Mechanical Systems and Signal Processing 181 (2022) 109510

Contents lists available at ScienceDirect

Mechanical Systems and Signal Processing

journal homepage: www.elsevier.com/locate/ymssp

On the dynamic properties of statistically-independent nonlinear normal modes

M.D. Champneys [a,b,*], G. Tsialiamanis [b], T.J. Rogers [b], N. Dervilis [b], K. Worden [b]

[a] Industrial Doctorate Centre in Machining Science, Advanced Manufacturing Research Centre with Boeing, University of Sheffield, S60 5TZ, UK
[b] Dynamics Research Group, Department of Mechanical Engineering, University of Sheffield, Mappin Street, Sheffield, S1 3JD, UK

ARTICLE INFO

Keywords:
Nonlinear normal modes
Nonlinear system identification
Machine learning
cycle-GAN
Higher-order frequency-response functions

ABSTRACT

Much attention has been given to the study of *nonlinear normal modes* (NNMs), a nonlinear extension to the eminently useful framework for the analysis of linear dynamics provided by linear modal analysis (LMA). In the literature, several approaches have gained traction, with each able to preserve a subset of the useful properties of LMA. A recently-proposed framework (Worden and Green, 2017) casts nonlinear modal analysis as a problem in machine learning, viewing the NNM as directions in a latent modal coordinate space within which the modal dynamics are statistically uncorrelated. Thus far, the performance of this framework has been measured in a largely qualitative way. This paper presents, for the first time, an exploration into the underlying dynamics of the statistically-independent NNMs using techniques from nonlinear system identification (NLSI) and higher-order frequency-response functions (HFRFs).

In this work, the statistically-uncorrelated NNMs are found for two simulated nonlinear cubic-stiffness systems using a recently-proposed neural-network based approach. NLSI models are fitted to both physical and modal displacements and the HFRFs of these models are compared to theoretical values. In particular, it is found for both systems that the modal decompositions here permit an independent single-input single-output (SISO) representation that can be projected back onto the original displacements with low error. It is also shown via the HFRFs that the underlying linear natural frequencies of the modal dynamics lie very close to the underlying linear natural frequencies of the nonlinear systems, indicating that a true nonlinear decomposition has been identified.

1. Introduction

Linear modal analysis (LMA) is a mainstay in the practical analysis of linear dynamic systems. The framework has a number of desirable properties that permit analysis of linear dynamics over a wide array of applications. The principal concept of linear modal analysis is the specification of invariant, or *modal*, properties that are sufficient to entirely describe the underlying linear equations of motion (EOM). To the dynamicist, these invariants are familiar as the mode shapes, natural frequencies, damping ratios and frequency response functions. The specification of these quantities has been attempted in myriad ways, using approaches ranging from direct derivation from the EOM [1], to sophisticated statistical methods that glean the parameters from measured data [2]. Once specified, these parameters give rise to the other great advantage of modal analysis, that of linear *decoupling*. Given proportional (or Raleigh type) damping, the dynamics of a generic multi-input–multi-output (MIMO) linear system can be exactly

* Corresponding author at: Dynamics Research Group, Department of Mechanical Engineering, University of Sheffield, Mappin Street, Sheffield, S1 3JD, UK.
E-mail address: mdchampneys1@sheffield.ac.uk (M.D. Champneys).

https://doi.org/10.1016/j.ymssp.2022.109510
Received 19 January 2022; Received in revised form 18 May 2022; Accepted 27 June 2022
Available online 5 July 2022

expressed as a linear combination of single-input–single-output (SISO) oscillators and visa-versa. This modal decomposition can expressed as,

$$\mathbf{y} = \Psi\mathbf{u}, \quad \mathbf{u} = \Psi^{-1}\mathbf{y} \tag{1}$$

where $\mathbf{y}$ are the physical displacements, $\mathbf{u}$ are the modal displacements and Ψ is the modal matrix. This decoupling permits a greatly-simplified analysis, as each mode can be analysed and tested against prospective inputs individually, without concern about interaction with other modes. The overall response of the system can be found as a simple linear sum of such analyses—this is the principal of *superposition*.

The presence of nonlinearity in structural dynamics is inescapable in the modern engineering landscape. As the demands of engineering structures become ever more complex, composite materials, demanding geometries and loading conditions outside linear-elastic regions are increasingly part of cutting-edge design. Regrettably, these conditions can all lead to nonlinearity.

Nonlinearity breaks linear modal analysis in more than one place. The invariant modal properties cease to be invariant and often come to depend on the excitation energy of the system. Furthermore, a linear decoupling into SISO oscillators is no longer possible. Superposition is generally no longer available.

It is without wonder that much attention has been given to the extension of modal analysis to the nonlinear case. Although unproven, it is difficult to imagine that every positive attribute of modal analysis can be preserved in the nonlinear case and so a pragmatic approach is required. In [3], it is argued that the engineer can view a nonlinear normal mode in one of two ways.

1. A (local or global) coherent motion of a structure; this generalises the idea of the linear modeshape.
2. A decomposition into lower-order dynamical systems, the motions of which correspond to the ‘modes’; this generalises the idea of a modal decomposition.

Approaches corresponding to both viewpoints can be found in the literature. The first, from Rosenberg [4], generalises the idea of the linear modeshape by allowing an arbitrary function to describe the displacements of the system given a single displacement. The Rosenberg ansatz is therefore,

$$y_i = f_i(y_1) \quad \forall i = 2, \dots, D \tag{2}$$

where D is the number of degrees-of-freedom. In practice, any displacement can be selected for the ‘input’ displacement, but y_1 is used here without loss of generality. The Rosenberg framework has desirable properties – including reduction to the linear case in the absence of nonlinearity – but also comes with some significant drawbacks. Neither non-conservative forces nor internal resonances are natively supported, although there have been contributions to address these shortcomings [5]. For a more detailed description of the method, the reader is directed to the Ref. [6].

A geometrically more general approach was later proposed by Shaw and Pierre in [7]. In the extended framework, the idea of a coherent motion is extended to the phase-space of the nonlinear system, where the displacements and *velocities* of the system response are permitted to depend functionally on a single displacement–velocity pair. The ansatz is now,

$$\begin{aligned} y_i &= f_i(y_1, \dot{y}_1) \\ \dot{y}_i &= h_i(y_1, \dot{y}_1) \end{aligned} \tag{3}$$

This approach retains the properties of the Rosenberg ansatz, but extends the approach naturally to non-conservative forces. The fundamentally linear notion of a planar modeshape in the configuration space has been generalised to the phase space and is now an *invariant manifold*. These curved manifolds in the phase space retain many of the desirable properties of a nonlinear modal analysis. If motion is initiated on such a manifold, then it remains on that manifold for all time; this is akin to the idea that no linear mode can excite any others. Secondly, the manifolds are shown by Shaw and Pierre to be tangent to the underlying linear modes (whereby all nonlinear terms in the EOM are deleted) at the equilibrium point. This ensures that the Shaw–Pierre NNM is equivalent to the linear mode in the limit of zero nonlinearity. This approach has generated a great deal of interest and development. For example, in [8] a number of uniqueness and existence results are presented. The Shaw–Pierre framework has also been extensively used in the treatment of geometric nonlinearities, the interested reader is directed to [9] for a recent survey.

Modal decomposition into lower order systems, the second viewpoint of [3], leads to a somewhat more practical view of NNMs. Here, the ethos is to retain the properties of linear modal analysis that facilitate simpler analyses. Many techniques are applicable to this problem, including nonlinear embedding methods [10], polynomial decoupling methods [11] and statistical decoupling approaches such as proper orthogonal decomposition [12]. It is argued here that the metrics of success for NNMs with these motivations are,

1. The ability of the decomposition to render the dynamics into independent modal dynamics (preferably SISO) wherein the dynamics are simplified.
2. The extent to which the original dynamics can be recovered from the decomposition (nonlinear superposition).

Ref. [3], proposes a third class of NNM, based on a nonlinear decomposition into uncorrelated time series. The motivation for this NNM is that often during analysis, the engineer does not have access to the underlying EOM of their system. An output-only approach is also desirable.

The fundamental idea behind this NNM can be stated simply. By generalising the linear idea of decomposition, the framework specifies a static map f from the physical displacements to a new coordinate system within which the *nonlinear modal* displacements

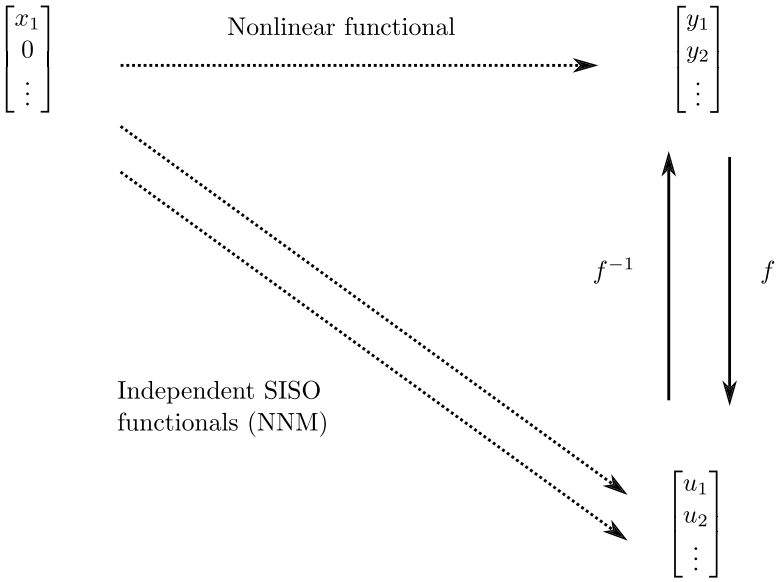

Fig. 1. Overview of the novel NNM approach.

are uncorrelated time series. As an additional *inductive bias*, the framework seeks to retain the idea of orthogonality that is present in linear modal analysis (the mode shapes are an orthogonal basis), by requiring that the forward and inverse transformations are *conformal* and locally preserve orthogonality. A conformal map $f : X \to Y$ locally preserves angles between vectors but not necessarily lengths. Thus, for two vectors originating from a single point $\mathbf{x_i}, \mathbf{x_j} \in X$. From the definition of the dot product, f is conformal if,

$$\frac{\mathbf{x}_i \cdot \mathbf{x}_j}{\|\mathbf{x}_i\|\|\mathbf{x}_j\|} = \frac{f(\mathbf{x}_i) \cdot f(\mathbf{x}_j)}{\|f(\mathbf{x}_i)\|\|f(\mathbf{x}_j)\|} \quad \forall \mathbf{x}_i, \mathbf{x}_j \in X \tag{4}$$

Fig. 1 depicts the statistical independence approach graphically. In the figure, dotted lines are functionals and solid lines represent static maps. As can be seen, the framework also defines the inverse map f^{-1}, permitting an approximate nonlinear superposition. Following a machine-learning approach, f is learnt from data via an objective function which penalises the pairwise correlations of the modal displacements and encourages orthogonality of the transformation.

Although this approach may at first seem to bear close resemblance to the Rosenberg framework, there are structural differences. In the Statistically independent framework, every physical displacement is used to build a map onto each modal displacement u_i. In this fashion there is no concept of a 'driving DOF' as seen the in the NNM frameworks of Rosenberg and Shaw–Pierre. The new ansatz is,

$$u_i = f_i(y_1, y_2, \dots, y_D) \quad \forall i = 1, \dots, D \tag{5}$$

It is certainly interesting to imagine if any equivalence could be established linking the statistically-independent NNMs to previous approaches. However, the authors do not claim any such equivalence at this time.

In terms of language, the authors believe that any decomposition from the full nonlinear dynamics onto a simplified (by some definition) set of coordinates that, in the limit of linearity, reduces correctly to the linear modes can rightly be called an NNM.

The original work [3], adopts a truncated multinomial expansion for f and a Gaussian process (GP) form for f^{-1}. Orthogonality is introduced by a term in the objective function that encourages orthogonality between rows in the parameter matrices of the multinomial. However, this approach is shown in [13], to lead to bloated models that give poor performance in the heuristic optimisation methods used to tune the parameters of the multinomial. Despite this, the polynomial approach showed good performance on a range of nonlinear single-input multiple-output (SIMO) systems comprising both simulated and experimental datasets.

A recent work adopts an alternative *neural* architecture for f and f^{-1} [14]. In this work, a cycle-consistent generative adversarial network (cycle-GAN) is used [15]. The cycle-GAN architecture consists of two generator models (feed-forward neural networks) representing f and its inverse. These models are trained *adversarially* against discriminator models that encourage the generators to map to a target distribution. For additional details on adversarial training and GANs, the reader is directed to the seminal paper [16].

There are several advantages to the cycle-GAN approach. Principally, the neural architectures are extremely flexible and can learn complex nonlinear mappings if required. When compared to other neural architectures used for embeddings (such as auto-encoders [17] or their variational extensions [18,19]), there is no need for a bottleneck layer to avoid learning trivial mappings. Retaining the dimensionality of the input ensures that as many modal coordinates are defined as there are degrees-of-freedom in

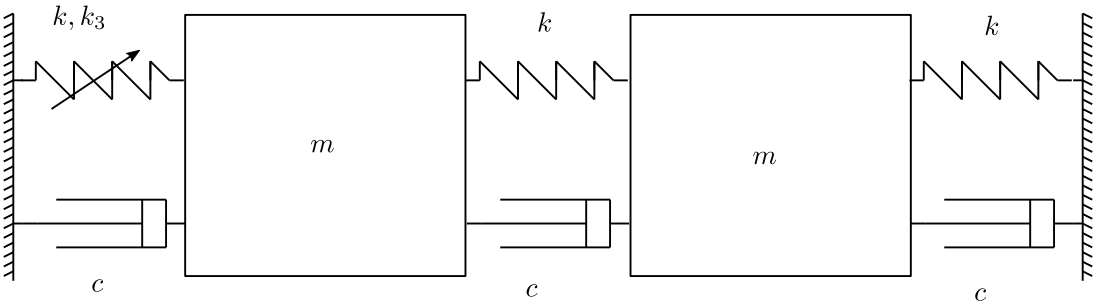

Fig. 2. Schematic of the two-DOF system used in this paper.

the nonlinear structure. The target distributions of the generator models can be chosen freely to ensure independence (in [14], independent Gaussians are used). There is also no need to learn a separate inverse mapping, because the forward and inverse mappings are learnt simultaneously. Additionally, a gradient-descent approach permits good optimisation performance on arbitrary (differentiable) objective functions, even in the presence of large numbers of model parameters. The cycle-GAN NNMs of [14] showed improved performance over the multinomial GP approach in both decomposition and reconstruction tasks.

Thus far, the performance of these NNMs have been judged by a mixture of qualitative and quantitative measures. When first proposed, the quality of the simplification was judged 'by-eye' to be the extent to which the modal displacements produced solitary peaks in their power-spectral densities (PSD). In [13], this was extended to a convolution-based approach, and in [14], to an approach based on the inner-product. The quality of the reconstruction can of course be evaluated by a normalised mean-square error (NMSE) metric.

The methodology of statistically-independent NNMs has already been successfully applied to a number of case studies, both simulated and experimental. However, until now, little attention has been given to the precise nature of the dynamics that are present in the NNMs generated under the statistically-independent framework. This paper seeks to address this issue by utilising techniques from nonlinear system identification (NLSI). The approach here is to model the dynamics of the nonlinear functional $\mathbf{x} \to \mathbf{u}$, and compare them to equivalent models of the physical dynamics $\mathbf{x} \to \mathbf{y}$, to look for more concrete evidence of simplification.

The contributions of this paper are the following: For the first time, the nature of the underlying dynamics in the statistically-uncorrelated NNMs are examined. For two benchmark nonlinear dynamical systems, the dynamics are approximated by discrete-time linear and nonlinear models. The relevant fidelity of these models (and their ability to reconstruct the physical dynamics) is used as justification for the extent to which the NNMs can be described as nonlinear-SISO systems in their own right. Also presented, is a study comparing the performance of the modal transformation in generalising between excitation levels. Finally, the higher-order frequency response functions (HFRFs) of the NLSI models are derived in a vectorised form and computed. The HFRFs form an infinite series and can be used to glean information about polynomial nonlinearities present in the dynamics. Of particular interest is the quantity $\mathbf{H}_1$ which is equivalent to the linear FRF of a system if all nonlinear elements are deleted. This quantity is used to determine the extent to which the NNM are equivalent to their linear counterparts in the absence of nonlinearity. An incidental novel result here, is the derivation of the first three HFRFs of a Gaussian-process nonlinear autoregressive with exogeneous inputs (GP-NARX) model in the case of independent lengthscales. These are furthermore presented in a novel vectorised form, offering a significant computational speedup over previously presented expressions.

The structure of the remainder of this paper is as follows; the second section describes the specification and simulation of the nonlinear benchmark systems used in this paper. The third section covers the nonlinear modal analysis and the construction of the NNMs by the cycle-GAN method. The fourth section introduces the training of NLSI models of the physical and modal displacements. The fifth section describes the derivation and computation of HFRFs for the NLSI models. A final section holds some discussions over the results of this paper and discusses limitations and future avenues for investigation.

2. Benchmark nonlinear dynamical systems

Statistically-independent NNMs have thus far been found for a range of dynamical systems undergoing random excitation. Both numerical and experimental systems have been considered across a range of types of nonlinearities, including polynomial and impact types [3,14].

For the present study, the NNMs of two and three degree-of-freedom (DOF) systems with a static cubic stiffness nonlinearity will be investigated. Fig. 2 gives a schematic view of the two-DOF system under investigation.

In both case studies, nonlinearity is introduced into the system by the addition of a fixed cubic stiffness element between the first degree of freedom and the base. The equations of motion for the two and three DOF systems are therefore given by,

$$M\ddot{\mathbf{y}} + C\dot{\mathbf{y}} + K\mathbf{y} + K_3\mathbf{y}^3 = \mathbf{x}(t) \tag{6}$$

where, M, C, K, K_3 are square parameter matrices corresponding to the nonlinear system depicted in Fig. 2, and have values as given by Table 1.

Table 1
Parameters for the nonlinear benchmark systems.

Parameter	Description	Value
m	Mass (Kg)	1
c	Viscous damping coefficient (Ns/m)	20
k	Linear stiffness (N/m)	1×10^4
k_3	Cubic stiffness (N/m^3)	5×10^9
$[\omega_1, \omega_2]$	Linear natural frequencies (2 DOF) (Hz)	[15.9, 27.5]
$[\omega_1, \omega_2, \omega_3]$	Linear natural frequencies (3 DOF) (Hz)	[12.1, 22.5, 29.4]
f_s	Sampling frequency (Hz)	500
f_c	Lowpass cut-off frequency (Hz)	50
σ_x	Excitation standard deviation (N)	20
N_s	Points sampled	1×10^5

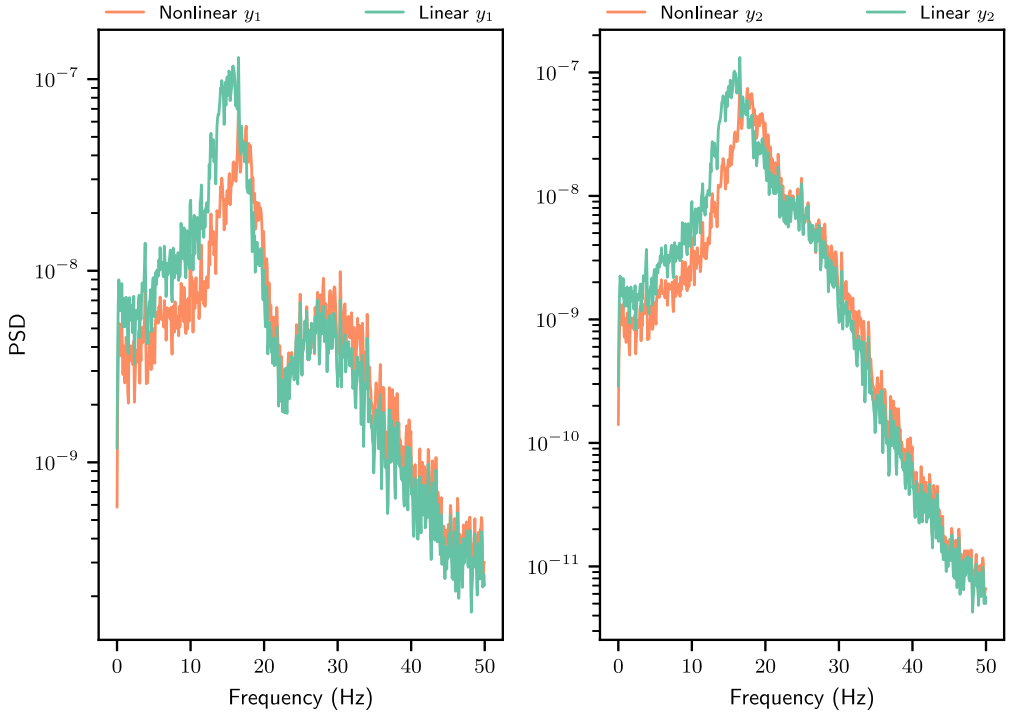

Fig. 3. PSDs (estimated by the Welch method [20]) for the nonlinear and underlying linear system, the nonlinear data shows a clear hardening via shifts in the principal resonance frequencies.

The equations of motion are integrated forward in time by a fixed-step fourth-order Runge–Kutta scheme. In total 10^5 points are generated in order to achieve good estimates of the spectral densities. The input excitations are white Gaussians $x \sim N(0, \sigma_x)$ on the first degree of freedom. An additional low-pass filter is applied to the excitations with a cut-off frequency of 50 Hz. Overall, data at three excitation levels of $\sigma_x = [8, 14, 20]$ are collected for each configuration. For convenience, all parameters pertaining to the numerical simulation are collected in Table 1.

After simulation, the data are divided into training, validation and testing sets, with each set consisting of 10^3 points. In order to remove the effect of transient behaviour introduced by the initial conditions or the numerical solver, the first 10^4 points from the simulation are discarded.

In order to verify that the dynamics are indeed nonlinear, the spectral densities (estimated by the Welch method [20]) of the nonlinear data are compared to the equivalent system with the nonlinear elements removed. Nonlinear cubic hardening is demonstrated for the two degree-of-freedom system and $\sigma_x = 20$ in Fig. 3. In the plot there is clear evidence of hardening of the resonance peaks from the inclusion of the cubic stiffness.

3. Modal analysis

With the data from the benchmark nonlinear systems established, attention can be turned to the construction of the modal transformations. In [3,13], the NNMs are constructed analytically via a polynomial expansion of the input displacements. The parameters in the polynomial expansion were optimised via a meta-heuristic method that solved a multi-objective optimisation problem. This approach was found to be effective for small numbers of degrees of freedom; however, in [13], it was demonstrated that as the number of degrees-of-freedom or terms in the polynomial expansion grow, the optimisation problem quickly becomes computationally impractical without access to the parameter gradients.

In order to address this problem, the recent contribution [14], demonstrates that *cycle-consistent generative adversarial networks* (cycle-GAN) provide better access to the NNMs than an explicit form of the transformation into modal coordinates. The inverse mapping is handled naturally and network parameters can be optimised via gradient descent. Another useful contribution of the work is the proposal of an inner-product metric for evaluating the separation of the modes in the frequency domain; this replaces the 'by-eye' approach used in [3].

Clearly any reasonable framework for NNMs should have as many modal directions as there are physical degrees-of-freedom (even if a reduced-order representation is desirable) and so these architectures become problematic. The cycle-GAN does not have such a limitation. In this work, the cycle-GAN approach is adopted. For a full presentation of the method the reader is directed to the original paper [14], but a brief description of the approach is included here.

The structure of the cycle-GAN model here is essentially two generative adversarial networks (GANs) [16], tasked with learning the forward ($f : Y \to U$) and inverse ($f^{-1} : U \to Y$) mapping to the modal coordinates. Within each GAN, are two neural networks each possessed of a single hidden layer. Of these two networks, the first, – the *generators* (f, f^{-1}) – learn the desired mapping, while a second – the *discriminators* (D_f, $D_{f^{-1}}$) – learn to classify according to the target distributions.

Here, as in [14], the targeted distribution for the modal coordinates is a pair of independent Gaussian distributions. The target distribution for the physical coordinates is learnt from the measured samples. The adversarial training of the GANs ensures that the generator models learn the required mappings, while respecting the target distributions. Adversarial training leads to a number of loss functions for the cycle-GAN that are optimised simultaneously by a stochastic gradient-descent algorithm. The first are the adversarial losses, computed as in the original cycle-GAN paper [15],

$$\begin{aligned}\mathcal{L}_1^{\mathbf{y}} &= \mathbb{E}[\log D_f(\mathbf{y})] + \mathbb{E}[\log(1 - D_f(f(\mathbf{u})))]\\ \mathcal{L}_1^{\mathbf{u}} &= \mathbb{E}[\log D_{f^{-1}}(\mathbf{u})] + \mathbb{E}[\log(1 - D_{f^{-1}}(f^{-1}(\mathbf{u})))]\end{aligned} \tag{7}$$

where $\mathbb{E}$ is the expectation operator, f (resp. f^{-1}) are the forward and inverse modal transformations and D_f (resp. $D_{f^{-1}}$) are the discriminator models trained to classify the target modal and physical distributions. The next objective function is the reconstruction loss,

$$\begin{aligned}\mathcal{L}_2^{\mathbf{y}} &= \|\mathbf{y} - f(f^{-1}(\mathbf{y}))\|_2\\ \mathcal{L}_2^{\mathbf{u}} &= \|\mathbf{u}^* - f^{-1}(f(\mathbf{u}^*))\|_2\end{aligned} \tag{8}$$

where the $\mathbf{u}^*$ are samples from independent Gaussian distributions. Finally, the inductive bias of orthogonality is introduced by a third loss term $\mathcal{L}_3$, that ensures the conformality of the inverse modal transformation $\mathbf{u} \to \mathbf{y}$. It is only required to place this restriction on one of either f of f^{-1} since the conformal property necessarily applies to inverses. In the interest of brevity, the algorithm used to compute $\mathcal{L}_3$ is not included here. For a detailed explanation of the orthogonality enforcement assembly, the interested reader is directed to [14]. The networks are trained over a number of epochs consisting of 2048 samples per epoch using an ADAM [21], stochastic gradient-descent algorithm. The loss from a given training epoch is given by,

$$\mathcal{L} = \lambda_1(\mathcal{L}_1^{\mathbf{y}} + \mathcal{L}_1^{\mathbf{u}}) + \lambda_2(\mathcal{L}_1^{\mathbf{y}} + \mathcal{L}_1^{\mathbf{u}}) + \lambda_3\mathcal{L}_3 \tag{9}$$

where the λ_i are weights in the objective function. Following the approach of [14] and here, values of $\lambda_1 = 1$, $\lambda_2 = 10$ and $\lambda_3 = 1$ are used.

During each training epoch, gradient descent is performed on the cycle-GAN in both forward and inverse passes. Each pass consists of feedforward and backpropagation steps. In the forward configuration, the observed physical displacements y are mapped onto their modal counterparts via the generator of f and back onto the reconstructed physical displacements $\hat{y}$ by the generator of f^{-1}. Gradient is then backpropagated through all models according to the ADAM optimiser. In the inverse pass, the model is inverted and draws from a Gaussian distribution are mapped onto samples from a distribution learnt from the physical displacements by the inverse mapping f^{-1} and then back onto the random samples by f. A graphical depiction of the forward pass through the model is depicted in Fig. 4.

Following [14], in order to remove any dominant linear correlations, the input displacements are first transformed by a PCA decomposition. Transforming the data in this way frees the neural architecture to learn any nonlinearity that might be required for f without having to learn all the linear transformations as well. The same structure is of course present in the inverse mapping.

The network structures and parameters are detailed in Table 2. The number of hidden nodes in both the forward and inverse mappings are set to the same value and training is repeated over a range of values to ensure the best possible chance of learning a good decomposition and to ensure validation of the model architecture. For this paper, values between 10 and 200 are considered with an increment of 10, with 10 training repeats per increment.

All networks, across all repeats, are trained for 5000 epochs each consisting of 2048 training examples. Once trained, an inner-product metric is used to select the best encoder and decoder pair. This metric ensures that the transformation gives good separation

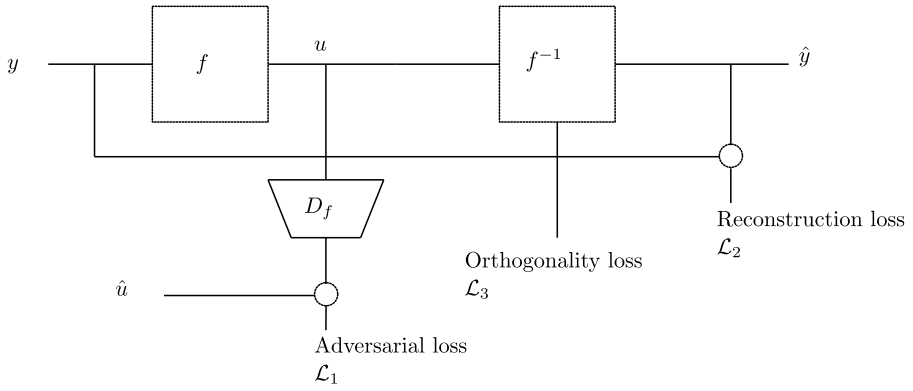

Fig. 4. Overview of the cycle-GAN structure for the forward pass.

Table 2
Network parameters for the cycle-GAN network.

Parameter	Description	Value
n_r	Training repeats	10
n_e	Training epochs	5000
$n_s e$	Training examples per epoch	2048
n_l	Number of hidden layers in f and f^{-1}	1
h_l	Min. hidden nodes in f and f^{-1}	10
h_m	Max. hidden nodes in f and f^{-1}	200
n_s	Hidden node step size	10
W_{hann}	Samples per Welch segment	2048
W_o	Welch segment overlap	0

of peaks in the frequency domain. To simplify the computational complexity of the approach, the inner-product score is computed every 100 training epochs. The metric is defined over the power-spectral densities (PSD) of modal coordinates $P\mathbf{u}$ and is given by,

$$\mathcal{L}_{\text{inner}} = \sum_{i=1,j=i+1}^{n_{\text{dof}}} \frac{Pu_i \cdot Pu_j}{\|Pu_i\| \, \|Pu_j\|} \tag{10}$$

In the above, the Pu_i and Pu_j are estimated by a Welch method [20]. This choice leads to the introduction of some additional hyperparameters arising from the window length and overlap size in the Welch method. The authors' experience with the cycle-GAN approach has shown that too much noise in the PSD can lead to poor performance and so values of the parameters that promote smoothness are selected. In the present study, a Hamming window of length 2^{11} samples is used with zero overlap.

The PSDs of the modal coordinates are compared to those of the physical coordinates for the two-DOF and three-DOF systems in Figs. 5 and 6. When judged visually, the peaks in the PSDs are well separated, indicating a good decomposition. Whilst this qualitative analysis is promising, the remaining sections of this present a more rigorous examination of the decomposition using NLSI.

4. Nonlinear system identification

Now that the $\mathbf{y}$ and $\mathbf{u}$ are established, attention can be turned towards analysis of the underlying dynamics. Following linear modal analysis, it is assumed that the $\mathbf{u}$ are the outputs of some *modal* dynamic system $\mathbf{x} \to \mathbf{u}$ that represents a decoupled formulation of the physical dynamics. Unlike previous work that relied on qualitative metrics of decomposition, this work uses NSLI to model the modal dynamics themselves. The approach in this work will be to conduct NLSI using both linear and nonlinear models. For the linear modelling, auto-regressive models with exogenous inputs (ARX) will be used. ARX are a discrete time formulation where the model form is,

$$y_t = \sum_{i=1}^{n_y} a_i y_{t-i} + \sum_{i=0}^{n_x-1} b_i x_{t-i} \tag{11}$$

where n_x and n_y refer to the number of lagged inputs and outputs respectively and a and b are vectors of model parameters. For a given lag structure, ARX models permit a vectorised formulation,

$$\mathbf{y} = H \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix} \tag{12}$$

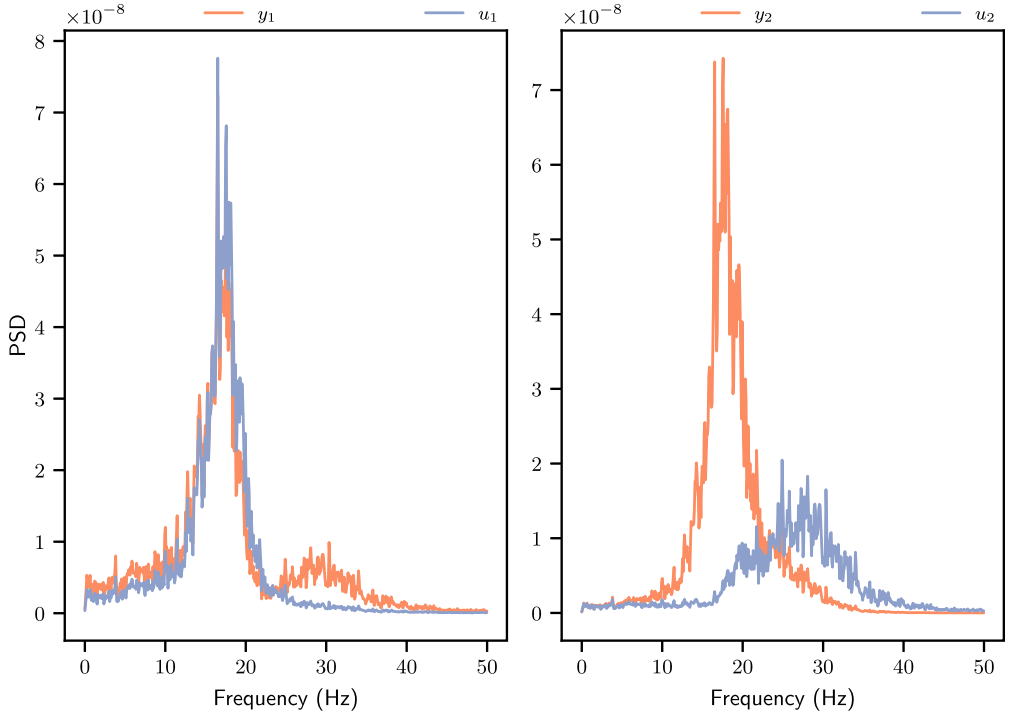

Fig. 5. PSD of physical and modal displacements: two-DOF system.

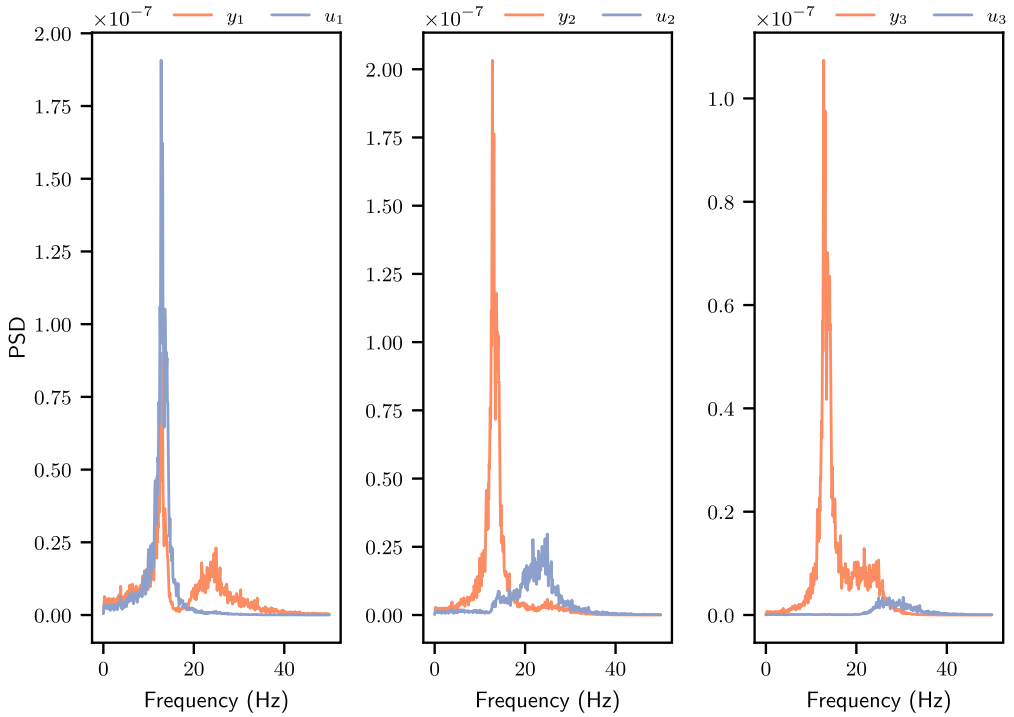

Fig. 6. PSD of physical and modal displacements: three-DOF system.

where H is the Hankel matrix of lagged inputs and outputs at time t given by,

$$H = \begin{bmatrix} y_{t-1} & y_{t-2} & \cdots & y_{t-n_y} & x_t & x_{t-1} & x_{t-2} & \cdots & x_{t-n_x+1} \\ y_{t-2} & y_{t-3} & \cdots & y_{t-n_y-1} & x_{t-1} & x_{t-2} & x_{t-3} & \cdots & x_{t-n_x} \\ y_{t-3} & y_{t-4} & \cdots & y_{t-n_y-2} & x_{t-2} & x_{t-3} & x_{t-4} & \cdots & x_{t-n_x-1} \\ \vdots & & & & \vdots & & & & \\ y_{t-N+p} & & & & x_{t-N+p-1} & & & & \end{bmatrix} \tag{13}$$

where N is the total number of temporal points in the training data. ARX models are completely defined by the model weights a_i and b_i which can be simply fitted to input data by least-squares solution of the linear equation in (12).

For the nonlinear models, Gaussian process NARX models will be used. The model formulation is that of a generic nonlinear auto-regressive model with exogenous inputs (NARX) as,

$$y_t = F(y_{t-1}, y_{t-2}, \dots, y_{t-n_y}, x_t, x_{t-1}, x_{t-2}, \dots, x_{t-n_x+1}) \tag{14}$$

where F is a static nonlinear map from the Hankel matrix to the predictions. As before, NARX models can be expressed in terms of the Hankel matrix,

$$\mathbf{y} = F(H) \tag{15}$$

For the GP-NARX formulation it is assumed that F can be approximated by a Gaussian process. This formulation comes with a number of advantages over more traditional approaches, such as a polynomial expansion. Uncertainty quantification is handled naturally, and the non-parametric form permits a high degree of model flexibility. A full description of Gaussian processes and the GP-NARX model is beyond the scope of this paper and so only a brief overview is included here; for excellent references on the topics, the reader is directed to [22–24]. A Gaussian process (GP) is a essentially a regression over a space of functions. For a multi-input single-output (MISO) model the core of the GP is the regression formulation,

$$y_i = f(\mathbf{x}) + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0, \sigma_n^2) \tag{16}$$

where f is a *latent function* that has not been observed directly, but can only be accessed via the observations y. The GP is formed by assuming a distribution over possible values of the latent function as,

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x})) \tag{17}$$

where k is a positive semi-definite covariance kernel $k(\mathbf{x}, \cdot) \to \mathbb{R}^+$ and $m(\mathbf{x}) \to \mathbb{R}$ is the mean function. Several choices for both of these functions are applicable and their selection is often driven by domain knowledge [22]. In the absence of any prior knowledge, the zero-mean $m(\mathbf{x}) = \mathbf{0}$ function is the safest assumption and ensures that the GP function is entirely shaped by the data.

The GP formulation allows one to marginalise over all possible latent functions f. The result is a multivariate normal distribution over some observed data $\{X, \mathbf{y}\}$ and unobserved testing points $\{X^*, \mathbf{y}^*\}$.

$$\begin{pmatrix} \mathbf{y} \\ \mathbf{y}^* \end{pmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} K(X,X) + \sigma_n^2 I & K(X, X^*) \\ K(X^*, X) & K(X^*, X^*) + \sigma_n^2 I \end{bmatrix}\right) \tag{18}$$

where $K(X, \cdot) \to \mathbb{R}^{n+}$ is a vectorised formulation of the covariance kernel for n training inputs. From this distribution, it is straightforward to calculate the posterior predictive distribution for unobserved variables $\mathbf{y}^*$ as,

$$\mathbf{y}^* \sim \mathcal{N}(\mathbb{E}[\mathbf{y}*], \mathbb{V}[\mathbf{y}*]) \tag{19}$$

where the posterior mean prediction is given by,

$$\mathbb{E}[\mathbf{y}*] = K\left(X^*, X\right)\left[K(X,X) + \sigma_n^2 I\right]^{-1}\mathbf{y} \tag{20}$$

and the posterior variance is given by,

$$\mathbb{V}[\mathbf{y}*] = K\left(X^*, X^*\right) - K\left(X^*, X\right)\left[K(X,X) + \sigma_n^2 I\right]^{-1} K\left(X, X^*\right) + \sigma_n^2 \tag{21}$$

Substituting the zero-mean function and the posterior variance into the formulation of a NARX model in (15) and taking the mean prediction of the posterior distribution (equivalent to a maximum *a posteriori* (MAP) evaluation) gives the *prediction* or *one step ahead* (OSA) prediction from the GP-NARX on some unseen lagged inputs H^* as,

$$\mathbf{y}^*{}_{\text{OSA}} = \mathbb{E}[y*] = K\left(H^*, H\right)\left[K(H,H) + \sigma_n^2 I\right]^{-1}\mathbf{y} \tag{22}$$

where H is the Hankel matrix of training data as before and H^* is the Hankel matrix of inputs for the prediction points. A more rigorous test of the performance of the model is the *simulation* or *model-predicted output* (MPO). To obtain this type of prediction, the model predictions are fed back into the model as inputs to the next time step. The predictions from an MPO run treat the discrete-time model as a differential equation in its own right subject to an initial value problem (IVP). Such predictions give a more accurate picture of the dynamics that have been learned during training. MPO simulation requires new predictions for each time step and so the vectorised form of the equations above cannot be used. The prediction at time t for MPO prediction is given by,

$$y_t^* = k\left(\mathbf{h}_t^*, H\right)\left[K(H,H) + \sigma_n^2 I\right]^{-1}\mathbf{y} \tag{23}$$

Table 3
Parameters used in the optimisation of the GP-NARX models.

Parameter	Description	Value
S_p	Population size	200
S_g	Number of generations	200
n_{train}	Training points	1000
n_{val}	Validation points	1000
n_{test}	Testing points	1000
σ^2_{RMS}	Regularisation noise (MPO training only)	1% RMS

where $\mathbf{h}^*_t$ are the lagged inputs and lagged predicted outputs from the current time step,

$$\mathbf{h}^*_t = \left[\hat{y}^*_{t-1} \quad \hat{y}^*_{t-2} \quad \cdots \quad \hat{y}^*_{t-n_y} \quad x_t \quad x_{t-1} \quad x_{t-2} \quad \cdots \quad x_{t-n_x+1}\right] \tag{24}$$

where the $\hat{y}^*$ are model predictions from previous time steps that are fed back into the model. MPO prediction raises some difficulties in the context of GP-NARX models. The major issue is that, by definition, the GP is not permitted to have uncertainty on the values of the inputs. This limitation makes it impossible to fully incorporate uncertainty quantification during MPO prediction, and in practice, leads to underestimates of the predicted variance. One approach to circumvent this issue is to use a Monte Carlo-based sampling technique [23,24]. However because uncertainty quantification is not the principal aim of the current investigation this has not been considered here.

Practically, there are still a small number of hyperparameters that must be addressed when fitting GP-NARX models. The first is the specification of the covariance function. A common choice for physical processes is the squared-exponential (SE) kernel. Although there are several potential choices for the covariance function, the SE kernel is used here. The SE kernel is known to produce smooth (class $\mathbf{C}^\infty$) functions [22], and so is an appropriate choice for NLSI where the nonlinearities are expected to be continuous. The SE kernel is given by,

$$k\left(x, x'\right) = \sigma_f^2 \exp\left(-\frac{1}{2\boldsymbol{\ell}^2}\left\|x - x'\right\|^2\right) \tag{25}$$

where σ_f^2 is the signal variance and $\boldsymbol{\ell}$ is a $1 \times (n_x + n_y)$ vector of length scales that allow for differently scaled dynamics on each input dimension. Utilising a vector of length scales rather than a single hyperparameter allows for greater model flexibility and is referred to as automatic-relevance detection (ARD) in the GP literature [22]. These parameters, along with the noise variance σ_n^2 and lag structure, are hyperparameters of the GP-NARX model and must be optimised to ensure a good model.

The training approach for GP-NARX models trained in this paper is the following. Firstly, a training, validation and testing set of data are taken from the simulated and modal data, each comprising 10^3 points. In the authors experience with NLSI models datasets of this size represent a good trade-off between generalisation performance and computational-complexity. The next step is to fix the lag structure. This is achieved by a grid search over the maximum number of lags in the input n_x and output n_y respectively. A full combinatorial lag selection is not deemed necessary as it is assumed that the effect of any unnecessary lags will be removed by the model weights in the ARX models or the lengthscales in the GP-NARX models. For each point in the grid, ARX and GP-NARX models are trained on the training data and the remaining hyperparameters $(\sigma_n^2, \sigma_f^2, \boldsymbol{\ell})$ are optimised. In order to alleviate some of the computational difficulties associated with training GP-NARX models in such a brute-force approach, the optimisation of the hyperparameters during lag structure optimisation (LSO) is completed using an evidence framework for which the objective function is given by the negative log of the marginal likelihood,

$$J_{\text{LSO}}(\sigma_n^2, \sigma_f^2, \boldsymbol{\ell}) = -\frac{1}{2}\mathbf{y}^T\left[K(X,X) + \sigma_n^2 I\right]\mathbf{y} - \frac{1}{2}\log\left|K(X,X) + \sigma_n^2 I\right| \tag{26}$$

With ARX and GP-NARX models trained at each point on the grid, predictions are then generated for the validation data using the MPO approach described above. The lag structure is chosen to be the one that produces the lowest normalised mean-square error (NMSE) on the validation data. The NMSE metric is defined by,

$$J_{\text{NMSE}} = \frac{100}{\sigma_y^2 N_J}\sum_i^{N_J}\left(y_i - \hat{y}_i\right)^2 \tag{27}$$

where $N_J = N_p - (n_x + n_y)$. A visualisation of the resultant cost surfaces is given for the two-DOF system in Fig. 7. As can be seen in the figures, the cost surfaces show a steep gradient for low integer numbers of lags but beyond this threshold there is a noisy plateau indicating that an appropriate number of lags have been considered. With the lag structure set, the models are re-trained using an NMSE loss function on the MPO predictions; this is done to encourage the models to learn the underlying dynamics of the data.

All GP-NARX hyperparameter optimisations are completed using a particle swarm algorithm [25]. The optimisation and training parameters are collected in Table 3.

For each SISO functional mapping $x_i \to y_i$ and $x_i \to u_i$, both ARX and GP-NARX models are fitted using the approach described above.

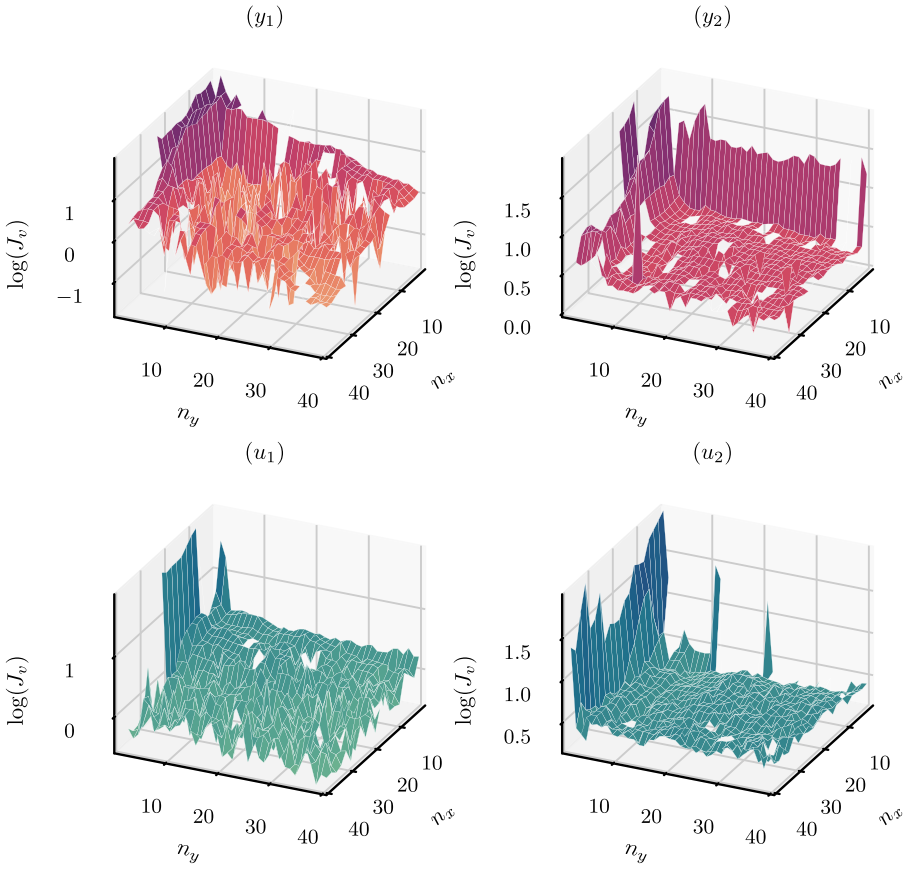

Fig. 7. GP-NARX NMSE surface during LSO training. NMSE scores exceeding 100 are excluded from the plot.

Table 4
Lag structures and MPO NMSE scores for the trained models on the unseen testing data for the two-DOF system.

Model	Target	n_x	n_y	MPO NMSE
ARX	y_1	11	13	4.08
	y_2	3	14	4.95
	u_1	7	9	4.25
	u_2	15	13	3.62
GP-NARX	y_1	17	22	0.0502
	y_2	21	34	0.234
	u_1	39	38	0.154
	u_2	1	26	3.16

For the two-DOF system, the MPO predictions on the unseen training data are depicted in Figs. 8 and 9 for the physical and modal displacements respectively. MPO NMSE scores and lag structures are collected in Table 4. Overall, the GP-NARX models are outperforming the linear ARX models in every case except u_2, where performance is similar.

As can be seen from the results, the quality of fit is good for the ARX models (NMSE generally less than 5%) and excellent for the GP-NARX models (NMSE generally less than 1%). It is interesting to note the wide range of lag structures that are present in the results. This can likely be attributed to the difficulty of the optimisation problem leading to very rough LSO surfaces like the one seen in Fig. 7. It is likely that there are a number of 'good' choices for lag structure in the NLSI models and so it is imagined that prediction accuracy on the unseen testing data is fairly insensitive to the actual values of n_x and n_y beyond a certain level.

For the three-DOF system, the model predictions on y and u are shown in Figs. 10 and 11 respectively. Lag structures and NMSE scores are collected in Table 5.

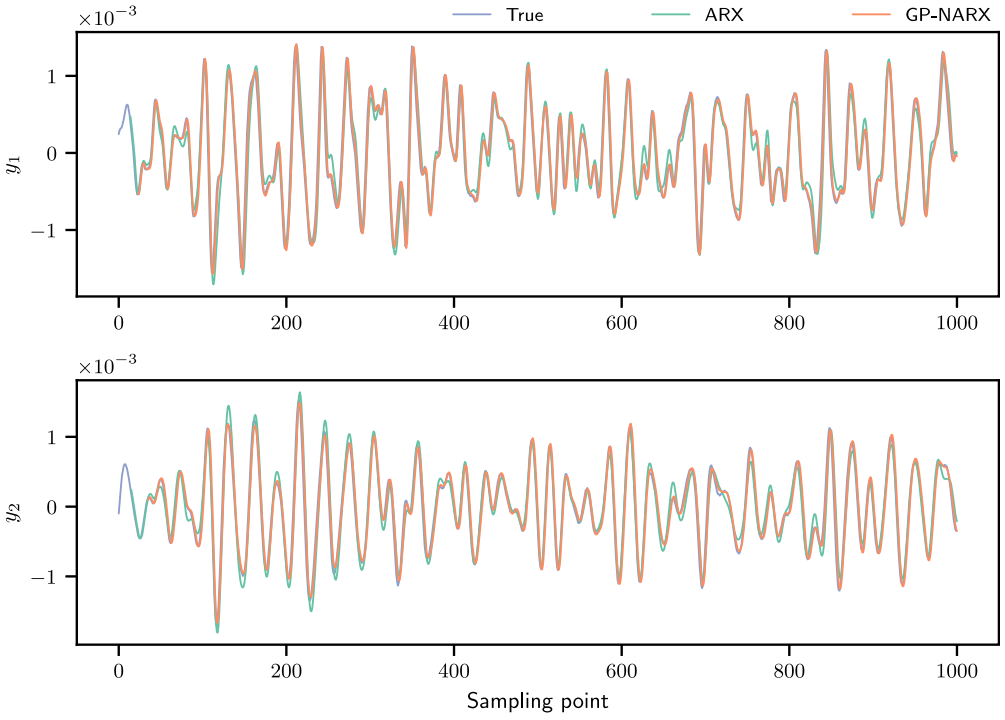

Fig. 8. MPO predictions on the unseen testing data for the NLSO models of the functional $\mathbf{x} \to \mathbf{y}$ for the two-DOF system.

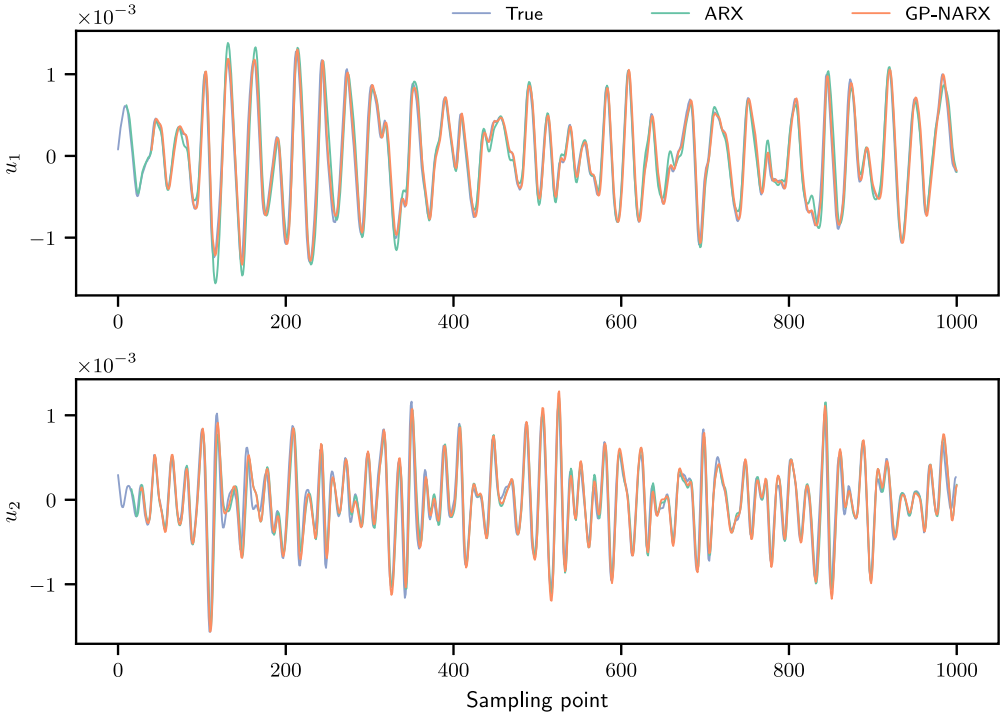

Fig. 9. MPO predictions on the unseen testing data for the NLSO models of the functional $\mathbf{x} \to \mathbf{u}$ for the two-DOF system.

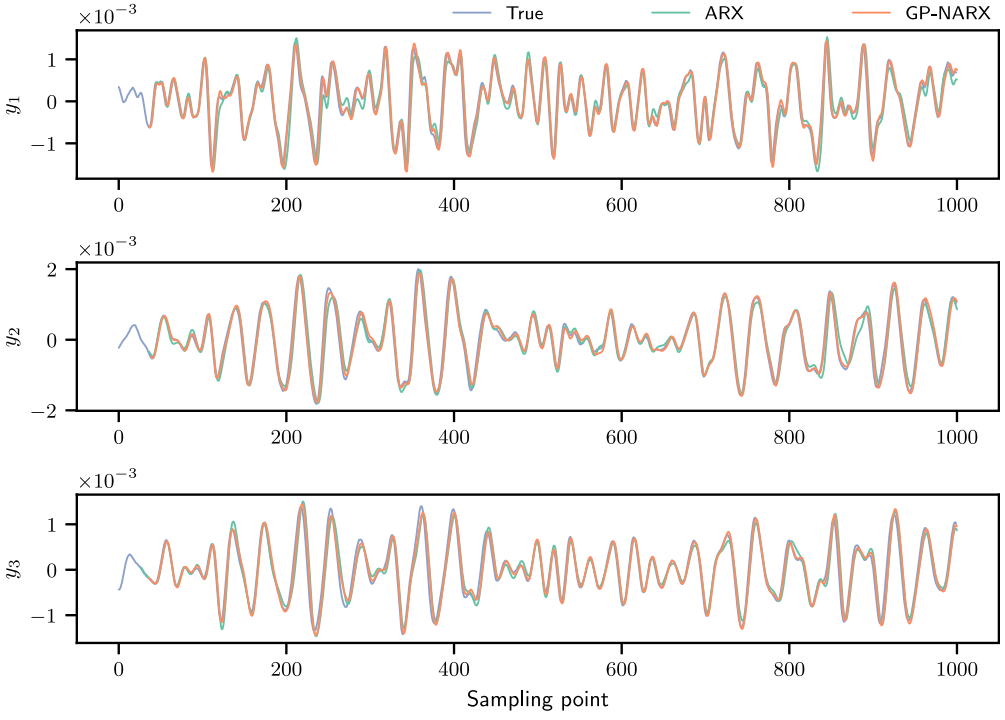

Fig. 10. MPO predictions on the unseen testing data for the NLSO models of the functional $\mathbf{x} \to \mathbf{y}$ for the three-DOF system.

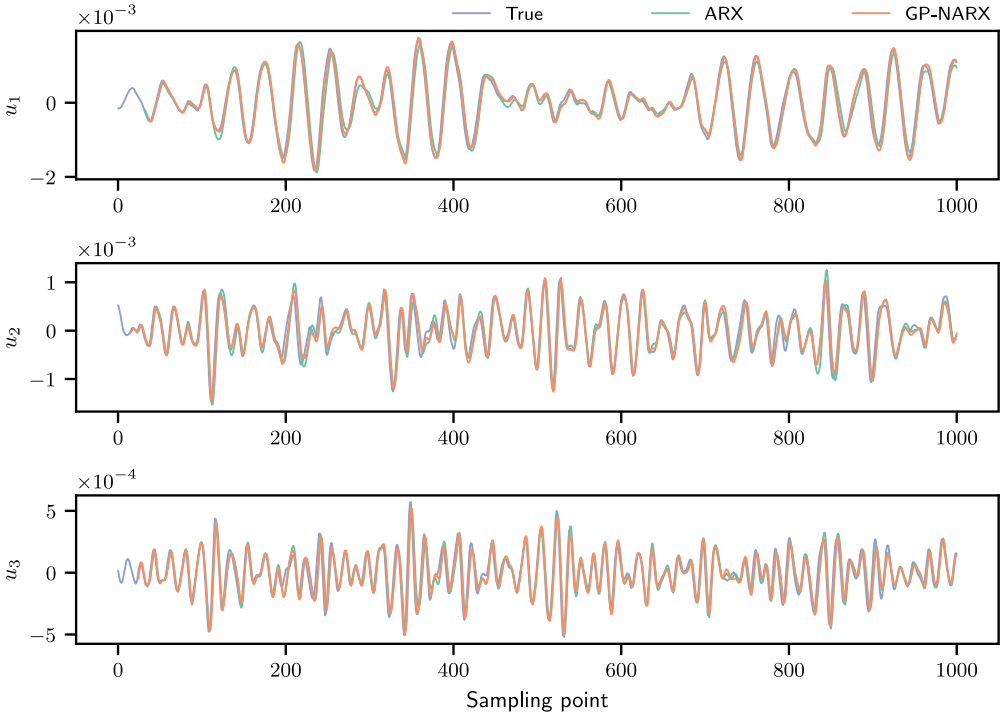

Fig. 11. MPO predictions on the unseen testing data for the NLSO models of the functional $\mathbf{x} \to \mathbf{u}$ for the three-DOF system.

Table 5
Lag structures and MPO NMSE scores for the trained models on the unseen testing data for the three-DOF system.

Model	Target	n_x	n_y	MPO NMSE
ARX	y_1	37	39	4.52
	y_2	35	3	3.85
	y_3	25	25	4.28
	u_1	31	8	3.38
	u_2	33	24	5.65
	u_3	33	11	4.49
GP-NARX	y_1	27	35	0.510
	y_2	13	38	0.687
	y_3	19	35	1.36
	u_1	35	22	0.575
	u_2	13	13	4.25
	u_3	23	2	3.28

5. Higher-order frequency response functions

When studying linear dynamic systems, it is often very useful to examine the frequency response functions (FRFs) in order to understand and predict how a system might respond to various loading conditions. Physically, the FRF of linear a system is known to arise from Duhamel's integral [26], which for a SDOF system can be written,

$$y(t) = \int_{-\infty}^{\infty} \mathrm{d}\tau h(\tau) x(t-\tau) \tag{28}$$

where $h(\tau)$ is the impulse response of the system and is sufficient to characterise the dynamics entirely [26]. The FRF of the system $H(\omega)$ can be obtained by taking Eq. (28) into the frequency domain, via the Fourier transform. Thus,

$$Y(\omega) = H(\omega)X(\omega) \tag{29}$$

The FRFs of a general linear system can be derived by the approach of harmonic probing. For a linear dynamic system given by some equation of motion,

$$M\ddot{\mathbf{y}}(t) + C\dot{\mathbf{y}}(t) + K\mathbf{y}(t) = \mathbf{x}(t) \tag{30}$$

with mass, stiffness and damping parameter matrices given by M, C, K respectively. The FRF $H^{(kr)}$ describing the response of the system at location r to an excitation at location k can be found by substituting in the probing input,

$$x_k(t) = e^{i\omega t} \tag{31}$$

From (28) and (29) the response of the system is therefore,

$$y_r(t) = H^{(kr)}(\omega)X_k(\omega) = H^{(kr)}(\omega)e^{i\omega t} \tag{32}$$

These expressions can be substituted into the equation of motion in (30), and in solving for $H^{(kr)}$, one arrives at the familiar formulation,

$$H^{(kr)}(\omega) = [M(i\omega)^2 + C(i\omega) + K]^{-1} \tag{33}$$

In the case of linear systems, the principle of superposition means that, if instead of $x_k(t) = e^{i\omega t}$, the probing input was comprised of independent harmonics $x_k(t) = e^{i\omega_1 t} + e^{i\omega_2 t}$, the overall response can still be calculated in terms of the FRF as,

$$y_r(t) = e^{i\omega_1 t}H^{(kr)}(\omega_1) + e^{i\omega_2 t}H^{(kr)}(\omega_2) \tag{34}$$

Nonlinear systems do not obey the principle of superposition. The impulse response function of a nonlinear system is no longer sufficient to characterise the dynamics; this means that Eq. (34) no longer holds and a more general approach is required. Fortunately, a generalisation of Eqs. (28) and (29) was developed by several authors (and named for the late Volterra) [27] in the late 1950s. The *Volterra series* extends Duhamel's integral as an infinite series,

$$y(t) = y_1(t) + y_2(t) + y_3(t) + \cdots \tag{35}$$

where,

$$y_1(t) = \int_{-\infty}^{+\infty} \mathrm{d}\tau h_1(\tau) x(t-\tau) \tag{36}$$

$$y_2(t) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} \mathrm{d}\tau_1\ \mathrm{d}\tau_2 h_2\left(\tau_1,\tau_2\right) x\left(t-\tau_1\right) x\left(t-\tau_2\right) \tag{37}$$

and so on. In this expansion, the $h_n(\tau_1,\ldots,\tau_n)$ are known as the *Volterra kernels* and act as generalisations of the impulse response of a dynamic system to multiple independent impulses. One can think of the $h_1(\tau_1)$ as being the impulse response of the system to

an excitation at time τ_1 as before. $h_2(\tau_1, \tau_2)$ is the nonlinear response of a system to impulses at τ_1 and τ_2. Clearly in the linear case, one has no such nonlinear contributions and so,

$$h_n(\tau_1, \dots, \tau_n) = 0 \quad \forall n > 1 \tag{38}$$

As before, a frequency-domain representation is possible. In this domain, the Volterra kernels transform to higher-order frequency response functions (HFRFs). The HFRFs generalise the FRFs for nonlinear systems and represent the nonlinear system responses to multiple excitation harmonics. A full derivation of the series expansions for arbitrary input is beyond the scope of this paper, but for a SISO system with a single excitation harmonic ($x(t) = e^{i\omega t}$), the response can be shown to be [26],

$$y(t) = H_1(\omega)\mathrm{e}^{i\omega t} + H_2(\omega, \omega)\mathrm{e}^{i2\omega t} + H_3(\omega, \omega, \omega)\mathrm{e}^{\mathrm{i}3\omega t} + \cdots \tag{39}$$

For a detailed explanation of the underlying theory of the Volterra series and general derivation for the MIMO case, the reader is directed towards the monograph [26].

As in the linear case, the HFRFs can be extracted from the equations of motion by using a harmonic probing approach [28,29]. For MIMO systems this process grows in complexity very quickly. Not only the HFRFs pertaining to differing input and output locations (i.e. $H^{(ij)}$ in linear analysis) must be considered, but also the cross-HFRFs arising from responses from nonlinear interactions between different input locations. In keeping with the notation of [26], a HFRF of order k, encoding the response at location j, from multiple independent harmonic inputs at locations a, b etc. may be written $H_k^{(j:aabb)}$. Note that independent harmonics may be applied at the same input location.

In the SIMO (and therefore SISO) case, a significant simplification is available. Because only a single input is present, all such cross-HFRFS can be ignored. This means that the HFRF structure can be expressed in terms of the HFRF vectors,

$$\mathbf{H}_k(\omega_1, \dots, \omega_k) = \begin{bmatrix} H_k^{(j:1)}(\omega_1, \dots, \omega_k) \\ \vdots \\ H_k^{(j:n)}(\omega_1, \dots, \omega_k) \end{bmatrix} \tag{40}$$

If the NLSI has been successful, the HFRFs of the physical displacements are expected to be close to their theoretical values, which are known for a SIMO quadratic-cubic Duffing-type system (Eq. (A.1)) to be equal to,

$$\mathbf{H}_1(\omega) = [M(i\omega)^2 + C(i\omega) + K]^{-1} \tag{41}$$

$$\mathbf{H}_2(\omega_1, \omega_2) = -K_2 \left(\mathbf{H}_1(\omega_1) \circ \mathbf{H}_1(\omega_2) \circ \mathbf{H}_1(\omega_1 + \omega_2)\right) \tag{42}$$

$$\begin{aligned} \mathbf{H}_3(\omega_1, \omega_2, \omega_3) = -\frac{1}{6}\mathbf{H}_1\left(\omega_1 + \omega_2 + \omega_3\right) \\ \circ \big[\ 4K_2\mathbf{H}_1\left(\omega_1\right) \circ \mathbf{H}_2\left(\omega_2, \omega_3\right) \\ +4K_2\mathbf{H}_1\left(\omega_2\right) \circ \mathbf{H}_2\left(\omega_3, \omega_1\right) \\ +4K_2\mathbf{H}_1\left(\omega_3\right) \circ \mathbf{H}_2\left(\omega_1, \omega_2\right) \\ +6K_3\mathbf{H}_1\left(\omega_1\right) \circ \mathbf{H}_1\left(\omega_2\right) \circ \mathbf{H}_1\left(\omega_3\right)\ \big] \end{aligned} \tag{43}$$

whereby $\circ$ denotes an elementwise product between vectors, and parameter matrices M, C, K, K_2, K_3 have their familiar meaning. These quantities are derived from the Volterra series [26] and can be obtained for arbitrary equations of motion using the method of harmonic probing [28,29]. The interested reader is directed to the appendices of this paper for a full derivation.

The utility of the above quantities in interpreting the nonlinear properties of the NLSI models of the NNMs is hopefully clear. $\mathbf{H}_1(\omega)$ describes the underlying linear part of the dynamics and it is interesting to compare this quantity to that of the underlying linear modes. The quantities $\mathbf{H}_2(\omega_1, \omega_2)$ and $\mathbf{H}_3(\omega_1, \omega_2, \omega_3)$ are completely scaled by the quadratic and cubic nonlinearities, providing insight into the types of nonlinearity present in the dynamics.

In order to examine the properties of the NLSI models in close detail, the HFRFs are also constructed for the GP-NARX models. The derivation of these quantities is algebraically laborious and so is reserved for the appendices at the end of this paper. The HFRFs of a SISO GP-NARX model are presented for the first time in vectorised form here. First let,

$$v_{jk} = \frac{H_{jk}}{\ell_k^{(y)}} \quad j \in 1, \dots, N_p \quad k \in 1, \dots, n_y \tag{44}$$

$$u_{jm} = \frac{H_{jm}}{\ell_m^{(x)}} \quad j \in 1, \dots, N_p \quad m \in n_y + 1, \dots, n_y + n_x \tag{45}$$

be training parameter matrices corresponding to the lagged outputs and inputs respectively, scaled by the lengthscales on the inputs ($\ell_k^{(y)}$) and outputs ($\ell_m^{(x)}$) respectively. Furthermore let,

$$\mathbf{A} = \sigma_f^2(\boldsymbol{\alpha} \circ \boldsymbol{\gamma}) = \sigma_f^2 \left[\mathrm{e}^{-\frac{1}{2}\left(\sum_k v_{jk}^2 + u_{jm}^2\right)} \circ \left[K(H, H) + \sigma_n^2 I\right]^{-1}\right] \mathbf{y} \tag{46}$$

be a $1 \times N_p$ vector of weights derived from the training data and kernel. Next define,

$$\boldsymbol{\Delta}_{\mathbf{y}}(\omega)_k = \frac{e^{-ki\omega,\Delta_t}}{\ell_k^{(y)}}, \quad k \in 1, \dots, n_y \tag{47}$$

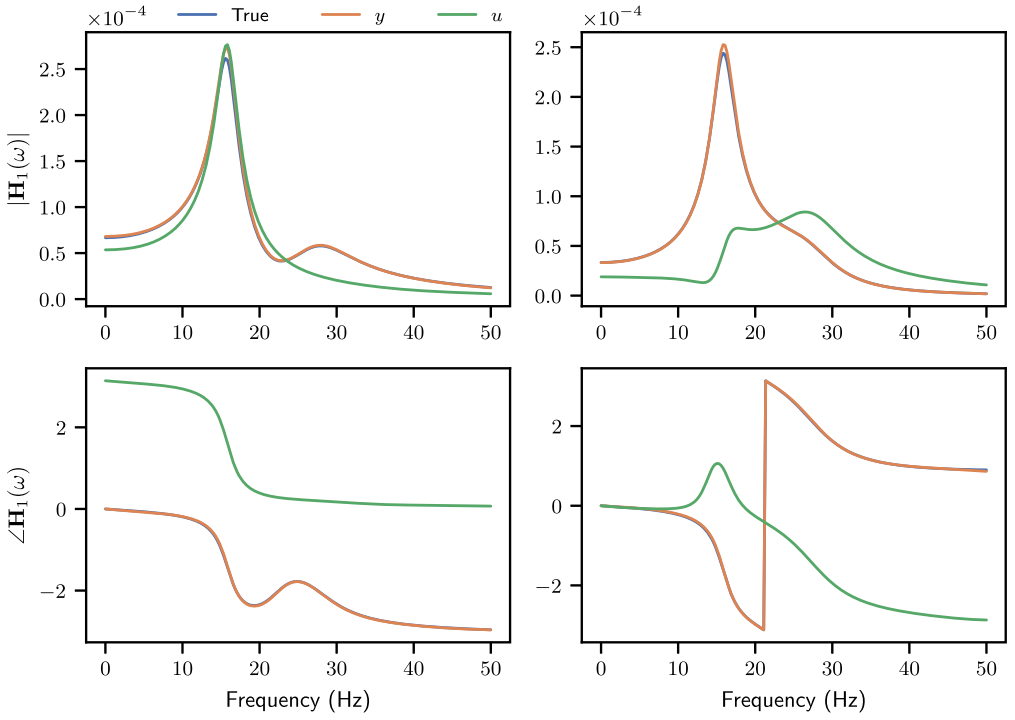

Fig. 12. Comparison of the $\mathbf{H}_1$s for the two-DOF nonlinear system.

$$\boldsymbol{\Delta}_{\mathbf{x}}(\omega)_m = \frac{e^{-(m-1)i\omega\Delta_t}}{\ell_m^{(x)}}, \quad m \in 1, \dots, n_x \tag{48}$$

as $1 \times n_y$ and $1 \times n_x$ vectors of lag operators scaled by the lengthscales where Δ_t is the sampling period of the data. Then the HFRFs can be expressed as,

$$\mathbf{H}_1(\omega) = \frac{\boldsymbol{\Delta}_{\mathbf{x}}(\omega) U A^{\mathsf{T}}}{\boldsymbol{\Delta}_{\mathbf{y}}(\omega) V A^{\mathsf{T}} - 1} \tag{49}$$

$$\mathbf{H}_2(\omega_1, \omega_2) = \frac{N_1 + N_2 + N_3 + N_4}{D - 1} \tag{50}$$

where,

$$N_1 = -\mathbf{H}_1(\omega_1)\mathbf{H}_1(\omega_2)\left[\left(\boldsymbol{\Delta}_{\mathbf{y}}(\omega_1)V\right) \circ \left(\boldsymbol{\Delta}_{\mathbf{y}}(\omega_2)V\right) - \boldsymbol{\Delta}_{\mathbf{y}}(\omega_1)^{\mathsf{T}}\boldsymbol{\Delta}_{\mathbf{y}}(\omega_2)\right]\mathbf{A}^{\mathsf{T}} \tag{51}$$

$$N_2 = -\mathbf{H}_1(\omega_1)\left[\left(\boldsymbol{\Delta}_{\mathbf{y}}(\omega_1)V\right) \circ \left(\boldsymbol{\Delta}_{\mathbf{x}}(\omega_2)U\right)\right]\mathbf{A}^{\mathsf{T}} \tag{52}$$

$$N_3 = -\mathbf{H}_1(\omega_2)\left[\left(\boldsymbol{\Delta}_{\mathbf{x}}(\omega_1)U\right) \circ \left(\boldsymbol{\Delta}_{\mathbf{y}}(\omega_2)V\right)\right]\mathbf{A}^{\mathsf{T}} \tag{53}$$

$$N_4 = -\left[\left(\boldsymbol{\Delta}_{\mathbf{x}}(\omega_1)U\right) \circ \left(\boldsymbol{\Delta}_{\mathbf{x}}(\omega_2)U\right) - \boldsymbol{\Delta}_{\mathbf{x}}(\omega_1)^{\mathsf{T}}\boldsymbol{\Delta}_{\mathbf{x}}(\omega_2)\right]\mathbf{A}^{\mathsf{T}} \tag{54}$$

$$D = \left[\boldsymbol{\Delta}_{\mathbf{y}}(\omega_1 + \omega_2)V\right]\mathbf{A}^{\mathsf{T}} \tag{55}$$

All other symbols retain their meaning from the previous sections. The expression for $\mathbf{H}_3(\omega_1, \omega_2, \omega_3)$ is more lengthy and is comprised of fourteen terms in the numerator and so is reserved for the appendices.

The formulations of the HFRFs for a GP-NARX model presented in this work offer several improvements over previously derived expressions. Primarily, the expressions given here consider for the first time, a more general GP case that incorporates differing lengthscales on each input to the GP (ARD). Another advantage is that these expressions are given in terms of precomputable vectors, offering a dramatic improvement in computational complexity compared to the equivalent-polynomial NARX method in [24].

The first three HFRFs are calculated for each SISO mapping $\mathbf{x} \to \{\mathbf{y}, \mathbf{u}\}$ for both of the benchmark nonlinear systems. The $\mathbf{H}_1(\omega)$s are plotted for the two-DOF system in Fig. 12 and for the three-DOF system in Fig. 13.

As can be seen in the figures, there is excellent agreement between the NLSI estimates and theoretical values of $\mathbf{H}_1(\omega)$ for the physical displacements. Using an NMSE metric to compare the true values to the predicted ones derived from the NLSI models returns

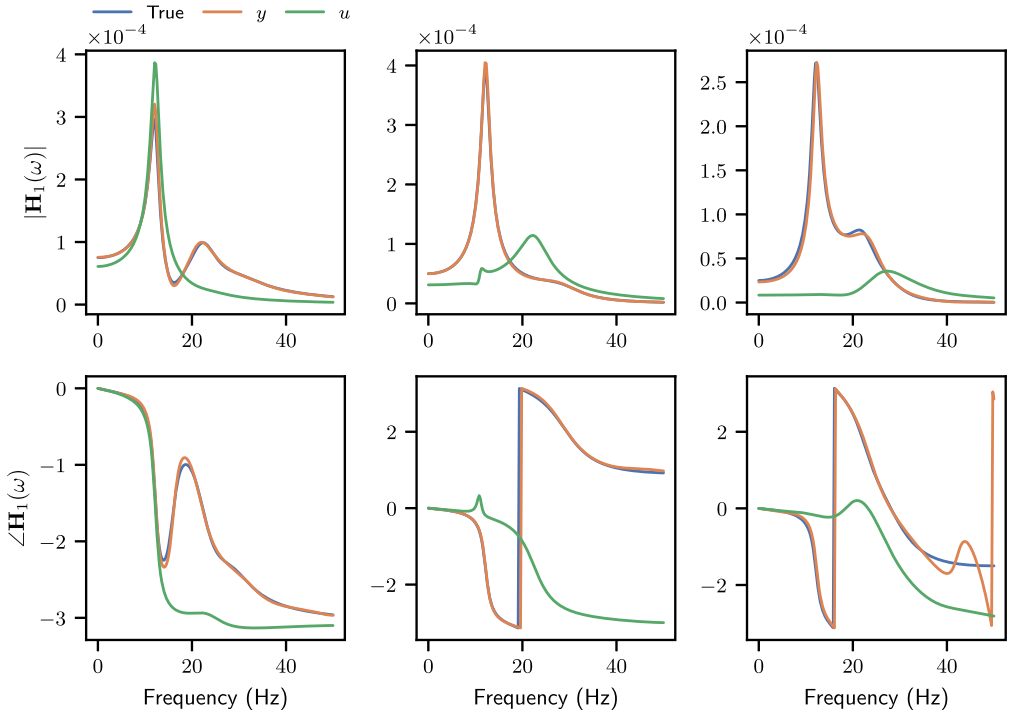

Fig. 13. Comparison of the $\mathbf{H}_1$s for the three-DOF nonlinear system.

values less then 0.008% in all cases. This agreement implies that the NLSI has been successful in reconstructing the dynamics of the underlying linear model precisely. The $\mathbf{H}_1$ of the modal coordinates show predominantly single frequency responses, indicating that a good decomposition into SISO systems has been achieved. It is interesting to note that there is still some evidence of structure at other frequencies in some of the modal coordinates, indicating that some dependence between the systems may not have been completely eliminated by modal transformation.

The benchmark nonlinear systems considered in this paper do not have any quadratic stiffness elements and so the expected values of the $\mathbf{H}_2(\omega_1, \omega_2)$ are zero for the physical displacements; these are plotted in Figs. 14 and 15 for the two-DOF system. As can be seen in the figures there is evidence of non-zero components in these plots that implies that some small spurious quadratic term may have been picked up by the GP-NARX formulation, which is perhaps unsurprising given the flexibility of the model class. However, the presence of peaks in the $\mathbf{H}_2(\omega_1, \omega_2)$ magnitudes does not necessarily mean that there is a meaningful contribution from these terms. Because the $\mathbf{H}_2$ are complex-valued, and derived from the $\mathbf{H}_1$, any structure (even noise) will present as peaks at locations ω_1, ω_2 and $\omega_1 + \omega_2$. The remaining plots visualising the higher-dimensional HFRFs are excluded from the body of this paper for reasons of space.

6. Discussion

It is argued in this paper that for a useful nonlinear modal decomposition there are two criteria for success: A decomposition into SISO functionals and nonlinear reconstruction of the physical displacements. The statistically-independent NNMs considered here demonstrate strong performance in both areas.

Considering the success of the SISO decomposition, it can be seen from the results of the NLSI that both the physical and modal displacements can be well approximated by SISO models of the dynamics. As expected the GP-NARX formulation outperforms the linear ARX models in almost every case indicating that nonlinearity remains present in the dynamics of the modal displacements. Some good NMSE scores are achieved by the ARX models but these can likely be explained by the linear model learning a biased linear representation of the cubic stiffness.

Also indicative of a good decomposition into SISO functionals are the largely single-peak structures of the $\mathbf{H}_1$ for the modal coordinates in fact. Another particularly encouraging observation is that the peaks of the modal $\mathbf{H}_1$ are very close to the underlying linear natural frequencies. Percentage differences between the peaks in the modal $\mathbf{H}_1$ and underlying linear natural frequencies are $[0.45\%, 4.1\%]$ for the two-DOF system and $[0.33\%, 1.73, 6.85\%]$ for the three-DOF system. These results indicate that the underlying linear modal structure has been retained and the discrepancies can likely be explained by a decreasing signal-to-noise ratio in the higher frequency modes. As discussed, the $\mathbf{H}_2$ (and therefore $\mathbf{H}_3$) of the GP-NARX contain some spurious structure. One explanation for this observation is the flexibility of the GP-NARX model class. However, it is not possible to rule out the possibility that the

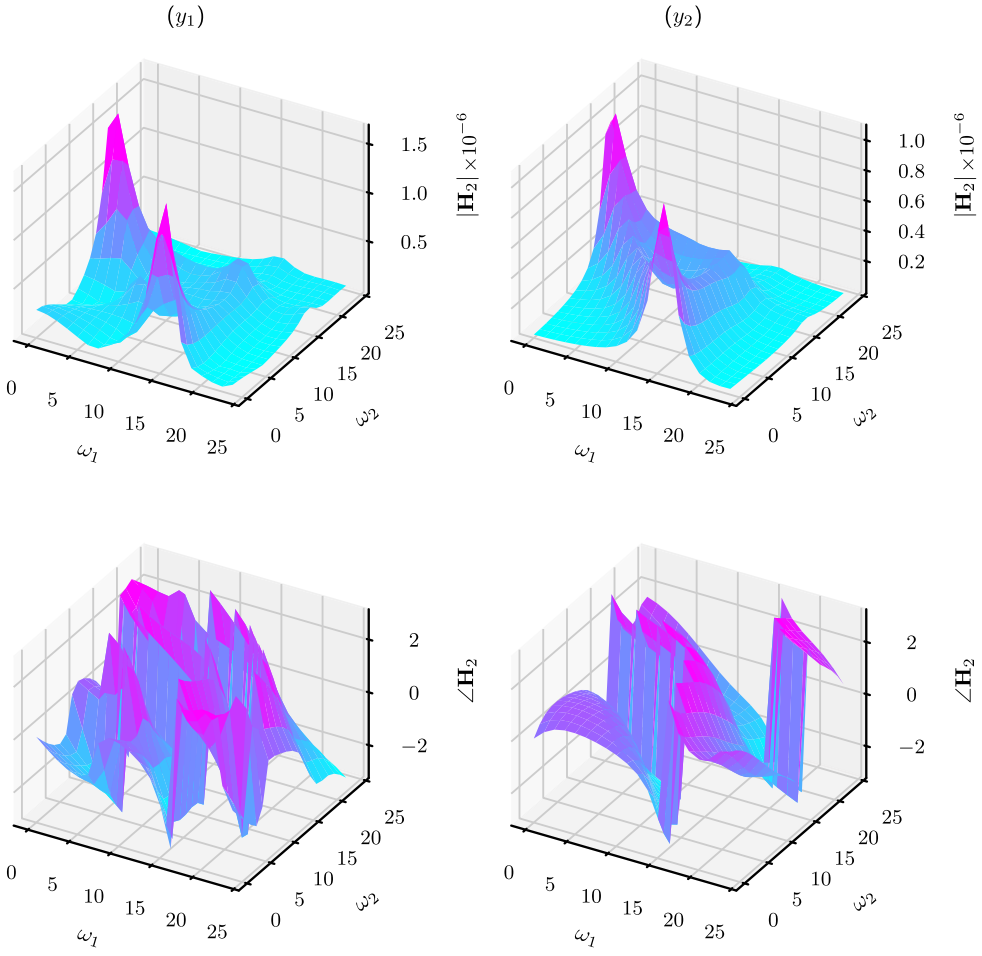

Fig. 14. $\mathbf{H}_2$ of the GP-NARX model of $\mathbf{x} \to \mathbf{y}$.

dynamics $\mathbf{x} \to \mathbf{u}$ have not been truly decoupled by a static map on the output only. Some possible extensions to the framework to improve this are discussed later in this section.

With regard to the modal reconstructions, the performance of the transformations learned in this work are excellent. An advantage of the cycle-GAN is that the inverse mapping is constructed alongside the forward modal transformation. Practically, this results in excellent reconstruction performance. For the two-DOF system, the overall reconstruction introduces NMSEs of 0.0262% and 0.0133% for each DOF respectively over the entire 10^5 simulated points. Performance on the three-DOF system degrades slightly, but still provides a high-fidelity reconstruction with NMSE scores of 0.0972%, 1.032% and 0.371% for the three degrees of freedom respectively. Once the cycle-GAN is trained, the overall mapping $\mathbf{y} \to \hat{\mathbf{y}}$ or $f(f^{-1}(\mathbf{y}))$ is static and is plotted in Fig. 16 for the two-DOF transformation. As can be seen in the figure, the overall reconstruction mapping is very close to unity, indicating an excellent level of nonlinear reconstruction. The regions of Fig. 16 that depart from unity are the regions of the input-space corresponding to high amplitudes. Because of the nature of the Duffing-like equation, these regions of the input-space are inevitably under-sampled compared to lower amplitude regions and it is therefore unsurprising that these regions are less well reconstructed.

A stringent test of both the SISO decomposition and reconstruction is to project the MPO model predictions of the modal coordinates back onto the physical coordinate system via the inverse modal transformation. These estimates can then be compared to the true observed physical displacements by an NMSE score. This reconstruction enables a quantitative measure of how well the NNMs function as a SISO basis for representing the SIMO dynamics of the problem. In effect, this allows one to measure the effectiveness of the decomposition quantitatively. The results of such an investigation are presented in Table 6. In the table, the SISO model reconstruction (the y_i reconstructed from the NLSI predictions of the u_i) are compared to the SIMO reconstructions ($\hat{\mathbf{y}} = f^{-1}(f(\mathbf{y}))$) for 900 sampling points from the unseen testing set (the first 100 points are discarded to remove the effects of differing lag structures between NLSI models).

As can be seen in Table 6, the SISO decomposition produces excellent reconstructions for the two-DOF and good reconstructions three-DOf systems. This result indicates quantitatively that the modal decomposition represents a practical nonlinear decomposition of the SIMO functional into SISO models. It is interesting to note that some of the SISO reconstruction errors are worse than the

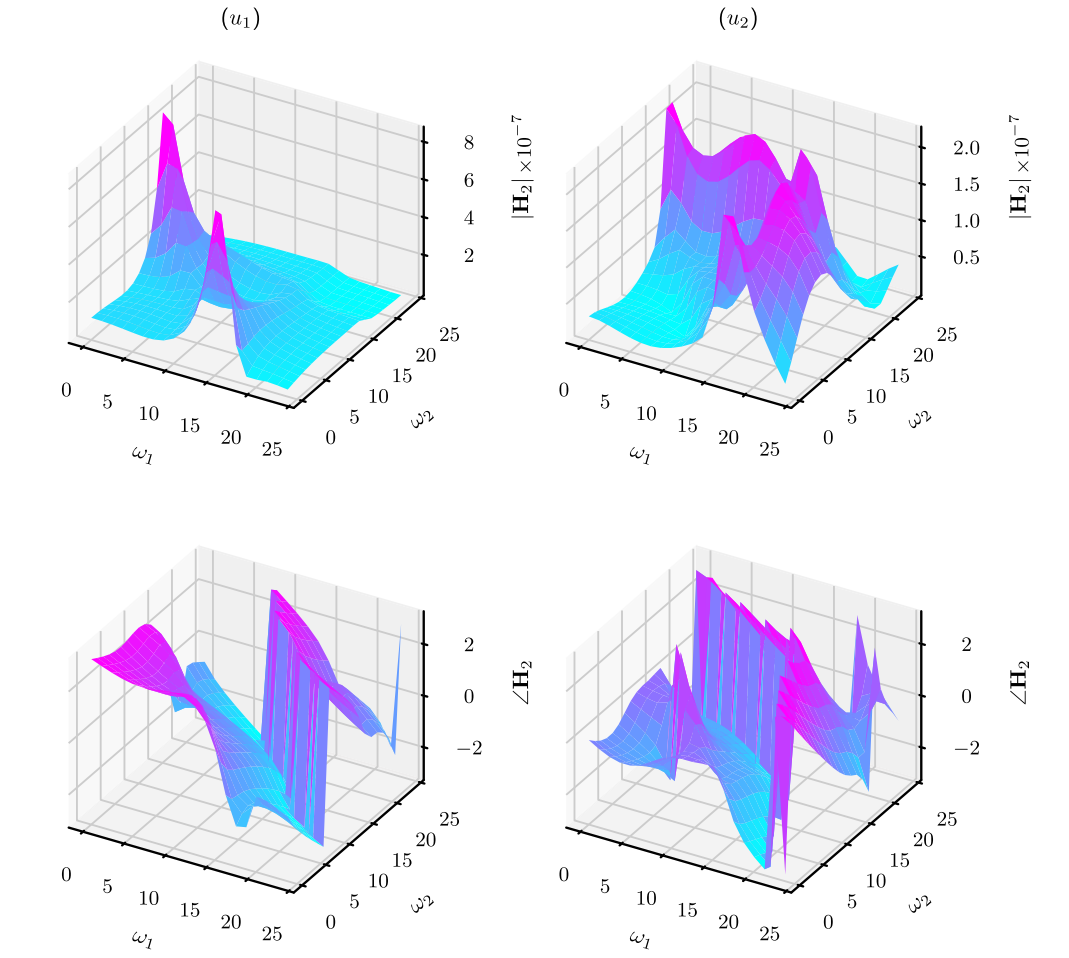

Fig. 15. $\mathbf{H}_2$ of the GP-NARX model of $\mathbf{x} \to \mathbf{u}$.

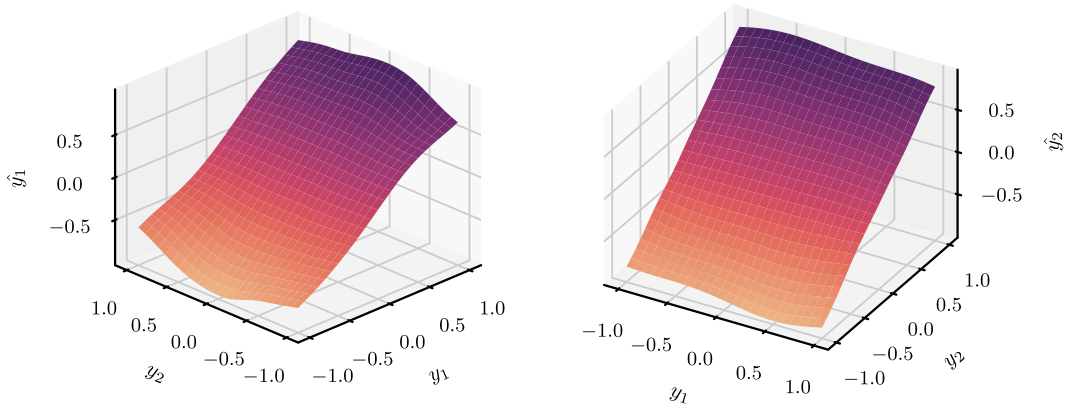

Fig. 16. Overall functional reconstruction of the forward and inverse modal transformations for the two-DOF system.

NMSE MPO errors on the corresponding modal coordinates. An example is present for the three-DOF system; the SISO reconstruction of y_1 is 2.25% but the NMSE MPO error for u_1 is 0.575%. This is not an unexpected result. The reconstruction is known to introduce some error due to the inevitable difficulty of finding an invertible map from $\mathbf{y} \to \mathbf{u}$. Another important consideration is that the inverse map f^{-1} constructs each y_i from all modal coordinates in a nonlinear way resulting in SISO reconstructions that may be

Table 6
Comparison of NMSE reconstruction performance between the SISO modal models and the reconstructed input.

System	DOF	Cycle-GAN reconstruction	SISO reconstruction
2-DOF	1	0.0347	0.660
	2	0.0252	0.765
3-DOF	1	0.103	2.25
	2	0.704	1.34
	3	0.319	1.74

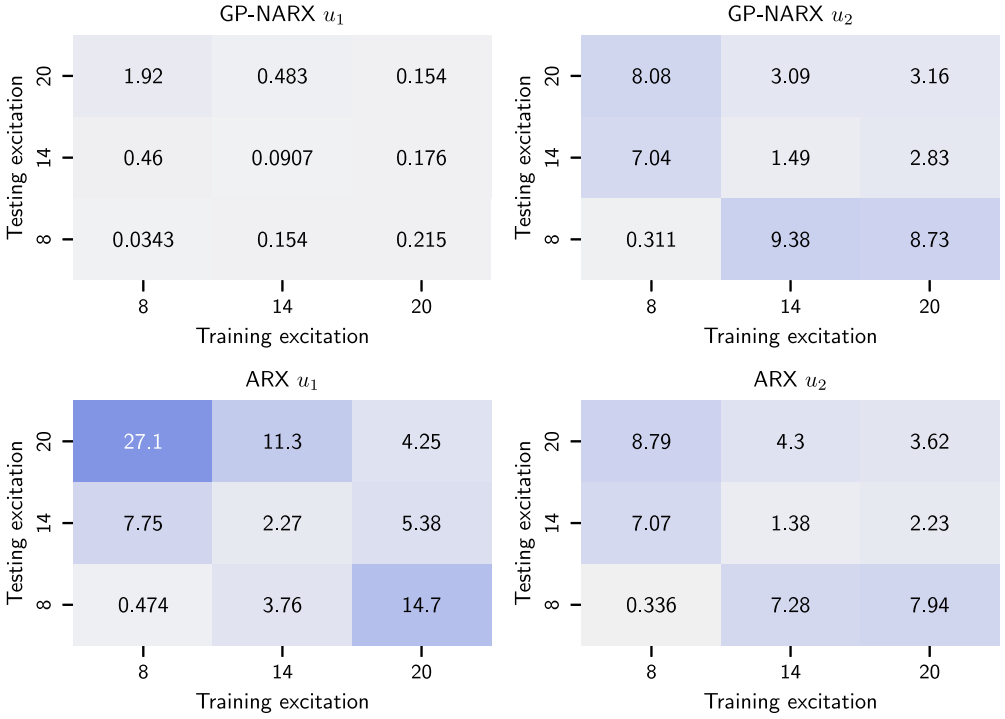

Fig. 17. Cross-validation NMSE of GP-NARX and ARX models of modal displacements at different excitation levels for the two-DOF system.

worse than the corresponding NLSI model for the modal coordinates. However, it is still preferable to conduct the NLSI on an independent basis. Assuming a priori that the modal DOFs are independent allows one to ignore all cross dependencies between DOFs. In the example of a polynomial-NARX model, the number of terms in a SISO polynomial basis of order r with d DOFs is given by,

$$d \times \binom{r + n_x + n_y}{r} \tag{56}$$

where n_x and n_y are the number of lags on the input and output respectively. In the full MIMO case the number of terms scales combinatorically, representing a significant increase in the difficulty of the identification problem.

It is also interesting to examine the generalisation performance of the nonlinear transformation. In order to achieve this, additional nonlinear data is generated for both the two-DOF and three-DOF systems at the lower excitation levels of $\sigma_x = 14$ and $\sigma_x = 8$. The modal displacements for these additional data are then produced by the same forward modal transformation f that was used for the data at $\sigma_x = 20$. The NLSI process is then repeated for each of the new excitation levels. The resultant MPO prediction NMSE scores (on each of the unseen testing data) are collected for the modal displacements in Figs. 17 and 18.

The figures show good generalisation of the nonlinear models compared to linear models which constitutes further evidence of nonlinearity in the functional $\mathbf{x} \to \mathbf{u}$. As one might expected, the best generalisation performance for linear and nonlinear models is present when the training excitation level is close to the testing level. For the two-DOF system, the nonlinear models show an extremely good level of generalisation (all but one NMSE $< 1\%$) for the first DOF and fairly strong generalisation between excitation levels for the second DOF. For the three-DOF system there is a similar pattern with very good generalisation (all but one NMSE $< 2\%$) in the first DOF and reasonable generalisation in the second and third DOFs.

A particularly interesting observation is that, for some of the modal coordinates $u_i, \quad i > 1$, the linear models also show a reasonable level of generalisation. The authors offer two explanations for this phenomena. The first is that the NLSI may be entirely

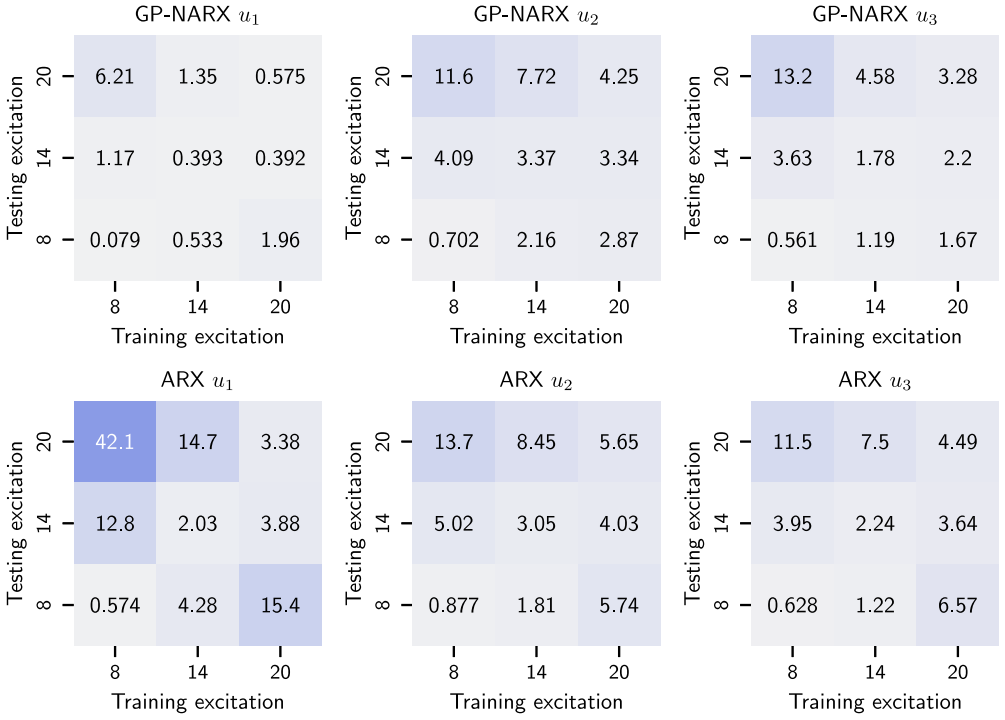

Fig. 18. Cross-validation NMSE of GP-NARX and ARX models of modal displacements at different excitation levels for the three-DOF system.

limited by the poor signal-to-noise ratio in the data and so the GP-NARX makes a simplistic linear approximation. This seems unlikely given the flexibility of the GP-NARX model and the relatively good NMSE scores that are found by both model classes when trained and tested at the same excitation level. A second potentially very interesting explanation is that the cycle-GAN has learned a static map that is able to decouple the SIMO nonlinear functional into SISO nonlinear and SISO linear functionals. Such a decoupling would represent a highly-useful tool in the analysis of nonlinear structural dynamics. Although the generalisation study presented here provides some evidence of such a decomposition, a formal proof is not available at this stage. Certainly, there are analyses that could be fruitful in this direction and the authors are actively pursuing these.

One possible issue with the nonlinear modal analysis considered by this paper is the normalisation scheme in the training of the cycle-GAN. All input displacements to the model are pre-scaled onto the interval $[-1, 1]$. Although this is common practice in machine-learning, the scaling may be acting to limit the generalisation potential of the modal transformation, as both low and high-amplitude responses are mapped to the same interval despite having potentially very dissimilar dynamics. A more sophisticated normalisation scheme is an interesting avenue for additional work and may enable a greater generalisation of the modal transformation between excitation levels.

7. Limitations and future work

The statistically-independent NNMs have been shown to have a number of desirable properties for the practising engineer in terms of decomposition and modelling. Also of practical concern is the application of the method in an experimental setting. Although the approach has already been applied to a number of simulated and experimental case-studies, there has not yet been a thorough treatment of how the method might be applied in an experimental setting generally. For example, the specification of the number of modes to include in the model (and therefore the number of sensors to provision) is an open problem. A related problem is the treatment of missing measurements. The authors envisage that these practical concerns can likely be approached in much the same manner as is done in experimental modal analysis in the linear case. For example, the appropriate number of modes could be determined by qualitative assessment of the PSDs of the structure of interest, whereby the number of modes can be related to the number of resonances. A full treatment of these ideas is beyond the scope of the current investigation and is therefore left for future investigation.

Another limitation of the present methodology is the time taken to learn and validate the maps in the cycle-GAN model. Whilst not insurmountable for the low numbers of DOFs considered here, the complexity increases somewhat with increasing dimensionality. At the core of the statistically-independent framework is the machine-learning problem of specifying the forward and inverse maps, the computational complexity of which is strongly related to model choice. Indeed, the cycle-GAN approach presented in [14]

already represents a significant computational saving over the approach of [3]. Alternative methods of computing the statistically-independent NMMs are already an active area of investigation. For example, Ref. [30] uses an approach based on normalising-flows and [31] uses an approach based on variational-autoencoder models.

All of the statistically-independent nonlinear modal analyses considered thus far have been focussed on a single output-only transformation on the physical displacements that map to their modal counterparts. However, it is interesting to note that this is not even sufficient for a total decoupling of the dynamics in the linear case. In linear modal analysis, the modal matrix also transforms the input to the systems into the *modal force vector*,

$$\Psi^{\top}\mathbf{x} = \mathbf{z} \tag{57}$$

so that the functional,

$$\Psi^{\top}\mathbf{x} = \mathbf{z} \rightarrow \mathbf{u} = \Psi\mathbf{y} \tag{58}$$

is truly decoupled. It is of interest to imagine what might be achievable by the application of an additional nonlinear map on the input to a nonlinear system, even in the SIMO case.

As stated above, no equivalence can be currently claimed between the present framework and the previous approaches of Rosenberg and Shaw–Pierre, owing to structural differences in the modal ansatz. However, it is interesting to imagine extensions to the present investigation that might reconcile these differences. Certainly, if theoretical manifolds of Shaw–Pierre could be learnt via a data-driven statistical framework (perhaps by the application of additional inductive biases) it would represent a significant tool in the analysis of nonlinear structural dynamics.

8. Conclusions

Statistically-independent NNMs are an interesting avenue for the analysis of nonlinear structures subject to broadband random excitation – a common scenario in structural dynamics. The NNMs constructed in this paper have been shown to be successful in both metrics defined as criteria for a practical nonlinear extension to modal analysis; SISO decomposition and modal reconstruction.

In terms of the first criteria, a static-decomposition into SISO functionals has been identified. The independence of these functionals is verified by the fitting of SISO NLSI models that are able to achieve very high-fidelity representations of the modal dynamics. It is also demonstrated that the SISO functionals are able to locally-generalise across excitation-levels indicating that the nonlinear dynamics of the system are accurately preserved by the decomposed formulation. By analysis of the HFRFs of the nonlinear SISO functionals, it is also shown that the underlying linear modes are preserved in the dynamics of the nonlinear system. This implies that the statistically-independent NNMs identified here are akin to the linear modal dynamics with additional nonlinear structure.

In terms of a modal reconstruction, the statistically-independent NNMs considered in this paper are also shown to be successful. The inverse modal transformation is able to reconstruct the physical displacements of the nonlinear system with a very high degree of fidelity. It is also shown that a strictly-SISO reconstruction (wherein SISO NLSI models of the modal displacements are used) also permits a good reconstruction, indicating that a practical modal reconstruction is possible from a simplified SISO basis.

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.

Acknowledgements

MDC would like to acknowledge the support of the EPSRC, UK grant EP/L016257/1.

GT would like to thank the EU's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement DyVirt (764547).

KW would like to acknowledge support via the EPSRC Established Career Fellowship, UK EP/R003625/1 and Programme Grant EP/R006768/1.

Appendix A. Harmonic probing of a quadratic-cubic simo duffing oscillator

The derivations presented in this section follow the results derived in [32] for the HFRFs of MDOF nonlinear systems but are presented in a convenient vectorised formulation. The application of the harmonic probing algorithm permits the derivation of analytical HFRFs, when the equations-of-motion are known. For the SIMO quadratic-cubic system with n DOFs under consideration, the EOM in matrix form are given by,

$$M\ddot{\mathbf{y}} + C\dot{\mathbf{y}} + K\mathbf{y} + K_2\mathbf{y}^2 + K_3\mathbf{y}^3 = \mathbf{x}(t) \tag{A.1}$$

where $\mathbf{y}$ and $\mathbf{x}$ are vectors, and M, C, K, K_2, and K_3 are square $n \times n$ parameter matrices.

The $\mathbf{H}_k$ encode the kth-order frequency domain response of the system to an input at location j_{input}.

The $\mathbf{H}_1$ encode the linear responses of the structure. The appropriate probing input is,

$$\mathbf{x}(t) = \mathbf{p}e^{i\omega t} \tag{A.2}$$

where,

$$\mathbf{p} = \delta_{jj_{\text{input}}} \tag{A.3}$$

is a *one-hot* vector corresponding to the input location of the SIMO system. The generalised probing output is given by (39),

$$\mathbf{y}(t) = \begin{bmatrix} H_1^{(1:1)}(\omega)e^{i\omega t} + H_2^{(1:1,1)}(\omega,\omega)\mathrm{e}^{i2\Omega t} + \cdots \\ \vdots \\ H_1^{(n:1)}(\omega)e^{i\omega t} + H_2^{(1:1,1)}(\omega,\omega)\mathrm{e}^{i2\Omega t} + \ldots \end{bmatrix} = \mathbf{H}_1 e^{i\omega t} \tag{A.4}$$

In the above expression, the coefficients of the $\mathbf{H}_1$ are all $e^{i\omega t}$. By utilising the orthogonality properties of independent harmonic inputs, all higher-order terms in the above expansion can be neglected leaving,

$$\mathbf{y}(t) = \begin{bmatrix} H_1^{(1:1)}(\omega)e^{i\omega t} \\ \vdots \\ H_1^{(n:1)}(\omega)e^{i\omega t} \end{bmatrix} = \mathbf{H}_1 e^{i\omega t} \tag{A.5}$$

Substituting (A.2) and (A.5) into the equation-of-motion gives,

$$\mathbf{H}_1(\omega)e^{i\omega t}[M(i\omega)^2 + C(i\omega) + K] + N = \mathbf{p}e^{i\omega t} \tag{A.6}$$

where N contains the terms with harmonics at integer multiples of ω from the quadratic and cubic stiffness terms. Equating coefficients of $e^{i\omega t}$ in (A.6) gives,

$$\mathbf{H}_1(\omega)[M(i\omega)^2 + C(i\omega) + K] = \mathbf{p} \tag{A.7}$$

and so the $\mathbf{H}_1$ can be read off,

$$\mathbf{H}_1(\omega) = [M(i\omega)^2 + C(i\omega) + K]^{-1}\mathbf{p} \tag{A.8}$$

As expected, the expression is identical to the FRF of the underlying linear SIMO system (the case where $K_2 = \mathbf{0}$ and $K_3 = \mathbf{0}$).

In order to analyse the nonlinear behaviour of the system, the higher-order $\mathbf{H}_k$ must be extracted. The process of harmonic-probing for $\mathbf{H}_2(\omega,\omega)$ is identical, although this time the probing input is comprised of two harmonics,

$$\mathbf{x}(t) = \mathbf{p}\left(e^{i\omega_1 t} + e^{i\omega_2 t}\right) \tag{A.9}$$

so that the effect of interacting harmonics can be captured. The system response to this input can be derived from the Volterra series. Considering only the terms that can generate coefficients of the harmonic $e^{i(\omega_1+\omega_2)t}$ one has,

$$\mathbf{y}(t) = \mathbf{H}_1(\omega_1)e^{i\omega_1 t} + \mathbf{H}_1(\omega_2)e^{i\omega_2 t} + 2\mathbf{H}_2(\omega_1,\omega_2)e^{i(\omega_1+\omega_2)t} \tag{A.10}$$

Substituting (A.9) and (A.10) into (A.1) as before and extracting the coefficients of $e^{i(\omega_1+\omega_2)t}$ results in

$$\left[-2M\omega_1^2 + (2iC - 4M\omega_2)\omega_1 + 2iC - 2M\omega_2^2 + 2K\right]\mathbf{H_2}\left(\omega_1,\omega_2\right) + 2K_2\mathbf{H_1}\left(\omega_1\right)\mathbf{H_1}\left(\omega_2\right) = \mathbf{0} \tag{A.11}$$

Solving the above for $\mathbf{H}_2$ results in,

$$\mathbf{H}_2(\omega_1,\omega_2) = -K_2\mathbf{H}_1(\omega_1)\circ\mathbf{H}_1(\omega_2)[M(i(\omega_1+\omega_2))^2 + C(i(\omega_1+\omega_2)) + K]^{-1} \tag{A.12}$$

where $\circ$ denotes an elementwise product over the degrees-of-freedom. Simplifying in terms of the previously computed $\mathbf{H}_1(\omega)$,

$$\mathbf{H}_2(\omega_1,\omega_2) = -K_2\mathbf{H}_1(\omega_1)\circ\mathbf{H}_1(\omega_2)\circ\mathbf{H}_1(\omega_1+\omega_2) \tag{A.13}$$

It is clear that in the absence of any quadratic nonlinear terms ($K_2 = \mathbf{0}$) there can be no contribution from $\mathbf{H}_2(\omega_1,\omega_2)$ to the overall response.

The calculation for $\mathbf{H}_3$ is somewhat more tedious, but follows the same approach as the lower-order terms. Starting with the probing input,

$$\mathbf{x}(t) = \mathbf{p}\left(e^{i\omega_1 t} + e^{i\omega_2 t} + e^{i\omega_3 t}\right) \tag{A.14}$$

the output from the Volterra series is given by,

$$\begin{aligned}\mathbf{y}(t) =& \mathbf{H}_1(\omega_1)e^{i\omega_1 t} + \mathbf{H}_1(\omega_2)e^{i\omega_2 t} + \mathbf{H}_1(\omega_3)e^{i\omega_3 t} \\ &+ 2\mathbf{H}_2(\omega_1,\omega_2)e^{i(\omega_1+\omega_2)t} + 2\mathbf{H}_2(\omega_2,\omega_3)e^{i(\omega_2+\omega_3)t} + +2\mathbf{H}_2(\omega_1,\omega_3)e^{i(\omega_1+\omega_3)t} \\ &+ 6\mathbf{H}_3(\omega_1,\omega_2,\omega_3)e^{i(\omega_1+\omega_2+\omega_3)t}\end{aligned} \tag{A.15}$$

where higher-order terms that cannot produce coefficients of $e^{i(\omega_1+\omega_2+\omega_3)t}$ can be ignored. Substituting (A.14) and (A.15) into (A.1) and extracting the coefficients of the harmonic $e^{i(\omega_1+\omega_2+\omega_3)t}$ gives,

$$\begin{aligned}6\mathbf{H}_3(\omega_1,\omega_2,\omega_3) = &\\ &\left(-6\left(\omega_1+\omega_2+\omega_3\right)^2 M + 6iC\omega_1 + 6iC\omega_2 + 6iC\omega_3 + 6K\right)\mathbf{H}_3\left(\omega_1,\omega_2,\omega_3\right)\\ &+ 6\mathbf{H}_1\left(\omega_1\right)\mathbf{H}_1\left(\omega_2\right)\mathbf{H}_1\left(\omega_3\right)K_3 + 4K_2\mathbf{H}_1\left(\omega_2\right)\mathbf{H}_2\left(\omega_1,\omega_3\right)\\ &+ 4K_2\mathbf{H}_1\left(\omega_3\right)\mathbf{H}_2\left(\omega_1,\omega_2\right) + 4\mathbf{H}_1\left(\omega_1\right)\mathbf{H}_2\left(\omega_2,\omega_3\right)K_2\end{aligned} \tag{A.16}$$

Solving finally for $\mathbf{H}_3(\omega_1,\omega_2,\omega_3)$ and substituting in the definitions of $\mathbf{H}_1$ and $\mathbf{H}_2$ gives the simplified solution,

$$\begin{aligned}\mathbf{H}_3(\omega_1,\omega_2,\omega_3) = -\frac{1}{6}&\mathbf{H}_1\left(\omega_1+\omega_2+\omega_3\right)\\ &\circ\left[\ 4K_2\mathbf{H}_1\left(\omega_1\right)\circ\mathbf{H}_2\left(\omega_2,\omega_3\right)\right.\\ &\quad +4K_2\mathbf{H}_1\left(\omega_2\right)\circ\mathbf{H}_2\left(\omega_3,\omega_1\right)\\ &\quad +4K_2\mathbf{H}_1\left(\omega_3\right)\circ\mathbf{H}_2\left(\omega_1,\omega_2\right)\\ &\quad \left.+6K_3\mathbf{H}_1\left(\omega_1\right)\circ\mathbf{H}_1\left(\omega_2\right)\circ\mathbf{H}_1\left(\omega_3\right)\right]\end{aligned} \tag{A.17}$$

Appendix B. Harmonic probing of the SISO GP-NARX model

One begins with the generic formulation for a kernel-NARX model,

$$y_t = \sigma_f^2\sum_j^{N-p}\alpha_j\kappa(X,X') \tag{B.1}$$

where the α_j are the kernel weights. In a GP these are given by finding the product of the inverse of the covariance matrix and the observations,

$$\alpha = [K(X,X)+\sigma_n^2\mathbf{I}]^{-1}\mathbf{y} \tag{B.2}$$

In the above expressions, X is the Hankel matrix of the training inputs. The GP-NARX formulation treats both lagged input and output values as pure inputs to the GP,

$$X = \begin{bmatrix} y_{t-1} & y_{t-2} & \cdots & y_{t-n_y} & x_t & x_{t-1} & x_{t-2} & \cdots & x_{t-n_x+1}\\ y_{t-2} & y_{t-3} & \cdots & y_{t-n_y-1} & x_{t-1} & x_{t-2} & x_{t-3} & \cdots & x_{t-n_x}\\ y_{t-3} & y_{t-4} & \cdots & y_{t-n_y-2} & x_{t-2} & x_{t-3} & x_{t-4} & \cdots & x_{t-n_x-1}\\ \vdots & & & & \vdots & & & & \\ y_{t-N+p} & & & & x_{t-N+p-1} & & & & \end{bmatrix} \tag{B.3}$$

For a NARX(n_y, n_x) formulation, X has shape $p\times(N-p)$, where $p=(n_y+n_x)$. For notational convenience, let V refer to the first n_y columns of X and U refer to the last n_x columns.

In practice there are a number of suitable choices for the kernel function κ. In [24] the squared exponential kernel [22] is used and the HFRF are extracted based on a decomposition of the kernel into an equivalent NARX model. In this paper a similar approach based on a multivariate Taylor series expansion is used.

The squared-exponential kernel is given by,

$$\kappa(x_i,x_j) = \sigma_f^2\exp\left(-\frac{\left(x_i-x_j\right)^2}{2\ell^2}\right) \tag{B.4}$$

where ℓ is a hyperparameter controlling the length-scales of the process. In order to provide a better fit to the data, individual length-scales can be fitted to each input (or lag) in the GP. Denoting these by the vectors $\ell^{(x)}$ and $\ell^{(y)}$ respectively and then substituting (B.4) into the explicit form of the kernel-NARX model in (B.1) gives,

$$y_t = \sigma_f^2\sum_{j=1}^{N-p}\alpha_j\exp\left(-\frac{1}{2}\left[\sum_{k=1}^{n_y}\frac{(v_{jk}-y_{t-k})^2}{(\ell_k^{(y)})^2}+\sum_{m=0}^{n_x}\frac{(u_{jm}-y_{t-m})^2}{(\ell_m^{(x)})^2}\right]\right) \tag{B.5}$$

This expression can be considered to be a nonlinear discrete-time equation of motion for a system with input $x(t)$ and output $y(t)$. In order to apply the harmonic-probing algorithm the expression must first be converted into polynomial form by taking a multivariate series expansion in y_{t-k} and x_{t-m}. In the derivation that proceeds, the following notational shortcuts will be adopted for simplicity. Let,

$$v_{jk} = \frac{H_{jk}}{\ell_k^{(y)}} \quad j\in 1,\ldots,N_p \quad k\in 1,\ldots,n_y \tag{B.6}$$

$$u_{jm} = \frac{H_{jm}}{\ell_m^{(x)}} \quad j\in 1,\ldots,N_p \quad m\in n_y+1,\ldots,n_y+n_x \tag{B.7}$$

be training parameter matrices corresponding to the lagged outputs and inputs respectively, scaled by the lengthscales corresponding to that lag. Furthermore let,

$$\mathbf{A} = \sigma_f^2(\boldsymbol{\alpha}\circ\boldsymbol{\gamma}) = \sigma_f^2\left[\gamma\circ\left[K(H,H)+\sigma_n^2 I\right]^{-1}\right]\mathbf{y} \tag{B.8}$$

be a $1\times N_p$ vector of weights derived from the training data and kernel, where,

$$\gamma = \exp\left(-\frac{1}{2}\left[\sum_j V_j^2 + \sum_m U_m^2\right]\right) \tag{B.9}$$

are the terms inside the kernel that do not depend on $\mathbf{x}$ or $\mathbf{y}$.

The explicit representation in (B.5) can now be rewritten in vectorised form as,

$$y_t = \exp\left(\frac{-1}{2}\left[-2\mathbf{y}V+\mathbf{y}^{\mathsf{T}}\mathbf{y}-2\mathbf{x}U+\mathbf{x}^{\mathsf{T}}\mathbf{x}+\left(\sum_j V_j^2+\sum_m U_m^2\right)\right]\right)\mathbf{A}^{\mathsf{T}} \tag{B.10}$$

Expanding the exponential in the above as a series expansion in X and Y gives,

$$\begin{aligned} y_t = & \\ & \Big[\, 1 + \mathbf{x}U + \mathbf{y}V + (\mathbf{x}U)\circ(\mathbf{y}V) \\ & + \frac{1}{2}(\mathbf{y}V)\circ(\mathbf{y}V) + \frac{1}{2}(\mathbf{x}U)\circ(\mathbf{x}U) - \frac{1}{2}\mathbf{y}^{\mathsf{T}}\mathbf{y} - \frac{1}{2}\mathbf{x}^{\mathsf{T}}\mathbf{x} \\ & + \frac{1}{6}(\mathbf{y}V)\circ(\mathbf{y}V)\circ(\mathbf{y}V) + \frac{1}{6}(\mathbf{x}U)\circ(\mathbf{x}U)\circ(\mathbf{x}U) \\ & - \frac{1}{2}\mathbf{y}^{\mathsf{T}}(\mathbf{y}\circ\mathbf{y}) - \frac{1}{2}\mathbf{x}^{\mathsf{T}}(\mathbf{x}\circ\mathbf{x}) \\ & + \frac{1}{8}(\mathbf{y}V)\circ(\mathbf{y}V)\circ(\mathbf{x}U) + \frac{1}{8}(\mathbf{y}V)\circ(\mathbf{x}U)\circ(\mathbf{x}U) + \cdots \Big]\,\mathbf{A}^{\mathsf{T}} \end{aligned} \tag{B.11}$$

This expression now has a polynomial form in X and Y, and the harmonic probing algorithm can be applied. One might imagine that truncating the series might lead to inevitable errors in the calculations of the HFRF; however, this is not the case. To see why, a similar argument to the analytical approach above can be adopted. Coefficients of the appropriate probing input can only be generated by terms in the expansion up to a certain order. For the probing input of $x_p(t) = e^{i\omega t}$, only the first order terms in X and Y can produce coefficients of $e^{i\omega t}$. For a probing input of $x_p(t) = e^{i(\omega_1+\omega_2)t}$ only terms up to second order can produce the appropriate coefficients. This limits the algebraic difficulty of the harmonic-probing approach considerably.

One begins with the probing input for H_1 for a SISO system,

$$x^{(p)}(t) = e^{i\omega t} \tag{B.12}$$

and corresponding output from the Volterra series,

$$y^{(p)}(t) = e^{i\omega t}H_1(\omega) \tag{B.13}$$

where higher-order terms that cannot produce coefficients of $e^{i\omega t}$ can be ignored. The GP-NARX formulation is in discrete time and so the differential operator must be applied to lagged inputs and outputs,

$$x_{t-m}^{(p)} = e^{i\omega t}e^{-im\omega\Delta_t} \tag{B.14}$$

$$y_{t-k}^{(p)} = e^{i\omega t}H_1(\omega)e^{-ik\omega\Delta_t} \tag{B.15}$$

where Δ_t is the sampling period of the discrete time model. Next define,

$$\boldsymbol{\Delta}_{\mathbf{y}}(\omega)_k = \frac{e^{-ki\omega,\Delta_t}}{\ell_k^{(y)}}, \quad k\in 1,\ldots,n_y \tag{B.16}$$

$$\boldsymbol{\Delta}_{\mathbf{x}}(\omega)_m = \frac{e^{-(m-1)i\omega,\Delta_t}}{\ell_m^{(x)}}, \quad m\in 1,\ldots,n_x \tag{B.17}$$

as vectorised lag operators so that Eqs. (B.14) and (B.15) may be written,

$$\mathbf{y}^p = e^{i\omega t}H_1(\omega)\boldsymbol{\Delta}_{\mathbf{y}}(\omega) \tag{B.18}$$

$$\mathbf{x}^p = e^{i\omega t}\boldsymbol{\Delta}_{\mathbf{x}}(\omega) \tag{B.19}$$

Substituting these into the series expansion in (B.11) gives,

$$e^{i\omega t}H_1(\omega) = \left[e^{i\omega t}\boldsymbol{\Delta}_{\mathbf{x}}(\omega)U + e^{i\omega t}H_1(\omega)\boldsymbol{\Delta}_{\mathbf{y}}(\omega)V + \cdots\right]\mathbf{A}^{\mathsf{T}} \tag{B.20}$$

where no other terms are able to generate coefficients of $e^{i\omega t}$. Solving the above for $H_1(\omega)$ produces the desired result,

$$H_1(\omega) = \frac{\boldsymbol{\Delta}_{\mathbf{x}}(\omega) U A^{\mathsf{T}}}{\boldsymbol{\Delta}_{\mathbf{y}}(\omega) V A^{\mathsf{T}} - 1} \tag{B.21}$$

For $H_2(\omega_1, \omega_2)$, the vectorised probing inputs and outputs can be written,

$$\mathbf{y}^p = e^{i\omega_1 t} H_1(\omega_1) \boldsymbol{\Delta}_{\mathbf{y}}(\omega_1) + e^{i\omega_2 t} H_1(\omega_2) \boldsymbol{\Delta}_{\mathbf{y}}(\omega_2) + 2e^{i\omega_1+\omega_2 t} H_2(\omega_1, \omega_2) \boldsymbol{\Delta}_{\mathbf{y}}(\omega_1 + \omega_2) \tag{B.22}$$

$$\mathbf{x}^p = e^{i\omega_1 t} \boldsymbol{\Delta}_{\mathbf{x}}(\omega_1) + e^{i\omega_2 t} \boldsymbol{\Delta}_{\mathbf{x}}(\omega_2) \tag{B.23}$$

Substituting these into (B.11) and considering only terms that generate coefficients of $e^{i\omega_1+\omega_2 t}$ and then solving for $H_2(\omega_1, \omega_2)$ produces,

$$H_2(\omega_1, \omega_2) = \frac{N_1 + N_2 + N_3 + N_4}{D - 1} \tag{B.24}$$

where,

$$N_1 = -\mathbf{H}_1(\omega_1)\mathbf{H}_1(\omega_2) \left[\left(\boldsymbol{\Delta}_{\mathbf{y}}(\omega_1) V \right) \circ \left(\boldsymbol{\Delta}_{\mathbf{y}}(\omega_2) V \right) - \boldsymbol{\Delta}_{\mathbf{y}}(\omega_1)^{\mathsf{T}} \boldsymbol{\Delta}_{\mathbf{y}}(\omega_2) \right] \mathbf{A}^{\mathsf{T}} \tag{B.25}$$

$$N_2 = -\mathbf{H}_1(\omega_1) \left[\left(\boldsymbol{\Delta}_{\mathbf{y}}(\omega_1) V \right) \circ \left(\boldsymbol{\Delta}_{\mathbf{x}}(\omega_2) U \right) \right] \mathbf{A}^{\mathsf{T}} \tag{B.26}$$

$$N_3 = -\mathbf{H}_1(\omega_2) \left[\left(\boldsymbol{\Delta}_{\mathbf{x}}(\omega_1) U \right) \circ \left(\boldsymbol{\Delta}_{\mathbf{y}}(\omega_2) V \right) \right] \mathbf{A}^{\mathsf{T}} \tag{B.27}$$

$$N_4 = -\left[\left(\boldsymbol{\Delta}_{\mathbf{x}}(\omega_1) U \right) \circ \left(\boldsymbol{\Delta}_{\mathbf{x}}(\omega_2) U \right) - \boldsymbol{\Delta}_{\mathbf{x}}(\omega_1)^{\mathsf{T}} \boldsymbol{\Delta}_{\mathbf{x}}(\omega_2) \right] \mathbf{A}^{\mathsf{T}} \tag{B.28}$$

$$D = \left[\boldsymbol{\Delta}_{\mathbf{y}}(\omega_1 + \omega_2) V \right] \mathbf{A}^{\mathsf{T}} \tag{B.29}$$

Finally for $H_3(\omega_1, \omega_2, \omega_3)$, the appropriate probing inputs and outputs are,

$$\begin{aligned} \mathbf{y}^p = & \\ & + e^{i\omega_1 t} H_1(\omega_1) \boldsymbol{\Delta}_{\mathbf{y}}(\omega_1) + e^{i\omega_2 t} H_1(\omega_2) \boldsymbol{\Delta}_{\mathbf{y}}(\omega_2) \\ & + 2e^{i\omega_1+\omega_2 t} H_2(\omega_1, \omega_2) \boldsymbol{\Delta}_{\mathbf{y}}(\omega_1 + \omega_2) \\ & + 2e^{i\omega_2+\omega_3 t} H_2(\omega_2, \omega_3) \boldsymbol{\Delta}_{\mathbf{y}}(\omega_2 + \omega_3) \\ & + 2e^{i\omega_1+\omega_3 t} H_2(\omega_1, \omega_3) \boldsymbol{\Delta}_{\mathbf{y}}(\omega_1 + \omega_3) \\ & + 6e^{i\omega_1+\omega_2+\omega_3 t} H_3(\omega_1, \omega_2, \omega_3) \boldsymbol{\Delta}_{\mathbf{y}}(\omega_1 + \omega_2 + \omega_3) \end{aligned} \tag{B.30}$$

$$\mathbf{x}^p = e^{i\omega_1 t} \boldsymbol{\Delta}_{\mathbf{x}}(\omega_1) + e^{i\omega_2 t} \boldsymbol{\Delta}_{\mathbf{x}}(\omega_2) + + e^{i\omega_3 t} \boldsymbol{\Delta}_{\mathbf{x}}(\omega_3) \tag{B.31}$$

Substituting these into (B.11) and considering terms up to third order that can generate coefficients of $e^{i(\omega_1+\omega_2+\omega_3)t}$ results in a very large expression, the solution of which for $H_3(\omega_1, \omega_2, \omega_3)$ gives an expression with fourteen terms in the numerator. The solution is,

$$H_3(\omega_1, \omega_2, \omega_3) = \frac{\sum_{i=1}^{14} N_i}{D - 1} \tag{B.32}$$

where,

$$\begin{aligned} N_1 = & -\frac{1}{6} H_1(\omega_1) H_1(\omega_2) H_1(\omega_3) \\ & \Big[(\boldsymbol{\Delta}_{\mathbf{y}}(\omega_1) V) \circ (\boldsymbol{\Delta}_{\mathbf{y}}(\omega_2) V) \circ (\boldsymbol{\Delta}_{\mathbf{y}}(\omega_3) V) \\ & - \boldsymbol{\Delta}_{\mathbf{y}}(\omega_1) V \boldsymbol{\Delta}_{\mathbf{y}}(\omega_2)^{\mathsf{T}} \boldsymbol{\Delta}_{\mathbf{y}}(\omega_3) - \boldsymbol{\Delta}_{\mathbf{y}}(\omega_2) V \boldsymbol{\Delta}_{\mathbf{y}}(\omega_3)^{\mathsf{T}} \boldsymbol{\Delta}_{\mathbf{y}}(\omega_1) - \boldsymbol{\Delta}_{\mathbf{y}}(\omega_3) V \boldsymbol{\Delta}_{\mathbf{y}}(\omega_1)^{\mathsf{T}} \boldsymbol{\Delta}_{\mathbf{y}}(\omega_2) \Big] \mathbf{A}^{\mathsf{T}} \end{aligned} \tag{B.33}$$

$$\begin{aligned} N_2 = & -\frac{1}{6} H_1(\omega_2) H_1(\omega_3) \\ & \Big[(\boldsymbol{\Delta}_{\mathbf{x}}(\omega_1) U) \circ (\boldsymbol{\Delta}_{\mathbf{y}}(\omega_2) V) \circ (\boldsymbol{\Delta}_{\mathbf{y}}(\omega_3) V) - \boldsymbol{\Delta}_{\mathbf{x}}(\omega_1) U \boldsymbol{\Delta}_{\mathbf{y}}(\omega_2)^{\mathsf{T}} \boldsymbol{\Delta}_{\mathbf{y}}(\omega_3) \Big] \mathbf{A}^{\mathsf{T}} \end{aligned} \tag{B.34}$$

$$\begin{aligned} N_3 = & -\frac{1}{6} H_1(\omega_1) H_1(\omega_3) \\ & \Big[(\boldsymbol{\Delta}_{\mathbf{y}}(\omega_1) V) \circ (\boldsymbol{\Delta}_{\mathbf{x}}(\omega_2) U) \circ (\boldsymbol{\Delta}_{\mathbf{y}}(\omega_3) V) - \boldsymbol{\Delta}_{\mathbf{x}}(\omega_2) U \boldsymbol{\Delta}_{\mathbf{y}}(\omega_1)^{\mathsf{T}} \boldsymbol{\Delta}_{\mathbf{y}}(\omega_3) \Big] \mathbf{A}^{\mathsf{T}} \end{aligned} \tag{B.35}$$

$$\begin{aligned} N_4 = & -\frac{1}{6} H_1(\omega_1) H_1(\omega_2) \\ & \Big[(\boldsymbol{\Delta}_{\mathbf{y}}(\omega_1) V) \circ (\boldsymbol{\Delta}_{\mathbf{y}}(\omega_2) V) \circ (\boldsymbol{\Delta}_{\mathbf{x}}(\omega_3) U) - \boldsymbol{\Delta}_{\mathbf{x}}(\omega_3) U \boldsymbol{\Delta}_{\mathbf{y}}(\omega_1)^{\mathsf{T}} \boldsymbol{\Delta}_{\mathbf{y}}(\omega_2) \Big] \mathbf{A}^{\mathsf{T}} \end{aligned} \tag{B.36}$$

$$N_5 = -\frac{1}{3}H_1(\omega_1)H_2(\omega_2,\omega_3) \left[(\boldsymbol{\Delta}_{\mathbf{y}}(\omega_1)V)\circ(\boldsymbol{\Delta}_{\mathbf{y}}(\omega_2+\omega_3)V) - \boldsymbol{\Delta}_{\mathbf{y}}(\omega_1)^{\mathsf{T}}\boldsymbol{\Delta}_{\mathbf{y}}(\omega_2+\omega_3)\right]\mathbf{A}^{\mathsf{T}} \tag{B.37}$$

$$N_6 = -\frac{1}{3}H_1(\omega_2)H_2(\omega_1,\omega_3) \left[(\boldsymbol{\Delta}_{\mathbf{y}}(\omega_2)V)\circ(\boldsymbol{\Delta}_{\mathbf{y}}(\omega_1+\omega_3)V) - \boldsymbol{\Delta}_{\mathbf{y}}(\omega_2)^{\mathsf{T}}\boldsymbol{\Delta}_{\mathbf{y}}(\omega_1+\omega_3)\right]\mathbf{A}^{\mathsf{T}} \tag{B.38}$$

$$N_7 = -\frac{1}{3}H_1(\omega_3)H_2(\omega_1,\omega_2) \left[(\boldsymbol{\Delta}_{\mathbf{y}}(\omega_3)V)\circ(\boldsymbol{\Delta}_{\mathbf{y}}(\omega_1+\omega_2)V) - \boldsymbol{\Delta}_{\mathbf{y}}(\omega_3)^{\mathsf{T}}\boldsymbol{\Delta}_{\mathbf{y}}(\omega_1+\omega_2)\right]\mathbf{A}^{\mathsf{T}} \tag{B.39}$$

$$N_8 = -\frac{1}{3}H_2(\omega_2,\omega_3) \left[(\boldsymbol{\Delta}_{\mathbf{x}}(\omega_1)U)\circ(\boldsymbol{\Delta}_{\mathbf{y}}(\omega_2+\omega_3)V)\right]\mathbf{A}^{\mathsf{T}} \tag{B.40}$$

$$N_9 = -\frac{1}{3}H_2(\omega_1,\omega_3) \left[(\boldsymbol{\Delta}_{\mathbf{x}}(\omega_2)U)\circ(\boldsymbol{\Delta}_{\mathbf{y}}(\omega_1+\omega_3)V)\right]\mathbf{A}^{\mathsf{T}} \tag{B.41}$$

$$N_{10} = -\frac{1}{3}H_2(\omega_1,\omega_2) \left[(\boldsymbol{\Delta}_{\mathbf{x}}(\omega_3)U)\circ(\boldsymbol{\Delta}_{\mathbf{y}}(\omega_1+\omega_2)V)\right]\mathbf{A}^{\mathsf{T}} \tag{B.42}$$

$$N_{11} = -\frac{1}{6}H_1(\omega_1) \left[(\boldsymbol{\Delta}_{\mathbf{y}}(\omega_1)V)\circ(\boldsymbol{\Delta}_{\mathbf{x}}(\omega_2)U)\circ(\boldsymbol{\Delta}_{\mathbf{x}}(\omega_3)U) - \boldsymbol{\Delta}_{\mathbf{y}}(\omega_1)V\boldsymbol{\Delta}_{\mathbf{x}}(\omega_2)^{\mathsf{T}}\boldsymbol{\Delta}_{\mathbf{x}}(\omega_3)\right]\mathbf{A}^{\mathsf{T}} \tag{B.43}$$

$$N_{12} = -\frac{1}{6}H_1(\omega_2) \left[(\boldsymbol{\Delta}_{\mathbf{x}}(\omega_1)U)\circ(\boldsymbol{\Delta}_{\mathbf{y}}(\omega_2)V)\circ(\boldsymbol{\Delta}_{\mathbf{x}}(\omega_3)U) - \boldsymbol{\Delta}_{\mathbf{y}}(\omega_2)V\boldsymbol{\Delta}_{\mathbf{x}}(\omega_1)^{\mathsf{T}}\boldsymbol{\Delta}_{\mathbf{x}}(\omega_3)\right]\mathbf{A}^{\mathsf{T}} \tag{B.44}$$

$$N_{13} = -\frac{1}{6}H_1(\omega_3) \left[(\boldsymbol{\Delta}_{\mathbf{x}}(\omega_1)U)\circ(\boldsymbol{\Delta}_{\mathbf{x}}(\omega_2)U)\circ(\boldsymbol{\Delta}_{\mathbf{y}}(\omega_3)V) - \boldsymbol{\Delta}_{\mathbf{y}}(\omega_3)V\boldsymbol{\Delta}_{\mathbf{x}}(\omega_1)^{\mathsf{T}}\boldsymbol{\Delta}_{\mathbf{x}}(\omega_2)\right]\mathbf{A}^{\mathsf{T}} \tag{B.45}$$

$$N_{14} = -\frac{1}{6}\Big[(\boldsymbol{\Delta}_{\mathbf{x}}(\omega_1)U)\circ(\boldsymbol{\Delta}_{\mathbf{x}}(\omega_2)U)\circ(\boldsymbol{\Delta}_{\mathbf{x}}(\omega_3)U) - \boldsymbol{\Delta}_{\mathbf{x}}(\omega_1)U\boldsymbol{\Delta}_{\mathbf{y}}(\omega_2)^{\mathsf{T}}\boldsymbol{\Delta}_{\mathbf{x}}(\omega_3) - \boldsymbol{\Delta}_{\mathbf{x}}(\omega_2)U\boldsymbol{\Delta}_{\mathbf{x}}(\omega_3)^{\mathsf{T}}\boldsymbol{\Delta}_{\mathbf{x}}(\omega_1) - \boldsymbol{\Delta}_{\mathbf{x}}(\omega_3)U\boldsymbol{\Delta}_{\mathbf{x}}(\omega_1)^{\mathsf{T}}\boldsymbol{\Delta}_{\mathbf{x}}(\omega_2)\Big]\mathbf{A}^{\mathsf{T}} \tag{B.46}$$

$$D = \left[\boldsymbol{\Delta}_{\mathbf{y}}(\omega_1+\omega_2+\omega_3)V\right]\mathbf{A}^{\mathsf{T}} \tag{B.47}$$

References

[1] D.J. Ewins, Modal testing: Theory, practice and application, 2009.
[2] B. Peeters, G. De Roeck, Reference-based stochastic subspace identification for output-only modal analysis, Mech. Syst. Signal Process. 13 (6) (1999) 855–878.
[3] K. Worden, P.L. Green, A machine learning approach to nonlinear modal analysis, Mech. Syst. Signal Process. 84 (2017) 34–53.
[4] R.M. Rosenberg, The normal modes of nonlinear n-degree-of-freedom systems, J. Appl. Mech. 29 (1962) 7–14.
[5] G. Kerschen, M. Peeters, J.C. Golinval, A.F. Vakakis, Nonlinear normal modes, Part I: A useful framework for the structural dynamicist, Mech. Syst. Signal Process. 23 (2009) 170–194.
[6] A.F. Vakakis, L.I. Manevitch, Y.V. Mikhlin, V.N. Pilipchuk, A.A. Zevin, Normal Modes and Localization in Nonlinear Systems, Springer, 2001.
[7] S.W. Shaw, C. Pierre, Normal modes for non-linear vibratory systems, J. Sound Vib. 164 (1) (1993) 85–124.
[8] G. Haller, S. Ponsioen, Nonlinear normal modes and spectral submanifolds: existence, uniqueness and use in model reduction, Nonlinear Dynam. 86 (3) (2016) 1493–1534.
[9] C. Touzé, A. Vizzaccaro, O. Thomas, Model order reduction methods for geometrically nonlinear structures: a review of nonlinear techniques, Nonlinear Dynam. 105 (2) (2021) 1141–1190.
[10] N. Dervilis, T.E. Simpson, D.J. Wagg, K. Worden, Nonlinear modal analysis via non-parametric machine learning tools, Strain 55 (2018) 12297.
[11] J. Decuyper, P. Dreesen, J. Schoukens, M.C. Runacres, K. Tiels, Decoupling multivariate polynomials for nonlinear state-space models, IEEE Control Syst. Lett. 3 (2019).
[12] B.F. Feeny, R. Kappagantu, On the physical interpretation of proper orthogonal modes in vibrations, J. Sound Vib. 211 (1998) 607–616.
[13] M.D. Champneys, K. Worden, N. Dervilis, Nonlinear modal analysis based on complete statistical independence, in: Nonlinear Vibrations, Localisation and Energy Transfer, Vol. 160, 2019, p. 978.
[14] G. Tsialiamanis, M.D. Champneys, N. Dervilis, D.J. Wagg, K. Worden, On the application of generative adversarial networks for nonlinear modal analysis, Mech. Syst. Signal Process. 166 (2022) 108473.
[15] J.Y. Zhu, T. Park, P. Isola, A.A. Efros, Unpaired image-to-image translation using cycle-consistent adversarial networks, in: IEEE International Conference on Computer Vision, 2017.
[16] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in: Advances in Neural Information Processing Systems, Vol. 27, 2014.
[17] G.E. Hinton, R.R. Salakhutdinov, Reducing the dimensionality of data with neural networks, Science 313 (2006) 504–507.

[18] D.P. Kingma, M. Welling, An introduction to variational autoencoders, Found. Trends Mach. Learn. 12 (2019) 307–392.
[19] A. Larsen, S. Søren Kaae, O. Winther, Autoencoding beyond pixels using a learned similarity metric, in: International Conference on Machine Learning, PMLR, 2016, pp. 1558–1566.
[20] P.D. Welch, A direct digital method of power spectrum estimation, IBM J. Res. Dev. 5 (2010) 141–156.
[21] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2014, arXiv preprint arXiv:1412.6980.
[22] C.E. Rasmussen, C.K.I. Williams, Gaussian Processes for Machine Learning, 2006.
[23] T.J. Rogers, Towards Bayesian System Identification: With Application to SHM of Offshore Structures (Ph.D. thesis), University of Sheffield, 2019.
[24] K. Worden, W.E. Becker, T.J. Rogers, E.J. Cross, On the confidence bounds of Gaussian process NARX models and their higher-order frequency response functions, Mech. Syst. Signal Process. 104 (2018) 188–223.
[25] J. Sun, C.H. Lai, X.J. Wu, Particle Swarm Optimisation: Classical and Quantum Perspectives, CRC Press, 2016.
[26] K. Worden, G.R. Tomlinson, Nonlinearity in Structural Dynamics: Detection, Identification and Modelling, IOP, 2001.
[27] V. Volterra, E.T. Whittaker, Theory of Functionals and of Integral and Integro-Differential Equations, Dover publications, 1959.
[28] E. Bedrosian, S.O. Rice, The output properties of Volterra systems (nonlinear systems with memory) driven by harmonic and Gaussian inputs, Proc. IEEE 59 (12) (1971) 1688–1707.
[29] S.A. Billings, H.B. Jamaluddin, S. Chen, Properties of neural networks with applications to modelling non-linear dynamical systems, Internat. J. Control 55 (1) (1992) 193–224.
[30] L.A. Bull, P.A. Gardner, N. Dervilis, K. Worden, Normalising flows and nonlinear normal modes, IFAC-PapersOnLine 54 (7) (2021) 655–660.
[31] T. Simpson, G. Tsialiamanis, N. Dervilis, K. Worden, E. Chatzi, On the use of variational autoencoders for nonlinear modal analysis, in: IMAC 2022, ETH Zurich, Institute of Structural Engineering, 2022.
[32] S.J. Gifford, G.R. Tomlinson, Recent advances in the application of functional series to non-linear structures, J. Sound Vib. 135 (2) (1989) 289–317.