KYBERNETIKA — VOLUME 57 (2021), NUMBER 3, PAGES 546–566

NEURAL NETWORK OPTIMAL CONTROL FOR NONLINEAR SYSTEM BASED ON ZERO-SUM DIFFERENTIAL GAME

Fu Xingjian and Li Zizheng

In this paper, for a class of the complex nonlinear system control problems, based on the two-person zero-sum game theory, combined with the idea of approximate dynamic programming(ADP), the constrained optimization control problem is solved for the nonlinear systems with unknown system functions and unknown time-varying disturbances. In order to obtain the approximate optimal solution of the zero-sum game, the multilayer neural network is used to fit the evaluation network, the execution network and the disturbance network of ADP respectively. The Lyapunov stability theory is used to prove the uniform convergence, and the system control output converges to the neighborhood of the target reference value. Finally, the simulation example verifies the effectiveness of the algorithm.

Keywords: zero-sum game, nonlinear system, neural network, approximate dynamic programming

Classification: 93C10, 93D21, 91A80

1. INTRODUCTION

Game Theory is a mathematical method for studying the equilibrium of multiplayer strategies, and it is an important branch of modern economics and operations research. In 1944, John von Neumann and Oskar Morgenstern systematically defined game theory in their book "Theory of Games and Economic Behavior" [22], which marked the establishment of game theory as a discipline. The prisoner's dilemma problem raised by Albert Tucker in 1950 became one of the most classic cases in non-cooperative games. In 1950, "Equilibrium points in n-person games" published by John Nash formally defined the concept of equilibrium and used the fixed point theory to prove the existence of equilibrium points [20]. A year later, Nash's doctoral dissertation "Non-cooperativegames" was published in "Annals of mathematics" [21]. His important research result, "Nash equilibrium" theory, laid the foundation for the development of game theory.

Equilibrium is the most important concept for solving game theory problems. The key to solve the game problem is to find a strategy combination that can make all the players in the game unwillingness to change their strategies individually. As long as this strategy exists, there is a solution to the game problem. If this strategy combination is

DOI: 10.14736/kyb-2021-3-0546

unique, then the solution of the game is uniquely determined. In the game, the strategy adopted by any participant for their own best interests should be the best response to the strategies of other participants. The strategy combination in which each player in the game is unwilling or will not change their strategy individually is the solution of the game, which is the Nash equilibrium [11, 17, 21, 31]. In the Nash equilibrium, no player in the game can obtain more benefits by individually changing his own strategy while the other participants keep their strategies unchanged.

In the 1950s, the team of Dr. R. Isaacs conducted research on the problem of the chasing and escaping where everyone in the game can determine the action plan independently. In 1965, he published the monograph "Different Game" by sorting out research results, which was the first monograph on the theory of differential game [10]. In game theory, by drawing on the tools and concepts of modern control theory, differential game theory can solve dynamic decision problems well. Differential game theory is a dynamic strategy, which uses differential equations or equations to describe the internal laws of competitive relations. The theoretical research results based on differential game theory play an important role in the field of automatic control and game theory. Many scholars continued to study and excavate this field. In [15], a greedy heuristic dynamic programming iteration algorithm is developed to solve the zero-sum game problems, which can be used to solve the Hamilton–Jacobi–Isaacs equation associated with H_∞ optimal regulation control problems. In [4], for the attitude control problem of a failed spacecraft with exhausted fuel, a state-dependent Riccati equation (SDRE) differential game control method in which multiple microsatellites cooperate to achieve attitude stability is proposed. In [14], the problem of two-person Nash differential games for delayed stochastic systems with state-and control-dependent noise is discussed. A sufficient condition for the existence of the Nash equilibrium strategy is presented in terms of coupled Hamilton-Jacobi equations (HJEs). Due to disadvantages of single canard fin control or tail fin control for bounded-control interception missiles, a novel dual and bounded controlled differential game guidance law is presented based on two-sided optimization differential game theory [9]. In [[25], in order to overcome the difficulty in real-time effectively acquiring the target parameters of differential game guidance in a complex underwater environment, the differential game guidance of underwater nonlinear tracking control based on continuous time generalized predictive correction is proposed. In [16], an online optimal distributed learning algorithm is proposed to solve leader-synchronization problem of nonlinear multi-agent differential graphical games. Each player approximates its optimal control policy using a single-network ADP. In [19], the optimization problem of the two-person zero-sum difference game with control constraints under the event trigger framework is studied. Relying on reinforcement learning, an adaptive dynamic programming algorithm is developed to approximate the optimal solution of the zero-sum game. In [33], two distributed dynamic optimization structures, receding non-cooperative game optimization (RNGO) and receding cooperative game optimization (RCGO), are presented for analyzing distributed dynamic optimization for chemical process networks. In [18], the risk-sensitive optimal control and differential game of the stochastic differential delay equation driven by Brownian motion are considered. In [26], an approximate optimal critic learning algorithm based on single neural network strategy iteration was established to solve continuous-time two-

person zero-sum games. In [8], the system application of the hybrid system framework in differential games is studied. Two types of special switching rules: time-related and state-related switching are discussed. In [19], the authors solve the two-person zero-sum difference game problem of a nonlinear dynamic system with a nonlinear non-quadratic cost function. The goal of the tracker is to minimize the cost function and ensure the asymptotic stability of the closed-loop system.

The optimal control for the nonlinear systems is the difficult and challenging topic. Dynamic programming (DP) is a very useful method to solve the optimal control problem. In particular, it can be easily applied to the nonlinear systems where the control input and state variables are not constrained.The theory of dynamic programming was founded by Bellman [2]. He developed the HJB theory in variational science by relying on the optimality principle, and finally formed the DP theory. DP theory can deal with optimal control problems under constrained and unconstrained conditions [6, 7, 23, 24]. However, with the increase in the operating speed of computer storage space, people have entered the era of big data. The limitations of DP have gradually emerged. Under small-scale calculations, the DP theory does have its own unique advantages, but as the dimensionality of the data increases, the amount of DP calculations also increases exponentially. This is the so-called "Dimension disaster" problem. In order to solve this problem, Werbos proposed the idea of approximate/adaptive dynamic programming (ADP) in 1977, which is a new nonlinear optimization method [28]. ADP uses the function approximation structure to approximate the cost function and control strategy in the DP equation, so that the optimal cost function and optimal control strategy are obtained, so the limitations of the DP method are overcome. Therefore, the ADP method can solve the optimal control problem of general nonlinear systems and is more suitable for applications in systems with strong coupling and high complexity. Subsequently, many scholars used ADP to deal with the optimal control problem of the nonlinear systems [3, 5, 27, 29, 30]. In [27], a value iteration adaptive dynamic programming (ADP) algorithm is developed to solve infinite horizon undiscounted optimal control problems for discrete-time nonlinear systems. The present value iteration ADP algorithm permits an arbitrary positive semi-definite function to initialize the algorithm. In [3], a novel optimal control design scheme is proposed for continuous-time nonaffine nonlinear dynamic systems with unknown dynamics by adaptive dynamic programming (ADP). The proposed methodology iteratively updates the control policy online by using the state and input information without identifying the system dynamics. In [29], the influence of time delay on system stability and controller design is studied, and heuristic dynamic programming theory is introduced into iterative algorithms to solve the optimal control problem of the system. A nonlinear robust optimal control (NROC) scheme for uncertain two-axis motion control systems through adaptive dynamic programming (ADP) and neural networks (NNs) is proposed in [5]. Nowadays, based on the idea of two-person zero-sum game, there are relatively few studies on the optimal control for the nonlinear systems with disturbances.

At present, many research results have been obtained by adopting adaptive control methods to deal with nonlinear system problems [32, 13, 32]. In [12], this paper presents an adaptive control method for a class of uncertain strict-feedback switched nonlinear systems. Based on the backstepping technique, the integral Barrier Lyapunov functions

(iBLFs) are adopted to solve the full state constraint problems. In [32], this paper presents an adaptive neural network output feedback control method for stochastic nonlinear systems with full state constraints. The barrier Lyapunov functions are used to conquer the effect of state constraints to system performance. In the control process of nonlinear systems, there are a lot of uncertain disturbances and uncontrollable factors. For example, the disturbance of the external environment or the aging of the internal components of the system, these factors are not artificially controllable. The main idea of the zero-sum game is that both sides of the game are trying to eliminate the other, and the benefits that one party gets are exactly what the other party loses. According to the characteristics of the zero-sum game, the control input of the nonlinear system can be regarded as one party of the game, and the other party can be regarded as the uncertain disturbance of the outside world. The disturbances attempt to destroy the balance of the system, while the controlling party uses its own nature to eliminate the adverse effects of disturbance, which constitutes a game. Regardless of the form of external disturbance, it can be abstracted as a player. The game strategy is to design a strategy that allows the control input to overcome the external uncertain disturbance to achieve optimal control of the nonlinear system.

According to the dynamic programming theory, for the nonlinear systems, HJB partial differential equations need to be solved when seeking the optimal feedback control. It is often difficult. Sometimes, it is impossible to get an analytical solution. Usually, the system needs to be approximately processed. However, this kind of approximation is often idealized, which often leads to inaccurate results and loses the original features of the system. In this paper, the main contributions are as follows. For the complex nonlinear systems with unknown system functions and unknown time-varying disturbances, based on the two-person zero-sum game theory, combined with the idea of approximate dynamic programming, The constrained optimization control problem is solved. In order to obtain the approximate optimal solution of the HJI equation, the multilayer neural network is used to fit the evaluation network, the execution network and the disturbance network respectively. Through continuous network training to reduce the approximation error, the characteristics of the system model can be retained to the greatest extent. Assuming that the system is controllable, the Lyapunov stability theory is used to prove the uniformly convergence, and the system control output converges to the neighborhood of the target reference value. Compared with the existing methods, the proposed method reduces the requirement of parameter uncertainty conditions, which reduces the conservativeness of the optimal control. Finally, the simulation example verifies the effectiveness of the algorithm.

2. PROBLEM STATEMENT

Considering the following nonlinear discrete system with disturbances

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k) + \mathbf{g}(\mathbf{x}_k)\mathbf{u}_k + \mathbf{h}(\mathbf{x}_k)\mathbf{d}_k \tag{1}$$

where $k = 0, 1, 2, \ldots, n$, $\mathbf{x}_k \in R^n$ is the state, $\mathbf{f}(\mathbf{x}_k) \in \mathrm{R}^n$, $\mathbf{g}(\mathbf{x}_k) \in \mathrm{R}^{\mathrm{n}\times\mathrm{m}}$ and $\mathbf{h}(\mathbf{x}_k) \in \mathrm{R}^{\mathrm{n}\times\mathrm{q}}$ are the smooth and differentiable functions. $\mathbf{u}_k \subseteq \mathrm{R}^m$ is the control input, $\mathbf{d}_k \subseteq R^q$ is the disturbance input, and $\mathbf{d}_k \in L_2[0, \infty)$. The discrete system satisfies the characteristics of the Markov process. It is assumed that the system is controllable,

that is, there is at least one continuous control sequence $\mathbf{u}_k$ that can make the system asymptotically stable when there is a disturbance $\mathbf{d}_k$.

The stable control problem for the system (1) with disturbance can be regarded as a zero-sum differential game solving problem.The control input $\mathbf{u}_k$ is used as the minimization decision player. The disturbance signal $\mathbf{d}_k$ is used as the maximization decision player. According to the Nash equilibrium condition of the zero-sum differential game [11, 31], this problem can be converted to how to find the Nash equilibrium solution so that the minimized decision player $\mathbf{u}_k$ can balance the maximum decision player $\mathbf{d}_k$

In the zero-sum differential game, the infinite cost function is defined as follows

$$J(\mathbf{x}_0, \mathbf{u}, \mathbf{d}) = \sum_{k=0}^{\infty} U(\mathbf{x}_k, \mathbf{u}_k, \mathbf{d}_k), \tag{2}$$

where

$$U(\mathbf{x}_k, \mathbf{u}_k, \mathbf{d}_k) = \mathbf{x}_k^T \mathbf{Q} \mathbf{x}_k + \mathbf{u}_k^T \mathbf{R} \mathbf{u}_k - \gamma^2 \mathbf{d}_k^T \mathbf{S} \mathbf{d}_k. \tag{3}$$

For a controllable system, $U(\mathbf{x}_k, \mathbf{u}_k, \mathbf{d}_k)$ is a decreasing series, $U_k \geq U_{k+1}$. When $k \to \infty$, $U_k \to 0$ is a convergent series. The function $J(\mathbf{x}_0, \mathbf{u}, \mathbf{d})$ is the value of the performance index function in the state $\mathbf{x}_k$. By selecting a suitable control strategy, the $J(\mathbf{x}_0, \mathbf{u}, \mathbf{d})$ value can be minimized. The $\mathbf{Q}$,$\mathbf{R}$ and $\mathbf{S}$ are the positive definite matrices, and γ is a given constant.

Remark 1. For the optimal control of the system, the control strategy $\mathbf{u}(\mathbf{x}_k)$ is required to ensure that the system is stabilized and that the cost function is finite, which means that the control strategy $\mathbf{u}(\mathbf{x}_k)$ must be admissible control [24].

Definition 1. If the control strategy $\mathbf{u}(\mathbf{x}_k)$ can stabilize the system (1), $\mathbf{u}(0) = 0$, and for any initial state $\mathbf{x}_0$, $J(\mathbf{x}_0)$ is a finite value. When the disturbance $\mathbf{d}_k$ exists, the control strategy $\mathbf{u}(\mathbf{x}_k)$ is the admissible control.

For the feedback control strategy $\mathbf{u}(\mathbf{x}_k)$ and the disturbance strategy $\mathbf{d}_k$, the infinite value cost function is defined as

$$V(\mathbf{x}_k, \mathbf{u}_k, \mathbf{d}_k) = \sum_{i=k}^{\infty} U(\mathbf{x}_i, \mathbf{u}_i, \mathbf{d}_i). \tag{4}$$

The Hamilton function is

$$H(\mathbf{x}_k, \mathbf{u}_k, \mathbf{d}_k) = V(\mathbf{x}_{k+1}) - V(\mathbf{x}_k) + \mathbf{x}_k^T \mathbf{Q} \mathbf{x}_k + \mathbf{u}_k^T \mathbf{R} \mathbf{u}_k - \gamma^2 \mathbf{d}_k^T \mathbf{S} \mathbf{d}_k. \tag{5}$$

According to the differential game theory, for the two control strategies $\mathbf{u}$ and $\mathbf{d}$, one tries to minimize its own cost function, and the other tries to maximize its own cost function.

It is known that we want to minimize the decision player $\mathbf{u}_k$ to balance the maximum decision player $\mathbf{d}_k$, that is, to find a feedback saddle point solution $\mathbf{u}^*(k) = \mathbf{u}(\mathbf{x}_k)$ and $\mathbf{d}^*(k) = \mathbf{d}(\mathbf{x}_k)$, so that

$$V(\mathbf{u}^*, \mathbf{d}^*) = \min_u \max_d V(\mathbf{u}, \mathbf{d}). \tag{6}$$

The necessary and sufficient condition for a two-person zero-sum differential game to have a unique solution is that the following Nash equilibrium condition holds

$$\min_u \max_d V(\mathbf{u}, \mathbf{d}) = \max_d \min_u V(\mathbf{u}, \mathbf{d}). \tag{7}$$

According to the Bellman optimality principle [2], the optimal value function $V^*(\mathbf{x}_k)$ satisfies the following HJI equation

$$V^*(\mathbf{x}_k) = \min_u \max_d \{U(\mathbf{x}_k, \mathbf{u}_k, \mathbf{d}_k)\}. \tag{8}$$

The optimal control strategy $\mathbf{u}^*(k)$ and the worst disturbance $\mathbf{d}^*(k)$ satisfy the static equilibrium conditions of the optimal control theory

$$\partial H(\mathbf{x}_k, \mathbf{u}_k, \mathbf{d}_k)/\partial \mathbf{u}_k = 0 \tag{9}$$

and

$$\partial H(\mathbf{x}_k, \mathbf{u}_k, \mathbf{d}_k)/\partial \mathbf{d}_k = 0. \tag{10}$$

Then, the optimal control strategy $\mathbf{u}^*(k)$ can be obtained according to the equation (9)

$$\mathbf{u}^*(\mathbf{x}_k) = -\frac{1}{2}\mathbf{R}^{-1}\mathbf{g}^T(\mathbf{x}_k)\frac{\partial V^*(\mathbf{x}_k)}{\partial \mathbf{x}_k}. \tag{11}$$

According to the equation (10), the worst disturbance $\mathbf{d}^*(k)$ can be obtained

$$\mathbf{d}^*(\mathbf{x}_k) = \frac{1}{2\gamma^2}\mathbf{S}^{-1}\mathbf{h}^T(\mathbf{x}_k)\frac{\partial V^*(\mathbf{x}_k)}{\partial \mathbf{x}_k}. \tag{12}$$

Then the optimal feedback control strategyies $\mathbf{u}^*(\mathbf{x}_k)$ and $\mathbf{d}^*(k)$ are Nash equilibrium strategies. Substituting the equation (11) and the equation (12) into the equation (8), there is the following discrete-time HJI equation

$$\begin{aligned} &\frac{\partial V^{*T}(\mathbf{x}_k)}{\partial \mathbf{x}_k} f(\mathbf{x}_k) + \frac{1}{4}\frac{\partial V^{*T}(\mathbf{x}_k)}{\partial \mathbf{x}_{k+1}}\mathbf{g}(\mathbf{x}_k)\mathbf{R}^{-1}\mathbf{g}^T(\mathbf{x}_k)\frac{\partial V^*(\mathbf{x}_k)}{\partial \mathbf{x}_k} \\ &-\frac{1}{4\gamma^2}\frac{\partial V^{*T}(\mathbf{x}_k)}{\partial \mathbf{x}_k}\mathbf{h}(\mathbf{x}_k)\mathbf{S}^{-1}\mathbf{h}^T(\mathbf{x}_k)\frac{\partial V^*(\mathbf{x}_k)}{\partial \mathbf{x}_k} + \mathbf{x}_k^T\mathbf{Q}\mathbf{x}_k = 0. \end{aligned} \tag{13}$$

In the nonlinear systems, the equation (13) is a nonlinear partial differential equation and cannot be directly solved analytically. An ADP learning strategy based on the neural network is proposed, which solves the equation (13) by continuously approximating the value function$V^*(\mathbf{x}_k)$.

3. ADP DESIGN BASED ON ZERO-SUM GAME

In order to obtain the approximate solution of the HJI equation, it is necessary to use the neural network to approximate the value function$V(\mathbf{x}_k)$, the control strategy $\mathbf{u}(\mathbf{x}_k)$ and the disturbance strategy $\mathbf{d}(\mathbf{x}_k)$ [24]. That is, evaluation network, execution network and disturbance network.

A block schematic of the whole algorithm scheme is given in Figure 1. The evaluation network approximates the value function, the execution network approximates the control strategy, and the disturbance network approximates the disturbance strategy. The weight of evaluation network 2 is iterated in the outer loop, and the weight of evaluation network 1 is iterated in the inner loop. In the inner loop iteration process, the weight of the evaluation network 2 remains unchanged. Once the inner loop iteration is completed, the weight of the evaluation network 1 is sent to the evaluation network 2 as its latest weight. The output of the evaluation network 1 is an estimate of the value function.

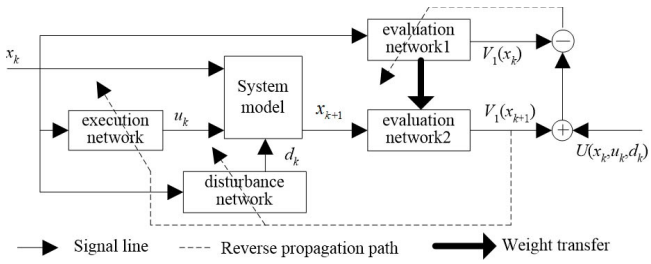

Fig. 1. A block schematic of the algorithm scheme.

3.1. Evaluation network design

The evaluation network can use the neural network, which has strong generalization ability and nonlinear function approximation ability, which is very suitable for approximating performance index functions. The evaluation network trains the network through the gradient descent method, so that it can output the value function.

According to the approximation properties of the neural network, the value function $V_1(\mathbf{x}_k)$ can be approximated as

$$V_1(\mathbf{x}_k) = \mathbf{W}^T \mathbf{j}(\mathbf{x}_k) + \varepsilon(k) \tag{14}$$

where $\mathbf{W}$ is the weight of the evaluation network, $\mathbf{j}(\cdot)$ is the activation function of the evaluation network, and $\varepsilon(k)$ is the approximate error of the evaluation network.

Defining $\hat{\mathbf{W}}$ as the estimated value of the $\mathbf{W}$, then the output of the evaluation network is

$$\hat{V_1}(\mathbf{x}_k) = \hat{\mathbf{W}}^T(k)\mathbf{j}(\mathbf{x}_k). \tag{15}$$

For the control strategy $\mathbf{u}_k$ and disturbance $\mathbf{d}_k$, the approximate Hamilton function can be written as

$$H(\mathbf{x}_k, \mathbf{u}_k, \mathbf{d}_k) = U(\mathbf{x}_k, \mathbf{u}_k, \mathbf{d}_k) + \hat{V_1}(\mathbf{x}_{k+1}) - \hat{V_1}(\mathbf{x}_k). \tag{16}$$

In addition, an auxiliary variable $\mathbf{E}(k)$ is defined as

$$\mathbf{E}(k) = \hat{\mathbf{W}}^T \mathbf{J}(\mathbf{x}_k) \tag{17}$$

where $\mathbf{J}(x_k) = [\Delta\mathbf{j}(x_k), \Delta\mathbf{j}(x_{k-1}), \ldots, \Delta\mathbf{j}(x_{k-j})], \Delta\mathbf{j}(x_k) = \mathbf{j}(x_k) - \mathbf{j}(x_{k-1})$.

The mean square error function of the evaluation network is defined as

$$\hat{\mathbf{E}} = \frac{1}{2}\mathbf{E}^T(k)\mathbf{E}(k). \tag{18}$$

The design purpose is to choose $\hat{\mathbf{W}}$ to minimize the mean square error to make $\hat{\mathbf{W}} \to \mathbf{W}$. The weight update of the evaluation network is realized by the gradient descent method, and the weight update law is

$$\hat{\mathbf{W}}(k+1) = \hat{\mathbf{W}}(k) - \alpha\frac{\partial\hat{\mathbf{E}}}{\partial\hat{\mathbf{W}}} \tag{19}$$

where $\alpha > 0$ is the learning rate of the evaluation network. The weight estimation error is $\Delta\mathbf{W}(k) = \hat{\mathbf{W}}(k+1) - \mathbf{W}(k)$.

The general steps for the evaluation network are as follows:

(1) The structure of the evaluation network is initialized. The number of hidden layer nodes, learning rate $\alpha_c > 0$ and initial weight are given. The number of iterations is k=1.

(2) The feedback states $\mathbf{x}_k$,$\mathbf{x}_{k+1}$, the control u_k,$u_{k+1}$for the execution network and the disturbance d_k and d_{k+1}in the disturbance network are obtained. The value V_{k+1} is calculated in the evaluation network.

(3) The function $\mathbf{U}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{d}_k)$ is calculated.

(4) $\mathbf{E}_c(k)$ is established. The evaluation network is trained, and the weights of the evaluation network are updated by using the gradient descent method. The next iteration is waiting to happen.

(5) If the weight error meets the accuracy requirements, stop the calculation and output the value $\mathbf{V}_k$. Otherwise, let k = k +1 and return to step (2).

3.2. Execution network design

According to the approximation properties of the neural network, the control input $\mathbf{u}(\mathbf{x}_k)$ is approximated by the neural network as

$$\mathbf{u}(\mathbf{x}_k) = \mathbf{W}_e^T\mathbf{j}_\mathrm{e}(\mathbf{x}_k) + \varepsilon_e(k) \tag{20}$$

where $\mathbf{W}_e$ is the ideal weight of the execution network, $\mathbf{j}_\mathrm{e}(\cdot)$ is the activation function of the execution network, and $\varepsilon_e(k)$ is the approximate error of the execution network.

Defining $\hat{\mathbf{W}}_e$ as the estimated value of $\mathbf{W}_e$, then the actual output of the execution network is

$$\hat{\mathbf{u}}(\mathbf{x}_k) = \hat{\mathbf{W}}_e^T(k)\mathbf{j}_\mathrm{e}(\mathbf{x}_k). \tag{21}$$

The feedback error of the execution network is defined as the difference between the actual control signal on the system (1) and the minimized ideal control input signal

$$\mathbf{E}_e(k) = \hat{\mathbf{W}}_e^T(k)\mathbf{j}_\mathrm{e}(\mathbf{x}_k) + \frac{1}{2}\mathbf{R}^{-1}\mathbf{g}^T(\mathbf{x}_k)\frac{\partial\hat{\mathbf{V}}(\mathbf{x}_{k+1})}{\partial\mathbf{x}_{k+1}}. \tag{22}$$

The goal of the execution network is to minimize the equation (23)

$$\hat{E}_e = \frac{1}{2}\mathbf{E}_e^T(k)\mathbf{E}_e(k). \tag{23}$$

By using the gradient descent method, the weight update law of the execution network can be obtained

$$\hat{\mathbf{W}}_e(k+1) = \hat{\mathbf{W}}_e(k) - \alpha_e \frac{\partial \hat{\mathbf{E}}_e}{\partial \hat{\mathbf{W}}_e} \tag{24}$$

where $\alpha_e > 0$ is the learning rate of the execution network. The weight estimation error is $\Delta\mathbf{W}_e(k) = \hat{\mathbf{W}}_e(k+1) - \mathbf{W}_e(k)$.

3.3. Disturbance network design

According to the approximation properties of the neural network, the disturbance input $\mathbf{d}(\mathbf{x}_k)$ is approximated by the neural network as

$$\mathbf{d}(\mathbf{x}_k) = \mathbf{W}_d^T \mathbf{j}_{\mathrm{d}}(\mathbf{x}_k) + \varepsilon_d(k). \tag{25}$$

Where $\mathbf{W}_d$ is the ideal weight of the disturbance network, $\mathbf{j}_{\mathrm{d}}(\cdot)$ is the activation function of the disturbance network, and $\varepsilon_d(k)$ is the approximate error of the disturbance network.

Defining $\hat{\mathbf{W}}_d$ as the estimated value of $\mathbf{W}_d$, then the actual output of the disturbance network is

$$\hat{\mathbf{d}}(\mathbf{x}_k) = \hat{\mathbf{W}}_d^T(k)\mathbf{j}_{\mathrm{d}}(\mathbf{x}_k). \tag{26}$$

The feedback error of the disturbance network is defined as the difference between the actual signal on the system (1) and the minimized ideal input signal

$$\mathbf{E}_d(k) = \hat{\mathbf{W}}_d^T(k)\mathbf{j}_{\mathrm{d}}(\mathbf{x}_k) - \frac{1}{2\gamma^2}\mathbf{S}^{-1}\mathbf{h}^{\mathrm{T}}(\mathbf{x}_k)\frac{\partial \hat{V}(\mathbf{x}_{k+1})}{\partial \mathbf{x}_{k+1}}. \tag{27}$$

The goal of the disturbance network is to minimize equation (28)

$$\hat{\mathbf{E}}_d = \frac{1}{2}\mathbf{E}_d^T(k)\mathbf{E}_d(k). \tag{28}$$

By using the gradient descent method, the weight update law of the disturbance network can be obtained

$$\hat{\mathbf{W}}_d(k+1) = \hat{\mathbf{W}}_d(k) - \alpha_d \frac{\partial \hat{\mathbf{E}}_d}{\partial \hat{\mathbf{W}}_d} \tag{29}$$

where $\alpha_d > 0$ is the learning rate of the disturbance network.The weight estimation error is $\Delta\mathbf{W}_d(k) = \hat{\mathbf{W}}_d(k+1) - \mathbf{W}_d(k)$.

4. CONVERGENCE ANALYSIS

Next, the system stability analysis will be given based on the neural network approximation errors $\varepsilon(k), \varepsilon_e(k)$ and $\varepsilon_d(k)$. Before conclusions, the following assumptions are defined.

Assumption 2. The ideal weights of the evaluation network, the execution network and the disturbance network are all bounded, $||\mathbf{W}|| \leq \mathbf{W}_M, ||\mathbf{W}_e|| \leq \mathbf{W}_{eM}$, and $||\mathbf{W}_d|| \leq \mathbf{W}_{dM}$. $\mathbf{W}_M, \mathbf{W}_{eM}$ and $\mathbf{W}_{dM}$ are unknown normal numbers.

Assumption 3. The approximate errors of the evaluation network, the execution network and the disturbance network are all bounded, $||\varepsilon_(k)|| \leq \varepsilon_M, ||\varepsilon_e(k)|| \leq \varepsilon_{eM}$, and $||\varepsilon_d(k)|| \leq \varepsilon_{dM}$, $\varepsilon_M, \varepsilon_{eM}$ and ε_{dM} are the positive constants.

Assumption 4. The activation functions of the evaluation network, the execution network and the disturbance network are all bounded, $||\mathbf{j}(\cdot)|| \leq \mathrm{j}_M, ||\mathbf{j}_e(\cdot)|| \leq \mathrm{j}_{eM}$, and $||\mathbf{j}_d(\cdot)|| \leq \mathrm{j}_{dM}$, $\mathrm{j}_M, \mathrm{j}_{eM}$ and j_{dM} are the positive constants.

Assumption 5. $\mathbf{g}(\mathbf{x}_k)$,$\mathbf{f}(\mathbf{x}_k)$ and $\mathbf{h}(\mathbf{x}_k)$ are the local Lipschitz continuous functions. And there are the normal number g_M, h_M such that

$$\|g(x_k)\| \leq g_M, \quad \|h(x_k)\| \leq h_M. \tag{30}$$

Theorem 4.1. Considering the system (1), it is assumed that the weight update laws of the evaluation network, the execution network, and the disturbance network are given by the equations (19), (24) and (29), and the initial weight value of the execution network is selected so that the system is initially stable. If the evaluation network, execution network and the disturbance network are updated at the same time along the trajectory of the system (1), then the weight estimations $\hat{\mathbf{W}}(k), \hat{\mathbf{W}}_e(k)$ and $\hat{\mathbf{W}}_d(k)$ are convergent.

P r o o f. Considering the following Lyapunov function

$$V(\mathbf{x}_k) = V_2(\mathbf{x}_k) + V_3(\Delta\mathbf{W}(k)) + V_e(\Delta\mathbf{W}_e(k)) + V_d(\Delta\mathbf{W}_d(k)) \tag{31}$$

where

$$V_2(\mathbf{x}_k) = \mathbf{x}_k^T\mathbf{x}_k \tag{32}$$

$$V_3(\boldsymbol{\Delta}\mathbf{W}(k)) = tr\{\boldsymbol{\Delta}\mathbf{W}^T(k)\,\boldsymbol{\Delta}\mathbf{W}(k)\} \tag{33}$$

$$V_e(\boldsymbol{\Delta}\mathbf{W}_e(k)) = tr\{\boldsymbol{\Delta}\mathbf{W}_e^T(k)\,\boldsymbol{\Delta}\mathbf{W}_e(k)\} \tag{34}$$

$$V_d(\boldsymbol{\Delta}\mathbf{W}_d(k)) = tr\{\boldsymbol{\Delta}\mathbf{W}_d^T(k)\,\boldsymbol{\Delta}\mathbf{W}_d(k)\}. \tag{35}$$

By using the equations (21) and (26), the closed-loop system can be written as

$$\begin{aligned}\mathbf{x}_{k+1} &= \mathbf{f}(\mathbf{x}_k) + \mathbf{g}(\mathbf{x}_k)\hat{\mathbf{u}}(\mathbf{x}_k) + \mathbf{h}(\mathbf{x}_k)\hat{\mathbf{d}}(\mathbf{x}_k)\\ &= \mathbf{f}(\mathbf{x}_k) + \mathbf{g}(\mathbf{x}_k)\mathbf{u}^*(\mathbf{x}_k) + \mathbf{h}(\mathbf{x}_k)\mathbf{d}^*(\mathbf{x}_k) + \mathbf{g}(\mathbf{x}_k)\hat{\mathbf{W}}_e^T(k)\mathbf{j}_\mathbf{e}(\mathbf{x_k})\\ &\quad -\mathbf{g}(\mathbf{x}_k)\varepsilon_e(k) + \mathbf{h}(\mathbf{x}_k)\hat{\mathbf{W}}_d^T(k)\mathbf{j}_\mathbf{d}(\mathbf{x_k}) - \mathbf{h}(\mathbf{x}_k)\varepsilon_d(k)\end{aligned} \tag{36}$$

then the difference of $V_2(\mathbf{x}_k)$ is

$$\begin{aligned}&\Delta V_2(x_k) = \mathbf{x}_{k+1}^T\mathbf{x}_{k+1} - \mathbf{x}_k^T\mathbf{x}_k\\ &= ||\mathbf{f}(\mathbf{x}_k) + \mathbf{g}(\mathbf{x}_k)\mathbf{u}^*(\mathbf{x}_k) + \mathbf{h}(\mathbf{x}_k)\mathbf{d}^*(\mathbf{x}_k) + \mathbf{g}(\mathbf{x}_k)\hat{\mathbf{W}}_e^T(k)\mathbf{j}_\mathrm{e}(\mathbf{x}_k)\\ &+\mathbf{h}(\mathbf{x}_k)\hat{\mathbf{W}}_d^T(k)\mathbf{j}_\mathrm{d}(\mathbf{x}_k) - \mathbf{g}(\mathbf{x}_k)\varepsilon_e(k) - \mathbf{h}(\mathbf{x}_k)\varepsilon_d(k)||^2 - \mathbf{x}_k^T\mathbf{x}_k\\ &\leq 2||\mathbf{f}(\mathbf{x}_k) + \mathbf{g}(\mathbf{x}_k)\mathbf{u}^*(\mathbf{x}_k) + \mathbf{h}(\mathbf{x}_k)\mathbf{d}^*(\mathbf{x}_k)||^2\\ &+8||\mathbf{g}(\mathbf{x}_k)\hat{\mathbf{W}}_e^T(k)\mathbf{j}_\mathrm{e}(\mathbf{x}_k)||^2 + 8||\mathbf{h}(\mathbf{x}_k)\hat{\mathbf{W}}_d^T(k)\mathbf{j}_\mathrm{d}||^2 + 8||\mathbf{g}(\mathbf{x}_k)\varepsilon_e(k)||^2\\ &+8||\mathbf{h}(\mathbf{x}_k)\varepsilon_d(k)||^2 - \mathbf{x}_k^T\mathbf{x}_k.\end{aligned} \tag{37}$$

Suppose

$$||\mathbf{f}(\mathbf{x}_k) + \mathbf{g}(\mathbf{x}_k)\mathbf{u}^*(\mathbf{x}_k) + \mathbf{h}(\mathbf{x}_k)\mathbf{d}^*(\mathbf{x}_k)||^2 \leq K_1||\mathbf{x}_k||^2 \tag{38}$$

and let

$$\begin{cases} \mathbf{\Pi}_e = \left(\hat{\mathbf{W}}_e^T(k) - \mathbf{W}_e^T(k)\right)\mathbf{j}_{\mathrm{e}}(\mathbf{x}_k) \\ \mathbf{\Pi}_d = \left(\hat{\mathbf{W}}_d^T(k) - \mathbf{W}_d^T(k)\right)\mathbf{j}_{\mathrm{d}}(\mathbf{x}_k) \\ \mathbf{\Pi} = \left(\hat{\mathbf{W}}^T(k) - \mathbf{W}^T(k)\right)\mathbf{j}(\mathbf{x}_k). \end{cases} \tag{39}$$

By Assumption 5, then

$$\begin{aligned} \Delta V_2(\mathbf{x}_k) &\leq 2K_1||\mathbf{x}_k||^2 + 8g_M^2||\mathbf{\Pi}||^2 + 8h_M^2||\mathbf{\Pi}_d||^2 + 8g_M^2\varepsilon_M^2 + 8h_M^2\varepsilon_{dM}^2 - ||\mathbf{x}_k||^2 \\ &= -(1-2K_1)||\mathbf{x}_k||^2 + 8g_M^2||\mathbf{\Pi}||^2 + 8h_M^2||\mathbf{\Pi}_d||^2 + 8g_M^2\varepsilon_M^2 + 8h_M^2\varepsilon_{dM}^2. \end{aligned} \tag{40}$$

The difference of $\Delta V_e\left(\mathbf{\Delta W}_e(k)\right)$ is

$$\begin{aligned} &\Delta V_e\left(\hat{\mathbf{W}}_e(k)\right) = tr\left\{\hat{\mathbf{W}}_e^T(k+1)\hat{\mathbf{W}}_e(k+1)\right\} - tr\left\{\hat{\mathbf{W}}_e^T(k)\hat{\mathbf{W}}_e(k)\right\} \\ &= tr\left\{\left(\hat{\mathbf{W}}_e^T(k) - \alpha_e\left(\frac{\partial\hat{E}_e(k)}{\partial\hat{\mathbf{W}}_e(k)}\right)^T\right)\left(\hat{\mathbf{W}}_e(k) - \alpha_e\frac{\partial\hat{E}_e(k)}{\partial\hat{\mathbf{W}}_e(k)}\right)\right\} \\ &-tr\left\{\hat{\mathbf{W}}_e^T(k)\hat{\mathbf{W}}_e(k)\right\} \\ &= tr\left\{\hat{\mathbf{W}}_e^T(k)\hat{\mathbf{W}}_e(k) - \alpha_e\hat{\mathbf{W}}_e(k)\left(\frac{\partial\hat{E}_e(k)}{\partial\hat{\mathbf{W}}_e(k)}\right)^T - \alpha_e\hat{\mathbf{W}}_e^T(k)\frac{\partial\hat{E}_e(k)}{\partial\hat{\mathbf{W}}_e(k)}\right. \\ &\left.+\alpha_e^2\left(\frac{\partial\hat{E}_e(k)}{\partial\hat{\mathbf{W}}_e(k)}\right)^T\frac{\partial\hat{E}_e(k)}{\partial\hat{\mathbf{W}}_e(k)}\right\} - tr\left\{\hat{\mathbf{W}}_e^T(k)\hat{\mathbf{W}}_e(k)\right\}. \end{aligned} \tag{41}$$

Substituting the equations (23) and (24) into the equation (41), we get

$$\begin{aligned} &\Delta V_e\left(\mathbf{\Delta W}_e(k)\right) = tr\left\{\alpha_e^2\left(\mathbf{j}_{\mathrm{e}}(\mathbf{x}_k) + \frac{1}{2}\mathbf{R}^{-1}\mathbf{g}^T(\mathbf{x}_k)\left(\frac{\partial\hat{\mathbf{V}}(\mathbf{x}_k)}{\partial\mathbf{x}_k}\right)^T\hat{\mathbf{W}}_e(k)\right)^T\right. \\ &\times\left(\mathbf{j}_{\mathrm{e}}(\mathbf{x}_k) + \frac{1}{2}\mathbf{R}^{-1}\mathbf{g}^T(\mathbf{x}_k)\left(\frac{\partial\hat{\mathbf{V}}(\mathbf{x}_k)}{\partial\mathbf{x}_k}\right)^T\hat{\mathbf{W}}_e(k)\right) \\ &\left.-\alpha_e\hat{\mathbf{W}}_e(k)\left(\mathbf{j}_{\mathrm{e}}(\mathbf{x}_k) + \frac{1}{2}\mathbf{R}^{-1}\mathbf{g}^T(\mathbf{x}_k)\left(\frac{\partial\hat{\mathbf{V}}(\mathbf{x}_k)}{\partial\mathbf{x}_k}\right)^T\right)\right\} \end{aligned}$$

$$\leq \alpha_e^2 j_{eM}^2 + \frac{1}{4}\alpha_e^2 g_M^2\mathbf{R}^{-1}\left\|\mathbf{\Pi}_{\mathbf{e}}\right\|^2\mathbf{R}^{-1}g_M^2 + \alpha_e^2 g_M^2\mathbf{R}^{-1}\left\|\mathbf{\Pi}_{\mathbf{e}}\right\|^2 + \frac{1}{4}\varepsilon_{eM}^2. \tag{42}$$

The difference of $V_3\left(\boldsymbol{\Delta}\mathbf{W}\left(k\right)\right)$ is

$$\begin{aligned}
&\Delta V_3\left(\boldsymbol{\Delta}\mathbf{W}\left(k\right)\right) = tr\left\{\hat{\mathbf{W}}^T(k+1)\hat{\mathbf{W}}(k+1)\right\} - tr\left\{\hat{\mathbf{W}}^T(k)\hat{\mathbf{W}}(k)\right\}\\
&= tr\left\{\left(\hat{\mathbf{W}}^T(k) - \alpha\left(\frac{\partial\hat{E}(k)}{\partial\hat{\mathbf{W}}(k)}\right)^T\right)\left(\hat{\mathbf{W}}(k) - \alpha\frac{\partial\hat{E}(k)}{\partial\hat{\mathbf{W}}(k)}\right)\right\}\\
&-tr\left\{\hat{\mathbf{W}}^T(k)\hat{\mathbf{W}}(k)\right\}\\
&= tr\left\{\hat{\mathbf{W}}^T(k)\hat{\mathbf{W}}(k) - \alpha\hat{\mathbf{W}}(k)\left(\frac{\partial\hat{E}_v(k)}{\partial\hat{\mathbf{W}}(k)}\right)^T - \alpha\hat{\mathbf{W}}^T(k)\frac{\partial\hat{E}(k)}{\partial\hat{\mathbf{W}}(k)}\right.\\
&\left.+\alpha^2\left(\frac{\partial\hat{E}(k)}{\partial\hat{\mathbf{W}}(k)}\right)^T\frac{\partial\hat{E}(k)}{\partial\hat{\mathbf{W}}(k)}\right\} - tr\left\{\hat{\mathbf{W}}^T(k)\hat{\mathbf{W}}(k)\right\}.
\end{aligned} \tag{43}$$

Substituting the equations (18) and (19) into the equation (43), the process is similar to equation (42), so it is omitted here. we can get

$$\Delta V_3\left(\boldsymbol{\Delta}\mathbf{W}\left(k\right)\right) \leq -\alpha^2 j_M^2 - \varepsilon_M^2 + \frac{1}{4}g_M^2\left\|\Pi\right\|^2. \tag{44}$$

The difference of $V_d(\boldsymbol{\Delta}\mathbf{W}_d(k))$ is

$$\begin{aligned}
&\Delta V_d\left(\boldsymbol{\Delta}\mathbf{W}_d\left(k\right)\right) = tr\left\{\hat{\mathbf{W}}_d^T(k+1)\hat{\mathbf{W}}_d(k+1)\right\} - tr\left\{\hat{\mathbf{W}}_d^T(k)\hat{\mathbf{W}}_d(k)\right\}\\
&= tr\left\{\left(\hat{\mathbf{W}}_d^T(k) - \alpha_d\left(\frac{\partial\hat{E}_d(k)}{\partial\hat{\mathbf{W}}_d(k)}\right)^T\right)\left(\hat{\mathbf{W}}_d(k) - \alpha_d\frac{\partial\hat{E}_d(k)}{\partial\hat{\mathbf{W}}_d(k)}\right)\right\}\\
&-tr\left\{\hat{\mathbf{W}}_d^T(k)\hat{\mathbf{W}}_d(k)\right\}\\
&= tr\left\{\hat{\mathbf{W}}_d^T(k)\hat{\mathbf{W}}_d(k) - \alpha_d\hat{\mathbf{W}}_d(k)\left(\frac{\partial\hat{E}_d(k)}{\partial\hat{\mathbf{W}}_d(k)}\right)^T - \alpha_d\hat{\mathbf{W}}_d^T(k)\frac{\partial\hat{E}_d(k)}{\partial\hat{\mathbf{W}}_d(k)}\right.\\
&\left.+\alpha_d^2\left(\frac{\partial\hat{E}_d(k)}{\partial\hat{\mathbf{W}}_d(k)}\right)^T\frac{\partial\hat{E}_d(k)}{\partial\hat{\mathbf{W}}_d(k)}\right\} - tr\left\{\hat{\mathbf{W}}_d^T(k)\hat{\mathbf{W}}_d(k)\right\}.
\end{aligned} \tag{45}$$

Substituting the equations (28) and (29) into the equation (45), the derivation process is similar to the equation (42), so it is omitted here. We get

$$\begin{aligned}
\Delta V_d\left(\boldsymbol{\Delta}\mathbf{W}_d\left(k\right)\right) \leq{}& \alpha_d^2 j_{dM}^2 - \frac{1}{\gamma^2}\alpha_d^2 h_M^2\mathbf{S}^{-1}\left\|\Pi_d\right\|^2\mathbf{S}^{-1}\\
&-\frac{1}{4}\varepsilon_{dM}^2 + \frac{1}{4\gamma^2}\alpha_d^2 h_M^4\mathbf{S}^{-1}\left\|\Pi_d\right\|^2.
\end{aligned} \tag{46}$$

Putting the equations (40), (42), (44) and (46) into the equation (31), the difference $\Delta V(\mathbf{x}_k)$ is

$$\begin{aligned}
&\Delta V(\mathbf{x}_k) \leq -\left(1-2K_1\right)\left\|\mathbf{x}_k\right\|^2 + 8g_M^2\left\|\boldsymbol{\Pi}\right\|^2 + 8h_M^2\left\|\boldsymbol{\Pi}_d\right\|^2\\
&+\frac{1}{4}\alpha_e^2 g_M^2\mathbf{R}^{-1}\left\|\Pi_e\right\|^2\mathbf{R}^{-1}g_M^2 + G_M + \alpha_e^2 g_M^2\mathbf{R}^{-1}\left\|\Pi_e\right\|^2\\
&-\frac{1}{\gamma^2}\alpha_d^2 h_M^2\mathbf{S}^{-1}\left\|\Pi_d\right\|^2\mathbf{S}^{-1} + \frac{1}{4\gamma^2}\alpha_d^2 h_M^4\mathbf{S}^{-1}\left\|\Pi_d\right\|^2 - E_M
\end{aligned} \tag{47}$$

where

$$G_M = 8g_M^2\varepsilon_M^2 + 8h_M^2\varepsilon_{dM}^2 + \frac{1}{4}\varepsilon_{eM}^2 + \alpha_d^2 j_{dM}^2 + \alpha_e^2 j_{eM}^2 \tag{48}$$

$$E_M = \frac{1}{4}\varepsilon_{dM}^2 + \alpha^2 j_M^2 + \varepsilon_M^2. \tag{49}$$

If $0 < K_1 < 1/2$, $E_M > G_M$ and the following inequality (50) and (51) holds

$$||\Pi_e|| > \sqrt{8g_M^2 + \frac{1}{4}\alpha_e^2\mathbf{R}^{-1}g_M^4\mathbf{R}^{-1} + \alpha_e^2\mathbf{R}^{-1}g_M^2} \tag{50}$$

$$||\Pi_d|| > \sqrt{8h_M^2 + \frac{1}{4\gamma^2}\mathbf{S}^{-1}\alpha_d^2 h_M^4 + \frac{1}{\gamma^2}\mathbf{S}^{-1}\alpha_d^2 h_M^2\mathbf{S}^{-1}} \tag{51}$$

then, $\Delta V < 0$ can be obtained. Therefore, according to the Lyapunov stability theory, it can be known that the weight estimation $\hat{\mathbf{W}}(k), \hat{\mathbf{W}}_e(k)$ and $\hat{\mathbf{W}}_d(k)$ are convergent. The theorem proof is completed. □

Remark 2. According to the criterion of convergence theorem, all functions in the uniformly convergence function family are bounded, and the total function family is convergence. If it is proved that the system is uniformly convergent, then it is necessary to set the sub-functions to be convergent in advance. According to Theorem 4.1, it is proved that the system is uniform convergent.

Theorem 4.2. If the control input $\hat{\mathbf{u}}(\mathbf{x}_k)$ and the disturbance input $\hat{\mathbf{d}}(\mathbf{x}_k)$ are obtained according to Theorem 4.1, then the $\hat{\mathbf{u}}(\mathbf{x}_k)$ and the $\hat{\mathbf{d}}(\mathbf{x}_k)$ will converge to the Nash equilibrium solution of the zero-sum differential game, that is, the control input $\hat{\mathbf{u}}(\mathbf{x}_k)$ is in the neighborhood of the optimal control input $\mathbf{u}^*(\mathbf{x}_k)$, $||\hat{\mathbf{u}}(\mathbf{x}_k) - \mathbf{u}^*(\mathbf{x}_k)|| \leq \varepsilon_u$. The disturbance input $\hat{\mathbf{d}}(\mathbf{x}_k)$ is in the neighborhood of the optimal disturbance input $\mathbf{d}^*(\mathbf{x}_k)$, $||\hat{\mathbf{d}}(\mathbf{x}_k) - \mathbf{d}^*(\mathbf{x}_k)|| \leq \varepsilon_d$. Where ε_u and ε_d are small normal number.

P r o o f . According to the equation (15), after derivation, substituting into the equation (11), the optimal estimation $\hat{\mathbf{u}}(\mathbf{x}_k)$ can be obtained as

$$\hat{\mathbf{u}}(\mathbf{x}_k) = -\frac{1}{2}\mathbf{R}^{-1}\mathbf{g}^T(\mathbf{x}_k)\left(\frac{\partial j(\mathbf{x}_k)}{\partial \mathbf{x}_k}\right)^T \mathbf{W}(\mathbf{x}_k). \tag{52}$$

According to the equations (11) and (52), and the boundedness of $\left\|\dfrac{\partial j(\mathbf{x}_k)}{\partial \mathbf{x}_k}\right\|$ and $\|\Delta W(\mathbf{x}_k)\|$, we can get

$$\begin{aligned}\hat{\mathbf{u}}(\mathbf{x}_k) - \mathbf{u}^*(\mathbf{x}_k)|| &\leq \left\|-\frac{1}{2}\mathbf{R}^{-1}\mathbf{g}^T(\mathbf{x}_k)\left(\frac{\partial j(\mathbf{x}_k)}{\partial \mathbf{x}_k}\right)^T \Delta\mathbf{W}(\mathbf{x}_k)\right\| \\ &\leq \lambda_{\max}\left(\mathbf{R}^{-1}g_M j_M \varepsilon_M\right) = \varepsilon_u.\end{aligned} \tag{53}$$

Similarly, the optimal estimation $\hat{\mathbf{d}}(\mathbf{x}_k)$ can be obtained

$$\hat{\mathbf{d}}(\mathbf{x}_k) = \frac{1}{2\gamma^2}\mathbf{S}^{-1}\mathbf{h}^T(\mathbf{x}_k)\left(\frac{\partial j(\mathbf{x}_k)}{\partial \mathbf{x}_k}\right)^T \mathbf{W}(\mathbf{x}_k) \tag{54}$$

we can get

$$\begin{aligned}\hat{\mathbf{d}}(\mathbf{x}_k) - \mathbf{d}^*(\mathbf{x}_k)|| &\leq \left\| \frac{1}{2\gamma^2}\mathbf{S}^{-1}\mathbf{h}^T(\mathbf{x}_k)\left(\frac{\partial j(\mathbf{x}_k)}{\partial \mathbf{x}_k}\right)^T \Delta\mathbf{W}(\mathbf{x}_k)\right\| \\ &\leq \lambda_{\max}\left(\frac{1}{2\gamma^2}S^{-1}h_M j_M \varepsilon_M\right) = \varepsilon_d.\end{aligned} \tag{55}$$

Therefore, $\hat{\mathbf{u}}(\mathbf{x}_k)$ and $\hat{\mathbf{d}}(\mathbf{x}_k)$ converge to the Nash equilibrium point of the zero-sum game. The proof is complete. □

5. SIMULATION STUDY

Considering the following discrete-time nonlinear system

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k) + \mathbf{g}(\mathbf{x}_k)\mathbf{u}_k + \mathbf{h}(\mathbf{x}_k)\mathbf{d}_k$$

where

$$\mathbf{f}(\mathbf{x}_k) = \begin{bmatrix} \cos(0.2\mathbf{x}_{1k} - 0.6\mathbf{x}_{2k}) \\ \sin(0.5\mathbf{x}_{1k} - \mathbf{x}_{2k}) + 1.6\mathbf{x}_{2k} \end{bmatrix},$$
$$\mathbf{g}(\mathbf{x}_k) = \begin{bmatrix} 0 \\ -\mathbf{x}_{2k} \end{bmatrix}, \mathbf{h}(\mathbf{x}_k) = \begin{bmatrix} 0 \\ 0.2 \end{bmatrix}.$$

The cost function is defined by the equation (2), where $\mathbf{Q}$ and $\mathbf{R}$ are the matrices of appropriate dimensions,let $Q = \mathrm{diag}\{5, 10\}, R = 2I, S = I$, I represents a unit matrix with appropriate dimensions. Let $\gamma = 5$.

Next, two neural network structures are selected for simulation to verify the effectiveness of the proposed method.

In two neural network structures, the algorithm is executed to complete 200 iterations, and each iteration includes 500 trainings for the three networks to achieve the given calculation accuracy $e = 10^{-6}$. The neural network parameters are initialized. Assume that the weights from the input layer to the hidden layer are 1. The initial values of the weights from the hidden layer to the output layer are randomly selected from [-0.5, 0.5]. All the activation functions are selected as sigmoid(·).

(1) The neural networks with the structures 2-8-1,2-6-1 and 2-6-1 are established to fit the evaluation network, execution network and disturbance network respectively. The learning rates of the three networks are selected as $\alpha_c = 0.15$, $\alpha_a = 0.25$ and $\alpha_d = 0.15$. Let disturbance $d_k = 0.02\sin(0.2k)$.

In this NN structure,the convergence curve of the value function is shown in Figure 2. The convergence curve of the weights for the evaluation network, execution network and disturbance network is shown in Figure 3 to Figure 5. The trajectory curves of the system states are shown in Figure 6. The optimal control curve is shown in Figure 7.

(2) The neural networks with the structures 2-4-1,2-3-1 and 2-3-1 are established to fit the evaluation network, execution network and disturbance network respectively. The learning rates of the three networks are selected as $\alpha_c = 0.1$, $\alpha_a = 0.15$ and $\alpha_d = 0.25$. Let disturbance $d_k = 0.02\cos(0.3k)$.

In this structure, the convergence curve for the value function is shown in Figure 8.The convergence curves of the weights for the evaluation network, execution network and

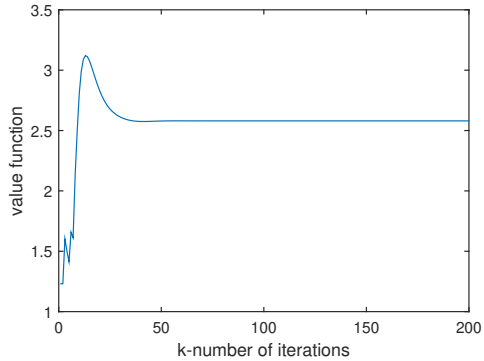

Fig. 2. The value of cost function.

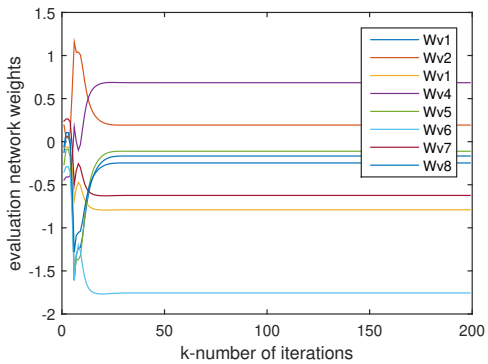

Fig. 3. The weights of the evaluation network.

disturbance network is shown in Figure 9 to Figure 11. The trajectory curves of the system states are shown in Figure 12. The optimal control curve is shown in Figure 13.

In two neural network structures,it can be seen from Figure 2 and Figure 8 that the value function is upper bound, and it is stable after about 40 iterations. The effectiveness of the algorithm is confirmed.

The convergence curves of the weights for the evaluation network are shown in Figure 3 and Figure 9. The convergence curves of the weights for the execution network are shown in Figure 4 and Figure 10. The convergence curves of the weights for the disturbance network are shown in Figure 5 and Figure 11. In two neural network structures, all three networks weights can be stable after training. It shows the boundedness of the NN estimation. The proposed optimization design has been verified to achieve good performance.

The trajectory curves of the system states are shown in Figure 6 and Figure 12. The optimal control curves are shown in Figure 7 and Figure 13. It shows that the proposed neural network optimization design can obtain good control performance, and

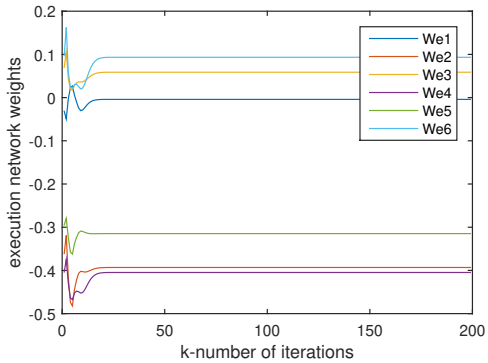

Fig. 4. The weights of the execution network.

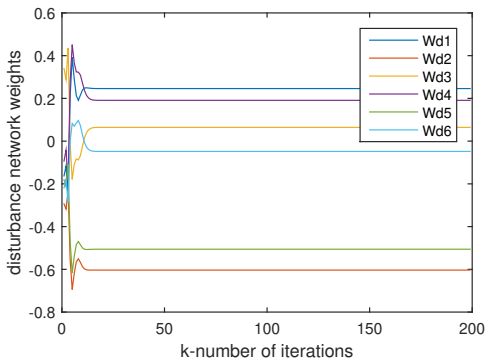

Fig. 5. The weights of the disturbance network.

can ensure that the signal in the closed loop system is bounded, which is consistent with the theoretical analysis. The feasibility and effectiveness of the proposed design are verified.

6. CONCLUSION

In this paper, based on the zero-sum differential game theory, by using the approximate dynamic programming algorithm, the constrained optimization control problem is solved for the nonlinear systems. The neural network is used to fit the evaluation network, the execution network and the disturbance network in order to obtain the approximate solution of the HJI equation. The Lyapunov stability theory is used to prove the uniform convergence, and the system control output converges to the neighborhood of the target reference value. Finally, the simulation example verifies the effectiveness of the algorithm. But there are still some challenges. For example, the guideline of simulation parameters adjustment is not discussed in this paper. In fact, this is indeed an important subject that needs further research.

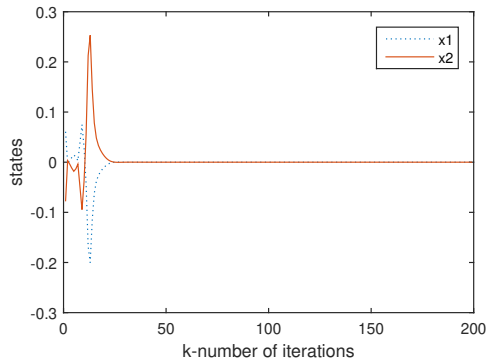

Fig. 6. The system states curves.

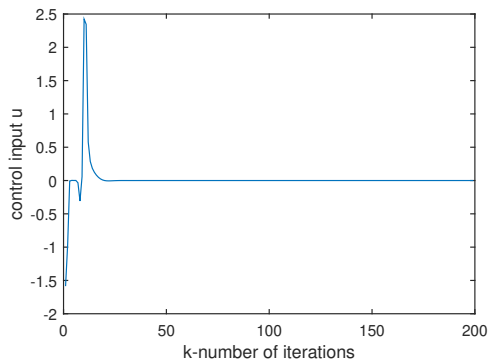

Fig. 7. The curve of optimal control input u.

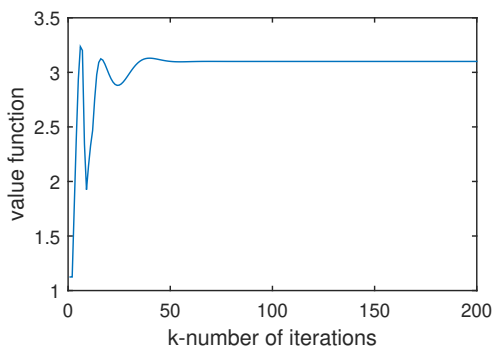

Fig. 8. The value of cost function.

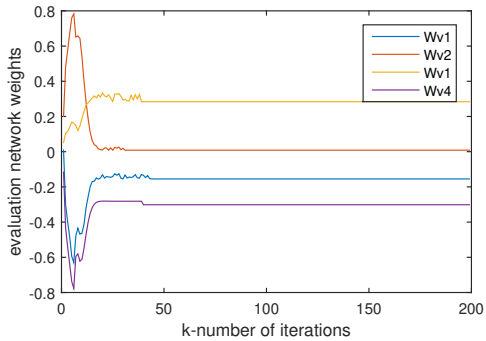

Fig. 9. The weights of the evaluation network.

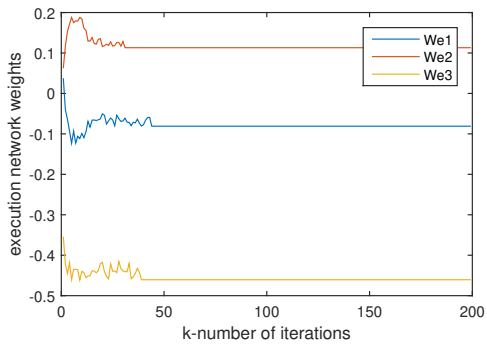

Fig. 10. The weights of the execution network.

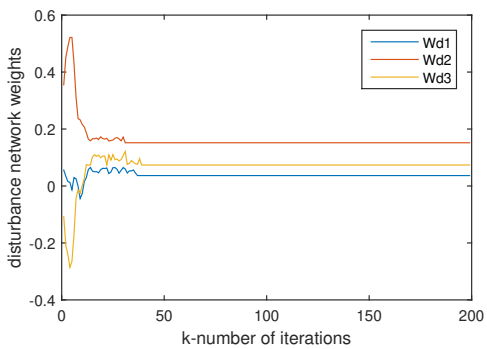

Fig. 11. The weights of the disturbance network.

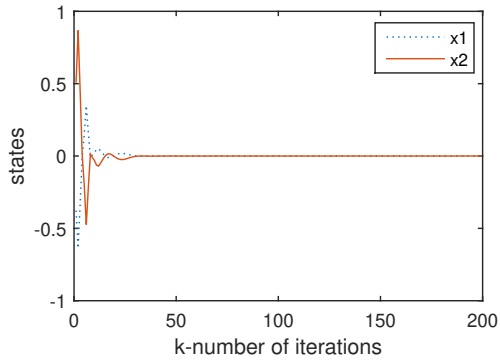

Fig. 12. The system states curves.

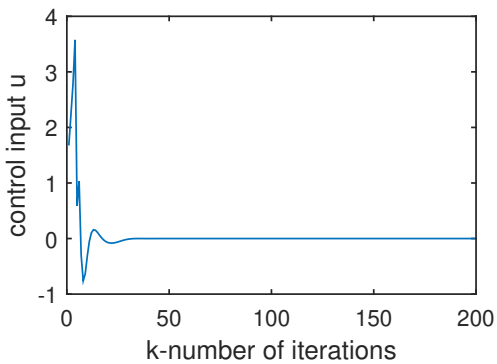

Fig. 13. The curve of optimal control input u.

ACKNOWLEDGEMENT

This work was supported by the National Natural Science Foundation of China under Grant 61573230, and supported by Research Development Project of Beijing Information Science and Technology University under Grant 5221823306.

(Received October 2, 2020)

REFERENCES

[1] A. L'Afflitto: Differential games, continuous Lyapunov functions, and stabilization of non-linear dynamical systems. IET Control Theory Appl. *11* (2017), 2486–2496. DOI:10.1049/iet-cta.2017.0271

[2] R. E. Bellman: Dynamic Programming. Princeton University Press, Princeton 1957.

[3] T. Bian, Y. Jiang, and Z. P. Jiang: Adaptive dynamic programming and optimal control of nonlinear nonaffine systems. Automatica *50* (2014), 2624–2632. DOI:10.1016/j.automatica.2014.08.023

[4] Y. Chai, J.-J. Luo, N. Han, and J.-F. Xie: Attitude takeover control of failed spacecraft using SDRE based differential game approach. J. Astronaut. *41* (2020), 191–198.

[5] F. F. M. El-Sousy and K. A. Abuhasel: Nonlinear robust optimal control via adaptive dynamic programming of permanent-magnet linear synchronous motor drive for uncertain two-axis motion control system. IEEE Trans. Industry Appl. *56* (2020), 1940–1952. DOI:10.1109/ias.2018.8544612

[6] S. Federico and E. Tacconi: Dynamic programming for optimal control problems with delays in the control variable. SIAM J. Control Optim. *52* (2014), 1203–1236. DOI:10.1137/110840649

[7] Y. H. Garcia and J. Gonzalez-Hernandez: Discrete-time Markov control processes with recursive discount rates. Kybernetika *52* (2016), 403–426.

[8] D. Gromov and E. Gromova: On a class of hybrid differential games. Dynamic Games Appl. *7* (2017), 266–288. DOI:10.1007/s13235-016-0185-3

[9] W. Hua, Q. Meng, and J. Zhang: Differential game guidance law for dual and bounded controlled missiles. J. Bejing Univ. Aeronaut. Astronaut. *42* (2016), 1851–1856.

[10] R. Isaacs: Differential Games. John Wiley and Sons, New York 1965.

[11] A. Krasnosielska-Kobos: Construction of Nash equilibrium based on multiple stopping problem in multi-person game. Math. Methods Oper. Res. *83* (2016), 53–70. DOI:10.1007/s00186-015-0519-8

[12] L. Lei, L. Yan-Jun, Ch. Aiqing, T. Shaocheng, and C. L. P. Chen: Integral barrier Lyapunov function based adaptive control for switched nonlinear systems. Science China Inform. Sci. *63* (2020), 132203. DOI:10.1007/s11432-019-2714-7

[13] L. Lei, J. Yan-Jun, L. Dapeng, T. Shaocheng, and W. Zhanshan: Barrier Lyapunov function based adaptive fuzzy FTC for switched systems and its applications to resistance inductance capacitance circuit system. IEEE Trans. Cybernet. *50* (2020), 3491–3502. DOI:10.1109/TCYB.2019.2931770

[14] J.-M. Li and H.-N. Zhu: Nash differential games for delayed nonlinear stochastic systems with state-and control-dependent noise. J. Guangdong Univ. Technol. *35* (2018), 41–45.

[15] D. R. Liu, H. L. Li, and D. Wang: Neural-network-based zero-sum game for discrete time nonlinear systems via iterative adaptive dynamic programming algorithm. Neurocomputing *110* (2013), 92–100. DOI:10.1016/j.neucom.2012.11.021

[16] M. Majid, Seyed Kamal Hosseini Sani: A Novel distributed optimal adaptive control algorithm for nonlinear multi-agent differential graphical games. IEEE/CAA J. Automat. Sinica *5* (2018), 331–341. DOI:10.1109/JAS.2017.7510784

[17] M. Marzieh, B. Karimi, and M. Mahootchi: A differential Stackelberg game for pricing on a freight transportation network with one dominant shipper and multiple oligopolistic carriers. Scientia Iranica *23* (2016), 2391–2406. DOI:10.24200/sci.2016.3964

[18] J. Moon: Necessary and sufficient conditions of risk – sensitive optimal control and differential games for stochastic differential delayed equations. Int. J. Robust Nonlinear Control *29* (2019), 4812–4827. DOI:10.1002/rnc.4655

[19] C. Mu and K. Wang: Approximate-optimal control algorithm for constrained zero-sum differential games through event-triggering mechanism. Nonlinear Dynamics *95* (2019), 2639–2657. DOI:10.1007/s11071-018-4713-0

[20] J. F. Nash: Equilibrium points in n-person games. Proc. Nat. Acad. Sci. USA *36* (1950), 1, 48–49.

[21] J. Nash: Non-cooperative games. Ann. Math. *54* (1951), 286–295.

[22] J. von Neumann and O. Morgenstern: Theory of Games and Economic Behavior. Princeton University Press, Princeton 1944.

[23] H. Pham and X. Wei: Dynamic programming for optimal control of stochastic McKean–Vlasov dynamics. SIAM J. Control Optim. *55* (2016), 1069–1101. DOI:10.1137/16M1071390

[24] Ch. Qin: Research on Optimal Control Based on Approximate Dynamic Programming Application in Power System. Doctoral dissertation, Northeastern University 2014.

[25] C. Rui-Feng, L. Wei-Dong, and E. G. Li: Differential game guidance of underwater nonlinear tracking control based on continuous time generalized predictive correction. Acta Physica Sinica *67* (2018), 050501. DOI:10.7498/aps.67.20171185

[26] R. Song, J. Li, and F. L. Lewis: Robust optimal control for disturbed nonlinear zero-sum differential games based on single NN and least qquares. IEEE Trans. Systems Man Cybernet. Systems *PP99* (2019), 4009–4019. DOI:10.1109/TSMC.2019.2897379

[27] Q. Wei, D. Liu, and H. Lin: Value iteration adaptive dynamic programming for optimal control of discrete-time nonlinear systems. IEEE Trans Cybernet. *46* (2016), 840–853. DOI:10.1109/TCYB.2015.2492242

[28] P. J. Werbos: Advances forecasting methods for global crisis warning and models of intelligence. General Systems Yearbook *22* (1977), 25–38.

[29] B. Yang, C. L. Xuesong: Heuristic dynamic programming based optimal control for multiple time delay systems. J. Theoret. Appl. Inform. Technol. *48* (2013), 876–881.

[30] H. Zhang, Ch. Qin, B. Jiang, and Y. Luo: Online adaptive policy learning algorithm for Hinf state feedback control of unknown affine nonlinear discrete-time systems. IEEE Trans. Cybernet. *44* (2014), 2706–2718. DOI:10.1109/TCYB.2014.2313915

[31] F. Zhang, G. S. Shan, and H. Gao: Rheumatoid arthritis analysis by Nash equilibrium game analysis. J. Medical Imaging Health Inform. *9* (2019), 1382–1385. DOI:10.1166/jmihi.2019.2760

[32] Q. Zhu, Y. Liu, and G. Wen: Adaptive neural network output feedback control for stochastic nonlinear systems with full state constraints. ISA Trans. *101* (2020), 60–68. DOI:10.1016/j.isatra.2020.01.021

[33] Q. Zhu, K. Wang, and Z. Shao: Distributed dynamic optimization for chemical process networks based on differential games. Industr. Engrg. Chem. Res. *59* (2020), 2441–2456. DOI:10.1021/acs.iecr.9b04663

Fu Xingjian, School of Automation, Beijing Information Science and Technology University. P. R. China.
e-mail: fxj@bistu.edu.cn

Li Zizheng, School of Automation, Beijing Information Science and Technology University. P. R. China.
e-mail: lzz9408@126.com