IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 7 March 2024, accepted 24 March 2024, date of publication 4 April 2024, date of current version 25 April 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3385236

RESEARCH ARTICLE

Incorporating Recurrent Networks for Online System Identification Alongside Traditional Sine-Sweep Experiments

C. A. L. SEGURA, (Member, IEEE)
Escuela de Ingenieros Industriales y Civiles EIIC Universidad de Las Palmas de Gran Canaria ULPGC, 35014 Las Palmas, Spain
e-mail: lopezseguracesarantonio@gmail.com

ABSTRACT The experimental identification of an unknown system, and the blind system identification (BSI) methods, allows engineers to establish mathematical models that represent the real system behavior. However, when the system operates in a non-stationary environments influenced by external disturbances, models with adaptive properties are required for predicting the real-time domain response. This study defines and analyzes in detail two system identification methods. The first method, which operates offline and requires post-processing, is mathematically defined to achieve the highest level of automation. It is based on sine sweep theory and involves conducting long-term experiments on a real system to determine its frequency domain properties. The second method, which operates online, employs computational learning theory and information theory to predict the system response through online learning. This modern approach uses convex optimization to obtain the optimal parameters of a time-lagged recurrent network (TLRN) in each iteration, which incorporates, among other features, a gamma filter as a mapper. This iterative online method was mathematically described addressing stability, convergence, and disturbances issues.

INDEX TERMS Adaptive models, online learning, artificial neural networks, time-lagged recurrent network (TLRN), recurrent neural network (RNN).

I. INTRODUCTION

The identification of electromechanical systems can be approached through analytical or empirical methods. Analytical methods, based on the development of physical laws, are particularly useful in the design stage. On the other hand, empirical methods are employed by engineers when the system has been constructed, and efficient extraction of computational models is required for transient and stationary response analysis. Within empirical methods, we pay special attention to those executed offline and online, with the latter allowing the designer to characterize the plant in real time and extract the map that models the system.

Furthermore, the design of controllers for stable electromechanical systems heavily relies on accurately mathematically characterizing the system, especially when references, noise, and perturbations are distributed across a wide frequency spectrum. In the control design stage, the objective is to establish analytically either global exponential or global asymptotic tracking (GET, GAT), or at least local exponential or asymptotic reference tracking. Additionally, when adaptive controller designs require the asymptotic identification of time-varying uncertainties, it is common to integrate a machine learning model to automatically identify these uncertainties

In this paper, we analyze the fundamentals of two system identification (SI) techniques and discuss their respective performances. These techniques serve as indispensable tools for analyzing system responses, designing digital controllers, and extracting the relevant information that allows the characterization of a real system.

When an electromechanical system exhibits the properties of superposition and homogeneity, traditional system identification techniques involve exciting the system with input commands. Subsequently, the output states, measured by sensors, are processed using offline linearization methods. In contrast, in fields such as machine learning, which adhere to the universal approximation theorem, system identification

The associate editor coordinating the review of this manuscript and approving it for publication was Ikramullah Lali.

focuses on solving regression problems. Here, a mapping function based on artificial neural networks (ANN) is employed. These networks are trained using offline methods (full-batch or mini-batch) or online learning techniques. This approach allows replacing a black-box or unknown model with a stable white-box model.

In this study, a stable electromechanical system was identified using two SI techniques. The first method was the sine sweep, which is based on sinusoidal experiments and automatic offline data processing. The second was an online learning approach where the parameters of the real system are obtained in each iteration through stochastic gradient descent. Additionally, the online iterative method was evaluated in terms of disturbances, convergence, and stability to demonstrate its real performance.

II. RELATED WORK

The rapid advancement of machine learning and control systems has led to a proliferation of studies addressing various aspects of system-identification techniques. In this section, we review and compare key contributions and highlight the evolution of research in this domain.

A. EXTRACTION OF LINEAR SYSTEMS MODELS

The identification of linear or linearizable systems at specific operating points, based on the detailed extraction of the frequency domain response through experimental methods, is currently being addressed [9], [11], [24], [25], [26]. Accurate extraction of the frequency domain response of a system during the control design stage is often underestimated. This underestimation is attributed to the robustness to parametric uncertainty demonstrated by closed-loop feedback control, a characteristic that has made it popular in the industry.

B. ADAPTIVE DESIGNS WITH ARTIFICIAL NEURAL NETWORKS (ANN)

The design of system state trajectory estimators, the utilization of recurrent neural networks in control system structures, and approaches involving reinforcement learning as well as heuristic dynamic programming (HDP), are abundant in the state-of-the-art [12], [17], [18], [19], [20], [27]. An interesting example in this field is the ''truck backer-upper'' published in April 1990 [15].

C. ADAPTIVE NETWORK DESIGN AND PARAMETRIZATION

Adaptive control design, with a derived structure based on the Lyapunov stability theorem, including real-time estimation of both structured and non-structured parametric uncertainties using neural networks, has been evaluated and published in the last two decades [5], [6], [7]. The prediction based on filters and plant estimation in stochastic control theory was introduced in [30].

The parameterization of artificial neural networks for use in control system architectures, as well as in online system identification, can be carried out through analytical resolution, or through non-linear iterative methods such as Newton's method or gradient descent and its variations [1], [2], [3], [4], [8].

D. COST FUNCTION MECHANISM IN ADAPTIVE SOLUTIONS

When pursuing an iterative adaptive solution, we recognize that the heart of an adaptive system is the cost function. In this study, the cost function mechanism is based on the mean squared error (MSE), which is suitable when the prediction error follows a Gaussian distribution. On the other hand, researchers have proposed alternatives to MSE to reduce prediction errors [14], [16]. Moreover, when reduced memory and stationary parameters are required in non-linear prediction, researchers tend to favor the new generation of online learning algorithms defined in the reproducing kernel Hilbert space (RKHS) [10].

E. ADAPTIVE FILTERING IN RECURRENT NEURAL NETWORK ARCHITECTURES

The inclusion of adaptive filtering [21] as an activation function in recurrent neural network (RNN) architectures is a well-known practice and has been extensively discussed in several studies [22], [23].

III. DESCRIPTION

In this section, two system identification methods (III-A and III-B) were described in detailed mathematical terms. The first method, sine sweep transfer function estimation (III-A), was explained in four steps (III-A1, III-A2, III-A3, and III-A4), while the second method, online-learning identification (III-B), was explained in three subsections (III-B1, III-B2, and III-B3). Finally, computer simulations were developed in section III-C.

A. SINE SWEEP TRANSFER FUNCTION ESTIMATION

1) SINUSOIDAL EXPERIMENTS OVER THE REAL SYSTEM

A Bounded-Input Bounded-Output (BIBO) stable electromechanical linear system was subjected to an input $\mathbf{u_0}(\mathbf{t_0}) = A\sin(\alpha t_0)$. The observable output state of the system was measured using a sensor, and the output response took the form $\mathbf{y_0}(\mathbf{t_0}) = A|G(j\alpha)|\sin(\alpha t_0 + \angle G(j\alpha)) + \epsilon$. Here, $G(j\alpha)$ represents the unknown transfer function of the system, and ϵ denotes an error that tends to zero as $t \to \infty$. The parameter α is defined as $\alpha = 2\pi f$. This process was repeated for various input frequencies α_i, and each input-output pair was recorded in matrices U_0 and Y_0, respectively, for different input frequencies α.

2) EXPANDING OR INCREASING THE DIMENSIONALITY OF DATA

When conducting long-term experiments on a real system becomes infeasible due to operational limitations or because the system response is influenced by non-constant random variables over extended periods, it may be necessary to

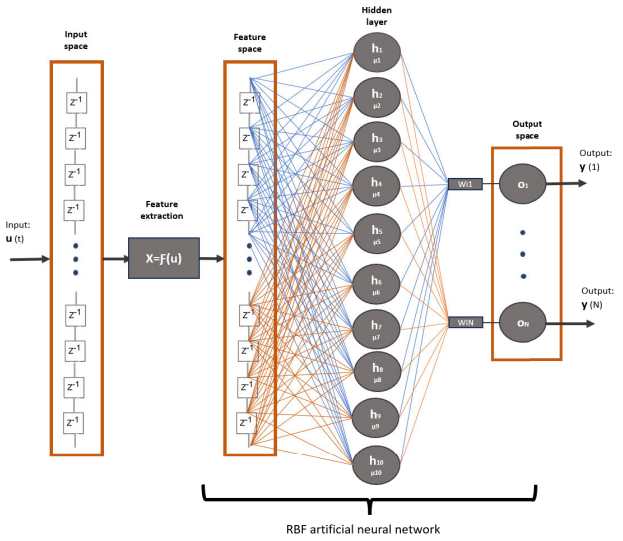

FIGURE 1. A radial basis artificial neural network (ANN) was employed to predict $y(t)$. The network was trained using the signals $y_0(t^0)$ and $u_0(t^0)$, where $t^0 \ll t$.

enhance the dimensionality of the sensor data. This involves transforming the original data from $y_0 \in \mathbb{R}^{N_0}$ to $y \in \mathbb{R}^N$, where $N_0 \ll N$.

The accuracy of the magnitude estimation, denoted as $\hat{g} \approx |G(j\alpha)|$, and the phase estimation, denoted as $\hat{\varphi} \approx \angle G(j\alpha)$), improves as the dimensionality increases converging to $\epsilon \to 0$ as $t \to \infty$.

The challenge of increasing dimensionality is a problem addressed in fields such as machine learning [13]. The solution involves selecting the network architecture, training it using an iterative method, a direct method based on cost function derivation, or a combination of both, and then making predictions through the trained network. The approach proposed in this study utilizes a radial basis network (RBN) and evaluates its performance when the network is fed with either the raw input data or with feature extraction applied to the input domain.

The training process involved using the input-output matrices as features and targets, denoted as $X_0 = U_0 \in \mathbb{R}^{K\times N_0}$ and $T_0 = Y_0 \in \mathbb{R}^{K\times N_0}$, respectively, to train the network when K is the number of experiments with each frequency α, N_0 and N represents the dimensionality before and after the dimensionality expansion. These matrices were recorded during the experimental stage (section III-A1). Subsequently, the network was used to predict the matrix $Y \in \mathbb{R}^{K\times N}$ when the input matrix $X = U \in \mathbb{R}^{K\times N}$ was introduced (Figure 1).

The optimal radial basis network parameters, denoted by w, minimize the cost function in Eq. 2, can be derived and solved in only one iteration.

The hidden layer of the network was implemented with Gaussian radial basis functions, defined as: $\phi_j^{RBF} = \exp\left(-\frac{(x-\mu_j)^2}{2s^2}\right)$, where s is a hyperparameter, and $\mathbf{x}$ represents the input feature.

The output $\mathbf{y}$ is defined as the weighted sum, that is:

$$\mathbf{y_j} = \sum_{i=1}^{L} w_i \Phi_{ij}^{RBF} \tag{1}$$

where 'L' represents the total number of μ centers and the optimal parameters $\boldsymbol{w}$ are derived from a cost function (2) that includes both the Lasso and Ridge regression terms.

$$E(w) = \frac{1}{2} \underbrace{\|\phi^{\mathbf{RBF}}\mathbf{w} - \mathbf{t_{target}}\|_2^2}_{\text{data dependent convex cost}} + \underbrace{\beta^*(\lambda_1\|\mathbf{w}\|_1 + \lambda_2\|\mathbf{w}\|_2^2)}_{\text{elastic net regularization}} \tag{2}$$

Denote the RBF features ϕ^{RBF} as ϕ for simplification. Then, the optimal parameters w will be:

$$(\phi^{\mathbf{T}}\phi + 2\beta^*\lambda_2)\mathbf{w} + \beta^*\lambda_1 sgn(w) - \mathbf{t_{target}}\phi^{\mathbf{T}} = \frac{\partial E(w)}{\partial w} = 0 \tag{3}$$

The last equation Eq. 3 reveals that $\frac{\partial E(w)}{\partial w}$ depends on the term sgn(w). This implies that the parameter resolution is affected by the initialization dependence issue, leading to three possible initial values for obtaining the optimal parameters:

1) If $w < 0$, then $sgn(w) = -1$

$$\begin{aligned}(\phi^{\mathbf{T}}\phi + 2\beta^*\lambda_2)\mathbf{w} - \beta^*\lambda_1 - \mathbf{t_{target}}\phi^{\mathbf{T}} &= 0\\ \mathbf{w} &= (\phi^{\mathbf{T}}\phi + 2\beta^*\lambda_2)^{-1}(\beta^*\lambda_1 + \mathbf{t_{target}}\phi^{\mathbf{T}})\end{aligned} \tag{4}$$

2) If $w = 0$, then $sgn(w) = 0$

$$\begin{aligned}(\phi^{\mathbf{T}}\phi + 2\beta^*\lambda_2)\mathbf{w} - \mathbf{t_{target}}\phi^{\mathbf{T}} &= 0\\ \mathbf{w} &= (\phi^{\mathbf{T}}\phi + 2\beta^*\lambda_2)^{-1}(\mathbf{t_{target}}\phi^{\mathbf{T}})\end{aligned} \tag{5}$$

3) If $w > 0$, then $sgn(w) = +1$

$$\begin{aligned}(\phi^{\mathbf{T}}\phi + 2\beta^*\lambda_2)\mathbf{w} + \beta^*\lambda_1 - \mathbf{t_{target}}\phi^{\mathbf{T}} &= 0\\ \mathbf{w} &= (\phi^{\mathbf{T}}\phi + 2\beta^*\lambda_2)^{-1}(\mathbf{t_{target}}\phi^{\mathbf{T}} - \beta^*\lambda_1)\end{aligned} \tag{6}$$

In addition, two hyperparameters are adjusted by designers λ_1 and β^*, considering $\lambda_2 = 1 - \lambda_1$. Finally, the parameter vector $\boldsymbol{w}$ that minimized the identification error across all possible initializations was selected as the optimal set.

3) MAGNITUDE AND PHASE ESTIMATION

This section describes the process of determining both the magnitude (amplitude) and the phase of a signal or a component within a signal. This estimation is commonly performed in various fields, such as signal processing, control systems, and electrical engineering [9]. The method described here allows for magnitude and phase estimation even when sensor measurements are noisy.

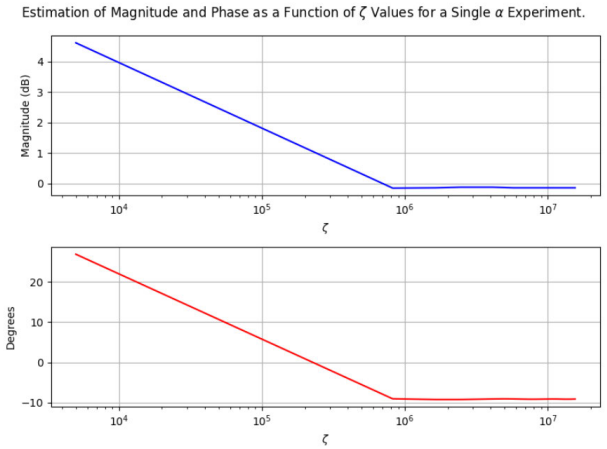

FIGURE 2. The magnitude $\hat{g}$ and phase $\hat{\varphi}$ estimation for one experiment, denoted as α_i, involved short-term experiments with $\zeta_0 = 5000$, increasing their dimensionality with ANN to $\zeta > 10^7$. The estimation reached steady-state when $\zeta \gg \zeta_0$, that means a large number of cycles *NP*.

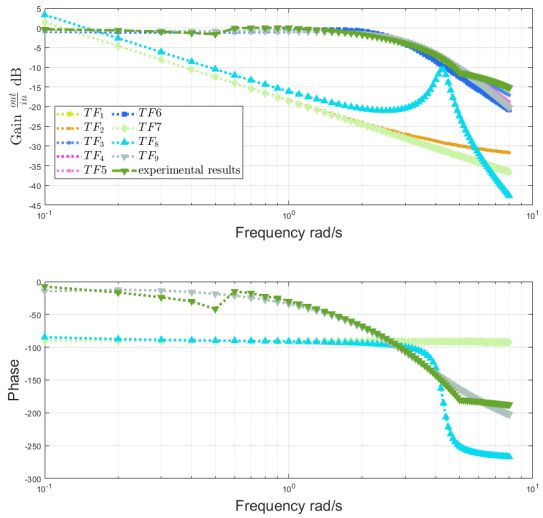

FIGURE 3. Several transfer functions (TF) have been proposed to identify the experimental data magnitude and phase. The relative degrees (r.d) evaluated were 1, 2 and 3.

Then, the magnitude $g = |G(j\alpha)|$ and phase $\varphi = \angle G(j\alpha)$ could be estimated as:

$$\hat{g} \triangleq \frac{2}{A}\sqrt{I_p^2 + I_q^2} \tag{7}$$

$$\hat{\varphi} \triangleq \tan^{-1}\left(\frac{I_q}{I_p}\right) \tag{8}$$

where $I_p = \int_0^{NP} y(t)s(t)\,dt$ and, $I_q = \int_0^{NP} y(t)c(t)\,dt$. Additionally, the in-phase term $s(t)$ and quadrature term $c(t)$ are defined as follows:

$$s(t) \triangleq \sin(\alpha t) \tag{9}$$

$$c(t) \triangleq \cos(\alpha t) \tag{10}$$

The accuracy of estimation requires that $NP = \frac{2\pi}{\alpha_i}\zeta$. Denote $NP_0 = \frac{2\pi}{\alpha_i}\zeta_0$ as the actual number of periods used in each real short-time sinusoidal experiment. As explored in this paper, an alternative to unfeasible long-term experiments is to introduce a small value ζ_0, indicating short-time duration experiments using $u_0(t)$ and $y_0(t)$ when $\zeta_0 \ll \zeta$. Subsequently, the long-term output $y(t)$ can be estimated using a trained radial basis artificial neural network (as discussed in Section III-A2). It is necessary that the integer ζ be significantly greater than the integer ζ_0 to ensure estimation accuracy as has been demonstrated in Figure 2.

4) TRANSFER FUNCTION IDENTIFICATION

Using the experimental magnitude and phase determined by the sine sweep method explained in Section III-A, and applying the dimensionality-increasing technique detailed in Section III-A2, it is possible to determine the transfer function or state-space model that minimizes the error between the target magnitude-phase (m_t, ph_t) and, the transfer function magnitude-phase response (m, ph) across a range of frequencies denoted by α with a total of K frequency points. The error function is represented by Eq. 11.

$$E = \frac{\sum_{i=1}^{K}[m(\alpha_i) - m_t(\alpha_i)]^2 + [ph(\alpha_i) - ph_t(\alpha_i)]^2}{K} \tag{11}$$

The transfer function that minimizes Eq. 11 has the form of Eq. 12:

$$H(s) = \frac{B(s)}{A(s)} = \frac{b_1 s^z + b_2 s^{z-1} + \ldots + b_{z+1}s^0}{a_1 s^p + a_2 s^{p-1} + \ldots + a_{p+1}s^0} \tag{12}$$

If we define $h(k) = m_t(k)e^{jph_t(k)}$, and let $A(\alpha(k))$ and $B(\alpha(k))$ denote the Fourier transforms of the polynomials a and b at the frequency $\alpha(k)$ respectively, and n be the number of frequency points, it is possible to find the best stable model $H(s)$ for a and b with a relative degree $r.d = p - z$ that minimizes Eq. 13 and 14, as demonstrated in [28], [29].

$$\min_{a,b} \sum_{k=1}^{n} \alpha t(k)\,|h(k)A(\alpha(k)) - B(\alpha(k))|^2 \tag{13}$$

$$\min_{a,b} \sum_{k=1}^{n} \alpha t(k) \left|h(k) - \frac{B(\alpha(k))}{A(\alpha(k))}\right|^2 \tag{14}$$

Figures 3 and Table 1 show the linearization sensitivity analysis. Finally, we can conclude that transfer functions 3, 4, and 9 produced accurate results compared to the magnitude and phase of the real system in the analyzed frequency interval.

It was demonstrated that it is possible to obtain a reduced-order linear model that identifies the stationary frequency response of the studied system. During this process, it was necessary to increase the dimensions of the experimental dataset through the resolution of a regression problem using a radial basis ANN that was trained with a reduced dimension. The dimensionality-increase method allowed us to minimize the time spent in the experimental

TABLE 1. Transfer function (TF) and relative degree (r.d) versus the normalized system identification mean-squared error (MSE) in terms of magnitude and phase (eq. 11). In general, increasing the relative degree minimizes the identification error. Transfer function TF_3, $(-0.6509s^2 + 7.13s + 0.1684)/(s^3 + 3.63s^2 + 8.215s + 0.01086)$, minimized MSE and r.d..

TF	MSE	r.d	TF	MSE	r.d
1	0.0365	1	6	0.9266	2
2	0.7205	1	7	0.0374	2
3	0.0006	1	8	0.0275	3
4	0.0005	2	9	0.0005	3
5	1	3			

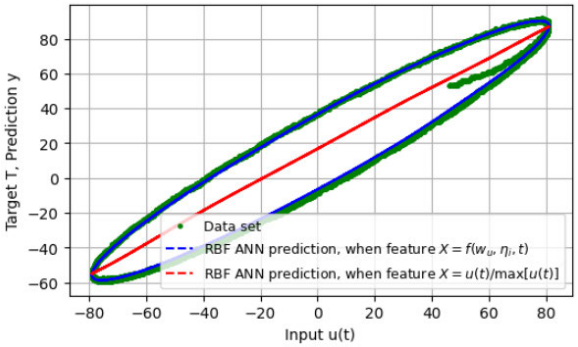

FIGURE 4. Regression analysis involves predicting a curved trajectory in the input-output plane. When the feature is the normalized raw input $X = u(t)_n$, the network prediction is a straight line in the input-output plane. However, when feature extraction $X = f[u(t)]$ is applied using Equation 15, the prediction closely resembles the curve of the dataset.

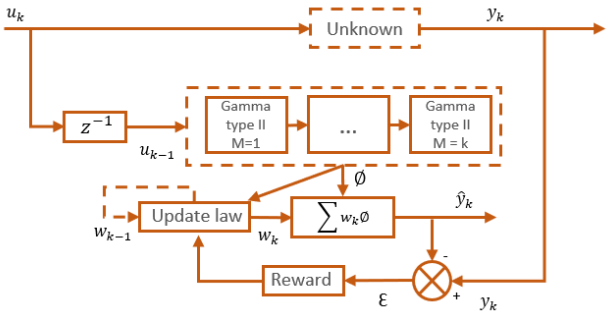

FIGURE 5. An M=k order Gamma Type II Time-Lagged Recurrent Network is used to predict the plant response $\hat{y}k$ based on the past input u_{k-1}. The iterative method for updating the parameters involves computing the reward based on the error ξ and then applying the stable update law.

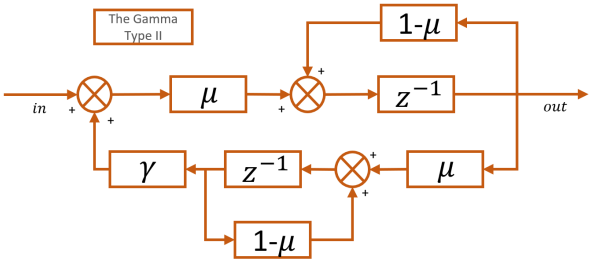

FIGURE 6. The z discrete domain Gamma Type II architecture is employed in the TLRN to identify $\hat{y}$. Two hyperparameters, μ and γ, are utilized. This configuration is equivalent to a Finite Impulse Response (FIR) filter when $\mu = 1$ and $\gamma = 0$. Stable $\forall\gamma, \mu \in \mathbb{R} : |1 - \mu \pm \mu\sqrt{\gamma}| < 1$

stage on the real system and improve the magnitude and phase estimation, reaching steady-state values.

When discussing feature extraction, several common techniques can be applied, including statistical shape descriptors and time-series features. However, in this case, a tailored strategy was employed, as outlined in Eq. 15:

$$X = \frac{\sin(t\alpha_{u(t)} + \eta_1) + \ldots + \sin(t\alpha_{u(t)} + \eta_n)}{max[\sin(t\alpha_{u(t)} + \eta_1) + \ldots + \sin(t\alpha_{u(t)} + \eta_n)]} = \frac{\sum_{i=1}^{n} \sin(t\alpha_{u(t)} + \eta_i)}{\max[\sum_{i=1}^{n} \sin(t\alpha_{u(t)} + \eta_i)]} = \frac{\psi(t, \alpha_u, \eta)}{max[\psi(t, \alpha_u, \eta)]} \quad (15)$$

where η_i is a hyperparameter defined in radians, $\alpha_{u(t)}$ is the frequency in radians of the input signal $u(t)$, and $\mathbf{t}$ is the time vector.

Figure 4 shows in the input-output space, the predictions of the radial basis network during the training process for the frequency α_i. Two cases were considered: first, when the features are equal to the raw input $u(t)$, and second, when Equation 15 is applied for feature extraction.

B. ONLINE-LEARNING IDENTIFICATION

1) INTRODUCTION

In this case, and considering the linear scenario with a discrete input $u(k) = A\sin[\alpha t(k)]$ and the plant output feedback obtained from the sensor measurement $y(k) = A|G(j\alpha)|\sin[\alpha t(k) + \angle G(j\alpha)] + \epsilon$, it is possible to determine the TLRN parameters that predict the sensor measurement $y(k)$.

Three essential tools are necessary: first, the cost function, mapper, and an online stable update law.

In this study, we explore the architecture of a TLRN mapper (Figure 5). Specifically, we delve into a TLRN and its integration with Type II Gamma Filters. This architecture presents two additional hyperparameters that improve performance and enhance gradient descent compared to simpler architectures such as TLRNs based on FIR filters. FIR filters represent a special case of the Gamma Type II architecture when $\mu = 1$ and $\gamma = 0$. Notably, a combination of these architectures is considered an online deep learning algorithm (Figure 6). The filter hyperparameters γ, μ have the next stability condition defined in Eq. 16.

$$\forall\gamma, \mu \in \mathbb{R} : |1 - \mu \pm \mu\sqrt{\gamma}| < 1 \quad (16)$$

The ability of the online learning RNN to predict the plant output y_k using past information from the input u_{k-1} was tested with a swept-frequency cosine input signal, or an up-chirp signal. The mapper can learn both plant transients and stationary responses, enabling it to adapt to dynamic changes in the system behavior and accurately model its steady-state characteristics (Figure 7).

2) ONLINE PARAMETERS ESTIMATION

Consider random variable y_k, and its estimator $\hat{y}_k$. We defined the expected value of the squared L2 norm of the prediction

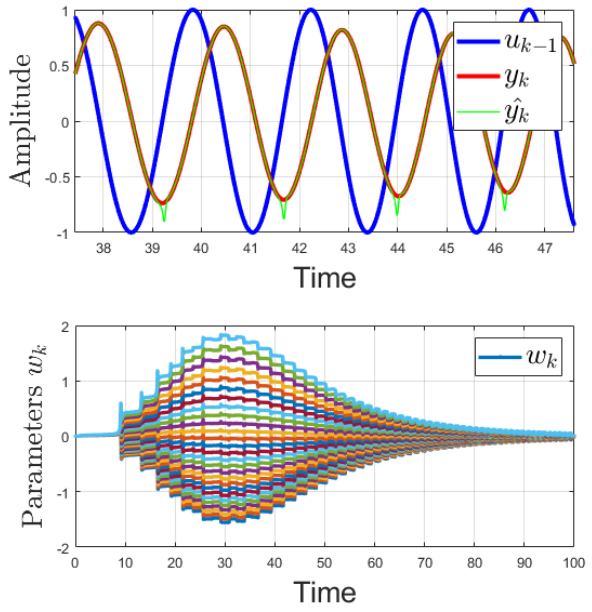

FIGURE 7. Upper Figure - Up-Chirp Test: Evaluation of RNN performance. The graph illustrates the RNN's predictive ability during the up-chirp test. The plot compares the RNN's predicted plant output ($\hat{y}_k = \sum w\phi$) against the actual plant responses y_k, showcasing its capacity to adapt to rapid variations in input frequencies. Bottom Figure - Recurrent Network Parameters This figure visualizes the key architectural components and parameters of the RNN, including weights and biases. The network order is denoted as M=27.

error as $J(w) = E[\xi^2] = E[\|y_k - \hat{y}_k\|_2^2]$, where $\hat{y}_k = w^T\phi^{TLRN}$. Denote RNN features ϕ^{TLRN} as ϕ for simplification. The parameters that minimize the cost function w are determined as follows:

$$\begin{aligned}\frac{\partial J}{\partial w} &= \frac{\partial E[\|y_k - \mathbf{w}^T\phi\|_2^2]}{\partial w} = 0\\ -2\xi\phi &= -2(y_k - \mathbf{w}^T\phi)\phi = 0\\ \mathbf{w}^* &= (\phi^T\phi)^{-1}\phi y_k\end{aligned} \tag{17}$$

The Eq. 17 leads to the most fundamental concept in control systems, where the optimal parameters of the predictor are the product of the inverse autocorrelation of the feature and the cross-correlation between the feature and target y_k.

Using as iterative method the gradient descent, we will have:

$$\begin{aligned}\mathbf{w}_{k+1} &\leftarrow \mathbf{w}_k - \mu\nabla J(w)\\ \mathbf{w}_{k+1} &\leftarrow \mathbf{w}_k + \mu 2\xi\phi\\ \mathbf{w}_{k+1} &\leftarrow \mathbf{w}_k + \eta\xi\phi\end{aligned} \tag{18}$$

where the learning rate is denoted by η and the convex cost is represented by J, based on the expected value of the squared error ξ^2. The iterative equation Eq. (18) was derived using the Least Mean Square method LMS, which in practice shows a noisy response in the gradient estimation.

In [1], a method to improve the gradient estimation called NADAM was demonstrated. However, this study explores a solution aimed at minimizing weight disturbances while maintaining convergence. Therefore, two terms were defined: the first term for reducing the weight disturbances (see Eq. 19), which means that a small error is maintained between the parameter calculations in iteration $k+1$ compared to k. The second term is for convergence (see Eq. 20), where the product of the parameters in $k+1$ multiplied by the feature ϕ equals the random value y_k produced by the sensor.

$$\|\Delta\mathbf{w_{k+1}}\|_2^2 = \|\mathbf{w_{k+1}} - \mathbf{w_k}\|_2^2 \tag{19}$$

$$\begin{aligned}\mathbf{w_{k+1}}\phi_k &= y_k\\ y_k - \mathbf{w_{k+1}}\phi_k &= 0\end{aligned} \tag{20}$$

Improving the parameter optimization leads to a cost definition in which the Lagrange multiplier method is utilized.

$$\begin{aligned}J &= \|\Delta\mathbf{w_{k+1}}\|_2^2 + \lambda(y_k - \mathbf{w_{k+1}}\phi_k)\\ \frac{\partial J}{\partial \mathbf{w_{k+1}}} &= 2(\mathbf{w_{k+1}} - \mathbf{w_k}) - \lambda\phi_k = 0\\ \mathbf{w_{k+1}} &= \mathbf{w_k} + \frac{\lambda\phi_k}{2}\end{aligned} \tag{21}$$

Replacing the Eq. 21 in Eq. 20.

$$\begin{aligned}y_k = \mathbf{w_{k+1}}\phi_k &= \mathbf{w_k}\phi_k + \frac{\lambda\|\phi_k\|_2^2}{2}\\ \lambda = \frac{2(y_k - \mathbf{w_k}\phi_\mathbf{k})}{\|\phi_k\|_2^2} &= \frac{2(\xi)}{\|\phi_k\|_2^2}\end{aligned} \tag{22}$$

The iterative method that minimizes the weight disturbance while maintaining convergence, called the normalized least mean square (NLMS), is (see Eq. 23):

$$\begin{aligned}\mathbf{w_{k+1}} &= \mathbf{w_k} + \frac{1}{\|\phi_k\|_2^2}\phi_k\xi\\ \mathbf{w_{k+1}} &= \mathbf{w_k} + \frac{\eta\phi_k\xi}{\|\phi_k\|_2^2 + \sigma}\end{aligned} \tag{23}$$

where the constant 1 is replaced by the learning rate η, and σ is a gain to avoid stability problems (regularization). The iterative method is normalized by the feature autocorrelation $\phi\phi^T$, which contains a maximum eigenvalue λ_{max}.

3) NLMS STABILITY ANALYSIS

Assuming a radially unbounded positive definite Lyapunov candidate (see Eq. 26), it was possible to demonstrate the stability of the algorithm using Lyapunov. Two error parameter functions were introduced (Eq. 24-25), where w^* represents the optimal filter parameters.

$$\Delta\mathbf{w_k} = w^* - \mathbf{w_k} \tag{24}$$

$$\Delta\mathbf{w_{k+1}} = w^* - \mathbf{w_{k+1}} = \Delta\mathbf{w_k} - \frac{\eta\phi_k\xi}{\|\phi_k\|_2^2 + \sigma} \tag{25}$$

$$V_k = \frac{1}{2}\|\Delta\mathbf{w_k}\|^2 \tag{26}$$

TABLE 2. In general, increasing the RNN order M reduces the MSE between the target y_k and the predicted value $\hat{y_k}$.

M	MSE	M	MSE	M	MSE
2	0.0263	17	0.0023	32	0.0006
3	0.0218	18	0.0021	33	0.0006
4	0.0187	19	0.0018	34	0.0006
5	0.0163	20	0.0016	35	0.0006
6	0.0140	21	0.0014	36	0.0005
7	0.0118	22	0.0013	37	0.0005
8	0.0099	23	0.0012	38	0.0005
9	0.0083	24	0.0011	39	0.0005
10	0.0070	25	0.0010	40	0.0005
11	0.0059	26	0.0009	41	0.0005
12	0.0050	27	0.0008	42	0.0005
13	0.0042	28	0.0008	43	0.0005
14	0.0036	29	0.0007	44	0.0005
15	0.0031	30	0.0007	45	0.0004
16	0.0027	31	0.0007	46	0.0004

The time derivative of the Lyapunov candidate (Eq. 27) is calculated as the difference between two iterations as follows:

$$\dot{V_k} = \Delta V_k = V_{k+1} - V_k$$
$$\dot{V_k} = \frac{1}{2}\|\Delta \mathbf{w_{k+1}}\|^2 - \frac{1}{2}\|\Delta \mathbf{w_k}\|^2 \tag{27}$$

Substituting Eq. 25 in Eq. 27:

$$\dot{V_k} = \frac{1}{2}\|\Delta \mathbf{w_k} - \frac{\eta \phi_k \xi}{\|\phi_k\|_2^2 + \sigma}\|^2 - \frac{1}{2}\|\Delta \mathbf{w_k}\|^2$$
$$\dot{V_k} = \frac{\eta}{2(\|\phi_k\|_2^2 + \sigma)}\left[-2\Delta \mathbf{w_k}\phi_k\xi + \frac{\eta\|\phi_k\|^2\|\xi\|_2^2}{\|\phi_k\|_2^2 + \sigma}\right]$$
$$\dot{V_k} = \frac{\bar{\eta}}{2}\left[-2\Delta \mathbf{w_k}\phi_k\xi + \bar{\eta}\|\phi_k\|^2\|\xi\|_2^2\right] \tag{28}$$

Because an asymptotic solution implies a negative-definite Lyapunov candidate time derivative (Eq. 28), where $\dot{V_k} < 0$, then the normalized learning rate $\bar{\eta}$ for an stable NLMS will be within the next interval (see Eq. 29):

$$0 < \bar{\eta} < \frac{2E[\Delta w_k \phi_k \xi]}{E[\|\phi\|_2^2\|\xi\|_2^2]} \tag{29}$$

where $\bar{\eta} = \frac{\eta}{\|\phi_k\|_2^2+\sigma}$, ϕ_k represents the feature, ξ is the prediction error $y_k - \hat{y}_k$ and Δw_k is defined in Eq. 24. The algorithm is less sensitive to the selection of the learning rate η because of the normalization term, where $\bar{\eta}$ normalization is related to the maximum eigenvalue λ_{max} of the feature autocorrelation $\|\phi_k\|_2^2$.

The prediction in this case was calculated as $\sum_{i=1}^{M} \mathbf{w_i}\phi_\mathbf{i} = \mathbf{w}^T\phi$. Table 2 shows the prediction accuracies for the different TLRN orders.

C. COMPUTER SIMULATIONS

In this section, we compare the performance of the simulation models. Additionally, we introduce and demonstrate the combination of online learning with the transferred knowledge into our learning system scheme. Finally, a comparison between established approaches and the solution proposed in this paper is depicted in terms of learning curves.

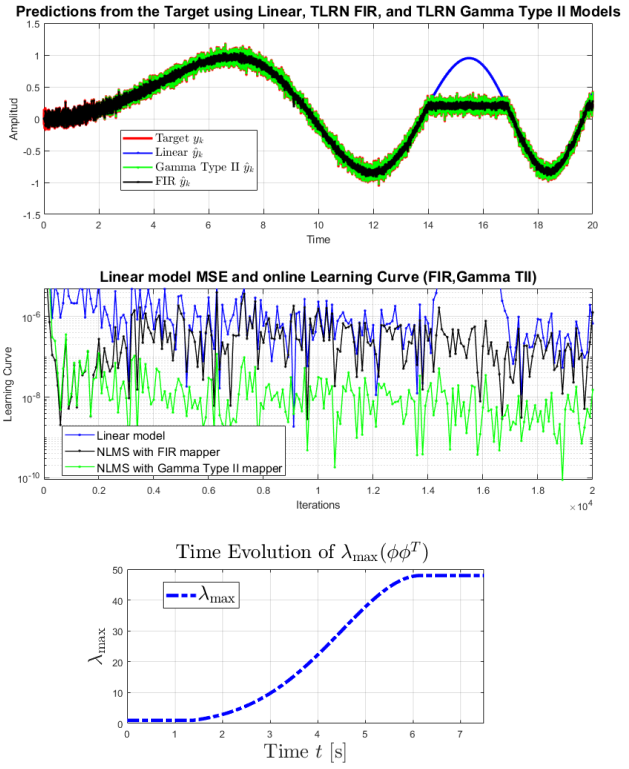

FIGURE 8. Upper Figure: The output of the plant y_k is affected by saturation and additive noise; saturation dynamically changes after $t = 14$ s. The TLRN estimation $\hat{y}k$ learns the plant online. The output of the linear system $\hat{y}k$ lacks adaptability. The middle figure shows the mean square error and the learning curve. The bottom figure shows the response of the maximum eigenvalue λ_{max} for Gamma Type II.

1) ANALYSIS OF THE RESPONSE TO PREDICTIONS

The models of the system determined using the methods described in Sections III-A and III-B were compared in a dynamic environment characterized by the presence of noise and saturation within a specific time interval, and the performance was depicted in Figure 8.

In this study, the TLRN demonstrated its adaptability functionality, in contrast to the linear model transfer function number 3 TF_3 (see Table 1), which maintains its expected behavior in response to chirp-up signal input. Accurately determining the system trajectories in the presence of dynamic variations requires adaptive models, particularly when linear dynamic constraints are not guaranteed.

2) ONLINE-LEARNING AND PREDICTIONS

Another important insight derived from this work is the architecture that combines online learning executed at each iteration with a system to which the learning is transferred. We will refer to the feature determined by the RNN when the input is u_{k-1} as $\phi_{u_{k-1}}^{RNN}$, and when the input is the actual iteration u_k as $\phi_{u_k}^{RNN}$. Then, if the information was learned, $\hat{y}_{k+1}$ is predictable; otherwise, it will be a learning iteration (Figure 9 for clarification). When the transfer has been completed, the map of the system will be determined, and it

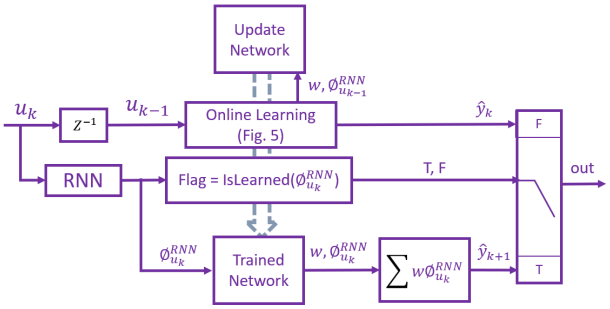

FIGURE 9. The architecture that facilitates online learning, depicted in Figure 5, has been integrated into a system where relevant information from the learning process is transferred online.

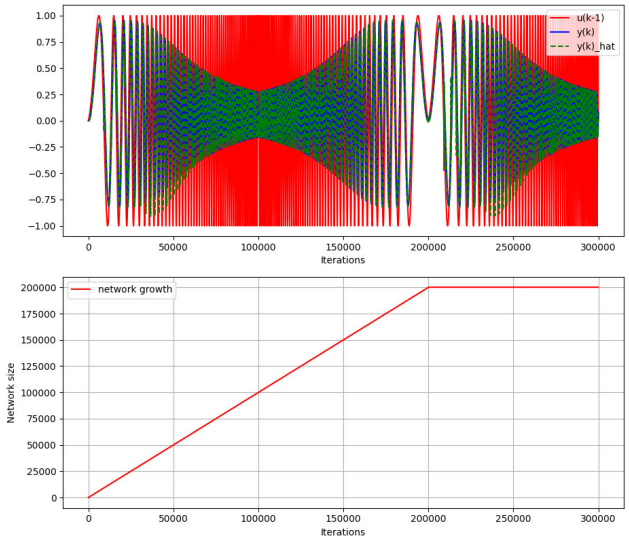

FIGURE 10. The upper figure shows the input u_{k-1}, sensor output y_k, and the predicted value $y_{(k)\ hat} = \hat{y}_k$ during the online learning. The lower figure shows the growth of the information transferred to our learning system and how the steady state was reached after 200,000 iterations.

will not be necessary to learn any new samples, as the growth of the network will be in a steady state Figure 10. The predicted values using the transferred learning system are depicted in Figure 11.

3) COMPARATIVE EVALUATION OF NLMS-BASED RNN AGAINST ESTABLISHED APPROACHES

In this paper, an RNN based on the NLMS search algorithm is developed when the mapper is a Type II Gamma filter. However, although this configuration offers promising results for linear electromechanical systems affected by saturations and noise, there are other methods that have proven highly effective for nonlinear systems and are also efficient in computational terms. Consequently, the performance of the proposed method has been evaluated against two widely recognized approaches in the literature. Firstly, it is compared with the affine projection algorithm (APA). Additionally, it is contrasted with the kernel least mean square method (KLMS), known for its ability to effectively

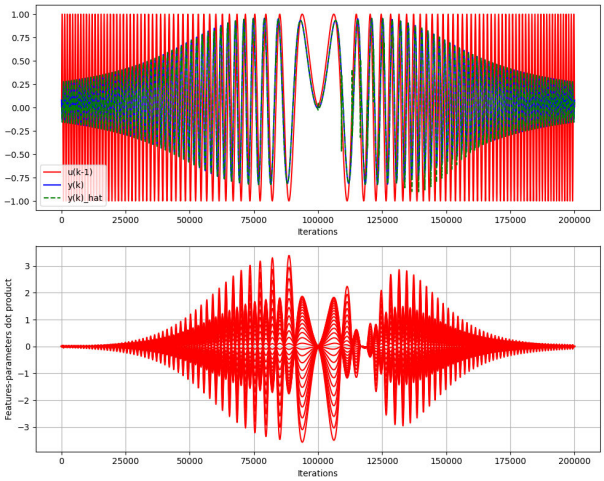

FIGURE 11. Upper figure: The transferred learning system was tested with the flipped input used during the training process. Lower figure: The dot product between parameters and features extracted by the RNN.

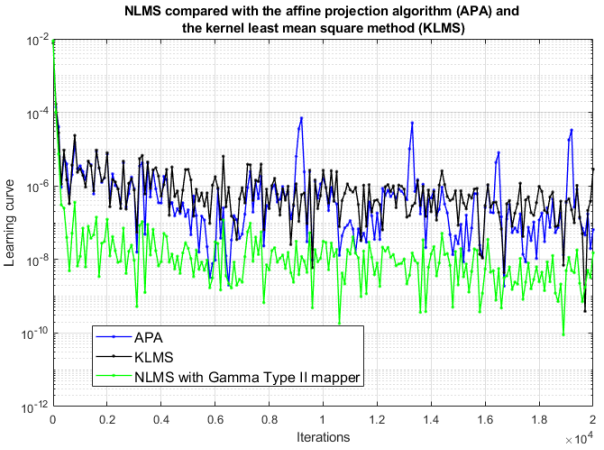

FIGURE 12. Learning curve comparison: NLMS RNN with Gamma type II mapper versus established approaches such as APA and KLMS.

address nonlinear problems in an adaptive learning context (see Figure 12).

IV. SUMMARY AND CONCLUSION

In conclusion, this study presents two strategies for identifying the linear dynamic response of a black-box electromechanical system.

The first strategy involves conducting long-duration sinusoidal experiments, which can be replaced with short-duration experiments by applying a dimensionality expansion technique, evaluated in this study using a radial basis artificial neural network (RBF ANN). Finally, the frequency properties of the steady states of the system are extracted to introduce stable reduced-order linear models with equivalent frequency response. This approach improves data quality, optimizes accuracy, and reduces experimental time.

The second strategy employs online learning techniques to train a TLRN recurrent network integrated with Type II Gamma filters. This autonomous, robust, and adaptive approach accurately models the response of the system in both linear and nonlinear scenarios, even in situations with saturation. Online learning enables engineers to use analytical methods to develop autonomous systems capable of extracting the optimized mapper that defines the real response of a system. Subsequently, through dimensionality reduction techniques, it becomes possible to develop computationally efficient self-determined models online with the real information extracted from the sensors.

These significant findings not only contribute to the field of system identification but also have practical implications for the development of autonomous systems.

REFERENCES

[1] T. Dozat, "Incorporating Nesterov momentum into Adam," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2016, pp. 1–4.

[2] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, San Diego, CA, USA, 2015, pp. 1–11.

[3] L. Bottou, F. E. Curtis, and J. Nocedal, "Optimization methods for large-scale machine learning," in *Proc. 34th Int. Conf. Mach. Learn. (ICML)*, 2017, pp. 3–80.

[4] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proc. COMPSTAT*, 2010, pp. 177–186.

[5] O. S. Patil, A. Isaly, B. Xian, and W. E. Dixon, "Exponential stability with RISE controllers," *IEEE Control Syst. Lett.*, vol. 6, pp. 1592–1597, 2022.

[6] P. M. Patre, W. MacKunis, K. Kaiser, and W. E. Dixon, "Asymptotic tracking for uncertain dynamic systems via a multilayer neural network feedforward and RISE feedback control structure," *IEEE Trans. Autom. Control*, vol. 53, no. 9, pp. 2180–2185, Oct. 2008.

[7] P. M. Patre, W. MacKunis, C. Makkar, and W. E. Dixon, "Asymptotic tracking for systems with structured and unstructured uncertainties," *IEEE Trans. Control Syst. Technol.*, vol. 16, no. 2, pp. 373–379, Mar. 2008.

[8] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, Oct. 1986.

[9] T.-H. Oh, J. Han, Y.-S. Kim, D.-Y. Yang, T.-H. Kim, S.-H. Lee, and D. D. Cho, "Sine-sweep input generation with minimum energy restriction for obtaining a precise frequency response function in industrial servo systems," in *Proc. 21st Int. Conf. Control, Autom. Syst. (ICCAS)*, Oct. 2021, pp. 1830–1834.

[10] W. Liu, J. C. Principe, and S. Haykin, *Kernel Adaptive Filtering*. Hoboken, NJ, USA: Wiley, 2010, pp. 20–32.

[11] S. Goyal and P. Barooah, "A method for model-reduction of non-linear thermal dynamics of multi-zone buildings," *Energy Buildings*, vol. 47, pp. 332–340, Apr. 2012.

[12] P. J. Werbos, "Consistency of HDP applied to a simple reinforcement learning problem," *Neural Netw.*, vol. 3, no. 2, pp. 179–189, Jan. 1990.

[13] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006, pp. 299–301.

[14] S. Zhao, B. Chen, and J. C. Príncipe, "Kernel adaptive filtering with maximum correntropy criterion," in *Proc. Int. Joint Conf. Neural Netw.*, Jul. 2011, pp. 2012–2017.

[15] D. H. Nguyen and B. Widrow, "Neural networks for self-learning control systems," *IEEE Control Syst. Mag.*, vol. 10, no. 3, pp. 18–23, Apr. 1990, doi: 10.1109/37.55119.

[16] A. Singh and J. C. Principe, "Using correntropy as a cost function in linear adaptive filters," in *Proc. Int. Joint Conf. Neural Netw.*, Jun. 2009, pp. 2950–2955.

[17] A. G. Barto, R. S. Sutton, and C. W. Anderson, "Neuronlike adaptive elements that can solve difficult learning control problems," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-13, no. 5, pp. 834–846, Sep. 1983.

[18] B. A. Pearlmutter, "Learning state space trajectories in recurrent neural networks," *Neural Comput.*, vol. 1, no. 2, pp. 263–269, Jun. 1989.

[19] R. J. Williams and D. Zipser, "A learning algorithm for continually running fully recurrent neural networks," *Neural Comput.*, vol. 1, no. 2, pp. 270–280, Jun. 1989.

[20] R. J. Williams and D. Zisper, "Gradient-based learning algorithms for recurrent networks and their computational complexity," in *Back-Propagation: Theory, Architectures and Applications Hillsdale*, Y. Chauvia and D. E Rumelhart, Eds. Hillsdale, NJ, USA: Erbaum, 2013, ch. 1.

[21] N. Wiener, *Extrapolation, Interpolation and Smoothing of Time Series, With Engineering Applications*. Hoboken, NJ, USA: Wiley, 1949.

[22] J. C. Principe, B. de Vries, and P. G. de Oliveira, "The gamma-filter—A new class of adaptive IIR filters with restricted feedback," *IEEE Trans. Signal Process.*, vol. 41, no. 2, pp. 649–656, Feb. 1993.

[23] M. A. Masnadi-Shirazi and N. Ahmed, "Optimum Laguerre networks for a class of discrete-time systems," *IEEE Trans. Signal Process.*, vol. 39, no. 9, pp. 2104–2108, Sep. 1991.

[24] P. Crama and J. Schoukens, "Initial estimates of Wiener and Hammerstein systems using multisine excitation," *IEEE Trans. Instrum. Meas.*, vol. 50, no. 6, pp. 1791–1795, Dec. 2001.

[25] A. Novak, L. Simon, F. Kadlec, and P. Lotton, "Nonlinear system identification using exponential swept-sine signal," *IEEE Trans. Instrum. Meas.*, vol. 59, no. 8, pp. 2220–2229, Aug. 2010.

[26] J. Schoukens, R. Pintelon, and Y. Rolain, "Identification of nonlinear and linear systems, similarities, differences, challenges," in *Proc. 14th IFAC Symp. Syst. Identificat.*, 2006, pp. 122–124.

[27] W. T. Miller, R. S. Sutton, and P. J. Werbos, *Neural Networks for Control*. Cambridge, MA, USA: MIT Press, 1995.

[28] E. C. Levy, "Complex-curve fitting," *IRE Trans. Autom. Control*, vols. AC–4, no. 1, pp. 37–43, May 1959.

[29] J. E. Dennis Jr. and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Upper Saddle River, NJ, USA: Prentice-Hall, 1983.

[30] K. J. Åström, *Introduction to Stochastic Control Theory*, 1st ed. New York, NY, USA: Academic Press, 1970.

C. A. L. SEGURA (Member, IEEE) received the M.Sc. degree in industrial engineering and in industrial electronics and automation engineering from Universidad de Las Palmas de Gran Canaria (ULPGC), Spain, in 2012, and the master's degree in control systems from the Department of Mechanical and Aerospace Engineering, University of Florida (UF), in 2022. His research interests include the intersection of machine learning, control systems, and neural networks.