Adaptive Dynamic Programming for Two-Player Zero-Sum Differential Games with Completely Unknown Systems

Hongliang Li, *Student Member, IEEE*
Supervisor: Derong Liu and Huaguang Zhang

***Abstract*—In this report, an adaptive dynamic programming algorithm is developed to solve online the Nash equilibrium for two-player zero-sum differential games with completely unknown continuous-time systems. The developed scheme updates the value function, control and disturbance policies at the same time. It is shown that the algorithm is mathematically equivalent to Newton's method. To facilitate the implementation of the algorithm, one critic network and two action networks are used to approximate the value function, control and disturbance policies respectively. The least squares method is used to estimate the unknown parameters. The effectiveness of the developed scheme is demonstrated in the simulation experiment by designing a load-frequency controller for a power system.**

***Index Terms*—Adaptive dynamic programming, approximate dynamic programming, reinforcement learning**

I. Introduction

ADAPTIVE dynamic programming (ADP) [1]–[9] methods can solve optimal control problems forward in time by making use of online measured data. These algorithms have been successfully applied to many areas like residential energy system control [10], engine control [11], and call admission control [12], etc.

For continuous-time dynamical systems, Doya [13] presented a reinforcement learning (RL) framework without a priori discretization of time, state and control. Vamvoudakis and Lewis [14] proposed a synchronous policy iteration (PI) algorithm for learning online the continuous-time optimal control with known dynamics, where both action and critic neural networks were simultaneously tuned. Zhang et al. [15] extended the synchronous PI algorithm to the optimal tracking problem for unknown nonlinear systems, and added a robust term to compensate for the neural network approximation errors. Bhasin et al. [16] presented an actor-critic-identifier structure to implement the PI algorithm without the requirement of complete knowledge of the dynamics. Vrabie et al. derived an integral RL algorithm to obtain direct adaptive optimal control for partially unknown linear and nonlinear systems [17], [18]. Some researchers try to propose adaptive optimal control algorithms for completely unknown systems without identification. Mehta and Meyn [19] established connections between Q-learning and nonlinear optimal control of continuous-time models, and proposed continuous-time Q-learning for completely unknown systems. Lee et al. [20] derived an integral Q-learning for continuous-time linear systems without the knowledge of the system dynamics. Jiang and Jiang [21] presented a computational adaptive optimal control algorithm for continuous-time linear systems with completely unknown system dynamics.

Game theory [22] provides an ideal environment to study multi-player optimal decision and control problems. Two-player noncooperative zero-sum differential game [23] has received much attention since it provides the solution of the H_∞ optimal control [24]. The Nash equilibrium solution is usually obtained by means of offline iterative computation, and requires the exact knowledge of the system dynamics. For nonlinear continuous-time systems, Abu-Khalaf et al. [25], [26] derived an H_∞ suboptimal state feedback controller for constrained input systems. Zhang et al. [27] used four action networks and two critic networks to obtain the saddle point solution of the game problems. Vamvoudakis and Lewis [28] presented an online synchronous PI to solve the continuous-time two-player zero-sum game with infinite horizon cost for nonlinear systems with known dynamics. In [29], a neural-network-based online simultaneous policy update algorithm with only one iterative loop was proposed to solve the zero-sum games for partial unknown systems.

For linear continuous-time systems, finding the Nash equilibrium of the zero-sum game problem reduces to solving the game algebraic Riccati equation (GARE). Vrabie and Lewis [30] proposed an online data-based ADP algorithm based on the idea of integral RL for two-player zero-sum linear differential games without requiring the knowledge of internal system dynamics. Wu and Luo [31] proposed an online simultaneous policy update algorithm for H_∞ state feedback control to improve the efficiency by updating both control and disturbance policies simultaneously. However, it is difficult to obtain the knowledge of the system dynamics for many practical problems.

To the best of our knowledge, there are still not model-free ADP methods for zero-sum games with completely unknown continuous-time systems. In this report, we develop an online model-free ADP algorithm to learn the Nash equi-

The research is funded by the IEEE Computational Intelligence Society Graduate Student Research Grant 2013.

This work was done while the author visited Northeastern University, China. The author would like to thank Derong Liu and Huaguang Zhang for their support in the completion of this research.

Hongliang Li and Derong Liu are with The State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: hongliang.li@ia.ac.cn). Huaguang Zhang is with Northeastern University, Shenyang, China.

librium solution for two-player zero-sum differential games with completely unknown systems. Only one iterative loop is involved to improve the efficiency of the learning process. The developed scheme updates the value function, control and disturbance policies at the same time. It is shown that the algorithm is mathematically equivalent to Newton's method. To facilitate the implementation of the algorithm, one critic network and two action networks are used to approximate the value function, control and disturbance policies respectively. The least squares method is used to estimate the unknown parameters. The effectiveness of the developed scheme is demonstrated in the simulation experiment by designing a load-frequency controller for a power system.

The rest of this report is organized as follows. Section II provides the formulation of a two-payer zero-sum differential game. In Section III, we first develop a model-free ADP algorithm for zero-sum games, then provide the convergence analysis, and finally give the least squares method to estimate the unknown parameters. Section IV presents a simulation example in power system to demonstrate the effectiveness of the developed algorithm and is followed by conclusion and future work in Section V.

Notations: $\mathbb{R}^+$, $\mathbb{R}^n$ and $\mathbb{R}^{n\times m}$ are the set of positive real numbers, the n-dimensional Euclidean space and the set of all real $n\times m$ matrices, respectively. $\|\cdot\|$ denotes the vector norm or matrix norm in $\mathbb{R}^n$ or $\mathbb{R}^{n\times m}$. I_n denotes the n-dimensional identity matrix. Denote $\mathbb{Z}_+$ the set of nonnegative integers. Use $vec(X)$ for $X\in\mathbb{R}^{n\times m}$ as a vectorization map from a matrix into an mn-dimensional column vector which stacks the column of X on top of one another. For $X\in\mathbb{R}^{n\times m}$ and $Y\in\mathbb{R}^{n\times m}$, we let $X\otimes Y$ be a Kronecker product of X and Y. The superscript T is used for the transpose. $\nabla_x f(x,y)\triangleq\partial f(x,y)/\partial x$ denotes a gradient operator notation.

II. Problem Statement

Consider the following continuous-time linear dynamical system described by

$$\dot{x}=Ax+B_1u+B_2w \tag{1}$$

where $x\in\mathbb{R}^n$ is the system state with the initial state x_0, $u\in\mathbb{R}^m$ is the control input, and $w\in\mathbb{R}^q$ is the external disturbance input and $w\in L_2[0,\infty)$. $A\in\mathbb{R}^{n\times n}$, $B_1\in\mathbb{R}^{n\times m}$, and $B_2\in\mathbb{R}^{n\times q}$ are unknown system matrices.

Define the infinite horizon performance index as

$$\begin{aligned}J(x_0,u,w)&=\int_0^\infty(x^\mathsf{T}Qx+u^\mathsf{T}Ru-\gamma^2w^\mathsf{T}w)\mathrm{d}\tau\\&\triangleq\int_0^\infty r(x,u,w)\mathrm{d}\tau\end{aligned} \tag{2}$$

with $Q=Q^\mathsf{T}\geq 0$, $R=R^\mathsf{T}>0$, and a prescribed constant $\gamma\geq\gamma^*\geq 0$, where γ^* denotes the smallest γ for which the system (1) is stabilized. For feedback policy $u(x)$ and disturbance policy $w(x)$, we define the value function of the policies as

$$V(x_t,u,w)=\int_t^\infty(x^\mathsf{T}Qx+u^\mathsf{T}Ru-\gamma^2w^\mathsf{T}w)\mathrm{d}\tau. \tag{3}$$

Then, we define the two-player zero-sum differential game as

$$\begin{aligned}V^*(x_0)&=\min_u\max_w J(x_0,u,w)\\&=\min_u\max_w\int_0^\infty(x^\mathsf{T}Qx+u^\mathsf{T}Ru-\gamma^2w^\mathsf{T}w)\mathrm{d}\tau\end{aligned} \tag{4}$$

where the control policy player u seeks to minimize the performance index while the disturbance policy player w desires to maximize it. The goal is to find the saddle point (u^*,w^*) which satisfies the following inequalities

$$J(x_0,u,w^*)\geq J(x_0,u^*,w^*)\geq J(x_0,u^*,w) \tag{5}$$

for any state feedback control policy u and disturbance policy w.

Denote $u=-Kx$ and $w=Lx$ for the state feedback control policy and the disturbance policy respectively. Then, the value function (3) can be represented as $V(x_t)=x_t^\mathsf{T}Px_t$, where the matrix P is determined by K and L. The saddle point can be obtained by solving the following continuous-time GARE [23]

$$A^\mathsf{T}P^*+P^*A+Q-P^*B_1R^{-1}B_1^\mathsf{T}P^*+\gamma^{-2}P^*B_2B_2^\mathsf{T}P^*=0. \tag{6}$$

Defining P^* as the unique positive definite solution of (6), the saddle point of the zero-sum game is

$$u^*=-K^*x=-R^{-1}B_1^\mathsf{T}P^*x \tag{7}$$

$$w^*=L^*x=\gamma^{-2}B_2^\mathsf{T}P^*x \tag{8}$$

and the value function is

$$V^*(x_0)=x_0^\mathsf{T}P^*x_0. \tag{9}$$

Assume that the pair (A,B_1) is stabilizable and the pair $(A,Q^{1/2})$ is observable, so that a stabilizing control policy exists.

III. Main Results

In this section, we first develop an online model-free ADP algorithm for the zero-sum game with the completely unknown linear continuous-time system, then provide the convergence analysis, and finally present the online implementation using the least squares method.

A. Online Model-free ADP for Zero-sum Games

In this subsection, we will develop an online model-free ADP algorithm for the linear continuous-time zero-sum differential game with completely unknown systems. First, we assume an initial stabilizing control matrix K_0 is known. Define $V_i(x)=x^\mathsf{T}P_ix$, $u_i(x)=-K_ix$ and $w_i(x)=L_ix$ as the value function, control policy and disturbance policy respectively, for each iterative step $i\geq 0$.

To relax the assumptions of exact knowledge on A, B_1 and B_2, we denote e_1 and e_2 to be the exploration signals added to the control policy u_i and disturbance policy w_i respectively. The exploration signals are assumed to be any non-zero measurable signal which is bounded and exactly known a priori. Then the original system (1) becomes

$$\dot{x}=Ax+B_1(u_i+e_1)+B_2(w_i+e_2). \tag{10}$$

Algorithm 1 Online Model-free ADP for Zero-sum Games

Step 1. Give an initial stabilizing policy $u_1 = -K_1 x$ and $w_1 = L_1 x$. Set $i = 1$ and $P_0 = 0$.

Step 2. (Policy Evaluation and Policy Improvement)
For the system (10) with policies $u_i = -K_i x$ and $w_i = L_i x$, and exploration signals e_1 and e_2, solve the following equation for P_i, K_{i+1} and L_{i+1}

$$x_t^T P_i x_t = x_{t+T}^\mathsf{T} P_i x_{t+T} + \int_t^{t+T} r(x, u_i, w_i)\mathrm{d}\tau \qquad (13)$$
$$- 2\int_t^{t+T} x^\mathsf{T} K_{i+1}^\mathsf{T} R e_1 \mathrm{d}\tau - 2\gamma^2 \int_t^{t+T} x^\mathsf{T} L_{i+1}^\mathsf{T} e_2 \mathrm{d}\tau.$$

Step 3. If $\|P_i - P_{i-1}\| \leq \xi$ (ξ is a prescribed small positive real number), stop and output P_i; else, set $i = i + 1$ and go to Step 2.

The derivative of the value function with respect to time is calculated as

$$\dot{V}_i(x) = - x^\mathsf{T} Q x - x^\mathsf{T} K_i^\mathsf{T} R K_i x + \gamma^2 x^\mathsf{T} L_i^\mathsf{T} L_i x \qquad (11)$$
$$+ 2x^\mathsf{T} K_{i+1}^\mathsf{T} R e_1 + 2\gamma^2 x^\mathsf{T} L_{i+1}^\mathsf{T} e_2.$$

Integrating (11) from t and $t+T$ with any time interval $T > 0$, we have

$$x_{t+T}^\mathsf{T} P_i x_{t+T} - x_t^\mathsf{T} P_i x_t = -\int_t^{t+T} r(x, u_i, w_i)\mathrm{d}\tau \qquad (12)$$
$$+2\int_t^{t+T} x^\mathsf{T} K_{i+1}^\mathsf{T} R e_1 \mathrm{d}\tau + 2\gamma^2 \int_t^{t+T} x^\mathsf{T} L_{i+1}^\mathsf{T} e_2 \mathrm{d}\tau$$

where the values of the state at time t and $t + T$ are denoted with x_t and x_{t+T}. Therefore, we obtain the online model-free ADP algorithm for zero-sum differential games.

Remark 1: The equation (13) plays an important role in relaxing the assumption of the knowledge of system dynamics, since A, B_1 and B_2 do not appear in (13) anymore. To run this algorithm, it only requires online data measured along the system trajectories. This method avoids the identification of A, B_1 and B_2 whose information is embedded in the online measured data. In other words, the lack of knowledge about the system dynamics does not have any impact on this method to obtain the Nash equilibrium. Thus, this method will not be affected by the errors between the identification model and the real system, and it can respond fast to the change of the system dynamics.

Remark 2: This algorithm is actually the PI method, but the policy evaluation and policy improvement are performed at the same time. Compared with the model-based method [28] and partially model-free method [31], this algorithm is a fully model-free method which does not require any knowledge of the system dynamics. Different from the iterative method with inner loop on disturbance policy and outer loop on control policy [28], and the method with only one iterative loop by updating control and disturbance policies simultaneously [31], the developed method here updates the value function, control and disturbance policies at the same time. Hence, this method will have higher efficiency.

Remark 3: To guarantee persistence of excitation condition, the state may need to be reset during the iterative process, but it results in technical problems for stability analysis of the closed-loop system. An alternative way is to add exploration noises. However, the added exploration noises may make the solution different from the exact one determined by the GARE. Compared with these methods, we consider the effects of exploration noises. Therefore, the solution obtained by our method is exactly the same as the one determined by the GARE.

B. Convergence Analysis

In this part, we will provide a convergence analysis of the developed algorithm for two-player zero-sum differential games. It can be shown that the developed model-free ADP algorithm is equivalent to Newton's method.

Theorem 1: For an initial stabilizing control policy $u_1 = -K_1 x$, the sequences of $\{P_i\}_{i=1}^\infty$, $\{K_i\}_{i=1}^\infty$, and $\{L_i\}_{i=1}^\infty$ obtained by solving (13) converge to the optimal solution P^* of GARE, the saddle point K^*, and L^* respectively, as $i \to \infty$.

Proof: For an initial stabilizing control policy $u_1 = -K_1 x$, we can prove that the developed algorithm is equivalent to the following Lyapunov equation

$$A_i^\mathsf{T} P_i + P_i A_i = -M_i, \qquad (14)$$

where

$$A_i = A - B_1 K_i + B_2 L_i \qquad (15)$$
$$M_i = Q + K_i^\mathsf{T} R K_i - \gamma^2 L_i^\mathsf{T} L_i. \qquad (16)$$

With the control policy $u_i = -K_i x$, the disturbance policy $w_i = L_i x$, and the exploration signals e_1 and e_2, the closed-loop system (1) becomes

$$\dot{x} = A_i x + B_1 e_1 + B_2 e_2, \qquad (17)$$

where $A_i = A - B_1 K_i + B_2 L_i$. Considering the Lyapunov function $V_i(x) = x^\mathsf{T} P_i x$, its derivative can be calculated as

$$\dot{V}_i(x) = \dot{x}^\mathsf{T} P_i x + x^\mathsf{T} P_i \dot{x} = x^\mathsf{T} A_i^\mathsf{T} P_i x + x^\mathsf{T} P_i A_i x \qquad (18)$$
$$+ (B_1 e_1 + B_2 e_2)^\mathsf{T} P_i x + x^\mathsf{T} P_i (B_1 e_1 + B_2 e_2)$$
$$= x^\mathsf{T}(A_i^\mathsf{T} P_i + P_i A_i)x + 2x^\mathsf{T} K_{i+1}^\mathsf{T} R e_1 + 2\gamma^2 x^\mathsf{T} L_{i+1}^\mathsf{T} e_2.$$

Integrating (18) from t and $t + T$ yields

$$V_i(x_{t+T}) - V_i(x_t) = \int_t^{t+T} x^\mathsf{T}(A_i^\mathsf{T} P_i + P_i A_i)x \mathrm{d}\tau \qquad (19)$$
$$+ 2\int_t^{t+T} x^\mathsf{T} K_{i+1}^\mathsf{T} R e_1 \mathrm{d}\tau + 2\int_t^{t+T} \gamma^2 x^\mathsf{T} L_{i+1}^\mathsf{T} e_2 \mathrm{d}\tau.$$

From (13), we can have

$$V_i(x_{t+T}) - V_i(x_t) = -\int_t^{t+T} r(x, u_i, w_i)\mathrm{d}\tau \qquad (20)$$
$$+ 2\int_t^{t+T} x^\mathsf{T} K_{i+1}^\mathsf{T} R e_1 \mathrm{d}\tau + 2\int_t^{t+T} \gamma^2 x^\mathsf{T} L_{i+1}^\mathsf{T} e_2 \mathrm{d}\tau.$$

Therefore, considering (19) and (20), we can get

$$x^\mathsf{T}(A_i^\mathsf{T} P_i + P_i A_i)x = -r(x, u_i, w_i) \qquad (21)$$
$$= -x^\mathsf{T}(Q + K_i^\mathsf{T} R K_i - \gamma^2 L_i^\mathsf{T} L_i)x$$

i.e.,

$$A_i^{\mathsf{T}} P_i + P_i A_i = -M_i \tag{22}$$

where

$$M_i = Q + K_i^{\mathsf{T}} R K_i - \gamma^2 L_i^{\mathsf{T}} L_i. \tag{23}$$

According to the result in [31], the sequence $\{P_i\}_{i=1}^{\infty}$ generated by (14) is equivalent to Newton's method and converges to the optimal solution P^* of GARE, as $i \to \infty$. Furthermore, the sequences of $\{K_i\}_{i=1}^{\infty}$ and $\{L_i\}_{i=1}^{\infty}$ converge to the saddle point K^* and L^*, as $i \to \infty$.

■

C. Online Implementation

In this part, we will present an online implementation of the developed model-free ADP algorithm with least squares method. Here parametric structures are used to approximate the value function, control policy and disturbance policy.

Given a stabilizing control policy $u_i = -K_i x$, a pair of matrices (P_i, K_{i+1}, L_{i+1}) with $P_i = P_i^{\mathsf{T}} > 0$, can be uniquely determined by (13). We define the following two operators: $P \in \mathbb{R}^{n\times n} \to \hat{P} \in \mathbb{R}^{\frac{1}{2}n\times(n+1)}$, $x \in \mathbb{R}^n \to \bar{x} \in \mathbb{R}^{\frac{1}{2}n\times(n+1)}$, where

$$\hat{P} = [p_{11}, 2p_{12}, \ldots, 2p_{1n}, p_{22}, 2p_{23}, \ldots, 2p_{(n-1)n}, p_{nn}]^{\mathsf{T}} \tag{24}$$

$$\bar{x} = [x_1^2, x_1x_2, \ldots, x_1x_n, x_2^2, x_2x_3, \ldots, x_{n-1}x_n, x_n^2]^{\mathsf{T}}. \tag{25}$$

Hence we can have

$$\begin{aligned} & x_{t+(k-1)T}^{\mathsf{T}} P_i x_{t+(k-1)T} - x_{t+kT}^{\mathsf{T}} P_i x_{t+kT} \\ & = (\bar{x}_{t+(k-1)T} - \bar{x}_{t+kT})^{\mathsf{T}} \hat{P} \end{aligned} \tag{26}$$

where $k \in \mathbb{Z}_+$ and $k \geq 1$. Using Kronecker product $\otimes$, we can obtain

$$x^{\mathsf{T}} K_{i+1}^{\mathsf{T}} R e_1 = (x \otimes e_1)^{\mathsf{T}} (I_n \otimes R) vec(K_{i+1}) \tag{27}$$

$$x^{\mathsf{T}} L_{i+1}^{\mathsf{T}} e_2 = (x \otimes e_2)^{\mathsf{T}} vec(L_{i+1}). \tag{28}$$

Using the expressions established above, (13) can be rewritten to be a general compact form

$$\psi_k^{\mathsf{T}} \begin{bmatrix} \hat{P}_i \\ vec(K_{i+1}) \\ vec(L_{i+1}) \end{bmatrix} = \theta_k, \quad \forall i \in \mathbb{Z}_+ \tag{29}$$

with

$$\theta_k = \int_{t+(k-1)T}^{t+kT} r(x, u_i, w_i) \mathrm{d}\tau \tag{30}$$

$$\begin{aligned} \psi_k = \Bigg[& (\bar{x}_{t+(k-1)T} - \bar{x}_{t+kT})^{\mathsf{T}}, 2\int_{t+(k-1)T}^{t+kT} (x \otimes e_1)^{\mathsf{T}} \mathrm{d}\tau (I_n \otimes R), \\ & 2\gamma^2 \int_{t+(k-1)T}^{t+kT} (x \otimes e_2)^{\mathsf{T}} \mathrm{d}\tau \Bigg]^{\mathsf{T}} \end{aligned} \tag{31}$$

where the measurement time is from $t + (k-1)T$ to $t + kT$. Since (29) is only a 1-dimensional equation, the uniqueness of the solution can not be guaranteed. We will use the least squares method to solve this problem, where the parameter vector is found in a least squares sense over a compact set Ω.

For any positive integer N, denote $\Phi = [\psi_1, \ \ldots, \ \psi_N]$ and $\Theta = [\theta_1, \ \ldots, \ \theta_N]^{\mathsf{T}}$. Then we have the following N-dimensional equation

$$\Phi^{\mathsf{T}} \begin{bmatrix} \hat{P}_i \\ vec(K_{i+1}) \\ vec(L_{i+1}) \end{bmatrix} = \Theta, \quad \forall i \in \mathbb{Z}_+. \tag{32}$$

If Φ^{T} has full column rank, the parameters can be solved by

$$\begin{bmatrix} \hat{P}_i \\ vec(K_{i+1}) \\ vec(L_{i+1}) \end{bmatrix} = (\Phi\Phi^{\mathsf{T}})^{-1}\Phi\Theta. \tag{33}$$

Therefore, we need to have the number of collected points N at least $N_{\min} = rank(\Phi)$, i.e.,

$$N_{\min} = \frac{n(n+1)}{2} + nm + nq, \tag{34}$$

which will make $(\Phi\Phi^{\mathsf{T}})^{-1}$ exist.

The least squares problem in (33) can be solved in real time by collecting enough data points generated from the system (10). The solution can be obtained using the batch least squares algorithm, the recursive least squares algorithm, or the gradient descent algorithm.

The sequence $\{\hat{P}_i\}_{i=1}^{\infty}$ calculated by the least squares method converges to the approximate solution of GARE. The persistence of excitation condition is required to make the algorithm converge. Several types of exploration signal have been used, such as piecewise constant exploration signals [20], sinusoidal signals with different frequencies [21], and exponentially decreasing probing noise [28].

IV. Simulation Study

In this section, we will demonstrate the effectiveness of the developed algorithm by designing an H_∞ robust load-frequency controller for a power system.

Consider the following linear model of a power system that was studied in [30]

$$\begin{aligned} \dot{x} &= Ax + B_1 u + B_2 w \\ &= \begin{bmatrix} -0.0665 & 8 & 0 & 0 \\ 0 & -3.663 & 3.663 & 0 \\ -6.86 & 0 & -13.736 & -13.736 \\ 0.6 & 0 & 0 & 0 \end{bmatrix} x \\ &\quad + \begin{bmatrix} 0 \\ 0 \\ 13.736 \\ 0 \end{bmatrix} u + \begin{bmatrix} -8 \\ 0 \\ 0 \\ 0 \end{bmatrix} w \end{aligned} \tag{35}$$

where the state vector is $x = [\triangle f \ \ \triangle P_g \ \ \triangle X_g \ \ \triangle E]^{\mathsf{T}}$, $\triangle f$ (Hz) is the incremental frequency deviation, $\triangle P_g$ $(p.u.MW)$ is the incremental change in generator output, $\triangle X_g$ $(p.u.MW)$ is the incremental change in governor value position, and $\triangle E$ is the incremental change in integral control. We assume that the exact knowledge of the dynamics is completely unknown. The matrices Q and R in the performance index are identity matrices of appropriate dimensions, and

$\gamma = 3.5$. Using the system model (35), the optimal value function of the zero-sum game is

$$P^* = \begin{bmatrix} 0.8335 & 0.9649 & 0.1379 & 0.8005 \\ 0.9649 & 1.4751 & 0.2358 & 0.8046 \\ 0.1379 & 0.2358 & 0.0696 & 0.0955 \\ 0.8005 & 0.8046 & 0.0955 & 2.6716 \end{bmatrix}. \quad (36)$$

Now we will use the developed online model-free ADP algorithm to solve this problem. The initial state is selected as $x_0 = [0.1\ 0.2\ 0.2\ 0.1]^{\mathsf{T}}$. The simulation is conducted using data obtained along the system trajectory at every $0.01s$. The least squares problem is solved after 50 data samples are acquired, and thus the parameters of the control policy is updated every $0.5s$. The parameters of the critic network, the control action network and the disturbance action network are all initialized to zero. The persistence of excitation condition is ensured by adding a small probing noise to the control and disturbance inputs.

Fig. 1 presents the evolution of the parameters of the critic network during the learning process. It is clear that the developed algorithm is convergent after 10 iterative steps. The obtained approximate value function is given by the matrix

$$P_{10} = \begin{bmatrix} 0.8335 & 0.9649 & 0.1379 & 0.8005 \\ 0.9649 & 1.4752 & 0.2359 & 0.8047 \\ 0.1379 & 0.2359 & 0.0696 & 0.0956 \\ 0.8005 & 0.8047 & 0.0956 & 2.6718 \end{bmatrix}, \quad (37)$$

and $\|P_{10} - P^*\| = 2.9375 \times 10^{-4}$. We can find that the solution obtained by the online model-free ADP algorithm is quite close to the exact one obtained by solving GARE. Fig. 2 and Fig. 3 show the convergence process of the parameters of the control and disturbance action networks. The obtained H_∞ state feedback control policy is $u_{11} = -[1.8941,\ 3.2397,\ 0.9563,\ 1.3126]x$.

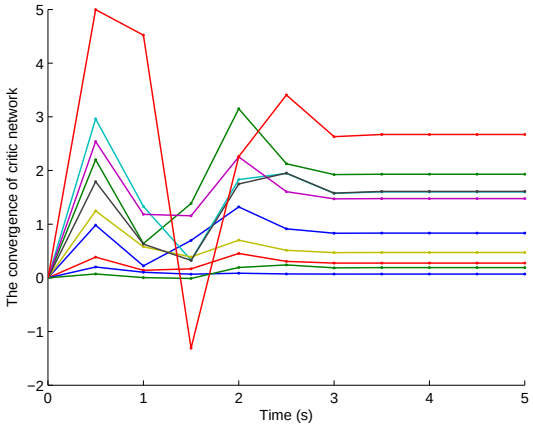

Fig. 1. Convergence of the game value function matrix P_i

V. Conclusion and Future Work

In this research, we developed a novel adaptive dynamic programming algorithm to learn online the Nash equilibrium for two-player zero-sum differential games with completely unknown continuous-time systems. We also proved the convergence of this algorithm. One critic network and two action networks were used to approximate the value function, control and disturbance policies to implement the algorithm. And the least squares method was adopted to estimate the unknown parameters. We applied the developed scheme to design an H_∞ state feedback control for a power system. In the future, we will extend the results to two-player zero-sum differential games with completely unknown nonlinear continuous-time dynamics.

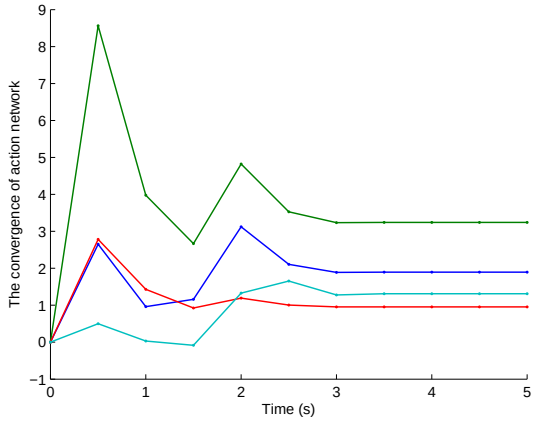

Fig. 2. Convergence of the control action network parameters K_i

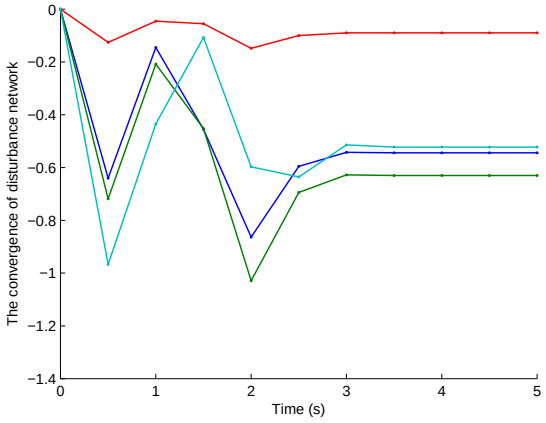

Fig. 3. Convergence of the disturbance action network parameters L_i

References

[1] F. L. Lewis and D. Liu, *Approximate Dynamic Programming and Reinforcement Learning for Feedback Control*, Hoboken, NJ: Wiley, 2012.

[2] F. Y. Wang, H. Zhang, and D. Liu, "Adaptive dynamic programming: an introduction," *IEEE Computational Intelligence Magazine*, vol. 4, no. 2, pp. 39–47, May 2009.

[3] F. L. Lewis and D. Vrabie, "Reinforcement learning and adaptive dynamic programming for feedback control," *IEEE Circuits and Systems Magazine*, vol. 9, no. 3, pp. 32–50, July 2009.

[4] F. L. Lewis and D. Vrabie, "Reinforcement learning and feedback control: using natural decision methods to design optimal adaptive controllers," *IEEE Control Systems Magazine*, vol. 32, no. 6, pp. 76–105, Dec. 2012.

[5] A. Al-Tamimi, F. L. Lewis, and M. Abu-Khalaf, "Discrete-time nonlinear HJB solution using approximate dynamic programming: convergence proof," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 38, no. 4, pp. 943–949, Aug. 2008.

[6] F. Wang, N. Jin, D. Liu, and Q. Wei, "Adaptive dynamic programming for finite-horizon optimal control of discrete-time nonlinear sytems with ε-error bound," *IEEE Trans. Neural Netw.*, vol. 22, no. 12, pp. 1854–1862, Dec. 2011.

[7] D. Wang, D. Liu, Q. Wei, D. Zhao, and N. Jin, "Optimal control of unknown nonaffine nonlinear discrete-time systems based on adaptive dynamic programming," *Automatica*, vol. 48, no. 8, pp. 1825–1832, Aug. 2012.

[8] D. Liu, D. Wang, D. Zhao, Q. Wei, and N. Jin, "Neural-network-based optimal control for a class of unknown discrete-time nonlinear systems using globalized dual heuristic programming," *IEEE Trans. Autom. Sci. Eng.*, vol. 9, no. 3, pp. 628–634, July 2012.

[9] H. Li and D. Liu, "Optimal control for discrete-time affine nonlinear systems using general value iteration," *IET Control Theory and Applications*, vol. 6, no. 18, pp. 2725–2736, Dec. 2012.

[10] T. Huang and D. Liu, "A self-learning scheme for residential energy system control and management," *Neural Computing and Applications*, vol. 22, no. 2, pp. 259–269, Feb. 2013.

[11] D. Liu, H. Javaherian, O. Kovalenko, and T. Huang, "Adaptive critic learning techniques for engine torque and air–fuel ratio control," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 38, no. 4, pp. 988–993, Aug. 2008.

[12] D. Liu, Y. Zhang, and H. Zhang, "A self-learning call admission control scheme for CDMA cellular networks," *IEEE Trans. Neural Netw.*, vol. 16, no. 5, pp. 1219–1228, Sept. 2005.

[13] K. Doya, "Reinforcement learning in continuous time and space," *Neural Computation*, vol. 12, no. 1, pp. 219–245, 2000.

[14] K. G. Vamvoudakis and F. L. Lewis, "Online actor-critic algorithm to solve the continuous-time infinite horizon optimal control problem," *Automatica*, vol. 46, no. 5, pp. 878–888, May 2010.

[15] H. Zhang, L. Cui, X. Zhang, and Y. Luo, "Data-driven robust approximate optimal tracking control for unknown general nonlinear systems using adaptive dynamic programming method," *IEEE Trans. Neural Netw.*, vol. 22, no. 12, pp. 2226–2236, Dec. 2011.

[16] S. Bhasin, R. Kamalapurkar, M. Johnson, K. G. Vamvoudakis, F. L. Lewis, and W. E. Dixon, "A novel actor-critic-identifier architecture for approximate optimal control of uncertain nonlinear systems," *Automatica*, vol. 49, no. 1, pp. 82–92, Jan. 2013.

[17] D. Vrabie, O. Pastravanu, M. Abu-Khalaf, and F. L. Lewis, "Adaptive optimal control for continuous-time linear systems based on policy iteration," *Automatica*, vol. 45, no. 2, pp. 477–484, Feb. 2009.

[18] D. Vrabie and F. L. Lewis, "Neural network approach to continuous-time direct adaptive optimal control for partially unknown nonlinear systems," *Neural Networks*, vol. 22, no. 3, pp. 237–246, Apr. 2009.

[19] P. Mehta and S. Meyn, "Q-learning and pontryagins minimum principle," in *Proceedings of IEEE Conference on Decision and Control*, Shanghai, China, Dec. 2009, pp. 3598–3605.

[20] J. Y. Lee, J. B. Park, and Y. H. Choi, "Integral Q-learning and explorized policy iteration for adaptive optimal control of continuous-time linear systems," *Automatica*, vol. 48, no. 11, pp. 2850–2859, Nov. 2012.

[21] Y. Jiang and Z.-P. Jiang, "Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics," *Automatica*, vol. 48, no. 10, pp. 2699–2704, Oct. 2012.

[22] S. Tijs, *Introduction to game theory*, India Hindustan Book Agency: 2003.

[23] T. Basar and G. J. Olsder, *Dynamic Noncooperative Game*, Second Edition, Boston: 1997.

[24] T. Basar and P. Bernhard, *H_∞ Optimal Conrol and Related Minimax Design Problems: A Dynamic Game Approach*, Second Edition, Boston: 1995.

[25] M. Abu-Khalaf, F. L. Lewis, and J. Huang, "Policy iterations and the Hamilton-Jacobi-Isaacs equation for H_∞ state feedback control with input saturation," *IEEE Trans. Autom. Control*, vol. 51, no. 12, pp. 1989–1995, Dec. 2006.

[26] M. Abu-Khalaf, F. L. Lewis, and J. Huang, "Neurodynamic progarmming and zero-sum games for constrained control systems," *IEEE Trans. Neural Netw.*, vol. 19, no. 7, pp. 1243–1252, July 2008.

[27] H. Zhang, Q. Wei, and D. Liu, "An iterative adaptive dynamic programming method for solving a class of nonlinear zero-sum differential games," *Automatica*, vol. 47, no. 1, pp. 207–214, Jan. 2011.

[28] K. G. Vamvoudakis and F. L. Lewis, "Online solution of nonlinear two-player zero-sum games using synchronous policy iteration," *Int. J. Robust Nonlinear Control*, vol. 22, no. 13, pp. 1460–1483, 2011.

[29] H. Wu and B. Luo, "Neural network based online simultaneous policy update algorithm for solving the HJI equation in nonlinear H_∞ control," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 12, pp. 1884–1895, Dec. 2012.

[30] D. Varbie and F. L. Lewis, "Adaptive dynamic programming for online solution of a zero-sum differential game," *J. Control Theory Appl*, vol. 9, no. 3, pp. 353–360, 2011.

[31] H. Wu and B. Luo, "Simultaneous policy update algorithms for learning the solution of linear continuous-time H_∞ state feedback control," *Information Sciences*, vol. 222, no. 10, pp. 472–485, Feb. 2013.