

Adaptive Dynamic Programming for Finite-Horizon Optimal Tracking Control of a Class of Nonlinear Systems

WANG Ding, LIU Derong, WEI Qinglai

Key Laboratory of Complex Systems and Intelligence Science, Institute of Automation, Chinese Academy of Sciences,
 Beijing 100190, P. R. China
 E-mail: {ding.wang, derong.liu, qinglai.wei}@ia.ac.cn

Abstract: This paper deals with the finite-horizon optimal tracking control for a class of discrete-time nonlinear systems using the iterative adaptive dynamic programming (ADP) algorithm. First, the optimal tracking problem is converted into designing a finite-horizon optimal regulator for the tracking error dynamics. Then, with convergence analysis in terms of cost function and control law, the iterative ADP algorithm via heuristic dynamic programming (HDP) technique is introduced to obtain the finite-horizon optimal tracking controller which makes the cost function close to its optimal value within an ε -error bound. Moreover, three neural networks are used to implement the algorithm, which aims at approximating the cost function, the control law, and the error dynamics, respectively. At last, an example is included to demonstrate the effectiveness of the proposed approach.

Key Words: Adaptive critic designs, Adaptive dynamic programming, Approximate dynamic programming, Finite-horizon optimal tracking control, Learning control, Neural control

1 Introduction

The optimal tracking problems have been the focus of control systems community for several decades since they are usually encountered in real world systems. For finite-horizon tracking control problems, the system must be tracked to a reference trajectory in a finite duration of time. In this paper, we will solve the problems through the framework of Hamilton-Jacobi-Bellman (HJB) equation from optimal control theory. Unlike the open-loop optimal controller design for nonlinear systems, however, for closed-loop optimal feedback control, it is difficult to solve directly the time-varying HJB equation which involves solving either nonlinear partial difference or differential equations. Though dynamic programming (DP) has been an useful technique in solving optimal control problems for many years, it is often computationally untenable to run it to obtain the optimal solution due to the “curse of dimensionality”.

With strong capabilities of self-learning and adaptivity, artificial neural networks (ANN or NN) are an effective tool to implement intelligent control. Besides, it has been used for universal function approximation in adaptive/approximate dynamic programming (ADP) algorithms, which were proposed in [1] as a method to solve optimal control problems forward-in-time. There are several synonyms used for ADP including “adaptive dynamic programming”, “approximate dynamic programming”, “neuro-dynamic programming”, “neural dynamic programming”, “adaptive critic designs” and “reinforcement learning”. As an effective intelligent control method, ADP and the related research have gained much attention from researchers [1–11]. According to [1] and [6], ADP approaches were classified into several main schemes: heuristic dynamic programming (HDP), action-dependent HDP (ADHDP), also known as Q-learning, dual heuristic dynamic programming (DHP), ADDHP, globalized DHP (GDHP), and ADGDHP.

Recently, various new strategies basing on NN were pro-

posed to deal with the tracking control problems [9–12]. Dierks and Jagannathan [11] utilized neural dynamic programming to solve the HJB equation forward-in-time for optimal tracking control of affine nonlinear systems. However, there is still no result to solve the finite-horizon optimal tracking control problems for discrete-time nonlinear systems based on iterative ADP algorithm via HDP technique (iterative HDP algorithm for brief). In this paper, we will provide an iterative ADP algorithm to find the finite-horizon near-optimal tracking controller for a class of discrete-time nonlinear systems.

2 Problem Statement

In this paper, we will study the discrete-time nonlinear system given by

$$x_{k+1} = f(x_k) + g(x_k)u_p(x_k) \quad (1)$$

where $x_k \in \mathbb{R}^n$ is the state and $u_p(x_k) \in \mathbb{R}^m$ is the control vector, $f(\cdot)$ and $g(\cdot)$ are differentiable in their argument with $f(0) = 0$ and $g(0) = 0$. Assume that $f + gu_p$ is Lipschitz continuous on a set Ω in \mathbb{R}^n containing the origin, and that the system (1) is controllable in the sense that there exists a continuous control on Ω that asymptotically stabilizes the system. In the following part, $u_p(x_k)$ is denoted by u_{pk} for simplicity.

The objective for optimal tracking control problem is to find the optimal control law u_{pk}^* , so as to make the nonlinear system in (1) to track a reference (or desired) trajectory r_k in an optimal manner. Here, we assume that the reference trajectory r_k satisfies

$$r_{k+1} = \phi(r_k) \quad (2)$$

where $r_k \in \mathbb{R}^n$ and $\phi(r_k) \in \mathbb{R}^n$. Then, we define the tracking error as

$$e_k = x_k - r_k. \quad (3)$$

Inspired by the work of [10–12], we define the steady control corresponding to the reference trajectory r_k as

$$u_{dk} = g^{-1}(r_k)(\phi(r_k) - f(r_k)) \quad (4)$$

This work was supported in part by the NSFC under grants 60904037, 60921061, and 61034002, and by Beijing Natural Science Foundation under grant 4102061.

where $g^{-1}(r_k)g(r_k) = I_m$ and I_m is an $m \times m$ identity matrix.

By denoting

$$u_k = u_{pk} - u_{dk} \quad (5)$$

and using (1)–(4), we obtain

$$\begin{cases} e_{k+1} &= f(e_k + r_k) + g(e_k + r_k)g^{-1}(r_k)(\phi(r_k) \\ &\quad - f(r_k)) - \phi(r_k) + g(e_k + r_k)u_k \\ r_{k+1} &= \phi(r_k) \end{cases} \quad (6)$$

as the new system. Note that in system (6), e_k and r_k are regarded as the system variables while u_k is seen as system input. The second equation of system (6) only gives the evolution of the reference trajectory which is not affected by the system input. It will be used in the study of the first equation of system (6), which for simplicity, can be rewritten as

$$e_{k+1} = F(e_k, u_k). \quad (7)$$

Now, let e_0 be an initial state of system (7) and define $\underline{u}_0^{N-1} = (u_0, u_1, \dots, u_{N-1})$ be a control sequence with which the system (7) gives a trajectory starting from e_0 : e_1, e_2, \dots, e_N . We call the number of elements in the control sequence \underline{u}_0^{N-1} the length of \underline{u}_0^{N-1} and denote it as $|\underline{u}_0^{N-1}|$. Then, $|\underline{u}_0^{N-1}| = N$. The final state under the control sequence \underline{u}_0^{N-1} can be denoted as $e^{(f)}(e_0, \underline{u}_0^{N-1}) = e_N$.

Let $\underline{u}_k^{N-1} = (u_k, u_{k+1}, \dots, u_{N-1})$ be the control sequence starting at k with length $N - k$. For finite-horizon optimal tracking control problem, it is desired to find the control sequence which minimizes the following cost function

$$J(e_k, \underline{u}_k^{N-1}) = \sum_{i=k}^{N-1} U(e_i, u_i) \quad (8)$$

where U is the utility function, $U(0, 0) = 0$, $U(e_i, u_i) \geq 0$ for $\forall e_i, u_i$. In this paper, the utility function is chosen as the quadratic form given by $U(e_i, u_i) = e_i^T Q e_i + u_i^T R u_i$. This quadratic cost function can not only force the system state to follow the reference trajectory, but also force the system input to be close to the steady value in maintaining the state to its reference value. In fact, it can also be expressed as

$$U(e_i, u_i) = \begin{bmatrix} e_i^T & r_i^T \end{bmatrix} \begin{bmatrix} Q & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} e_i \\ r_i \end{bmatrix} + u_i^T R u_i$$

when considered from the angle of system (6).

In this sense, the nonlinear tracking problem is converted into a regulation problem and the finite-horizon cost function for tracking is written in terms of e_k and u_k . Then, the problem of solving the finite-horizon optimal tracking control law u_p^* for system (1) is transformed into seeking the finite-horizon optimal control law u^* for system (7) with respect to (8). As a result, we will focus on how to design u^* in the following sections.

For finite-horizon optimal control problems, the designed feedback control must be finite-horizon admissible, which means it must not only stabilize the controlled system on Ω within finite number of time steps but also guarantee the cost function is finite.

Definition 1 A control sequence \underline{u}_k^{N-1} is said to be finite-horizon admissible for a state $e_k \in \mathbb{R}^n$ with respect to (8) on Ω if \underline{u}_k^{N-1} is continuous on a compact set $\Omega \in \mathbb{R}^m$, $u(0) = 0$, $e^{(f)}(e_k, \underline{u}_k^{N-1}) = 0$ and $J(e_k, \underline{u}_k^{N-1})$ is finite.

Let $\mathfrak{A}_{e_k} = \{\underline{u}_k : e^{(f)}(e_k, \underline{u}_k) = 0\}$ be the set of all finite-horizon admissible control sequences of e_k and

$$\mathfrak{A}_{e_k}^{(i)} = \left\{ \underline{u}_k^{k+i-1} : e^{(f)}(e_k, \underline{u}_k^{k+i-1}) = 0, |\underline{u}_k^{k+i-1}| = i \right\}$$

be the set of all finite-horizon admissible control sequences of e_k with length i . Define the optimal cost function as

$$J^*(e_k) = \inf_{\underline{u}_k} \{J(e_k, \underline{u}_k) : \underline{u}_k \in \mathfrak{A}_{e_k}\}. \quad (9)$$

Note that equation (8) can be written as

$$\begin{aligned} J(e_k, \underline{u}_k^{N-1}) &= e_k^T Q e_k + u_k^T R u_k + \sum_{i=k+1}^{N-1} U(e_i, u_i) \\ &= e_k^T Q e_k + u_k^T R u_k + J(e_{k+1}, \underline{u}_{k+1}^{N-1}). \end{aligned} \quad (10)$$

According to Bellman's optimality principle, the optimal cost function $J^*(e_k)$ satisfies the DTHJB equation

$$J^*(e_k) = \min_{u_k} \{e_k^T Q e_k + u_k^T R u_k + J^*(e_{k+1})\}. \quad (11)$$

The optimal control u^* satisfies the first-order necessary condition, which can be given by

$$u^*(e_k) = -\frac{1}{2} R^{-1} g^T(e_k + r_k) \frac{\partial J^*(e_{k+1})}{\partial e_{k+1}}. \quad (12)$$

Since the above DTHJB equation cannot be solved exactly, we will present a novel algorithm to approximate the cost function iteratively in next section. Before that, we make the following assumption.

Assumption 1 For system (6), the inverse of the control coefficient matrix $g(e_k + r_k)$ exists.

3 Finite-Horizon Optimal Tracking Control Using the Iterative ADP Algorithm

3.1 Derivation of the Iterative ADP Algorithm

In this part, we present the iterative ADP algorithm, where the cost function and the control law are updated by recursive iterations.

First, we start with the initial cost function $V_0(\cdot) = 0$, and then solve for the law of single control vector $v_0(e_k)$ as follows:

$$\begin{aligned} v_0(e_k) &= \arg \min_{u_k} \{U(e_k, u_k) + V_0(e_{k+1})\} \\ \text{subject to } F(e_k, u_k) &= 0. \end{aligned} \quad (13)$$

Once the control law $v_0(e_k)$ is determined, we update the cost function as

$$\begin{aligned} V_1(e_k) &= \min_{u_k} \{U(e_k, u_k) + V_0(e_{k+1})\} \\ &= U(e_k, v_0(e_k)), \end{aligned}$$

which can also be written as

$$\begin{aligned} V_1(e_k) &= \min_{u_k} U(e_k, u_k) \text{ subject to } F(e_k, u_k) = 0 \\ &= U(e_k, v_0(e_k)). \end{aligned} \quad (14)$$

Then, for $i = 1, 2, \dots$, the iterative algorithm can be implemented between the control law

$$\begin{aligned} v_i(e_k) &= \arg \min_{u_k} \{U(e_k, u_k) + V_i(e_{k+1})\} \\ &= -\frac{1}{2} R^{-1} g^T(e_k + r_k) \frac{\partial V_i(e_{k+1})}{\partial e_{k+1}} \end{aligned} \quad (15)$$

and the cost function

$$\begin{aligned} V_{i+1}(e_k) &= \min_{u_k} \{U(e_k, u_k) + V_i(e_{k+1})\} \\ &= U(e_k, v_i(e_k)) + V_i(F(e_k, v_i(e_k))). \end{aligned} \quad (16)$$

Next, we will present a convergence proof of the iteration between (15) and (16) with the cost function $V_i \rightarrow J^*$ and the control law $v_i \rightarrow u^*$ as $i \rightarrow \infty$. Before that, we will see what $V_{i+1}(e_k)$ will be when it is expanded. According to (14) and (16), we can obtain

$$\begin{aligned} V_{i+1}(e_k) &= \min_{u_k} \{U(e_k, u_k) + V_i(e_{k+1})\} \\ &= \min_{\underline{u}_k^{k+1}} \{U(e_k, u_k) + U(e_{k+1}, u_{k+1}) \\ &\quad + V_{i-1}(e_{k+2})\} \\ &\vdots \\ &= \min_{\underline{u}_k^{k+i-1}} \{U(e_k, u_k) + U(e_{k+1}, u_{k+1}) \\ &\quad + \cdots + U(e_{k+i-1}, u_{k+i-1}) + V_1(e_{k+i})\} \end{aligned} \quad (17)$$

where

$$\begin{aligned} V_1(e_{k+i}) &= \min_{u_{k+i}} U(e_{k+i}, u_{k+i}) \\ \text{subject to } F(e_{k+i}, u_{k+i}) &= 0. \end{aligned} \quad (18)$$

Then, we have

$$\begin{aligned} V_{i+1}(e_k) &= \min_{\underline{u}_k^{k+i}} \sum_{j=0}^i U(e_{k+j}, u_{k+j}) \\ \text{subject to } F(e_{k+i}, u_{k+i}) &= 0 \\ &= \min_{\underline{u}_k^{k+i}} \left\{ J(e_k, \underline{u}_k^{k+i}) : \underline{u}_k^{k+i} \in \mathfrak{A}_{e_k}^{(i+1)} \right\}, \end{aligned} \quad (19)$$

which can also be written as

$$V_{i+1}(e_k) = \sum_{j=0}^i U(e_{k+j}, v_{i-j}(e_{k+j})) \quad (20)$$

when using the notation in (15). These equations will be useful in the convergence proof of the iterative ADP algorithm.

3.2 Convergence Proof of the Iterative ADP Algorithm

Theorem 1 Suppose $\mathfrak{A}_{e_k}^{(1)} \neq \emptyset$. Then, the cost function sequence $\{V_i\}$ obtained by (13)–(16) is a monotonically non-increasing sequence satisfying $V_{i+2}(e_k) \leq V_{i+1}(e_k)$ for $\forall i \geq 0$, i.e., $V_1(e_k) = \max\{V_i(e_k) : i = 1, 2, \dots\}$.

Proof We prove this theorem by mathematical induction. First, we let $i = 0$. The cost function $V_1(e_k)$ is given in (14) and the finite-horizon admissible control sequence is $\hat{u}_k^k = (v_0(e_k))$. Now, we show that there exists a finite-horizon admissible control sequence \hat{u}_k^{k+1} with length 2 such that $J(e_k, \hat{u}_k^{k+1}) = V_1(e_k)$. Let $\hat{u}_k^{k+1} = (\hat{u}_k^k, 0)$, then $|\hat{u}_k^{k+1}| = 2$. Since $e_{k+1} = F(e_k, v_0(e_k)) = 0$ and $\hat{u}_{k+1} = 0$, we have $e_{k+2} = F(e_{k+1}, \hat{u}_{k+1}) = F(0, 0) = 0$. Thus, \hat{u}_k^{k+1} is a finite-horizon admissible control sequence. Since $U(e_{k+1}, \hat{u}_{k+1}) = U(0, 0) = 0$, considering (20), we can obtain

$$\begin{aligned} J(e_k, \hat{u}_k^{k+1}) &= U(e_k, v_0(e_k)) + U(e_{k+1}, \hat{u}_{k+1}) \\ &= U(e_k, v_0(e_k)) \\ &= V_1(e_k). \end{aligned}$$

On the other hand, according to (19), we have

$$V_2(e_k) = \min_{\underline{u}_k^{k+1}} \left\{ J(e_k, \underline{u}_k^{k+1}) : \underline{u}_k^{k+1} \in \mathfrak{A}_{e_k}^{(2)} \right\},$$

which reveals that

$$V_2(e_k) \leq J(e_k, \hat{u}_k^{k+1}) = V_1(e_k). \quad (21)$$

Therefore, the theorem holds for $i = 0$.

Next, assume that the theorem holds for any $i = q - 1$, where $q > 1$. The current cost function can be expressed as

$$V_q(e_k) = \sum_{j=0}^{q-1} U(e_{k+j}, v_{q-1-j}(e_{k+j})),$$

where $\hat{u}_k^{k+q-1} = (v_{q-1}(e_k), v_{q-2}(e_{k+1}), \dots, v_0(e_{k+q-1}))$ is the corresponding finite-horizon admissible control sequence.

Then, for $i = q$, we can construct a control sequence $\hat{u}_k^{k+q} = (v_{q-1}(e_k), v_{q-2}(e_{k+1}), \dots, v_0(e_{k+q-1}), 0)$ with length $q + 1$, under which the error trajectory is given as $e_k, e_{k+1} = F(e_k, v_{q-1}(e_k)), e_{k+2} = F(e_{k+1}, v_{q-2}(e_{k+1})), \dots, e_{k+q} = F(e_{k+q-1}, v_0(e_{k+q-1})) = 0, e_{k+q+1} = F(e_{k+q}, \hat{u}_{k+q}) = F(0, 0) = 0$. This shows that \hat{u}_k^{k+q} is a finite-horizon admissible control sequence. As $U(e_{k+q}, \hat{u}_{k+q}) = U(0, 0) = 0$, considering (20), we can acquire

$$\begin{aligned} J(e_k, \hat{u}_k^{k+q}) &= U(e_k, v_{q-1}(e_k)) + U(e_{k+1}, v_{q-2}(e_{k+1})) \\ &\quad + \cdots + U(e_{k+q-1}, v_0(e_{k+q-1})) \\ &\quad + U(e_{k+q}, \hat{u}_{k+q}) \\ &= \sum_{j=0}^{q-1} U(e_{k+j}, v_{q-1-j}(e_{k+j})) \\ &= V_q(e_k). \end{aligned}$$

On the other hand, according to (19), we have

$$V_{q+1}(e_k) = \min_{\underline{u}_k^{k+q}} \left\{ J(e_k, \underline{u}_k^{k+q}) : \underline{u}_k^{k+q} \in \mathfrak{A}_{e_k}^{(q+1)} \right\},$$

which implies that

$$V_{q+1}(e_k) \leq J(e_k, \hat{u}_k^{k+q}) = V_q(e_k). \quad (22)$$

Accordingly, we complete the proof by mathematical induction. ■

We have concluded that the cost function sequence $\{V_i(e_k)\}$ is a monotonically nonincreasing sequence which is bounded below, and therefore, its limit exists. Here, we denote it as $V_\infty(e_k)$, i.e., $\lim_{i \rightarrow \infty} V_i(e_k) = V_\infty(e_k)$. Next, let us consider what will happen when we make $i \rightarrow \infty$ in (16).

Theorem 2 For any discrete time step k and tracking error e_k , the following equation holds:

$$V_\infty(e_k) = \min_{u_k} \{U(e_k, u_k) + V_\infty(e_{k+1})\}. \quad (23)$$

Proof For any admissible control $\tau_k = \tau(e_k)$ and i , according to Theorem 1 and (16), we have

$$\begin{aligned} V_\infty(e_k) &\leq V_{i+1}(e_k) \\ &= \min_{u_k} \{U(e_k, u_k) + V_i(e_{k+1})\} \\ &\leq U(e_k, \tau_k) + V_i(e_{k+1}). \end{aligned}$$

Let $i \rightarrow \infty$, we get

$$V_\infty(e_k) \leq U(e_k, \tau_k) + V_\infty(e_{k+1}).$$

Note that in the above equation, τ_k is chosen arbitrarily, thus, we can obtain

$$V_\infty(e_k) \leq \min_{u_k} \{U(e_k, u_k) + V_\infty(e_{k+1})\}. \quad (24)$$

On the other hand, let $\delta > 0$ be an arbitrary positive number. Then, there exists a positive integer l such that

$$V_l(e_k) - \delta \leq V_\infty(e_k) \leq V_l(e_k) \quad (25)$$

because $V_i(e_k)$ is nonincreasing for $i \geq 1$ with $V_\infty(e_k)$ as its limit. Besides, from (16), we can acquire

$$\begin{aligned} V_l(e_k) &= \min_{u_k} \{U(e_k, u_k) + V_{l-1}(e_{k+1})\} \\ &= U(e_k, v_{l-1}(e_k)) + V_{l-1}(F(e_k, v_{l-1}(e_k))). \end{aligned}$$

Combining with (25), we can obtain

$$\begin{aligned} V_\infty(e_k) &\geq U(e_k, v_{l-1}(e_k)) + V_{l-1}(F(e_k, v_{l-1}(e_k))) - \delta \\ &\geq U(e_k, v_{l-1}(e_k)) + V_\infty(F(e_k, v_{l-1}(e_k))) - \delta \\ &\geq \min_{u_k} \{U(e_k, u_k) + V_\infty(e_{k+1})\} - \delta, \end{aligned}$$

which reveals that

$$V_\infty(e_k) \geq \min_{u_k} \{U(e_k, u_k) + V_\infty(e_{k+1})\} \quad (26)$$

because of the arbitrariness of δ . Based on (24) and (26), we can conclude that (23) is true. ■

Theorem 3 Define the cost function sequence $\{V_i\}$ as in (16) with $V_0(\cdot) = 0$. If the system state e_k is controllable, then J^* is the limit of the cost function sequence $\{V_i\}$, i.e.,

$$V_\infty(e_k) = J^*(e_k).$$

Proof On the one hand, in accordance with (9) and (19), we can acquire

$$\begin{aligned} J^*(e_k) &= \inf_{\underline{u}_k} \{J(e_k, \underline{u}_k) : \underline{u}_k \in \mathcal{A}_{e_k}\} \\ &\leq \min_{\underline{u}_k^{k+i-1}} \left\{ J(e_k, \underline{u}_k^{k+i-1}) : \underline{u}_k^{k+i-1} \in \mathcal{A}_{e_k}^{(i)} \right\} \\ &= V_i(e_k). \end{aligned}$$

Let $i \rightarrow \infty$, we get

$$J^*(e_k) \leq V_\infty(e_k). \quad (27)$$

On the other hand, according to the definition of $J^*(e_k)$, for any $\eta > 0$, there exists an admissible control sequence $\underline{\sigma}_k \in \mathcal{A}_{e_k}$ such that

$$J(e_k, \underline{\sigma}_k) \leq J^*(e_k) + \eta. \quad (28)$$

Now, we suppose that $|\underline{\sigma}_k| = q$, which shows that $\underline{\sigma}_k \in \mathcal{A}_{e_k}^{(q)}$. Then, we can obtain

$$\begin{aligned} V_\infty(e_k) &\leq V_q(e_k) \\ &= \min_{\underline{u}_k^{k+q-1}} \left\{ J(e_k, \underline{u}_k^{k+q-1}) : \underline{u}_k^{k+q-1} \in \mathcal{A}_{e_k}^{(q)} \right\} \\ &\leq J(e_k, \underline{\sigma}_k), \end{aligned}$$

when using Theorem 1 and (19). Combining with (28), we get

$$V_\infty(e_k) \leq J^*(e_k) + \eta.$$

Noticing that η is chosen arbitrarily in the above equation, we have

$$V_\infty(e_k) \leq J^*(e_k). \quad (29)$$

Based on (27) and (29), we can acquire that $J^*(e_k)$ is the limit of the cost function sequence $\{V_i\}$ as $i \rightarrow \infty$, i.e., $V_\infty(e_k) = J^*(e_k)$. ■

From Theorems 1–3, we can obtain that the cost function sequence $\{V_i(e_k)\}$ converges to the optimal cost function $J^*(e_k)$ of the DTHJB equation, i.e., $V_i \rightarrow J^*$ as $i \rightarrow \infty$. Then, according to (12) and (15), we can conclude the convergence of the corresponding control law sequence, i.e., $\lim_{i \rightarrow \infty} v_i(e_k) = u^*(e_k)$.

3.3 The ε -Optimal Control Algorithm

According to Theorems 1–3, we should run the iterative ADP algorithm (13)–(16) until $i \rightarrow \infty$ to obtain the optimal cost function $J^*(e_k)$, and then to get a control vector $v_\infty(e_k)$ based on which we can construct a control sequence $\underline{u}_\infty(e_k) = (v_\infty(e_k), v_\infty(e_{k+1}), \dots, v_\infty(e_{k+i}), \dots)$ to control the state to reach the target. Obviously, $\underline{u}_\infty(e_k)$ has infinite length. Though it is feasible in terms of theory, it is always not practical to do so because most real world systems need to be effectively controlled within finite-horizon. Therefore, in this section, we will propose a novel ε -optimal control strategy using the iterative ADP algorithm to deal with the problem. The idea is, for a given error bound $\varepsilon > 0$, the iterative number i will be chosen so that the error between $V_i(e_k)$ and $J^*(e_k)$ is within the bound.

Let $\varepsilon > 0$ be any small number, e_k be any controllable state, and $J^*(e_k)$ be the optimal value of the cost function

sequence defined as in (16). From Theorem 3, it is clearly that there exists a finite i such that

$$|V_i(e_k) - J^*(e_k)| \leq \varepsilon. \quad (30)$$

The length of the optimal control sequence starting from e_k with respect to ε is defined as

$$K_\varepsilon(e_k) = \min\{i : |V_i(e_k) - J^*(e_k)| \leq \varepsilon\}. \quad (31)$$

The corresponding control law

$$\begin{aligned} v_{i-1}(e_k) &= \arg \min_{u_k} \{U(e_k, u_k) + V_{i-1}(e_{k+1})\} \\ &= -\frac{1}{2} R^{-1} g^T(e_k + r_k) \frac{\partial V_{i-1}(e_{k+1})}{\partial e_{k+1}} \end{aligned} \quad (32)$$

is called the ε -optimal control and is denoted as $\mu_\varepsilon^*(e_k)$.

In this sense, we can see that an error ε between $V_i(e_k)$ and $J^*(e_k)$ is introduced into the iterative ADP algorithm, which makes the cost function sequence $\{V_i(e_k)\}$ converge within finite number of iteration steps.

However, the optimal criterion (30) is difficult to verify because the optimal cost function $J^*(e_k)$ is unknown in general. Consequently, we will use an equivalent criterion, i.e.,

$$|V_i(e_k) - V_{i+1}(e_k)| \leq \varepsilon \quad (33)$$

to replace (30).

In fact, if $|V_i(e_k) - J^*(e_k)| \leq \varepsilon$ holds, we have $V_i(e_k) \leq J^*(e_k) + \varepsilon$. Combining with $J^*(e_k) \leq V_{i+1}(e_k) \leq V_i(e_k)$, we can find that

$$0 \leq V_i(e_k) - V_{i+1}(e_k) \leq \varepsilon,$$

which means

$$|V_i(e_k) - V_{i+1}(e_k)| \leq \varepsilon.$$

On the other hand, according to Theorem 3, $|V_i(e_k) - V_{i+1}(e_k)| \rightarrow 0$ connotes that $V_i(e_k) \rightarrow J^*(e_k)$. As a result, if $|V_i(e_k) - V_{i+1}(e_k)| \leq \varepsilon$ holds for any given small ε , we can derive the conclusion that $|V_i(e_k) - J^*(e_k)| \leq \varepsilon$ holds if i is sufficiently large.

Remark 1 After the optimal control law $u^*(e_k)$ for system (6) is derived under the given error bound ε , we can compute the optimal tracking control input for original system (1) by

$$\begin{aligned} u_{pk}^* &= u^*(e_k) + u_{dk} \\ &= u^*(e_k) + g^{-1}(r_k)(\phi(r_k) - f(r_k)). \end{aligned} \quad (34)$$

4 NN Implementation of the Iterative Algorithm

In this section, we implement the iterative HDP algorithm in (13)–(16) using NNs. In the iterative HDP algorithm, there are three networks, which are model network, critic network and action network. All the networks are chosen as the three-layer feedforward NNs. The structure diagram of the iterative HDP algorithm is shown in Fig. 1. The weights are updated using the gradient-based adaption rule and the detailed training processes are omitted here, which can be referred to [5, 8].

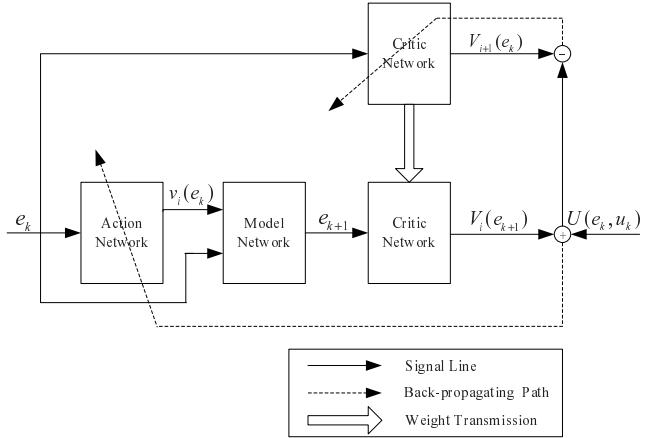


Fig. 1: The structure diagram of the iterative HDP algorithm

5 Simulation Study

Consider the nonlinear system $x_{k+1} = f(x_k) + g(x_k)u_{pk}$ where $x_k = [x_{1k} \ x_{2k}]^T \in \mathbb{R}^2$ and $u_{pk} = [u_{p1k} \ u_{p2k}]^T \in \mathbb{R}^2$ are the state and control variables, respectively. The parameters of the cost function are chosen as $Q = 0.5I$ and $R = 2I$, where I denotes the identity matrix with suitable dimensions. The state of the controlled system is initialized to be $x_0 = [0.8 \ -0.5]^T$. The system functions are given as

$$\begin{aligned} f(x_k) &= \begin{bmatrix} \sin(0.5x_{2k})x_{1k}^2 \\ \cos(1.4x_{2k}) \sin(0.9x_{1k}) \end{bmatrix} \\ g(x_k) &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \end{aligned}$$

The reference trajectory for the above system is selected as

$$r_k = \begin{bmatrix} \sin(0.25k) \\ \cos(0.25k) \end{bmatrix}.$$

We set the error bound of the iterative ADP algorithm as $\varepsilon = 10^{-5}$ and implement the algorithm at time instant $k = 0$. The initial control vector of system (6) can be computed as $v_0(e_0) = [0.64 \sin(0.25) \ -\sin(0.72) \cos(0.7)]^T$, where $e_0 = [0.8 \ -1.5]^T$. Then, we choose three-layer feed-forward NNs as model network, critic network and action network with the structures 4–8–2, 2–8–1, 2–8–2, respectively. The initial weights of the three networks are all set to be random in $[-1, 1]$. It should be mentioned that the model network should be trained first. We train the model network for 1000 steps using 500 data samples under the learning rate $\alpha_m = 0.1$. After the training of the model network is completed, the weights are kept unchanged. Then, we train the critic network and action network for 20 iterations (i.e., for $i = 1, 2, \dots, 20$) with each iteration of 2000 training steps to make sure the given error bound $\varepsilon = 10^{-5}$ is reached. In the training process, the learning rate $\alpha_c = \alpha_a = 0.05$. The convergence process of the cost function of the iterative HDP algorithm is shown in Fig. 2, for $k = 0$. We can see that the iterative cost function sequence does converge to the optimal one quite rapidly, which indicates the effectiveness of the iterative HDP algorithm. Actually, we have $|V_{19}(e_0) - V_{20}(e_0)| \leq \varepsilon$, which means that the step number of ε -optimal control is $K_\varepsilon(e_0) = 19$. Besides, the ε -optimal

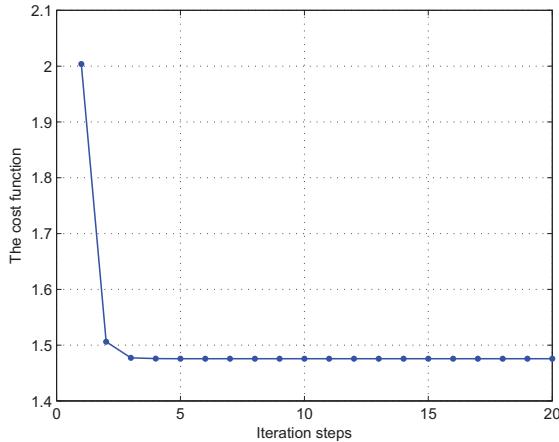


Fig. 2: The convergence process of the cost function

control law $\mu_\varepsilon^*(e_0)$ for system (6) can also be obtained during the iteration process.

Next, we compute the near-optimal tracking control law for original system (1) using (34) and apply it to the controlled system for 40 time steps. The tracking control curves and the tracking errors are shown in Fig. 3 and Fig. 4, respectively. Besides, we can derive that the tracking error becomes $e_{19} = [0.2778 \times 10^{-5} \quad -0.8793 \times 10^{-5}]^T$ after 19 time steps. These simulation results verify the excellent performance of the tracking controller developed by the iterative ADP algorithm.

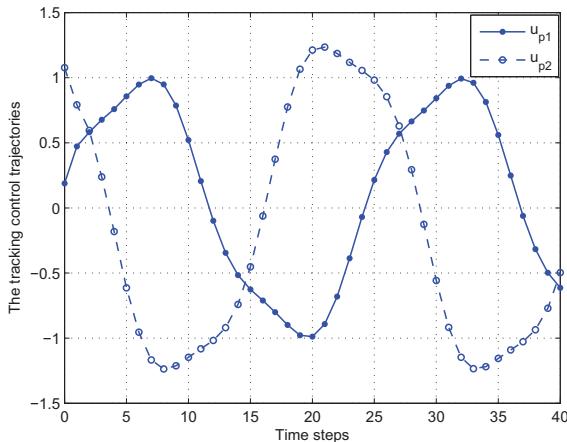


Fig. 3: The tracking control trajectories u_p

6 Conclusions

In this paper, an effective learning control method is proposed to design the finite-horizon near-optimal tracking controller for a class of discrete-time nonlinear systems. The iterative ADP algorithm is introduced to solve the cost function of the DTHJB equation with convergence analysis, which obtains a finite-horizon near-optimal tracking controller that makes the cost function close to its optimal value within an ε -error bound. Three NNs are used to approximate the cost function, the control law, and the tracking error dynamics, respectively. The simulation example confirmed the validity of the tracking control scheme.

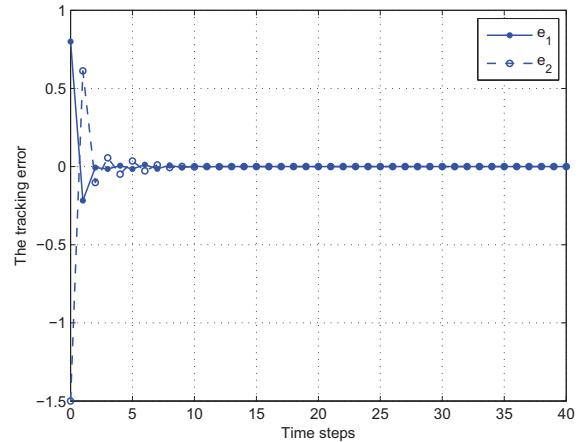


Fig. 4: The tracking error e

References

- [1] P. J. Werbos, Approximate dynamic programming for real-time control and neural modeling, in *Handbook of Intelligent Control*, D. A. White, D. A. Sofge, Eds. New York: Van Nostrand Reinhold, 1992, chapter 13.
- [2] F. Y. Wang, H. Zhang, and D. Liu, Adaptive dynamic programming: an introduction, *IEEE Computational Intelligence Magazine*, 4(2): 39–47, 2009.
- [3] F. L. Lewis and D. Vrabie, Reinforcement learning and adaptive dynamic programming for feedback control, *IEEE Circuits and Systems Magazine*, 9(3): 32–50, 2009.
- [4] A. Al-Tamimi, F. L. Lewis, and M. Abu-Khalaf, Discrete-time nonlinear HJB solution using approximate dynamic programming: convergence proof, *IEEE Trans. on Systems, Man, and Cybernetics—Part B: Cybernetics*, 38(4): 943–949, 2008.
- [5] J. Si and Y. T. Wang, On-line learning control by association and reinforcement, *IEEE Trans. on Neural Networks*, 12(2): 264–276, 2001.
- [6] D. V. Prokhorov and D. C. Wunsch, Adaptive critic designs, *IEEE Trans. on Neural Networks*, 8(5): 997–1007, 1997.
- [7] D. Liu, Approximate dynamic programming for self-learning control, *Acta Automatica Sinica*, 31(1): 13–18, 2005.
- [8] H. Zhang, Y. Luo, and D. Liu, Neural-network-based near-optimal control for a class of discrete-time affine nonlinear systems with control constraints, *IEEE Trans. on Neural Networks*, 20(9): 1490–1503, 2009.
- [9] L. Yang, J. Si, K. S. Tsakalis, and A. A. Rodriguez, Direct heuristic dynamic programming for nonlinear tracking control with filtered tracking error, *IEEE Trans. on Systems, Man, and Cybernetics—Part B: Cybernetics*, 39(6): 1617–1622, 2009.
- [10] H. Zhang, Q. Wei, and 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. on Systems, Man, and Cybernetics—Part B: Cybernetics*, 38(4): 937–942, 2008.
- [11] T. Dierks and S. Jagannathan, Optimal tracking control of affine nonlinear discrete-time systems with unknown internal dynamics, in *Proceedings of Joint 48th IEEE Conference on Decision and Control and 28th Chinese Control Conference*, 2009: 6750–6755.
- [12] Y. M. Park, M. S. Choi, and K. Y. Lee, An optimal tracking neuro-controller for nonlinear dynamic systems, *IEEE Trans. on Neural Networks*, 7(5): 1099–1110, 1996.