



# Event-triggered design for discrete-time nonlinear systems with control constraints

Chaoxu Mu · Kaiju Liao · Ke Wang

Received: 15 January 2020 / Accepted: 11 January 2021 / Published online: 11 February 2021  
© The Author(s), under exclusive licence to Springer Nature B.V. part of Springer Nature 2021

**Abstract** In order to solve the constrained-input problem and reduce the computing resources, a novel event-triggered optimal control method is proposed for a class of discrete-time nonlinear systems. In the proposed method, the event-triggered control policy is applied to the globalized dual heuristic dynamic programming (GDHP) algorithm. Compared with the traditional adaptive dynamic programming (ADP) control, the event-triggered GDHP control can reduce the computation while ensuring the system performance. In this paper, a non-quadratic function is given to code the control constraints and the trigger condition with the stability analysis is provided. Neural networks (NN) are constructed in the GDHP structure, where the model network is designed to identify the unknown nonlinear system, the critic network is used to learn the cost function and its partial derivative, and the action network is designed to obtain the approximate optimal control law. Three simulation examples are presented to demonstrate the performance of the proposed event-triggered design for constrained discrete-time nonlinear systems.

**Keywords** Event-triggered control · Globalized dual heuristic dynamic programming (GDHP) · Control constraints · Neural networks (NN)

## 1 Introduction

Traditional period control uses a time-triggered mechanism in which the control law is executed for every fixed time interval. Such a sampling mechanism greatly increases the computing burden and causes computing waste. Compared with the traditional time-triggered control, the event-triggered control can effectively utilize computing resources and reduce the resources waste [1, 2]. The event-triggered control updates the control signal irregularly by setting the trigger condition on the premise of guaranteeing the system performance [3–5]. Under this event-triggered mechanism, the control law is only executed when the trigger condition is satisfied.

Event-triggered control has been widely studied and applied in general linear systems, continuous-time systems, discrete-time systems and other fields [6–11]. For instance, Zhang et al. [7] proposed a state-feedback event-triggered control method for linear systems. Tabuada [8] introduced an event-triggered mechanism for nonlinear systems, which can make the Lyapunov function decrease strictly along the solution curve. Qiu et al. [10] designed a fuzzy event-triggered control for pure-feedback nonlinear systems with unknown states. Recently, many scholars have applied the event-

---

C. Mu (✉) · K. Liao · K. Wang  
School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China  
e-mail: cxmu@tju.edu.cn

K. Liao  
e-mail: liaokaiju@tju.edu.cn

K. Wang  
e-mail: walker\_wang@tju.edu.cn

triggered control to the adaptive dynamic programming (ADP) algorithm [12–18]. Zhong et al. [15] proposed an event-triggered mechanism for continuous-time nonlinear systems and studied the unknown system dynamic by a NN observer. Literature [16] gave an event-triggered ADP control method and designed a dynamic NN structure to identify the system internal states for continuous-time systems. In [17], Yang et al. used NN structure to reconstruct the angular position and angular velocity signals of the robot arm, and introduced the event-triggered ADP control method to approximate the performance index of the robot.

ADP algorithm is a useful method of iteratively solving the optimal control for various systems, which satisfies the principle of Bellman optimality [19–23]. In 1977, Werbos [19] first proposed the framework of the ADP algorithm. The main idea of this method is to use the function approximation structure (such as neural network, fuzzy model, polynomial, etc.) to approximate the cost function and control law. Subsequently, Murray et al. [20] presented a specific ADP iterative algorithm for continuous systems and gave strict proof of stability and convergence. Prokhorov and Wunsch [24] summarized that the main structures of ADP algorithm can be classified as: heuristic dynamic programming (HDP), dual heuristic dynamic programming (DHP), globalized DHP (GDHP), and their action-dependent structure. On the basis of previous studies, Jiang et al. [21] introduced a novel policy iterative algorithm without relying on the system dynamic. Al-Tamimi et al. [22] proposed an value-iteration-based ADP algorithm for discrete-time nonlinear systems with unknown internal dynamic. In [23], the greedy ADP algorithm was proposed to solve the tracking control problem for discrete-time nonlinear systems by converting the optimal tracking problem into the optimal adjustment problem.

Control constraints widely exist in practical worlds, which can easily damage the overall performance of the system and maybe lead to system instability [25–28]. As one of the powerful method to solve the optimal control problem for nonlinear systems, ADP method also plays an important role in the system with control constraints. Na et al. [29] proposed a novel online control policy for constrained nonlinear systems based on iterative ADP algorithm. Fan et al. [30] solved the output-constrained optimal control problem for continuous-time nonlinear systems. In addition, researchers have applied iterative ADP event-triggered control to a class

of constrained-input systems. For continuous-time constrained nonlinear systems, Zhu et al. [31] introduced an event-triggered optimal control policy and gave the detailed Lyapunov analysis. Literature [32] considered the global stability of the saturated system, and proposed a state-dependent non-quadratic event-triggered control method. In [33], an event-triggered state feedback control policy was provided for constrained linear systems, the positive lower bounds and the self-triggered method were also given.

Recently, scholars have proposed many event-triggered methods for linear systems, but there are few studies that focus on discrete-time nonlinear systems. Besides, for discrete-time nonlinear systems, researches usually use the basic structures (such as HDP and DHP) of ADP algorithm, and do not consider the constrained-input problem. Motivated by this, we propose a novel ADP-based event-triggered approximate optimal control for discrete-time nonlinear systems with control constraints. In this paper, the globalized dual heuristic dynamic programming (GDHP) structure is designed to learn the event-triggered optimal control, in which the information of cost function and its partial derivative are both studied by the critic network. Compared with the HDP and DHP structures, the GDHP structure learns more system information, which enables the GDHP method to obtain better control performance. The contributions of this paper are summarized as follows: (1) the event-triggered design is developed on the GDHP technique for discrete-time nonlinear systems, where the control law generated from action network is updated when the trigger condition is satisfied. (2) The control constraints are well considered in the event-triggered design based on GDHP structure, and the stability of event-triggered constrained systems has been provided.

The rest of this paper is organized as follows. Section 2 introduces the adaptive event-triggered control for a class of discrete-time nonlinear systems with control constraints. The trigger condition and the corresponding stability analysis are given in Sect. 3. The approximate optimal learning method using GDHP structure and the detailed iterative process are discussed in Sect. 4. Section 5 presents three simulation examples to prove the effectiveness of the proposed method. Finally, Sect. 6 gives the conclusion and discussion.

## 2 Problem formulation

Consider a class of discrete-time nonlinear systems described as

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

where  $x_k = [x_{1k}, x_{2k}, \dots, x_{nk}]^T \in \mathbb{R}^n$  is the state vector,  $u_k = [u_{1k}, u_{2k}, \dots, u_{mk}]^T \in \mathbb{R}^m$  is the control input vector. For any  $x_k$ ,  $f(x_k) : \mathbb{R}^n \rightarrow \mathbb{R}^n$  is differentiable with  $f(0) = 0$ ,  $g(x_k) : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times m}$  is nonsingular. Assume that system (1) is Lipschitz continuous on a set  $\Omega$  in  $\mathbb{R}^n$  containing the origin, and the system (1) is controllable in the sense that there exists a continuous control on  $\Omega$  to stabilize the system. Let us define  $\Omega_u = \{u_k | u_k = [u_{1k}, u_{2k}, \dots, u_{mk}]^T \in \mathbb{R}^m, |u_{ik}| \leq \bar{u}_i, i = 1, 2, \dots, m\}$ , where  $\bar{u}_i$  is the saturating boundary of the  $i$ th executor.  $\bar{U} \in \mathbb{R}^{m \times m}$  is the constant diagonal matrix given by  $\bar{U} = \text{diag} [\bar{u}_1, \bar{u}_2, \dots, \bar{u}_m]$ .

In the event-triggered control, we set the time  $\{k_i\}_{i=0}^\infty$  as sampling instant, which means the controller only samples at discrete-time points  $k_0, k_1, k_2, \dots$ . The state feedback control law  $u_k$  satisfies

$$u_k = \mu(x_{k_i}), \tag{2}$$

where  $x_{k_i}$  represents the state vector at sampling instant  $k_i \leq k < k_{i+1}, i = 0, 1, 2, \dots$ . In addition, we design a zero-order-hold (ZOH) device to maintain the control input during the trigger interval. Thus, a continuous control input sequence can be obtained by the ZOH.

Define the event-triggered error as

$$e_k = x_{k_i} - x_k; \quad k \in [k_i, k_{i+1}), \tag{3}$$

where  $x_{k_i}$  represents the state at the sampling instant,  $x_k$  represents the current state. By (2) and (3), we can get

$$u_k = \mu(e_k + x_k). \tag{4}$$

Then, applying (4) into (1), we have

$$x_{k+1} = f(x_k) + g(x_k)\mu(e_k + x_k). \tag{5}$$

The general discrete-time optimal control problem is to find the control law  $u_k$  that can minimize the following infinite domain cost function

$$V(x_k) = \sum_{j=k}^\infty U(x_j, \mu(e_j + x_j)), \tag{6}$$

where  $\mu(e_j + x_j) = \mu(x_{k_i})$ , and  $U(x_j, \mu(e_j + x_j))$  is the utility function. The utility function is usually a quadratic form which can be described as

$$U(x_k, \mu(x_{k_i})) = x_k^T Q x_k + \mu^T(x_{k_i}) R \mu(x_{k_i}), \tag{7}$$

where  $Q$  and  $R$  are symmetric positive definite matrices with appropriate dimensions, and  $U(0, 0) = 0$ . However, such a quadratic utility function is not suitable for the system with control constraints. Thus, a non-quadratic form is provided to solve the constrained-input problem, and the utility function becomes

$$U(x_k, \mu(x_{k_i})) = x_k^T Q x_k + 2 \int_0^{\mu(x_{k_i})} \varphi^{-T}(\bar{U}^{-1} \tau) \bar{U} R d\tau, \tag{8}$$

where  $\tau \in \mathbb{R}^m$ ,  $\varphi(\cdot) \in \mathbb{R}^m$  is a bounded one-to-one function satisfying  $|\varphi(\cdot)| \leq 1$ . Here,  $U(x_k, \mu(x_{k_i}))$  is denoted by  $U_k$ .

Based on Bellman optimality principle, we can obtain the optimal cost function  $V^*(x_k)$  as

$$V^*(x_k) = \min_{\mu(x_{k_i})} \{U_k + V^*(x_{k+1})\}. \tag{9}$$

In addition, the control law  $u_k$  satisfies the first-order necessary condition of optimal control [34]. For  $k \in [k_i, k_{i+1}), i = 0, 1, 2, \dots$ , the optimal control law  $\mu^*(x_{k_i})$  can be obtained as

$$\begin{aligned} \mu^*(x_{k_i}) &= \arg \min_{\mu(x_{k_i})} \{U_k + V^*(x_{k+1})\} \\ &= \bar{U} \varphi \left( -\frac{1}{2} (\bar{U} R)^{-1} g^T(x_k) \frac{\partial V^*(x_{k+1})}{\partial x_{k+1}} \right). \end{aligned} \tag{10}$$

Next, we will give an event-triggered condition and prove the corresponding stability for system (5).

## 3 Stability proof under the event-triggered condition

**Definition 1** (cf. [35]) For  $\forall x_k \in \Omega$ , a control law  $u_k$  is admissible with respect to (6) on  $\Omega$  if  $u_k$  is continuous and stabilizes (1) on  $\Omega, u_k = 0$  if  $x_k = 0$ , and  $\forall x_0 \in \Omega, V(x_0)$  is finite. For the constant diagonal matrix  $\bar{U} \in \mathbb{R}^{m \times m}$ , let  $m = 1$ , there is  $\bar{u} = \bar{U}$  with  $\|\mu_{k_i}\| \leq \bar{u}$ .

For system (5), we define an event-triggered condition  $\|e_k\| \leq e_T$ , where  $e_T$  is the trigger threshold. During the event-triggered control, the action network will update the corresponding control law only if the trigger condition is satisfied. Besides, at each sampling instant  $k_i, i = 0, 1, 2, \dots$ , the trigger error  $\|e_k\|$  will be reset to zero.

**Assumption 1** (cf. [36]) If function  $V: \mathbb{R}^n \rightarrow \mathbb{R}^n \geq 0$  is continuously differentiable, the state vector  $x_k$  and the trigger error  $e_k$  satisfy

$$\|f(x_k - e_k)\| \leq P_1 \|e_k\| + P_1 \|x_k\|, \tag{11}$$

$$\|g(x_k - e_k)\| \leq P_2 \|e_k\| + P_2 \|x_k\|, \tag{12}$$

$$\alpha_1(\|x\|) \leq V(x_k) \leq \alpha_2(\|x\|), \quad \forall x \in \mathbb{R}^n \tag{13}$$

$$V(x_{k+1}) - V(x_k) \leq -\alpha V(x_k) + \beta \|e_k\|, \tag{14}$$

$$\alpha_1^{-1}(\|x\|) \leq L \|x\|. \tag{15}$$

where  $L, P_1, P_2, \alpha$  and  $\beta$  are the positive constants,  $\alpha_1$  and  $\alpha_2$  are the class  $\kappa_\infty$  functions.

Among them, if (13) and (14) hold, function  $V$  is called an input-to-state stability Lyapunov (ISS-Lyapunov) function [37].

According to (3), for each  $k \in [k_i, k_{i+1})$ , we have

$$e_{k+1} = x_{k_i} - x_{k+1}, \tag{16}$$

where  $k_i$  is the latest sampling instant. In addition, according to [36], we can get

$$\|e_{k+1}\| \leq \|x_{k+1}\|. \tag{17}$$

From Assumption 1, by applying (3) and (5) into (17), we have

$$\begin{aligned} \|e_{k+1}\| &\leq \|f(x_k) + g(x_k)\mu(x_{k_i})\| \\ &\leq \|f(x_k)\| + \|g(x_k)\| \bar{\mu} \\ &= \|f(x_{k_i} - e_k)\| + \|g(x_{k_i} - e_k)\| \bar{\mu} \\ &\leq (P_1 + P_2 \bar{\mu}) \|x_{k_i}\| + (P_1 + P_2 \bar{\mu}) \|e_k\| \\ &\leq (P_1 + P_2 \bar{\mu}) \|x_{k_i}\| + 2(P_1 + P_2 \bar{\mu}) \|e_k\|. \end{aligned} \tag{18}$$

Therefore, we can obtain

$$\begin{aligned} \|e_k\| &\leq 2(P_1 + P_2 \bar{\mu}) \|e_{k-1}\| + (P_1 + P_2 \bar{\mu}) \|x_{k_i}\| \\ &\leq 2(P_1 + P_2 \bar{\mu})(2(P_1 + P_2 \bar{\mu}) \|e_{k-2}\| \\ &\quad + (P_1 + P_2 \bar{\mu}) \|x_{k_i}\|) + (P_1 + P_2 \bar{\mu}) \|x_{k_i}\| \cdots \\ &\leq (2(P_1 + P_2 \bar{\mu}))^{k-k_i} \|e_{k_i}\| \\ &\quad + (2(P_1 + P_2 \bar{\mu}))^{k-k_i-1} (P_1 + P_2 \bar{\mu}) \|x_{k_i}\| \\ &\quad + (2(P_1 + P_2 \bar{\mu}))^{k-k_i-2} (P_1 + P_2 \bar{\mu}) \|x_{k_i}\| \\ &\quad + \cdots + (P_1 + P_2 \bar{\mu}) \|x_{k_i}\|. \end{aligned} \tag{19}$$

Set the initial condition  $e_{k_i} = 0$ , Eq. (19) can be solved as

$$\|e_k\| \leq \frac{1 - (2(P_1 + P_2 \bar{\mu}))^{k-k_i}}{1 - 2(P_1 + P_2 \bar{\mu})} (P_1 + P_2 \bar{\mu}) \|x_{k_i}\|. \tag{20}$$

So, we define (20) as the event-triggered condition, such that

$$\begin{aligned} \|e_k\| &\leq e_T \\ &= \frac{1 - (2(P_1 + P_2 \bar{\mu}))^{k-k_i}}{1 - 2(P_1 + P_2 \bar{\mu})} (P_1 + P_2 \bar{\mu}) \|x_{k_i}\|. \end{aligned} \tag{21}$$

In the following, we will give the stability proof of the system (5) with control constraints under the condition (21).

**Theorem 1** According to Assumption 1, if  $0 \leq P_1 + P_2 \bar{\mu} \leq 1$  and the function  $V(x_k)$  satisfies

$$\begin{aligned} V(x_k) &\leq V(x_{k_i+1}) \\ &= -\xi \alpha V(x_{k_i})(k_{i+1} - k_i) + V(x_{k_i}), \end{aligned} \tag{22}$$

for  $k \in [k_i, k_{i+1})$ ,  $i = 0, 1, 2, \dots$ , where  $\xi \in (0, 1)$ , the event-triggered control system (5) with control constraints is asymptotically stable.

*Proof* By (13) and (15), one can get

$$\|x_{k_i}\| \leq \alpha_1^{-1}(V(x_{k_i})) \leq L V(x_{k_i}). \tag{23}$$

Substituting (20) into (14), one has

$$\begin{aligned} V(x_{k+1}) - V(x_k) &\leq -\alpha V(x_k) \\ &\quad + \beta \frac{1 - (2(P_1 + P_2 \bar{\mu}))^{k-k_i}}{1 - 2(P_1 + P_2 \bar{\mu})} (P_1 + P_2 \bar{\mu}) \|x_{k_i}\|. \end{aligned} \tag{24}$$

Then, considering (22) and (23), one can define

$$\psi_k = \beta \frac{1 - (2(P_1 + P_2 \bar{\mu}))^{k-k_i}}{1 - 2(P_1 + P_2 \bar{\mu})} L (P_1 + P_2 \bar{\mu}). \tag{25}$$

Equation (24) can be written as

$$V(x_{k+1}) \leq (1 - \alpha)V(x_k) + \psi_k V(x_{k_i}). \tag{26}$$

Therefore, it obtains

$$V(x_k) \leq (1 - \alpha)V(x_{k-1}) + \psi_{k-1} V(x_{k_i})$$

$$\begin{aligned}
 &\leq (1 - \alpha)(1 - \alpha)V(x_{k-2}) + \psi_{k-2}V(x_{k_i}) \\
 &\quad + \psi_{k-1}V(x_{k_i}) \cdots \\
 &\leq (1 - \alpha)^{k-k_i}V(x_{k_i}) \\
 &\quad + (1 - \alpha)^{k-k_i-1}\psi_{k_i}V(x_{k_i}) + \cdots \\
 &\quad + (1 - \alpha)\psi_{k-2}V(x_{k_i}) + \psi_kV(x_{k_i}). \tag{27}
 \end{aligned}$$

According to theorem 1,  $\psi_k$  is a monotonically increasing function with positive common ratio. Then, (27) can be solved as

$$\begin{aligned}
 V(x_k) &\leq (1 - \alpha)^{k-k_i}V(x_{k_i}) \\
 &\quad + \psi_k \frac{1 - (1 - \alpha)^{k-k_i}}{\alpha} V(x_{k_i}). \tag{28}
 \end{aligned}$$

Based on (22), one has

$$V(x_k) \leq -\xi\alpha V(x_{k_i})(k - k_i) + V(x_{k_i}). \tag{29}$$

To simplify the calculation, one can define

$$M(x_k) = -\xi\alpha V(x_{k_i})(k - k_i) + V(x_{k_i}). \tag{30}$$

Therefore, one has

$$V(x_k) \leq M(x_k). \tag{31}$$

From (30), the first difference of  $M(x_k)$  can be obtained as

$$\begin{aligned}
 \Delta M &= M(x_{k+1}) - M(x_k) \\
 &= -\xi\alpha V(x_{k_i}). \tag{32}
 \end{aligned}$$

Then, substituting (13) into (32), it obtains

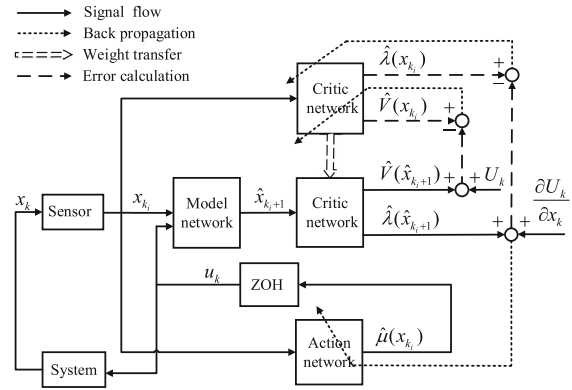
$$\Delta M \leq -\xi\alpha \cdot \alpha_1(x \|k_i\|) < 0. \tag{33}$$

This completes the proof. □

From the above derivation process, it can be concluded that the event-triggered control system (5) with control constrains is asymptotically stable.

### 4 Event-triggered controller design based on GDHP structure

In this section, we will apply the event-triggered condition into the GDHP structure. Due to the actual optimal control law cannot be obtained in theory, an iterative stopping criterion is designed to obtain the approximate optimal control law. Only when the stopping condition is satisfied, an iterative process is completed.



**Fig. 1** Event-triggered control system based on the GDHP structure

Then, the trigger error between the sampled state and the current state is compared with the trigger threshold online. When the designed trigger condition is violated, the current control law will be sampled. Otherwise, the control law is maintained by a ZOH.

This section is divided into three parts. Firstly, an approximate optimal event-triggered controller is designed in the first part. Then, the specific implementation process of NN is given in the second part. Subsequently, the iterative stopping criterion is provided in the third part.

#### 4.1 Event-triggered controller design

The event-triggered controller design based on GDHP structure is displayed in Fig. 1. It can be seen that the proposed event-triggered GDHP method is implemented by three NN structures, where the model network is designed to obtain the system dynamic, the critic network is used to approximate the cost function and its partial derivative, and the action network is designed to obtain the event-triggered approximate optimal control law.

In addition, a sensor device is designed to judge whether the trigger condition is violated. When the trigger condition is violated, the current time is set to the sampling instant  $k_i, i = 1, 2, \dots$ , and the control law  $\mu(x_{k_i})$  is maintained by the ZOH device during  $k_i \leq k < k_{i+1}$ .

## 4.2 NN implementation of GDHP technique

### 4.2.1 Model network

For the unknown system, before performing the iterative calculation, a model network is first constructed to obtain the system dynamic. The number of hidden layer neurons is set as  $N_m$ . Let  $v_m$  denote the weight matrix of input-to-hidden layer, and  $\omega_m$  denote the weight matrix of hidden-to-output layer. According to the state vector  $x_k$  and the control law  $\mu(x_k)$ , the state vector  $\hat{x}_{k+1}$  for the next time step can be obtained as

$$\hat{x}_{k+1} = \omega_m^T \phi(v_m^T x_{mk}), \tag{34}$$

where  $\phi(\cdot) \in R^{N_m}$  is the activation function, which satisfies

$$\phi(a) = \frac{1 - \exp(-a)}{1 + \exp(-a)}. \tag{35}$$

Set the error function as

$$e_{mk} = \hat{x}_{k+1} - x_{k+1} \tag{36}$$

and the objective error function as

$$E_{mk} = \frac{1}{2} e_{mk}^T e_{mk}. \tag{37}$$

In the iterative training of the model network, the weights are updated based on the gradient descent rule, which are

$$\omega_{m(k+1)} = \omega_{mk} - \vartheta_m \left[ \frac{\partial E_{mk}}{\partial \omega_{mk}} \right], \tag{38}$$

$$v_{m(k+1)} = v_{mk} - \vartheta_m \left[ \frac{\partial E_{mk}}{\partial v_{mk}} \right], \tag{39}$$

where  $\vartheta_m$  is the learning rate. Notice that the weights are kept unchanged after a sufficient training and will be used in the following training.

### 4.2.2 Critic network

It is well known that the critic network of the HDP structure learns the information of the cost function  $V(x_k)$  and the critic network of the DHP structure learns the information of the partial derivatives  $\partial V(x_k)/\partial x_k$  of the cost function. However, in the GDHP structure, the critic network not only learns the information of the cost function  $V(x_k)$ , but also studies the knowledge of its partial derivatives  $\partial V(x_k)/\partial x_k$ . Because the GDHP structure learning more information about the system,

the more excellent control performance can be obtained by this method. For simplicity, the partial derivative is denoted as  $\lambda(x_k) = \partial V(x_k)/\partial x_k$ .

In the critic network, the outputs can be obtained as

$$\widehat{V}(x_{k_i}) = \omega_c^{VT} \phi(v_c^T x_{k_i}), \tag{40}$$

$$\widehat{\lambda}(x_{k_i}) = \omega_c^{\lambda T} \phi(v_c^T x_{k_i}), \tag{41}$$

where  $v_c$  represents the weight matrix of input-to-hidden layer, and  $\omega_c$  represents the weight matrix of hidden-to-output layer. The target function can be expressed as

$$V(x_{k_i}) = U_k + \widehat{V}(\hat{x}_{k_i+1}), \tag{42}$$

$$\begin{aligned} \lambda(x_{k_i}) = & 2Qx_k + 2 \left( \frac{\partial \mu(x_{k_i})}{\partial x_{k_i}} \right) \overline{U} R \varphi^{-1}(\overline{U}^{-1} \widehat{\mu}(x_{k_i})) \\ & + \left( \frac{\partial \hat{x}_{k_i+1}}{\partial x_{k_i}} + \frac{\partial \hat{x}_{k_i+1}}{\partial \widehat{\mu}(x_{k_i})} \frac{\partial \widehat{\mu}(x_{k_i})}{\partial x_{k_i}} \right)^T \\ & \times \widehat{\lambda}(\hat{x}_{k_i+1}). \end{aligned} \tag{43}$$

Hence, the error function can be obtained as

$$e_{ck}^V = \widehat{V}(x_{k_i}) - V(x_{k_i}), \tag{44}$$

$$e_{ck}^\lambda = \widehat{\lambda}(x_{k_i}) - \lambda(x_{k_i}). \tag{45}$$

The minimized objective error function is

$$E_{ck} = (1 - \rho) \left( \frac{1}{2} e_{ck}^{VT} e_{ck}^V \right) + \rho \left( \frac{1}{2} e_{ck}^{\lambda T} e_{ck}^\lambda \right). \tag{46}$$

According to (46) and the gradient descent rule, the weights of the critic network are updated as

$$\omega_{c(k+1)} = \omega_{ck} - \vartheta_c \left[ (1 - \rho) \frac{\partial E_{ck}^V}{\partial \omega_{ck}} + \rho \frac{\partial E_{ck}^\lambda}{\partial \omega_{ck}} \right], \tag{47}$$

$$v_{c(k+1)} = v_{ck} - \vartheta_c \left[ (1 - \rho) \frac{\partial E_{ck}^V}{\partial v_{ck}} + \rho \frac{\partial E_{ck}^\lambda}{\partial v_{ck}} \right], \tag{48}$$

where  $\vartheta_c > 0$  is the learning rate.  $0 \leq \rho \leq 1$  is a constant that reflects the weights of HDP and DHP in GDHP structure, which means that when  $\rho = 0$ , the structure reduces to a pure HDP, and when  $\rho = 1$ , the used structure reduces to a pure DHP.

### 4.2.3 Action network

In the action network, the sampled state  $x_{k_i}$  is taken as the input and the obtained  $\widehat{\mu}(x_{k_i})$  is used to approxi-

mate  $\mu(x_{k_i})$ . With the activation function  $\phi(\cdot) \in R^{N_m}$ ,  $\widehat{\mu}(x_{k_i})$  can be formulated as

$$\widehat{\mu}(x_{k_i}) = \omega_a^T \phi(v_a^T x_{k_i}), \tag{49}$$

where  $v_a$  represents the weight matrix of input-to-hidden layer, and  $\omega_a$  represents the weight matrix of hidden-to-output layer. According to (10), the target control input at the sampling instant  $k_i$  can be obtained as

$$\mu(x_{k_i}) = \bar{U}\varphi\left(-\frac{1}{2}(\bar{U}R)^{-1}g^T(x_{k_i})\frac{\partial V^*(x_{k_i+1})}{\partial x_{k_i+1}}\right). \tag{50}$$

Define the error function  $e_{ak}$  and the objective error function  $E_{ak}$  as

$$e_{ak} = \widehat{\mu}(x_{k_i}) - \mu(x_{k_i}), \tag{51}$$

$$E_{ak} = \frac{1}{2}e_{ak}^T e_{ak}. \tag{52}$$

Similarly, the weights updating rule of the action network can be formulated as

$$\omega_{a(k+1)} = \omega_{ak} - \vartheta_a \left[ \frac{\partial E_{ak}}{\partial \omega_{ak}} \right], \tag{53}$$

$$v_{a(k+1)} = v_{ak} - \vartheta_a \left[ \frac{\partial E_{ak}}{\partial v_{ak}} \right], \tag{54}$$

where  $\vartheta_a > 0$  is the learning rate.

### 4.3 Approximate optimal algorithm design

Define  $V_\infty(x_k) = \lim_{l \rightarrow \infty} V_l(x_k)$ , if the system state  $x_k$  is controllable, the cost function  $V_\infty(x_k)$  is equal to the optimal cost function  $V^*(x_k)$

$$\lim_{l \rightarrow \infty} V_l(x_k) = V^*(x_k), \tag{55}$$

where  $l$  is the outer loop iteration index. As  $l \rightarrow \infty$ , we have  $V_l(x_k) \rightarrow V^*(x_k)$  in theory. However, it is not possible to perform iterations indefinitely in the actual calculation process. Thus, we introduce an error  $\varepsilon$  to make the cost function  $V(x_k)$  converge after a finite number of iterations [38]. That is to say, there is a limited  $l$  that can make the cost function  $V_l(x_k)$  satisfy

$$|V^*(x_k) - V_l(x_k)| \leq \varepsilon. \tag{56}$$

In the iterative ADP algorithm, this design achieves the purpose of approximate optimal regulation. However, the optimal cost function  $V^*(x_k)$  is unknown in general, it is difficult to use the termination criterion

(56) to verify whether the iterative algorithm satisfies the requirements. So, we use the following criterion

$$|V_{l+1}(x_k) - V_l(x_k)| \leq \varepsilon \tag{57}$$

to replace (56).

## 5 Simulation results and analysis

In this section, the event-triggered GDHP method is applied in three discrete-time systems. Simulations show the advantages of the proposed method by comparing with the traditional GDHP method.

### 5.1 Case 1: two-dimensional system

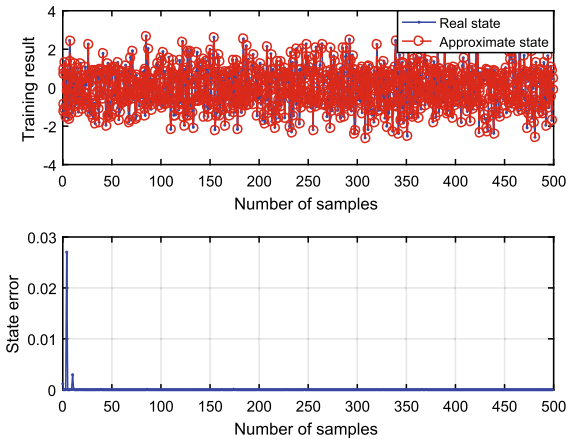
Consider the following discrete-time system:

$$x_{k+1} = \begin{bmatrix} x_{1k} + 0.1x_{2k} \\ -2x_{1k} + 0.7x_{2k} \end{bmatrix} + \begin{bmatrix} 0 \\ x_{1k} \end{bmatrix} u_k, \tag{58}$$

where  $x_k = [x_{1k}, x_{2k}]^T \in \mathbb{R}^2$  is the state vector and  $u_k \in \mathbb{R}$  is the control input vector. The initial state vector is set as  $x_0 = [-1, 1]$  and the boundary of the saturated actuator is chosen as  $|u| \leq 0.1$ . Let the parameters  $Q = I_2$  and  $R = I$ , where the subscript represents the dimensions of the identity matrix. Based on (21), we set  $P_1 + P_2\bar{u} = 0.2$ . The event-triggered threshold can be obtained as

$$e_T = \frac{1 - 0.4^{k-k_i}}{1 - 0.4} \cdot 0.2 \|x_{k_i}\|. \tag{59}$$

The model network needs to be pre-trained to obtain the system dynamic before implementing the proposed method. The structure of model network is designed as 3–8–2 and the learning rate is set as  $\vartheta_m = 0.1$ . It is well known that the setting of parameters will affect the convergence speed and control effect of the algorithm to a certain extent. In order to obtain a good control performance, the initial weights of the three networks are randomly selected from  $[-0.1, 0.1]$  after multiple experiments, which enables the algorithm to have a high control accuracy. To obtain sufficient system dynamic, 500 sets of data are randomly selected from  $[-1, 1]$  for training. Model network is trained for 50 time steps on each sample and the training performance is shown in Fig. 2. From Fig. 2, we can see that the training error has a large value at the beginning, as the samples increase, the training error becomes smaller and eventually converges to zero. After the training,

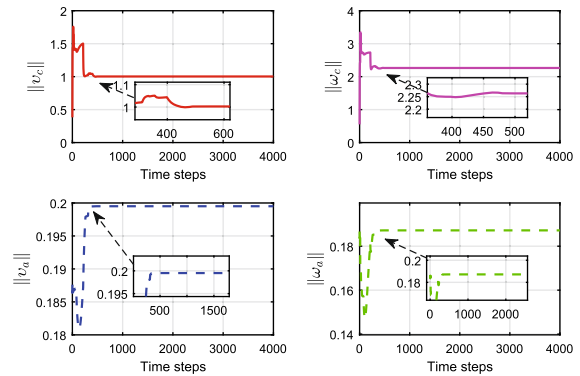


**Fig. 2** The training result and the state error of model network

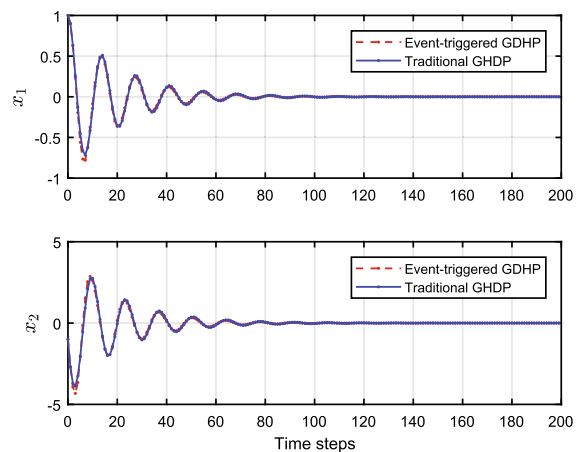
the obtained weights will be kept unchanged for the following training.

Next, the critic network and the action network are designed with structures 2–8–3 and 2–8–1. The initial weights of the two networks are all randomly generated within  $[-0.5, 0.5]$  and the adjusting parameter is set as  $\rho = 0.5$ . The learning rates are chosen as  $\vartheta_c = 0.01$  and  $\vartheta_a = 0.1$ . During iterative process, each network is trained for 200 inner-loop steps with each iteration of 4000 training steps. In addition, we set  $\varepsilon = 10^{-6}$  as the termination condition for each state, which ensures the control law  $u$  is an approximate optimal control. As shown in Fig. 3, the weight norms of the two networks are all convergent after the training. It should be noted that due to the weights of Fig. 3 are updated on line. During an iterative process Of state variables, the weights are updated with the iterative calculation until the termination criterion is satisfied. In the entire iterative update process, it can be seen that the iteration is performed on 4000 training steps.

In order to prove the effectiveness of the proposed method, the traditional GDHP method is also applied in this example to make a contrast. The state trajectories and the control input curves in 200 time steps under the two methods are shown in Figs. 4 and 5. As can be seen in Fig. 5, compared with the traditional GDHP method, our method overcomes the constrained-input problem. The trigger error  $\|e_k\|$  and the trigger threshold  $e_T$  are given in Fig. 6. The error  $\|e_k\|$  between the current state and the sampled state is calculated in each sampling process, if  $\|e_k\| > e_T$ ,  $\|e_k\|$  will be reset to zero. In this simulation, the action network of traditional GDHP



**Fig. 3** The norm of weight matrix for action and critic networks with GDHP method



**Fig. 4** State trajectories with the traditional GDHP method and the event-triggered GDHP method

method needs 200 samples to update the control input, while the event-triggered GDHP method only needs 82 samples.

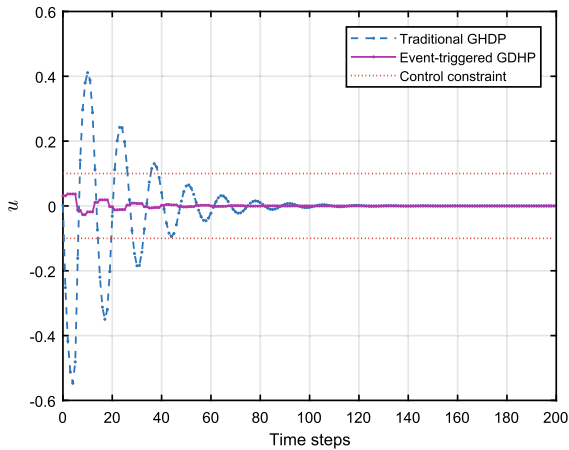
### 5.2 Case 2: three-dimensional system

Consider the following discrete-time affine nonlinear system presented in [39]:

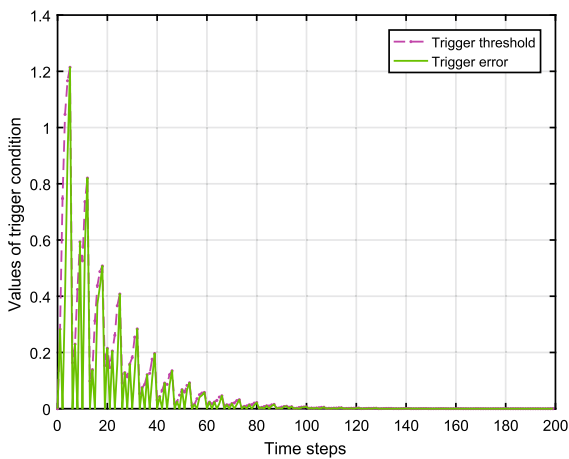
$$x_{k+1} = \begin{bmatrix} x_{1k}x_{2k} \\ x_{1k}^2 - 0.5 \sin(x_{2k}) \\ x_{3k} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ -1 \end{bmatrix} u_k, \quad (60)$$

where the state vector  $x_k = [x_{1k}, x_{2k}, x_{3k}]^T \in \mathbb{R}^3$ . Set the initial state vector as  $x_0 = [-0.5, 0.5, 1]^T$  and the saturating boundary as  $|u| \leq 0.1$ . The performance index function is chosen the same as case 1 with  $Q = I_n$  and  $R = I_m$ . The termination condition error is set as





**Fig. 5** Control input trajectories  $u$  with the traditional GHDP method and the event-triggered GHDP method



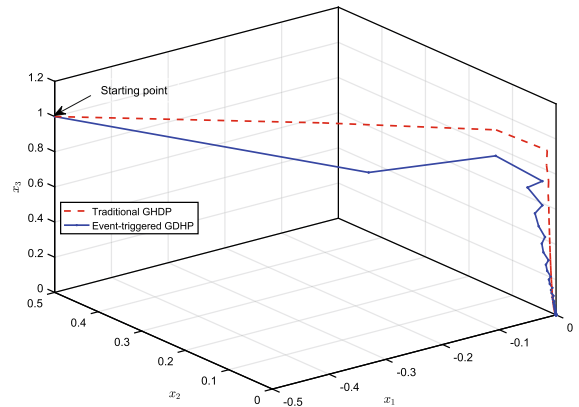
**Fig. 6** Trigger error and trigger threshold

$\varepsilon = 10^{-4}$ . In this case, we let  $P_1 + P_2\bar{u} = 0.1$ . So, the trigger threshold can be obtained as

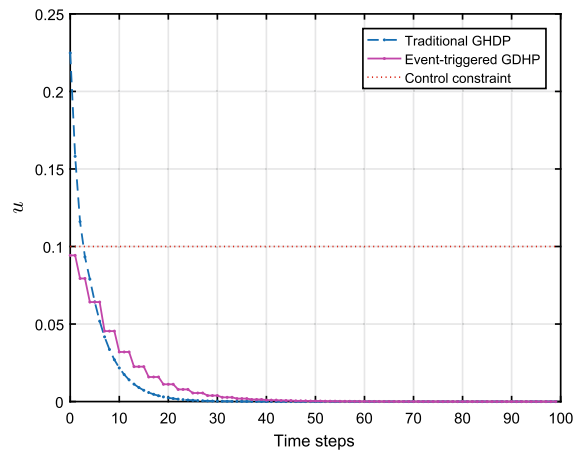
$$e_T = \frac{1 - 0.2^{k-k_i}}{1 - 0.2} \cdot 0.1 \|x_{k_i}\|. \tag{61}$$

The structures of model, critic and action networks are designed as 4–8–3, 3–8–1, and 3–8–1. All the initial weights of three networks are randomly set in  $[-0.1, 0.1]$ , and other parameters are designed the same as case 1. Similar to case 1, the model network is pre-trained at first and the unchanged weights are used to train the critic network and action network.

In this case, we also compare the traditional GHDP method with the event-triggered GDHP method. The state trajectories in 100 time steps under the two different methods are shown in Fig. 7. According to the

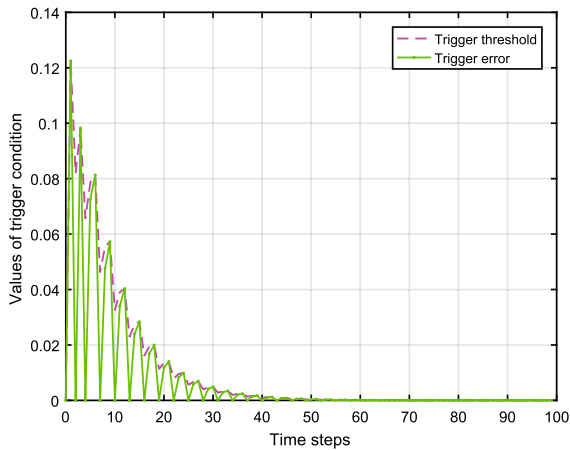


**Fig. 7** State trajectories  $x$  with the traditional GHDP method and the event-triggered GHDP method

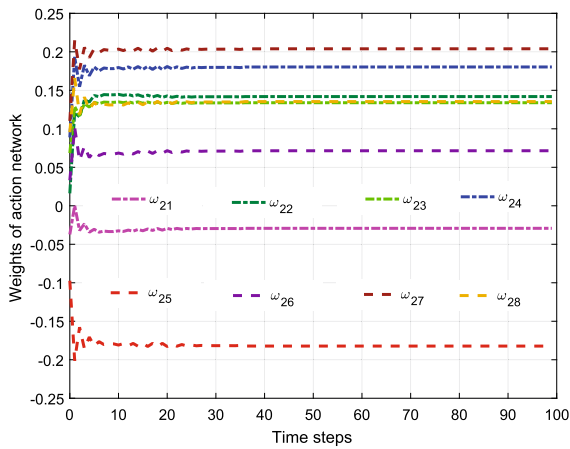


**Fig. 8** Control input trajectories  $u$  with the traditional GHDP method and the event-triggered GHDP method

state trajectories, we can get that the two methods have similar performance. Figure 8 shows the control input trajectories under the two methods. As can be seen from Fig. 8, the proposed method overcomes the control constraints as well as reduces the computing resources. The event-triggered error and trigger threshold are shown in Fig. 9, where we can see that the action network of traditional GHDP method is updated 100 times, while in the proposed method, the action network is only updated 34 times. Besides, the updated weights process of the action network from hidden-to-output layer are shown in Fig. 10. Due to the weights in Fig. 10 obtained through the external iteration, which are only updated at the sampling instant, the iterative time steps of the action network are the same as the time steps of state trajectory.



**Fig. 9** Trigger error and trigger threshold



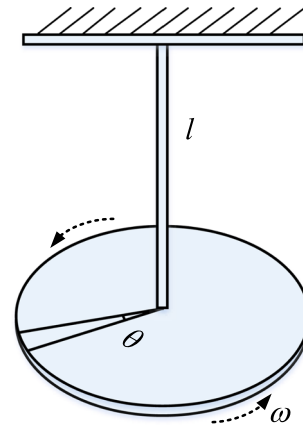
**Fig. 10** The weights of action network from hidden-to-output layer

### 5.3 Case 3: torsional pendulum system

In this case, we apply the event-triggered GDHP method to the dynamical system of torsional pendulum, whose mechanical model is shown in Fig. 11. The mathematical description of this system is as follows [40]:

$$\begin{cases} \frac{d\theta}{dt} = \omega \\ G \frac{d\omega}{dt} = u - Mgl \sin \theta - f_d \frac{d\theta}{dt}, \end{cases} \quad (62)$$

where  $M = 1/3 \text{ kg}$  is the mass,  $l = 2/3 \text{ m}$  and  $G = 4/3Ml^2$  are the length of the pendulum bar and the rotary inertia, respectively. Let  $f_d = 0.2$  be the friction factor and  $g = 9.8 \text{ m/s}^2$  be the acceleration



**Fig. 11** The mechanical model of the torsional pendulum

of gravity. The angle  $\theta$  and the angular velocity  $\omega$  are the inputs of the system.

Using the sampling interval  $\Delta t = 0.1 \text{ s}$ , the dynamic function of the torsional pendulum system can be discretized as

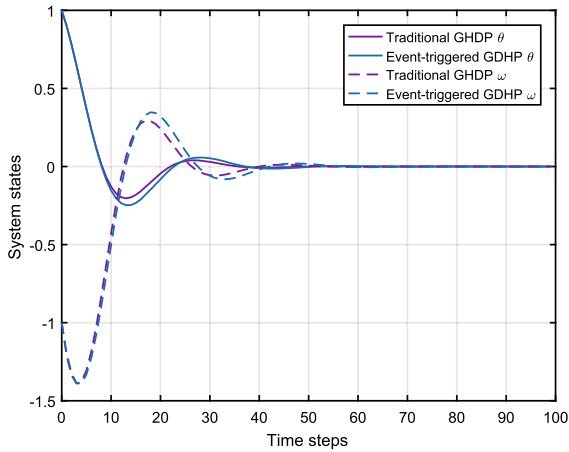
$$x_{k+1} = \begin{bmatrix} 0.1x_{2k} + x_{1k} \\ -0.49 \sin(x_{1k}) - 0.1f_d \cdot x_{2k} + x_{2k} \end{bmatrix} + \begin{bmatrix} 0 \\ 0.1 \end{bmatrix} u_k,$$

where  $x_{1k} = \theta$  and  $x_{2k} = \omega$ . The initial state and the control constraint are set as  $x_0 = [-1, 1]^T$ ,  $|u| \leq 0.3$ , respectively. The structures of model, critic and action networks are designed as 3–8–2, 2–8–1, and 2–8–1. The termination condition error is set as  $\varepsilon = 10^{-3}$ . Besides, the trigger threshold is chosen the same as case 2 and all other parameters are the same as case 1.

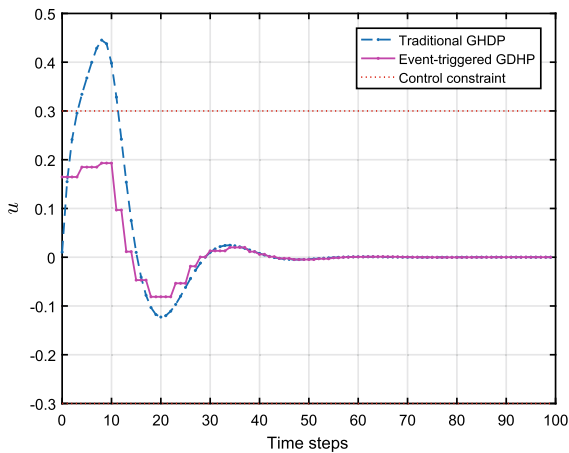
In this case, the traditional GHDP method and the proposed method are applied in torsional pendulum system. The state trajectories in 100 time steps under the two method are shown in Fig. 12, which indicates the control performance of the proposed method is similar to that of the traditional GHDP method. However, the event-triggered GDHP method is able to constrain the control input to a certain range as shown in Fig. 13. To further illustrate the effectiveness of the proposed algorithm, the trigger threshold and the event-triggered error are given in Fig. 14. Compared with the traditional time-triggered method with the requirement of 100 samples, the event-triggered method only needs 34 samples, representing a save of 66%.

## 6 Conclusion

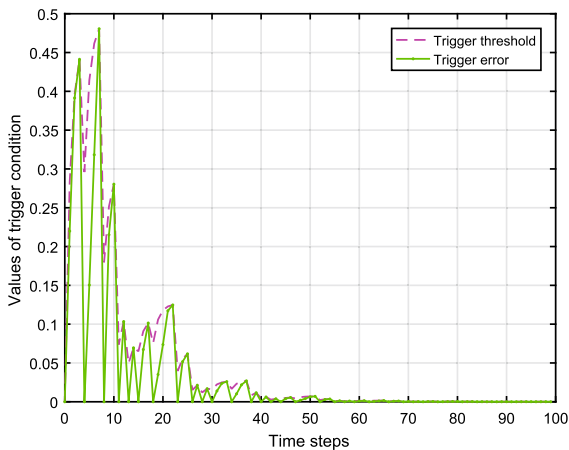
In this paper, a novel event-triggered method based on GDHP technique is proposed for a class of discrete-



**Fig. 12** State trajectories with the traditional GHDP method and the event-triggered GHDP method



**Fig. 13** Control input trajectories  $u$  with the traditional GHDP method and the event-triggered GHDP method



**Fig. 14** Trigger error and trigger threshold

time systems with control constraints. In order to solve the constrained-input problem, a non-quadratic performance index is introduced in the utility function. Additionally, we give a trigger threshold and use Lyapunov technique to prove the stability of the event-triggered systems. Then, the NN implementation based on GDHP technique is given and an iterative termination criterion is also designed to obtain the approximate optimal control. Finally, three cases are given to demonstrate the effectiveness of the event-triggered GDHP method by comparing with the traditional GDHP method. According to the simulation results from case 1, case 2 and case 3, it can be known that compared with the traditional time-triggered method, the event-triggered method saves 59%, 66% and 66% of computing resources, respectively. Therefore, it can be concluded that the event-triggered GDHP method can solve the constrained-input problem and reduce computing resources while ensuring system performance.

**Acknowledgements** This work was supported in part by the National Natural Science Foundation of China under Grants 62022061, 61773284, and in part by Tianjin Natural Science Foundation under Grant 20JCYBJC00880.

**Compliance with ethical standards**

**Conflict of interest** The authors declare that they have no conflict of interest.

**Ethical approval** No conflict of interest exists in this submission, and the research work does not involve any human participants and/or animals. The manuscript is approved by all authors for publication.

**References**

1. Zhao, L., Xiong, H., Zheng, Z., Li, Q.: Improving worst-case latency analysis for rate-constrained traffic in the time-triggered ethernet network. *IEEE Commun. Lett.* **18**(11), 1927–1930 (2014)
2. Donkers, M., Heemels, W.: Output-based event-triggered control with guaranteed  $\mathcal{L}_\infty$ -gain and improved and decentralized event-triggering. *IEEE Trans. Autom. Control* **57**(6), 1362–1376 (2012)
3. Molin, A., Hirche, S.: On the optimality of certainty equivalence for event-triggered control systems. *IEEE Trans. Autom. Control* **58**(2), 470–474 (2013)
4. Zhang, X., Han, Q., Zhang, B.: An overview and deep investigation on sampled-data-based event-triggered control and filtering for networked systems. *IEEE Trans. Ind. Inf.* **13**(1), 4–16 (2017)

5. Zhang, H., Han, J., Wang, Y., Jiang, H.:  $H_\infty$  consensus for linear heterogeneous multiagent systems based on event-triggered output feedback control scheme. *IEEE Trans. Cybern.* **49**(6), 2268–2279 (2019)
6. Ma, Y., Wu, W., Görges, D., Cui, B.: Event-triggered feedback control for discrete-time piecewise affine systems subject to input saturation. *Nonlinear Dyn.* **95**(3), 2353–2365 (2019)
7. Zhang, J., Feng, G.: Event-driven observer-based output feedback control for linear systems. *Automatica* **50**, 1852–1859 (2014)
8. Tabuada, P.: Event-triggered real-time scheduling of stabilizing control tasks. *IEEE Trans. Autom. Control* **52**(9), 1680–1685 (2007)
9. Tallapragada, P., Chopra, N.: On event triggered tracking for nonlinear systems. *IEEE Trans. Autom. Control* **58**(9), 2343–2348 (2013)
10. Qiu, J., Sun, K., Wang, T., Gao, H.: Observer-based fuzzy adaptive event-triggered control for pure-feedback nonlinear systems with prescribed performance. *IEEE Trans. Fuzzy Syst.* **27**(11), 2152–2162 (2019)
11. Heemels, W.P.M.H., Donkers, M.C.F., Teel, A.R.: Periodic event-triggered control for linear systems. *IEEE Trans. Autom. Control* **58**(4), 847–861 (2013)
12. Yang, D., Li, T., Xie, X., Zhang, H.: Event-triggered integral sliding-mode control for nonlinear constrained-input systems with disturbances via adaptive dynamic programming. *IEEE Trans. Syst. Man Cybern. Syst.* **50**(11), 4086–4096 (2020)
13. Mu, C., Wang, K., Sun, C.: Policy-Iteration-Based Learning for Nonlinear Player Game Systems With Constrained Inputs. *IEEE Trans. Syst. Man Cybern. Syst.* (in press). <https://doi.org/10.1109/TSMC.2019.2962629>
14. Luo, B., Yang, Y., Liu, D., Wu, H.: Event-triggered optimal control with performance guarantees using adaptive dynamic programming. *IEEE Trans. Neural Netw. Learn. Syst.* **31**(1), 76–88 (2020)
15. Zhong, X., He, H.: An event-triggered ADP control approach for continuous-time system with unknown internal states. *IEEE Trans. Cybern.* **47**(3), 683–694 (2017)
16. Shi, J., Yue, D., Yang, Y., Hu, S.: Event-triggered control based on adaptive dynamic programming for continuous-time nonlinear systems with completely unknown dynamics. In: 2016 12th World Congress on Intelligent Control and Automation, pp. 2035–2040 (2016)
17. Yang, Y., Xu, C., Meng, Q., Tan, J.: An event-triggered ADP controller for single link robot arm system based on output position. In: 2018 Chinese Control And Decision Conference, pp. 2271–2276 (2018)
18. Mu, C., Wang, K.: Single-network ADP for near optimal control of continuous-time zero-sum games without using initial stabilising control laws. *IET Control Theory Appl.* **12**(18), 2449–2458 (2018)
19. Werbos, P. J.: Advanced forecasting methods for global crisis warning and models of intelligence. *Gen. Syst. Yearb.* **22**, 25–38 (1977)
20. John, M., Chadwick, C., George, L., Richard, S.: Applications and reviews. *IEEE Trans. Syst. Man Cybern. Part C* **32**, 140–153 (2002)
21. Jiang, Y., Jiang, Z.: Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics. *Automatica* **48**, 2699–2704 (2012)
22. Al-Tamimi, A., Lewis, F.L., Abu-Khalaf, M.: Discrete-time nonlinear HJB solution using approximate dynamic programming: convergence proof. *IEEE Trans. Syst. Man Cybern. Part B* **38**(4), 943–949 (2008)
23. Zhang, H., Wei, Q., Luo, Y.: 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 Cybern. Part B Cybern.* **38**(4), 937–942 (2008)
24. Prokhorov, D., Wunsch, D.: Adaptive critic designs. *IEEE Trans. Neural Netw.* **8**, 997–1007 (1997)
25. Eun, Y., Kabamba, P.T., Meerkov, S.M.: Analysis of random reference tracking in systems with saturating actuators. *IEEE Trans. Autom. Control* **50**(11), 1861–1866 (2005)
26. Mu, C., Wang, K.: Approximate-optimal control algorithm for constrained zero-sum differential games through event-triggering mechanism. *Nonlinear Dyn.* **95**, 2639–2657 (2019)
27. Eun, Y., Kabamba, P.T., Meerkov, S.M.: System types in feedback control with saturating actuators. *IEEE Trans. Autom. Control* **49**(2), 287–291 (2004)
28. He, W., Ge, S.: Vibration control of a flexible string with both boundary input and output constraints. *IEEE Trans. Control Syst. Technol.* **23**(4), 1245–1254 (2015)
29. Na, J., Wang, B., Li, G., Zhan, S., He, W.: Nonlinear constrained optimal control of wave energy converters with adaptive dynamic programming. *IEEE Trans. Ind. Electron.* **66**(10), 7904–7915 (2019)
30. Fan, B., Yang, Q., Tang, X., Sun, Y.: Robust ADP design for continuous-time nonlinear systems with output constraints. *IEEE Trans. Neural Netw. Learn. Syst.* **29**(6), 2127–2138 (2018)
31. Zhu, Y., Zhao, D., He, H., Ji, J.: Event-triggered optimal control for partially unknown constrained-input systems via adaptive dynamic programming. *IEEE Trans. Ind. Electron.* **64**(5), 4101–4109 (2017)
32. Zhang, L., Chen, M.: Event-based global stabilization of linear systems via a saturated linear controller. *Int. J. Robust Nonlinear Control* **26**(5), 1073–1091 (2016)
33. Ni, W., Zhao, P., Wang, X.: Event-triggered control of linear systems with saturated inputs. *Asian J. Control* **17**(4), 1196–1208 (2015)
34. Lewis, F.L., Syrmos, V.: *Optimal Control*. Wiley, New York (1995)
35. Abu-Khalaf, M., Lewis, F.L.: Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach. *Automatica* **41**(5), 779–791 (2005)
36. Eqtami, A., Dimarogonas, D.V., Kyriakopoulos, K.J.: Event-triggered control for discrete-time systems. In: Proceedings of the 2010 American Control Conference, pp. 4719–4724 (2010)
37. Jiang, Z.P., Wang, Y.: Input-to-state stability for discrete-time nonlinear systems. *Automatica* **37**(6), 857–869 (2001)

38. Wang, F., Jin, N., Liu, D., Wei, Q.: Adaptive dynamic programming for finite-horizon optimal control of discrete-time nonlinear systems with  $\varepsilon$ -error bound. *IEEE Trans. Neural Netw.* **22**(1), 24–36 (2011)
39. Mu, C., Wang, D., He, H.: Novel iterative neural dynamic programming for data-based approximate optimal control design. *Automatica* **81**, 240–252 (2017)
40. Liu, D., Wei, Q.: Policy iteration adaptive dynamic programming algorithm for discrete-time nonlinear systems. *IEEE Trans. Neural Netw. Learn. Syst.* **25**(3), 621–634 (2014)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.