Design of a Deep Neural Network-Based Integral Sliding Mode Control for Nonlinear Systems Under Fully Unknown Dynamics

Edoardo Vacchini, Nikolas Sacchi, Gian Paolo Incremona, *Senior Member, IEEE*, and Antonella Ferrara, *Fellow, IEEE*

***Abstract*— In this paper a novel deep neural network based integral sliding mode (DNN-ISM) control is proposed for controlling perturbed systems with fully unknown dynamics. In particular, two DNNs with an arbitrary number of hidden layers are exploited to estimate the unknown drift term and the control effectiveness matrix of the system, which are instrumental to design the ISM controller. The DNNs weights are adjusted according to adaptation laws derived directly from Lyapunov stability analysis, and the proposal is satisfactorily assessed in simulation relying on benchmark examples.**

***Index Terms*— Sliding mode control, deep neural networks, uncertain systems.**

I. Introduction

WHEN dealing with systems affected by external disturbances and modeling mismatches, an effective technique is sliding mode control (SMC). In fact, it guarantees robustness of the controlled system against matched uncertainties thanks to the discontinuous nature of the control law, which allows to drive the systems states towards the sliding manifold in a finite time [1]. Classical SMC presents two main drawbacks. The former is the so-called chattering phenomenon, caused by the discontinuous nature of the control signal and affected by its magnitude. The latter is that, in the time period during which the states are approaching the sliding manifold, the system is sensitive to the uncertainties. For chattering reduction, methodologies like higher order SMC [2], [3], adaptive strategies [4], [5] and internal model principle based strategies [6] have been proposed.

To improve the robustness of SMC, the ISM paradigm has been introduced in [7]. The core idea of ISM is to rely on an additional term called *transient function* to ensure that the states lie on the sliding manifold from the initial time instant. During the years, several improvements to ISM have been made, see, e.g., [8] among others, while its efficacy has been assessed in several works, such as [9], [10].

However, in order to design an ISM control scheme, the knowledge of the nominal dynamics of the system is required. In many practical implementations, such a knowledge is not available and only conservative bounds are retrieved relying on physical characteristics of the system or experimental data.

In the domain of the control theory, neural networks (NNs) and deep neural networks (DNNs) have been employed, e.g., [11], [12], among many others. However, in many works, the efficacy of DNNs is assessed only empirically, without theoretical guarantees. One of the first works in which such guarantees are provided is [13], in which the weights of the employed NN are adjusted with adaptation laws obtained through Lyapunov stability analysis. Similar concepts have been adopted in other works to approximate the optimal control law, see, e.g., [14], or to partially estimate the model of the system [15]. In [16], [17], NNs have been applied also in the domain of SMC, e.g., [18], [19]. Then, for what concerns the ISM framework, [19] proposes a couple of two-layer NNs to estimate the drift term and the control effectiveness of a particular kind of system, instrumental to the design of the sliding manifold. This last work gave rise to the so-called NN-ISM control approach. Such an approach presents two main limitations. The former is that only systems with scalar input are considered, while the latter is that NNs with only two layers are used for the approximation of the dynamics. When systems exhibit a highly nonlinear dynamics, better approximation capabilities may be required. As shown for instance in [20]–[22], NNs with a deep architecture represent an efficient solution.

Motivated by the results obtained in [19], and with the aim of overcoming the aforementioned limitations, in this work we propose a novel DNN-ISM control algorithm. Specifically, we consider a generic nonlinear system with fully unknown nominal dynamics. Then, we exploit two DNNs with an arbitrary number of hidden layers to estimate the drift term and the control effectiveness matrix, instrumental to the design of the integral sliding manifold. Despite a more complex architecture, it allows to reduce the bound on the error between the approximated function and the real one, and to reduce also the control gain with beneficial effects in terms of chattering. The weights of the two DNNs are adjusted relying on adaptation

This is the final version of the paper accepted for publication in IEEE Control Systems Letters, doi: 10.1109/LCSYS.2023.3281288.
E. Vacchini, N. Sacchi and A. Ferrara are with the Dipartimento di Ingegneria Industriale e dell'Informazione, University of Pavia, 27100, Pavia, Italy (e-mail: *edoardo.vacchini01@universitadipavia.it, nikolas.sacchi01@universitadipavia.it, antonella.ferrara@unipv.it*). G. P. Incremona is with Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, 20133 Milan, Italy (e-mail: *gianpaolo.incremona@polimi.it*).

laws directly obtained from stability analysis. More precisely, in this paper the considered control problem is addressed for two specific classes of systems. First, the multi-input case is taken into account, analysing the tractable case of number of states being equal to an integer multiple of the number of inputs. Then, as particular case of the previous one, the class of single-input systems is discussed. It is worth highlighting that these systems categories are widely exploited for modeling in the literature and practical applications. The proposed DNN-ISM control scheme is finally assessed in simulation on two benchmark examples.

Notation: Given $A \in \mathbb{R}^{n\times m}$, then $\mathrm{vec}(A) \in \mathbb{R}^{nm}$ is the vectorization operation. Given a real square matrix $A \in \mathbb{R}^{n\times n}$, then $\bar{\lambda}(A)$ and $\underline{\lambda}(A)$ are the greatest and smallest singular values of A. Given two matrices $A \in \mathbb{R}^{m\times n}$ and $B \in \mathbb{R}^{p\times q}$, their Kronecker product is denoted as $A\otimes B \in \mathbb{R}^{pm\times qn}$. Given $a \in \mathbb{R}^{nm}$, the inverse of the vectorization operation is defined as $\mathrm{vec}^{-1}(a) = \left(\mathrm{vec}(I_m)^\top \otimes I_n\right)(I_m \otimes a) \in \mathbb{R}^{n\times m}$. Given a DNN $\Upsilon(x) : \mathbb{R}^p \to \mathbb{R}^q$, characterized by $k_\Upsilon \in \mathbb{N}$ hidden layers, the number of neurons in the jth layer is $L_{\Upsilon_j} \in \mathbb{R}_{>0}$, for $j = 1, \dots, k_\Upsilon + 1$ with $L_{\Upsilon_0} = p$ and $L_{\Upsilon_{k_\Upsilon+1}} = q$. Given the matrices A_i, $i = 1, 2, \dots, N$, with compatible dimensions one has that $\prod_{i=1}^{\curvearrowleft N} A_i = A_N A_{N-1} \dots A_1$, and $\prod_{i=p}^{\curvearrowleft p-1} = 1$. Given $f(x) : \mathbb{R}^n \to \mathbb{R}^m$, then $f' \in \mathbb{R}^{n\times m}$ is its Jacobian.

II. Preliminaries on ISM and Problem Statement

In this section, the dynamical system considered in the paper is introduced. Moreover, the main features of ISM control, originally presented in [7], are recalled. Consider the nonlinear system

$$\dot{x} = f(x(t)) + B(x(t))u(t) + h(t), \quad x(0) = x_0, \tag{1}$$

where $x \in \Omega$ is the system states, with $\Omega \subset \mathbb{R}^n$ being a compact set containing the origin, $f : \Omega \to \mathbb{R}^n$ represents the drift dynamics, $B : \Omega \to \mathbb{R}^{n\times m}$ is the control effectiveness matrix, and both functions are bounded and belong to $\mathrm{C}^0(\Omega)$. Then, $u \in \mathbb{R}^m$ is the control vector, while $h : \mathbb{R}_{\geq 0} \to \mathbb{R}^n$ represents the perturbation vector. The following assumption, common in the sliding mode control theory, is needed.

$\mathcal{A}_1$: The uncertainty h is such that $h \in \mathcal{H}$, where $\mathcal{H} \subset \mathbb{R}^n$ is a compact set containing the origin, with $\bar{h} := \sup_{h\in\mathcal{H}}\|h\|$ known.

To counteract the effect of the external perturbation, an ISM controller can be designed [7]. In particular, the control action is defined as $u = u_0 + u_1$, with $u_0 \in \mathbb{R}^m$ being a bounded control law which makes the state $x^* \in \mathbb{R}^n$ an asymptotically stable equilibrium point for the nominal dynamics, that is (1) with $h = 0$, while $u_1 \in \mathbb{R}^m$ is a discontinuous control signal whose aim is to make the system robust against uncertainties. Since in the case of $m > 1$, system (1) is multi-input-multi-output, possibly coupled, and nonlinear, the discontinuous control law is defined according to the unit vector approach [1], i.e.,

$$u_1 = -\rho\frac{\sigma(x)}{\|\sigma(x)\|}, \tag{2}$$

where $\rho \in \mathbb{R}_{>0}$ is a constant gain chosen so that the worst realization of the perturbation is dominated, while $\sigma(x) : \mathbb{R}^n \to \mathbb{R}^m$ is the *integral sliding variable*, defined as

$$\sigma(x) = \sigma_0(x) + z(x), \quad \sigma(x_0) = 0. \tag{3}$$

The term $\sigma_0 : \mathbb{R}^n \to \mathbb{R}^m$ is chosen by the designer, for instance as a linear combination of states. The term $z(x) : \mathbb{R}^n \to \mathbb{R}^m$ appearing in (3) is instead the so-called *transient function*, and its dynamics is defined as

$$\dot{z} = -\frac{\partial\sigma_0(x)}{\partial x}\left(f(x) + B(x)u_0\right), \quad z(x_0) = -\sigma_0(x_0), \tag{4}$$

with $\frac{\partial\sigma_0(x)}{\partial x} \in \mathbb{R}^{m\times n}$. Then, suitably defining the stabilizing control law u_0 and the discontinuous control gain ρ, a sliding mode is enforced. As a result, the robustness of the controlled system against matched disturbances h can be proved [7].

III. Deep Neural Network-Based Dynamics Approximators

As described in [7] and highlighted in (4), the knowledge of the drift term f and the control effectiveness matrix B is required to design an ISM control. In this paper, the nominal model of the system is considered fully unknown [19]. Therefore, the aim of this section is to introduce two DNNs which estimate the unknown terms.

Let $x_h = \begin{bmatrix} x^\top & 1 \end{bmatrix}^\top \in \mathbb{R}^{n+1}$, since both f and B are continuous, then, by virtue of the so-called *universal approximation property* [23], there exist two ideal DNNs, namely $\Phi(x_h) : \mathbb{R}^{n+1} \to \mathbb{R}^n$ and $\Psi(x_h) : \mathbb{R}^{n+1} \to \mathbb{R}^{nm}$, characterized by $k_\Phi, k_\Psi > 2$ hidden layers, which approximate the nominal dynamics of (1) as

$$f(x) = \Phi(x_h) + \varepsilon_\Phi(x), \tag{5}$$
$$\mathrm{vec}(B(x)) = \Psi(x_h) + \varepsilon_\Psi(x), \tag{6}$$

where $\varepsilon_\Phi(x) : \Omega \to \mathbb{R}^n$ and $\varepsilon_\Psi(x) : \Omega \to \mathbb{R}^{nm}$ are the so-called *approximation errors*. More in depth, the DNNs can be written as

$$\Phi(x_h) = V_{k_\Phi}^\top \phi_{k_\Phi} \circ \dots \circ V_1^\top \phi_1 \circ V_0^\top x_h \tag{7}$$
$$\Psi(x_h) = U_{k_\Psi}^\top \psi_{k_\Psi} \circ \dots \circ U_1^\top \psi_1 \circ U_0^\top x_h, \tag{8}$$

where $V_j \in \mathbb{R}^{L_{\Phi_j}\times L_{\Phi_{j+1}}}$, with $j = 0, 1, \dots, k_\Phi$, and $U_j \in \mathbb{R}^{L_{\Psi_j}\times L_{\Psi_{j+1}}}$, with $j = 0, 1, \dots, k_\Psi$, are the ideal weights of DNNs. Finally, $\phi_j(\cdot)$, with $j = 1, 2, \dots, k_\Phi$, and $\psi_j(\cdot)$, with $j = 1, 2, \dots, k_\Psi$, are the activation functions of $\Phi(x)$ and $\Psi(x)$, respectively, both being C^1 and Lipschitz continuous.

Since $\mathrm{vec}^{-1}(\Psi(x_h) + \varepsilon_\Psi) = B(x)$, let $\mathrm{vec}^{-1}(\Psi(x_h))^{(i)} \in \mathbb{R}^n$ be the ith column of $\mathrm{vec}^{-1}(\Psi(x_h))$, with $i = 1, 2, \dots, m$. Consider the matrix $U_{k_\Psi} \in \mathbb{R}^{L_{k_\Psi}\times nm}$ written as the concatenation of different sub-matrices $U_{k_\Psi}^{(i)} \in \mathbb{R}^{L_{k_\Psi}\times n}$, horizontally stacked as $U_{k_\Psi} = \begin{bmatrix} U_{k_\Psi}^{(1)} & U_{k_\Psi}^{(2)} & \dots & U_{k_\Psi}^{(m)} \end{bmatrix}$. Then, the expression of $\mathrm{vec}^{-1}(\Psi(x_h))^{(i)}$ is given by

$$\mathrm{vec}^{-1}(\Psi(x_h))^{(i)} = (U_{k_\Psi}^{(i)})^\top \psi_{k_\Psi} \circ \dots \circ U_1^\top \psi_1 \circ U_0^\top x_h. \tag{9}$$

In order to design the proposed approach, it is convenient to express the DNNs in a recursive way, defining the output of the generic jth layer. For $\Phi(x_h)$, one has

$$\Phi_j = \begin{cases} V_j^\top \phi_j(\Phi_{j-1}) & \text{if } j = 1, 2, \dots, k_\Phi, \\ V_0^\top x_h & \text{if } j = 0, \end{cases} \tag{10}$$

where $\Phi_j \in \mathbb{R}^{L_{\Phi_{j+1}}}$. Meanwhile, the output of the jth layer of $\Psi(x_h)$ is given by

$$\Psi_j = \begin{cases} U_j^\top \psi_j(\Psi_{j-1}) & \text{if } j = 1, 2, \dots, k_\Psi, \\ U_0^\top x_h & \text{if } j = 0, \end{cases} \tag{11}$$

with $\Psi_j \in \mathbb{R}^{L_{\Psi_{j+1}}}$. Note that, recursively substituting values of Φ_{j-1} and Ψ_{j-1} in (10) and (11), respectively, one has that $\Phi(x_h) \equiv \Phi_{k_\Phi}$ and $\Psi(x_h) \equiv \Psi_{k_\Psi}$.

Let $\text{vec}^{-1}(\Psi_{k_\Psi})^{(i)} = (U_{k_\Psi}^{(i)})^\top \psi_{k_\Psi}(\Psi_{k_\Psi - 1}) \in \mathbb{R}^n$ be the ith column deriving from $\text{vec}^{-1}(\Psi_{k_\Psi})$, corresponding to the output of the last layer of $\Psi(x_h)$. By virtue of the boundedness of $f(x)$ and $B(x)$, the following assumption about the ideal DNNs holds.

$\boldsymbol{\mathcal{A}_2}$: There exist known constants $\bar{V}, \bar{U}, \bar{\varepsilon}_\Phi, \bar{\varepsilon}_\Psi \in \mathbb{R}_{>0}$ such that,

$$\begin{array}{ll} \sup_{x_h \in \Omega} \|V_j\| \leq \bar{V}, & \sup_{x_h \in \Omega} \|U_p\| \leq \bar{U}, \\ \sup_{x_h \in \Omega} \|\varepsilon_\Phi\| \leq \bar{\varepsilon}_\Phi, & \sup_{x_h \in \Omega} \|\text{vec}^{-1}(\varepsilon_\Psi)\| \leq \bar{\varepsilon}_\Psi \end{array}$$

with $j = 0, 1, \dots, k_\Phi$, $p = 0, 1, \dots, k_\Psi$, and $i = 1, 2, \dots, m$.

Since the ideal DNNs (7) and (8) are not known, their approximation is used to estimate the model (1), i.e.,

$$\widehat{f}(x) = \widehat{\Phi}(x_h), \quad \text{vec}(\widehat{B}(x)) = \widehat{\Psi}(x_h). \tag{12}$$

In particular, $\widehat{\Phi}(x_h) : \mathbb{R}^{n+1} \to \mathbb{R}^n$ and $\widehat{\Psi}(x_h) : \mathbb{R}^{n+1} \to \mathbb{R}^{nm}$ are defined as

$$\widehat{\Phi}(x_h) = \widehat{V}_{k_\Phi}^\top \phi_{k_\Phi} \circ \cdots \circ \widehat{V}_1^\top \phi_1 \circ \widehat{V}_0^\top x_h \tag{13}$$

$$\widehat{\Psi}(x_h) = \widehat{U}_{k_\Psi}^\top \psi_{k_\Psi} \circ \cdots \circ \widehat{U}_1^\top \psi_1 \circ \widehat{U}_0^\top x_h, \tag{14}$$

where $\widehat{V}_j \in \mathbb{R}^{L_{\Phi_j} \times L_{\Phi_{j+1}}}$, with $j = 0, 1, \dots, k_\Phi$, and $\widehat{U}_j \in \mathbb{R}^{L_{\Psi_j} \times L_{\Psi_{j+1}}}$, with $j = 0, 1, \dots, k_\Psi$, are the estimates of the ideal weights of DNNs.

As it occurs for the ideal DNNs, it is possible to express the output of the jth layer of (13) and (14), i.e., $\widehat{\Phi}_j$ and $\widehat{\Psi}_j$, as reported in (10) and (11), respectively. The only change is that the estimates of the ideal weights, i.e., $\widehat{V}_j$ and $\widehat{U}_j$, must be used. Note that, for sake of readability, the activation functions $\phi_j(\Phi_{j-1})$ and $\psi_j(\Psi_{j-1})$ will be referred as ϕ_j and ψ_j, respectively. Moreover, we will indicate $\widehat{\phi}_j = \phi_j(\widehat{\Phi}_{j-1})$ and $\widehat{\psi}_j = \psi_j(\widehat{\Psi}_{j-1})$.

In the following, the difference between the output of the ideal DNNs and the ones with estimated weights is defined for each layer j. For what concerns Φ_j, adding and subtracting $V_j^\top \widehat{\phi}_j$ one has

$$\widetilde{\Phi}_j = \Phi_j - \widehat{\Phi}_j = \widetilde{V}_j^\top \widehat{\phi}_j + V_j^\top (\phi_j - \widehat{\phi}_j), \tag{15}$$

with $j = 1, 2, \dots, k_\Phi$, $\widetilde{V}_j = V_j - \widehat{V}_j$ and $\widetilde{\Phi}_0 = \widetilde{V}_0^\top x_h$. Since $\phi_j(\Phi_{j-1})$ is not known, ϕ_j can be approximated using first order Taylor approximation around $\widehat{\Phi}_{j-1}$, obtaining

$$\phi_j(\Phi_{j-1}) = \phi_j(\widehat{\Phi}_{j-1}) + \widehat{\phi}_j' \widetilde{\Phi}_{j-1} + O^2(\widetilde{\Phi}_{j-1}), \tag{16}$$

where $\widehat{\phi}_j' = \phi_j'(\widehat{\Phi}_{j-1}) \in \mathbb{R}^{L_{\Phi_j} \times L_{\Phi_j}}$, while $O^2(z)$ denotes term of order two [13]. Note that, such an approximation is instrumental only to derive the weight update laws, without interfering with the nonlinear approximation capabilities of the adopted DNNs. Exploiting the fact that $V_j = \widetilde{V}_j + \widehat{V}_j$ and substituting (16), one can reformulate (15) as

$$\widetilde{\Phi}_j = \widetilde{V}_j^\top \widehat{\phi}_j + \widehat{V}_j^\top \widehat{\phi}_j' \widetilde{\Phi}_{j-1} + \Delta_{\Phi_j}, \tag{17}$$

where $\Delta_{\Phi_j} = \widetilde{V}_j^\top \widehat{\phi}_j' \widetilde{\Phi}_{j-1} + V_j^\top O^2(\widetilde{\Phi}_{j-1})$. Moreover, since $\widetilde{V}_j^\top \widehat{\phi}_j = \text{vec}(\widetilde{V}_j^\top \widehat{\phi}_j) = \text{vec}(\widehat{\phi}_j^\top \widetilde{V}_j I_{L_{\Phi_{j+1}}})$, it is true that $\widetilde{V}_j^\top \widehat{\phi}_j = \left(I_{L_{\Phi_{j+1}}} \otimes \widehat{\phi}_j^\top\right) \text{vec}(\widetilde{V}_j)$ [24]. Hence, the error associated with the jth layer can be then reformulated as

$$\widetilde{\Phi}_j = \left(I_{L_{\Phi_{j+1}}} \otimes \widehat{\phi}_j^\top\right) \text{vec}(\widetilde{V}_j) + \widehat{V}_j^\top \widehat{\phi}_j' \widetilde{\Phi}_{j-1} + \Delta_{\Phi_j}, \tag{18}$$

with $\widetilde{\Phi}_0 = \left(I_{L_{\Phi_1}} \otimes x_h^\top\right) \text{vec}(\widetilde{V}_0)$. By iteration (see Lemma 1 in [15]), one can write

$$\widetilde{\Phi}_{k_\Phi} = \sum_{j=0}^{k_\Phi} \Lambda_{\Phi_j} \text{vec}(\widetilde{V}_j) + \sum_{j=1}^{k_\Phi} \Xi_{\Phi_j} \Delta_{\Phi_j}, \tag{19}$$

where $\Xi_{\Phi_j} \in \mathbb{R}^{n \times L_{\Phi_{j+1}}}$ and $\Lambda_{\Phi_j} \in \mathbb{R}^{n \times (L_{\Phi_j} L_{\Phi_{j+1}})}$ are given by

$$\Xi_{\Phi_j} = \overset{\curvearrowleft}{\prod_{p=j+1}^{k_\Phi}} \widehat{V}_p^\top \widehat{\phi}_p', \quad \Lambda_{\Phi_j} = \Xi_{\Phi_j} \left(I_{L_{\Phi_{j+1}}} \otimes \widehat{\phi}_j^\top\right), \tag{20}$$

with $\Lambda_{\Phi_0} = \Xi_{\Phi_0} \left(I_{L_{\Phi_1}} \otimes x_h\right)$.

As for Ψ, the error at the jth layer, up to the penultimate one, can be computed following the same reasoning made for Φ. In particular, for $j = 0, 1, \dots, k_\Psi - 1$, it is possible to express the errors as

$$\widetilde{\Psi}_j = \left(I_{L_{\Psi_{j+1}}} \otimes \widehat{\psi}_j^\top\right) \text{vec}(\widetilde{U}_j) + \widehat{U}_j^\top \widehat{\psi}_j' \widetilde{\Psi}_{j-1} + \Delta_{\Psi_j}, \tag{21}$$

with $\widetilde{\Psi}_0 = \left(I_{L_{\Psi_1}} \otimes x_h^\top\right) \text{vec}(\widetilde{U}_0)$ and $\Delta_{\Psi_j} \in \mathbb{R}^{L_{\Psi_{j+1}}}$ defined as $\Delta_{\Psi_j} = \widetilde{U}_j^\top \widehat{\psi}_j' \widetilde{\Psi}_{j-1} + U_j^\top O^2(\widetilde{\Psi}_{j-1})$.

As for the error associated with the last layer, i.e., $j = k_\Psi$, it can be computed column-wise as

$$\begin{aligned} \text{vec}^{-1}(\widetilde{\Psi}_{k_\Psi})^{(i)} = &\left(I_n \otimes \widehat{\psi}_{k_\Psi}^\top\right) \text{vec}(\widetilde{U}_{k_\Psi}^{(i)}) + \\ &+ (\widehat{U}_{k_\Psi}^{(i)})^\top \widehat{\psi}_{k_\Psi}' \widetilde{\Psi}_{k_\Psi - 1} + \text{vec}^{-1}(\Delta_{\Psi_{k_\Psi}})^{(i)}. \end{aligned} \tag{22}$$

Exploiting its recursive nature, it can be written as

$$\begin{aligned} \text{vec}^{-1}(\widetilde{\Psi}_{k_\Psi})^{(i)} = &\Lambda_{\Psi_{k_\Psi}}^{(i)} \text{vec}(\widetilde{U}_{k_\Psi}^{(i)}) + \text{vec}^{-1}(\Delta_{\Psi_{k_\Psi}})^{(i)} + \\ &+ \sum_{j=0}^{k_\Psi - 1} \Lambda_{\Psi_j}^{(i)} \text{vec}(\widetilde{U}_j) + \sum_{j=1}^{k_\Psi - 1} \Xi_{\Psi_j}^{(i)} \Delta_{\Psi_j}, \end{aligned} \tag{23}$$

where $\Xi_{\Psi_j}^{(i)} = (\widehat{U}_{k_\Psi}^{(i)})^\top \widehat{\psi}_{k_\Psi}' \overset{\curvearrowleft}{\prod}_{l=j+1}^{k_\Psi - 1} \widehat{U}_l^\top \widehat{\psi}_l' \in \mathbb{R}^{n \times L_{\Psi_{j+1}}}$, while $\Lambda_{\Psi_j}^{(i)} = \Xi_{\Psi_j}^{(i)} \left(I_{L_{\Psi_{j+1}}} \otimes \widehat{\psi}_{L_{\Psi_j}}^\top\right) \in \mathbb{R}^{n \times L_{\Psi_j} L_{\Psi_{j+1}}}$, with $\Lambda_{\Psi_0}^{(i)} = \Xi_{\Psi_0}^{(i)} \left(I_{L_{\Psi_1}} \otimes x_h^\top\right)$ and $\Lambda_{\Psi_{k_\Psi}}^{(i)} = \left(I_n \otimes \widehat{\psi}_{k_\Psi}^\top\right)$.

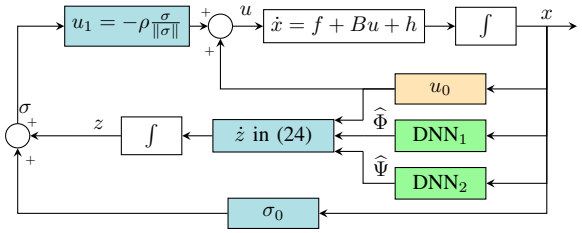


Fig. 1: The proposed DNN-ISM control scheme.

IV. DNN-ISM Control Scheme

The aim of this section is to introduce the proposed DNN-ISM control scheme, illustrated in Fig 1.

Relying on the estimation introduced in (12), it is possible to approximate the dynamics of the transient function (4) as

$$\dot{z} = -\frac{\partial \sigma_0}{\partial x}\left(\widehat{\Phi}(x_h) + \text{vec}^{-1}(\widehat{\Psi}(x_h))u_0\right), \tag{24}$$

with $z(x_0) = -\sigma_0(x_0)$.

The adaptation laws for the weights of the DNNs are directly derived from the stability analysis. As for $\widehat{\Phi}$, the weights of the layers $j = 0, 1, \ldots, k_\Phi$ are adapted with

$$\text{vec}\left(\dot{\widehat{V}}_j\right) = \text{proj}\left(\Gamma_{\Phi_j}\Lambda_{\Phi_j}^\top \frac{\partial \sigma_0}{\partial x}^\top \sigma\right), \tag{25}$$

where $\Gamma_{\Phi_j} \in \mathbb{R}^{L_{\Phi_j}L_{\Phi_{j+1}} \times L_{\Phi_j}L_{\Phi_{j+1}}}$ is a diagonal gain matrix. For what concerns $\widehat{\Psi}$, its weights are updated as

$$\text{vec}\left(\dot{\widehat{U}}_j\right) = \text{proj}\left(\Gamma_{\Psi_j}\left(\sum_{i=1}^{m} u_{0,i}(\Lambda_{\Psi_j}^{(i)})^\top\right)\frac{\partial \sigma_0}{\partial x}^\top \sigma\right) \tag{26}$$

for the layers $j = 0, 1, \ldots, k_\Psi - 1$, where $u_{0,i}$ is the ith element of the vector u_0, while for the last layer one has

$$\text{vec}\left(\dot{\widehat{U}}_{k_\Psi}^{(i)}\right) = \text{proj}\left(\Gamma_{\Psi_{k_\Psi}} u_{0,i}(\Lambda_{\Psi_{k_\Psi}}^{(i)})^\top \frac{\partial \sigma_0}{\partial x}^\top \sigma\right), \tag{27}$$

where $\Gamma_{\Psi_j} \in \mathbb{R}^{L_{\Psi_j}L_{\Psi_{j+1}} \times L_{\Psi_j}L_{\Psi_{j+1}}}$, for $j = 0, 1, \ldots, k_\Psi$. The operator $\text{proj}(\cdot)$ is the projection operator defined as in [25] and it ensures that $\text{vec}(\widehat{V}_j) \in \mathcal{B}_j^\Phi$ and $\text{vec}(\widehat{U}_j) \in \mathcal{B}_j^\Psi$, with $\mathcal{B}_j^\Phi := \{\theta_V \in \mathbb{R}^{L_{\Phi_j}L_{\Phi_{j+1}}} : \|\theta_V\| \leq \bar{V}\}$ and $\mathcal{B}_j^\Psi := \{\theta_U \in \mathbb{R}^{L_{\Psi_j}L_{\Psi_{j+1}}} : \|\theta_U\| \leq \bar{U}\}$.

The presence of the projection operator, along with $\boldsymbol{\mathcal{A}_2}$, allows to determine bounds for the residual terms in (19) and (23), i.e., $\sum_{j=1}^{k_\Phi}\Xi_{\Phi_j}\Delta_{\Phi_j}$, and $\text{vec}^{-1}(\Delta_{\Psi_{k_\Psi}})^{(i)} + \sum_{j=1}^{k_\Psi - 1}\Xi_{\Psi_j}^{(i)}\Delta_{\Psi_j}$. In particular, since $\text{proj}(\cdot)$ bounds the estimated weights, and the activation functions are chosen with bounded gradients, the norms of both Ξ_{Φ_j} and $\Xi_{\Psi_j}^{(i)}$ are bounded. Moreover, due to the fact that the terms of order two in the Taylor expansions are bounded as detailed in [13], there exist known constants $c_\Phi, c_\Psi \in \mathbb{R}_{>0}$ such that the inequalities $\left\|\sum_{j=1}^{k_\Phi}\Xi_{\Phi_j}\Delta_{\Phi_j}\right\| \leq c_\Phi$, and $\left\|\text{vec}^{-1}(\Delta_{\Psi_{k_\Psi}})^{(i)} + \sum_{j=1}^{k_\Psi - 1}\Xi_{\Psi_j}^{(i)}\Delta_{\Psi_j}\right\| \leq c_\Psi$ hold.

Using (1), (5), (6), (24), and (12), and since $\widetilde{\Phi}_{k_\Phi} = \Phi_{k_\Phi} - \widehat{\Phi}_{k_\Phi}$ and $\widetilde{\Psi}_{k_\Psi} = \Psi_{k_\Psi} - \widehat{\Psi}_{k_\Psi}$, one has

$$\begin{aligned}\dot{\sigma} =& \frac{\partial \sigma_0}{\partial x}\Big[\widetilde{\Phi}_{k_\Phi} + \varepsilon_\Phi + \text{vec}^{-1}(\widetilde{\Psi}_{k_\Psi} + \varepsilon_\Psi)u_0 + \\ &+ B(x)u_1 + h\Big].\end{aligned} \tag{28}$$

In the following, two main results are introduced for the specific cases of multi-input and single-input systems, respectively, which are however often used for modeling in many practical application domains.

A. The multi-input case (κm = n)

Now the stability analysis in the specific multi-input case $\kappa m = n$, with $\kappa \in \mathbb{N}_{>0}$ is discussed. For the considered kind of systems, let σ_0 such that

$$\frac{\partial \sigma_0}{\partial x} = \begin{bmatrix} G_1 & G_2 & \cdots & G_\kappa \end{bmatrix} \in \mathbb{R}^{m \times \kappa m}, \tag{29}$$

with $G_p = G_p^\top \in \mathbb{R}^{m \times m}$ being positive definite, while the following assumption holds.

$\boldsymbol{\mathcal{A}_3}$: The control effectiveness term $B(x) \in \mathbb{R}^{\kappa m \times m}$ can be written as $B(x) = \begin{bmatrix} B_1(x) & B_2(x) & \cdots & B_\kappa(x) \end{bmatrix}^\top$, with $B_p(x) = B_p^\top(x) \in \mathbb{R}^{m \times m}$ being positive semi-definite. Moreover, at least one B_p is positive definite and there exists a known constant $\gamma \in \mathbb{R}_{>0}$ such that $\underline{\lambda}(B_p) > \gamma$.

The following theorem, instrumental for the choice of the control gain ρ, is now introduced.

Theorem 1: Consider the nonlinear system (1) in the case of $\kappa m = n$, with $\kappa \in \mathbb{N}_{>0}$, control law u, sliding variable as in (3) and (24), and the weight adaptation laws (25), (26), and (27). If $\boldsymbol{\mathcal{A}_1}$, $\boldsymbol{\mathcal{A}_2}$, $\boldsymbol{\mathcal{A}_3}$, and (29) hold, and

$$\rho > \frac{\left\|\frac{\partial \sigma_0}{\partial x}\right\|\left[c_\Phi + m(c_\Psi + \bar{\varepsilon}_\Psi)\|u_0\| + \bar{\varepsilon}_\Phi + \bar{h}\right] + \bar{\eta}}{\min_p \underline{\lambda}(G_p)\gamma} \tag{30}$$

with $\bar{\eta} > 0$, then, a sliding mode $\sigma(t) = 0$ is enforced for $t \geq \bar{t} \geq 0$.

Proof: Consider a Lyapunov-like candidate function $v(x) : \mathbb{R}^n \to \mathbb{R}$ selected as

$$\begin{aligned}v(x) =& \frac{1}{2}\sigma^\top\sigma + \frac{1}{2}\sum_{j=0}^{k_\Phi}\text{vec}(\widetilde{V}_j)^\top\Gamma_{\Phi_j}^{-1}\text{vec}(\widetilde{V}_j) + \\ &+ \frac{1}{2}\sum_{j=0}^{k_\Psi}\text{vec}(\widetilde{U}_j)^\top\Gamma_{\Psi_j}^{-1}\text{vec}(\widetilde{U}_j),\end{aligned} \tag{31}$$

with time derivative equal to $\dot{v}(x) = \sigma^\top\dot{\sigma} - \sum_{j=0}^{k_\Phi}\text{vec}(\widetilde{V}_j)^\top\Gamma_{\Phi_j}^{-1}\text{vec}(\dot{\widehat{V}}_j) - \sum_{j=0}^{k_\Psi}\text{vec}(\widetilde{U}_j)^\top\Gamma_{\Psi_j}^{-1}\text{vec}(\dot{\widehat{U}}_j)$. Substituting (28), (19), (23), and expanding the last term of

the above equation, this can be rewritten as

$$\dot{v}(x) = \sigma^\top \frac{\partial \sigma_0}{\partial x} \Bigg\{ \sum_{j=0}^{k_\Phi} \Lambda_{\Phi_j} \mathrm{vec}(\widetilde{V}_j) + \sum_{j=1}^{k_\Phi} \Xi_{\Phi_j} \Delta_{\Phi_j} + \\ + \sum_{i=1}^{m} \Big[\Lambda^{(i)}_{\Psi_{k_\Psi}} \mathrm{vec}(\widetilde{U}^{(i)}_{k_\Psi}) + \mathrm{vec}^{-1}(\Delta_{\Psi_{k_\Psi}})^{(i)} + \sum_{j=1}^{k_\Psi - 1} \Xi^{(i)}_{\Psi_j} \Delta_{\Psi_j} + \\ + \sum_{j=0}^{k_\Psi - 1} \Lambda^{(i)}_{\Psi_j} \mathrm{vec}(\widetilde{U}_j) \Big] u_{0,i} + \varepsilon_\Phi + \mathrm{vec}^{-1}(\varepsilon_\Psi) u_0 + B(x) u_1 + \\ + h(t) \Bigg\} - \sum_{i=1}^{m} \mathrm{vec}(\widetilde{U}^{(i)}_{k_\Psi})^\top \Gamma^{-1}_{\Psi_{k_\Psi}} \mathrm{vec}(\dot{\widehat{U}}^{(i)}_{k_\Psi}) + \\ - \sum_{j=0}^{k_\Phi} \mathrm{vec}(\widetilde{V}_j)^\top \Gamma^{-1}_{\Phi_j} \mathrm{vec}(\dot{\widehat{V}}_j) - \sum_{j=0}^{k_\Psi - 1} \mathrm{vec}(\widetilde{U}_j)^\top \Gamma^{-1}_{\Psi_j} \mathrm{vec}(\dot{\widehat{U}}_j).$$

Then, since from [25] it holds that $-\theta^\top \Gamma^{-1} \mathrm{proj}(z) \leq -\theta^\top \Gamma^{-1} z$, if one applies the adaptation laws (25), (26), and (27), and substitutes (2), the above equation can be upper bounded as

$$\dot{v}(x) \leq \sigma^\top \frac{\partial \sigma_0}{\partial x} \Bigg\{ \sum_{j=1}^{k_\Phi} \Xi_{\Phi_j} \Delta_{\Phi_j} + \sum_{i=1}^{m} \Big[\mathrm{vec}^{-1}(\Delta_{\Psi_{k_\Psi}})^{(i)} + \\ + \sum_{j=1}^{k_\Psi - 1} \Xi^{(i)}_{\Psi_j} \Delta_{\Psi_j} \Big] u_{0_i} + \varepsilon_\Phi + \mathrm{vec}^{-1}(\varepsilon_\Psi) u_0 + h(t) \Bigg\} + \\ - \rho \sigma^\top \frac{\partial \sigma_0}{\partial x} B(x) \frac{\sigma}{\|\sigma\|}. \tag{32}$$

Moreover, if $\boldsymbol{\mathcal{A}_1}$, $\boldsymbol{\mathcal{A}_2}$, $\boldsymbol{\mathcal{A}_3}$, and (29) hold, equation (32) can be further bounded as

$$\dot{v}(x) \leq \|\sigma\| \left\| \frac{\partial \sigma_0}{\partial x} \right\| \Big\{ c_\Phi + m(c_\Psi + \bar{\varepsilon}_\Psi) \|u_0\| + \bar{\varepsilon}_\Phi + \bar{h} \Big\} + \\ - \|\sigma\| \rho \sum_{p=1}^{\kappa} \underline{\lambda}(G_p) \underline{\lambda}(B_p).$$

Hence, one obtains $\dot{v}(x) \leq \|\sigma\| \left\| \frac{\partial \sigma_0}{\partial x} \right\| \Big\{ c_\Phi + m(c_\Psi + \bar{\varepsilon}_\Psi) \|u_0\| + \bar{\varepsilon}_\Phi + \bar{h} \Big\} - \|\sigma\| \rho \min_p \underline{\lambda}(G_p) \gamma \leq -\eta \|\sigma\|$, with $\eta = \rho \min_p \underline{\lambda}(G_p) \gamma - \left\| \frac{\partial \sigma_0}{\partial x} \right\| [c_\Phi + m(c_\Psi + \bar{\varepsilon}_\Psi) \|u_0\| + \bar{\varepsilon}_\Phi + \bar{h}]$. If one designs ρ as in (30), then $\dot{v}(x) \leq -\eta \|\sigma\| < 0$. Exploiting the boundedness property guaranteed by the projection operator, the choice of a control gain as in (30) guarantees $\sigma = 0$ in a finite time $\bar{t} \geq 0$. ■

Note that the previous theorem also implies the boundedness of the weights of the DNNs. The convergence of such weights to their ideal values is beyond the scope of the paper, but we refer to classical paper as [26] for further discussion.

B. The single-input case m = 1

In this section, stability analysis in the case of scalar input, i.e., $m = 1$, is performed. The following assumption about the effectiveness control term can be introduced.

$\boldsymbol{\mathcal{A}_4}$: Given $B(x) = \begin{bmatrix} b_1 & b_2 & \cdots & b_n \end{bmatrix}^\top \in \mathbb{R}^n$, then $b_p \geq 0$, for $p = 1, 2, \ldots, n$. Moreover, there exists $\underline{b} \in \mathbb{R}_{>0}$ so that $\|B(x)\| \geq \underline{b}$, $\forall x \in \Omega$.

The following theorem can be now introduced.

Theorem 2: Consider the nonlinear system (1) in the case of $m = 1$, control law u, sliding variable as in (3) and (24), and the weight adaptation laws (25), (26), and (27). If $\boldsymbol{\mathcal{A}_1}$, $\boldsymbol{\mathcal{A}_2}$, $\boldsymbol{\mathcal{A}_4}$, and (29) hold, and

$$\rho > \frac{c_\Phi + (c_\Psi + \bar{\varepsilon}_\Psi)|u_0| + \bar{\varepsilon}_\Phi + \bar{h} + \bar{\eta}}{\underline{b}} \tag{33}$$

with $\bar{\eta} > 0$, then, a sliding mode $\sigma(t) = 0$ is enforced for $t \geq \bar{t} \geq 0$.

Proof: Consider the Lyapunov-like candidate function as in (31). Then, performing the same steps as in the previous section with $m = 1$ and $\kappa = n$, its derivative can be upper bounded as in (32). Then, if $\boldsymbol{\mathcal{A}_1}$, $\boldsymbol{\mathcal{A}_2}$, $\boldsymbol{\mathcal{A}_4}$, and (29) hold, $\dot{v}(x)$ can be bounded as

$$\dot{v}(x) \leq |\sigma| \left\| \frac{\partial \sigma_0}{\partial x} \right\| \Big\{ c_\Phi + (c_\Psi + \bar{\varepsilon}_\Psi)|u_0| + \bar{\varepsilon}_\Phi + \bar{h} \Big\} + \\ - \rho \left\| \frac{\partial \sigma_0}{\partial x} \right\| \underline{b} |\sigma| \leq -\eta |\sigma|,$$

where $\eta = \left\| \frac{\partial \sigma_0}{\partial x} \right\| [\rho \underline{b} - c_\Phi + (c_\Psi + \bar{\varepsilon}_\Psi)|u_0| + \bar{\varepsilon}_\Phi + \bar{h}]$. If one selects ρ so that condition (33) is satisfied, then $\dot{v}(x) \leq -\eta |\sigma| < 0$, which guarantees $\sigma = 0$ in a finite time $\bar{t} \geq 0$. ■

V. Numerical Examples

Simulations results on two systems with $m = n$ (i.e., $\kappa = 1$) and $m = 1$, respectively, are hereafter shown.

A. Multi-input case: the double tank system

To test the proposal in the case $m = n$, a double-tank system, inspired by [27], is considered, whose dynamics is

$$\begin{cases} \dot{x}_1 &= -\frac{a_1}{A_1}\sqrt{2g x_1} + \frac{a_2}{A_2}\sqrt{2g x_2} + \frac{1}{A_1} u_1 + h_1 \\ \dot{x}_2 &= -\frac{a_2}{A_2}\sqrt{2g x_2} + \frac{1}{A_2} u_2 + h_2, \end{cases} \tag{34}$$

where x_1, x_2, $A_1 = 0.28\,\mathrm{m}^2$, $A_2 = 0.32\,\mathrm{m}^2$, $a_1 = 0.007\,\mathrm{m}^2$ and $a_2 = 0.005\,\mathrm{m}^2$, represent the water levels, the cross-section of the tanks, and the cross-section of the output valves, respectively, and $g = 9.8\,\mathrm{m/s}^2$. The disturbance terms are $h_1 = 0.4\cos(4t)$ and $h_2 = 0.25\sin(0.5t) + 0.125\cos(t)$. The DNNs are characterized by, $k_\Phi = 3$, with $L_{\Phi_0} = 3$, $L_{\Phi_1} = 10$, $L_{\Phi_2} = 50$, $L_{\Phi_3} = 20$, $L_{\Phi_4} = 2$, while $k_\Psi = 3$, with $L_{\Psi_0} = 3$, $L_{\Psi_1} = 10$, $L_{\Psi_2} = 100$, $L_{\Psi_3} = 20$, $L_{\Psi_4} = 4$. Moreover, $\Gamma_{\Phi_j} = 100 \cdot I_{L_{\Phi_j}}$ and $\Gamma_{\Psi_j} = 100 \cdot I_{L_{\Psi_j}}$. The stabilizing control law has been chosen as $u_0 = (\mathrm{vec}^{-1}(\widehat{\Psi}))^+ [-\widehat{\Phi} - K(x - x^*)]$, with $K = 2 \cdot I_2$, and $x^* = \begin{bmatrix} 0.75 & 0.4 \end{bmatrix}^\top$. The sliding variable is chosen so that $\sigma_0 = (x - x^*)$. Finally, given $\bar{h} = 0.6$, finding $c_\Phi + \bar{\varepsilon}_\Phi + \bar{h} = 3.6$ and $c_\Psi + \bar{\varepsilon}_\Psi = 0.15$, and $\bar{\eta} = 0.3$, the control gain is selected as $\rho = 1.3 + 0.05|u_0|$, which satisfies (30). The system has been simulated for $20\,\mathrm{s}$ with a time-step of $0.005\,\mathrm{s}$ and $x_0 = \begin{bmatrix} 0.5 & 0.5 \end{bmatrix}^\top$. The results of the simulation are presented in Fig. 2 and Fig. 3, in which it is possible to see that condition $\sigma = 0$ is lost for a transient due to the adaptation of the weights, and then it is again enforced, as expected from Theorem 1. Moreover, the system is correctly controlled towards the desired set-point.

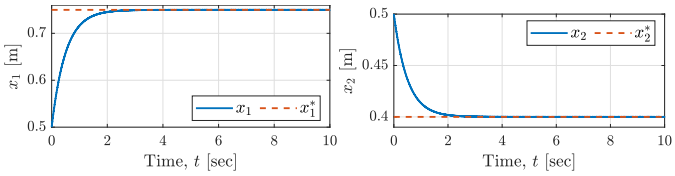


Fig. 2: Water level in the tanks.

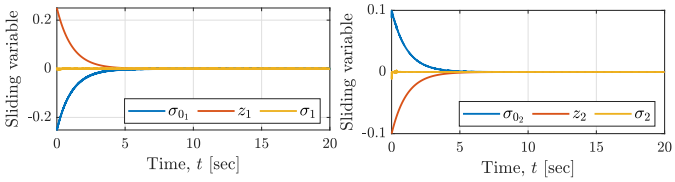


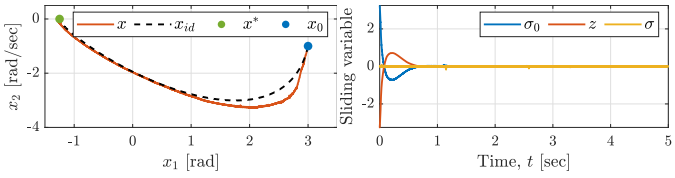
Fig. 3: Sliding variable in the double-tank simulation.

B. Scalar case: the Duffing oscillator

To test the proposed algorithm in the single-input case, i.e., $m = 1$, the model of the Duffing oscillator [1, Chapter 1], has been used. The dynamics are given by

$$\begin{cases} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_1(1 - x_1^2) - x_2 + u + h_2, \end{cases} \tag{35}$$

with x_1 and x_2 being the position and the velocity of the mass, respectively, while $h_2 = 0.5\sin(0.5t)+0.25\cos(t)$. The DNNs are structured so that $k_\Phi = k_\Psi = 5$, with $L_{\Phi_0} = L_{\Psi_0} = 3$, $L_{\Phi_1} = L_{\Psi_1} = 20$, $L_{\Phi_2} = L_{\Psi_2} = 50$, $L_{\Phi_3} = L_{\Psi_3} = 50$, $L_{\Phi_4} = L_{\Psi_4} = 50$, $L_{\Phi_5} = L_{\Psi_5} = 20$, $L_{\Phi_6} = L_{\Psi_6} = 2$. Moreover, $\Gamma_{\Phi_j} = 550 \cdot I_{L_{\Phi_j}}$ and $\Gamma_{\Psi_j} = 550 \cdot I_{L_{\Psi_j}}$. The stabilizing control law has been chosen as $u_0 = \widehat{\Psi}^+[-\widehat{\Phi} - K(x - x^*)]$, with $K = \begin{bmatrix} 3 & 3 \end{bmatrix}$ and $x^* = \begin{bmatrix} -1.25 & 0 \end{bmatrix}^\top$. The initial condition is $x_0 = \begin{bmatrix} 3 & -1 \end{bmatrix}^\top$. The sliding variable is chosen so that $\sigma_0 = \begin{bmatrix} 1 & 1 \end{bmatrix}(x - x^*)$. Moreover, given $\bar{h} = 0.75$, finding $c_\Phi + \bar{\varepsilon}_\Phi + \bar{h} = 1.9$ and $c_\Psi + \bar{\varepsilon}_\Psi = 0.5$, $\bar{\eta} = 0.1$, the control gain is selected as $\rho = 2 + 0.5|u_0|$, which satisfies (33). The system has been simulated for 5 s, with a time-step of 0.0001 s. The results of the simulation are presented in Fig. 4. From the two pictures it is possible to see that that condition $\sigma = 0$ is lost for a very short transient due to the adaptation of the weights, and then it is again enforced, as expected from Theorem 2.



(a) System states (b) Sliding variable

Fig. 4: Outcome of the Duffing oscillator simulation in terms of state phase portrait and sliding variable, with x_{id} being the state in the case of fully known dynamics and $\rho = 1$.

VI. Conclusions

In this paper, we propose a DNN-ISM control algorithm for nonlinear systems in presence of external disturbances and in the case of fully unknown dynamics. In particular, the unknown drift term and control effectiveness matrix are estimated relying on two DNNs with an arbitrary number of hidden layers. The weights of the DNNs are adjusted according to adaptive laws derived from the stability analysis, relying on two different classes of systems. Finally, the proposal has been satisfactorily assessed in simulation on a double-tank system and on the classic Duffing oscillator.

References

[1] A. Ferrara, G. P. Incremona, and M. Cucuzzella, *Advanced and Optimization Based Sliding Mode Control: Theory and Applications*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2019.

[2] G. Bartolini, A. Ferrara, A. Levant, and E. Usai, "On second order sliding mode controllers," *Variable structure systems, sliding mode and nonlinear control*, pp. 329–350, 1999.

[3] A. Levant, "Higher-order sliding modes, differentiation and output-feedback control," *International Journal of Control*, vol. 76, no. 9-10, pp. 924–941, 2003.

[4] G. Bartolini, A. Ferrara, L. Giacomini, and E. Usai, "A combined backstepping/second order sliding mode approach to control a class of nonlinear systems," in *Proceedings. 1996 IEEE International Workshop on Variable Structure Systems.-VSS'96-*. IEEE, 1996, pp. 205–210.

[5] A. Pisano, M. Tanelli, and A. Ferrara, "Switched/time-based adaptation for second-order sliding mode control," *Automatica*, vol. 64, pp. 126–132, 2016.

[6] G. P. Incremona, L. Mirkin, and P. Colaneri, "Integral sliding-mode control with internal model: A separation," *IEEE Control Systems Letters*, vol. 6, pp. 446–451, 2021.

[7] V. Utkin and J. Shi, "Integral sliding mode in systems operating under uncertainty conditions," in *35th IEEE Conference on Decision and Control*, vol. 4, Kobe, Japan, Dec. 1996, pp. 4591–4596.

[8] M. Rubagotti, A. Estrada, F. Castaños, A. Ferrara, and L. Fridman, "Integral sliding mode control for nonlinear systems with matched and unmatched perturbations," *IEEE Transactions on Automatic Control*, vol. 56, no. 11, pp. 2699–2704, 2011.

[9] A. Ferrara and G. P. Incremona, "Design of an integral suboptimal second-order sliding mode controller for the robust motion control of robot manipulators," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 6, pp. 2316–2325, 2015.

[10] A. Ferrara, G. P. Incremona, and B. Sangiovanni, "Tracking control via switched integral sliding mode with application to robot manipulators," *Control Engineering Practice*, vol. 90, pp. 257–266, 2019.

[11] B. Karg and S. Lucia, "Efficient representation and approximation of model predictive control laws via deep learning," *IEEE Transactions on Cybernetics*, vol. 50, no. 9, pp. 3866–3878, 2020.

[12] F. Bonassi, J. Xie, M. Farina, and R. Scattolini, "Towards lifelong learning of recurrent neural networks for control design," in *2022 European Control Conference (ECC)*. IEEE, 2022, pp. 2018–2023.

[13] F. L. Lewis, A. Yesildirek, and K. Liu, "Multilayer neural-net robot controller with guaranteed tracking performance," *IEEE Transactions on Neural Networks*, vol. 7, no. 2, pp. 388–399, 1996.

[14] L. Cheng, Z. Wang, F. Jiang, and J. Li, "Adaptive neural network control of nonlinear systems with unknown dynamics," *Advances in Space Research*, vol. 67, no. 3, pp. 1114–1123, 2021.

[15] O. S. Patil, D. M. Le, M. L. Greene, and W. E. Dixon, "Lyapunov-derived control and adaptive update laws for inner and outer layer weights of a deep neural network," *IEEE Control Systems Letters*, vol. 6, pp. 1855–1860, 2021.

[16] Y. Chu, J. Fei, and S. Hou, "Adaptive global sliding-mode control for dynamic systems using double hidden layer recurrent neural network structure," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 4, pp. 1297–1309, 2019.

[17] Y. Wang and J. Zhao, "Neural-network-based event-triggered sliding mode control for networked switched linear systems with the unknown nonlinear disturbance," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[18] A. Bhatti, S. Spurgeon, and X. Lu, "A nonlinear sliding mode control design approach based on neural network modelling," *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, vol. 9, no. 7, pp. 397–423, 1999.

[19] N. Sacchi, G. P. Incremona, and A. Ferrara, "Neural network-based practical/ideal integral sliding mode control," *IEEE Control Systems Letters*, vol. 6, pp. 3140–3145, 2022.
[20] J. Hastad, "Almost optimal lower bounds for small depth circuits," in *Proceedings of the eighteenth annual ACM symposium on Theory of computing*, 1986, pp. 6–20.
[21] R. Eldan and O. Shamir, "The power of depth for feedforward neural networks," in *Conference on learning theory*. PMLR, 2016, pp. 907–940.
[22] S. Liang, "Why deep neural networks for function approximation?" *arXiv preprint arXiv:1610.04161*, 2016.
[23] Z. Lu, H. Pu, F. Wang, Z. Hu, and L. Wang, "The expressive power of neural networks: A view from the width," *Advances in Neural Information Processing Systems*, vol. 30, 2017.
[24] D. S. Bernstein, "Matrix mathematics," in *Matrix Mathematics*. Princeton university press, 2009.
[25] M. Krstic, P. V. Kokotovic, and I. Kanellakopoulos, *Nonlinear and adaptive control design*. John Wiley & Sons, Inc., 1995.
[26] S. Lu and T. Basar, "Robust nonlinear system identification using neural-network models," *IEEE Transactions on Neural networks*, vol. 9, no. 3, pp. 407–429, 1998.
[27] K. H. Johansson, "The quadruple-tank process: A multivariable laboratory process with an adjustable zero," *IEEE Transactions on Control Systems Technology*, vol. 8, no. 3, pp. 456–465, 2000.