Identification of structured nonlinear state–space models for hysteretic systems using neural network hysteresis operators

Document Version:
Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

Identification of structured nonlinear state–space models for hysteretic systems using neural network hysteresis operators

Konstantinos Krikelis [a], Jin-Song Pei [b], Koos van Berkel [c], Maarten Schoukens [a,*]

[a] *Control Systems Group, Department of Electrical Eng., Eindhoven University of Technology, Postbus 513, 5600 MB Eindhoven, The Netherlands*
[b] *School of Civil Engineering & Environmental Science, 202 W. Boyd, Room 327E, Norman, OK 73019, USA*
[c] *Research Department, ASML Netherlands B.V., 5504 DR Veldhoven, The Netherlands*

A R T I C L E   I N F O

A B S T R A C T

Hysteretic system behavior is ubiquitous in science and engineering fields including measurement systems and applications. In this paper, we put forth a nonlinear state–space system identification method that combines the state–space equations to capture the system dynamics with a compact and exact artificial neural network (ANN) representation of the classical Prandtl–Ishlinskii (PI) hysteresis. These ANN representations called PI hysteresis operator neurons employ recurrent ANNs with classical activation functions, and thus can be trained with classical neural network learning algorithms. The structured nonlinear state–space model class proposed in this paper, for the first time, offers a flexible interconnection of PI hysteresis operators with a linear state–space model through a linear fractional representation. This results in a comprehensive and flexible model structure. The performance is validated both on numerical simulation and on measurement data.

1. Introduction

Hysteretic system behavior appears in diversified areas of science such as electromagnetism, solid mechanics, aerodynamics, and many others [1–3]. In measurement applications, there are studies towards quantifying the hysteresis effects in terms of both taking measurements [4,5] and studying caused errors [6–8]. While understanding the cause and extent of hysteresis in measurement systems (e.g., [9]) is important, modeling hysteresis (e.g., [8]), exploiting hysteresis (e.g., [10]), and compensating for hysteresis (e.g., [11]) are also important. Various models with descriptive switching rules to simulate and characterize different hysteretic phenomena have been proposed in the literature. However, there is no universal model that can describe all hysteresis effects. Refer to, for example, [12,13] for an overview of hysteresis models.

As in the literature (e.g., [14,12]), hysteresis models are usually classified into two main types: physics-based models and phenomenological models. Physics-based models are derived based on the first principles of physics effects, while phenomenological models are derived based on the appearance and the shape of the observed hysteresis loops. In general, the underlying physics of hysteretic system behavior is usually very complicated so that it is difficult to derive physics-based models. Moreover, a physics-based model of one hysteretic system can usually not be applied to another system. Phenomenological models

on the other hand can describe behaviors similar to those of physical models without the need for detailed physical insight into the modeling problem [13]. Hence these types of models are considered much more flexible in terms of hysteresis modeling. Such models include the Preisach model, Prandtl–Ishlinskii model, Bouc–Wen model, Duhem model, Maxwell-Slip model, Krasnoselskii–Pokrovskii model, and many others. This study contributes to phenomenological hysteresis models.

The identification of a hysteretic system is challenging due to its complex dynamic nonlinear behavior. Furthermore, the hysteretic behavior is often associated with other dynamic behavior, e.g. by interconnected mass–spring–damper systems. Most of the hysteretic system identification literature is limited to isolated hysteresis behavior [15–17], or to the slightly more general Hammerstein nonlinear model structure [18–21]. Black-box nonlinear state–space identification approaches have been demonstrated to be able to capture hysteretic system behavior together with system dynamics [22], and the nonlinear state–space model structure is also known to be able to model more complex system structures [23].

One of the main challenges when using a nonlinear state–space model structure for identification is to keep the complexity of the nonlinearity under control for an increasing state dimension. One approach to achieve this is to reduce the nonlinearity dimension and decouple the nonlinearity in the model using tensor decomposition

---

approaches [24,25]. Another approach is to impose this reduced non-linearity dimensionality from the start, for instance, using a nonlinear LFR (linear fractional representation) model structure [26–28]. On the other hand, the nonlinear LFR model structure can also be interpreted as a generalization of the more limited Hammerstein and Wiener block-oriented model structures [29], which are often used in conjunction with PI hysteresis models to capture the system dynamics.

In this paper, we present a compact and exact ANN representation of the classical Prandtl–Ishlinskii (PI) hysteresis operator, named PI hysteresis operator neuron throughout the paper, that is subsequently used for the identification of structured nonlinear hysteretic state–space models. The PI hysteresis operator neurons are represented by recurrent ANNs using classical activation functions and can be trained with classical neural network learning algorithms. The three PI operators that are considered in this study are the stop, play, and generalized play operators. One of the benefits of the PI hysteresis operator neuron is the flexibility it offers to be combined with each other to form a layer for hysteresis components and with various other comprehensive model structures. This paper introduces a structured nonlinear state–space identification approach, using the so-called PI-LFR model class. The PI-LFR model class is obtained by replacing the static nonlinearity of the nonlinear LFR model structure with the aforementioned PI operator neurons to obtain a dedicated hysteretic nonlinearity, as depicted in Fig. 1. The proposed model structure enhances the versatility of the LFR model class to model a wide variety of hysteretic systems and its performance is validated both on numerical simulation and on measurement data. This results in the following main contributions:

1. The derived exact representations of the stop, play, and generalized play PI operators with each using a simple and small recurrent ANN employing widely used activation functions (ReLU, SSLU)
2. The Integration of these recurrent ANNs as substructures with the LFR framework, being able to treat the hysteresis just like other dynamic components. This allows for the simultaneous identification of hysteresis and linear system dynamics using a structured nonlinear state–space model.
3. The use of "rich" input signals such as multisine experiments allows to simultaneously capture the broadband system dynamics and hysteretic effects, similar to [22]

To the authors' knowledge, there are no other identification approaches available that can jointly identify system dynamics and the PI hysteresis containing similar complex interactions as discussed in this paper. The current state of the art for structured dynamic hysteresis models is largely limited to the well-known Hammerstein and Wiener case [18–21,30]. This highlights the significance of the intended contributions. Comparative exercises in this study using Masing hysteresis vividly demonstrate the approximation efficiency of the intended contribution No. 1.

The remainder of the paper is organized as follows: Section 2 reviews the Prandtl–Ishlinskii model and its classical hysteresis operators. In Section 3, the hysteresis operator neurons are formulated for three classical Prandtl–Ishlinskii hysteresis operators. The PI-LFR model that is considered for the identification of hysteretic systems is introduced in Section 4, while Section 5 presents the proposed identification method. In Sections 6 and 7, the proposed approaches are respectively numerically and experimentally validated using real-world measurement data.

2. The Prandtl–Ishlinskii (PI) hysteresis model

The Prandtl model is a well-known phenomenological operator-based model for characterizing hysteresis behavior. In the Prandtl
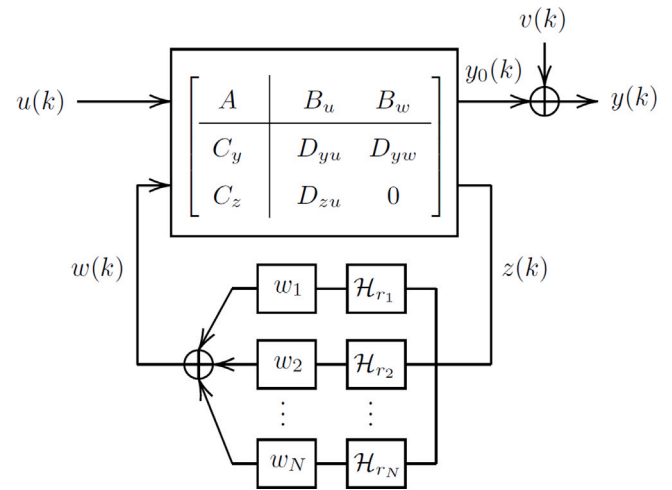

Fig. 1. The proposed PI-LFR structure obtained by interconnecting a MIMO LTI state–space representation with an ANN layer, which is given as a weighted sum over a finite number N of PI hysteresis neurons $\mathcal{H}_r$.
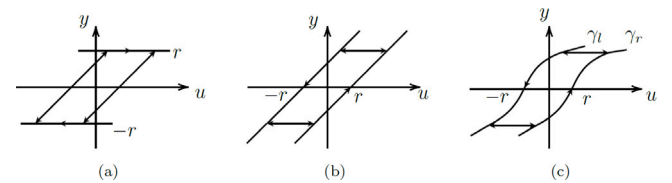

Fig. 2. Hysteresis operators: (a) stop operator (b) play operator (c) generalized play operator for arbitrary envelope functions.

model formulation as presented in [14], the output signal $y(t)$ is determined through a linear superposition of an infinite set of hysteresis operators applied to the input signal $u(t)$ and is expressed as:

$$y(t) = \int_0^\infty w(r)\mathcal{H}_r[u(t)]dr, \tag{1}$$

where t denotes the time, $w(r) > 0$ is the density function that determines the hysteresis loop shape, $r \geq 0$ is a single threshold variable that parameterizes the hysteresis operator $\mathcal{H}_r[\cdot]$. The hysteresis operator in the Prandtl model can either be the stop, the play, or the generalized play operator as illustrated in Fig. 2. The discrete form of the Prandtl model is known as the Prandtl–Ishilinskii (PI) model which is expressed as a weighted sum over a finite number of hysteresis operators:

$$y(k) = \sum_{i=1}^N w_i \mathcal{H}_{r_i}[u(k)], \tag{2}$$

where k is the sample index, N is the total number of hysteresis operators, w_i is a scalar gain of the ith operator satisfying $w_i \geq 0$ $\forall i \in [1, N]$. $\mathcal{H}_{r_i}[\cdot]$ is the ith discrete hysteresis operator given by a stop, play, or generalized play operator.

2.1. The stop operator

$$
\begin{aligned}
y_i(k) &= S_{r_i}[u(k)], \\
&= \min\{r_i, \max\{-r_i, u(k) - u(k-1) + y_i(k-1)\}\},
\end{aligned} \tag{3}
$$

with the initial condition given by:

$$y_i(0) = \max\{r_i, \max\{-r_i, u(0) - u(-1) + y_{i0}\}\}, \tag{4}$$

where usually, but not necessarily, zero initial conditions are used for the values $u(-1)$ and y_{i0}. The input–output map of a stop operator is depicted in Fig. 2(a).

2.2. The play operator

The play operator is parameterized by a threshold variable $r_i \geq 0$. Its discrete-time (DT) output $y_i(k)$, given an input signal $u(k)$, is provided by [14]:

$$
\begin{aligned}
y_i(k) &= \mathcal{P}_{r_i}[u(k)], \\
&= \max\{u(k) - r_i, \min\{u(k) + r_i, y_i(k-1)\}\},
\end{aligned}
\tag{5}
$$

with the initial condition given by:

$$
y_i(0) = \max\{u(0) - r_i, \min\{u(0) + r_i, y_{i0}\}\},
\tag{6}
$$

where usually, but not necessarily, zero initial conditions are used for y_{i0}. The input–output map of a play operator parameterized by its threshold variable r is depicted in Fig. 2(b).

2.3. The generalized play operator

The classical play operator is extended towards the generalized play operator by using two auxiliary functions γ_r and γ_l. An increase in input $u(k)$ causes the output $y_i(k)$ to increase along the curve γ_r, while a decrease in input $u(k)$ causes the output $y_i(k)$ to decrease along the curve γ_l. This results in the following equations [14,31]:

$$
\begin{aligned}
y_i(k) &= \mathcal{GP}_{r_i}[u(k) \mid \gamma_l(\cdot), \gamma_r(\cdot)] \\
&= \max\{\gamma_l(u(k)) - r_i, \min\{\gamma_r(u(k)) + r_i, y(k-1)\}\}
\end{aligned}
\tag{7}
$$

where the envelope functions $\gamma_l(\cdot) \geq \gamma_r(\cdot)$ are both continuous and non-decreasing functions [14,31]. The generalized play operator results in asymmetric hysteresis loops when different envelope functions are used. The input–output map of a generalized play operator for arbitrary envelope functions is depicted in Fig. 2(c).

2.4. Properties

The play and stop operators are linked by the complementary property:

$$
\mathcal{P}_{r_i}[u(k)] + \mathcal{S}_{r_i}[u(k)] = u(k)
\tag{8}
$$

The proof of this property is provided in [14]. Furthermore, the different operators are Lipschitz continuous and stable operators [32], complying with both Madelung's and Masing's rules [14]. Hence, the PI model inherits the continuity and stability properties by linear superposition.

3. Neural hysteretic operators

Hysteresis models based on ANNs have been developed to model hysteresis behavior in several applications. In [33], an ANN-based PI model that utilized stop operators, different from the ANN-based PI model introduced in this paper, was developed to model the hysteresis behavior in nonlinear inelastic frames and trusses. Due to the difficult computation of the operators' gradients, the ANN-based PI model was trained with a genetic algorithm (GA). This technical difficulty is also present in other works [34–36] where the PI hysteresis operators are either not trained at all during optimization or the entire ANN model is trained with a GA. Some researchers have also proposed their own hysteresis operators neurons [37,36] which were inspired by the classical hysteresis modeling tools.

In this section, we formulate the classical play, stop, and generalized play hysteresis operators as recurrent ANNs. Note that these ANN formulations are simply obtained by rewriting the original hysteresis operators, no approximation is taking place in the process (see Appendix) as an example. They are constructed with classical activation functions, thus can be realized as hysteresis operator neurons in any ANN model structure, and can be trained with a wide set of machine/deep learning tools in a computationally efficient way.

By interconnecting these hysteresis operator neurons in parallel into a larger layer, ANN hysteresis models are realized. These hysteretic ANN layers can be easily integrated into other (deep) neural network architectures, as illustrated in Section 4, allowing for a wide range of possible model structures that can capture hysteretic system behavior at virtually no extra expense during the model structure design and parameter estimation phase.

3.1. Stop operator

The stop operator given in Eq. (3) is represented in this study by the following recurrent ANN:

$$
y(k) = r f\left(\frac{u(k) - u(k-1) + y(k-1)}{r} \right)
\tag{9}
$$

where $u(k)$, and $y(k)$ are the input and output signal of a stop operator parameterized by the threshold variable r respectively. The symmetric saturated linear unit (SSLU) activation function $f(\cdot)$ in the hidden layer realizes the stop operator:

$$
f(x) = \begin{cases} 1, & \text{if } x \geq 1 \\ -1, & \text{if } x \leq -1 \\ x, & \text{if } -1 < x < 1 \end{cases}
\tag{10}
$$

Note that the neural stop operator is not defined for $r = 0$, while the classical stop operator is defined for all positive threshold values including zero. However, the response of the classical stop operator for $r = 0$ is 0. Hence, this does not pose any issue in the representation above. A schematic of the stop hysteresis operator neuron is depicted in Fig. 4, and its resulting behavior is depicted in Fig. 3.

3.2. Play operator

The play operator given in Eq. (5) is represented using the following recurrent ANN:

$$
\begin{aligned}
y(k) &= u(k) + \begin{bmatrix} -1 & 1 \end{bmatrix} s(k) + r \\
s(k) &= g\left(\begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} u(k) \\ y(k-1) \end{bmatrix} + \begin{bmatrix} r \\ -r \end{bmatrix} \right)
\end{aligned}
\tag{11}
$$

where $u(k)$, $y(k)$ are the input and output signal of a play operator parameterized by the threshold variable r respectively. The commonly-used rectified linear unit (ReLU) activation function $g(\cdot)$ in the hidden layer realizes the play operator:

$$
g(x) = \max(x, 0)
\tag{12}
$$

Note that the activation function $g(x)$ is applied element-wise, and that threshold value r which is the only variable that parameterizes a classical play operator is directly associated with the bias terms of the proposed neural play operator. A schematic of the play hysteresis operator neuron is depicted in Fig. 5, and its resulting behavior is depicted in Fig. 3.

3.3. Generalized play operator

The generalized play operator, starting from Eq. (7) [14,31], is represented in this paper by the following recurrent ANN:

$$
\begin{aligned}
y(k) &= \gamma_l(u(k)) - r + g(-\gamma_l(u(k)) + r + s(k)) \\
s(k) &= \gamma_r(u(k)) + r - g(\gamma_r(u(k)) + r - y(k-1))
\end{aligned}
\tag{13}
$$

where $u(k)$ is the input, $y(k)$ is the output, and $s(k)$ is an internal state of the generalized play operator parameterized by the threshold variable r, using the ReLU activation function as before. Additionally, both envelope functions $\gamma_l(\cdot)$ and $\gamma_r(\cdot)$ can be approximated with arbitrary precision using universal approximators such as sigmoidal [38] or ReLU [39] networks, i.e.:

$$
\begin{aligned}
\gamma_l(u(k)) &\cong \mathcal{NN}_l(u(k) | \theta_l) \\
\gamma_r(u(k)) &\cong \mathcal{NN}_r(u(k) | \theta_r)
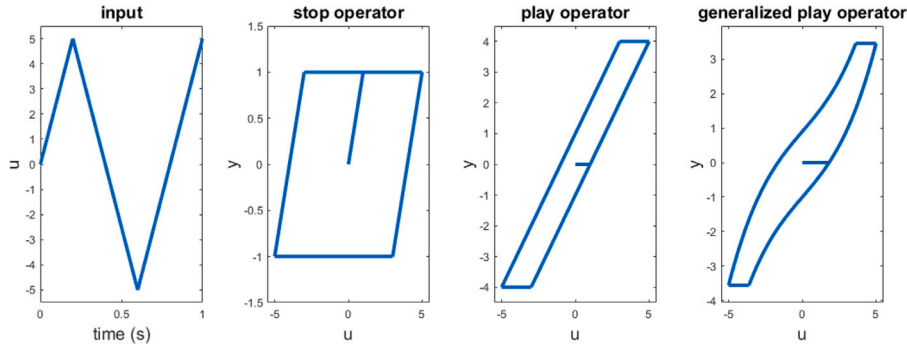\end{aligned}
\tag{14}
$$

Fig. 3. The stop, play, and generalized play operator as obtained by Eqs. (9), (11), and (13) respectively for $r = 1$, starting from zero initial conditions. The generalized play operator uses the functions $\gamma_l(x) = \frac{x}{2} + (\frac{x}{4})^3$ and $\gamma_r(x) = \frac{x}{2} + (\frac{x}{4})^3 - 0.1$.
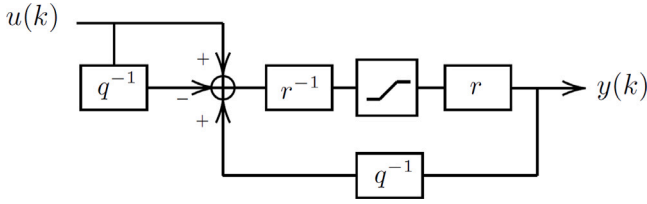

Fig. 4. Schematic of the stop hysteresis operator neuron represented in terms of the input–output signals, the SSLU activation function and the trainable weights parameterized by the threshold value r.
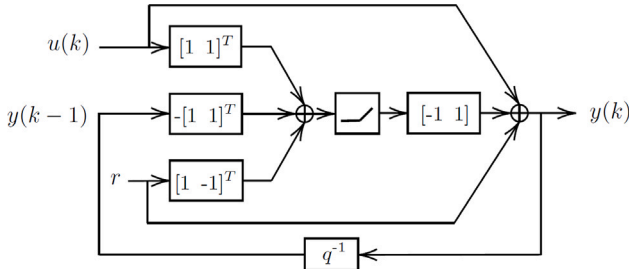

Fig. 5. Schematic of the play hysteresis operator neuron represented in terms of the input–output signals, the ReLU activation function, the non-trainable weights and the threshold value r.

where θ_l and θ_r are the parameter vectors that contain the weights and biases of the parameterized networks $\mathcal{NN}_l$ and $\mathcal{NN}_r$. Its resulting behavior is depicted in Fig. 3.

3.4. Complementary operator

The complementary property presented in Eq. (8) allows to represent both the play and stop operators with SSLU and ReLU layers respectively and vice versa. Both activation functions are almost identical in terms of performance and convergence speed [40]. However, the ReLU representation offers an easier parametrization of the hysteresis operator since it provides a direct mapping between the threshold value r and the bias term of the hysteresis neuron.

4. Hysteretic LFR model structure

The previously introduced ANN hysteresis operators are combined with the LFR model structure in this section. This results in a nonlinear state–space model that can simultaneously capture complex system dynamics and nonlinear hysteresis. The LFR structure can be understood as a structured nonlinear state–space representation [28] or can be interpreted as a generalization of the more limited Hammerstein and Wiener block-oriented model structures [29].

4.1. Model structure

The proposed DT nonlinear model structure consists of a PI hysteresis nonlinearity interconnected with the linear fractional representation, as shown in Fig. 1. The linear dynamics are represented using a state–space representation describing the dynamic relation between the LFR inputs $u(k) \in \mathbb{R}^{n_u \times 1}$, $w(k) \in \mathbb{R}^{1 \times 1}$ and the outputs $y(k) \in \mathbb{R}^{n_y \times 1}$, $z(k) \in \mathbb{R}^{1 \times 1}$. The PI hysteresis nonlinearity is represented using an ANN layer consisting of N PI hysteresis operator neurons. This results in the following model equations:

$$\begin{bmatrix} x(k+1) \\ y_0(k) \\ z(k) \end{bmatrix} = \left[\begin{array}{c|cc} A & B_u & B_w \\ \hline C_y & D_{yu} & D_{yw} \\ C_z & D_{zu} & 0 \end{array} \right] \begin{bmatrix} x(k) \\ u(k) \\ w(k) \end{bmatrix} \tag{15}$$

and

$$w(k) = \sum_{i=1}^{N} w_i \mathcal{H}_{r_i}[z(k)] \tag{16}$$

where N is the total number of hysteresis operators, $w_i \; \forall i \in [1, N]$ are scalar weights and $\mathcal{H}_{r_i}[\cdot]$ is a user-specified hysteresis operator which can be the stop operator presented in Eq. (9), the play operator presented in Eq. (11) or even the generalized play operator presented in Eq. (13), i.e.:

$$\mathcal{H}_{r_i}[z(k)] = \begin{cases} \mathcal{S}_{r_i}[z(k)] \\ \mathcal{P}_{r_i}[z(k)] \\ \mathcal{GP}_{r_i}[z(k) \mid \mathcal{NN}_l(\cdot|\theta_l), \mathcal{NN}_r(\cdot|\theta_r)] \end{cases} \tag{17}$$

The states are represented by $x(k) \in \mathbb{R}^{n_x \times 1}$ and the matrices $A \in \mathbb{R}^{n_x \times n_x}$, $B_u \in \mathbb{R}^{n_x \times n_u}$, $B_w \in \mathbb{R}^{n_x \times 1}$, $C_y \in \mathbb{R}^{n_y \times n_x}$, $C_z \in \mathbb{R}^{1 \times n_x}$, $D_{zu} \in \mathbb{R}^{1 \times 1}$, $D_{yu} \in \mathbb{R}^{n_y \times n_u}$, $D_{yw} \in \mathbb{R}^{n_y \times 1}$ correspond to the parameters to be estimated together with the parameters of the hysteresis model in Eq. (16). The term D_{zw} is not present in the considered state–space formulation as this prevents the presence of algebraic loops in the model expression. Additionally, a zero-mean, additive noise source $v(k)$ is assumed to present at the output $y(k)$ only. The problem of selecting the number of states n_x, the number of hysteresis operators N, and the type of the hysteresis operator is a classical model complexity selection problem. Typical solutions to this problem include the use of prior expert knowledge provided by the user and grid search approaches.

Observe that the PI-LFR model given by Eqs. (15) and (16) and depicted in Fig. 1 can be interpreted both as a highly flexible block-oriented model [29], it is a superset of the popular PI-Hammerstein and PI-Wiener model structures, or as a highly structured nonlinear state–space representation [28].

Furthermore, due to the presence of the direct feedthrough term D_{yu} in the PI-LFR model and due to the complementary property of the classical PI operators, a PI-LFR model with play operators is mathematically equivalent to a PI-LFR model with stop operators and vice versa.

Finally, note that the linear state–space model is a linear recurrent neural network. Here we implement the complete PI-LFR model as a structured recurrent neural network. Hence, the model can be identified using the effective optimization tools available for neural network training.

5. Identification method

5.1. Excitation signal

The considered excitation signal for the identification and test is a random phase multisine signal. A multisine is a periodic signal with random phases and a user-defined amplitude spectrum [41]. Although the proposed identification approach is not limited to this type of input signal, compared to other signals such as random Gaussian inputs, it has a random appearance in the time domain due to the random phases while it retains a deterministic, user-specified, amplitude spectrum in the frequency domain. One period of the signal is defined for $0 \leq k \leq (N_s - 1)/f_s$ where f_s is the sampling frequency and N_s the numbers of samples of the signal:

$$u(k) = \frac{1}{\sqrt{N_s}} \sum_{n=1}^{n_{\max}} A_n \sin\left(2\pi k \frac{f_s}{N_s} n + \phi_n\right), \tag{18}$$

with

$$n_{\max} = \text{round}\left(\frac{f_{\max}}{f_s} N_s\right) \tag{19}$$

where f_{max} is a user specified maximum frequency, A_n is a user specified amplitude spectrum and the phases, $\phi_n \in [0, 2\pi)$ are drawn randomly from a uniform distribution. Note that the proposed method is not limited to multisine excitation signals. Any other type of excitation signal that sufficiently excites the system dynamics, e.g. random noise, sinesweep, or pseudorandom binary sequence (PRBS) signals, can be used.

5.2. Parameter estimation

The mean squared simulation error is considered as the cost function to minimize:

$$\theta_o = \arg\min_{\theta} \frac{1}{M} \sum_{k=1}^{M} (y(k) - \hat{y}(k|\theta))^2 \tag{20}$$

where o stands for optimized, M is the total number of samples over which the cost function is computed, $y(k)$ is the measured output and $\hat{y}(k|\theta)$ is the output simulated by the PI-LFR model given the parameter vector θ. This vector includes the vectorized state–space matrices of the LTI subblock as well as the scalar weights w_i and threshold values r_i of the PI ANN layer. In the case of a PI ANN layer with generalized play operator neurons, the parameter vector θ contains also all the weights and biases of the envelope function approximators.

The considered cost function is nonlinear in the parameters and non-convex. It is minimized using gradient-based or Gauss-Newton-like algorithms. In this paper, the Levenberg–Marquardt algorithm [42,43] is used since it combines both minimization methods and is used in recurrent neural network design [44].

5.3. Parameter initialization

A 'good' initial guess of the parameter values is required to initiate the optimization algorithm as it will only converge to the 'closest' local minimum of the cost function. LTI approximations of the nonlinear system under consideration have proven effective on multiple occasions to initialize nonlinear identification algorithms [29,45]. In this paper, the best linear approximation (BLA) framework is used. The BLA of a nonlinear system is an LTI model $G_{bla}(q^{-1})$ that approximates the system output best in least square sense [41] i.e.:

$$G_{bla}(q^{-1}) = \arg\min_{G(q^{-1})} E_{u,v}\left\{\left|\tilde{y}(k) - G(q^{-1})\tilde{u}(k)\right|^2\right\}$$

$$\tilde{u}(k) = u(k) - E_u\{u(k)\}$$

$$\tilde{y}(k) = y(k) - E_{u,v}\{y(k)\} \tag{21}$$

where $E_{u,v}$ denotes the expected value operator taken with respect to the random variations due to the input $u(k)$ and the output noise $v(k)$, q^{-1} denotes the backward shift operator. As can be observed in Eq. (21), the BLA of a nonlinear system is dependent on the properties of the considered input signal class. The BLA state–space matrices (A_{BLA}, B_{BLA}, C_{BLA}, D_{BLA}) are estimated using a subspace algorithm [46], are then transformed such that each of the states has a unit variance [28] resulting in the transformed state–space matrices ($\tilde{A}_{BLA}$, $\tilde{B}_{BLA}$, $\tilde{C}_{BLA}$, $\tilde{D}_{BLA}$). Also note that starting from the BLA estimate allows the user to perform the state order (n_x) selection problem in an LTI setting, which greatly simplifies the problem [30].

To start the parameter identification for PI-LFR, these transformed matrices are finally embedded into the PI-LFR model:

$$\begin{aligned} A &= \tilde{A}_{BLA} & B_u &= \tilde{B}_{BLA} \\ C_y &= \tilde{C}_{BLA} & D_{yu} &= \tilde{D}_{BLA} \end{aligned} \tag{22}$$

while the rest of the LFR parameters are initialized as:

$$\begin{aligned} B_w &= 0 & D_{yw} &= 0 \\ C_z &= \mathcal{U}(-1, 1) & D_{zu} &= \mathcal{U}(-1, 1) \end{aligned} \tag{23}$$

where $\mathcal{U}(a, b)$ denotes a uniform distribution set with minimum value a and maximum b. The hysteresis model is initialized slightly differently. The PI model satisfies: $w_i, r_i \geq 0 \quad \forall i \in [1, N]$. Hence, the weights w_i and the threshold values r_i are all initialized as:

$$w_i \in \mathcal{U}(0, C), \quad r_i \in \mathcal{U}(0, R) \qquad \forall i \in [1, N] \tag{24}$$

The maximum values $C > 0$ and $R > 0$ are user-specified upper bounds of the sets. In this paper, the value C is often chosen as $C = 1/N$, such that the signal $w(k)$ has a comparable magnitude with that of the signal $z(k)$ while R is set to 0.1 by default. N will be discussed later. With this approach, it is ensured that the hysteresis operators are active at the beginning of the optimization.

In the case of generalized play operator neurons, the weights and biases that correspond to the envelope function approximators $\mathcal{NN}_l$, $\mathcal{NN}_r$ are initialized with random values that are all drawn uniformly in $\mathcal{U}(-1, 1)$ and zeros respectively.

6. Numerical validation

In this section, the proposed model structure and identification method are numerically validated on a hysteretic system following the Masing hysteresis rules. Examples of Masing hysteresis include the polarization process of piezoelectric materials [47], in hysteretic friction [17], or hysteretic effects in systems under earthquake excitations [48] and many others.

6.1. System description

Consider the CT 2-DOF mass–spring–damper (MSD) system as shown in Fig. 6 which is governed by the following equations of motion:

$$m_1 \ddot{x}_1 + d_2(\dot{x}_1 - \dot{x}_2) + k_2(x_1 - x_2) + d_1 \dot{x}_1 + k_1 x_1 + k_{\mathcal{H}}(x_1) = 0$$

$$m_2 \ddot{x}_2 + d_2(\dot{x}_2 - \dot{x}_1) + k_2(x_2 - x_1) = u, \tag{25}$$

where u is the input force acting on the system, x_2, which is considered as the system output, is the measured displacement of mass 2 and $k_{\mathcal{H}}$ denotes a hysteretic spring response that affects the displacement of
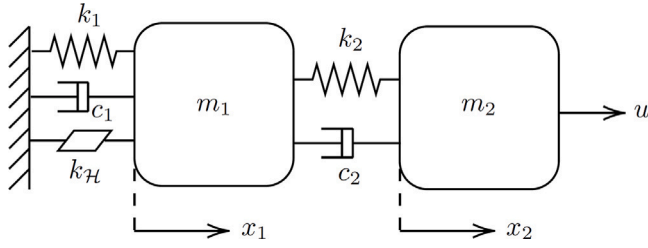
Fig. 6. Schematic of the hysteretic 2-DOF mass–spring–damper system where the hysteretic spring element is represented by $k_\mathcal{H}$. The force u is the system input, x_2 is the measured system output.

Table 1
Parameters of the hysteretic mass–spring–damper system.

m_1 [kg]	m_2 [kg]	k_1 [N/m]	d_1 [Ns/m]	k_2 [N/m]	d_2 [Ns/m]	K [N/m]	r_u [N]	n
2.27	2.86	1100	15	$1.1 \cdot 10^4$	19.9	3000	0.8	1

mass 1 which obeys the extended Masing rules of hysteresis. Following [49] on the extended Masing models (EMM), we use this specific model:

$$\dot{k}_\mathcal{H}(x_1) = \begin{cases} K\left[1 - \left|\frac{k_\mathcal{H}}{r_u}\right|^n\right]\dot{x}_1, & \text{initial loading} \\ K\left[1 - \left|\frac{k_\mathcal{H}-r^*}{2r_u}\right|^n\right]\dot{x}_1, & \text{other branches} \end{cases}, \quad (26)$$

where K and r_u are system properties for initial stiffness and ultimate strength, respectively, while n controls the transition rate from fully elastic to fully plastic behavior. The restoring force value r^* needs to be updated "on the fly" following the appropriate load reversal points after implementing the EMM switching rules in a numerical simulation. The system's parameters are shown in Table 1.

Following [49], the rule-based computation of the EMM is numerically realized in this study using a hybrid dynamical system approach for the extensively involved nonlocal memory. This 2-DOF system is solved by using the state event location algorithm under MATLAB routine ode45 [50]. The hysteretic spring response in terms of the (unmeasured) displacement x_1 is visualized in Fig. 7. Overall, it can be observed that the hysteresis nonlinearity is almost insignificant at high frequencies while it is dominant at low frequencies. Additionally, the hysteresis width is the greatest at approximately 3 Hz, which is the first system resonance. Finally, the measured output x_2 is affected by zero-mean white Gaussian noise with a signal-to-noise ratio of 50 dB. The objective is to find a dynamic relationship between the input force $u(k)$ and the measured displacement $x_2(k)$.

6.2. The data

Three datasets were considered for the nonlinear identification problem. The training dataset was used for the estimation of an LTI model as well as for training the PI-LFR model structure. The remaining two test datasets were used to evaluate the performance of the trained model on unseen data.

6.2.1. Training data

The training dataset was obtained under multisine excitation of the MSD system. The input signal consists of 3 signal realizations using Eq. (18) with $N_s = 1024$, $f_{max} = 30$ Hz, $f_s = 120$ Hz and normalized with respect to their standard deviation. Each signal realization consists of two half periods that are shaped by a Tukey window function and one full steady-state period in between. The implemented Tukey window function ensures that the transition between each signal realization is smooth and thus avoiding any potential numerical issues during the simulation of the system.
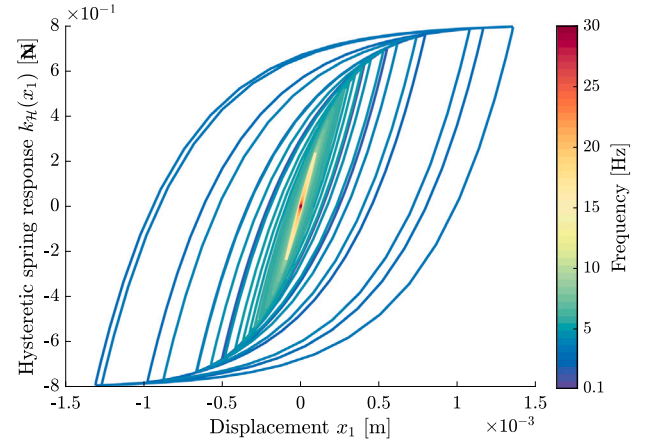

Fig. 7. Mapping between the hysteretic spring response $k_\mathcal{H}(x_1)$ and the unmeasured state variable x_1 of the mass–spring–damper system for various frequencies of the input signal u. The hysteretic loops were generated using a unit amplitude sine sweep signal that covers the frequency band from 0.1 Hz to 30 Hz at a sweep rate of 1.17 Hz/s.
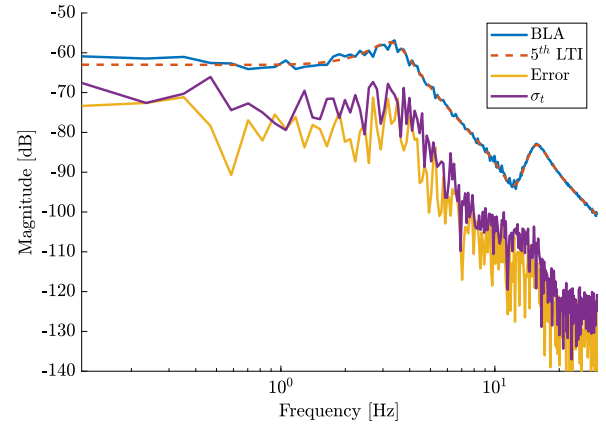

Fig. 8. Linear analysis of the 2-DOF mass–spring–damper system. The BLA estimate is shown in blue, the magnitude response of the 5th order LTI model is shown in red, the error between them is shown in yellow and the total variance σ_t^2 (variance due to noise + stochastic nonlinear distortion) of the BLA estimate is shown in purple.

6.2.2. Test data

Two test datasets are considered to evaluate the performance of the trained nonlinear model and are obtained by simulating the system under multisine and sinesweep excitations:

- The **multisine test** data was obtained using a multisine input signal that has the same properties as the one used for training (same frequency range but with different random phases).
- The **sinesweep test** data was obtained using a unit amplitude sinesweep input signal that covers the frequency band from 0.1 Hz to 30 Hz at a sweep rate of 1.17 Hz/s.

6.3. Linear analysis results

A linear model of the hysteretic 2-DOF mass–spring–damper system was obtained using the robust approach of the BLA framework [41]. The BLA was estimated using only the three steady-state periods and is shown in Fig. 8. A parametric LTI model was identified starting from the BLA estimate and its magnitude response is also shown in Fig. 8. The order of the LTI model was set to 5 during the identification in order to capture any dynamical effect presented by the hysteresis nonlinearity. With this additional state, the identified parametric LTI model resulted in lower error especially at low frequencies compared
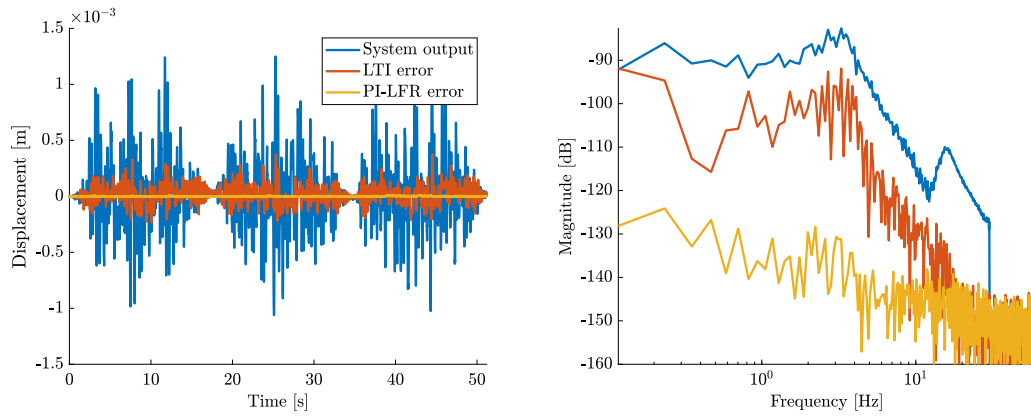
Fig. 9. Time-domain (left) and frequency-domain (right) test results of the trained PI-LFR model structure using multisine signals. The true system output is shown in blue, the residuals obtained with a 5th-order LTI model (the one used to initialize the PI-LFR model) are shown in red and the residuals obtained with the PI-LFR model are shown in yellow.

to a 4th-order LTI model. The error between the BLA estimate and the 5th-order LTI model can be seen matching the total distortion level. This indicates that the identified LTI model sufficiently captures all the linear dynamics of the system. This linear model is used for the initialization of the PI-LFR model structure.

6.4. Nonlinear model identification results

Two PI-LFR models, one with $N = 10$ and a second with $N = 20$ play operator neurons were trained using the training dataset. The proposed PI-LFR approach is compared to more classical (non)linear system identification approaches:

- A 5th-order LTI state–space model identified using the 'ssest' Matlab command [51],
- A general purpose nonlinear state–space model similar to the one used in [22] to identify a Bouc–Wen hysteretic system. However, this time with a 1-hidden layer neural network with n_n neurons representing the nonlinear state-transition and output functions [52] instead of a polynomial. 4 cases are considered covering $n_x \in [5, 10]$ and $n_n \in [10, 20]$. These models are denoted as state–space neural networks (SS-NN),
- A Hammerstein model using a PI hysteretic nonlinearity (PI-Hamm) with N PI element, where $N \in [10, 20]$, and a 5th-order state–space LTI block [53]. This model is trained in the same way as the PI-LFR model. Actually, the PI Hammerstein model is a special case of the PI-LFR model class.
- A NARX models with $n_b = n_a \in [5, 10]$ delayed inputs and outputs and a 1-hidden layer neural network with $n_n \in [5, 10]$ neurons to model the nonlinearity [54].

The optimization of the PI-LFR, the SS-NN, and the NARX model is performed using the Neural Network Toolbox of Matlab. The PI-LFR and the SS-NN model training use a Levenberg–Marquardt optimizer with a maximum of 1000 epochs. However, all identification runs converged before the maximum epoch was reached. Beyond varying the classical hyperparameters, such as the number of input and output lags and the number of neurons in the neural network, the NARX model training uses the standard narxnet settings of Matlab.

As a measure of model quality, the normalized RMS error (normalized with respect to the standard deviation of the output signal) is reported for the three datasets:

$$e_{rms} = \sqrt{\frac{1}{M} \sum_{k=1}^{M} \left(\frac{y(k) - \hat{y}(k|\theta_o)}{\sigma_y} \right)^2}, \tag{27}$$

where σ_y represents the standard deviation of the output signal.

The time domain and frequency domain test results of the PI-LFR ($N = 20$) under multisine and sinesweep excitations are shown in Figs. 9 and 10 respectively. It is observed that the trained PI-LFR model significantly outperforms the LTI model.

Furthermore, a Monte-Carlo simulation of 100 independent runs has been carried out to study the variability of the final identified models due to the random initialization of the model parameters. The results, shown in Fig. 11, indicate significant variability in the quality of the model. However, for an increasing number of PI elements, it becomes more likely to obtain high-quality models (i.e. the 25th percentile is dropping significantly, and less high outliers are observed). When compared to the PI-LFR model with fewer hysteresis operator neurons, the reported PI-LFR $N = 20$ model achieves better performance for the multisine cases while it performs slightly worse for the sinesweep case (see also Table 2). The estimated standard deviation of the noise $\hat{\sigma}_n = 0.0033$ is close to the RMS errors obtained by the PI-LFR models. Hence, it is reasonable to expect that by running the identification procedure 5–10 times, a high-quality estimate will be obtained. Note that a single run of the identification algorithm on the considered simulation problem takes approximately 2 minutes on an average laptop (Intel Core i7-10750H CPU 2.60 GHz, 16 GB RAM, Windows 10).

Table 2 shows a comparison of the results obtained using the considered model structures for various model orders. The results reported for the PI-LFR model correspond to the 25th percentile of the boxplots shown in Fig. 11. It can be observed that the reported PI-LFR model outperforms all the other approaches obtaining a model error that is approximately 10× smaller than the closest other result.

The black-box identification approaches using the SS-NN and the NARX model structure have difficulty obtaining high-quality results due to the challenging dynamic nature of the Masing hysteretic nonlinearity present in the data-generating system which increases the order of the dynamics drastically.[1] Such a nonlinearity requires a large state space (SS-NN), or a large number of delayed inputs and outputs (NARX) to capture the hysteresis dynamics. Due to this, it can be observed that the SS-NN models are quickly overfitting when increasing the numbers of states from $n_x = 5$ to $n_x = 10$, and even can become unstable on the test set, using the given training dataset. The PI-LFR model adds the hysteretic dynamics by increasing the number of PI hysteresis neurons in the model, each neuron contains one state. Finally, the PI-Hammerstein model is not even outperforming the LTI model. This is mainly due to the mismatch between the model structure and the true system structure. A Hammerstein model assumes that the hysteretic nonlinearity is present at the input of the system, while this is not the

---

[1] This is in contrast with, for instance, Bouc–Wen hysteretic systems, that only add one extra state in the state–space equations, see for instance [22]
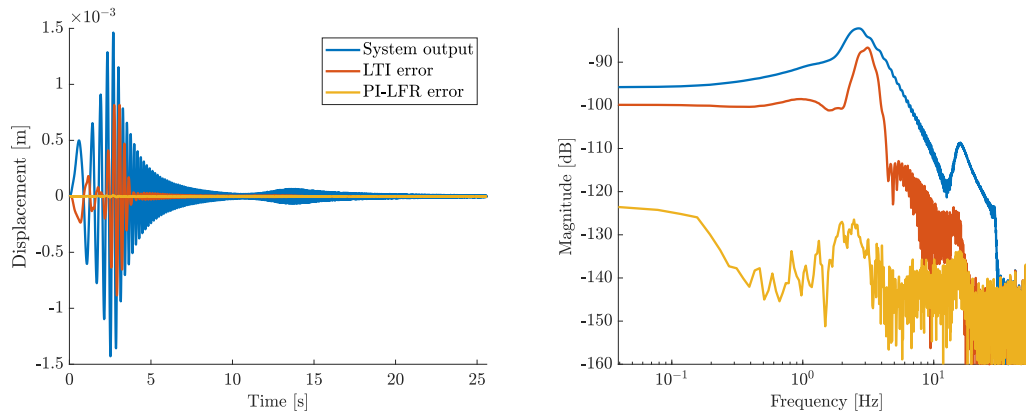
Fig. 10. Time-domain (left) and frequency-domain (right) test results of the trained PI-LFR model structure using the sinesweep test data. The true system output is shown in blue, the residuals obtained with a 5th-order LTI model (the one used to initialize the PI-LFR model) are shown in red and the residuals obtained with the PI-LFR model are shown in yellow.
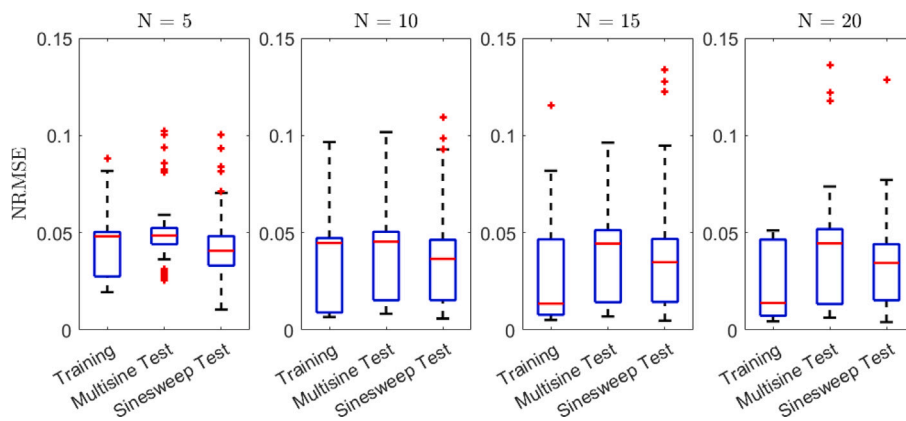

Fig. 11. Monte-Carlo simulation results for the LFR-PI identification using varying numbers of PI operators (N). The boxplots indicate the median (horizontal red line) and the 25th and 75th percentile interval (blue box). The red crosses indicate data points that are considered to be outliers.

Table 2
Normalized RMS simulation error of the identified models of the hysteretic MSD system.

	Multisine training	Multisine test	Sinesweep test
LTI (5th order)	0.3026	0.2910	0.4799
PI-LFR$_{N=5}$	0.0275	0.0441	0.0331
PI-LFR$_{N=10}$	0.0090	0.0153	0.0154
PI-LFR$_{N=15}$	0.0078	0.0143	**0.0144**
PI-LFR$_{N=20}$	**0.0073**	**0.0134**	0.0153
SS-NN$_{n_x=5, n_n=5}$	0.0595	0.1410	0.3982
SS-NN$_{n_x=5, n_n=10}$	0.0380	0.1278	1.2101
SS-NN$_{n_x=10, n_n=5}$	0.0441	Unstable	0.2172
SS-NN$_{n_x=10, n_n=10}$	0.0357	0.2727	2.0248
PI-Hamm$_{N=10, n_x=5}$	0.2826	0.3240	0.5096
PI-Hamm$_{N=20, n_x=5}$	0.2809	0.3120	0.4930
NARX$_{n_a=n_b=5, n_n=5}$	0.2411	0.2289	0.1611
NARX$_{n_a=n_b=5, n_n=10}$	0.2341	0.2285	0.1606
NARX$_{n_a=n_b=10, n_n=5}$	0.1479	0.1592	0.1800
NARX$_{n_a=n_b=10, n_n=10}$	0.1600	0.2257	0.1716

case in this example. Hence, the Hammerstein model cannot capture the hysteretic behavior of the system.

7. Experimental validation

In this section, the proposed model structure and identification method are experimentally validated on a practical measurement application.

7.1. System description

In order to assess the performance of the proposed model, real physical piezo-driven motion platform measurement results have been utilized, obtained from the experimental setup depicted in Fig. 12. It consists of a piezoelectric actuator that is placed between a mass and a pillar which is mounted to a frame. The piezoelectric actuator is a lead zirconate-titanate Pb(ZrTi)O$_3$, commonly called PZT. Pre-loading springs are used to pull the mass on top of the actuator. Capacitive sensors are located on both sides of the mass which measure its displacement. The considered experimental setup can be roughly represented as a classical mass–spring–damper (MSD) system where the input restoring force is generated by the piezoelectric actuator. The main objective is to find a dynamic relation between the input signal which is the voltage that drives the piezo and the output displacement of the mass.

7.2. Linear analysis

Before the nonlinear model identification starts, a linear model of the system is estimated over a broad frequency range (0–3000 Hz) using 10 multisine realizations with 5 steady-state periods of 8192 samples each as an input signal. A nonparametric linear model is obtained using the robust approach of the BLA framework and afterwords a parametric model is identified using the frequency domain subspace identification method [46].

The normalized BLA estimate is shown in Fig. 13. At first glance, it can be seen that the FRF of the BLA estimate is much more complicated
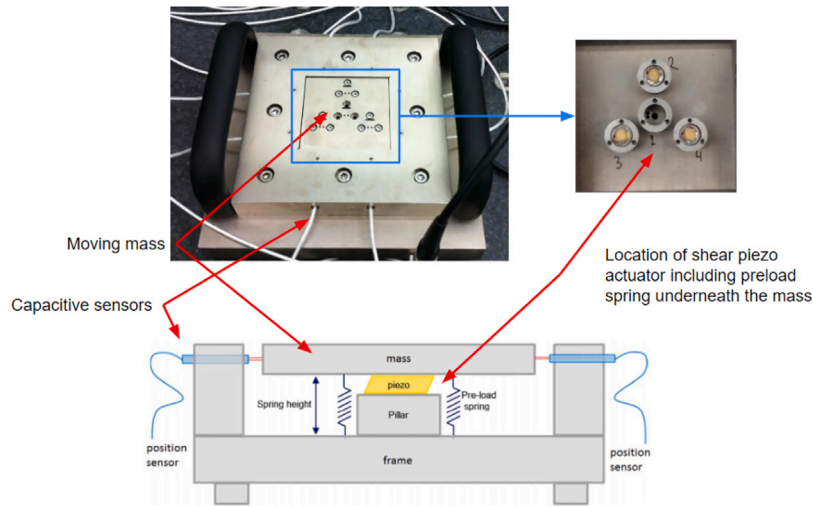
Fig. 12. Foto and schematic side view of the considered experimental piezo-driven motion setup [55].
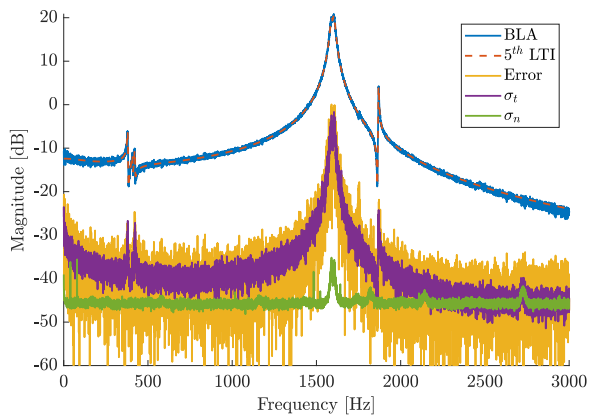

Fig. 13. Linear analysis of the system using periodic multisine signals. The estimated BLA is shown in blue, the total variance of the BLA estimate is shown in purple, the variance of the BLA estimate due to noise errors is shown in green, the identified LTI parametric BLA is shown in red and the error between non-parametric and parametric models is shown in yellow.

if compared to the FRF of a classical MSD system. There are multiple resonance peaks present, both at low and high frequencies. The dominant resonance peaks are located at approximately 1.7 KHz and are 35 dB higher than the DC level of the system. The total variance of the BLA estimate which is due to the nonlinearities and noise errors is also shown in Fig. 13, which is higher than the noise variance at low frequencies. This indicates the presence of nonlinear distortions at that frequency range. The nonlinear distortion level is the largest in the dominant resonance peaks of the system.

A parametric 15th order LTI model was identified starting from the non-parametric BLA estimate. The magnitude response of the parametric model as well as the error between parametric and non-parametric models are shown in Fig. 13. By comparing the total variance with the error, it can be observed that the linear model manages to capture all the linear dynamics since the error is matching the total distortion level. This linear model is used as an initial estimate for the LTI block in the nonlinear model structure.

7.3. Nonlinear model identification

A nonlinear model for the piezo-actuated system was identified by exciting the piezo-actuated motion system over the bandwidth that is of interest during system operation (0–800 Hz). 9 multisine input

signal realizations (Eq. (18)) with $f_{max} = 800$ Hz, $f_s = 10$ kHz, $N_s = 8192$ number of samples are considered for training. 1 multisine input realization is considered for testing. 5 steady-state periods of the system response were measured for each multisine realization.

The same model classes are considered during the identification of the piezo-actuated system as in Section 6:

- A state–space LTI model with $n_x = 8$ states (obtained using all 10 multisine realizations).[2]
- Nonlinear Output Error (NOE) models with n_b, n_a delayed inputs and outputs, where $n_b = n_a \in [5, 10]$ delayed inputs and outputs and a 1-hidden layer neural network with $n_n \in [10, 20]$ neurons to model the nonlinearity. NOE models are known as closed-loop NARX networks in the Matlab Neural Network toolbox [54].
- SS-NN models with n_x states, $n_x \in [8, 15]$, and n_n neurons, $n_n \in [10, 20]$ to represent the nonlinearities present in the model.
- PI-LFR models with n_x states with $n_x \in [8, 15]$ states and with N play operator neuron elements, where $N \in [10, 20]$.
- A Hammerstein model using a PI hysteretic nonlinearity (PI-Hamm) with n_x states with $n_x \in [8, 15]$ states and with N play operator neuron elements, where $N \in [10, 20]$ [53].
- An estimate of the noise floor obtained by averaging over the measured steady-state periods is also provided as a reference.

The NOE model structure is used instead of the NARX structure as the NARX models failed to generalize well towards the simulation setting. NOE models are known to generalize better in simulation tasks [56]. Furthermore, early stopping is applied while training the NOE models to avoid strong overfitting.

The time-domain and frequency-domain test results of the estimated PI-LFR model are shown in Fig. 14. It can be observed that the estimated PI-LFR model, as expected, outperforms the LTI model and that it achieves a model error that is for most frequencies close to the noise level of the output after averaging over the periods.

As a measure of model quality, the normalized RMSE given in Eq. (27) is reported. Table 3 shows a comparison of the results obtained using the considered model classes. It can be observed that the PI-LFR model is over two times more accurate compared to the LTI model, about 10% better than the best SS-NN result. Both the PI-LFR$_{n_x=15, N=10}$ and PI-LFR$_{n_x=15, N=20}$ models obtain high-quality results. While the $N = 10$ case has the lowest training result, the $N = 20$ result performs better

---

[2] Observe that a lower model order is used to obtain this linear estimate compared to Section 7.2 as the dataset considered in this section only ranges from 0 to 800 Hz, while the dataset in Section 7.2 ranged up to 3000 Hz.
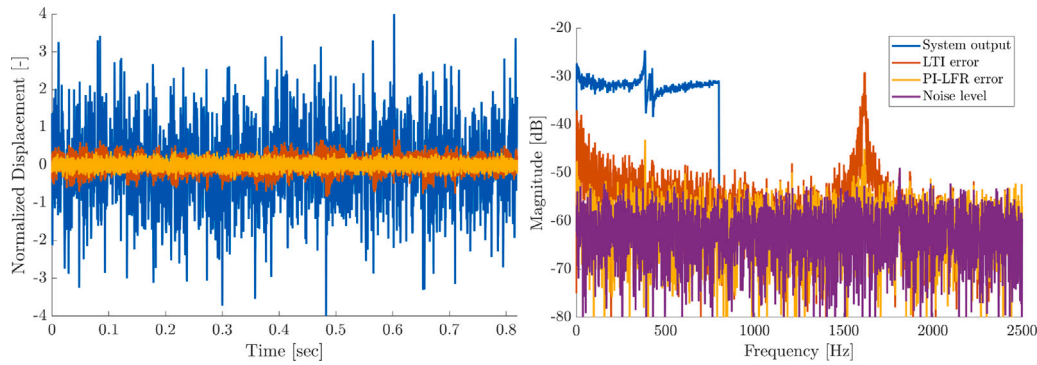
Fig. 14. Time-domain (left) and frequency-domain (right) test of the identified nonlinear model. The true system output is shown in blue, the residuals obtained with an LTI model are shown in red and the residuals obtained with the nonlinear PI-LFR model are shown in yellow. An estimate of the noise level in the frequency domain is shown in purple.
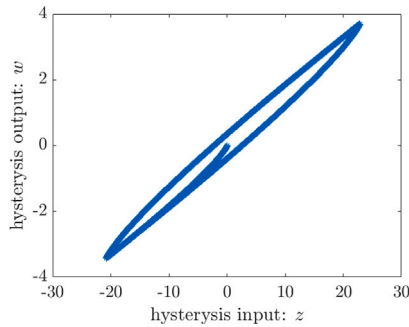

Fig. 15. Identified hysteretic mapping between the PI-LFR model input of the PI submodel ($z(k)$) and output of the PI submodel ($w(k)$).

in test. These differences can be partially explained by the variability present in the data, and by the nonlinear-in-the-parameters cost that requires minimization. On top of this, the PI-LFR model uses a much more structured representation than the SS-NN model. This allows for easier analysis and a better explainability of the obtained model. For example, the identified hysteretic nonlinearity can easily be extracted, see Fig. 15 and visualized by using the w and z signals in the PI-LFR model. The PI-Hamm results indicate that the underlying system structure almost has a Hammerstein structure as the PI-LFR results are only 3% better than the best PI-Hamm result. Finally, observe that the normalized RMS error for the estimated PI-LFR model is not far away from the estimated noise floor after averaging over the periods, confirming the results observed in Fig. 14. The dominant remaining error is present in the system resonance peaks around 0.4 and 1.6 kHz. This error is probably caused by additional nonlinearities present in the system.

8. Conclusion

This paper has introduced a structured nonlinear state–space model identification approach for hysteretic systems by using a flexible interconnection of PI artificial neural network hysteresis operators with a linear state–space model through a linear fractional representation. Firstly, we have demonstrated how Prandtl–Ishlinskii hysteretic operators can be represented as compact recurrent networks called PI hysteresis operator neurons using ReLU and SSLU activation functions. Such PI hysteresis operator neurons, connected in parallel into an ANN layer, have the flexibility to be combined with classical recurrent neural networks or other dynamical system representations. Hence, it is able to model complex hysteretic systems using widely available and effective ANN optimization tools. Secondly, we introduced an efficient initialization of the PI-LFR model parameters. The proposed

Table 3
Normalized RMS errors for the multisine training and test dataset reported for LTI, SS-NN, NOE, and PI-LFR model estimates. An estimate of the noise floor is also reported.

	Training	Test
$\text{LTI}_{n_x=8}$	0.2104	0.2084
$\text{NOE}_{n_a=n_b=8,\,n_n=10}$	0.1719	0.1713
$\text{NOE}_{n_a=n_b=8,\,n_n=20}$	0.2229	0.2256
$\text{NOE}_{n_a=n_b=15,\,n_n=10}$	0.1590	0.1612
$\text{NOE}_{n_a=n_b=15,\,n_n=20}$	0.1610	0.1645
$\text{SS-NN}_{n_x=8,\,n_n=10}$	0.1891	0.1891
$\text{SS-NN}_{n_x=8,\,n_n=20}$	0.1905	0.1914
$\text{SS-NN}_{n_x=15,\,n_n=10}$	0.1148	0.1155
$\text{SS-NN}_{n_x=15,\,n_n=20}$	0.1181	0.1289
$\text{PI-Hamm}_{n_x=8,\,N=10}$	0.1820	0.1860
$\text{PI-Hamm}_{n_x=8,\,N=20}$	0.1827	0.1863
$\text{PI-Hamm}_{n_x=15,\,N=10}$	0.1060	0.1072
$\text{PI-Hamm}_{n_x=15,\,N=20}$	0.1073	0.1098
$\text{PI-LFR}_{n_x=8,\,N=10}$	0.1862	0.1871
$\text{PI-LFR}_{n_x=8,\,N=20}$	0.1856	0.1868
$\textbf{PI-LFR}_{n_x=15,\,N=10}$	**0.1056**	0.1052
$\text{PI-LFR}_{n_x=15,\,N=20}$	0.1057	**0.1044**
Noise level	0.0884	

PI-LFR approach results in the simultaneous identification of both the hysteretic and other system dynamics present. The effectiveness of the proposed approach is validated using a 2-DOF mass–spring–damper system with a Masing-type hysteretic spring simulation example and on measured piezo-actuated motion platform data. These are two illustrative examples of how this study can benefit estimation techniques, measurement data processing algorithms, and mathematical models for measurement-oriented purposes.

CRediT authorship contribution statement

Konstantinos Krikelis: Methodology, Software, Validation, Investigation, Writing – original draft, Visualization, Data curation, Formal analysis. **Jin-Song Pei:** Software, Validation, Investigation, Resources, Writing – review & editing, Data curation. **Koos van Berkel:** Conceptualization, Methodology, Validation, Investigation, Resources, Supervision, Project administration. **Maarten Schoukens:** Conceptualization, Methodology, Software, Validation, Investigation, Writing – original draft, Writing – review & editing, Formal analysis, Visualization, Supervision, Project administration.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data that has been used is confidential.

Appendix. Derivations

A.1. Neural play operator derivation

A detailed study of the neural play operator representation is given below. The play operator defined in Eq. (5) can be visualized in Fig. A.16 while the proposed recurrent neural network representation is given in Eq. (11).

As the function $g(\cdot)$ is applied element-wise, Eq. (11) can be split into the following:

$$
\begin{aligned}
s_1(k) &= g(u(k) - y(k-1) + r) \\
s_2(k) &= g(u(k) - y(k-2) - r) \\
y(k) &= u(k) + s_2(k) - s_1(k) + r
\end{aligned}
\tag{A.1}
$$

The activation function $g(\cdot)$ is given by the ReLu function. Hence, these equations can be written as:

$$
s_1(k) = \begin{cases} u(k) - y(k-1) + r & \text{if } y(k-1) < u(k) + r \\ 0 & \text{if } y(k-1) \geq u(k) + r \end{cases}
\tag{A.2}
$$

$$
s_2(k) = \begin{cases} u(k) - y(k-1) + r & \text{if } y(k-1) < u(k) - r \\ 0 & \text{if } y(k-1) \geq u(k) - r \end{cases}
\tag{A.3}
$$

Since $r > 0$ if it holds that $y(k-1) < u(k) - r$, it also holds that $y(k-1) < u(k) + r$. Hence we can distinguish three possible cases:

a. If $y(k-1) \geq u(k) - r$ and $y(k-1) \geq u(k) + r$, then

$$
y(k) = u(k) + 0 - 0 + r = u(k) + r
\tag{A.4}
$$

b. If $y(k-1) \geq u(k) - r$ and $y(k-1) < u(k) + r$

$$
y(k) = u(k) + 0 - (u(k) - y(k-1) + r) + r = y(k-1)
\tag{A.5}
$$

c. If $y(k-1) < u(k) - r$ and $y(k-1) < u(k) + r$

$$
y(k) = u(k) + (u(k) - y(k-1) - r) - (u(k) - y(k-1) + r) + r = u(k) - r
\tag{A.6}
$$

These expressions match exactly the original Play Operator (Eq. (5)) and its visualization in Fig. A.16.

A.2. Generalized play operator derivation

A detailed study of the generalized play operator representation is given below. The generalized play operator is obtained by introducing the left and right bounding functions γ_l and γ_r as in Eq. (7), where $\gamma_l(u) < \gamma_r(u)$ for all u and $r > 0$. This equation, similar to the standard play operator, can be visualized in Fig. A.17

As $g(\cdot)$ is the ReLu activation function and it is applied element-wise, we can again evaluate the different cases in a piece-wise manner:

$$
s(k) = \begin{cases} y(k-1) & \text{if } y(k-1) < \gamma_r(u(k)) + r \\ \gamma_r(u(k)) + r & \text{if } y(k-1) \geq \gamma_r(u(k)) + r \end{cases}
\tag{A.7}
$$

$$
y(k) = \begin{cases} \gamma_l(u(k)) - r & \text{if } s(k) < \gamma_l(u(k)) - r \\ s(k) & \text{if } s(k) \geq \gamma_l(u(k)) - r \end{cases}
\tag{A.8}
$$

Similarly, we can distinguish three possible cases:

a. If $y(k-1) \geq \gamma_r(u(k)) + r$ and $s(k) = \gamma_r(u(k)) + r$, then as $\gamma_l(u) < \gamma_r(u)$ we have that $s(k) \geq \gamma_l(u(k)) - r$. It follows that

$$
y(k) = s(k) = \gamma_r(u(k)) + r
\tag{A.9}
$$

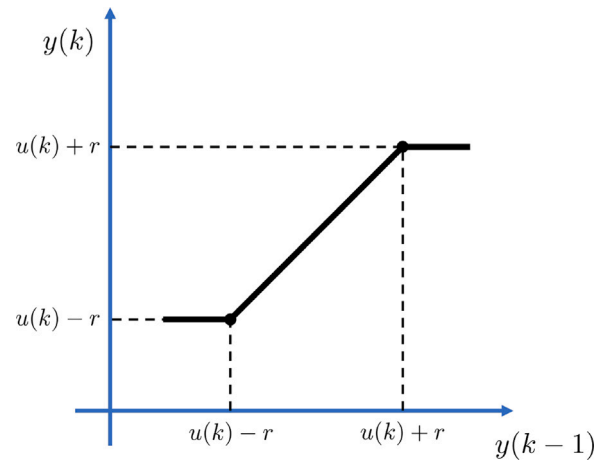b. If $y(k-1) < \gamma_r(u(k)) + r$, then $s(k) = y(k-1)$

Fig. A.16. An alternative visual representation of the play operator as prescribed by Eq. (5).
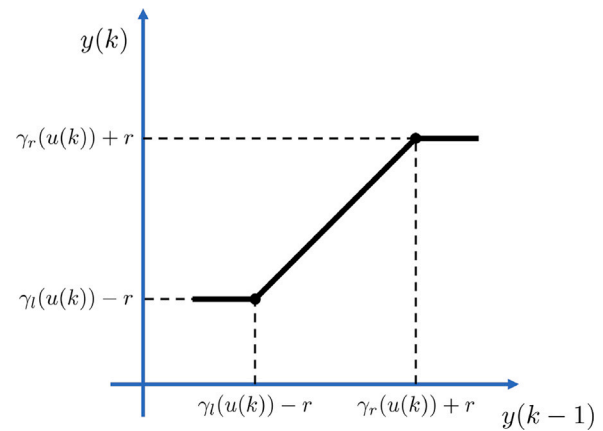

Fig. A.17. An alternative visual representation of the generalized play operator as prescribed by Eq. (7).

i. If $y(k-1) < \gamma_r(u(k)) - r$, then we have that:

$$
y(k) = \gamma_r(u(k)) - r
\tag{A.10}
$$

ii. If $y(k-1) \geq \gamma_r(u(k)) - r$, then we have that:

$$
y(k) = s(k) = y(k-1)
\tag{A.11}
$$

These expressions match exactly the generalized Play Operator equation (Eq. (7)), and its visualization in Fig. A.17.

References

[1] T.J. Mueller, The influence of laminar separation and transition on low Reynolds number airfoil hysteresis, J. Aircr. 22 (9) (1985) 763–770, http://dx.doi.org/10.2514/3.45199.

[2] G. Bertotti, Hysteresis in Magnetism: For Physicists, Materials Scientists, and Engineers, Academic Press, San Diego, CA, 1998.

[3] D.J. Morrison, Y. Jia, J.C. Moosbrugger, Cyclic plasticity of nickel at low plastic strain amplitude: Hysteresis loop shape analysis, Mater. Sci. Eng. A 314 (1) (2001) 24–30, http://dx.doi.org/10.1016/S0921-5093(00)01914-6.

[4] A. Wońiak, K. Meczyńska, Measurement hysteresis of touch-trigger probes for CNC machine tools, Measurement 156 (2020) 107568.

[5] F.J. Flores-Ruiz, J.J. Gervacio-Arciniega, E. Murillo-Bracamontes, M.P. Cruz, J.M.Y. nez Limón, J.M. Siqueiros, An alternative scheme to measure single-point hysteresis loops using piezoresponse force microscopy, Measurement 108 (2017) 143–151.

[6] B. Aydemir, L. Yagmur, S. Fank, Hysteresis errors of commonly used sensor materials, Measurement 43 (6) (2010) 792–796.

[7] B. Aydemir, E. Kaluc, S. Fank, Influence of heat treatment on hysteresis error of force transducers manufactured from 17-4PH stainless steel, Measurement 39 (10) (2006) 892–900.

[8] J.M. Dias Pereira, P. Silva Girao, A. Cruz Serra, Dithering performance of oversampled ADC systems affected by hysteresis, Measurement 32 (1) (2002) 51–59.

[9] H. Habibullah, 30 Years of atomic force microscopy: Creep, hysteresis, cross-coupling, and vibration problems of piezoelectric tube scanners, Measurement 159 (2020) 107776.

[10] Y. Guo, C. Ma, X. Dong, Y. Liang, B. Hu, A novel health indicator based on hysteresis loop for health prediction of flight control systems, Measurement 186 (2021) 110076.

[11] R.V. Iyer, X. Tan, Control of hysteretic systems through inverse compensation, IEEE Control Syst. Mag. 29 (1) (2009) 83–99.

[12] J. Gan, X. Zhang, A review of nonlinear hysteresis modeling and control of piezoelectric actuators, AIP Adv. 9 (4) (2019) 040702, http://dx.doi.org/10.1063/1.5093000.

[13] G. Gu, L. Zhu, C. Su, H. Ding, S. Fatikow, Modeling and control of piezo-actuated nanopositioning stages: A survey, IEEE Trans. Autom. Sci. Eng. 13 (1) (2016) 313–332, http://dx.doi.org/10.1109/TASE.2014.2352364.

[14] M. Brokate, J. Sprekels, Hysteresis and Phase Transitions, Springer New York, 1996, http://dx.doi.org/10.1007/978-1-4612-4048-8.

[15] M. Al Janaideh, S. Rakheja, C. Su, An analytical generalized Prandtl–Ishlinskii model inversion for hysteresis compensation in micropositioning control, IEEE/ASME Trans. Mechatronics 16 (4) (2011) 734–744, http://dx.doi.org/10.1109/TMECH.2010.2052366.

[16] P. Berenyi, G. Horvath, V. Lampaert, J. Swevers, Nonlocal hysteresis function identification and compensation with neural networks, IEEE Trans. Instrum. Meas. 54 (6) (2005) 2227–2238, http://dx.doi.org/10.1109/TIM.2005.858822.

[17] F. Al-Bender, V. Lampaert, J. Swevers, The generalized Maxwell-slip model: A novel model for friction simulation and compensation, IEEE Trans. Automat. Control 50 (11) (2005) 1883–1887, http://dx.doi.org/10.1109/TAC.2005.858676.

[18] J.T. Hsu, K.D.T. Ngo, A Hammerstein-based dynamic model for hysteresis phenomenon, IEEE Trans. Power Electron. 12 (3) (1997) 406–413, http://dx.doi.org/10.1109/63.575667.

[19] F. Giri, Y. Rochdi, F.Z. Chaoui, A. Brouri, Identification of Hammerstein systems in presence of hysteresis-backlash and hysteresis-relay nonlinearities, Automatica 44 (3) (2008) 767–775, http://dx.doi.org/10.1016/j.automatica.2007.07.005.

[20] M.A. Janaideh, D.S. Bernstein, Adaptive control of Hammerstein systems with unknown Prandtl–Ishlinskii hysteresis, Proc. Inst. Mech. Eng. I 229 (2) (2015) 149–157, http://dx.doi.org/10.1177/0959651814551660.

[21] A.Y.K. Yong, A.H. Tan, C.L. Cham, Identification of block-oriented systems with rate saturation nonlinearity, IFAC-PapersOnLine 48 (28) (2015) 939–944, http://dx.doi.org/10.1016/j.ifacol.2015.12.251, 17th IFAC Symposium on System Identification SYSID 2015.

[22] J.P. Noël, A.F. Esfahani, G. Kerschen, J. Schoukens, A nonlinear state-space approach to hysteresis identification, Mech. Syst. Signal Process. 84 (B) (2017) 171–184, http://dx.doi.org/10.1016/j.ymssp.2016.08.025.

[23] J. Paduart, L. Lauwers, J. Swevers, K. Smolders, J. Schoukens, R. Pintelon, Identification of nonlinear systems using polynomial nonlinear state space models, Automatica 46 (4) (2010) 647–656, http://dx.doi.org/10.1016/j.automatica.2010.01.001.

[24] M. Schoukens, Y. Rolain, Cross-term elimination in parallel Wiener systems using a linear input transformation, IEEE Trans. Instrum. Meas. 61 (3) (2012) 845–847, http://dx.doi.org/10.1109/TIM.2011.2174851.

[25] A. Fakhrizadeh Esfahani, P. Dreesen, K. Tiels, J.P. Noël, J. Schoukens, Parameter reduction in nonlinear state-space identification of hysteresis, Mech. Syst. Signal Process. 104 (2018) 884–895, http://dx.doi.org/10.1016/j.ymssp.2017.10.017.

[26] C. Novara, T.L. Vincent, K. Hsu, M. Milanese, K. Poolla, Parametric identification of structured nonlinear systems, Automatica 47 (2011) 711–721, http://dx.doi.org/10.1016/j.automatica.2011.01.063.

[27] L. Vanbeylen, Nonlinear LFR block-oriented model: Potential benefits and improved, user-friendly identification method, IEEE Trans. Instrum. Meas. 62 (12) (2013) 3374–3383, http://dx.doi.org/10.1109/TIM.2013.2272868.

[28] M. Schoukens, R. Toth, On the initialization of nonlinear LFR model identification with the best linear approximation, in: IFAC 2020 World Congress, Berlin, Germany. 12 - 17 july, 2020.

[29] M. Schoukens, K. Tiels, Identification of block-oriented nonlinear systems starting from linear approximations: A survey, Automatica 85 (2017) 272–292, http://dx.doi.org/10.1016/j.automatica.2017.06.044.

[30] M. Schoukens, K. Tiels, Identification of block-oriented nonlinear systems starting from linear approximations: A survey, Automatica 85 (2017) 272–292.

[31] M.A. Krasnosel'skil, A.V. Pokrovskii, Systems with Hysteresis, Springer-Verlag Berlin Heidelberg, 1989, http://dx.doi.org/10.1007/978-3-642-61302-9.

[32] P. Krejci, K. Kuhnen, Inverse control of systems with hysteresis and creep, IEEE Proc. D 148 (3) (2001) 185–192, http://dx.doi.org/10.1049/ip-cta:20010375.

[33] A. Joghataie, M. Farrokh, Dynamic analysis of nonlinear frames by Prandtl neural networks, J. Eng. Mech. 134 (11) (2008) 961–969, http://dx.doi.org/10.1061/(ASCE)0733-9399(2008)134:11(961).

[34] Y. Chen, J. Qiu, H. Sun, A hybrid model of Prandtl–Ishlinskii operator and neural network for hysteresis compensation in piezoelectric actuators, Int. J. Appl. Electromagn. Mech. 41 (3) (2013) 335–347, http://dx.doi.org/10.3233/JAE-131648.

[35] X. Zhang, Y. Tan, M. Su, Y. Xie, Neural networks based identification and compensation of rate-dependent hysteresis in piezoelectric actuators, Physica B 405 (12) (2010) 2687–2693, http://dx.doi.org/10.1016/j.physb.2010.03.050.

[36] M. Farrokh, M.S. Dizaji, F.S. Dizaji, N. Moradinasab, Universal hysteresis identification using extended Preisach neural network, 2019, arXiv:2001.01559.

[37] L. Ma, Y. Shen, J. Li, A neural-network-based hysteresis model for piezoelectric actuators, Rev. Sci. Instrum. 91 (1) (2020) 015002, http://dx.doi.org/10.1063/1.5121471.

[38] G. Cybenko, Approximation by superpositions of a sigmoidal function, Math. Control Signals Systems 2 (4) (1989) 303–314, http://dx.doi.org/10.1007/bf02551274.

[39] B. Hanin, Universal function approximation by deep neural nets with bounded width and ReLU activations, Mathematics 7 (10) (2019) 992, http://dx.doi.org/10.3390/math7100992.

[40] D. Stursa, P. Dolezel, Comparison of ReLU and linear saturated activation functions in neural network for universal approximation, in: 2019 22nd International Conference on Process Control, PC19, 2019, pp. 146–151, http://dx.doi.org/10.1109/PC.2019.8815057.

[41] R. Pintelon, J. Schoukens, System Identification: A Frequency Domain Approach, second ed., Wiley-IEEE Press, 2012.

[42] K. Levenberg, A method for the solution of certain non-linear problems in least squares, Quart. Appl. Math. 2 (2) (1944) 164–168, http://dx.doi.org/10.1090/qam/10666.

[43] D.W. Marquardt, An algorithm for least-squares estimation of nonlinear parameters, J. Soc. Ind. Appl. Math. 11 (2) (1963) 431–441, http://dx.doi.org/10.1137/0111030.

[44] M.T. Hagan, H.B. Demuth, M. Beale, O. De Jesús, Neural Network Design, second ed., PWS Publishing Company, 2014.

[45] A.Y.K. Yong, A.H. Tan, C.L. Cham, Identification of block-oriented systems with rate saturation nonlinearity, IFAC-PapersOnLine 48 (28) (2015) 939–944, http://dx.doi.org/10.1016/j.ifacol.2015.12.251.

[46] R. Pintelon, Frequency-domain subspace system identification using non-parametric noise models, Automatica 38 (8) (2002) 1295–1311, http://dx.doi.org/10.1016/s0005-1098(02)00036-5.

[47] M. Goldfarb, N. Celanovic, Modeling piezoelectric stack actuators for control of micromanipulation, in: Proceedings of the 1996 IEEE International Conference on Robotics and Automation, Minneapolis, MN, 1996, pp. 69–79, http://dx.doi.org/10.1109/37.588158.

[48] P.C. Jennings, Periodic response of a general yielding structure, J. Eng. Mech. Div. Proc. Am. Soc. Civ. Eng. 90 (EM2) (1964) 131–166, http://dx.doi.org/10.1061/JMCEA3.0000463.

[49] J.L. Beck, J.S. Pei, Demonstrating the power of extended masing models for hysteresis through model equivalencies and numerical investigation, Nonlinear Dynam. 108 (2022) 827–856, http://dx.doi.org/10.1007/s11071-022-07237-5.

[50] L.F. Shampine, I. Gladwell, R.W. Brankin, Reliable solution of special event location problems for ODEs, ACM Trans. Math. Software 17 (1) (1991) 11–25.

[51] L. Ljung, System Identification: Theory for the User, second ed., Prentice Hall, 1999.

[52] M. Schoukens, Improved initialization of state-space artificial neural networks, in: 2021 European Control Conference, ECC, 2021, pp. 1913–1918, http://dx.doi.org/10.23919/ECC54610.2021.9655207.

[53] K. Krikelis, K. van Berkel, M. Schoukens, Artificial neural network hysteresis operators for the identification of Hammerstein hysteretic systems, IFAC-PapersOnLine 54 (7) (2021) 702–707, http://dx.doi.org/10.1016/j.ifacol.2021.08.443, 19th IFAC Symposium on System Identification SYSID 2021.

[54] The Mathworks, narxnet. mathworks.com/help/deeplearning/ref/narxnet.html. (Accessed 7 February 2022).

[55] R. Di Filippo, Active damping for charge driven piezoelectric nanopositioning stage using self sensing control, 2018.

[56] J. Schoukens, L. Ljung, Nonlinear system identification: A user-oriented road map, IEEE Control Syst. 39 (6) (2019) 28–99.