Biologically plausible robust control with Neural Network weight reset for unmanned aircraft systems under impulsive disturbances

Ignacio Rubio Scola, Luis Rodolfo Garcia Carrillo, *Member, IEEE*,
Andrew T. Sornborger, and João P. Hespanha, *Fellow, IEEE*

***Abstract*— Self-learning control techniques mimicking the functionality of the limbic system in the mammalian brain have shown advantages in terms of superior learning ability and low computational cost. However, accompanying stability analyses and mathematical proofs rely on unrealistic assumptions which limit not only the performance, but also the implementation of such controllers in real-world scenarios. In this work the limbic system inspired control (LISIC) framework is revisited, introducing three contributions that facilitate the implementation of this type of controller in real-time. First, an extension enabling the implementation of LISIC to the domain of SISO affine systems is proposed. Second, a strategy for resetting the controller's Neural Network (NN) weights is developed, in such a way that now it is possible to deal with piece-wise smooth references and impulsive perturbations. And third, for the case when a nominal model of the system is available, a technique is proposed to compute a set of optimal NN reset weight values by solving a convex constrained optimization problem. Numerical simulations addressing the stabilization of an unmanned aircraft system via the robust LISIC demonstrate the advantages obtained when adopting the extension to SISO systems and the two NN weight reset strategies.**

I. Introduction

The dynamics of unmanned aircraft systems (UASs) are generally represented using a nonlinear mathematical model described by a set of first order differential equations. For UAS control design purposes, feedback linearization is a well known technique which, under the implementation of an appropriate feedback controller, renders the input-output dynamics of a nonlinear plant linear. Once a linearizing controller has been constructed, desired output trajectories for the nonlinear plant can be tracked using a variety of linear control techniques. However, the calculation of a linearizing controller requires a precise knowledge of the nonlinear dynamic model of the system, which are usually not available or disclosed by the manufacturers of this kind of system.

Related Work

In practice, a dynamic model can be completely known a priori as a result of offline identification techniques. In [1], the authors addressed theoretical and practical problems associated with autonomously controlling a commercially–available and ready–to–fly UAS prototype whose dynamic model and controller running on onboard its autopilot are not disclosed. The first step towards developing a controller was the system identification of the UAS dynamics. This was possible making use of the Extended Least Squares (ELS) algorithm in combination with input-output data pairs obtained during several real-time experimental flights.

In some cases the UAS dynamic model is only partially known or identified, and under such circumstances the problem simplifies to the estimation of the unknown components, which most of the time can be performed online. However, the identified model can turn unreliable due to the presence of external perturbations, which can make its way into the system from different sources, for example, from sensing devices used by the agent to obtain pose information [2], [3]. These sensor modalities typically exhibit impulsive disturbances in the sensory data: vision-based sensors can exhibit large errors when a visual landmark is temporarily obstructed or misinterpreted, and RF/acoustic sensors are prone to reporting false measurements due to multi-path reflections. Impulsive disturbances arise also for autonomous agents that generally move along fairly smooth paths but occasionally engage in sharp turns or evasive maneuvers [4]. The challenge to overcome for a UAS operating in real-time conditions is then the lack of knowledge of the state-dependent functions and the presence of unknown external perturbations.

In [5], a limbic system-inspired control (LISIC) strategy was proposed to estimate the state-dependent functions of a dynamic model using a biologically-inspired estimation and control algorithm based on a learning mechanism encountered in the limbic system of the human brain. Making use of a Lyapunov analysis, it was demonstrated that the proposed strategy is capable of guaranteeing a specific level of performance even in the presence of unknown external perturbations. Despite its many advantages, two major assumptions required in the stability analysis and mathematical proofs of LISIC can limit its real-world implementation. First, when dealing with UAS lateral and longitudinal dynamics, like those identified via the ELS algorithm in [1], tools to address affine state systems are needed, which are not directly accommodated by the LISIC theorem assumptions. Second, a UAS operating in realistic conditions is often subjected to piece-wise smooth references, and impulsive perturbations are common on the signals coming from its positioning

I. Rubio Scola is with the Department of Industrial Products Engineering INTI, CONICET and UNR, Rosario, Argentina. e-mail: irubio@inti.gob.ar

L.R. Garcia Carrillo is with the Klipsch School of Electrical and Computer Engineering, New Mexico State University, Las Cruces, NM, USA. e-mail: luisillo@nmsu.edu

J.P. Hespanha is with Center for Control, Dynamical Systems, and Computation, University of California, Santa Barbara, USA. e-mail: hespanha@ece.ucsb.edu

Andrew T. Sornborger is a research scientist at Los Alamos National Laboratory, Los Alamos, NM, USA. e-mail: sornborg@lanl.gov

sensors. These two conditions are also not compatible with the original assumptions of the LISIC stability theorem.

Main Contributions

This work extends the original LISIC strategy in three fundamental ways, enabling its implementation over systems with unknown dynamics subjected to bounded external perturbations of L_1 type. With these improvements, LISIC becomes functional over dynamic models commonly associated with UAS operating under realistic, uncertain conditions. First, a technique to extend the use of LISIC into SISO affine systems is introduced. Second, a strategy for resetting Neural Network (NN) weights to zero is proposed, allowing LISIC to operate under fully unknown dynamics, impulsive disturbances, and piece-wise smooth references. And third, if a nominal UAS dynamic model is available, a strategy for resetting NN weights to optimal values is proposed, which turn out to be very similar to the NN weights obtained if the real dynamic model would be available.

The rest of the manuscript is organized as follows. Section II introduces the problem statement. Next, Section III presents our main result in the form of a Theorem. Section IV demonstrates the performance of the proposed strategy making use of numerical simulations. Finally, conclusions and future directions are discussed in Section V

II. Problem Statement

Consider the dynamics of a class of nonlinear systems of order n, described by

$$x^{(n)} = f(\underline{x}) + g(\underline{x})u + d(\underline{x}, t) \tag{1}$$

where $\underline{x} = [x, \dot{x}, \ldots, x^{(n-1)}]^\top \in \mathbb{R}^n$ is the state vector, $\dot{x}$ is the derivative of x with respect to (w.r.t.) time, $x^{(n-1)}$ is the $(n-1)^{\text{th}}$ order derivative of x w.r.t. time, $u \in \mathbb{R}$ the control input, and $d(\underline{x}, t) \in \mathbb{R}$ a perturbation.

Assumption 1: the state vector $\underline{x}$ and the perturbation $d(\underline{x}, t)$ are bounded by known positive constants $\|\underline{x}\| \leq M_x$ and $\|d(\underline{x}, t)\| \leq M_d$, respectively.

Assumption 2: the function $g(\underline{x}) > 0$, and $1/g(\underline{x})$ and $f(\underline{x})$ are unknown continuous scalar functions.

Assumption 3: the desired trajectory x_d and its derivatives, up to its n^{th} order derivative, are smooth and bounded.

Defining a tracking error $e = x - x_d$, an auxiliary variable s is proposed depending on the system's tracking error and its derivatives as

$$s = e^{(n-1)} + \Delta_{n-1} e^{(n-2)} + \ldots + \Delta_1 e \tag{2}$$

where the terms Δ_k $(k = 1, 2, \ldots, n-1)$ represent constants such that the roots of the polynomial $\lambda^{n-1} + \Delta_{n-1}\lambda^{n-2} + \ldots + \Delta_1 = 0$ have negative real part. The derivative of the auxiliary variable s is calculated as

$$\dot{s} = f(\underline{x}) + g(\underline{x})u + q_a(t) + d(\underline{x}, t) \tag{3}$$

with $q_a = -x_d^{(n)} + \Delta_{n-1} e^{(n-1)} + \ldots + \Delta_1 \dot{e}$.

For the scenario where the functions $f(\underline{x})$ and $g(\underline{x})$ are known and $d(\underline{x}, t) = 0$, it is possible to achieve the dynamics $\dot{s} = -Ks + u_r$ with the following exact matching control law

$$u^* = -(f(\underline{x}) + q_a + Ks - u_r)/g(\underline{x}), \tag{4}$$

where the term u_r is an auxiliary input to be specified next.

In [5] [6], the authors proposed the incorporation of an auxiliary state $\xi(t) = \int s(t)dt$ which augments the system and improves performance through an integral action

$$\begin{bmatrix} \dot{s} \\ \dot{\xi} \end{bmatrix} = \begin{bmatrix} -K & 0 \\ K_I & 0 \end{bmatrix} \begin{bmatrix} s \\ \xi \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u_r = A_e s_e + B_e u_r \tag{5}$$

For this case, the auxiliary input term u_r can be obtained by solving the following Ricatti equation

$$0 = A_e^\top P_e + P_e A_e - P_e B_e R^{-1} B_e^\top P_e + Q_e \tag{6}$$

$$u_r = -B_e^\top T P_e s_e / r \ , \tag{7}$$

where $s_e = [s \ \xi]^\top$ and $R = \rho^2 r/(2\rho^2 - r)$, with tuning parameters $Q_e = Q_e^\top \succ 0$, $r > 0$ and $2\rho^2 > r$.

Remark 1: the implementation of the control law in equation (4) would require precise knowledge of the unknown functions $f(\underline{x})$ and $g(\underline{x})$. To overcome this challenge, it is required to construct online estimates $\hat{f}(\underline{x})$ and $\hat{h}(\underline{x})$ of the functions $f(\underline{x})$ and $h(\underline{x}) := 1/g(\underline{x})$, respectively, that appear in the control law in equation (4).

Estimates of the functions $\hat{f}(\underline{x})$ and $\hat{h}(\underline{x})$ can be built using a combination of Gaussian Radial Basis Functions (RBF) that emulates the emotional learning structure of the mammalian limbic system originally proposed in [7]:

$$\begin{aligned} \hat{f}(\underline{x}) :=& \hat{f}(\underline{x}, V_f, W_f) = V_f^\top \Phi_A(s(\underline{x})) - W_f^\top \Phi(s(\underline{x})) \\ \hat{h}(\underline{x}) :=& \hat{h}(\underline{x}, V_h, W_h) = V_h^\top \Phi_A(s(\underline{x})) - W_h^\top \Phi(s(\underline{x})) \end{aligned} \tag{8}$$

where the terms

$$\begin{aligned} &V_f = [V_{f1}, \ldots, V_{fp}, V_{f\text{th}}]^\top, W_f = [W_{f1}, W_{f2}, \ldots, W_{fp}]^\top, \\ &V_h = [V_{h1}, \ldots, V_{hp}, V_{h\text{th}}]^\top, W_h = [W_{h1}, W_{h2}, \ldots, W_{hp}]^\top \end{aligned}$$

are vectors of weight parameters and the terms Φ_j are Gaussian RBFs that can be represented using the structure

$$\Phi_j = \exp\left(-(s - \mu_j)^2/\sigma_j^2\right), \quad m = \max(\Phi) \tag{9}$$

where s is the error dynamics described by equation (2), and μ_j and σ_j are the corresponding mean and smoothing factors, respectively. The RBFs are $\Phi = [\Phi_1, \Phi_2, \ldots, \Phi_p]^\top$ and $\Phi_A = [\Phi, m]^\top$, where m represents a neural input arriving from the Thalamus, and V_{th} is its corresponding weight. As mentioned in [7] the weights V_f and V_h, which are associated with the Amygdala, are non negative.

Let the optimal NN weight parameters be defined as

$$\begin{aligned} [V_f^*, W_f^*] = &\underset{V_f, W_f}{\arg\min} [\sup_{\tilde{\underline{x}}} |V_f^\top \Phi_A(\tilde{\underline{x}}) - W_f^\top \Phi(\tilde{\underline{x}}) - f(\tilde{\underline{x}})|], \\ &\text{s.t. } V_f \geq 0 \end{aligned} \tag{10}$$

$$\begin{aligned} [V_h^*, W_h^*] = &\underset{V_h, W_h}{\arg\min} [\sup_{\tilde{\underline{x}}} |V_h^\top \Phi_A(\tilde{\underline{x}}) - W_h^\top \Phi(\tilde{\underline{x}}) - 1/g(\tilde{\underline{x}})|], \\ &\text{s.t. } V_h \geq 0 \end{aligned} \tag{11}$$

which are bounded by known positive constants $\|V_f^*\| \leq M_{fv}$, $\|W_f^*\| \leq M_{fw}$, $\|V_h^*\| \leq M_{hv}$, and $\|W_h^*\| \leq M_{hw}$, and $\underline{\tilde{x}}$ is a dummy variable.

Inspired by the original adaptation rules proposed in [8] and [9], novel adaptation rules were introduced in [5], which include a projection algorithm to guarantee boundedness of the weights V_f, W_f, V_h, and W_h:

$$\dot{V}_f = \alpha_f \Phi_A \max(B_e^\top P_e s_e, 0), \dot{W}_f = -\beta_f \Phi B_e^\top P_e s_e \quad (12)$$
$$\dot{V}_h = \alpha_h \Phi_A \max(B_e^\top P_e s_e u_h, 0), \dot{W}_h = -\beta_h \Phi B_e^\top P_e s_e u_h$$

To guarantee the positiveness of V_f and V_h, the derivatives $\dot{V}_f$ and $\dot{V}_h$ are non negative.

Theorem 1 (LISIC Theorem from [5]): Consider the nonlinear system in equation (1) together with *Assumptions* 1, 2 and 3, and the control law

$$u = -\hat{h}(\underline{x})(\hat{f}(\underline{x}) + q_a + Ks - u_r) \quad (13)$$

where the estimates $\hat{f}$ and $\hat{h}$ are given by equation (8), with adaptation laws inspired by the limbic system computational model as described in equation (12), and u_r as defined in equation (7). Along the solution trajectories of this system, the error function s remains bounded and the H_∞ tracking performance criteria satisfies

$$\int_0^\top s_e^\top Q_e s_e dt \leq \tilde{V}_f(0)^\top \tilde{V}_f(0)/\alpha_f + \tilde{W}_f(0)^\top \tilde{W}_f(0)/\beta_f + \tilde{V}_h(0)^\top \tilde{V}_h(0)/\alpha_h + \tilde{W}_h(0)^\top \tilde{W}_h(0)/\beta_h + s_e^\top(0) P_e s_e^\top(0) + \rho^2 \int_0^\top \omega^\top \omega dt. \quad (14)$$

where the term ω represents the worst case perturbation in the sense of maximizing the derivative of the associate Lyapunov function, and the maximum value for ω^2 is less than $s_e^\top Q_e s_e/\rho^2$.

Proof: the proof of Theorem 1, which relies on the validity of *Assumptions* 1, 2 and 3, is formalized in [6].

III. Main Results

This section presents the three main results of this paper, which overcome the limitation of the original LISIC strategy.

A. Extending the use of LISIC to SISO affine systems

Consider a SISO affine system of order n, with dynamics described by

$$\dot{z} = \underline{f}(z) + \underline{g}(z)(u + \bar{d}), \quad y = \underline{h}(z) \quad (15)$$

where $\underline{h}(z)$ is a known $\mathbb{R}^n \to \mathbb{R}$ function, but $\underline{f}(z)$ and $\underline{g}(z)$ are unknown $\mathbb{R}^n \to \mathbb{R}^n$ continuous vector functions, $z \in \mathbb{R}^n$ represents the state vector, $u \in \mathbb{R}$ the control input, and $\bar{d} \in \mathbb{R}$ a perturbation.

Assumption 4: Assume a relative degree r (in the sense of [10]) equal to the order of the system.

Assumption 5: A global diffeomorphism $\Xi(z)$: $\mathbb{R}^n \to \mathbb{R}^n$ exists, and $\Xi(z) = \underline{x}$ for the system at hand which, according to [10] [11] is expressed by

$$\Xi(z) = \left[\underline{h}(z), \mathcal{L}_{\underline{f}}\underline{h}(z), \cdots, \mathcal{L}_{\underline{f}}^{n-1}\underline{h}(z)\right]^\top \quad (16)$$

Define $\bar{\Xi}(\underline{x})$ as the inverse of $\Xi(z)$ such that $\Xi(\bar{\Xi}(\underline{x})) = \underline{x}$ and $x = \underline{h}(z)$. Using the diffeomorphism in equation (16), the system in equation (15) can be rewritten using the Lie derivative operator $\mathcal{L}$ as

$$\dot{\underline{x}}_i = \underline{x}_{i+1}, \ \forall i = 1, \ldots, n-1$$
$$\dot{\underline{x}}_n = x^{(n)} = \mathcal{L}_{\underline{f}}^n \underline{h}(\bar{\Xi}(\underline{x})) + \mathcal{L}_{\underline{g}}\mathcal{L}_{\underline{f}}^{n-1}\underline{h}\left(\bar{\Xi}(\underline{x})\right)(u + \bar{d}),$$

or equivalently as the form in expressed in equation (1) for which the following Lie derivative properties were considered: $\mathcal{L}_f h(\underline{x}) = \frac{\partial h}{\partial x} f(\underline{x})$, $\mathcal{L}_f^{n+1} h(\underline{x}) = \mathcal{L}_f \mathcal{L}_f^n h(\underline{x})$ for $l > 0$, and $\mathcal{L}_f^0 h(\underline{x}) = h(\underline{x})$.

Proposition 1: Under *Assumptions* 4 and 5, *Theorem* 1 can be applied to the system (15) through the diffeomorphism in equation (16).

Remark 2: the *Assumptions* 1, 2 and 3 in Theorem 1 establish that the reference must be smooth up to its n^{th}-derivative and both functions f and g should be also smooth. The following subsections propose techniques to enable LISIC to operate satisfactorily, even if such Assumptions are not fulfilled.

B. Improving LISIC to operate under reference discontinuities or impulsive disturbances

In order to handle piece-wise smooth reference signals with punctual discontinuities, a technique is proposed to reset the NN weights (V_f,W_f,V_h,W_h) to zero, at the time when the discontinuity is experienced. A similar strategy can be applied for the scenario when the system is under an impulsive disturbance.

Assumption 6: the discontinuity and the impulsive effect cannot appear simultaneously, and there exists a minimum window of time before one of these events reappears.
A temporal regularization is added, i.e., a dwell-time τ, with $\dot{\tau} = 1$, in order to avoid Zeno evolution, namely, to avoid resetting infinitely many times in a finite time interval (see [12] for more details)

$$\text{if } (\dot{x}_d > M_{rr} \text{ or } \|\dot{\underline{x}}\|_2 > M_{xr}) \text{ and } \tau \geq \rho_\tau \quad (17)$$
$$\text{then } V_f^+ = 0,\ W_f^+ = 0,\ V_h^+ = 0,\ W_h^+ = 0,\ \xi^+ = 0,\ \tau^+ = 0,$$

with $\rho_\tau > 0$ the minimum time between resets.

At the exact instant of time when Assumptions 1 or 3 are not fulfilled, the LISIC NN weights and the integrator state are reset to zero. Right after the reset, the time $t = 0$ in the integral defined in equation (14) from *Theorem* 1 is replaced with the time at which the reset was executed.

Figure 1 illustrates the interconnection of the main components of the LISIC computational model, together with the reset strategy proposed in this work. Figure 2 shows the conceptual implementation of the reset strategy, which closely follows and respects the biologically-inspired architecture of the limbic system computational model proposed in [7].

C. Improving the LISIC strategy using a nominal model

Implementing a system identification procedure, like the one proposed in [13], a known nominal model similar to the

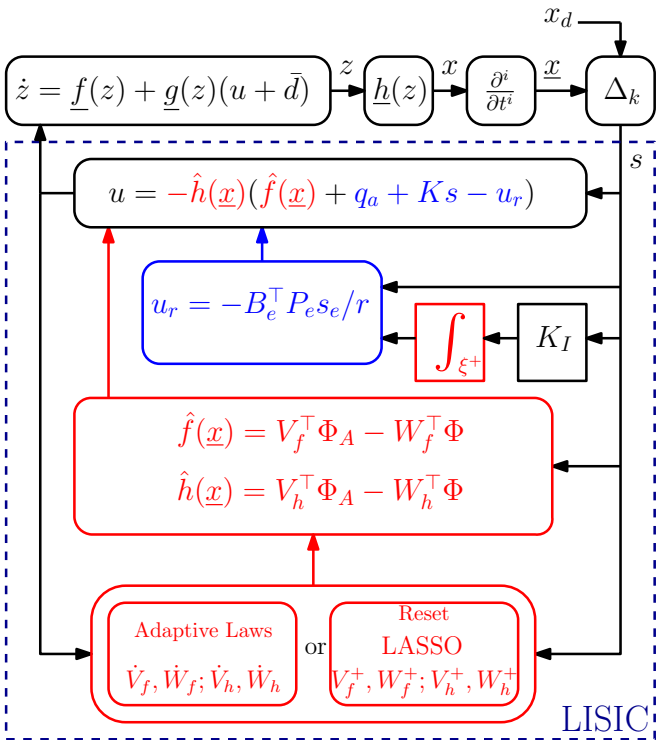

Fig. 1. Scheme of the reset strategy proposed in this work for making LISIC robust to unknown, possibly impulsive disturbances. The input is composed by two main components: the first one depending on a limbic system structure that estimates functions $f(\underline{x})$ and $g(\underline{x})$ (or $1/h(\underline{x})$) in red color, and the second depending on a state feedback with an integration action in blue color. The reset affects red components modifying the NN weights value for f and h. On the other hand, the transformed state ($\underline{x}$) is retrieved from the derivatives of $x = \underline{z}$. Reset values are defined with superindex $^+$.

one introduced in equation (1) can be made available as

$$z = \underline{f}_N(z) + \underline{g}_N(z)(u + \bar{d}), \quad y = \underline{h}_N(z) \tag{18}$$

For this mode, define additive model mismatches Δf and Δg associated with the functions f and g, respectively. Assuming a relative degree for $\Delta\underline{f}(z)$ and $\Delta\underline{g}(z)$ equal or greater than the nominal system r and using a diffeomorphism $\bar{\Xi}_N$ based on the nominal model, one obtains [14]:

$$\begin{aligned} x^{(n)} &= f_N(\underline{x}) + \Delta f(\underline{x}) + (g_N(\underline{x}) + \Delta g(\underline{x}))u + d(\underline{x}, t) \\ x^{(n)} &= f(\underline{x}) + g(\underline{x})u + d(\underline{x}, t) \\ \Delta f &= \mathcal{L}_{\Delta \underline{f}} \mathcal{L}_{\underline{f}_N}^{n-1} \underline{h}_N(\bar{\Xi}_N(\underline{x})),\ \Delta g = \mathcal{L}_{\Delta \underline{g}} \mathcal{L}_{\underline{g}_N}^{n-1} \underline{h}_N(\bar{\Xi}_N(\underline{x})) \end{aligned} \tag{19}$$

Optimal reset values (V_f^∇, W_f^∇, V_h^∇, W_h^∇) for the NN LISIC weight parameters can be computed using the nominal model and a partial constraint LASSO optimization as (see [15] for details):

$$\begin{aligned} [V_f^\nabla, W_f^\nabla] = \min_{V_f^\nabla, W_f^\nabla} \lambda_f \left\| \begin{bmatrix} V_f^\nabla \\ W_f^\nabla \end{bmatrix} \right\|_1 + \\ \left\| [\Phi_A^\top(s) \ -\Phi^\top(s)] \begin{bmatrix} V_f^\nabla \\ W_f^\nabla \end{bmatrix} - f_N(\underline{x}) \right\|_2^2 \\ \text{s.t. } V_f^\nabla \geq 0 \end{aligned} \tag{20}$$

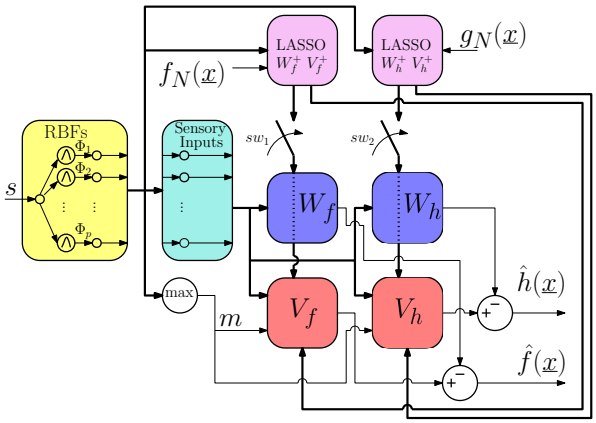

Fig. 2. The proposed limbic-system inspired computational model closely follows the biological structure of the limbic system in the mammalian brain. The analogy between the biological system and the proposed computational system are highlighted by means of the following color code: Orbitofrontal Cortex (OFC), Amygdala, Thalamus, Sensory cortex. The sensory input is processed in the Thalamus by means of multiple RBFs, generating a set of p sensory inputs. The output is an estimation of the unknown functions described in equation (8). In this schema, wider lines and thinner lines represent, respectively, vector and scalar variables. The two blocks in magenta take care of implementing the LASSO optimization using data from the nominal model. The numerical solution of these blocks is a set of optimal reset values. Closing the switches sw_1 and sw_1, theses values can be injected to replace the Amygdala and OFC weights.

$$\begin{aligned} [V_h^\nabla, W_h^\nabla] = \min_{V_h^\nabla, W_h^\nabla} \lambda_h \left\| \begin{bmatrix} V_h^\nabla \\ W_h^\nabla \end{bmatrix} \right\|_1 + \\ \left\| [\Phi_A^\top(s) \ -\Phi^\top(s)] \begin{bmatrix} V_h^\nabla \\ W_h^\nabla \end{bmatrix} - g_N^{-1}(\underline{x}) \right\|_2^2 \\ \text{s.t. } V_h^\nabla \geq 0 \end{aligned} \tag{21}$$

with positive tuning parameters λ_f and λ_h. At the exact moment where Assumptions 1 or 3 are not fulfilled, the LISIC NN weights are reset to the computed optimal values, as proposed in equation (17), but now using a set of optimal values V_f^∇, W_f^∇, V_h^∇, W_h^∇ instead of zeros. Right after the reset, the time $t = 0$ in the integral defined in equation (14) from *Theorem* 1 is replaced with the value of time at which the reset was executed.

IV. Numerical Simulation

The dynamic model of a Parrot Mambo UAS prototype, which was experimentally identified in [1], is adopted to demonstrate the soundness of the SISO extension and the NN weight resetting strategies. The control problem addresses the stabilization of the UAS translational dynamics, when the system is exposed to impulsive disturbances and non-smooth references. All simulations are implemented in Julia software [16].

A. Dynamic model of the UAS as a SISO affine system

The mathematical model of the latitudinal or longitudinal UAS translational dynamics corresponds to a linear system of order $n = 2$, which has the same form as equations (15), with functions associated as $\underline{f}(z) = Az$, $\underline{g}(z) = B$, and $\underline{h}(z) = C$. Computing a diffeomorphism as $\bar{\Xi}(z) = [C^\top \ A^\top C^\top]^\top$, the system can be rewritten in the form of equations (1) with

matrices $f(\underline{x}) = -CA([C^\top \; A^\top C^\top]^\top)^{-1}\underline{x}$ and $g(\underline{x}) = CB$. The structure of matrices A, B, C, and D is then:

$$A = \begin{bmatrix} 0 & a_1 \\ 0 & a_2 \end{bmatrix}, B = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}, B_d = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad (22)$$

The dynamic models for six Parrot Mambo UAS platforms were experimentally obtained. However, due to the different levels of wear and tear and manufacturing inconsistencies, all of the models obtained differed from each other. For this reason, nominal values for each identified parameter were chosen as the mean value of each set. Numerical values for the nominal model were: $a_{1,N} = 1000.8$, $a_{2,N} = -1.6013$, $b_{1,N} = -0.1172$, $b_{2,N} = 0.2345$, $f_N(\underline{x}) = 1000.8x_2$, $g_N(\underline{x}) = -0.1152$, $\Xi_N(z) = [z_1 \; 1000.8z_2]$, and $\bar{\Xi}_N(\underline{x}) = [x_1 \; 0.0009986x_2]$. Maximum values for these parameters were obtained as: $a_{1,max} = 1001.4$, $a_{2,max} = -1.0005$, $b_{1,max} = -0.1002$, $b_{2,max} = 0.2756$. Similarly, minimum values were obtained as: $a_{1,min} = 1000.5$, $a_{2,min} = -2.7036$, $b_{1,min} = -0.1378$, $b_{2,min} = 0.2004$.

Once the range for each parameter was defined, another combination of parameters was selected among the set, to simulate a non existing (i.e., virtual) but feasible UAS that could have been potentially developed by the manufacturer. The parameters adopted for this UAS correspond to $a_1 = a_{1,min}$, $a_2 = a_{2,max}$, $b_1 = -0.1152$, $b_2 = 0.2304$. Care was taken to make sure this new model was different enough from the six real UAS models. The virtual model was also included in the evaluation tests.

The nominal dynamic model was used to solve the LASSO optimization problem described by equations (20) and (21) in order to obtain the optimal NN reset weights.

B. Parameter selection for LISIC

LISIC parameters were heuristically tuned, using $p = 3$, $r = 0.00015$, $\rho = 0.01$, $\Delta_1 = 2$, $\alpha_f = 150$, $\beta_f = 80$, $\alpha_h = 0.0001$, $\beta_h = 0.005$, $K = 0.2$, $K_\text{I} = 10$, $Q_e = [20\,0.01; 0.01\,0.7]$, $\rho_\tau = 0.001$. The RBFs parameters were selected as $\mu_j = [-3, 0, 3]$ and σ_j are all equal to 3.

C. Definition of the piece-wise signals and impulsive noise

A piece-wise smooth reference signal was defined as $x_d = \pi \sin(2/5\pi t)/2 + J(t)$ with $J(0 < t < 10) = 0.2$, $J(10 \leq t < 25) = -3.4$ and $J(t \geq 25) = 0.2$. The piece-wise bounded perturbation $\bar{d}$ affected the system as: $\bar{d}(0\text{sec} \leq t < 35\text{sec}) = 0$, $\bar{d}(35\text{sec} \leq t < 45\text{sec}) = 120\text{N}$, $\bar{d}(t \geq 45\text{sec}) = 0$. An impulsive unknown L_1 perturbation was defined to appear at 63sec, which instantly and discontinuously moved the UAS position to $x = -3$.

D. Performance evaluation of the robust LISIC

The UAS started placed at the origin of the coordinate system and with no velocity, i.e. $z(0) = [0, 0]^\top$. From here, it was tasked with tracking a sinusoidal one-axis translational position reference. Figure 3 shows the reference to be tracked (red signal), together with the translational position of the UAS (blue signal) . The mismatch observed between $t = 10$sec and $t = 25$sec was due to the presence of an abrupt change of reference. These phenomena trigger two NN weight resets, one at 10sec (disturbance appears) and the other one at 25sec (disturbance disappears). Despite these abrupt changes, the reset strategy enabled the controller to stabilize the position of the UAS to the sinusoidal reference.

Figure 4 shows the time evolution of the position tracking error (blue signal) together with the UAS translational velocity. An unknown L_1 perturbation appears punctually at $t = 63$sec, and immediately affects the UAS position. The NN weight reset is then activated, re-enabling an effective trajectory tracking. This is observed since the tracking error remains small. As shown in the proof of Theorem 1, the tracking performance is guaranteed under an H_∞ performance criterion. The LISIC adaptation time after an NN weight reset is approximately 2sec, during which the tracking error is bounded.

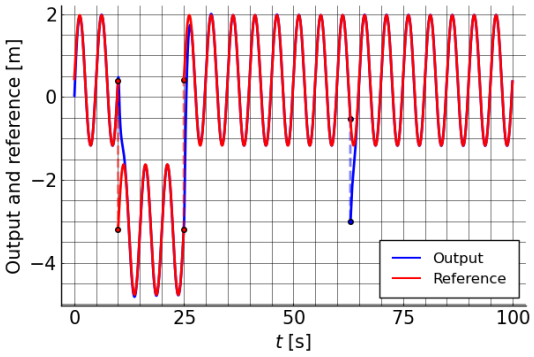

Fig. 3. The reference to be tracked (red), together with the position (blue) of the drone. The perturbation appears at $t = 35$sec and disappears at $t = 45$sec. The impulsive noise appears as $t = 63$sec. The symbol • represents a time discontinuity.

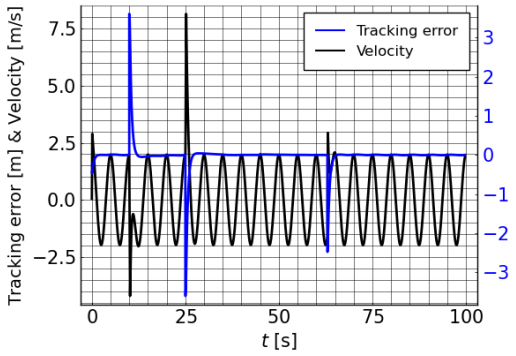

Fig. 4. Evolution of the tracking error e (blue) and velocity (black). The effect of the perturbation is noticeable on the velocity. However, in less than 2sec the velocity recovers its sinusoidal shape. Theorem 1 guarantees an H_∞ tracking performance criteria.

Figure 5 illustrates the time evolution of the control input generated by LISIC (blue signal) together with the perturbation $\bar{d}$, which appears at $t = 35$sec and disappears at $t = 45$sec. From the input signal, a fast reaction may be observed compensating for the perturbation.

Figure 6 shows the integrator state (ξ) from equation (5), illustrated in blue. The auxiliary variable s from equation (2) is shown in black. The quasi-constant value of the integrator

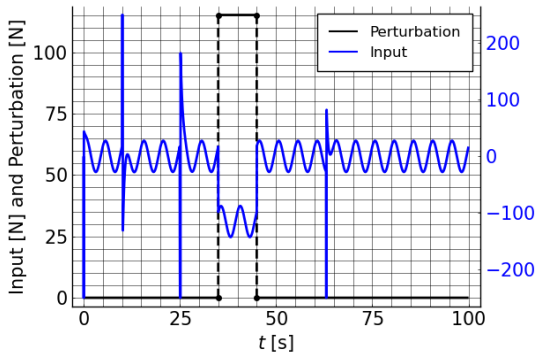

Fig. 5. The control input of the system is shown in blue, with corresponding axis value written on the right side of the plot. The perturbation $\bar{d}$ is shown in black, with corresponding axis value written on the left side of the plot. When the perturbation appears, the LISIC adaptation adjusts the input magnitude to compensate. Right after each reset, it is possible to observe its effects as high input values.

state between 35sec and 45sec enables compensation for the effect of the perturbation $\bar{d}$. Each time the integrator state is reset to 0, it rapidly evolves to compensate the effects of the new initial conditions, as observed, for example, when the controller is initialized at $t = 0$sec. During operation, each new reset event will define the initial time of the integral in equation (14).

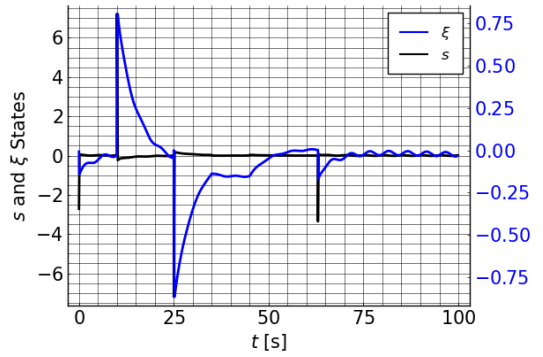

Fig. 6. Time evolution of the auxiliary signal (s), in black, and integrator state ξ, in blue.

V. Conclusions

The limbic system inspired control (LISIC) framework was revisited, introducing three contributions that facilitate its implementation in real-world conditions. The first contribution corresponds to an extension enabling the implementation of LISIC to the domain of SISO affine systems. The second contribution is a strategy for resetting the controller's NN weights, enabling it to deal with piece-wise smooth references and impulsive perturbations. The third contribution, which relies on the availability of a nominal model of the system, computes a set of optimal NN weight reset values by solving a convex constrained optimization problem online. Numerical simulations addressing the stabilization of the translational dynamics of a UAS via the robust LISIC demonstrate the benefits of adopting the extension to SISO systems and the two NN weight reset strategies.

Future work will explore Lyapunov stability theory for hybrid systems, in order to include the reset strategy in the H_∞ index. The implementation of the robust LISIC strategy under a Spiking NN approach will also be explored to continue the work presented in [17].

Acknowledgments

This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, participating as a funding organization for the NSF Collaborative Research in Computational Neuroscience (CRCNS) Program, under contract number KJ0403010 (L.R. Garcia Carrillo and A.T. Sornborger).

References

[1] I. Rubio Scola, G. A. Guijarro Reyes, L. R. Garcia Carrillo, J. P. Hespanha, and L. Burlion, "A robust control strategy with perturbation estimation for the parrot mambo platform," *IEEE Trans. Control Syst. Technol*, vol. 29, pp. 1389–1404, 2020.

[2] A. Kitanov, V. Tubin, and I. Petrovic, "Extending functionality of rf ultrasound positioning system with dead-reckoning to accurately determine mobile robot's orientation," in *2009 IEEE Control Applications,(CCA) & Intelligent Control,(ISIC)*, pp. 1152–1157, IEEE, 2009.

[3] D. Klein, J. Isaacs, S. Venkateswaran, J. Burman, T. Pham, J. Hespanha, and U. Madhow, "Source localization in a sparse acoustic sensor network using uav-based semantic data mules," *ACM Transactions on Sensor Networks*, vol. 9, pp. 1–29, 2013.

[4] L. R. G. Carrillo, W. J. Russell, J. P. Hespanha, and G. E. Collins, "State estimation of multiagent systems under impulsive noise and disturbances," *IEEE TCST*, vol. 23, no. 1, pp. 13–26, 2015.

[5] I. Rubio Scola, L. R. Garcia Carrillo, and J. P. Hespanha, "Stable robust controller inspired by the mammalian limbic system for a class of nonlinear systems," in *2020 American Control Conference (ACC)*, pp. 842–847, IEEE, 2020.

[6] I. Rubio Scola, L. R. Garcia Carrillo, and J. P. Hespanha, "Limbic system-inspired performance-guaranteed control for nonlinear multi-agent systems with uncertainties," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1 – 12, 2021. Early Access.

[7] C. Balkenius and J. Morén, "Emotional learning: A computational model of the amygdala," *Cybernetics & Systems*, vol. 32, no. 6, pp. 611–636, 2001.

[8] F. Baghbani, M.-R. Akbarzadeh-T, and M.-B. N. Sistani, "Stable robust adaptive radial basis emotional neurocontrol for a class of uncertain nonlinear systems," *Neurocomputing*, vol. 309, pp. 11–26, 2018.

[9] F. Baghbani, M.-R. Akbarzadeh-T, M.-B. Naghibi-Sistani, and A. Akbarzadeh, "Emotional neural networks with universal approximation property for stable direct adaptive nonlinear control systems," *Eng. Appl. Artif. Intell.*, vol. 89, p. 103447, Mar. 2020.

[10] J.-J. E. Slotine, W. Li, *et al.*, *Applied nonlinear control*, vol. 199. Prentice hall Englewood Cliffs, NJ, 1991.

[11] A. Isidori, *Nonlinear Control Systems, Third Edition*. Communications and Control Engineering, Springer, 1995.

[12] C. Prieur, I. Queinnec, S. Tarbouriech, and L. Zaccarian, "Analysis and synthesis of reset control systems," *Foundations and Trends® in Systems and Control*, vol. 6, no. 2-3, pp. 117–338, 2018.

[13] J. P. Hespanha, "Advanced undergraduate topics in control systems design," 2019. Available at www.ece.ucsb.edu/ hespanha/published.

[14] M. A. Henson and D. E. Seborg, "Feedback linearizing control," in *Nonlinear process control*, vol. 4, pp. 149–231, Prentice-Hall Upper Saddle River, NJ, USA, 1997.

[15] P. T. P. Tang, "Convergence of lca flows to (c) lasso solutions," *arXiv preprint arXiv:1603.01644*, 2016.

[16] J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah, "Julia: A fresh approach to numerical computing," *SIAM Review*, vol. 59, no. 1, pp. 65–98, 2017.

[17] I. Rubio Scola, L. R. Garcia Carrillo, T. C. Stewart, and A. T. Sornborger, "A neuromorphic control architecture inspired by the limbic system," in *60th IEEE CDC*, pp. 1413–1418, 2021.