Deep-Learning Tracking for Autonomous Flying Systems Under Adversarial Inputs

Luis Rodolfo Garcia Carrillo, *Member, IEEE,* and Kyriakos G. Vamvoudakis, *Senior Member, IEEE*

***Abstract*—We propose a game-theory based deep-learning tracking control scheme to enable a holonomic flying system to perform an autonomous trajectory tracking task, when considering saturating actuators, adversarial inputs, and non-quadratic cost functionals. The problem is formulated as a two-player zero-sum game, whose online solution is computed by learning the saddle point strategies in real time. Three approximators, namely a critic and two actors, are tuned online using data generated in real-time along the system trajectories. The adaptive control character of the algorithm requires a persistence of excitation condition to be a priori validated, which is relaxed by using a deep-learning approach, based on experience replay with multiple layers. A robustifying control term is added to eliminate the effect of residual errors, leading to asymptotic stability of the equilibrium point of the closed-loop system. A simulation of a target tracking application where the measurements available to the aerial system are perturbed by persistent adversaries is performed to validate the effectiveness of the proposed approach.**

***Index Terms*—Deep-learning tracking, autonomy, zero-sum game.**

I. Introduction

Basic requirements for enabling unmanned aircraft systems (UASs) to perform autonomous missions consist of an efficient attitude stabilization and a reliable trajectory tracking framework. By trajectory tracking we mean the problem of stabilizing the state, or an output function of the state, to a desired reference value, possibly time-varying. The trajectory tracking problem incorporates several problems addressed in the control literature, e.g., output feedback regulation, asymptotic stabilization of a fixed-point and, more generally, of admissible non-stationary trajectories. For specific examples of these problems, the interested reader is referred to [1]-[2], and the references therein. The use of UASs in real-time trajectory applications is challenging since most of the times these agents are tasked to accomplish a mission in a hostile environment where all sort of adversaries may exist e.g., cyber-physical attacks, network attacks, wind-gusts, and so on. Under such circumstances, the UAS must be able to adapt its control strategy according to the effects induced by adversaries. This specific characteristic is of importance because adversaries can easily drive the system to an unstable behavior, i.e., undesired/unreliable operations, or even to a mission fail. Establishing a well-defined state-action mapping for the autonomous system is a very complex task, unless the whole state space has been visited or searched exhaustively. Therefore, a machine learning (ML) mechanism adapting to dynamic environment is a more promising solution.

To attenuate adversaries corrupting the sensors and actuators of an unmanned agent, and to guarantee robustness, it is possible to formulate a two-player zero-sum (ZS) game, which is similar to an H_∞ control problem. The major drawback to the practical applications of the H_∞ control is the complications and difficulties involved in solving a highly nonlinear partial differential equation (PDE), which is called a Hamilton-Jacobi-Isaacs (HJI) equation. Indeed, if the system has nonlinearities or the cost is non-quadratic, there is no analytical approach for solving such equation. This has motivated alternative approaches for obtaining approximate solutions to the HJI equation. Recently, PI has emerged as an efficient method for approximating the HJI solution [3], [4]. Under this approach, the HJI is solved successively by breaking it into a sequence of linear PDEs, which are considerably easier to handle. For example, the authors in [5] used approximators to approximate the HJI equation. Despite offering an attractive solution for addressing the H_∞ control problem, the algorithm approached the problem from an offline viewpoint, which is not appropriate for the kind of scenario facing real-time autonomous systems operating in uncertain dynamic and adversarial environments.

The research in [6]-[7] proposed an online adaptive algorithm with guaranteed closed-loop stability of the equilibrium point for solving the HJI equation. However, the online algorithm does not take into account the input or operator constraints caused by actuators saturation. Considering input constraints is important since real world applications of control methods involve actuators with limitations in their amplitude. Ignoring these limitations may lead to undesirable transient response, could degrade closed-loop performance, and system instability. The aforementioned work requires a persistence of excitation (PE) condition that is equivalent to space exploration in Reinforcement Learning (RL) [4]. This condition is prohibitory and most of the times infeasible to implement in practice. Recently, the research presented in [8] introduced a method to adaptive control that relies on implementing current and recorded data concurrently for adaptation.

L.R. Garcia Carrillo is with the Unmanned Systems Laboratory, Department of Electrical Engineering, Texas A&M University - Corpus Christi, Corpus Christi, TX, 78412-5797, USA e-mail: luis.garcia@tamucc.edu

K. G. Vamvoudakis is with the Daniel Guggenheim School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA e-mail: kyriakos@gatech.edu

This work was supported in part, by ARO under grants W911NF1810210, W911NF-19-1-0270 by ONR Minerva under grant No. N00014-18-1-2160, by NATO under grant No. SPS G5176, by an NSF CAREER under grant No. CPS-1851588, and by NSF under grant No. SATC-1801611

Manuscript received Month Day, Year; revised Month Day, Year.

Related Work

Several techniques have been proposed to solve the tracking control problem for UASs, e.g., [9]-[12], and the references therein. These control frameworks allowed the system to track three desired positions and one heading angle in real-time. However, there has not been any work on optimal tracking control capable of simultaneously attenuating the effects of adversarial inputs. An algorithm for the generation of dynamically feasible trajectories subjected to collision and obstacle avoidance constraints has been developed in [9]. The trajectory tracking task is simplified by neglecting important vehicle dynamic constraints, and therefore making possible the real-time planning in cluttered environments. Adaptive switching supervisory control combined with a nonlinear Lyapunov-based tracking control law was implemented in [10] for an underactuated vehicle, allowing to solve the problem of global boundedness and convergence of the position tracking error to a neighborhood of the origin. A two-stages attitude and translational control designed for a sub-actuated UAS allowing trajectory tracking without linear-velocity measurements was presented in [13]. The authors proposed a linear-velocity-free control torque, designed for ensuring tracking of the desired attitude derived at the first stage of the control design. In [14], a nonlinear controller for an UAS is proposed using output feedback and a novel virtual control input scheme, which allows controlling the six degrees of freedom (DOF). The controller performance is demonstrated under unknown nonlinear dynamics and disturbances, and simulation results are provided to validate their theory. Closely related, the work in [15] presents an optimal controller design based on Neural Networks (NN) for trajectory tracking of a UAS.

Recently, the research community has shown interest in Deep NN (DNNs) that learn to represent data in multiple layers of increasing abstraction. In many control systems, the dimension of the input is high, therefore, feature engineering algorithms used in conventional shallow NNs are not efficient enough to extract the complex and nonlinear patterns observed in high variety of sensory data. The problem of autonomous navigation of a UAS by using a model-based RL approach has been addressed in [16]. An important aspect in this kind of solutions is that a poor feature representation in conventional NNs used to approximate value function in RL can lead to a poor learning task. Layer-by-layer, learning in DNNs helps avoid local optima and alleviates the over-fitting problem encountered in traditional NNs [17]. Moreover, DNNs algorithms extract efficient complex features at high levels of abstraction in a greedy layer-wise fashion [18]-[19].

Heuristic approaches have demonstrated that data representations obtained from stacking up nonlinear feature extractors in DNNs yield better ML results, compared to conventional shallow learning approaches [20]. This motivated researches to combine DNNs with RL and introduce Deep RL (DRL) algorithms to approximate value functions to cope with large input dimensions. For example, [21]-[22] introduced DRL to reduce the need for sustained exhaustive exploration during learning. Deep Q-Network (DQN) uses DNN to approximate the Q-network and train this Q-network to predict total reward [23]. The Effects of Memory Replay in RL have been studied in [24], where the authors show that the amount of memory kept can affect the agents' performance; too much or too little memory both slow down learning. In [25] asynchronous actor-critic algorithm merges a DQN with a deep policy network for choosing actions. [26] proposed Double DQN (D-DQN) to tackle the overestimate problem in Q-learning. DRL has received attention for continuous control problems, e.g., robotic manipulation, locomotion, and games [27]-[34].

Contributions

This paper relies on the development of a DL algorithm based on experience replay with multiple layers, where the two players are represented by the autonomous operator (control) and the adversarial input. In order to approximate the HJI equation, three approximators, namely a critic and two actors (operator and adversary), are tuned online using data generated in real-time along the system trajectories. The problem is formulated as a two-player ZS game due to the fact that operator and adversary have opposite objectives. For that reason, it makes sense to look for saddle point policies. If a game theoretic saddle point exists, then the two-player optimal control tracking problem has a unique solution, equivalent to the Nash equilibrium, which is valid for all policies u_o, u_a.

This paper extends our previous results in [35] were PI algorithms and NN approximators are proposed for solving a two-player ZS game in real-time. The main contribution of the current study is in the updating strategy of the critic approximator. Instead of using only current data, the proposed approach makes use of recorded and instantaneous data concurrently for adaptation. This adaptation strategy or DL procedure mitigates requiring persistency of excitation in the approximator activation functions. The necessary mathematical proofs are provided in order to demonstrate stability of this solution. A second important contribution is in the application itself. While the previous study focuses on control of the dynamical system, the current study focuses on the error dynamics to achieve trajectory tracking task. A third contribution of this research consists of the incorporation of operator constraints in the performance function, which exemplify the physical limitations of real-time robotic systems. Saturation constrains introduce nonlinearities which are hard to address with conventional methodologies. The fourth contribution consists of considering the adversarial components, which may affect the system during the execution of the target tracking task. The proposed approach demonstrates its effectiveness and applicability by stabilizing a complex system, despite these theoretical and practical challenges.

Organization: The dynamics of a holonomic UAS is provided in Section II. In Section III we formulate the problem. Section IV introduces the combination of the HJI Equation and the ZS game. The proposed approximate solution, which consists of a DL structure based on approximators, is presented in Section V. The simulation of a trajectory tracking task in the presence of adversary inputs is presented in Section VI. Section VII provides conclusions and future directions of this research. Lyapunov proofs ensuring asymptotic stability of the system are provided in a the appendix Section VIII.

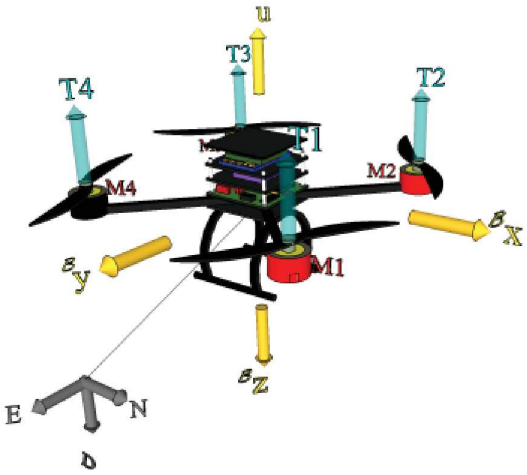

Fig. 1. The holonomic multirotorcraft UAS. This vehicle has four control inputs, and six states. The sub-actuated nature of the mulirotorcraft UAS makes it a very challenging system to control.

II. Background on UAS Dynamics

The nonlinear dynamic model of the holonomic multirotorcraft UAS is obtained in North-East-Down (NED) inertial and body-fixed coordinates, see Figure 1. Let $\mathcal{I} = \{N, E, D\}$ denote the inertial reference frame and $\mathcal{B} = \{^{\mathcal{B}}x, ^{\mathcal{B}}y, ^{\mathcal{B}}z\}$ a body-fixed frame. The position vector of the UAS center of mass is $\xi = [x, y, z]^{\mathrm{T}} \in \mathcal{I}$, representing the position coordinates relative to the $\mathcal{I}$. The orientation of the UAS with respect to (w.r.t.) the $\mathcal{I}$ is expressed by $[\psi, \theta, \phi]^{\mathrm{T}} \in \mathcal{I}$, where ψ, θ, and ϕ are the yaw, pitch, and roll Euler angles, respectively. Let $v \in \mathcal{I}$ represent the linear velocity expressed in $\mathcal{I}$ and $\Omega \in \mathcal{B}$ denote the angular velocity of the UAS expressed in $\mathcal{B}$. The mass of the UAS is denoted by m, and $\mathbb{I} \in \mathbb{R}^{3\times3}$ represents the constant inertia matrix around the centre of mass expressed in $\mathcal{B}$. Newton's equations of motion provide a dynamic model for the motion of the UAS by following [36] as:

$$\dot{\xi} = v \tag{1}$$
$$m\dot{v} = m\bar{g}e_3 + \mathcal{R}F \tag{2}$$
$$\dot{\mathcal{R}} = \mathcal{R}\mathrm{sk}(\Omega) \tag{3}$$
$$\mathbb{I}\dot{\Omega} = -\Omega \times \mathbb{I}\Omega + \Gamma. \tag{4}$$

The vector $F \in \mathcal{B}$ incorporates the non-conservative forces applied to the UAS including the thrusts (produced by the rotors) and drag terms associated with the rotors downwash on the airframe. The torque $\Gamma \in \mathcal{B}$ is obtained from differential thrust associated with pairs of rotors, together with aerodynamic effects and gyroscopic effects, e_3 denotes a unit vector in the D-axis direction, and $\bar{g} = 9.81$ m/s^2. Also, $\mathcal{R} \in SO(3)$ is a rotation matrix relating a vector in $\mathcal{B}$ to $\mathcal{I}$ [37]:

$$\mathcal{R} = \begin{bmatrix} c_\theta c_\psi & s_\phi s_\theta c_\psi - c_\phi s_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\theta s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{bmatrix}, \tag{5}$$

where $\mathcal{R}$ uses the notation $c_* = \cos(*)$ and $s_* = \sin(*)$. Note that $\|\mathcal{R}\|_F = \mathcal{R}_{\max}$ given a known constant $\mathcal{R}_{\max}$, $\mathcal{R}^{-1} = \mathcal{R}^{\mathrm{T}}$, $\dot{\mathcal{R}} = \mathcal{R}\mathrm{sk}(\Omega)$, and $\dot{\mathcal{R}}^{\mathrm{T}} = -\mathrm{sk}(\Omega)\mathcal{R}^{\mathrm{T}}$, where $\mathrm{sk}(*) \in \mathbb{R}^{3\times3}$ is a skew symmetric matrix satisfying $k^{\mathrm{T}}\mathrm{sk}(d)k = 0$, for any $k \in \mathbb{R}^3$ and $d \in \mathbb{R}^3$ [38]. This holds because we consider the UAS to operate only in regions where $-(\pi/2) < \phi < (\pi/2)$ and $-(\pi/2) < \theta < (\pi/2)$, i.e., the trajectory does not pass through any singularities [39].

For this study, the following model is proposed:

$$\dot{v} = \bar{g}e_3 - \frac{1}{m}T\mathcal{R}e_3 + \mathcal{N}_1(v) + \delta_1 \tag{6}$$
$$\mathbb{I}\dot{\Omega} = -\Omega \times \mathbb{I}\Omega + \tau + G_a + \mathcal{N}_2(\Omega) + \delta_2 \tag{7}$$

where $T = [0, 0, \bar{u}]^{\mathrm{T}}$, is the thrust along the $^{\mathcal{B}}z$-direction generated by the rotors, e.g., $\bar{u} = \sum_{i=1}^{4} \mathrm{T}_i$, $\mathcal{N}_i(*) \in \mathbb{R}^3$, $i = 1, 2$ are nonlinear aerodynamic effects, G_a are gyroscopic torques applied to the frame, and $\tau \in \mathbb{R}^3$ is defined as

$$\begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} -l & l & l & -l \\ -l & -l & l & l \\ -C_M & C_M & -C_M & C_M \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \end{bmatrix} \tag{8}$$

where τ_ϕ, τ_θ, τ_ψ are the rotational torques, C_M is a constant depending on the rotor characteristics, and l represents the distance between the center of mass and the center of the motor. $\delta_i \in \mathbb{R}^3$, $i = 1, 2$ represent unknown bounded disturbances such that $\|\delta_i\| < \delta_M$, for all t, with $\delta_M > 0$ as a known constant. The system in equations (6)-(7) is affected by two inputs: the control input, and the disturbance input.

A translational dynamics tracking error can be defined as

$$e_\xi = \xi - \xi_d \quad \in \mathcal{I}, \tag{9}$$

with corresponding velocity error as

$$e_v = v - v_d. \tag{10}$$

Define the new augmented variables $X := [\xi^{\mathrm{T}} \ [\psi \ \ \theta \ \ \phi]^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{6\times1}$ and $V := [v^{\mathrm{T}} \ \Omega^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{6\times1}$. To ensure that the system in equations (6)-(7) follows the desired trajectory expressed by $V_d \in \mathbb{R}^{6\times1}$ with dynamics given by

$$\dot{V}_d = f(V_d) + g(V_d)u_d \tag{11}$$

where $f(V_d) \in \mathbb{R}^{6\times1}$ represents the internal dynamics, expressed in terms of V_d, $g(V_d) \in \mathbb{R}^{6\times1}$ is given such that $g_{\min} \leq \|g\|_F \leq g_{\max}$, and u_d represents the required control input corresponding to the desired state behavior [14].

The state tracking error $e \in \mathbb{R}^{6\times1}$ can be expressed as

$$e = V - V_d, \tag{12}$$

with dynamics given by

$$\dot{e} = f(e(t)) + g(e(t))u_o(t) + k(e(t))u_a(t), \tag{13}$$

where $e(0) \equiv e_0$, $t \geq 0$, $k(e(t)) \in \mathbb{R}^{6\times1}$, $f(e) := f(V) - f(V_d)$, and $u_o := u_V - u_d$. Notice that the error system (13) has two inputs, the operator u_o, and the adversary u_a.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2019.2930017, IEEE Transactions on Aerospace and Electronic Systems

III. Problem Formulation

The goal is to design a strategy to track the desired trajectory $V_d(t)$, maintaining a stable flight, and attenuating the effects induced by the adversary, while simultaneously optimizing a tracking cost function related to equations (6)-(7). Towards this end, a cost function is defined as

$$J(e(0), u_o, u_a) = \int_0^\infty r(e(\tau), u_o(\tau), u_a(\tau))\mathrm{d}\tau, \qquad (14)$$

where the utility is given by

$$r(e, u_o, u_a) = Q(e) + R_s(u_o) - \gamma^2 \left\|u_a\right\|^2, \quad \forall\ e, u_o, u_a$$

with $Q(e) \succeq 0$, $\gamma \geq \gamma^* \geq 0$, and the dependence on t has been suppressed. The term γ^* is commonly known as the H_∞ gain, and is associated to the smallest γ for which the system is stabilized [40]. Hence, we are interested in finding the following optimal cost function

$$C^*(e(t)) \equiv \min_{u_o} \max_{u_a} \int_t^\infty r(e(\tau), u_o, u_a)\mathrm{d}\tau, \quad t \geq 0 \qquad (15)$$

subjected to the dynamical constraints (13). Note that the game is formulated in such a way the operator u_o is a minimizing player while the adversary u_a is a maximizing player.

In order to force *bounded inputs*, (e.g. $u_o \leq \bar{u}_o$) the term $R_s(u_o)$ is given by

$$R_s(u_o) = 2\int_0^{u_o} (\mathrm{sat}^{-1}(v))^{\mathrm{T}} R\mathrm{d}v, \quad \forall\ u_o \qquad (16)$$

where $R = R^{\mathrm{T}} \succ 0$, $v \in \mathbb{R}^m$, and $\mathrm{sat}(\cdot)$ is a continuous, one-to-one real-analytic integrable function of class C^μ, $\mu \geq 1$ used to map the interval $[-\bar{u}_o, \bar{u}_o]$ onto $\mathbb{R}$, and must satisfy $\mathrm{sat}(0) = 0$ [41], [42]. Also, note that $R_s(u_o)$ is positive definite because $\mathrm{sat}^{-1}(v)$ is monotonic odd. □

IV. HJI Equation and the Zero-Sum Game

Operator and adversary have opposite objectives and for that reason we look for saddle point policies. If a game theoretic saddle point (u_o^*, u_a^*) exists, the two-player optimal control tracking problem has a unique solution. That is, the following Nash condition must hold

$$\min_{u_o} \max_{u_a} J(e(0), u_o, u_a) = \max_{u_a} \min_{u_o} J(e(0), u_o, u_a) \qquad (17)$$

which is equivalent to the Nash equilibrium condition

$$J(e(0), u_o^*, u_a) \leq J(e(0), u_o^*, u_a^*) \leq J(e(0), u_o, u_a^*) \qquad (18)$$

and is valid for all policies u_o, u_a.

The Hamiltonian of dynamics in equation (13) and cost function in equation (14) is

$$H = r(e, u_o, u_a) + (\nabla C)^{\mathrm{T}}(f(e) + g(e)u_o + k(e)u_a), \ \forall e, u_o, u_a, \qquad (19)$$

where $\nabla C \equiv \partial C/\partial e \in \mathbb{R}^{6\times 1}$ is the transposed gradient. Given a solution $C^*(e) \geq 0 : \mathbb{R}^n \to \mathbb{R}$ to the Hamiltonian in equation (19), the associated operator and adversary for the system in equation (13) can be found by employing the stationarity conditions on equation (19) as

$$\frac{\partial H}{u_o} = 0 \Rightarrow u_o^* = -\mathrm{sat}\left(\frac{1}{2}R^{-1}g^{\mathrm{T}}(e)\nabla C^*(e)\right) \qquad (20)$$

$$\frac{\partial H}{u_a} = 0 \Rightarrow u_a^* = \frac{1}{2\gamma^2}k^{\mathrm{T}}(e)\nabla C^*(e). \qquad (21)$$

The optimal cost function in equation (15) and the associated constrained operator and adversary satisfy the HJI equation $\forall e$

$$H^* = Q(e) + \nabla C^{*\mathrm{T}}(e)f(e) - \frac{1}{4}\nabla C^{*\mathrm{T}}(e)g(e)R^{-1}g^{\mathrm{T}}(e)\nabla C^*(e) + \frac{1}{4\gamma^2}\nabla C^{*\mathrm{T}}(e)kk^{\mathrm{T}}\nabla C^*(e) = 0 \qquad (22)$$

with $C^*(0) = 0$.

Assumption 1: The solution $C^*(e)$ to equation (22) is smooth and positive definite, that is, $0 < C^*(e) \in C^1$. □

Lemma 1: Select $\gamma > 0$. Suppose that there exist a smooth positive definite solution $C(e)$, to the HJI equation (22). Assume equation (13) is zero-state observable. Then, the system in equation (13) has $L_2 -$ gain $\leq \gamma$. Moreover selecting the control in equation (20) in terms of the HJI solution solves the $L_2 -$ gain problem, and makes the equilibrium point locally asymptotically stable (when $u_a(t) = 0$).

Proof: The proof is provided in Subsection A of the Appendix. Further details can be found in [43]-[44] ■

Our algorithm to solve the HJI equation is based on the structure of PI. The proof of convergence is provided in [45].

Algorithm 1: PI Two-Player ZS Differential Games

procedure
  Start with a stabilizing feedback control policy $u_o^{(0)}$.
  for $j = 0, 1, \ldots$ given $u_o^{(j)}$ **do**
    set $u_a^{(0)}$
    for $i = 0, 1, \ldots$ **do**
      Solve for $C_j^{(i)}(e)$ using ZS Bellman equation

$$0 = Q(e) + \nabla C_j^{i\mathrm{T}}(e)(f + gu_o^{(j)} + ku_a^{(i)}) + R_s(u_o^{(j)}) - \gamma^2 \left\|u_a^{(i)}\right\|^2 \qquad (23)$$

      Update adversarial input $u_a^{(i+1)}$ according to

$$u_a^{(i+1)} = \arg\max_{u_a} \left[H(e, \nabla C_j^i, u_o^{(j)}, u_a)\right] = \frac{1}{2\gamma^2}k^{\mathrm{T}}(e)\nabla C_j^i \qquad (24)$$

    end for
    On convergence, set $C_{j+1}(e) = C_j^i(e)$
    Update control policy with

$$u_o^{(j+1)} = \arg\min_{u_o} \left[H(e, \nabla C_{j+1}), u_o, u_a)\right] = -\mathrm{sat}\left(\frac{1}{2}R^{-1}g^{\mathrm{T}}(e)\nabla C_{j+1}\right) \qquad (25)$$

    if $\left\|C_j^i - C_j^{i-1}\right\| \leq \epsilon_0$ **then** go to 14
    end if
  end for
end procedure

In **Algorithm 1**, $\epsilon_0 \in \mathbb{R}^+$ is a scalar that checks the algorithm convergence. The PI algorithm consists of two loops: an outer feedback operator update loop and an inner adversary update loop. The following section introduces the methodology for updating everything simultaneously, by using data along the system trajectories.

V. Online Solution

An online adaptive DL optimal control structure for solving the two-player ZS game problem in real-time is introduced.

A. The Critic Approximator

To approximate the optimal cost function given by equation (15) within a compact set $\Omega \subseteq \mathbb{R}^n$ that contains the origin, one can use an approximator in the form

$$C^*(e) = W_c^{*\mathrm{T}}\Phi_c(e) + \epsilon_c(e), \quad \forall\, e \tag{26}$$

where the ideal weights $W_c^* \in \mathbb{R}^N$ satisfy $\|W_c^*\| \leq W_{\mathrm{cmax}}$, $\Phi_c(e) : \Omega \to \mathbb{R}^N$ represents the approximator activation function vector composed by N basis functions $\{\varphi(e) : i = 1, \ldots, N\}$, and $\epsilon_c(e)$ is the online approximation error. The approximator activation functions must be chosen so that they provide a complete independent basis set, in such a way that $C(e)$ and its derivative are uniformly approximated [46]. Computing the derivative w.r.t. e yields

$$C_e^* = \left(\frac{\partial \Phi_c(e)}{\partial e}\right)^{\mathrm{T}} W_c^* + \frac{\partial \epsilon_c}{\partial e} = \nabla\Phi_c^{\mathrm{T}} W_c^* + \nabla\epsilon_c. \tag{27}$$

From the Weierstrass higher order approximation theorem, as $N \to \infty$ the approximation errors $\epsilon_c \to 0$, $\nabla\epsilon_c \to 0$ uniformly [41]. Additionally, for fixed N the approximation errors are locally bounded by constants [46].

Assumption 2: The approximation error ϵ_c and its derivative are bounded by $\epsilon_{c\mathrm{max}}, \epsilon_{c\mathrm{dmax}} \in \mathbb{R}^+$ in a compact set $\Omega \subseteq \mathbb{R}^n$ as, $\sup_{e\in\Omega}|\epsilon_c| \leq \epsilon_{c\mathrm{max}}$ and $\sup_{e\in\Omega}|\nabla\epsilon_c| \leq \epsilon_{c\mathrm{dmax}}$ respectively. Moreover, the activation functions Φ_c and their derivatives $\nabla\phi_c$ are upper bounded as, $|\Phi_c| \leq \Phi_{c\mathrm{max}}$ and $|\nabla\Phi_c| \leq \Phi_{c\mathrm{dmax}}$ respectively. □

Using equation (27) and fixed control/adversarial policies in equation (22), define an approximate Hamiltonian as

$$\begin{aligned} H(e, W_c^{*\mathrm{T}}\nabla\Phi_c, u_o, u_a) \equiv\ & Q(e) + R_s(u_o) \\ & + W_c^{*\mathrm{T}}\nabla\Phi(f + gu_o + ku_a) \\ =\ & \epsilon_{\mathrm{H}}, \quad \forall e, u_o, u_a \end{aligned} \tag{28}$$

with a residual error due to the function approximation as

$$\epsilon_{\mathrm{H}} = -\nabla\varepsilon_c^{\mathrm{T}}(f + gu_o + ku_a), \quad \forall e, u_o, u_a. \tag{29}$$

Assumption 3: The residual error ϵ_H is bounded by ϵ_{Hmax} on a compact set $\Omega \subseteq \mathbb{R}^n$, i.e., $\sup_{e\in\Omega}|\epsilon_H| \leq \epsilon_{\mathrm{Hmax}}$. □

The ideal weight vector W_c^* of the critic approximator which provides the best approximate solution for equation (28) is unknown. Then, the current critic approximator estimate is

$$\hat{C}(e) = \hat{W}_c^{\mathrm{T}}\Phi_c(e), \quad \forall e \tag{30}$$

where $\hat{W}_c$ represents the estimated values of W_c. The goal is to find an update law for $\hat{W}_c$ to ensure that $\hat{W}_c \to W_c^*$, such that the approximate Hamiltonian for fixed $u_o,\ u_a$ is

$$\begin{aligned} \hat{H}(e, \hat{W}_c^{\mathrm{T}}\nabla\Phi_c, u_o, u_a) \equiv\ & \hat{W}_c^{\mathrm{T}}\varrho(t) + Q(e(t)) \\ & + R_s(u_o(t)) - \gamma^2\left\|u_a(t)\right\|^2 \overset{t\to\infty}{\to} H^*, \quad \forall\, e, u_o, u_a \end{aligned} \tag{31}$$

with $\varrho(t) = \nabla\Phi_c(f(e(t)) + g(e(t))u_o(e(t)) + k(e(t))u_a(e(t)))$. To mitigate the need for persistence of excitation of the vector $\varrho(t)$, we follow the procedure in [47] for the model reference adaptive control case. The methodology proposed here uses a DL approach for storing past recorded data together with current data. Define an error associated to the current data as

$$e_{\mathrm{H}} := \hat{H} - H^*, \quad \forall\, e, t \tag{32}$$

and an $e_{\mathrm{HWi}} \in \mathbb{R}$ on the previous stored data be defined as

$$e_{\mathrm{HWi}} := \hat{H}_{Wi} - H^*, \quad \forall\, e, t, t_i \tag{33}$$

with $H^* = 0$ from equation (22). To drive e_{H} and e_{HWi} to zero, we rely on adaptive control techniques [44]. An error performance is defined from combining these terms as

$$E = \frac{1}{2}\frac{e_{\mathrm{H}}^{\mathrm{T}} e_{\mathrm{H}}}{(\varrho(t)^{\mathrm{T}}\varrho(t) + 1)^2} + \frac{1}{2}\sum_{i=1}^{k}\frac{e_{\mathrm{HWi}}^{\mathrm{T}} e_{\mathrm{HWi}}}{(\varrho(t_i)^{\mathrm{T}}\varrho(t_i) + 1)^2} \tag{34}$$

The size of the window of stored data is $k \in \mathbb{Z}^+$. A gradient descent rule is used for defining the tuning of the critic approximator as

$$\begin{aligned} \dot{\hat{W}}_c =\ & -\alpha\frac{\partial E}{\partial \hat{W}_c} \\ =\ & -\alpha\frac{\varrho(t)}{(\varrho(t)^{\mathrm{T}}\varrho(t) + 1)^2}e_{\mathrm{H}} - \alpha\sum_{i=1}^{k}\frac{\varrho(t_i)}{(\varrho(t_i)^{\mathrm{T}}\varrho(t_i) + 1)^2}e_{\mathrm{HWi}} \\ =\ & -\alpha\frac{\varrho(t)}{(\varrho(t)^{\mathrm{T}}\varrho(t) + 1)^2} \\ & \left(\varrho(t)^{\mathrm{T}}\hat{W}(t) + R_s(u_o(t)) + Q(e(t)) - \gamma^2\left\|u_a(t)\right\|^2\right) \\ & - \alpha\sum_{i=1}^{k}\frac{\varrho(t_i)}{(\varrho(t_i)^{\mathrm{T}}\varrho(t_i) + 1)^2}\left(\varrho(t_i)^{\mathrm{T}}\hat{W}(t)\right. \\ & \left. + Q(e(t_i)) + R_s(u_o(t_i)) - \gamma^2\left\|u_a(t_i)\right\|^2\right), \quad t > t_i \geq 0 \end{aligned} \tag{35}$$

where $\alpha > 0$ determines the speed of convergence.

The critic approximator weight estimation error is

$$\tilde{W}_c := W_c^* - \hat{W}_c \tag{36}$$

The dynamics of the critic weight estimation error are then

$$\begin{aligned} \dot{\tilde{W}}_c =\ & -\alpha\left(\frac{\varrho(t)\varrho(t)^{\mathrm{T}}}{(\varrho(t)^{\mathrm{T}}\varrho(t) + 1)^2}\right. \\ & \left. + \sum_{i=1}^{k}\frac{\varrho(t_i)\varrho(t_i)^{\mathrm{T}}}{(\varrho(t_i)^{\mathrm{T}}\omega\varrho(t_i) + 1)^2}\right)\tilde{W}_c(t) \\ & + \alpha\left(\frac{\varrho(t)}{(\varrho(t)^{\mathrm{T}}\varrho(t) + 1)^2}\epsilon_H(t)\right. \\ & \left. + \sum_{i=1}^{k}\frac{\varrho(t_i)}{(\varrho(t_i)^{\mathrm{T}}\varrho(t_i) + 1)^2}\epsilon_H(t_i)\right) \\ \equiv\ & -N_{\mathrm{s}} + P_{\mathrm{s}},\ t > t_i \geq 0 \end{aligned} \tag{37}$$

where the first term is the nominal system and the second term is the perturbation due to the error ϵ_{H}.

Theorem 1: Let the tuning of the critic approximator be given by equation (35). Then, the nominal system from equation (37) is exponentially stable with its trajectories satisfying $\left\|\tilde{W}_c(t)\right\| \leq \left\|\tilde{W}_c(t_0)\right\| k_1 e^{-k_2(t-t_0)}$, $\forall\, t > t_i \geq t_0 \geq 0$ and for some $k_1, k_2 \in \mathbb{R}^+$ provided that $\{\varrho(t_1), \ldots, \varrho(t_k)\}$ contains N linearly independent vectors.

Proof: The proof is provided in Subsection B of the Appendix.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2019.2930017, IEEE Transactions on Aerospace and Electronic Systems

B. Operator Approximator

It is possible to approximate the operator input in equation (20) by an actor approximator as follows

$$u_o^* = W_o^{*\mathrm{T}}\Phi_{\mathrm{uo}}(e) + \epsilon_{\mathrm{uo}}(e),\ \forall\ e \tag{38}$$

with $W_{\mathrm{uo}}^* \in \mathbb{R}^{N_2\times 4}$ representing the optimal weights, $\Phi_{\mathrm{uo}}(e)$ the approximator activation functions, N_2 is the number of basis functions, and an actor approximation error defined by ϵ_{uo}. The approximator activation functions has to define a complete independent basis set such that $u_{\mathrm{uo}}^*(e)$ is uniformly approximated. The Weierstrass higher order approximation theorem [46] ensures that, as the number of basis sets $N_2 \to \infty$, the approximation error $\epsilon_{\mathrm{uo}} \to 0$.

Due to the fact that the optimal weights W_o^* are not known, the operator actor approximator with current weights $\hat{W}_{\mathrm{uo}}$ is

$$\hat{u}_o(e) = \hat{W}_{\mathrm{uo}}^{\mathrm{T}}\Phi_{\mathrm{uo}}(e),\ \forall\ e. \tag{39}$$

The error $e_{\mathrm{uo}} \in \mathbb{R}$, between equations (39) and (20) is then

$$e_{\mathrm{uo}} = \hat{W}_{\mathrm{uo}}^{\mathrm{T}}\Phi_{\mathrm{uo}} + \mathrm{sat}\left(\frac{1}{2}R^{-1}g^{\mathrm{T}}(e)\nabla\Phi^{\mathrm{T}}\hat{W}_c\right)$$

The goal is to select weights $\hat{W}_{\mathrm{uo}}$ in such a way that the expression below is minimized

$$E_{\mathrm{uo}} = \frac{1}{2}e_{\mathrm{uo}}^{\mathrm{T}}e_{\mathrm{uo}},\quad \forall\ \ \hat{W}_c, e, u_o, t \geq 0. \tag{40}$$

The tuning for the weights of the operator are obtained from a gradient descent procedure in equation (40), yielding

$$\begin{aligned}\dot{\hat{W}}_{\mathrm{uo}} &= -\alpha_{\mathrm{uo}}\frac{\partial E_{\mathrm{uo}}}{\partial \hat{W}_{\mathrm{uo}}} = -\alpha_{\mathrm{uo}}\Phi_{\mathrm{uo}}e_{\mathrm{uo}} \\ &= -\alpha_{\mathrm{uo}}\phi_{\mathrm{uo}}\left(\hat{W}_{\mathrm{uo}}^{\mathrm{T}}\Phi_{\mathrm{uo}} + \mathrm{sat}\left(\frac{1}{2}R^{-1}g^{\mathrm{T}}(e)\nabla\Phi^{\mathrm{T}}\hat{W}_c\right)\right)^{\mathrm{T}}\end{aligned} \tag{41}$$

$\forall\ t \geq 0$, and the constant $\alpha_{\mathrm{uo}} > 0$ determines the speed of convergence. The weight estimation error for the operator approximator is given by

$$\tilde{W}_{\mathrm{uo}} := W_{\mathrm{uo}}^* - \hat{W}_{\mathrm{uo}}. \tag{42}$$

Following a similar approach with the one for the critic tuning law in (37), the operator error dynamics are given by

$$\begin{aligned}\dot{\tilde{W}}_{\mathrm{uo}} = &-\alpha_{\mathrm{uo}}\Phi_{\mathrm{uo}}\Phi_{\mathrm{uo}}^{\mathrm{T}}\tilde{W}_{\mathrm{uo}} \\ &- \alpha_{\mathrm{uo}}\Phi_{\mathrm{uo}}\mathrm{sat}\left(\frac{1}{2}R^{-1}g^{\mathrm{T}}(e)\nabla\Phi^{\mathrm{T}}\tilde{W}_c\right)^{\mathrm{T}} - \alpha_{\mathrm{uo}}\Phi_{\mathrm{uo}}\epsilon_{\mathrm{uo}} \\ &- \alpha_{\mathrm{uo}}\Phi_{\mathrm{uo}}\mathrm{sat}\left(\frac{1}{2}R^{-1}g^{\mathrm{T}}(e)\nabla\epsilon_c\right)^{\mathrm{T}},\ t \geq 0.\end{aligned} \tag{43}$$

C. Adversary Approximator

In a similar way, the worst case adversary (21) can be approximated by an adversary approximator as

$$u_a^*(e) = W_a^{*\mathrm{T}}\Phi_{\mathrm{ua}}(e) + \epsilon_{\mathrm{ua}}(e),\ \forall\ e \tag{44}$$

with $W_{\mathrm{ua}}^* \in \mathbb{R}^{N_2\times 4}$ representing the optimal weights, $\Phi_{\mathrm{ua}}(e)$ as the approximator activation functions, N_2 is the number of basis functions, and an actor approximation error ϵ_{ua}. The approximator activation functions has to define a complete independent basis set such that $u_{\mathrm{ua}}^*(e)$ is uniformly approximated. The Weierstrass higher order approximation theorem [46] ensures that, as the number of basis sets $N_2 \to \infty$, the approximation error $\epsilon_{\mathrm{ua}} \to 0$.

Since the W_a^* are unknown then the current adversarial approximator with weights $\hat{W}_{\mathrm{ua}}$ is written as

$$\hat{u}_a(e) = \hat{W}_{\mathrm{ua}}^{\mathrm{T}}\Phi_{\mathrm{ua}}(e),\ \forall\ e. \tag{45}$$

An expression for the error $e_{\mathrm{ua}} \in \mathbb{R}$ between equations (45) and (21) is then given by

$$e_{\mathrm{ua}} = \hat{W}_{\mathrm{ua}}^{\mathrm{T}}\Phi_{\mathrm{ua}} - \frac{1}{2\gamma^2}k^{\mathrm{T}}(e)\nabla\Phi^{\mathrm{T}}\hat{W}_c.$$

Our objective is to find the weights $\hat{W}_{\mathrm{ua}}$ such that the expression below is minimized

$$E_{\mathrm{ua}} = \frac{1}{2}e_{\mathrm{ua}}^{\mathrm{T}}e_{\mathrm{ua}},\quad \forall\ \hat{W}_c, e, u_a. \tag{46}$$

The tuning for the weights of the adversary is obtained from a gradient descent procedure in equation (46), yielding

$$\begin{aligned}\dot{\hat{W}}_{\mathrm{ua}} &= -\alpha_{\mathrm{ua}}\frac{\partial E_{\mathrm{ua}}}{\partial \hat{W}_{\mathrm{ua}}} = -\alpha_{\mathrm{ua}}\Phi_{\mathrm{ua}}e_{\mathrm{ua}} \\ &= -\alpha_{\mathrm{ua}}\phi_{\mathrm{ua}}\big(\hat{W}_{\mathrm{ua}}^{\mathrm{T}}\Phi_{\mathrm{ua}} - \frac{1}{2\gamma^2}k^{\mathrm{T}}(e)\nabla\Phi^{\mathrm{T}}\tilde{W}_c\big)^{\mathrm{T}}\end{aligned} \tag{47}$$

$\forall\ t \geq 0$, and $\alpha_{\mathrm{ua}} > 0$ is a constant value determining the speed of convergence. For the adversarial input, the weight estimation error is given by

$$\tilde{W}_{\mathrm{ua}} := W_{\mathrm{ua}}^* - \hat{W}_{\mathrm{ua}}. \tag{48}$$

The adversarial error dynamics are given by

$$\begin{aligned}\dot{\tilde{W}}_{\mathrm{ua}} = &-\alpha_{\mathrm{ua}}\Phi_{\mathrm{ua}}\Phi_{\mathrm{ua}}^{\mathrm{T}}\tilde{W}_{\mathrm{ua}} + \left(\frac{1}{2\gamma^2}k^{\mathrm{T}}(e)\nabla\Phi^{\mathrm{T}}\tilde{W}_c\right)^{\mathrm{T}} \\ &- \alpha_{\mathrm{ua}}\Phi_{\mathrm{ua}}\epsilon_{\mathrm{ua}} + \left(\frac{1}{2\gamma^2}k^{\mathrm{T}}(e)\nabla\Phi^{\mathrm{T}}\epsilon_{\mathrm{ua}}\right)^{\mathrm{T}},\qquad t \geq 0.\end{aligned} \tag{49}$$

The adaptive-optimal control algorithm is presented below as pseudo-code. Comments are shown after the symbol ▷.

Algorithm 2: DL Optimal Tracking

1: Start with initial state $e(0)$, random initial weights $\hat{W}_c(0)$, $\hat{W}_{\mathrm{uo}}(0)$, $\hat{W}_{\mathrm{ua}}(0)$, and $i = 1$
2: **procedure**
3: Propagate $t, e(t)$ using (13), $u_o(t) := \hat{W}_{\mathrm{uo}}^{\mathrm{T}}\Phi_{\mathrm{uo}}(e)$, and $u_a(t) := \hat{W}_{\mathrm{ua}}^{\mathrm{T}}\Phi_{\mathrm{ua}}(e)$ ▷ $\{e(t)$ comes from integrating the system (13) using any ordinary differential equation (ode) solver (e.g., Runge Kutta). Time t is given by the Runge Kutta integration, i.e. $[t_i, t_{i+1}],\ i \in \mathcal{N}$ where $t_{i+1} := t_i + h$ with $h \in \mathbb{R}^+$ the step size$\}$
4: Propagate $\hat{W}_c(t)$, $\hat{W}_{\mathrm{uo}}(t)$, $\hat{W}_{\mathrm{ua}}(t)$ ▷ $\{$integrate $\dot{\hat{W}}_c$ as in (35), $\dot{\hat{W}}_{\mathrm{uo}}$ as in (41), and $\dot{\hat{W}}_{\mathrm{ua}}$ as in (47) using ode solver$\}$
5: Compute $\hat{C}(e) = \hat{W}_c^{\mathrm{T}}\Phi_c(e)$ ▷ output of the Critic
6: Compute $\hat{u}_o(e) = \hat{W}_{\mathrm{uo}}^{\mathrm{T}}\Phi_{\mathrm{uo}}(e)$ ▷ output of the Operator
7: Compute $\hat{u}_a(e) = \hat{W}_{\mathrm{ua}}^{\mathrm{T}}\Phi_{\mathrm{ua}}(e)$ ▷ output of the Adversary
8: **if** $i \neq k$ **then** ▷ $\{\{\varrho(t_1), \varrho(t_2), \ldots, \varrho(t_i)\}$ has N linearly independent elements and t_k is the instant of time that this happens$\}$
9: Chose arbitrary data point, include it in the history stack.
10: $i := i + 1$
11: **end if** ▷ when history stack is full
12: **end procedure**

Remark 1: **Algorithm 2** is executed in real-time in a plug-n-play scheme, without any iterations. Every procedure happens simultaneously as soon as new measurements of the sate along the trajectories are received. One measures the state $e(t)$ and integrates the tuning laws (35), (41), (47) by using any ode solver, and then compute $\hat{C}(e) = \hat{W}_c^{\mathrm{T}}\Phi_c(e)$, $\hat{u}_o(e) = \hat{W}_{\mathrm{uo}}^{\mathrm{T}}\Phi_{\mathrm{uo}}(e)$, and $\hat{u}_a(e) = \hat{W}_{\mathrm{ua}}^{\mathrm{T}}\Phi_{\mathrm{ua}}(e)$. Numerical methods implemented in state of the art software are most of the time adaptive algorithms where the step size h is fine-tuned at each step, according to an estimate of the error at that particular step. Generally, the calculation time increases as h is decreased, but, it is also more precise. Unfortunately, if h is considerably decreased, the slight rounding occurring in the computer (since it is not able to exactly represent real numbers) starts to accumulate in such a way it will cause significant errors. For numerous higher order systems, it is extremely complicated to render the Euler approximation effective. Runge Kutta methods for non-stiff problems furnish calculations which are linear to the size of the problem. For stiff problems, more exact procedures were designed. □

D. Stability Analysis

Assumption 4: The function $g(\cdot)$ is uniformly bounded on Ω, i.e., $\sup_{e\in\Omega}\|g(e)\| < g_{\max}$. Similarly, the function $k(\cdot)$ is uniformly bounded on Ω, i.e., $\sup_{e\in\Omega}\|k(e)\| < k_{\max}$. □

Fact 1: There exists a constant $\varrho_{\max}$ such that the following normalized signal satisfies [44]

$$\left\| \frac{\varrho}{(\varrho^{\mathrm{T}}\varrho + 1)} \right\| \leq \varrho_{\max} := \frac{1}{2}, \quad \forall \varrho. \tag{50}$$

□

To dispose of the effects of the approximation errors ϵ_c, ϵ_{uo}, ϵ_{ua} (and corresponding partial derivatives) and secure an asymptotically stable closed-loop system, we include a robustifying control term to equations (39) and (45), leading to the following control law and adversarial input equations

$$u_o(t) = \hat{u}_o(e(t)) - \frac{e(t)^{\mathrm{T}}e(t)}{A + e(t)^{\mathrm{T}}e(t)} B\mathbf{1}_m, \quad \forall t \tag{51}$$

$$u_a(t) = \hat{u}_a(e(t)) - \frac{e(t)^{\mathrm{T}}e(t)}{A + e(t)^{\mathrm{T}}e(t)} D\mathbf{1}_m, \quad \forall t \tag{52}$$

where $\hat{u}_o$ is given by equation (39), $\hat{u}_a$ is given by equation (45), and $A, B, D \in \mathbb{R}^+$ with

$$\begin{aligned} B \geq \frac{A + e^{\mathrm{T}}e}{e^{\mathrm{T}}e(W_{c\max}\Phi_{c\mathrm{dmax}} + \epsilon_{c\mathrm{dmax}})g_{\max}} \Bigg\{ & \frac{1}{4\alpha}\left(2\varrho_{\max}\epsilon_{H\max}\right)^2 \\ & + \frac{\left(\Phi_{uo\max}\bar{u}_o + \Phi_{uo\max}\epsilon_{uo\max}\right)^2}{2} \\ & + \frac{g_{max}\Phi_{uo\max}}{2}\left((W_{c\max}\Phi_{c\mathrm{dmax}} + \epsilon_{c\mathrm{dmax}})\right)^2 \\ & + \frac{1}{2}(W_{c\max}\Phi_{c\mathrm{dmax}} + \epsilon_{c\mathrm{dmax}})^2 + \frac{1}{2}\epsilon_{uo\max}^2 \Bigg\} \end{aligned} \tag{53}$$

$$\begin{aligned} D \geq \frac{A + e^{\mathrm{T}}e}{e^{\mathrm{T}}e(W_{c\max}\Phi_{c\mathrm{dmax}} + \epsilon_{c\mathrm{dmax}})k_{\max}} \Big\{ & \\ \frac{\left(\frac{1}{2\gamma^2}\Phi_{ua\max}\mathrm{k}_{\max}\epsilon_{c\mathrm{dmax}} + \Phi_{ua\max}\epsilon_{ua\max}\right)^2}{2} & \\ + \frac{k_{\max}\Phi_{ua\max}}{2}\left((W_{c\max}\Phi_{c\mathrm{dmax}} + \epsilon_{c\mathrm{dmax}})\right)^2 & \\ + \frac{1}{2}(W_{c\max}\Phi_{c\mathrm{dmax}} + \epsilon_{c\mathrm{dmax}})^2 + \frac{1}{2}\epsilon_{ua\max}^2 \Big\} & \end{aligned} \tag{54}$$

where $\bar{u}_o$ is the saturation limit.

Now we write the system dynamics in equation (13) as

$$\begin{aligned} \dot{e} = & f(e) + g(e)\left((W_{\mathrm{uo}}^* - \tilde{W}_{\mathrm{uo}})^{\mathrm{T}}\Phi_{\mathrm{uo}}(e) - B\frac{e^{\mathrm{T}}e\mathbf{1}_m}{\left(A + e^{\mathrm{T}}e\right)}\right) \\ & + k(e)\left((W_{\mathrm{ua}}^* - \tilde{W}_{\mathrm{ua}})^{\mathrm{T}}\Phi_{\mathrm{ua}}(e) - D\frac{e^{\mathrm{T}}e\mathbf{1}_m}{\left(A + e^{\mathrm{T}}e\right)}\right) \quad t \geq 0. \end{aligned} \tag{55}$$

The following theorem presents our main results.

Theorem 2: Consider the dynamics given by equation (13), the operator given by equation (51), the adversary given by equation (52), and also that $e_{\mathrm{buff}\,j} = [\ \varrho(t_1)\ \ \varrho(t_2)\ \ \ldots\ \ \varrho(t_k)\ ]$, $\forall\ j \in \mathbb{Z}^+$ has N linearly independent elements. The tuning laws for the critic, the operator, and adversarial approximators are stated by equations (35), (41), and (47), respectively. Then, the solution $\left(e(t), \tilde{W}_c(t), \tilde{W}_{\mathrm{uo}}(t), \tilde{W}_{\mathrm{ua}}(t)\right)$ converges asymptotically to zero for all $\left(e(0), \tilde{W}_c(0), \tilde{W}_{\mathrm{uo}}(0), \tilde{W}_{\mathrm{ua}}(0)\right)$, provided that the inequalities below are satisfied

$$\begin{aligned} \Bigg(2\alpha\lambda_{\min}\Bigg(\sum_{i=1}^{k}\frac{\varrho(t_i)\varrho(t_i)^{\mathrm{T}}}{(\varrho(t_i)^{\mathrm{T}}\varrho(t_i)+1)^2}\Bigg) & \\ - \frac{1}{4\alpha} - \frac{1}{4\gamma^2}\Phi_{ua\max}\mathrm{k}_{\max}\Phi_{c\max}\Bigg) > 0 & \end{aligned} \tag{56}$$

$$\left(\Phi_{uo\max}^2 - \Phi_{uo\max}\bar{u}_o - \frac{1}{2} - \frac{g_{\max}\Phi_{uo\max}}{2}\right) > 0 \tag{57}$$

$$\begin{aligned} \Bigg(\Phi_{ua\max}^2 - \frac{1}{2} - \frac{k_{\max}\Phi_{ua\max}}{2} & \\ - \frac{1}{4\gamma^2}\Phi_{ua\max}\mathrm{k}_{\max}\Phi_{c\max}\Bigg) > 0. & \end{aligned} \tag{58}$$

Proof: The proof is provided in Subsection C of the Appendix.

VI. Numerical Simulations

For validating the theoretical developments, a quadrotorcraft UAS is tasked to perform a trajectory tracking mission, subjected to adversary inputs, e.g., cyber-attacks, jamming signals, or wind gusts. The goal is to follow the desired trajectory while keeping the deviations close to zero, despite the presence of adversary inputs affecting the performance. The desired trajectory corresponds to a circular shape of a radius equal to 5m, located at an altitude of 24m above the ground plane of $\mathcal{I}$. The desired trajectory is

$$\begin{aligned} x_d &= 5\cos(t/10)\ [\mathrm{m}]; & y_d &= 5\sin(t/10)\ [\mathrm{m}] \\ z_d &= 24\ [\mathrm{m}]; & \psi_d &= \pi/8\ [\mathrm{rad}] \end{aligned}$$

The saturation of the controller comes from attitude dynamics limitations, and is chosen as $\theta < \theta_{\max}$ and $\phi < \phi_{\max}$ for $\theta_{\max} = 40°$ and $\phi_{\max} = 40°$. This selection is done according to real bounds encountered in commercial UAS platforms, e.g., the Parrot ARDrone or Bebop [49], in such a way that

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2019.2930017, IEEE Transactions on Aerospace and Electronic Systems

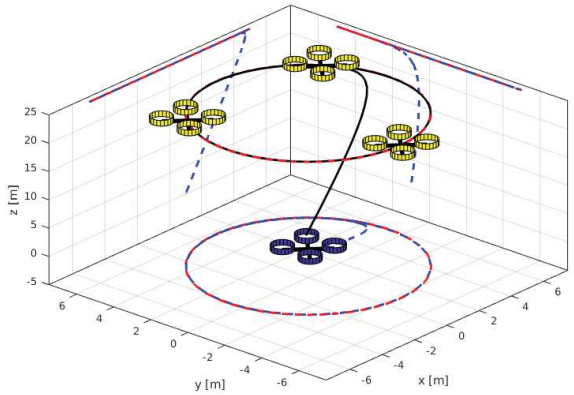

Fig. 2. A 3D plot showing the trajectory performed by the UAS. The desired trajectory is the red circle located at 24m above the (x,y) plane. The figure also shows the projection of the desired trajectory in the (x,z) plane, (y,z) plane, and (x,y) plane. The UAS starts at the origin, and then moves to the desired trajectory in a circular motion.

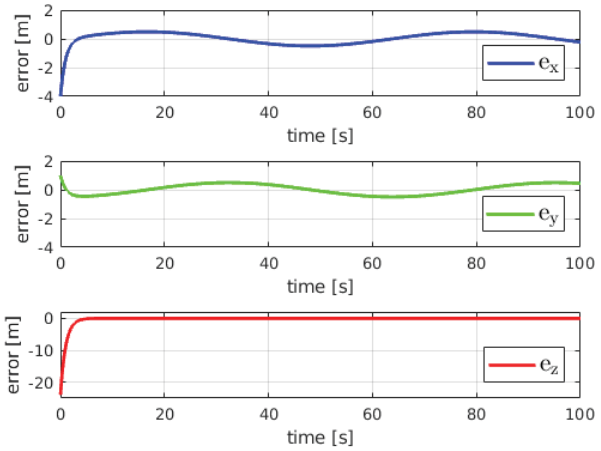

Fig. 3. Tracking errors associated with the 3-dimensional position. Notice that the errors approach zero as the mission is executed.

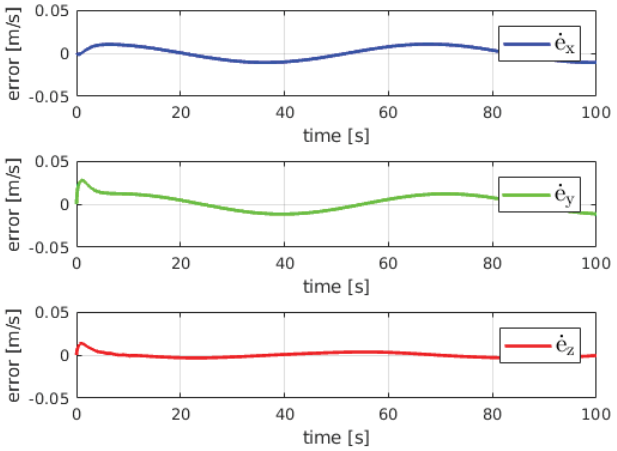

Fig. 4. Tracking errors associated with the translational velocities. Notice that the errors approach zero as the mission is executed.

the system maintains a stable flight, and the desired trajectory avoids mathematical singularities.

The operator u_o generates the overall (vertical) thrust as well as the three control torques τ_ψ, τ_ϕ, τ_θ required to produce translational and heading displacements. The combination of thrust and attitude motions enable the UAS to follow the desired trajectory. The initial weights of the NN are initialized randomly in $[0, 1]$, while $\alpha = 10$, $\alpha_{\text{uo}} = 2$, $\alpha_{\text{ua}} = 2$ were chosen as tuning gains. The H_∞ gain is $\gamma = 4$, the activation functions for the critic $\Phi_c(e)$ are picked quadratic and the activation functions for the operator, and for the adversary are picked as the Jacobians of the critic activation functions, i.e. $\equiv \Phi_{\text{uo}}(e) = \Phi_{\text{ua}}(e) \equiv \nabla\Phi_c(e)$, and $Q = \mathbb{I}^{12\times12}$, $R = \mathbb{I}^{4\times4}$.

At the beginning of the tracking mission, the UAS is at the origin of the (x,y) plane in $\mathcal{I}$. Next, the UAS starts regulating its altitude, while approaching the circular trajectory. These maneuvers are executed while the UAS is subjected to adversarial inputs. To exemplify the tracking mission, a 3D plot of the UAS's position is given in Figure 2, which includes also the desired trajectory. The plot also shows the projections of the desired trajectory in the (x,z) and (y,z) inertial planes.

Figure 3 and 4 show the position and velocity tracking errors, respectively. Figure 5 shows the attitude dynamics generated by the trajectory tracking controller. Operator signals are shown in Figure 6. The adversaries affecting the system dynamics are shown in Figure 7. Note that the DL procedure of the approximator takes place during the first 10 seconds. This mild exploration guarantees the persistence of excitation. This is the classical exploration/exploitation dilemma in every learning mechanism during transient. These results verify that the proposed DL controller converges to a near-optimal solution, as pointed out by the theoretical results.

To evaluate the robustness of the proposed approach against noise of different levels and characteristics, five additional simulations were performed, and the results are shown in Figures 8, 9, and 10. In these plots, the solid lines represent the original trajectory tracking presented in Figure 2, while the faded lines correspond to the five additional tests. From these tests, we observed that when the level of noise is low, (two of the simulations) the algorithm is able to execute the trajectory tracking in an appropriate way. On the other hand, we observed that that high levels of noise degrade the performance and convergence (three of the simulations).

From the results in Figures 8, 9, and 10, we concluded that high levels of noise interfere with the DL procedure. For this reason, an additional test was performed to evaluate the significance of the experience replay method. For the results shown in Figures 11, 12, 13, the buffer containing the data was modified in order to contain (i) no data at all, (ii) small amount of data, (iii) high amount of data, and (iv) the same amount of data used for the original tracking presented in Figure 2. From these results, we observed that the amount of data implemented directly affects the performance of the trajectory tracking. Indeed, too much or too little data are both an issue. While more data provides appropriate results,

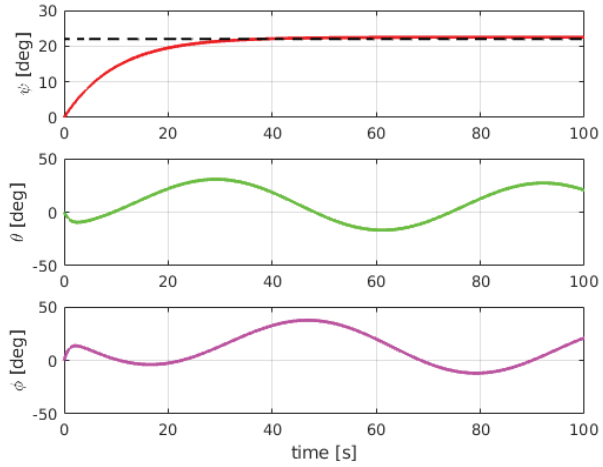

Fig. 5. Attitude dynamics generated by the deep-learning trajectory tracking controller. These angles, in combination with the overall thrust, generate the motion required for performing the trajectory tracking.

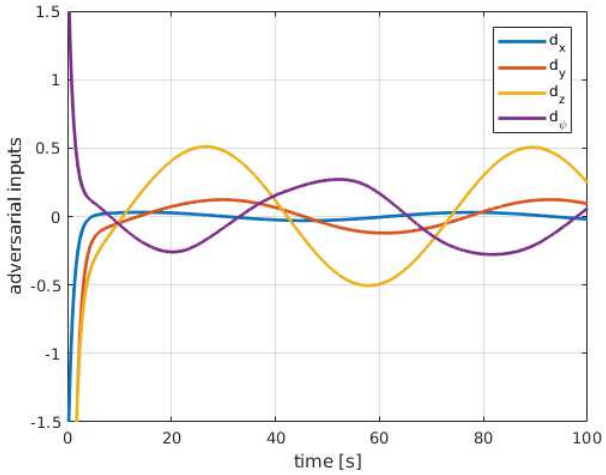

Fig. 7. Adversarial inputs affecting the dynamics while performing the autonomous mission. Notice that these signals are persistently affecting the UAS while executing the trajectory tracking mission.

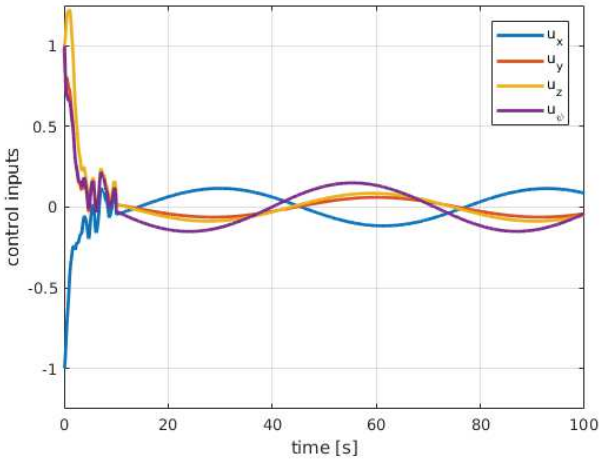

Fig. 6. Operator signals. Note that the DL procedure takes place during the first 10 seconds. This exploration guarantees persistence of excitation. The behavior observed is the classical exploration/exploitation dilemma in a learning mechanism during transient.

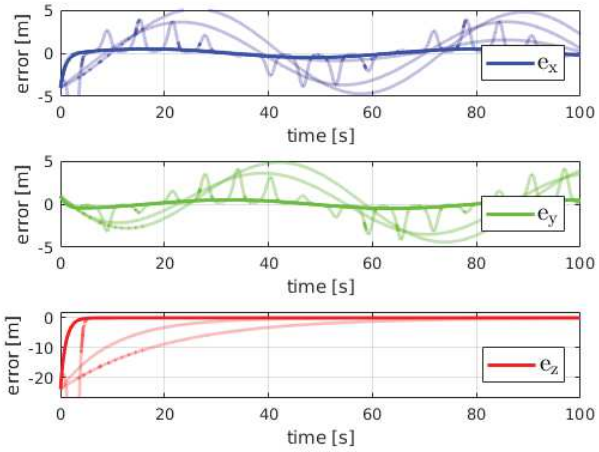

Fig. 8. Tracking errors associated with the 3D position. The faded lines correspond to the five additional tests. For low levels of noise, the convergence is not affected, while high levels of noise degrade the performance.

the algorithm required more time for generating the control signals.

Finally, Figure 14 shows the disturbance attenuation level achieved by the proposed methodology, for the tracking scenario presented in Figure 2. Notice that after the learning is done, the signal is always kept below a level of 4, which is in accordance with the H_∞ gain.

VII. Conclusions and Future Directions

This research presented a novel approximate dynamic programming DL algorithm for enabling a UAS to perform a trajectory tracking in the presence of adversarial inputs. The novel algorithm, which considers bounded control inputs, relaxes the restrictive PE condition by using a DL approach, storing data concurrently with current data in the update of the critic approximator. In order to subdue the effects of the critic and actors approximation errors, a new additional term has been incorporated to the controller, and, by taking into account a suitable Lyapunov function, asymptotic stability of the overall closed loop system is proved. Numerical results of the UAS performing a trajectory tracking mission under adversarial inputs demonstrate the effective and efficient performance of the proposed deep-learning approach.

Future research will extend the results to handle completely unknown systems, as well as multiple entry points for an potential adversary, including sensors and communication.

VIII. Appendix

A. Proof of Lemma 1

For any C^1 function $C(e) : \mathbb{R}^{12} \to \mathbb{R}$ one has the orbital derivative along the trajectories,

$$\dot{C} = \frac{\partial C}{\partial e}\big(f(e) + g(e)u_o(t) + k(e)u_a(t)\big).$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2019.2930017, IEEE Transactions on Aerospace and Electronic Systems

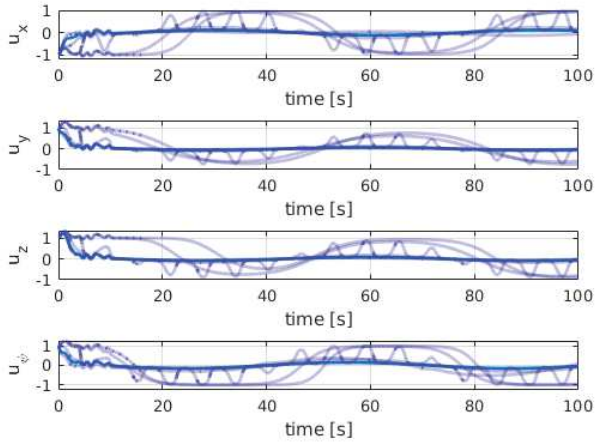

Fig. 9. Operator signals. The faded lines correspond to the five additional tests. For low levels of noise, the operator signals are smoother.

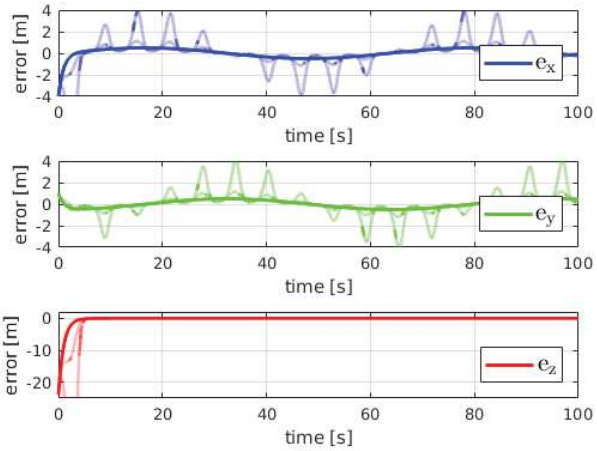

Fig. 11. Tracking errors associated with the 3D position. The faded lines correspond to the four additional tests. As more data is implemented, the errors become smaller.

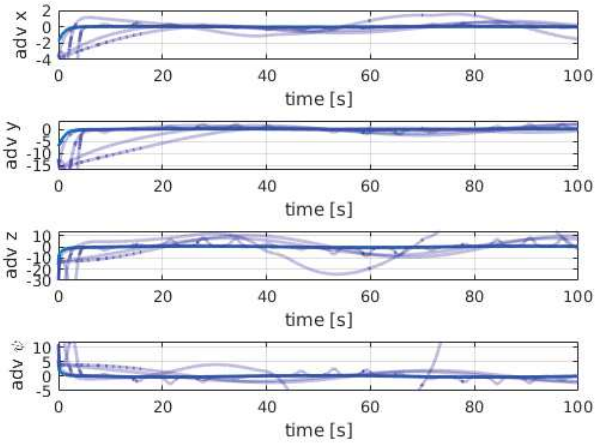

Fig. 10. Adversarial inputs affecting the dynamics. The faded lines correspond to the five additional tests. These signals are more aggressive than the ones used for the original trajectory tracking experiment.

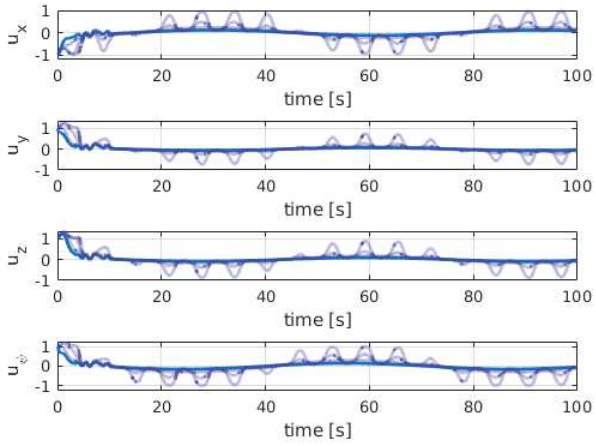

Fig. 12. Operator signals. The faded lines correspond to the four additional tests. Smoother signals are obtained as more data is implemented.

If $C(e) \geq 0$ satisfies the HJI equation (22), then, complete the squares in the Hamiltonian (19) to obtain

$$H = H^* - \gamma^2 \left\| u_a - u_a^* \right\|^2 + (u_o - u_o^*)^{\mathrm{T}} R (u_o - u_o^*) \quad (59)$$

with u_o^* and u_a^* given by (20) and (21), respectively.

Selecting now $u_o = u_o^*$ given by equation (20) with $C(e) \geq 0$ and integrating yields

$$C(e(T)) - C(e(0)) + \int_0^{\mathrm{T}} r(e(\tau), u_o(\tau), u_a(\tau)) \mathrm{d}\tau \leq 0 \quad (60)$$

for all $u_a(t)$. Since $C(0) = 0,\ C(e) \geq 0$, one has

$$\int_0^{\mathrm{T}} r(e(\tau), u_o^*(\tau), u_a(\tau)) \mathrm{d}\tau \leq 0, \forall T > 0.$$

Setting $u_o^*(t) = u_o(t),\ u_a(t) = 0$ in equation (59) yields

$$\dot{C} \leq -Q(e) - R_s(u_o), \quad (61)$$

so that $C(e)$ serves as a Lyapunov equation and the system without an adversarial input is locally stable. Assuming now equation (13) is zero-state observable, then equation (61) is negative definite and equation (13) is locally asymptotically stable. Then, $u_o(t) = u_o^*(t) \in L_2[0, \infty)$, $u_a(t) \in L_2[0, \infty)$, and as $T \to \infty$, equation (60) becomes

$$\int_0^{\infty} (Q(e) + R_s(u_o)) \mathrm{d}\tau \leq \gamma^2 \int_0^{\infty} \left\| u_a \right\|^2 \mathrm{d}\tau. \quad (62)$$

■

B. Proof of Theorem 1

Consider a Lyapunov function

$$\mathcal{Y} = \frac{1}{2\alpha} \tilde{W}_c(t)^{\mathrm{T}} \tilde{W}_c(t),\ \forall\ t \geq 0 \quad (63)$$

Differentiating equation (63) along the error dynamics of the nominal system trajectories yields

$$\dot{\mathcal{Y}} = -\tilde{W}_c(t)^{\mathrm{T}} \left(\frac{\varrho(t) \varrho(t)^{\mathrm{T}}}{(\varrho(t)^{\mathrm{T}} \varrho(t) + 1)^2} \right.$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TAES.2019.2930017, IEEE Transactions on Aerospace and Electronic Systems

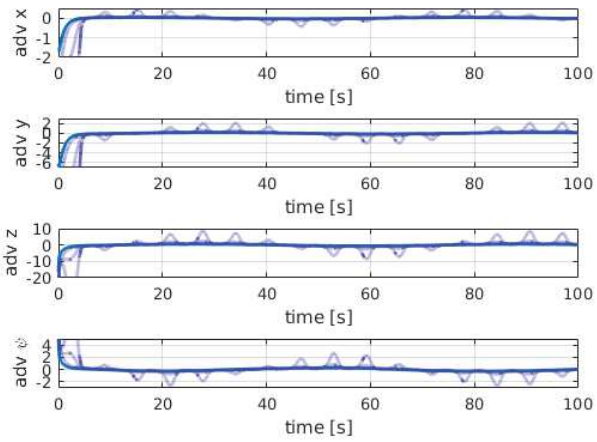

Fig. 13. Adversarial inputs affecting the dynamics. The faded lines correspond to the four additional tests. The adversary signals are more aggressive if the amount of data is too much or too little.

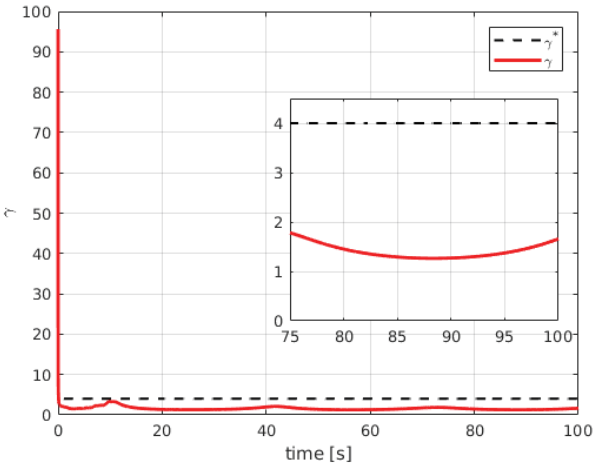

Fig. 14. Instantaneous disturbance attenuation, see equation (62) in Appendix part A. Notice that after the learning is done, the signal is always kept below a level of 4, which is in accordance with the H_∞ gain.

$$+\sum_{i=1}^{k}\frac{\varrho(t_i)\varrho(t_i)^{\mathrm{T}}}{(\varrho(t_i)^{\mathrm{T}}\varrho(t_i)+1)^2}\Big)\tilde{W}_c(t). \quad (64)$$

But $\frac{\varrho(t)\varrho(t)^{\mathrm{T}}}{(\varrho(t)^{\mathrm{T}}\varrho(t)+1)^2} > 0$, $\forall\varrho(t)$, and therefore

$$\dot{\mathcal{Y}} \leq -\tilde{W}_c(t)^{\mathrm{T}}\sum_{i=1}^{k}\frac{\varrho(t_i)\varrho(t_i)^{\mathrm{T}}}{(\varrho(t_i)^{\mathrm{T}}\varrho(t_i)+1)^2}\tilde{W}_c(t). \quad (65)$$

Letting $\mathcal{A} := \sum_{i=1}^{k}\frac{\varrho(t_i)\varrho(t_i)^{\mathrm{T}}}{(\varrho(t_i)^{\mathrm{T}}\varrho(t_i)+1)^2}$, equation (65) yields

$$\dot{\mathcal{Y}} \leq -\lambda_{\min}(\mathcal{A})\left\|\tilde{W}_c\right\|^2, \ t \geq 0, \quad (66)$$

from which the result follows. ■

C. Proof of Theorem 2

A Lyapunov equation is proposed, for all $t \geq 0$, as

$$\mathcal{V} = C^* + V_c(\tilde{W}) + V_{\mathrm{uo}} + V_{\mathrm{ua}} \quad (67)$$

$$\equiv C^* + V_c(\tilde{W}) + \frac{1}{2\alpha_{\mathrm{uo}}}tr\left\{\tilde{W}_{\mathrm{uo}}^{\mathrm{T}}\tilde{W}_{\mathrm{uo}}\right\} + \frac{1}{2\alpha_{\mathrm{ua}}}tr\left\{\tilde{W}_{\mathrm{ua}}^{\mathrm{T}}\tilde{W}_{\mathrm{ua}}\right\}$$

where the optimal value function is given by C^*, and $V_c(\tilde{W})$ is a Lyapunov function for the nominal of the critic error dynamics (see equation (37)). From Theorem 1 and class-$\mathcal{K}$ functions $\gamma_1(\cdot)$ and $\gamma_2(\cdot)$, it follows that

$$\gamma_1\left(\left\|\tilde{Z}\right\|\right) \leq \mathcal{V} \leq \gamma_2\left(\left\|\tilde{Z}\right\|\right),$$

for all $\tilde{Z} \equiv [\ e(t)^{\mathrm{T}} \quad \tilde{W}_c(t)^{\mathrm{T}} \quad \tilde{W}_{\mathrm{uo}}(t)^{\mathrm{T}} \quad \tilde{W}_{\mathrm{ua}}(t)^{\mathrm{T}}\ ]^{\mathrm{T}} \in B_\rho$, where $B_\rho \subset \Omega$ is a ball of radius $\rho \in \mathbb{R}^+$. Using the update for the operator in equation (41) and adversarial input in equation (47), and grouping terms, the derivative of equation (67) (first term w.r.t. the state trajectories with $\hat{u}_{o\mathrm{new}}$ and $\hat{u}_{a\mathrm{new}}$ (equation (55)), and the second term w.r.t. to the perturbed critic estimation error dynamics in equation (37)) becomes

$$\begin{aligned}
\dot{\mathcal{V}} = C_e^{*\mathrm{T}}\Big(& f(e) \\
& - g(e)\tilde{W}_{\mathrm{uo}}^{\mathrm{T}}\Phi_{\mathrm{uo}} + g(e)(u_o^* - \epsilon_{\mathrm{uo}}) - g(e)B\frac{e^{\mathrm{T}}e\mathbf{1}_m}{(A+e^{\mathrm{T}}e)} \\
& - k(e)\tilde{W}_{\mathrm{ua}}^{\mathrm{T}}\Phi_{\mathrm{ua}} + k(e)(u_a^* - \epsilon_{\mathrm{ua}}) - k(e)D\frac{e^{\mathrm{T}}e\mathbf{1}_m}{(A+e^{\mathrm{T}}e)}\Big) \\
& - \frac{\partial V_c}{\partial\tilde{W}_c}^{\mathrm{T}}\left(\frac{\varrho(t)\varrho(t)^{\mathrm{T}}}{(\varrho(t)^{\mathrm{T}}\varrho(t)+1)^2} + \sum_{i=1}^{k}\frac{\varrho(t_i)\varrho(t_i)^{\mathrm{T}}}{(\varrho(t_i)^{\mathrm{T}}\varrho(t_i)+1)^2}\right)\tilde{W}_c \\
& + \frac{\partial V_c}{\partial\tilde{W}_c}^{\mathrm{T}}\left(\frac{\varrho(t)}{(\varrho(t)^{\mathrm{T}}\varrho(t)+1)^2}\epsilon_H(t)\right. \\
& \left. + \sum_{i=1}^{k}\frac{\varrho(t_i)}{(\varrho(t_i)^{\mathrm{T}}\varrho(t_i)+1)^2}\epsilon_H(t_i)\right) \\
& + \tilde{W}_{\mathrm{uo}}^{\mathrm{T}}\Big(- \Phi_{\mathrm{uo}}\Phi_{\mathrm{uo}}^{\mathrm{T}}\tilde{W}_{\mathrm{uo}} - \Phi_{\mathrm{uo}}\mathrm{sat}\left(\frac{1}{2}R^{-1}g^{\mathrm{T}}(e)\nabla\Phi^{\mathrm{T}}\tilde{W}_c\right) \\
& - \Phi_{\mathrm{uo}}\mathrm{sat}\left(\frac{1}{2}R^{-1}g^{\mathrm{T}}(e)\nabla\epsilon_c\right) - \Phi_{\mathrm{uo}}\epsilon_{\mathrm{uo}}\Big) \\
& + \tilde{W}_{\mathrm{ua}}^{\mathrm{T}}\Big(- \Phi_{\mathrm{ua}}\Phi_{\mathrm{ua}}^{\mathrm{T}}\tilde{W}_{\mathrm{ua}} + \frac{1}{2\gamma^2}\Phi_{\mathrm{ua}}\left(k^{\mathrm{T}}(e)\nabla\Phi^{\mathrm{T}}\tilde{W}_c\right) \\
& - \frac{1}{2\gamma^2}\Phi_{\mathrm{ua}}\left(k^{\mathrm{T}}(e)\nabla\epsilon_c\right) - \Phi_{\mathrm{ua}}\epsilon_{\mathrm{ua}}\Big), \ t \geq 0. \quad (68)
\end{aligned}$$

For clarity, we will separate the following terms of equation (68)

$$\begin{aligned}
T_1 = & -\frac{\partial V_c}{\partial\tilde{W}_c}^{\mathrm{T}}\left(\frac{\varrho(t)\varrho(t)^{\mathrm{T}}}{(\varrho(t)^{\mathrm{T}}\varrho(t)+1)^2}\right. \\
& \left. + \sum_{i=1}^{k}\frac{\varrho(t_i)\varrho(t_i)^{\mathrm{T}}}{(\varrho(t_i)^{\mathrm{T}}\varrho(t_i)+1)^2}\right)\tilde{W}_c \\
& + \frac{\partial V_c}{\partial\tilde{W}_c}^{\mathrm{T}}\left(\frac{\varrho(t)}{(\varrho(t)^{\mathrm{T}}\varrho(t)+1)^2}\epsilon_H(t)\right. \\
& \left. + \sum_{i=1}^{k}\frac{\varrho(t_i)}{(\varrho(t_i)^{\mathrm{T}}\varrho(t_i)+1)^2}\epsilon_H(t_i)\right) \\
& \leq -2\alpha\lambda_{\min}\left(\sum_{i=1}^{k}\frac{\varrho(t_i)\varrho(t_i)^{\mathrm{T}}}{(\varrho(t_i)^{\mathrm{T}}\varrho(t_i)+1)^2}\right)\left\|\tilde{W}_c\right\|^2
\end{aligned}$$

$$
\begin{aligned}
&+\frac{1}{2\alpha}\left\|\tilde{W}_c\right\| 2\varrho_{\max}\epsilon_{H\max}\\
&\leq -2\alpha\lambda_{\min}\left(\sum_{i=1}^{k}\frac{\varrho(t_i)\varrho(t_i)^{\mathrm{T}}}{(\varrho(t_i)^{\mathrm{T}}\varrho(t_i)+1)^2}\right)\left\|\tilde{W}_c\right\|^2\\
&+\frac{1}{4\alpha}\left(2\varrho_{\max}\epsilon_{H\max}\right)^2, \qquad (69)
\end{aligned}
$$

$$
\begin{aligned}
T_2 =&\mathrm{tr}\Bigg\{\tilde{W}_{\mathrm{uo}}^{\mathrm{T}}\Bigg(-\Phi_{\mathrm{uo}}\Phi_{\mathrm{uo}}^{\mathrm{T}}\tilde{W}_{\mathrm{uo}}\\
&-\Phi_{\mathrm{uo}}\mathrm{sat}\left(\frac{1}{2}R^{-1}g^{\mathrm{T}}(e)\nabla\Phi^{\mathrm{T}}\tilde{W}_c\right)^{\mathrm{T}}\\
&-\Phi_{\mathrm{uo}}\mathrm{sat}\left(\frac{1}{2}R^{-1}g^{\mathrm{T}}(e)\nabla\epsilon_c\right)^{\mathrm{T}}-\Phi_{\mathrm{uo}}\epsilon_{\mathrm{uo}}\Bigg)\Bigg\}\\
&\leq -\Phi_{uo\max}^2\left\|\tilde{W}_{\mathrm{uo}}\right\|^2-\Phi_{uo\max}\bar{u}_o\left\|\tilde{W}_{\mathrm{uo}}\right\|\\
&-\left(\Phi_{uo\max}\bar{u}_o+\Phi_{uo\max}\epsilon_{uo\max}\right)\left\|\tilde{W}_{\mathrm{uo}}\right\|\\
&\leq -\Phi_{uo\max}^2\left\|\tilde{W}_{\mathrm{uo}}\right\|^2+\Phi_{uo\max}\bar{u}_o\left\|\tilde{W}_{\mathrm{uo}}\right\|^2\\
&+\frac{\left(\Phi_{uo\max}\bar{u}_o+\Phi_{uo\max}\epsilon_{uo\max}\right)^2}{2}+\frac{1}{2}\left\|\tilde{W}_{\mathrm{uo}}\right\|^2 \qquad (70)
\end{aligned}
$$

$$
\begin{aligned}
T_3 =&\mathrm{tr}\Bigg\{\tilde{W}_{\mathrm{ua}}^{\mathrm{T}}\Bigg(-\Phi_{\mathrm{ua}}\Phi_{\mathrm{ua}}^{\mathrm{T}}\tilde{W}_{\mathrm{ua}}+\frac{1}{2\gamma^2}\Phi_{\mathrm{ua}}\left(k^{\mathrm{T}}(e)\nabla\Phi_c^{\mathrm{T}}\tilde{W}_c\right)^{\mathrm{T}}\\
&-\frac{1}{2\gamma^2}\Phi_{\mathrm{ua}}\left(k^{\mathrm{T}}(e)\nabla\epsilon_c\right)^{\mathrm{T}}-\Phi_{\mathrm{ua}}\epsilon_{\mathrm{ua}}\Bigg)\Bigg\}\\
\leq&-\Phi_{ua\max}^2\left\|\tilde{W}_{\mathrm{ua}}\right\|^2\\
&+\frac{1}{2\gamma^2}\Phi_{ua\max}\mathrm{k}_{\max}\Phi_{cmax}\left\|\tilde{W}_c\right\|\left\|\tilde{W}_{\mathrm{ua}}\right\|\\
&-\left(\frac{1}{2\gamma^2}\Phi_{ua\max}\mathrm{k}_{\max}\epsilon_{\mathrm{cdmax}}+\Phi_{ua\max}\epsilon_{ua\max}\right)\left\|\tilde{W}_{\mathrm{ua}}\right\|\\
\leq&-\Phi_{ua\max}^2\left\|\tilde{W}_{\mathrm{ua}}\right\|^2\\
&+\frac{1}{4\gamma^2}\Phi_{ua\max}\mathrm{k}_{\max}\Phi_{cmax}\left(\left\|\tilde{W}_c\right\|^2+\left\|\tilde{W}_{\mathrm{ua}}\right\|^2\right)\\
&+\frac{\left(\frac{1}{2\gamma^2}\Phi_{ua\max}\mathrm{k}_{\max}\epsilon_{\mathrm{cdmax}}+\Phi_{ua\max}\epsilon_{ua\max}\right)^2}{2}\\
&+\frac{1}{2}\left\|\tilde{W}_{\mathrm{ua}}\right\|^2. \qquad (71)
\end{aligned}
$$

$$
\begin{aligned}
T_4 =&C_e^{*\mathrm{T}}\Bigg(f(e)-g(e)\tilde{W}_{\mathrm{uo}}^{\mathrm{T}}\Phi_{\mathrm{uo}}+g(e)(u_o^*-\epsilon_{\mathrm{uo}})\\
&-g(e)B\frac{e^{\mathrm{T}}e\mathbf{1}_m}{(A+e^{\mathrm{T}}e)}-k(e)\tilde{W}_{\mathrm{ua}}^{\mathrm{T}}\Phi_{\mathrm{ua}}+k(e)(u_a^*-\epsilon_{\mathrm{ua}})\\
&-k(e)D\frac{e^{\mathrm{T}}e\mathbf{1}_m}{(A+e^{\mathrm{T}}e)}\Bigg). \qquad (72)
\end{aligned}
$$

In equation (69), we used the results from Theorem 1 and $\frac{\partial C_e}{\partial W_c}\leq\frac{1}{2\alpha}\left\|\tilde{W}_c\right\|$. Using the HJI equation

$$
\begin{aligned}
C_e^{*\mathrm{T}}f(e) =&-C_e^{*\mathrm{T}}g(e)u_o^*-R_s(u_o^*)-Q(e)\\
&-C_e^{*\mathrm{T}}k(e)u_a^*+\gamma^2\left\|u_a^*\right\|^2, \quad \forall\ e
\end{aligned}
$$

in equation (72) yields

$$
\begin{aligned}
T_4 =&-R_s(u_o^*)-Q(e)\\
&-C_e^{*\mathrm{T}}\left(g(e)\tilde{W}_{\mathrm{uo}}^{\mathrm{T}}\Phi_{\mathrm{uo}}+g(e)\epsilon_{\mathrm{uo}}+g(e)B\frac{e^{\mathrm{T}}e\mathbf{1}_m}{(A+e^{\mathrm{T}}e)}\right)\\
&+\gamma^2\left\|u_a^*\right\|^2\\
&-C_e^{*\mathrm{T}}\left(k(e)\tilde{W}_{\mathrm{ua}}^{\mathrm{T}}\Phi_{\mathrm{ua}}+k(e)\epsilon_{\mathrm{ua}}+k(e)D\frac{e^{\mathrm{T}}e\mathbf{1}_m}{(A+e^{\mathrm{T}}e)}\right)\\
\leq&-R_s(u_o^*)-Q(e)\\
&-(W_{\mathrm{cmax}}\Phi_{\mathrm{cdmax}}+\epsilon_{\mathrm{cdmax}})g_{\max}\Phi_{uo\max}\left\|\tilde{W}_{\mathrm{uo}}\right\|\\
&-(W_{\mathrm{cmax}}\Phi_{\mathrm{cdmax}}+\epsilon_{\mathrm{cdmax}})\epsilon_{uo\max}\\
&-(W_{\mathrm{cmax}}\Phi_{\mathrm{cdmax}}+\epsilon_{\mathrm{cdmax}})g_{\max}B\frac{e^{\mathrm{T}}e\mathbf{1}_m}{A+e^{\mathrm{T}}e}\\
&+\gamma^2\left\|u_a^*\right\|^2\\
&-(W_{\mathrm{cmax}}\Phi_{\mathrm{cdmax}}+\epsilon_{\mathrm{cdmax}})k_{\max}\Phi_{ua\max}\left\|\tilde{W}_{\mathrm{ua}}\right\|\\
&-(W_{\mathrm{cmax}}\Phi_{\mathrm{cdmax}}+\epsilon_{\mathrm{cdmax}})\epsilon_{ua\max}\\
&-(W_{\mathrm{cmax}}\Phi_{\mathrm{cdmax}}+\epsilon_{\mathrm{cdmax}})k_{\max}D\frac{e^{\mathrm{T}}e\mathbf{1}_m}{A+e^{\mathrm{T}}e} \qquad (73)
\end{aligned}
$$

since $A+e^{\mathrm{T}}e>0$. Now, T_4 can be upper bounded as

$$
\begin{aligned}
T_4\leq&-R_s(u_o^*)-Q(e)\\
&+\frac{g_{\max}\Phi_{uo\max}}{2}\left((W_{\mathrm{cmax}}\Phi_{\mathrm{cdmax}}+\epsilon_{\mathrm{cdmax}})\right)^2\\
&+\frac{g_{\max}\Phi_{uo\max}}{2}\left\|\tilde{W}_{\mathrm{uo}}\right\|^2\\
&+\frac{1}{2}(W_{\mathrm{cmax}}\Phi_{\mathrm{cdmax}}+\epsilon_{\mathrm{cdmax}})^2+\frac{1}{2}\epsilon_{uo\max}^2\\
&-(W_{\mathrm{cmax}}\Phi_{\mathrm{cdmax}}+\epsilon_{\mathrm{cdmax}})g_{\max}B\frac{e^{\mathrm{T}}e\mathbf{1}_m}{A+e^{\mathrm{T}}e}+\gamma^2\left\|u_a^*\right\|^2\\
&+\frac{k_{\max}\Phi_{ua\max}}{2}\left((W_{\mathrm{cmax}}\Phi_{\mathrm{cdmax}}+\epsilon_{\mathrm{cdmax}})\right)^2\\
&+\frac{k_{\max}\Phi_{ua\max}}{2}\left\|\tilde{W}_{\mathrm{ua}}\right\|^2\\
&+\frac{1}{2}(W_{\mathrm{cmax}}\Phi_{\mathrm{cdmax}}+\epsilon_{\mathrm{cdmax}})^2+\frac{1}{2}\epsilon_{ua\max}^2\\
&-(W_{\mathrm{cmax}}\Phi_{\mathrm{cdmax}}+\epsilon_{\mathrm{cdmax}})k_{\max}D\frac{e^{\mathrm{T}}e\mathbf{1}_m}{A+e^{\mathrm{T}}e}. \qquad (74)
\end{aligned}
$$

After considering the bounds of B and D from inequalities in equations (53) and (54), respectively, an upper bound for equation (68) can be obtained as

$$
\begin{aligned}
\dot{\mathcal{V}}\leq&-\Bigg(2\alpha\lambda_{\min}\left(\sum_{i=1}^{k}\frac{\varrho(t_i)\varrho(t_i)^{\mathrm{T}}}{(\varrho(t_i)^{\mathrm{T}}\varrho(t_i)+1)^2}\right)-\frac{1}{4\alpha}\\
&-\frac{1}{4\gamma^2}\Phi_{ua\max}\mathrm{k}_{\max}\Phi_{cmax}\Bigg)\left\|\tilde{W}_c\right\|^2\\
&-\left(\Phi_{uo\max}^2-\Phi_{uo\max}\bar{u}_o-\frac{1}{2}-\frac{g_{\max}\Phi_{uo\max}}{2}\right)\left\|\tilde{W}_{\mathrm{uo}}\right\|^2\\
&-R_s(u_o^*)-Q(e)\\
&-\Bigg(\Phi_{ua\max}^2-\frac{1}{2}-\frac{k_{\max}\Phi_{ua\max}}{2}\\
&-\frac{1}{4\gamma^2}\Phi_{ua\max}\mathrm{k}_{\max}\Phi_{cmax}\Bigg)\left\|\tilde{W}_{\mathrm{ua}}\right\|^2
\end{aligned}
$$

$$+ \gamma^2 \left\| u_a^* \right\|^2, \ t \geq 0. \tag{75}$$

Considering the inequalities in equations (56), (57), (58), and the HJI inequality in equation (62), we have

$$\dot{\mathcal{V}} \leq 0, \ t \geq 0.$$

It follows from Barbalat's lemma [48] that as $t \to \infty$, then $\|e\| \to 0$, $\left\|\tilde{W}_c\right\| \to 0$, $\left\|\tilde{W}_{\text{uo}}\right\| \to 0$, and $\left\|\tilde{W}_{\text{ua}}\right\| \to 0$. Finally since $\|e\| \to 0$, it is straightforward that, as $t \to \infty$ then $\frac{e^{\mathrm{T}} e}{\left(A + e^{\mathrm{T}} e\right)} \to 0$ and hence $\|\hat{u}_o\| \to u_o^*$, $\|\hat{u}_a\| \to u_a^*$. ■

Remark 2: Note that the actual controller u_V in equation (13) is finally given as $u_V = u_o + u_d$. □

REFERENCES

[1] Garcia Carrillo, L.R., Dzul, A., and Lozano, R. "Hovering quad-rotor control: A comparison of nonlinear controllers using visual feedback". *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, issue 4, pp. 3159-3170, October 2012.

[2] Garcia Carrillo, L.R., Fantoni, I., Rondon, E., and Dzul, A. "3-dimensional Position and Velocity Regulation of a Quad-rotor Using Optical Flow". *IEEE Transactions on Aerospace and Electronic Systems*, vol. 52, issue 1, pp. 358-371, January 2015.

[3] Bertsekas, D.P., and Tsitsiklis, J.N. "Neuro-Dynamic Programming", Athena Scientific, MA, 1996.

[4] Sutton, R.S., and Barto, A.G., "Reinforcement learning - an introduction", MIT Press, Cambridge, MA, 1998.

[5] Abu-Khalaf, M., and Lewis, F.L., "Neuro Dynamic Programming and Zero-Sum Games for Constrained Control Systems", *IEEE Transactions on Neural Networks*, vol. 19, issue 7, pp. 1243-1252, July 2008.

[6] Vamvoudakis, K.G., and Lewis, F.L. "Multi-player non-zero-sum games: online adaptive learning solution of coupled Hamilton-Jacobi equations", *Automatica*, vol. 47, issue 8, pp. 1556-1569, October 2011.

[7] Bhasin, S., Kamalapurkar, R., Johnson, M., Vamvoudakis, K.G., Lewis, F.L., and Dixon, W.E. "A novel actor critic identifier architecture for approximate optimal control of uncertain nonlinear systems", *Automatica*, vol. 49, issue 1, pp. 82-92, January 2013.

[8] Chowdhary, G., and Johnson, E. "Concurrent learning for convergence in adaptive control without persistency of excitation", *Conference on Decision and Control*, pp. 3674-3679, Atlanta, GA, December 2010.

[9] Hoffmann, G.M., Waslander, S.L., and Tomlin, C.J. "Quadrotor helicopter trajectory tracking control", *AIAA Guidance, Navigation, and Control Conference*, Honolulu, HI, August 2008.

[10] Aguiar, A.P., and Hespanha, J.P. "Trajectory-tracking and path-following of underactuated autonomous vehicles with parametric modeling uncertainty", *IEEE Transactions on Automatic Control*, vol. 52, issue 8, pp 1362-137, August 2007.

[11] Lee, D., Burg, T.C., Xian, B., and Dawson, D.M. "Output feedback tracking control of an underactuated quad-Rotor UAV", *American Control Conference*, pp. 1775-1780, New York, NY, July 2007.

[12] Garcia Carrillo, L.R., Flores, G., Sanahuja, G., and Lozano, R. "Quad Rotorcraft Switching Control: An Application for the Task of Path Following", *IEEE Transactions on Control Systems Technology*, pp. 1255-1267, October 2013.

[13] Abdessameud, A., and Tayebi, A. "Global trajectory tracking control of VTOL-UAVs without linear velocity measurements", *Automatica*, vol. 46, issue 6, pp. 1053-1059, June 2010.

[14] Dierks, T., and Jagannathan, S. "Output Feedback Control of a Quadrotor UAV Using Neural Networks", *IEEE Transactions on Neural Networks*, vol. 21, issue 1, pp. 50-66, January 2010.

[15] Nodland, D., Zargarzadeh, H., and Jagannathan, S. "Neural network-based optimal control for trajectory tracking of a helicopter UAV", *IEEE Conference on Decision and Control and European Control Conference*, pp. 3876-3881, Orlando FL, December 2011.

[16] Imanberdiyev, N., Fu, C., Kayacan, E., and Chen, I.M. "Autonomous Navigation of UAV by Using Real-Time Model-Based Reinforcement Learning", 14th International Conference on Control, Automation, Robotics & Vision (ICARCV 2016), Phuket, Thailand, 13-15th November 2016

[17] Hinton, G., and Salakhutdinov, R. "Reducing the Dimensionality of Data with Neural Networks", *Science*, vol. 313, issue 5786, pp. 504-507, 2006.

[18] Chu, C.T., Kim, S.K., Lin, Y.A., Yu, Y.Y., Bradski, G., Ng, A.Y., and Olukotun, K. "Map-reduce for Machine Learning on Multicore", *International Conference on Neural Information Processing Systems*, pp. 281-288, Canada, December 2006.

[19] Panda, B., Herbach, J.S., Basu, S., and Bayardo, R.J. "MapReduce and Its Application to Massively Parallel Learning of Decision Tree Ensembles", *Scaling up Machine Learning: Parallel and Distributed Approaches*, pp. 23-48, Cambridge University Press, 2011.

[20] Bengio, Y., and Bengio, S. "Modeling high-dimensional discrete data with multi-layer neural networks", *Advances in Neural Information Processing Systems*, pp. 400-406, 2000.

[21] Munk, J., Kober, J., and Babuska, R. "Learning state representation for deep actor-critic control", *Conference on Decision and Control (CDC)*, Las Vegas, NV, USA, December 2016.

[22] De Bruin, T., Kober, J., Tuyls, K., and Babuska, R. "Improved deep reinforcement learning for robotics through distribution-based experience retention", *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3947-3952, Daejeon, Korea, October 2016.

[23] Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., and Hassabis, D. "Human-level control through deep reinforcement learning", *Nature*, vol. 518, issue 7540, pp. 529533, 2015.

[24] Liu, R., and Zou, J. "The Effects of Memory Replay in Reinforcement Learning", ICML 2017 Workshop on Principled Approaches to Deep Learning, Sydney, Australia, August 2017.

[25] Mnih, V., Badia, A.P., Mirza, M., Graves, A., Lillicrap, T.P., Harley, T., Silver, D., and Kavukcuoglu, K. "Asynchronous methods for deep reinforcement learning", *International Conference on Machine Learning*, New York, USA, June 2016.

[26] Van Hasselt, H., Guez, A., and Silver, D."Deep Reinforcement Learning with Double Q-learning", *AAAI Conference on Artificial Intelligence (AAAI)*, Phoenix, Arizona USA, February 2016.

[27] Levine, S., Finn, C., Darrell, T., and Abbeel, P. "End-to-End Training of Deep Visuomotor Policies", *Journal of Machine Learning Research*, vol. 17, issue 39, pp. 1-40, 2016.

[28] Schulman, J., Levine, S., Moritz, P., Jordan, M.I., and Abbeel, P. "Trust Region Policy Optimization", *International Conference on Machine Learning*, JMLR: W&CP volume 37, Lille, France, 2015.

[29] Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., and Riedmiller, M.A. "Playing Atari with Deep Reinforcement Learning", *NIPS DL Workshop*, Lake Tahoe, USA, December 2013.

[30] Guo, X., Singh, S., Lee, H., Lewis, R., and Wang, X. "Deep learning for real-time Atari game play using offline Monte-Carlo tree search planning", *In Advances in Neural Information Processing Systems (NIPS)*, 27, vol. 4, pp. 3338-3346, Montreal, Quebec, Canada, December 2014.

[31] Schulman, J., Moritz, P., Levine, S., Jordan, M.I., and Abbeel, P. "High-Dimensional Continuous Control Using Generalized Advantage Estimation", *International Conference of Learning Representations (ICLR)*, San Juan, Puerto Rico, May 2016.

[32] Silver, D., Huang, A., Maddison, C.J., Guez, A., Sifre, L., Van den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T., and Hassabis, D. "Mastering the game of Go with deep neural networks and tree search", *Nature*, volume 529, pp. 484-489, January 2016.

[33] Parisotto, E., Ba, L.J., and Salakhutdinov, R. "Actor-Mimic: Deep Multitask and Transfer Reinforcement Learning", *International Conference on Learning Representations*, San Juan, Puerto Rico, May 2016.

[34] Zhu, Y., Mottaghi, R., Kolve, E., Lim, J., Gupta, A., Fei-Fei, L., and Farhadi, A. "Target-driven Visual Navigation in Indoor Scenes using Deep Reinforcement Learning", *Int. Conference on Robotics and Automation*, pp. 3357-3364, Marina Bay Sands, Singapore, June 2017.

[35] KVamvoudakis, K.G., and Lewis, F.L. "Online solution of nonlinear twoplayer zerosum games using synchronous policy iteration", International Journal of Robust and Nonlinear Control, vol. 22, issue 13, pp. 1460-1483, September 2012

[36] Garcia Carrillo, L.R., Dzul, A., Lozano, R., and Pegard, C. "Quad Rotorcraft Control: Vision-Based Hovering and Navigation", *Advances in Industrial Control*, Springer, 2013.

[37] Voos, H. "Nonlinear State-Dependent Riccati Equation Control of a Quadrotor UAV", *IEEE International Conference on Control Applications*, Munich, Germany, October 2006.

[38] Goldstein, H. "Classical Mechanics", *Addison Wesley*, 2nd Ed., 1980.

[39] Koo, T.J., and Sastry, S. "Output tracking control design of a helicopter model based on approximate linearization", *IEEE Conference on Decision and Control*, Tampa, FL, USA, December 1998.
[40] Van Der Schaft, A.J. "L2-gain analysis of nonlinear systems and nonlinear state-feedback H-∞ control", *IEEE Transactions on Automatic Control*, vol. 37, issue 6, pp. 770-784, June 1992.
[41] Abu-Khalaf, M., and Lewis, F.L. "Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach" *Automatica*, vol. 41, issue 5, pp. 779-791, May 2005.
[42] Lyshevski, S.E. "Optimal control of nonlinear continuous-time systems: design of bounded controllers via generalized nonquadratic functionals", *American Control Conf.*, pp 205-209, Philadelphia, USA, June 1998.
[43] Igelnik, B., and Pao, Y.H. "Stochastic choice of basis functions in adaptive function approximation and the functional-link net"; IEEE Trans. Neural Netw., vol. 6, no. 6, pp. 1320-1329, Nov. 1995.
[44] Ioannou, P, and Fidan, B. "Adaptive control tutorial", *Advances in Design and Control*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 2006.
[45] Abu-Khalaf, M., Lewis, F.L., and Huang, J. "Policy Iterations and the Hamilton-Jacobi-Isaacs Equation for H-infinity state feedback control with input saturation", *IEEE Transactions on Automatic Control*, vol. 51, issue 12, pp. 1989-1995, December 2006.
[46] Hornik, K., Stinchcombe, M., and White, H. "Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks", *Neural Networks*, vol. 3, issue 5, pp. 551-560, January 1990.
[47] Chowdhary, G., Yucelen, T., Mühlegg, M., and Johnson, E.N. "Concurrent learning adaptive control of linear systems with exponentially convergent bounds", *International Journal of Adaptive Control Signal Process*, vol. 27, issue 4, pp. 280-301, April 2013.
[48] Haddad, W.M., and Chellaboina, V.S. "Nonlinear Dynamical Systems and Control: A Lyapunov-Based Approach", *Princeton Univ. Press*, 2008.
[49] Bristeau, P.J., Callou, F., Vissiere, D., and Petit, N. "The Navigation and Control technology inside the AR.Drone micro UAV", IFAC Proceedings Volumes, Volume 44, Issue 1, January 2011, Pages 1477-1484.

Luis Rodolfo Garcia Carrillo (M'13) was born in Gomez Palacio, Durango, Mexico. He received his B.S. in Electronic Engineering in 2003, and his M.S. in Electrical Engineering in 2007, both from the Institute of Technology of La Laguna, Coahuila, Mexico. He received his Ph.D. in Control Systems from the University of Technology of Compiegne, France, in 2011. From 2012 to 2013, he was a postdoctoral researcher at the Center for Control, Dynamical Systems and Computation at the University of California, Santa Barbara (UCSB). During this time he was also a researcher at the UCSB Institute for Collaborative Biotechnologies.

Dr. Garcia Carrillo is now an assistant professor at the department of Electrical Engineering at Texas A&M University - Corpus Christi. He is a coauthor of one patent, more than 65 technical publications, and one book. He currently is an associate editor of Mathematical Problems in Engineering. His research interests have focused on control systems, multi-agent systems, intelligent control, and the use of computer vision in feedback control.

Kyriakos G. Vamvoudakis (SM'15) was born in Athens, Greece. He received the Diploma (a 5 year degree, equivalent to a Master of Science) in Electronic and Computer Engineering from Technical University of Crete, Greece in 2006 with highest honors. After moving to the United States of America, he studied at The University of Texas and he received his M.S. and Ph.D. in Electrical Engineering in 2008 and 2011 respectively. During the period from 2012 to 2016 he was a project research scientist at the Center for Control, Dynamical Systems and Computation at the University of California, Santa Barbara. He was an assistant professor at the Kevin T. Crofton Department of Aerospace and Ocean Engineering at Virginia Tech until 2018. He is now an assistant professor at The Daniel Guggenheim School of Aerospace Engineering at Georgia Tech. His research interests include optimal control, reinforcement learning, and game theory. Recently, his research has focused on cyber-physical security, and safe autonomy. Prof. Vamvoudakis is the recipient of a 2019 ARO YIP award, a 2018 National Science Foundation CAREER Award, and of several international awards including, the 2016 International Neural Network Society Young Investigator (INNS) Award, the Best Paper Award for Autonomous/Unmanned Vehicles at the 27th Army Science Conference in 2010, the Best Presentation Award at the World Congress of Computational Intelligence in 2010, and the Best Researcher Award from the Automation and Robotics Research Institute in 2011. He is a coauthor of one patent, more than 130 technical publications, and two books. He is the General Chair, of the 9th International Conference on the Internet of Things (IoT 2019). He currently is an associate editor of Automatica, an associate editor of the IEEE Control Systems Letters, an associate editor of the IEEE Computational Intelligent Magazine, an associate editor of the Journal of Optimization Theory and Applications, an associate editor of Control Theory and Technology, a member of the IEEE Control Systems Society Conference Editorial Board, a registered electrical/computer engineer (PE) and a member of the Technical Chamber of Greece.