a new method to solve the optimal problems, the generalized policy iteration algorithm has drawn more and more researchers' attention. The generalized policy iteration algorithm for continuous-time systems was discussed in [11] and [12]. And the analysis of the generalized policy iteration algorithm for discrete-time systems was studied in [13] and [14]. To the best of our knowledge, however, there is no study on the optimal tracking controller using generalized policy iteration ADP algorithm. So in this paper, we will use the new algorithm to design optimal tracking controller for a class of discrete-time nonlinear systems. First, we will transform the tracking problem into an optimal regulation problem. Second, the detailed iteration procedure of the generalized policy iteration algorithm for discrete-time is presented. Next, the convergence criteria of the generalized policy iteration algorithm is proven. Neural networks (NNs) are introduced to implement the gen-

This work was supported in part by the National Natural Science Foundation of China under Grants 61233001, 61273140, 61304086, 61374105 and 61533017, and by Beijing Natural Science Foundation under Grant 4132078.

978-1-4673-9714-8/16/$31.00 ©2016 IEEE

eralized policy iteration algorithm. Finally, one example is given to confirm the effectiveness of the generalized policy iteration ADP algorithm.

2 PROBLEM STATEMENT

Consider the discrete-time nonlinear systems described as:

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

where $x(k) \in \mathbb{R}^n$, $f(x(k)) \in \mathbb{R}^n$, $g(x(k)) \in \mathbb{R}^{n\times m}$ and the input $u(x(k)) \in \mathbb{R}^m$. Here, we suppose that the system is strongly controllable on $\Omega \subset \mathbb{R}^n$, and the generalized inverse of $g(\cdot)$ exists. For optimal tracking control problem, the control objective is to find an optimal control law $u^*(x(k))$, so as to make the nonlinear systems (1) track the specified desired trajectory $\eta(k) \in \mathbb{R}^n$, where we suppose $\eta(k)$ satisfies $\eta(k+1) = \Gamma(\eta(k))$. In the following part, for simplicity, $u(x(k))$ is replaced by $u(k)$.
The tracking error is defined as

$$e(k) = x(k) - \eta(k). \tag{2}$$

Inspired by the study of [15], we can define

$$v(k) = u(k) - u_\eta(k), \tag{3}$$

where

$$u_\eta(k) = g^{-1}(\eta(k))(\eta(k+1) - f(\eta(k))). \tag{4}$$

Here, $u_\eta(k)$, which denotes the expected control, is introduced for analytical purpose. By substituting (2), (3) and (4) into (1), we can obtain a new system as follows:

$$\begin{aligned} e(k+1) =& f(e(k)+\eta(k)) + g(e(k)+\eta(k))g^{-1}(\eta(k)) \\ & \times (\eta(k+1) - f(\eta(k))) - \eta(k+1) + g(e(k) \\ & + \eta(k))v(k). \end{aligned} \tag{5}$$

For simplicity, the new system (5) can be represented as

$$e(k+1) = F(e(k), v(k)), \tag{6}$$

where $e(k)$ is the state vector and $v(k)$ is the control vector. Now let $\underline{v}(k) = \{v(k), v(k+1), v(k+2), \ldots\}$ be an arbitrary sequence of controls from k to ∞. The performance index function for initial state $e(0)$ is defined as

$$J(e(0), \underline{v}(0)) = \sum_{k=0}^{\infty} \{e^{\mathsf{T}}(k)Qe(k) + v^{\mathsf{T}}(k)Rv(k)\}, \tag{7}$$

where $Q \in \mathbb{R}^{n\times n}$, $R \in \mathbb{R}^{m\times m}$ are positive definite matrices and $\underline{v}(0) = \{v(0), v(1), v(2), \ldots\}$. Then, let the utility function U satisfy

$$U(e(k), v(k)) = e^{\mathsf{T}}(k)Qe(k) + v^{\mathsf{T}}(k)Rv(k). \tag{8}$$

In this sense, the nonlinear tracking problem is transformed into a regular optimal control problem. So the goal of this paper is not only to design the optimal control law $v^*(k)$ which makes $x(k)$ track the desired trajectory $\eta(k)$, but also minimizes the performance index function (7). Furthermore, the sequence of controls $\underline{v}(k)$ is a function of $e(k)$, hence the performance index $J(e(k), \underline{v}(k))$ can be rewritten as

$$J(e(k)) = \sum_{i=k}^{\infty} U(e(i), v(i)). \tag{9}$$

According to Bellmans optimality principle, the optimal performance index function

$$\begin{aligned} J^*(e(k)) &= \min J(e(k)) \\ &= \min_{v(k), v(k+1), \ldots, v(\infty)} \sum_{i=k}^{\infty} U(e(i), v(i)) \end{aligned} \tag{10}$$

can be rewritten as

$$\begin{aligned} J^*(e(k)) = \min_{v(k)} \Bigg\{ & U(e(k), v(k)) \\ & + \min_{v(k+1), \ldots, v(\infty)} \sum_{i=k+1}^{\infty} U(e(i), v(i)) \Bigg\}. \end{aligned} \tag{11}$$

In other words, $J^*(e(k))$ satisfies the discrete-time HJB (DTHJB) equation

$$J^*(e(k)) = \min_{v(k)} \{U(e(k), v(k)) + J^*(e(k+1))\}. \tag{12}$$

Therefore, the optimal control law can be expressed as

$$v^*(k) = \arg\min_{v(k)} \{U(e(k), v(k)) + J^*(e(k+1))\}. \tag{13}$$

From (8) and (13), we can obtain

$$v^*(k) = -\frac{1}{2} R^{-1} g^{\mathsf{T}}(e(k) + \eta(k)) \frac{\partial J^*(e(k+1))}{\partial e(k+1)}. \tag{14}$$

By using (13) and (14), the DTHJB equation can be rewritten as

$$\begin{aligned} J^*(e(k)) =& U(e(k), v^*(k)) + J^*(e(k+1)) \\ =& e^{\mathsf{T}}(k)Qe(k) + \frac{1}{4}\left(\frac{\partial J^*(e(k+1))}{\partial e(k+1)}\right)^{\mathsf{T}} \\ & \times g(e(k)+\eta(k))R^{-1}g^{\mathsf{T}}(e(k)+\eta(k)) \\ & \times \frac{\partial J^*(e(k+1))}{\partial e(k+1)} + J^*(e(k+1)). \end{aligned} \tag{15}$$

Generally, the partial differential of $J^*(e(k+1))$ is difficult to obtain. Therefore in order to solve the DTHJB equation, we will design a novel algorithm to approximate the performance index function in the following part.

3 OPTIMAL TRACKING CONTROL BASED ON GENERALIZED POLICY ITERATION ADP ALGORITHM

3.1 Derivation of the Generalized Policy Iteration ADP Algorithm

In this subsection, we present the details of generalized policy iterative ADP algorithm. First of all, we start with an initial admissible control law $\hat{v}_0(k)$, and let $V_0(e(k))$ satisfy the generalized HJB (GHJB) equation:

$$\begin{aligned} V_0(e(k)) &= U(e(k), \hat{v}_0(k)) + V_0(e(k+1)) \\ &= U(e(k), \hat{v}_0(k)) + V_0(F(e(k), \hat{v}_0(k)). \end{aligned} \tag{16}$$

Then, for $i = 1$, the iterative control law is obtained by

$$\hat{v}_1(k) = \arg\min_{v(k)} \{U(e(k), v(k)) + V_0(F(e(k), v(k)))\}. \tag{17}$$

Let $\{N_1, N_2, N_3, \ldots\}$ be a sequence, where $N_i \geq 0$, $i = 1, 2, 3, \ldots$, are non-negative integers. And let j_1 increase from 0 to N_1, then the value function is updated by

$$V_{1,j_1+1}(e(k)) = U(e(k), \hat{v}_1(k)) + V_{1,j_1}(F(e(k), \hat{v}_1(k))), \tag{18}$$

where

$$\begin{aligned} V_{1,0}(e(k)) &= \min_{v(k)} \{U(e(k), v(k)) + V_0(e(k+1))\} \\ &= U(e(k), \hat{v}_1(k)) + V_0(F(e(k), \hat{v}_1(k))). \end{aligned} \tag{19}$$

We define the iterative value function as

$$V_1(e(k)) = V_{1,N_1}(e(k)). \tag{20}$$

For $i = 2, 3, 4, \ldots$, we can implement the generalized policy iteration ADP algorithm by the following two iteration procedures.

1) i-iteration

$$\hat{v}_i(k) = \arg\min_{v(k)} \{U(e(k), v(k)) + V_{i-1}(F(e(k), v(k)))\}. \tag{21}$$

2) j-iteration

$$V_{i,j_i+1}(e(k)) = U(e(k), \hat{v}_i(k)) + V_{i,j_i}(F(e(k), \hat{v}_i(k))), \tag{22}$$

where the iteration index j_i increases from 0 to N_i,

$$\begin{aligned} V_{i,0}(e(k)) &= \min_{v(k)} \{U(e(k), v(k)) + V_{i-1}(e(k+1))\} \\ &= U(e(k), \hat{v}_i(k)) + V_{i-1}(F(e(k), \hat{v}_i(k))) \end{aligned} \tag{23}$$

and the iteration value function is given as

$$V_i(e(k)) = V_{i,N_i}(e(k)). \tag{24}$$

In fact, each j-iteration tries to solve the following generalizd HJB (GHJB) equation:

$$V_{i,j_i}(e(k)) = U(e(k), \hat{v}_i(k)) + V_{i,j_i}(F(e(k), \hat{v}_i(k))). \tag{25}$$

3.2 Convergence Analysis of the Generalized Policy Iteration ADP Algorithm

Theorem 1 Let the iteration control law $\hat{v}_i(k)$ and the iteration value function $V_{i,j_i}(e(k))$ be obtained by (16)–(25). Then, for $i = 1, 2, \ldots, j_i = 0, 1, 2, \ldots, N_i$ and for all $e(k) \in \Omega_e$, the iteration value function $V_{i,j_i}(e(k))$ is a monotonically non-increasing sequence satisfying:

$$V_{i,j_i+1}(e(k)) \leq V_{i,j_i}(e(k)) \tag{26}$$

and

$$V_{i+1,j_{(i+1)}}(e(k)) \leq V_{i,j_i}(e(k)) \tag{27}$$

where $0 \leq j_i \leq N_i$ and $0 \leq j_{i+1} \leq N_{i+1}$.

Proof. The inequality (26) can be proved in two steps by mathematical induction.

Step1: Let $i = 1$. According to (16) and (23), we can obtain

$$\begin{aligned} V_{1,0}(e(k)) &= U(e(k), \hat{v}_1(k)) + V_0(F(e(k), \hat{v}_1(k))) \\ &= \min_{v(k)} \{U(e(k), v(k)) + V_0(F(e(k), v(k)\} \\ &\leq U(e(k), \hat{v}_0(k)) + V_0(F(e(k), \hat{v}_0(k))) \\ &= V_0(e(k)). \end{aligned} \tag{28}$$

Then, using (22) and (28), for $j_1 = 0$, we have

$$\begin{aligned} V_{1,1}(e(k)) &= U(e(k), \hat{v}_1(k)) + V_{1,0}(F(e(k), \hat{v}_1(k))) \\ &\leq U(e(k), \hat{v}_1(k)) + V_0(F(e(k), \hat{v}_1(k))) \\ &= V_{1,0}(e(k)). \end{aligned} \tag{29}$$

By (22) and (29), for $j_1 = 1$, we can obtain that

$$\begin{aligned} V_{1,2}(e(k)) &= U(e(k), \hat{v}_1(k)) + V_{1,1}(F(e(k), \hat{v}_1(k))) \\ &\leq U(e(k), \hat{v}_1(k)) + V_{1,0}(F(e(k), \hat{v}_1(k))) \\ &= V_{1,1}(e(k)). \end{aligned} \tag{30}$$

For $j_1 = s$, where s is a positive integer and $1 < s \leq N_1$, then we have

$$\begin{aligned} V_{1,s+1}(e(k)) &= U(e(k), \hat{v}_1(k)) + V_{1,s}(F(e(k), \hat{v}_1(k))) \\ &\leq U(e(k), \hat{v}_1(k)) + V_{1,s-1}(F(e(k), \hat{v}_1(k))) \\ &= V_{1,s}(e(k)). \end{aligned} \tag{31}$$

Therefore (26) holds for $i = 1$.

Step 2: Assuming that (26) holds for $i = m$, we can get

$$V_{m,j_m+1}(e(k)) \leq V_{m,j_m}(e(k)). \tag{32}$$

Hence, according to (16) and (23), for $i = m + 1$

$$\begin{aligned} &V_{m+1,0}(e(k)) \\ &= U(e(k), \hat{v}_{m+1}(k)) + V_m(F(e(k), \hat{v}_{m+1}(k))) \\ &= \min_{v(k)} \{U(e(k), v(k)) + V_m(F(e(k), v(k)\} \\ &\leq U(e(k), \hat{v}_m(k)) + V_m(F(e(k), \hat{v}_m(k))) \\ &= V_{m,N_m+1}(e(k)) \\ &\leq V_{m,N_m}(e(k)) \\ &= V_m(e(k)). \end{aligned} \tag{33}$$

Next, by observing (22) and (28), for $j_{m+1} = 0$, we have

$$\begin{aligned} &V_{m+1,1}(e(k)) \\ &= U(e(k), \hat{v}_{m+1}(k)) + V_{m+1,0}(F(e(k), \hat{v}_{m+1}(k))) \\ &\leq U(e(k), \hat{v}_{m+1}(k)) + V_m(F(e(k), \hat{v}_{m+1}(k))) \\ &= V_{m+1,0}(e(k)). \end{aligned} \tag{34}$$

From (22) and (34), for $j_{m+1} = 1$

$$\begin{aligned}&V_{m+1,2}(e(k))\\&= U(e(k), \hat{v}_{m+1}(k)) + V_{m+1,1}(F(e(k), \hat{v}_{m+1}(k)))\\&\le U(e(k), \hat{v}_{m+1}(k)) + V_{m+1,0}(F(e(k), \hat{v}_{m+1}(k)))\\&= V_{m+1,1}(e(k)). \qquad (35)\end{aligned}$$

Using the same method as (31), for $j_{m+1} = q$, where q is positive integer and $1 < q \le N_{m+1}$,

$$\begin{aligned}&V_{m+1,q+1}(e(k))\\&= U(e(k), \hat{v}_{m+1}(k)) + V_{m+1,q}(F(e(k), \hat{v}_{m+1}(k)))\\&\le U(e(k), \hat{v}_{m+1}(k)) + V_{m+1,q-1}(F(e(k), \hat{v}_{m+1}(k)))\\&= V_{m+1,q}(e(k)). \qquad (36)\end{aligned}$$

So (26) holds for $i = m + 1$. The mathematical induction is completed.

In the following part, inequality (27) will be proven. Let $0 \le j_{i+1} \le N_{i+1}$. Then according to (24)–(26), we can get

$$\begin{aligned}V_{i+1}(e(k)) &= V_{i+1,N_{i+1}}(e(k)) \le V_{i+1,j_{i+1}}(e(k))\\&\le V_{i+1,0}(e(k)) \le V_i(e(k)). \qquad (37)\end{aligned}$$

Therefore the proof of (27) is completed.

From the inequalities (26) and (27), we can conclude that the iterative value function $V_{i,j_i}(e(k))$ is a monotonically nonincreasing sequence.

Theorem 2 For $i = 0, 1, 2, \ldots,$ and any $N_i \ge 0$, the iterative value function $V_{i,j_i}(e(k))$, which is obtained by (22), converges to the optimal performance index function $J^*(e(k))$, i.e.,

$$\lim_{i\to\infty} V_{i,j_i}(e(k)) = J^*(e(k)). \qquad (38)$$

Proof. We define $\{V_{i,j_i}(e(k))\} = \{V_0(e(k)), V_{1,0}(e(k)), V_{1,1}(e(k)), \ldots, V_{1,N_1}(e(k)), V_1(e(k)), V_{2,0}(e(k)), V_{2,1}(e(k)), \ldots, V_{2,N_2}(e(k)), \ldots\}$. Then, $\{V_i(e(k))\}$ is selected as a subsequence of $\{V_{i,j_i}(e(k))\}$, i.e., $\{V_i(e(k))\} = \{V_0(e(k)), V_1(e(k)), V_2(e(k)), \ldots\}$. Apostol [1] pointed that the sequence $\{V_{i,j_i}(e(k))\}$ and its subsequence $\{V_i(e(k))\}$ had the same limit, i.e.,

$$\lim_{i\to\infty} V_{i,j_i}(e(k)) = \lim_{i\to\infty} V_i(e(k)). \qquad (39)$$

Thus, we can choose to prove the following equation for simplicity,

$$\lim_{i\to\infty} V_i(e(k)) = J^*(e(k)). \qquad (40)$$

First, we define the limit of the iterative value function $\{V_i(e(k))\}$, i.e., $V_\infty(e(k)) = \lim_{i\to\infty} V_i(e(k))$. According to Theorem 1 and (23), we have

$$\begin{aligned}V_i(e(k)) &\le V_{i,0}(e(k))\\&= U(e(k), \hat{v}_i(k)) + V_{i-1}(F(e(k), \hat{v}_i(k)))\\&= \min_{v(k)} \{U(e(k), v(k)) + V_{i-1}(F(e(k), v(k)))\}.\end{aligned} \qquad (41)$$

Then, we can get

$$\begin{aligned}V_\infty(e(k)) &= \lim_{i\to\infty} V_i(e(k))\\&\le V_i(e(k))\\&\le \min_{v(k)} \{U(e(k), v(k)) + V_{i-1}(F(e(k), v(k)))\}.\end{aligned} \qquad (42)$$

Hence, letting $i \to \infty$, we can obtain

$$V_\infty(e(k)) \le \min_{v(k)} \{U(e(k), v(k)) + V_\infty(F(e(k), v(k)))\}. \qquad (43)$$

On the other hand, let $\gamma > 0$ be an arbitrary positive constant. From Theorem 1, we can get that $V_i(e(k))$ is nonincreasing sequence, so there exists a positive integer π such that

$$V_\pi(e(k)) - \gamma \le V_\infty(e(k)) \le V_\pi(e(k)). \qquad (44)$$

Thus, substituting (25) into (44), we can obtain

$$\begin{aligned}&V_\infty(e(k)) \qquad (45)\\&\quad\ge U(e(k), \hat{v}_\pi(k)) + V_\pi(F(e(k), \hat{v}_\pi(k))) - \gamma\\&\quad\ge U(e(k), \hat{v}_\pi(k)) + V_\infty(F(e(k), \hat{v}_\pi(k))) - \gamma\\&\quad= \min_{v(k)} \{U(e(k), v(k)) + V_\infty(F(e(k), v(k)))\} - \gamma,\end{aligned} \qquad (46)$$

which reveals that

$$V_\infty(e(k)) \ge \min_{v(k)} \{U(e(k), v(k)) + V_\infty(F(e(k), v(k)))\}, \qquad (47)$$

because of the arbitrariness of γ.

Combining (43) and (47), we can conclude that

$$V_\infty(e(k)) = \min_{v(k)} \{U(e(k), v(k)) + V_\infty(F(e(k), v(k)))\}. \qquad (48)$$

Second, on one hand, according to (9) and (12), for any $\xi > 0$, we can find an admissible control sequence $\underline{\omega}(k)$ that satisfies

$$J(e(k), \underline{\omega}(k)) \le J^*(e(k)) + \xi. \qquad (49)$$

Now, we suppose that the length of the control sequence $\underline{\omega}(k)$ is θ. Then using (7), (23) and Theorem 1, we can obtain

$$\begin{aligned}V_\infty(e(k)) &\le V_\theta(e(k))\\&\le \min_{v(k)} \{U(e(k), v(k)) + V_{\theta-1}(F(e(k), v(k)))\}\\&\le J(e(k), \underline{\omega}(k)). \qquad (50)\end{aligned}$$

Combining (49) with (50), we can get

$$V_\infty(e(k)) \le J^*(e(k)) + \xi, \qquad (51)$$

where ξ is arbitrary. Then, we have

$$V_\infty(e(k)) \le J^*(e(k)). \qquad (52)$$

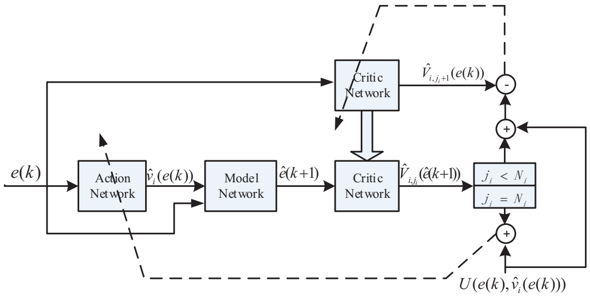

Figure 1: Structure diagram of the algorithm

On the other hand, from the definition of $J^*(e(k))$ in (10), for $i = 0, 1, 2, ..., V_i(e(k)) \geq J^*(e(k))$ holds for all $e(k) \in \Omega_e$. Letting $i \to \infty$, we can acquire $V_\infty(e(k)) \geq J^*(e(k))$. Therefore, we have (39) holds.

From Theorems 1–2, we can conclude that the value function sequence $\{V_{i,j_i}(e(k))\}$ is a non-increasing sequence, and converges to the optimal performance index function $J^*(e(k))$, i.e., $V_{i,j_i} \to J^*$ as $i \to \infty$. And on the basis of the definition of $v^*(k)$ in (13), it's not difficult to find that when $V_{i,j_i} \to J^*$, $\hat{v}_i \to v^*$ also holds as $i \to \infty$.

Remark 1 After the optimal control law $v^*(k)$ for system (6) is derived, we can obtain the optimal tracking control $u^*(k)$ for original system (1) by $u^*(k) = v^*(k) + g^{-1}(\eta(k))(\eta(k+1) - f(\eta(k)))$.

4 NN IMPLEMENTATION OF THE GENERALIZED POLICY ITERATION ADP ALGORITHM

In this paper, three NNs, called critic network, model network, and action network respectively, are used to implement the algorithm and approximate $\hat{v}_i(k)$ and $V_{i,j_i}(e(k))$. All the NNs are chosen as three-layer back-propagation (BP) networks. The structure diagram of the generalized policy iterative ADP algorithm is shown in Fig. 1. The weights are updated using the gradient-based adaption rule, which can be referred to [16].

5 SIMULATION STUDY

Consider the following discrete-time nonlinear system:

$$x(k+1) = f(x(k)) + g(x(k))u(k), \tag{53}$$

where

$$x(k) = [x_1(k), x_2(k)]^\mathsf{T},$$
$$u(k) = [u_1(k), u_2(k)]^\mathsf{T},$$
$$f(x(k)) = \begin{bmatrix} x_1(k) + 0.1x_2(k) \\ -0.1x_1(k) + 1.1x_2(k) - 0.1x_2(k)x_1^2(k) \end{bmatrix},$$
$$g(x(k)) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Let the initial state be $x(0) = [0.7, -1]^\mathsf{T}$ and the desired trajectory is specified as

$$\eta(k) = \left[-\frac{1}{\pi}\cos(0.1\pi k), \sin(0.1\pi k)\right]^\mathsf{T}.$$

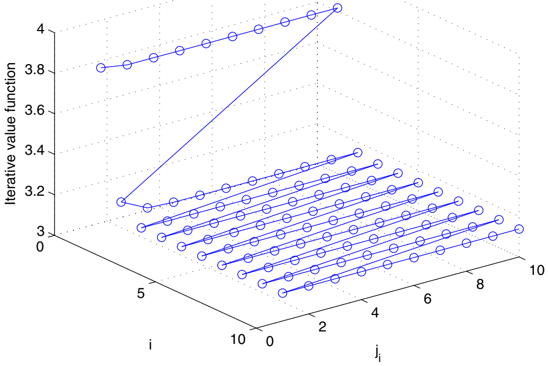

Figure 2: V_{i,j_i} for $k = 0$

Then, let the performance index function be expressed by (10). The parameters of the utility function are chosen as $Q = I_1, R = I_2$, where I_1 and I_2 denote the identity matrix with suitable dimensions. The error bound of the iterative ADP is chosen as $\varepsilon = 10^{-5}$. NNs are used to carry out the generalized policy iteration ADP algorithm. The model network, critic network and action network are chosen as three-layer BP NNs with the structures $4{-}8{-}2$, $2{-}8{-}1$, $2{-}8{-}2$, respectively. And all the initial weights are chosen from $[-1, 1]$ randomly. It should be noted that the model network should be trained first. The model network is trained under the learning rate $\alpha_m = 0.15$. Then, for each iteration step, the critic network and action network are trained for 1500 training steps using the learning rate of $\alpha_c = \alpha_a = 0.05$, so that the NN training error become less than ε.

We let iteration index $i = 10$ and choose the iteration sequence $\{N_i\} = 10$. The changing curve of V_{i,j_i} for $k = 0$ is shown in Fig. 2 and the trajectory of the iterative value function V_i for the entire state space is shown as Fig. 3. From Fig. 2 and Fig. 3, we can get that both the value function $V_{i,j_i}(e(k))$ and the subsequence $V_i(e(k))$ are monotonically nonincreasing sequence, where "In" indicates initial iteration and "Lm" means limiting iteration.

Then, we compute the tracking control law using (21) and apply it to the system (53) for 30 time steps. The tracking control curves are shown in Fig. 4. And the Fig. 5 shows the tracking error trajectory. From Fig. 5, we can see that the tracking errors become minimum, which shows that the control system has already tracked the reference trajectories within the allowable error. These simulation results confirm the excellent performance of the generalized policy iteration algorithm for optimal tracking control systems.

6 CONCLUSION

An effective generalized policy iteration ADP algorithm is proposed in this paper to solve the optimal tracking control problem. It has been proven that the iterative value functions are monotonically non-increasing and convergent to the optimum. NNs, which can approximate the nonlinear system, control law, and value function, are introduced to

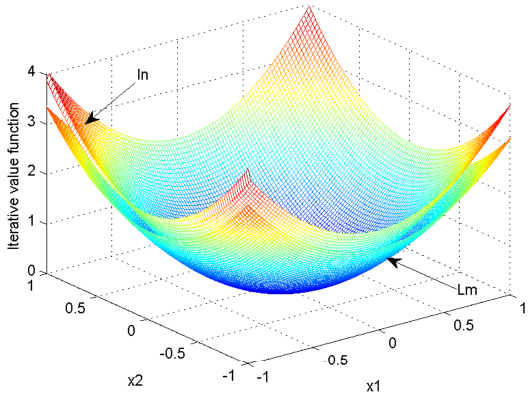

Figure 3: V_i for the entire state space

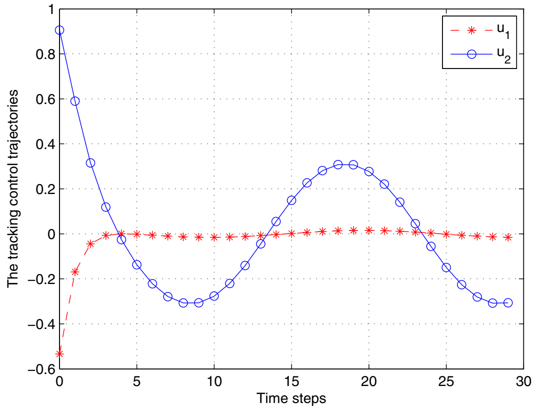

Figure 4: The tracking control trajectories u

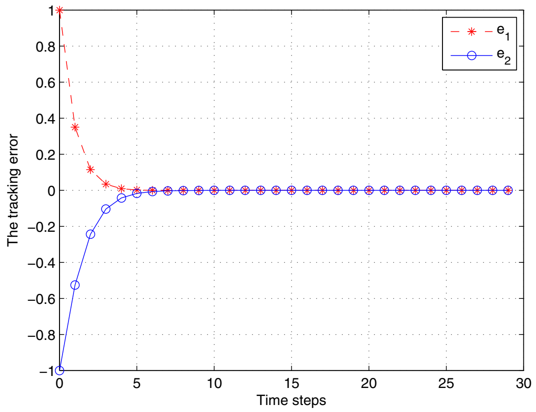

Figure 5: The tracking error e

implement the present algorithm. Finally, the simulation example is given to verify the effectiveness of the novel algorithm for tracking control systems.

REFERENCES

[1] T. M. Apostol, Mathematical Analysis (2nd ed), Boston, MA, USA: Addison-Wesley Press, 1974.

[2] I. J. Ha, and E. G. Gilbert, Robust tracking in nonlinear systems, IEEE Transactions on Automatic Control, Vol.32, No.9, 763–771, 1987.

[3] P. J. Werbos, W. T. Miller, R. S. Sutton, A menu of designs for reinforcement learning over time, Neural Networks for Control, Cambridge, MA, USA: MIT Press, 1991.

[4] D. V. Prokhorov and D. C. Wunsch, Adaptive critic designs, IEEE transactions on Neural Networks, Vol.8, No.5, 997–1007, 1997.

[5] R. S. Sutton and A. G. Barto, Reinforcement Learning: An Introduction, MA: MIT Press, 1998.

[6] 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, Vol.78, No.1, 14–22, 2012.

[7] 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 Transactions on Systems, Man, and Cybernetics, Vol.38, No.4, 937–942, 2008.

[8] M. Abu-Khalaf and F. L. Lewis, Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach, Automatica, Vol.41, No.5, 779–791, 2005.

[9] S. Bhasin, R. Kamalapurkar, M. Johnson, K. G. Vamvoudakis, F. L. Lewis, W. E. Dixon, A novel actor-critic-identifier architecture for approximate optimal control of uncertain nonlinear systems, Automatica, Vol.49, No.1, 82–92, 2013.

[10] B. Luo, H. Wu, H. Li, Adaptive optimal control of highly dissipative nonlinear spatially distributed processes with neuro-dynamic programming, IEEE Transactions on Neural Networks and Learning Systems, Vol.26, No.4, 684–696, 2015.

[11] D. Vrabie and F. L. Lewis, Generalized policy iteration for continuoustime systems," Proceedings of International Joint Conference on Neural Networks, Atlanta, Georgia, USA, 3224–3231, 2009.

[12] D. Vrabie, K. Vamvoudakis, F. L. Lewis, Adaptive optimal controllers based on generalized policy iteration in a continuous-time framework, in 17th Mediterranean Conference on Control & Automation, Thessaloniki, Greece, 1402–1409, 2009.

[13] Q. Wei, D. Liu, X. Yang, Infinite Horizon Self-Learning Optimal Control of Nonaffine Discrete-Time Nonlinear Systems, IEEE Transactions on Neural Networks and Learning Systems, Vol.26, No.4, 866–879, 2015.

[14] D. Liu, Q. Wei, P. Yan, Generalized Policy Iteration Adaptive Dynamic Programming for Discrete-Time Nonlinear Systems, IEEE Transactions on Systems, Man, and Cybernetics: Systems, accepted.

[15] Y. M. Park, M. S. Choi, K. Y. Lee, An optimal tracking neuro-controller for nonlinear dynamic systems, IEEE Transactions on Neural Networks Vol.7, 1099–1110, 1996.

[16] J. Si and Y. T. Wang, On-line learning control by association and reinforcement, IEEE Transactions on Neural Networks, Vol.12, No.2, 264–276, 2001.