2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

Model-free Adaptive Dynamic Programming for Online optimal Solution of the Unknown Nonlinear Zero-Sum Differential Game

Chunbin Qin
School of Computer and Information Engineering,
Henan University,
Kaifeng 475004,China
Email: qcb@henu.edu.cn

Huaguang Zhang
School of Information Science and Engineering,
Northeastern University
Shenyang 110004, China
Email: hgzhang@ieee.org

Yanhong Luo
School of Information Science and Engineering,
Northeastern University
Shenyang 110004, China
Email: neuluo@gmail.com

***Abstract*—It is well known that the two-player zero-sum differential game problem of the continuous-time nonlinear system relies on the solution of the Hamilton-Jacobi-Isaacs equation, which is a nonlinear partial differential equation that is difficult or impossible to solve. In this paper, a new model-free adaptive dynamic programming algorithm is developed for solving online the Hamilton-Jacobi-Isaacs equation for continuous-time nonlinear system with the fully unknown knowledge of the system dynamics. First, a simultaneous policy iteration algorithm will be given, which can solve the Hamilton-Jacobi-Isaacs equation in an off-line sense, in which the fully knowledge of the system dynamics is required. Second, based on the simultaneous policy iteration algorithm, a new model-free adaptive dynamic programming algorithm is developed for solving online the Hamilton-Jacobi-Isaacs equation, in which the fully knowledge of the system dynamics is not required. Finally, a numerical example is given to demonstrate the convergence and effectiveness of the proposed scheme.**

I. Introduction

In the practice, there are a large class of real systems which are controlled by more than one controller or decision maker with each using an individual strategy [1]. These controllers often operate in a group with a performance index function as game theory which has been widely applied in management, military battles, power networks and different types of contest. The two-player zero-sum game with a quadratic performance index plays an important role in the game theory. In the two-player zero-sum game, one player tries to minimize the performance index while the other tries to maximize it. Over the past few decades, a large number of theoretical results on the two-player zero-sum game have been reported [1]–[7]. Although the two-player zero-sum game theory has been well developed, the main bottleneck for its practical application is the need to solve the Hamilton-Jacobi-Isaacs (HJI) equation. However, the HJI equation is difficult or impossible to solve, and may not have global analytic solutions even in simple cases, since the HJI equation is a nonlinear partial differential equation.

In recent years, adaptive dynamic programming (ADP), which is a practical method for finding the optimal control solution online forward in time by using measured system data along the system trajectories [8]–[15], have appeared to be promising technique for approximately solving the two-player zero-sum game [16]–[21]. Abu-Khalaf et al. proposed the policy iteration for the affine nonlinear zero-sum game problem in [19] and used neural networks to solve it in [20]. Wei et al. [21] proposed an optimal (or suboptimal) control technique to a class of nonlinear quadratic two-person zero-sum problem where the general saddle point or the Nash equilibrium may not exist. Furthermore, Zhang et al. [22] proposed a new iterative ADP method which is effective both for the situations that the saddle point exists or does not exist. Authors in [23] proposed a computationally efficient simultaneous policy update algorithm (SPUA) based on Galerkins method for solving the two-player zero-sum problem, which is much simpler and easier to implement than the existing methods [19], [20], [22]. However, these methods are the off-line algorithms, that is to say that these methods are implemented in an off-line sense for finding the general saddle point or the Nash equilibrium of the two-person zero-sum game. Thus, authors [24], [25] developed the on-line algorithm for solving the HJI equation appearing in the two-player zero-sum game. Vamvoudakis and Lewis [24] proposed a new optimal adaptive algorithm that solved online the continuous-time two-player zero-sum game problem for affine in the inputs nonlinear systems with the known knowledge of system dynamics, in which the critic, actor, and disturbance neural networks are tuned simultaneously online to converge to the solution to the HJI equation and the saddle point policies. Wu and Luo [25] proposed a neural network (NN)-based online simultaneous policy update algorithm (SPUA) to solve the HJI equation, in which knowledge of internal system dynamics is not required.

However, a common feature of the all the existing ADP-based results for finding the solution to the HJI equation is that partial knowledge of the system dynamic is required to be exactly known in the setting of continuous-time nonlinear systems [19], [20], [22], [24], [25]. To remove the requirement

978-1-4799-1484-5/14/$31.00 ©2014 IEEE

of complete knowledge of the system dynamics, in this paper, we will propose a new model-free ADP algorithm which can solve online the HJI equation appearing in the two-player zero-sum game problems for continuous-time nonlinear systems with the fully unknown knowledge of the system dynamics. That is to say that the proposed algorithm only use the data generated in real time along the unknown system trajectories, and can learn the optimal Nash equilibrium solution of the two-player zero-sum game in an online sense without requiring the full knowledge of system dynamics.

The remainder of this paper is organized as follows. In Section II, we give the problem description. In Section III, we develop the online model-free ADP algorithm for the two-player zero-sum differential game. One numerical examples is provided in Section IV. Finally, the conclusions are drawn in Section V.

II. Problem Formulation

Consider the following nonlinear continuous-time system:

$$\dot{x}(t) = f(x) + g_1(x)u_1(t) + g_2(x)u_2(t) \tag{1}$$

where, $x(t) \in R^n$ is the system state vector, $u_1(t) \in R$ and $u_2(t) \in R$ are the control input vectors. $f(x)$, $g_1(x)$ and $g_2(x)$ are unknown continuous matrix functions of appropriate dimensions. The cost functional for the nonlinear system (1) is defined as

$$V(x(0), u_1(t), u_2(t)) = \int_0^\infty (x^T(t)Qx(t) + u_1^T(t)Ru_1(t) - \gamma^2 u_2^T(t)u_2(t))dt, \tag{2}$$

with $Q > 0, R > 0$ and $\gamma > 0$.

The two-player zero-sum differential game for the system (1) is to find the two optimal feedback control policies $u_1^*(x)$ and $u_2^*(x)$, such that $u_1^*(x)$ tries to minimize the cost functional (2) while $u_2^*(x)$ attempts to maximize the cost functional (2). That is to say that the goal is to find a saddle point ($u_1^*(x)$, $u_2^*(x)$) such that

$$V(u_1^*(x), u_2^*(x)) = \min_{u_1} \max_{u_2} V(u_1(t), u_2(t)), \tag{3}$$

and the following inequality

$$V(u_1^*(x), u_2(x)) \le V(u_1^*(x), u_2^*(x)) \le V(u_1(x), u_2^*(x)) \tag{4}$$

holds for any admissible control policies u_1 and u_2.

While the system dynamics are known, by the game theory [1], the closed-loop Nash equilibrium strategies are given by

$$u_1^*(x) = -\frac{1}{2}R^{-1}g_1^T(x)\nabla V^* \tag{5}$$

$$u_2^*(x) = \frac{1}{2}\gamma^{-2}g_2^T(x)\nabla V^* \tag{6}$$

where $\nabla V^* = \partial V^*/\partial t$, and $V^*(x)$ is the positive positive definite solution of the following HJI equation

$$\begin{aligned} 0 =& x^T Qx + (\nabla V^*(x))^T f(x) \\ & - \frac{1}{4}(\nabla V^*(x))^T g_1(x)R^{-1}g_1^T(x)\nabla V^*(x) \\ & + \frac{1}{4}\gamma^{-2}(\nabla V^*(x))^T g_2(x)g_2^T(x)\nabla V^*(x). \end{aligned} \tag{7}$$

However, the HJI equation (7) is difficult or impossible to solve, since the HJI equation (7) is a nonlinear partial differential equation. Thus, based on the simultaneous policy update algorithm [23], we will give the following simultaneous policy iteration algorithm for solving the HJI equation (7).

Algorithm 1: Simultaneous policy iteration algorithm

Step 1. Give the initial stabilizing control policies u_1^0 and u_2^0. Set $i = 0$.

Step 2. Solve the following Lyapunov function equation for the cost function V^i:

$$\begin{aligned} 0 =& (\nabla V^i)^T(f + g_1u_1^i + g_2u_2^i) \\ & + x^TQx + (u_1^i)^TRu_1^i - \gamma^2(u_2^i)^Tu_2^i, \end{aligned} \tag{8}$$

Step 3. Update the control policies with

$$u_1^{i+1} = -\frac{1}{2}R^{-1}g_1^T(x)\nabla V^i \tag{9}$$

$$u_2^{i+1} = \frac{1}{2}\gamma^{-2}g_2^T(x)\nabla V^i, \tag{10}$$

Step 4. Let $i = i + 1$. If $\| V^i - V^{i-1} \| \le \epsilon$ (the constant $\epsilon > 0$ is a predefined small threshold), go to Step 5; else, go to Step 2 and continue.

Step 5. Stop.

The simultaneous policy iteration algorithm can be seen as a trivial extension of the simultaneous policy update algorithm. Thus, the convergence of the simultaneous policy iteration algorithm is guaranteed by the following theorem, which can be seen as a trivial extension of Theorem 1 in [23].

Theorem 1: Consider V^i defined in (8), u_1^{i+1} and u_2^{i+1} defined in (9) and (10). If the solution $V^*(x)$ of (7) exists, then, for each given x, when i goes to infinity, V^i converges to $V^*(x)$, u_1^{i+1} and u_2^{i+1} also converge to u_1^* and u_2^*, respectively. Besides, for all $i = 0, 1, \cdots$, u_1^{i+1} and u_2^{i+1} are admissible.

According to Algorithm 1 and Theorem 1, it is shown that, by iteratively solving the equations (8) and (9, 10), the solution $V^*(x)$ for the HJI equation (7) can be obtained. And then, we can find the saddle point ($u_1^*(x)$, $u_2^*(x)$). However, the algorithm 1 is implemented off-line and requires the completely knowledge of the system dynamics. To avoid making use of any knowledge on the drift dynamics of the system (i.e. $f(x)$), Wu et al. [25] proposed an online simultaneous policy update algorithm for solving the HJI equation (7). In the online simultaneous policy update algorithm, (8) is implemented online by

$$V^i(x(t)) - V^i(x(t+\Delta t)) = \int_t^{t+\Delta t} x^TQx + (u_1^i)^TRu_1^i - \gamma^2(u_2^i)^Tu_2^i dt, \tag{11}$$

where u_1^i and u_2^i are the control policies of the system on the time interval$[t, t + \Delta t]$.

It is seen that the knowledge on the drift dynamics of the system ($f(x)$) is not needed for implementing the online simultaneous policy update algorithm, since the states $x(t)$ and $x(t + \Delta t)$ contain the information of the system matrix $f(x)$. However, as we can see from (9) and (10), the exact

knowledge of the system matrix $g_1(x)$ and $g_2(x)$ are required for the online SPUA. Therefore, in this paper, we will develop a new online scheme without requiring the full knowledge of system dynamics for obtaining the optimal Nash equilibrium solution of the two-player zero-sum differential game.

III. Main results

In this section, we will present a model-free adaptive dynamic programming scheme for finding online solution of the two-player zero-sum differential differential game of the nonlinear continuous-time system without requiring the full knowledge of system dynamics. First, inspired by [23], [25], [26], and based on the Theorem 1, we can obtain the following lemma.

Lemma 1: Assume that the control policies $u_1(t)$ and $u_2(t)$ make the system (1) stable. Let V^i be the solution of the equation (8), let u_1^{i+1} be obtained by (9) and u_2^{i+1} be obtained by (10). Then, for any time interval $[t, t+\Delta t]$, the following equation holds,

$$\begin{aligned} V^i(x(t+\Delta t)) - V^i(x(t)) = \\ - \int_t^{t+\Delta t} [x^T Q x + (u_1^i)^T R u_1^i - \gamma^2 (u_2^i)^T u_2^i] dt \\ - 2\int_t^{t+\Delta t} (R u_1^{i+1}(t))^T w_1^i(t) dt \\ + 2\gamma^2 \int_t^{t+\Delta t} (u_2^{i+1}(t))^T w_2^i(t) dt. \end{aligned} \tag{12}$$

Proof: First, for giving the stabilizing control policies $u_1(t)$ and $u_2(t)$, the system (1) can be rewritten as

$$\begin{aligned} \dot{x}(t) =& f(x) + g_1(x)u_1^i(t) + g_2(x)u_2^i(t) \\ &+ g_1(x)w_1^i(t) + g_2(x)w_2^i(t) \end{aligned} \tag{13}$$

where $w_1^i(t) = u_1(t) - u_1^i(t)$, $w_2^i(t) = u_2(t) - u_2^i(t)$.

For each $i > 0$ and $V^i(x)$, we have the time derivative of $V^i(x)$ along the trajectories of (13):

$$\begin{aligned} \dot{V}^i(x) =& (\nabla V^i)^T (f(x) + g_1(x)u_1^i(t) + g_2(x)u_2^i(t)) \\ &+ (\nabla V^i)^T (g_1(x)w_1^i(t) + g_2(x)w_2^i(t)). \end{aligned} \tag{14}$$

According to (8), (9) and (10), the equation (14) can be rewritten as

$$\begin{aligned} \dot{V}^i(x) =& -(x^T Q x + (u_1^i)^T R u_1^i - \gamma^2 (u_2^i)^T u_2^i) \\ &- 2(R u_1^{i+1}(t))^T w_1^i(t) \\ &+ 2\gamma^2 (u_2^{i+1}(t))^T w_2^i(t). \end{aligned} \tag{15}$$

For any time interval $[t, t+\Delta t]$, by integrating both sides of (15), we can obtain

$$\begin{aligned} V^i(x(t+\Delta t)) - V^i(x(t)) = \\ \int_t^{t+\Delta t} -(x^T Q x + (u_1^i)^T R u_1^i - \gamma^2 (u_2^i)^T u_2^i) dt \\ - 2\int_t^{t+\Delta t} (R u_1^{i+1}(t))^T w_1^i(t) dt \\ + 2\gamma^2 \int_t^{t+\Delta t} (u_2^{i+1}(t))^T w_2^i(t) dt. \end{aligned} \tag{16}$$

It is shown that the equation (16) is equal to the equation (12). This completes proof.

Note that the equation (16) contains V^i, u_1^{i+1} and u_2^{i+1}, which are obtained by the equations (8, 9, 10). This means that the unknown parameters (V^i, u_1^{i+1}, u_2^{i+1}) can be obtained by only using the equation (16). That is to say that the equation (12) is equal to the equations (8, 9, 10) in some degree. In the other side, (8) contains $f(x), g_1(x)$ and $g_2(x)$, (9) contains $g_1(x)$, (10) contains $g_2(x)$, it is to say that the unknown parameters (V^i, u_1^{i+1}, u_2^{i+1}) can be obtained by using the equations (8, 9, 10), while the fully knowledge of the system dynamics must be known. Besides, in the simultaneous policy update algorithm, $g_1(x)$ and $g_2(x)$ must be known for solving the unknown parameters (V^i, u_1^{i+1}, u_2^{i+1}). However, the equation (16) do not contain the any knowledge of the system dynamics, thus, we can use the equation (16) to obtain the unknown parameters (V^i, u_1^{i+1}, u_2^{i+1}) for the fully unknown knowledge of the system dynamics. Next, based on the equation (16), we will present a model-free adaptive dynamic programming algorithm for online solution of the two-player zero-sum differential game of the nonlinear continuous-time system with the fully unknown knowledge of the system dynamics.

To solve the unknown functions (V^i, u_1^{i+1}, u_2^{i+1}) in (12), we assume that the unknown functions (V^i, u_1^{i+1}, u_2^{i+1}) are the smooth functions. Then, as like [27], we can use neural networks to solve the unknown functions (V^i, u_1^{i+1}, u_2^{i+1}) along with the theory of successive approximation. Therefore, $V^i(x)$ is approximated by

$$V^i(x) = (W_v^i)^T \Phi_v^i(x), \tag{17}$$

which is a neural network with the activation functions $\phi_j^i(x)$, and $\phi_j^i(0) = 0$. The neural network weights are w_j^i. Assume that the number of hidden-layer neurons is L. Then, $\Phi_v^i(x) = [\phi_1^i(x) \;\; \phi_2^i(x) \;\; \cdots \;\; \phi_L^i(x)]^T$ is the vector activation function, and $W_v^i = [w_1^i \;\; w_2^i \;\; \cdots \;\; w_L^i]^T$ is the vector weight.

$u_1^{i+1}(x)$ can be approximated by

$$u_1^{i+1}(x) = (W_{u1}^{i+1})^T \Theta^{i+1}(x), \tag{18}$$

where, $\Theta^{i+1}(x) = [\theta_1^{i+1}(x) \;\; \theta_2^{i+1}(x) \;\; \cdots \;\; \theta_M^{i+1}(x)]^T$ is the vector activation function, $W_{u1}^{i+1} = [w_1^{i+1}(x) \;\; w_2^{i+1}(x) \;\; \cdots \;\; w_M^{i+1}(x)]^T$ is the vector weight, M is the the number of hidden-layer neurons of the neural networks.

$u_2^{i+1}(x)$ can be approximated by

$$u_2^{i+1}(x) = (W_{u2}^{i+1})^T \Psi^{i+1}(x), \tag{19}$$

where, $\Psi^{i+1}(x) = [\psi_1^{i+1}(x) \;\; \psi_2^{i+1}(x) \;\; \cdots \;\; \psi_N^{i+1}(x)]^T$ is the vector activation function, $W_{u2}^{i+1} = [w_1^{i+1}(x) \;\; w_2^{i+1}(x) \;\; \cdots \;\; w_N^{i+1}(x)]^T$ is the vector weight, N is the the number of hidden-layer neurons of the neural networks.

Substituting (17), (18), (19) into (12), we have

$$\begin{aligned}(W_v^i)^T\Phi_v^i(x(t+\Delta t)) - (W_v^i)^T\Phi_v^i(x(t)) = \\ -\int_t^{t+\Delta t} x^TQxdt - \int_t^{t+\Delta t}(u_1^i)^TRu_1^idt \\ +\int_t^{t+\Delta t}\gamma^2(u_2^i)^Tu_2^idt \\ -2\int_t^{t+\Delta t}(R(W_{u1}^{i+1})^T\Theta^{i+1}(x))^Tw_1^i(t)dt \\ +2\gamma^2\int_t^{t+\Delta t}((W_{u2}^{i+1})^T\Psi^{i+1}(x))^Tw_2^i(t)dt. \quad (20)\end{aligned}$$

From (20), it is shown that W_v^i, W_{u1}^{i+1} and W_{u2}^{i+1} are the unknown parameters. Note that W_v^i contains L unknown parameters, W_{u1}^{i+1} contains M unknown parameters, and W_{u2}^{i+1} contains N unknown parameters, but there is just one dimensional equation (20) provided for such calculations. Thus, we can use the the least squares method and the Kronecker product theory [28] for solving the equation (20) to obtain the unknown parameters (W_v^i, W_{u1}^{i+1}, W_{u2}^{i+1}). Further, for any positive integer $l > 0$, according to (20), we have

$$\delta_v = \begin{bmatrix} \Phi_v^i(x(t_1)) - \Phi_v^i(x(t_0)) \\ \Phi_v^i(x(t_2)) - \Phi_v^i(x(t_1)) \\ \vdots \\ \Phi_v^i(x(t_l)) - \Phi_v^i(x(t_{l-1})) \end{bmatrix}, \quad (21)$$

$$I_{xx} = \begin{bmatrix} \int_{t_0}^{t_1} x^TQxdt \\ \int_{t_1}^{t_2} x^TQxdt \\ \vdots \\ \int_{t_{l-1}}^{t_l} x^TQxdt \end{bmatrix}, \quad (22)$$

$$I_{\theta\theta} = \begin{bmatrix} \int_{t_0}^{t_1} \Theta(x)\otimes\Theta(x)dt \\ \int_{t_1}^{t_2} \Theta(x)\otimes\Theta(x)dt \\ \vdots \\ \int_{t_{l-1}}^{t_l} \Theta(x)\otimes\Theta(x)dt \end{bmatrix}, \quad (23)$$

$$I_{\varphi\varphi} = \begin{bmatrix} \int_{t_0}^{t_1} \Psi(x)\otimes\Psi(x)dt \\ \int_{t_1}^{t_2} \Psi(x)\otimes\Psi(x)dt \\ \vdots \\ \int_{t_{l-1}}^{t_l} \Psi(x)\otimes\Psi(x)dt \end{bmatrix}, \quad (24)$$

$$I_{\theta u_1} = \begin{bmatrix} \int_{t_0}^{t_1} \Theta(x)u_1(x)dt \\ \int_{t_1}^{t_2} \Theta(x)u_1(x)dt \\ \vdots \\ \int_{t_{l-1}}^{t_l} \Theta(x)u_1(x)dt \end{bmatrix}, \quad (25)$$

$$I_{\varphi u_2} = \begin{bmatrix} \int_{t_0}^{t_1} \Psi(x)u_2(x)dt \\ \int_{t_1}^{t_2} \Psi(x)u_2(x)dt \\ \vdots \\ \int_{t_{l-1}}^{t_l} \Psi(x)u_2(x)dt \end{bmatrix}. \quad (26)$$

Therefore, combining (21)-(26) with (20), we have

$$\Xi^i \begin{bmatrix} W_v^i \\ W_{u_1}^{i+1} \\ W_{u_2}^{i+1} \end{bmatrix} = \Upsilon^i, \quad (27)$$

where, $\Xi^i = [\Xi_1^i,\ \Xi_2^i,\ \Xi_3^i]$, $\Xi_1^i = \delta_v$, $\Xi_2^i = 2RI_{\theta u_1} - 2RI_{\theta\theta}(W_{u_1}^i \otimes I_M)$, $\Xi_3^i = -2\gamma^2 I_{\varphi u_2} + 2\gamma^2 I_{\varphi\varphi}(W_{u_2}^i \otimes I_N$, $\Upsilon^i = -I_{xx} - RI_{\theta\theta}(W_{u_1}^i \otimes W_{u_1}^i) + \gamma^2 I_{\varphi\varphi}(W_{u_2}^i \otimes W_{u_2}^i)$.

Thus, the unknown parameters (W_v^i, W_{u1}^{i+1}, W_{u2}^{i+1}) can be solved in the least-squares sense as follows:

$$\begin{bmatrix} W_v^i \\ W_{u_1}^{i+1} \\ W_{u_2}^{i+1} \end{bmatrix} = ((\Xi^i)^T\Xi^i)^{-1}(\Xi^i)^T\Upsilon^i. \quad (28)$$

Note that $l > L + M + N$ is the necessary condition for the excitation condition to ensure that the matrix $(\Xi^i)^T\Xi^i$ is invertible. Until now, based on the equation (28), we will present a model-free adaptive dynamic programming algorithm for online solution of the two-player zero-sum differential game of the nonlinear continuous-time system with the fully unknown knowledge of the system dynamics.

Algorithm 2: Model-free adaptive dynamic programming algorithm for the two-player zero-sum differential game

Step 1. Give the initial stabilizing control policies $u_1 = u_1^0 + e_1$ and $u_2 = u_2^0 + e_2$, e_1 and e_2 are the exploration noises. Set $i = 0$.

Step 2. Solve the unknown parameters (W_v^i, W_{u1}^{i+1}, W_{u2}^{i+1}) from (28).

Step 3. Set $i = i+1$, and repeat Step 2 until $\|W_v^i - W_v^i\| \le \varepsilon$ for $i \ge 1$, where the constant ε is a predefined small threshold.

Step 4. Use $u_1^{i+1}(x) = (W_{u1}^{i+1})^T\Theta^{i+1}(x)$ and $u_2^{i+1}(x) = (W_{u2}^{i+1})^T\Psi^{i+1}(x)$ as the closed-loop Nash equilibrium optimal strategies.

IV. Simulation results

In this section, a simulation example is carried out to demonstrate the feasibility of the model-free adaptive dynamic programming algorithm for the two-player zero-sum differential game. Consider the continuous-time nonlinear system as like (1), where,

$$f(x) = \begin{bmatrix} -0.25x_1 \\ 0.5x_1^2x_2 - 0.125x_2^3 - 0.5x_2 \end{bmatrix},$$

$$g_1(x) = \begin{bmatrix} 0 \\ x_1 \end{bmatrix} \quad g_1(x) = \begin{bmatrix} 0 \\ x_2 \end{bmatrix}.$$

One selects $Q = [1\ \ 0;\ \ 0\ \ 1]$, $R = 1$, and $\gamma = 2$. By using the converse HJB approach [29], we can obtain that the optimal value function $V^*(x)$ is

$$V^*(x) = 2x_1^2 + x_2^2, \quad (29)$$

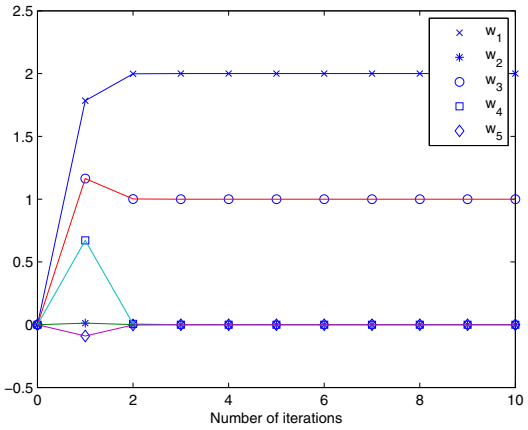

Fig. 1. The convergence curve of W_{u1}^{i+1} to its true value W_{u1}^{*}.

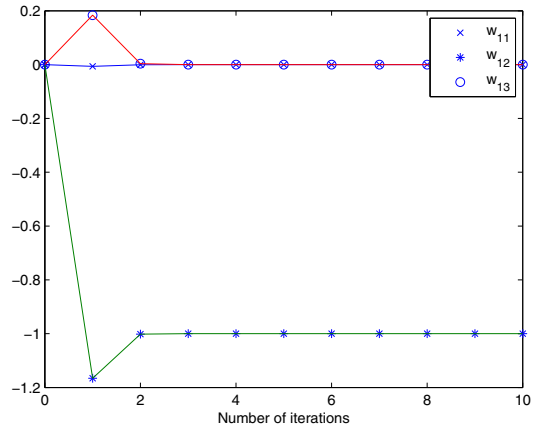

Fig. 2. The convergence curve of W_{u1}^{i+1} to its true value W_{u1}^{*}.

the control policies are

$$u_1^*(x) = -x_1 x_2, \tag{30}$$

and

$$u_2^*(x) = 0.25 x_2^2. \tag{31}$$

When the knowledge on the drift dynamics of the system is completely unknown, we can use the Algorithm 2 for finding the online solution of the zero-sum differential game of the continuous-time nonlinear system. The selection of parameters in Algorithm 2 are given as follows. The predefined small threshold is set as $\epsilon = 10^{-7}$. The learning time is selected as $\Delta t = 2s$. Set $l = 200$. The vector activation function $\Phi_v(x)$ is selected as $\Phi_v^i(x) = [x_1^2 \;\; x_1 x_2 \;\; x_2^2 \;\; x_1^4 \;\; x_2^4]^T$. The vector activation function $\Theta^{i+1}(x)$ and $\Psi^{i+1}(x)$ are selected as the gradient of the vector activation function $\Phi_v(x)$. The vector weight W_v^i is set as $W_v^i = [w_1 \;\; w_2 \;\; w_3 \;\; w_4 \;\; w_5]^T$, the vector weight W_{u1}^{i+1} is set as $W_{u1}^{i+1} = [w_{11} \;\; w_{12} \;\; w_{13}]^T$, the vector weight W_{u2}^{i+1} is set as $W_{u2}^{i+1} = [w_{21} \;\; w_{22} \;\; w_{23}]^T$.

Along the state trajectories from $t = 0s$ to $t = 2s$, the state and input information is collected. Then, the Algorithm 2 is run at $t = 2s$. Fig. 1 shows that the weight W_v^i converges to the true value W_v^* after 10 iteration, i.e. $W_v^i = [2 \;\; 0 \;\; 1 \;\; 0 \;\; 0]^T$. Fig. 2 shows that the weight W_{u1}^{i+1} converges to the true value W_{u1}^* after 10 iteration, i.e. $W_{u1}^{i+1} = [0 \;\; -1 \;\; 0]^T$. Fig. 3 shows that the weight W_{u2}^{i+1} converges to the true value W_{u2}^* after 10 iteration, i.e. $W_{u2}^{i+1} = [0 \;\; 0.25 \;\; 0]^T$. As a result, the simulation results demonstrate that the approximate optimal Nash equilibrium solution of the two-player zero-sum game of the nonlinear continuous-time system with the unknown system dynamics can be obtained by using the proposed algorithm in this paper.

V. Conclusion

In this paper, a new model-free adaptive dynamic programming algorithm has been presented to solve the two-player zero-sum differential game problem of the continuous-time nonlinear system with completely unknown system dynamics. The importance features of the proposed algorithm is that the proposed algorithm can solve online the optimal solution of the Hamilton-Jacobi-Isaacs equation by using data generated in real time along the state trajectories of the continuous-time nonlinear system, in which the knowledge of system dynamics is not required. Finally, simulation studies have demonstrated the effectiveness of the proposed algorithm. Our future work will extend the results to the multi-player nonzero-sum differential game problem of the general continuous-time nonlinear systems.

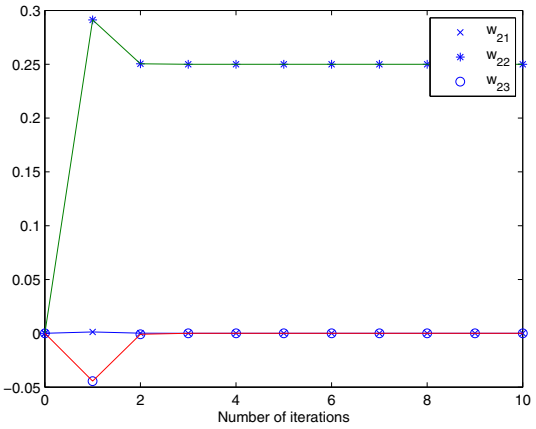

Fig. 3. The convergence curve of W_{u2}^{i+1} to its true value W_{u2}^{*} .

Acknowledgment

This work was supported by the National Natural Science Foundation of China (61034005, 61104010) and the National High Technology Research and Development Program of China (2012AA040104).

REFERENCES

[1] T. Basar and G. J. Olsder, *Dynamic noncooperative game, Second Edition,* Boston, 1997.

[2] P. Zhang, "Some results on two-person zero-sum linear quadratic differential games," *SIAM Journal on Control and Optimization,* vol. 43, pp. 2157–2165, 2005.

[3] H. S. Chang and S. I. Marcus, "Two-person zero-sum markov games: receding horizon approach," *IEEE Transactions on Automatic Control,* vol. 48, pp. 1951–1961, 2003.

[4] A. Al-Tamimi, M. Abu-Khalaf and F. L. Lewis, "Adaptive critic designs for discrete-time zero-sum games with application to H-infinity control," *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics,* vol. 37, pp. 240–247, 2007.

[5] A. Al-Tamimi, F. L. Lewis and M. Abu-Khalaf, "Model-free Q-learning designs for linear discrete-time zero-sum games with application to H-infinity control," *Automatica,* vol. 43, pp. 473–481, 2007.

[6] Q. L. Wei, H. G. Zhang and L. L. Cui. "Data-based optimal control for discrete-time zero-sum games of 2-D systems using adaptive critic designs," *ACTA Automatic Sinica,* vol. 35, pp. 682–692, 2009.

[7] X. Zhang, H. G. Zhang and X. Y. Wang, "A new iteration approach to solve a class of finite-horizon continuous-time nonaffine nonlinear zero-sum game," *International Journal of Innovative Computing, Information Control,* vol. 7, pp. 597–608, 2011.

[8] P. Werbos, "Neural networks for control and system identification," *In Proceedings of IEEE Conference on Decision and Control,* pp. 260–265, 1989.

[9] P. Werbos, "A menu of designs for reinforcement learning over time," *Neural Networks for Control,* MIT Press, 1991.

[10] F. Wang, H. Zhang and D. Liu, "Adaptive dynamic programming: an introduction," *IEEE Computational Intelligence Magazine,* vol. 43, pp. 9–47, 2009.

[11] F. Lewis, D. Vrabie and K. Vamvoudakis, "Reinforcement learning and feedback control using natural decision methods to design optimal adaptive controllers," *IEEE Systems Magazine,* vol. 32, pp. 76–105, 2012.

[12] H. Zhang, D. Liu, Y. Luo and D. Wang, *Adaptive dynamic programming for control: algorithms and stability,* Springer-Verlag, London, 2013.

[13] D. Vrabie, K. Vamvoudakis and F. L. Lewis, *Optimal adaptive control and differential games by reinforcement learning principles,* The Institution of Engineering and Technology, London, United Kingdom, 2013.

[14] D. Liu and Q. Wei, "Finite-approximation-error-based optimal control approach for discrete-time nonlinear systems," *IEEE Transactions on Cybernetics,* vol. 43, pp. 779–789, 2013.

[15] W. Powell, *Approximate dynamic programming: Solving the curses of dimensionality, 2nd edition*, John Wiley & Sons, 2011.

[16] S. Mehraeen, T. Dierks, S. Jagannathan, and Mariesa Crow, "Zero-sum two-player game theoretic formulation of affine nonlinear discrete-time systems using neural networks", *IEEE Transactions on Systems, Man and Cybernetics,* vol. 43, pp. 1641-1655, 2013.

[17] H. Xu and S. Jagannathan "Model-free H-infinite stochastic optimal design for unknown linear networked control system zero-sum games via Q-learning", *Proc. of the IEEE Symposium on Intelligent Control,* pp. 198-203, Sept. 2011.

[18] S. Mehraeen, T. Dierks, S. Jagannathan, and M. L. Crow, "Zero-sum two-player game theoretic formulation of affine nonlinear discrete-time systems using neural networks", *Proc. of the IEEE International Joint Conference on Neural Networks,* pp. 1-8, July 2010.

[19] M. Abu-Khalaf, F. L. Lewis and J. Huang, "Policy iterations on the Hamilton-Jacobi-Isaacs equation for H-infinity state feedback control with input saturation," *IEEE Transactions on Automatic Control,* vol. 51, pp. 1989–1995, 2006.

[20] M. Abu-Khalaf, F. L. Lewis and J. Huang, "Neurodynamic programming and zero-sum games for constrained control systems," *IEEE Transactions on Neural Networks,* vol. 19, pp. 1243–1252, 2008.

[21] Q. L. Wei, H. G. Zhang and D. R. Liu, "A new approach to solve a class of continuous-time non-linear quadratic zero-sum game using adp," *IEEE International Conference on Networking, Sensing and Control Sanya, China,* pp. 507–512, 2008.

[22] H. G. Zhang, Q. L. Wei and D. R. Liu, "An iterative adaptive dynamic programming method for solving a class of nonlinear zero-sum differential games," *Automatica,* vol. 47, pp. 207–214, 2011.

[23] B. Luo and H. N. Wu, "Computationally efficient simultaneous policy update algorithm for nonlinear H_∞ state feedback control with Galerkins method," *International Journal of Robust and Nonlinear Control,* vol. 23, pp. 991–1012, 2013.

[24] Kyriakos G. Vamvoudakis and F. L. Lewis, "Online solution of nonlinear two-player zero-sum games using synchronous policy iteration," *International Journal of Robust and Nonlinear Control,* vol. 22, pp. 1460–1483, 2012.

[25] H. N. Wu and B. Luo, "Neural network based online simultaneous policy update algorithm for solving the hji equation in nonlinear h_∞ control," *IEEE Transactions on Neural Networks and Learning Systems,* vol. 23, pp. 1884–1895, 2012.

[26] Y. Jiang ang Z. P. Jiang, "Robust adaptive dynamic programming for nonlinear control design," *IEEE 51st Annual Conference on Decision and Control, China,* pp. 1896–1901, 2012.

[27] M. Abu-Khalaf and F. L. Lewis, "Nearly optimal control laws for nonlinear systems withsaturating actuators using a neural network HJB approach," *Automatica,* vol. 41, pp. 779–791, 2005.

[28] J. Brewer, "Kronecker products and matrix calulus in system theory," *IEEE transactions on Circuits and Systems,* vol. 25, pp. 772–781, 1978.

[29] V. Nevisti'C and J. A. Primbs, "Constrained nonlinear optimal control: A converse HJB approach," Dept. Control & Dynamical Syst., California Inst. Technology, Pasadena, Tech. Rep. TR96–021, 1996.