2011 50th IEEE Conference on Decision and Control and
European Control Conference (CDC-ECC)
Orlando, FL, USA, December 12-15, 2011

Nonlinear Two-Player Zero-Sum Game Approximate Solution Using a Policy Iteration Algorithm

M. Johnson, S. Bhasin, and W. E. Dixon

***Abstract*—An approximate online solution is developed for a two-player zero-sum game subject to continuous-time nonlinear uncertain dynamics and an infinite horizon quadratic cost. A novel actor-critic-identifier (ACI) structure is used to implement the Policy Iteration (PI) algorithm, wherein a robust dynamic neural network (DNN) is used to asymptotically identify the uncertain system, and a critic NN is used to approximate the value function. The weight update laws for the critic NN are generated using a gradient-descent method based on a modified temporal difference error, which is independent of the system dynamics. This method finds approximations of the optimal value function, and the saddle point feedback control policies. These policies are computed using the critic NN and the identifier DNN and guarantee uniformly ultimately bounded (UUB) stability of the closed-loop system. The actor, critic and identifier structures are implemented in real-time, continuously and simultaneously.**

I. Introduction

Noncooperative game theory [1]–[3] can be used to provide a solution to a number of control engineering applications. In a differential game formulation, the controlled system is influenced by a number of different inputs, computed by different players that are individually trying to optimize a performance function. The control objective is to determine a set of policies that minimize individual performance functions to yield a Nash equilibrium , and are admissible [4], i.e. control policies that guarantee the stability of the dynamic system. The Nash solution is characterized by an equilibria, in which each player has an outcome that cannot be improved by a unilateral change of strategy. A Nash equilibrium formulation that has received heavy interest in control theory is the two-player min-max optimization H_∞ control problem [5], where the controller is a minimizing player and the disturbance is a maximizing player in a zero-sum game. In a zero-sum game with linear dynamics and an infinite horizon quadratic cost function, the Nash equilibrium solution is equivalent to solving the generalized game algebraic Riccati equation (GARE). However for nonlinear dynamics, developing an analytical solution is even further complicated by the sufficient condition of solving a Hamilton-Jacobi-Isaacs (HJI) partial differential equation; where a solution may not exist.

Given the difficulty in solving the HJI equation analytically, an alternative approach is to find an approximate solution to the HJI. Previous research on reinforcement learning (RL) using adaptive critics (AC) in the machine learning community [6]–[10] has provided inroads to determining approximate solutions of optimal control problems using Approximate Dynamic Programming (ADP) methods [11]–[15]. The discrete/iterative nature of the ADP formulation lends itself naturally to the design of discrete-time optimal controllers [14], [16]–[20]. Baird [21] proposed Advantage Updating, an extension of the Q-learning algorithm which could be implemented in continuous-time and provided fast convergence. A Hamiltonian Jacobi Bellman (HJB)-based framework is used in [22] and [23], and Galerkin's spectral method is used to approximate the generalized HJB solution in [24]. All of the aforementioned approaches for continuous-time nonlinear systems required complete knowledge of the dynamics. A contribution in [25] is the requirement of only partial knowledge of the system in the design of the controller for policy iteration (PI). Vamvoudakis and Lewis [26] extended the idea by designing a model-based online algorithm called synchronous PI which involved synchronous continuous-time adaptation of both actor and critic neural networks. The synchronous PI method was then further generalized to solve the two-player zero-sum game problem for nonlinear continuous-time systems with known dynamics in [27] . Bhasin et. al [28] developed an actor-critic-identifier (ACI) which uses a robust dynamic neural network (DNN) to identify the dynamics and a critic NN to approximate the value function, thereby removing the requirement of complete knowledge of the dynamics.

This paper generalizes the method given in [28] to solve a two-player zero-sum infinite horizon game subject to continuous-time unknown nonlinear dynamics. The novel ACI architecture implements the PI algorithm online which yields controller policies that converges to the solution of the two-player differential game. A DNN-based robust system identifier is used to identify the nonlinear plant. The policy evaluation process involves value function approximation which can be achieved by tuning the weights of the critic

This material is based upon work supported by the National Aeronautics and Space Administration through the University of Central Florida's Space Grant Consortium and the National Science Foundation ECCS Project. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Aeronautics and Space Administration or the National Science Foundation.

M. Johnson, S. Bhasin, and W. E. Dixon are with the Dept. of Mechanical and Aerospace Engineering, University of Florida, Gainesville, Florida 32611,{marc1518,sbhasin,wdixon}@ufl.edu.

NN using a temporal difference (TD) error [29] that does not depend on complete model knowledge. The critic NN is used to construct the actor, which is an approximate control law that stabilizes the closed-loop system. The proposed ACI architecture is a robust implementation of the PI algorithm which is shown to approach the solution of the two-player zero-sum game, and guarantee UUB stability in the sense of Lyapunov.

II. Two Player Zero-Sum Differential Game

Consider the nonlinear time-invariant control affine dynamic system given by

$$\dot{x} = f(x) + g(x)\,u(x) + k(x)\,d(x), \tag{1}$$

where $x(t) \in \mathbb{R}^n$ is the state vector, $u(x), d(x) \in \mathbb{R}^m$ are the control inputs, and $f(x) \in \mathbb{R}^n$, $g(x) \in \mathbb{R}^{n\times m}$ and $k(x) \in \mathbb{R}^{n\times p}$ are the drift, and the two input matrices, respectively. Assume that $f(x)$ and $g(x)$ are Lipschitz continuous and that $f(0) = 0$ so that $x = 0$ is an equilibrium point for (1). The performance index is given as [30]

$$J(x,u,d) = \int_0^\infty \left(Q(x) + u^T R u - \gamma^2 d^T d\right) dt,$$

where $Q(x) \in \mathbb{R}$, $R = R^T \in \mathbb{R}^{m\times m}$ are positive definite, and $\gamma \geq \gamma^* > 0$, where γ^* is the smallest γ for which the system is stabilized [31]. For the two player zero-sum differential game, the infinite-horizon scalar value or cost functional $V^u(x(t), u, d)$ associated with the control policies $\{u = u(x(s)); s \geq t\}$ and $\{d = d(x(s)); s \geq t\}$ can be defined as

$$V^u(x) = \min_u \max_d \int_t^\infty r(x(s), u(s), d(s))\, ds, \tag{2}$$

where t is the initial time, and $r(x,u,d) \in \mathbb{R}$ is the local cost for the state, and controllers, defined as

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

In this differential game, $u(x)$ is the minimizing player and $d(x)$ is the maximizing player. This two player optimal control problem has a unique solution if the Nash condition holds

$$\min_u \max_d J(x(0), u, d) = \max_d \min_u J(x(0), u, d).$$

The objective of the optimal control problem is to find admissible feedback policies [5] ($u^* = u(x)$ and $d^* = d(x)$), such that the cost in (2) associated with the system in (1) is minimized [32]. Assuming the value functional is continuously differentiable, Bellman's principle of optimality can be used to derive the following optimality condition

$$\begin{aligned} 0 &= \min_u \max_d \left[\frac{\partial V^*(x)}{\partial x}^T \left(f(x) + g(x)\,u + k(x)\,d\right) \right. \\ &\quad \left. + r(x,u,d)\right], \end{aligned} \tag{4}$$

which is a nonlinear PDE, also called the Hamilton-Jacobi-Isaacs (HJI) equation. A solution $V^*(x) \geq 0$ to (4) is the value (2) for the given feedback policies $u(x)$ and $d(x)$. Using the local cost given in (3) a closed form expression of the optimal controllers can be determined from (4) as

$$u^* = -\frac{1}{2} R^{-1} g^T(x) \frac{\partial V^*(x)}{\partial x} \tag{5}$$

$$d^* = \frac{1}{2\gamma^2} k^T(x) \frac{\partial V^*(x)}{\partial x}. \tag{6}$$

The closed form expression for the optimal control policies in (5) and (6), obviates the need to search for a feedback policy that minimizes the value function; however, the solution $V^*(x)$ to the HJI equation given in (4) is required. The HJI equation in (4), can be rewritten by substituting for the local cost in (3) and the optimal control policies in (5) and (6), as

$$\begin{aligned} 0 &= Q(x) + \left(\frac{\partial V^*}{\partial x}\right)^T f(x) \\ &\quad - \frac{1}{4}\left(\frac{\partial V^*}{\partial x}\right)^T g(x) R^{-1} g^T(x) \frac{\partial V^*}{\partial x} \\ &\quad + \frac{1}{4\gamma^2}\left(\frac{\partial V^*}{\partial x}\right)^T k(x) k^T(x) \frac{\partial V^*}{\partial x} \qquad V^*(0) = 0. \end{aligned} \tag{7}$$

Since the HJI equation is troublesome to solve in general, an approximate solution is sought.

III. Policy Iteration Algorithm

For a nonlinear system a policy iteration algorithm can provide an approximate solution to the HJI equation. A contribution of this paper is the use of the ACI architecture [28], which eliminates the need for complete model knowledge, to find the solution to the two player zero-sum game. Specifically, a DNN is used to robustly identify the system, a critic NN approximates the value function, and an actor is used to determine the optimal control policies, which minimize the value function. The PI algorithm involves: choosing admissible control policies estimating the value function with respect to those policies, and then improving the current policies based on information from the value function estimate. Typically, PI would be implemented in a two-step fashion, involving only policy evaluation and policy improvement; however, an additional system identification step is used to relax the requirement for complete model knowledge. The proposed PI algorithm is suited for online implementation, where the system identification, policy evaluation, and policy improvement steps are updated simultaneously.

IV. DNN-Based System Identification

Consider the system in (1), with additive unknown-unstructured disturbances

$$\dot{x} = f(x) + g(x)\,u(x) + k(x)\,d(x); \quad x(0) = x_0, \tag{8}$$

where the state $x(t)$ is assumed to be measurable. The following assumptions about the system will be utilized in the subsequent development.

Assumption 1: For any bounded admissible controls $u, d \in U$ with $||u|| \leq \bar{u}$ and $\|d\| \leq \bar{d}$, and any finite initial

condition x_0, the state trajectories are uniformly bounded for any $T > 0$, i.e. $x(T) \in \mathcal{L}_\infty$ [33].

Assumption 2: Given a continuous function $h : \mathbb{S} \to \mathbb{R}^n$, where $\mathbb{S}$ is a compact simply connected set, there exists ideal weights W, such that the output of the NN, $\hat{h}(x, W)$, approximates $h(x)$ to an arbitrary accuracy [34].

Using Assumption 2, the unknown nonlinear system in (8) can be represented as

$$\begin{aligned} \dot{x} &= A_s x + W_f^T \sigma_f(x) + \varepsilon_f + \left(W_g^T \sigma_g(x) + \varepsilon_g\right) u \\ &\quad + \left(W_k^T \sigma_k(x) + \varepsilon_k\right) d, \end{aligned} \tag{9}$$

where $A_s \in \mathbb{R}^{n \times n}$ is a Hurwitz matrix, the functions $f(x) - A_s x$, $g(x)$, and $k(x)$ are approximated by NNs as

$$f(x) - A_s x = W_f^T \sigma_f(x) + \varepsilon_f(x) \tag{10}$$

$$g(x) = W_g^T \sigma_g(x) + \varepsilon_g(x) \tag{11}$$

$$k(x) = W_k^T \sigma_k(x) + \varepsilon_k(x), \tag{12}$$

where $W_f \in \mathbb{R}^{N_f \times n}$, $W_g \in \mathbb{R}^{N_g \times n}$, and $W_k \in \mathbb{R}^{N_k \times n}$ are the constant bounded ideal weight matrices of the three NN with N_f, N_g, and N_k representing the neurons in the output layers, respectively. The activation functions are given by $\sigma_f(\cdot) \in \mathbb{R}^{N_f}$, $\sigma_g(\cdot) \in \mathbb{R}^{N_g}$, and $\sigma_k(\cdot) \in \mathbb{R}^{N_k}$, while $\varepsilon_f(\cdot) \in \mathbb{R}^n$, $\varepsilon_g(\cdot) \in \mathbb{R}^{n \times m}$, and $\varepsilon_k(\cdot) \in \mathbb{R}^p$ are the function reconstruction errors in approximating the functions $f(x)$, $g(x)$, and $k(x)$, respectively.

Assumption 3: The activation functions $\sigma_f(\cdot)$, $\sigma_g(\cdot)$, $\sigma_k(\cdot)$, and $\phi(\cdot)$, and their time derivatives with respect to their arguments are bounded.

Assumption 4: The ideal NN weights are bounded by a positive known constant [35] i.e. $||W_f|| \leq \bar{W}_f$, $||W_g|| \leq \bar{W}_g$, $||W_k|| \leq \bar{W}_k$, and $||W_v|| \leq \bar{W}_v$.

Assumption 5: The NN function reconstruction errors are bounded [35], as $||\varepsilon_f|| \leq \bar{\varepsilon}_f$, $||\varepsilon_g|| \leq \bar{\varepsilon}_g$, and $||\varepsilon_k|| \leq \bar{\varepsilon}_k$. For ease in deriving weight update laws, single-layer linear-in-the-parameter(LIP) NNs are used in (10), (11), and (12). The universal approximation property does not generally hold for LIP NNs, however, if the activation functions are chosen as a basis, the approximation property still holds [36].

The proposed DNN used to identify the system in (8) is

$$\begin{aligned} \dot{\hat{x}} &= A_s \hat{x} + \hat{W}_f^T \sigma_f(\hat{x}) + \hat{W}_g^T \sigma_g(\hat{x}) u \\ &\quad + \hat{W}_k^T \sigma_k(\hat{x}) d + \beta sgn(\tilde{x}), \end{aligned} \tag{13}$$

where $\hat{x}(t) \in \mathbb{R}^n$ is the state of the DNN, $\hat{W}_f \in \mathbb{R}^{N_f \times n}$, $\hat{W}_g \in \mathbb{R}^{N_g \times n}$, and $\hat{W}_k \in \mathbb{R}^{N_k \times n}$ are the estimates of the ideal weights of the NNs, and $\beta \in \mathbb{R}$ is a constant positive control gain. The measurable identification error $\tilde{x}(t) \in \mathbb{R}^n$ is defined as

$$\tilde{x} \triangleq x - \hat{x}. \tag{14}$$

Due to the NN reconstruction errors $\varepsilon_f(\cdot)$, $\varepsilon_g(\cdot)$, and $\varepsilon_k(\cdot)$, the classical Hopfield DNN structure [33], [37], [38] is modified by the addition a robust sliding mode term in (13). As proven in the subsequent stability analysis, the sliding mode term is used to guarantee asymptotic identification of the plant and robustly identify the disturbances in the system. The proposed structure in (13) is motivated by the desire to prove that with a suitable choice of weight update laws, the identification error converges to zero; thereby demonstrating that the input-output behavior of the DNN model approximates the input-output behavior of the plant. The identification error dynamics are developed by taking the time derivative of (14) and substituting for (9) and (13) as

$$\begin{aligned} \dot{\tilde{x}} &= A_s \tilde{x} - \hat{W}_f^T \sigma_f(\hat{x}) + W_f^T \sigma_f(x) + W_g^T \sigma_g(x) u \\ &\quad - \hat{W}_g^T \sigma_g(\hat{x}) u + W_k^T \sigma_k(x) d - \hat{W}_k^T \sigma_k(\hat{x}) d \\ &\quad + \varepsilon_f + \varepsilon_g u + \varepsilon_k d - \beta sgn(\tilde{x}). \end{aligned}$$

Adding and subtracting $W_f^T \sigma_f(\hat{x})$, $W_g^T \sigma_g(\hat{x}) u$, and $W_k^T \sigma_k(\hat{x}) d$, and grouping terms yields

$$\begin{aligned} \dot{\tilde{x}} &= A_s \tilde{x} + \tilde{W}_f^T \sigma_f(\hat{x}) + \tilde{W}_g^T \sigma_g(\hat{x}) u \\ &\quad + \tilde{W}_k^T \sigma_k(\hat{x}) d + h - \beta sgn(\tilde{x}), \end{aligned} \tag{15}$$

where $\tilde{W}_f = W_f - \hat{W}_f \in \mathbb{R}^{N_f \times n}$, $\tilde{W}_g = W_g - \hat{W}_g \in \mathbb{R}^{N_g \times n}$, and $\tilde{W}_k = W_k - \hat{W}_k \in \mathbb{R}^{N_k \times n}$ are the estimate mismatch of the DNN ideal weights, and the auxiliary signal $h(t) \in \mathbb{R}^n$ is given by

$$\begin{aligned} h &= W_f^T \left(\sigma_f(x) - \sigma_f(\hat{x})\right) + W_g^T \left(\sigma_g(x) - \sigma_g(\hat{x})\right) u \\ &\quad + W_k^T \left(\sigma_k(x) - \sigma_k(\hat{x})\right) d + \varepsilon_f + \varepsilon_g u + \varepsilon_k d. \end{aligned}$$

Using Assumptions 1-5, it is clear that $h(t)$ is bounded as follows

$$||h|| \leq \bar{h}, \tag{16}$$

where $\bar{h} \in \mathbb{R}$ is a known positive constant. Based on the subsequent stability analysis, the update law for the DNN can be designed as

$$\dot{\hat{W}}_f = \Gamma_f proj\left(\sigma_f(\hat{x}) \tilde{x}^T\right) \tag{17}$$

$$\dot{\hat{W}}_g = \Gamma_g proj\left(\sigma_g(\hat{x}) u \tilde{x}^T\right) \tag{18}$$

$$\dot{\hat{W}}_k = \Gamma_k proj\left(\sigma_k(\hat{x}) d \tilde{x}^T\right). \tag{19}$$

Theorem 1: The DNN-based robust identifier in (13), along with the NN weight update laws in (17)-(19) respectively, guarantees asymptotic identification, in the sense that

$$\lim_{t \to \infty} ||\tilde{x}(t)|| = 0, \tag{20}$$

provided the following sufficient gain condition is satisfied

$$\beta > \bar{h}, \tag{21}$$

where β and $\bar{h}$ were introduced in (13) and (16), respectively.

Proof: Consider a positive definite, continuously differentiable function $V(t)$ defined as

$$\begin{aligned} V &= \frac{1}{2} \tilde{x}^T \tilde{x} + \frac{1}{2} tr\left(\tilde{W}_f^T \Gamma_f^{-1} \tilde{W}_f\right) + \frac{1}{2} tr\left(\tilde{W}_g^T \Gamma_g^{-1} \tilde{W}_g\right) \\ &\quad + \frac{1}{2} tr\left(\tilde{W}_k^T \Gamma_k^{-1} \tilde{W}_k\right). \end{aligned} \tag{22}$$

Taking the time derivative of (22), and substituting the dynamics from (15), and update laws in (17)-(19) yields

$$\dot{V} = \tilde{x}^T \left(A_s \tilde{x} + \tilde{W}_f^T \sigma_f(\hat{x}) + \tilde{W}_g^T \sigma_g(\hat{x}) u \right. \quad (23)$$
$$+ \tilde{W}_k^T \sigma_k(\hat{x}) d + h - \beta sgn(\tilde{x}) - tr\left(\tilde{W}_f^T \sigma_f(\hat{x}) \tilde{x}^T \right)$$
$$\left. - tr\left(\tilde{W}_g^T \sigma_g(\hat{x}) u \tilde{x}^T \right) - tr\left(\tilde{W}_k^T \sigma_k(\hat{x}) d \tilde{x}^T \right) \right).$$

Simplifying and using (16), the expression in (23) can be upper-bounded as

$$\dot{V} \leq \tilde{x}^T A_s \tilde{x} + \bar{h} ||\tilde{x}|| - \beta \sum_{i=1}^{n} |\tilde{x}_i|.$$

Using the fact that $\sum_{i=1}^{n} |\tilde{x}_i| \geq ||\tilde{x}||$, the following inequality is obtained

$$\dot{V} \leq \tilde{x}^T A_s \tilde{x} + \left(\bar{h} - \beta \right) ||\tilde{x}||. \quad (24)$$

Provided the sufficient gain condition in (21) is satisfied, the expression in (24) can be written as

$$\dot{V} \leq \tilde{x}^T A_s \tilde{x}, \quad (25)$$

which proves that $V(t) \in \mathcal{L}_\infty$. Hence, $\tilde{x}(t)$,$\tilde{W}_f(t)$,$\tilde{W}_g(t)$, $\tilde{W}_k(t) \in \mathcal{L}_\infty$. Since A_s is Hurwitz, (25) indicates that $\tilde{x}(t) \in \mathcal{L}_2$. Furthermore, since $u(t)$, $d(t) \in \mathcal{L}_\infty$, from Assumption 1, $h(t) \in \mathcal{L}_\infty$ from (16), and $\tilde{x}(t)$,$\tilde{W}_f(t)$,$\tilde{W}_g(t)$, $\tilde{W}_k(t) \in \mathcal{L}_\infty$, it can be shown that $\dot{\tilde{x}} \in \mathcal{L}_\infty$ from (15); hence $\tilde{x}(t)$ is uniformly continuous (UC). Having shown that $\tilde{x}(t) \in \mathcal{L}_\infty \cap \mathcal{L}_2$ and $\tilde{x}(t)$ is UC, then Barbalat's lemma [39] can be used to prove the result in (20).

V. Policy Evaluation

The value function $V^u(x)$ is assumed to be a continuously differentiable function that can be represented by a single-layer NN as

$$V^u(x) = W_v^T \phi(x) + \epsilon(x), \quad (26)$$

where $W_v \in \mathbb{R}^M$ are the unknown ideal weights, $\phi(\cdot) \in \mathbb{R}^M$ is the basis activation function, M denotes the number of hidden layer neurons, and $\epsilon(x) \in \mathbb{R}$ is the bounded function reconstruction error of the NN (i.e. $||\epsilon|| \leq \bar{\epsilon}$). The object of the policy evaluation step is to design weight update laws that approximate the value function. Taking the time derivative of the value function in (2), the following expression is obtained

$$0 = \frac{\partial V^u(x(t))}{\partial x} \dot{x} + r(x, u, d), \quad (27)$$

which is also called the self consistency condition [22], and must hold for any control policy. Using the NN approximation of the value function in (26), the consistency equation in (27) can be written as

$$0 = \left(W_v^T \phi'(x) + \epsilon'(x) \right) \dot{x} + r(x, u, d), \quad (28)$$

where $\phi'(x) \triangleq \frac{\partial \phi}{\partial x} \in \mathbb{R}^{M \times n}$ and $\epsilon'(x) \triangleq \frac{\partial \epsilon}{\partial x} \in \mathbb{R}^{M \times n}$ are the gradients of the activation function and reconstruction error respectively. In addition to Assumption 6 which states that the reconstruction error is bounded, it is further assumed, in the case of the critic NN, that the gradient of the reconstruction error $\epsilon'(x)$ is also bounded. Since the ideal NN weights are unknown, (28) can be written in terms of estimates of the weights as

$$\delta = \hat{W}_v^T \phi'(x) \dot{x} + r(x, u, d), \quad (29)$$

where $\delta(t) \in \mathbb{R}$ is similar to a continuous-time version of the TD error. The goal is to tune the weights $\hat{W}_v(t) \in \mathbb{R}^M$ of the critic NN such that the error $\delta(t)$ is minimized and the critic NN approximately satisfies the self-consistency condition in (27); thus approximating the value function $V^u(x(t))$. It is clear from the expression of the TD error in (29) that knowledge of the system dynamics is required to minimize the error. To overcome this limitation, the DNN-based system identifier $\dot{\hat{x}}(t)$ in (13) is used to replace the system dynamics $\dot{x}(t)$ in (29) to yield a modified expression for the TD error

$$\delta_m = \hat{W}_v^T \phi'(x) \dot{\hat{x}} + r(x, u, d).$$

A standard steepest descent algorithm for NNs is used to minimize the modified TD error $\delta_m(t)$. The objective function $J_\delta(t) \in \mathbb{R}$ for steepest descent method is defined as

$$J_\delta(t) = \frac{1}{2} \delta_m(t)^2.$$

The gradient of the objective function with respect to the weight estimate is given by

$$\frac{\partial J_\delta}{\partial \hat{W}_v} = \delta_m \frac{\partial \delta_m}{\partial \hat{W}_v} = \delta_m \dot{\hat{x}}^T \phi'(x)^T. \quad (30)$$

Using (30), the critic NN weights can be updated as

$$\dot{\hat{W}}_v = -\eta proj\left(\frac{\partial J_\delta}{\partial \hat{W}_v} \right)^T = -\eta proj\left(\delta_m \phi'(x) \dot{\hat{x}} \right), \quad (31)$$

where $\eta \in \mathbb{R}^+$ is the learning rate of the critic NN, and $proj(\cdot)$ is a smooth projection used to guarantee that weight estimate $\hat{W}_v(t)$ remains bounded. Although projection is used in (31) to ensure bounded weights, a persistency of excitation (PE) condition could be used to develop a more precise bound on the weight mismatch error $\tilde{W}_v(t)$, as in [27]. The novelty of the ACI technique [28] is the use of a stable asymptotic identifier $\dot{\hat{x}}(t)$, for updating the critic weights in (31), thus removing the requirement for exact model knowledge.

VI. Policy Improvement

The objective of policy improvement is to select a policy which minimizes the current estimate of the value function in (26). The policy improvement step involves the use of the closed-form solutions in (5) and (6). After substituting for $g(x)$ and $k(x)$ from (11) and (12), and $\frac{\partial V^u(x)}{\partial x}$ from (27), the expressions in (5) and (6) are

$$u = \frac{-1}{2} R^{-1} \left(W_g^T \sigma_g + \varepsilon_g \right)^T \left(\phi'(x)^T W_v + \epsilon'^T \right) \quad (32)$$

$$d = \frac{1}{2\gamma^2} \left(W_k^T \sigma_k + \varepsilon_k \right)^T \left(\phi'(x)^T W_v + \epsilon'^T \right). \quad (33)$$

The approximate control policies $\hat{u}(t)$ and $\hat{d}(t)$, respectively, for the expressions defined in (32) and (33) can be developed as

$$\hat{u} = -\frac{1}{2}R^{-1}\left(\hat{W}_g^T\sigma_g(x)\right)^T\phi'(x)^T\hat{W}_v \quad (34)$$

$$\hat{d} = \frac{1}{2\gamma^2}\left(\hat{W}_k^T\sigma_k(x)\right)^T\phi'(x)^T\hat{W}_v. \quad (35)$$

The (actor) policies defined in (34) and (35) are approximations of the optimal Nash equilibrium policies, however further analysis is needed to ensure guaranteed closed-loop stability as the actor policies refine their approximation with time. The subsequent Lyapunov analysis proves the closed-loop stability of the proposed actor policies and ensures they are admissible.

Assumption 7: For given feedback control policies the nonlinear Lyapunov equation given as [27]

$$0 = Q(x) + \left(\frac{\partial V^u}{\partial x}\right)^T(f(x) + g(x)u + k(x)d) + u^TRu - \gamma^2 d^Td,$$

has a smooth local solution $V^u(x) \geq 0$.

Theorem 2: The approximate optimal control policies in (34) and (35), and the weight update laws in (17), (18), and (19) ensure that, for some initial condition $x(t_0) = x_0$, there exists a time $T(x_0, B)$ such that $x(t)$ is UUB, where the bound B is given by

$$||x(t)|| \leq \sqrt{\frac{\zeta}{\lambda_{\min}\{\bar{Q}\}}} \equiv B \qquad t \geq t_0 + T,$$

where $\zeta \in \mathbb{R}$ and $\bar{Q} \in \mathbb{R}^{n\times n}$ are known positive constants.

Proof: For a positive definite local cost $r(x, u, d)$, it can be shown that $V^u(x) < 0$ for trajectories generated by the optimal control policies $(u^*(t), d^*(t))$ in (5) and (6); hence $x(t)$ is asymptotically stable. To determine the stability of the approximate (actor) control policies in (34) and (35), we take the derivative of $V^u(x)$ along the trajectory generated by the approximate control policies $\hat{u}(t)$ and $\hat{d}(t)$, and use the dynamics in (8) as

$$\dot{V}^u = \frac{\partial V^u(x(t))}{\partial x}^T\left(f(x) + g(x)\hat{u} + k(x)\hat{d} + \varepsilon_t\right). \quad (36)$$

Considering the system with disturbance in (8) and using (7), the value function $V^u(x)$ satisfies the following HJI equation

$$0 = Q(x) + \left(\frac{\partial V^u}{\partial x}\right)^T(f(x) + \varepsilon_t) - \frac{1}{4}\left(\frac{\partial V^u}{\partial x}\right)^T g(x)R^{-1}g^T(x)\frac{\partial V^u}{\partial x} + \frac{1}{4\gamma^2}\left(\frac{\partial V^u}{\partial x}\right)^T k(x)k^T(x)\frac{\partial V^u}{\partial x}. \quad (37)$$

Substituting $\left(\frac{\partial V^u}{\partial x}\right)^T(f(x) + \varepsilon_t)$ from (37) into (36) yields

$$\dot{V}^u = \frac{\partial V^u(x(t))}{\partial x}^T\left(g(x)\hat{u} + k(x)\hat{d}\right) + \frac{1}{4}\left(\frac{\partial V^u}{\partial x}\right)^T g(x)R^{-1}g^T(x)\frac{\partial V^u}{\partial x} - \frac{1}{4\gamma^2}\left(\frac{\partial V^u}{\partial x}\right)^T k(x)k^T(x)\frac{\partial V^u}{\partial x} - Q(x). \quad (38)$$

Adding and subtracting $\frac{\partial V^u(x(t))}{\partial x}^T g(x)u$ to (38) and using (5) yields

$$\dot{V}^u = -\frac{\partial V^u(x(t))}{\partial x}^T\left(u - \hat{u} - k(x)\hat{d}\right) - \frac{1}{4}\left(g^T(x)\frac{\partial V^u}{\partial x}\right)^T R^{-1}\left(g^T(x)\frac{\partial V^u}{\partial x}\right) - \frac{1}{4\gamma^2}\left(k^T(x)\frac{\partial V^u}{\partial x}\right)^T\left(k^T(x)\frac{\partial V^u}{\partial x}\right) - Q(x). \quad (39)$$

Adding and subtracting $\frac{\partial V^u(x(t))}{\partial x}^T\sigma_g^T\hat{W}_g\phi'(x)^TW_v$, substituting (11), (12), (26), (32), (33) (34), and (35) into (39) and performing some algebraic manipulations, the following expression is obtained

$$\begin{aligned}\dot{V}^u = {} & -Q(x) - \frac{1}{4}\left(g^T(x)\frac{\partial V^u}{\partial x}\right)^T R^{-1}\left(g^T(x)\frac{\partial V^u}{\partial x}\right) - \frac{1}{4\gamma^2}\left(k^T(x)\frac{\partial V^u}{\partial x}\right)^T\left(k^T(x)\frac{\partial V^u}{\partial x}\right) \\ & + \left(W_v^T\phi' + \epsilon'\right)^T \\ & \times\left(\frac{1}{2}\left(W_g^T\sigma_g + \varepsilon_g\right)\left[\sigma_g^T\tilde{W}_g\phi'^TW_v + \sigma_g^TW_g\epsilon'^T - \sigma_g^T\hat{W}_g\phi'^T\tilde{W}_v + \varepsilon_g^T\phi'^TW_v + \left(\epsilon'\varepsilon_g\right)^T\right]\right. \\ & \left. - \left(W_k^T\sigma_k + \varepsilon_k\right)\left(\hat{W}_k^T\sigma_k\right)^T\phi'^T\hat{W}_v\right).\end{aligned} \quad (40)$$

Using (11), (12), Assumptions 1-5, (18), (19), (26), and (31) the last term in (40) can be bounded as

$$\begin{aligned}\zeta \geq {} & \left\|\left(\bar{W}_v^T\phi' + \epsilon'\right)^T \right. \\ & \times\left(\frac{1}{2}\left(\bar{W}_g^T\sigma_g + \bar{\varepsilon}_g\right)\left[\sigma_g^T\tilde{W}_g\phi'^T\bar{W}_v + \sigma_g^T\bar{W}_g\epsilon'^T - \sigma_g^T\hat{W}_g\phi'^T\tilde{W}_v + \bar{\varepsilon}_g^T\phi'^T\bar{W}_v + \left(\epsilon'\bar{\varepsilon}_g\right)^T\right]\right. \\ & \left.\left. - \left(\bar{W}_k^T\sigma_k + \bar{\varepsilon}_k\right)\left(\hat{W}_k^T\sigma_k\right)^T\phi'^T\hat{W}_v\right)\right\|,\end{aligned} \quad (41)$$

where $\zeta \in \mathbb{R}$ is a computable constant. Using (41), and the fact that R^{-1} is positive definite, (40) can be upper bounded as

$$\dot{V}^u(x) \leq -Q(x) + \zeta. \quad (42)$$

Assuming $Q(x)$ has a quadratic form $Q(x) = x^T\bar{Q}x$ where $\bar{Q} \in \mathbb{R}^{n\times n}$ is a constant positive definite matrix, $\dot{V}^u(x)$ in

(42) can be further upper bounded as

$$\dot{V}^u(x) \leq -\lambda_{\min}\left\{\bar{Q}\right\}||x||^2 + \zeta, \tag{43}$$

which shows that $\dot{V}^u(x)$ is negative whenever $x(t)$ lies outside the compact set $\Omega_x \triangleq \left\{x : ||x|| \leq \sqrt{\frac{\zeta}{\lambda_{\min}\left\{\bar{Q}\right\}}}\right\}$, and hence, $||x(t)||$ is UUB [40]. For values of $Q(x)$ that satisfy the Lyapunov equation in Assumption 7, the size of Ω_x can be made smaller by increasing $\lambda_{\min}\left\{\bar{Q}\right\}$, the penalty on the state $x(t)$. It can also be seen from (43) that the value function $V^u \in \mathcal{L}_\infty$. Since $x(t)$, $V^u \in \mathcal{L}_\infty$, then the approximate control policies $\hat{u}(t)$ and $\hat{d}(t)$ are admissible (see definition in [4]). Also, from the projection algorithms in (18), (19), and (31) and using the approximate control policies in (34) and (35), $\hat{u}(t)$,$\hat{d}(t) \in \mathcal{L}_\infty$; hence, Assumption 2 holds.

VII. Conclusion

A novel actor-critic-identifier architecture is generalized for a two-player zero-sum differential game. The ACI architecture implements the PI algorithm in real-time, where the actor, critic, and identifier operate simultaneously. The use of a robust DNN-based identifier circumvents the need for complete model knowledge, yielding an identifier which is proven to be asymptotically convergent. A gradient-based weight update law is used for the critic NN to approximate the value function. Using the identifier and the critic, an approximation to the optimal control law (actor) is developed which stabilizes the closed loop system and approaches the optimal solution to the two-player zero-sum game.

References

[1] R. Isaacs, *Differential games: a mathematical theory with applications to warfare and pursuit, control and optimization*. Dover Pubns, 1999.

[2] S. Tijs, *Introduction to Game Theory*. Hindustand Book Agency, 2003.

[3] T. Basar and G. Olsder, *Dynamic Noncooperative Game Theory*. SIAM, PA, 1999.

[4] M. Abu-Khalaf and F. Lewis, "Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach," *Automatica*, vol. 41, no. 5, pp. 779–791, 2005.

[5] T. Basar and P. Bernhard, *H-infinity Optimal Control and Related Minimax Design Problems*. Boston: Birkhäuser, 2008.

[6] A. Barto, R. Sutton, and C. Anderson, "Neuron-like adaptive elements that can solve difficult learning control problems," *IEEE Trans. Syst. Man Cybern.*, vol. 13, no. 5, pp. 834–846, 1983.

[7] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. MIT Press, 1998.

[8] R. Sutton, A. Barto, and R. Williams, "Reinforcement learning is direct adaptive optimal control," *IEEE Contr. Syst. Mag.*, vol. 12, no. 2, pp. 19–22, 1992.

[9] J. Campos and F. Lewis, "Adaptive critic neural network for feedforward compensation," in *Proc. Am. Control Conf.*, vol. 4, 1999.

[10] A. Al-Tamimi, F. L. Lewis, and M. Abu-Khalaf, "Adaptive critic designs for discrete-time zero-sum games with application to h-[infinity] control," *IEEE Trans. Syst. Man Cybern. Part B Cybern.*, vol. 37, pp. 240–247, 2007.

[11] P. Werbos, "Approximate dynamic programming for real-time control and neural modeling," in *Handbook of Intelligent Control: Neural, Fuzzy, and Adaptive Approaches*, D. A. White and D. A. Sofge, Eds. New York: Van Nostrand Reinhold, 1992.

[12] D. Bertsekas and J. Tsitsiklis, *Neuro-Dynamic Programming*. Athena Scientific, 1996.

[13] D. V. Prokhorov and I. Wunsch, D. C., "Adaptive critic designs," *IEEE Trans. Neural Networks*, vol. 8, pp. 997–1007, 1997.

[14] 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. Part B Cybern.*, vol. 38, pp. 943–949, 2008.

[15] ——, "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.

[16] S. Balakrishnan, "Adaptive-critic-based neural networks for aircraft optimal control," *J. Guid. Contr. Dynam.*, vol. 19, no. 4, pp. 893–898, 1996.

[17] G. Lendaris, L. Schultz, and T. Shannon, "Adaptive critic design for intelligent steering and speed control of a 2-axle vehicle," in *Int. Joint Conf. Neural Netw.*, 2000, pp. 73–78.

[18] S. Ferrari and R. Stengel, "An adaptive critic global controller," in *Proc. Am. Control Conf.*, vol. 4, 2002.

[19] D. Han and S. Balakrishnan, "State-constrained agile missile control with adaptive-critic-based neural networks," *IEEE Trans. Control Syst. Technol.*, vol. 10, no. 4, pp. 481–489, 2002.

[20] P. He and S. Jagannathan, "Reinforcement learning neural-network-based controller for nonlinear discrete-time systems with input constraints," *IEEE Trans. Syst. Man Cybern. Part B Cybern.*, vol. 37, no. 2, pp. 425–436, 2007.

[21] L. Baird, "Advantage updating," Wright Lab, Wright-Patterson Air Force Base, OH, Tech. Rep., 1993.

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

[23] J. Murray, C. Cox, G. Lendaris, and R. Saeks, "Adaptive dynamic programming," *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.*, vol. 32, no. 2, pp. 140–153, 2002.

[24] R. Beard, G. Saridis, and J. Wen, "Galerkin approximations of the generalized Hamilton-Jacobi-Bellman equation," *Automatica*, vol. 33, pp. 2159–2178, 1997.

[25] D. Vrabie and F. 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, 2009.

[26] K. Vamvoudakis and F. Lewis, "Online synchronous policy iteration method for optimal control," in *Recent Advances in Intelligent Control Systems*. Springer, 2009, pp. 357–374.

[27] ——, "Online neural network solution of nonlinear two-player zero-sum games using synchronous policy iteration," in *Proc. IEEE Conf. Decis. Control*, 2010.

[28] S. Bhasin, M. Johnson, and W. E. Dixon, "A model-free robust policy iteration algorithm for optimal control of nonlinear systems," in *Proc. IEEE Conf. Decis. Control*, 2010, pp. 3060–3065.

[29] R. Sutton, "Learning to predict by the methods of temporal differences," *Mach. Learn.*, vol. 3, no. 1, pp. 9–44, 1988.

[30] F. L. Lewis, *Optimal Control*. John Wiley & Sons, 1986.

[31] A. Van der Schaft, "L2-gain analysis of nonlinear systems and nonlinear H-[infinity] control," *IEEE Trans. Autom. Control*, vol. 37, no. 6, pp. 770–784, 1992.

[32] D. Kirk, *Optimal Control Theory: An Introduction*. Dover Pubns, 2004.

[33] M. Polycarpou and P. Ioannou, "Identification and control of nonlinear systems using neural network models: Design and stability analysis," *Systems Report 91-09-01, University of Southern California*, 1991.

[34] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Math. Control Signals Syst.*, vol. 2, pp. 303–314, 1989.

[35] F. L. Lewis, R. Selmic, and J. Campos, *Neuro-Fuzzy Control of Industrial Systems with Actuator Nonlinearities*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2002.

[36] F. L. Lewis, "Nonlinear network structures for feedback control," *Asian J. Control*, vol. 1, no. 4, pp. 205–228, 1999.

[37] J. Hopfield, "Neurons with graded response have collective computational properties like those of two-state neurons," *Proc. Nat. Acad. Sci. U.S.A.*, vol. 81, no. 10, p. 3088, 1984.

[38] A. Poznyak, E. Sanchez, and W. Yu, *Differential neural networks for robust nonlinear control: identification, state estimation and trajectory tracking*. World Scientific Pub Co Inc, 2001.

[39] J. Slotine and W. Li, *Applied Nonlinear Control*. Prentice Hall, 1991.

[40] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Prentice Hall, 2002.