THE UNIVERSITY OF SHEFFIELD

PhD Mechanical Engineering

Challenges in nonlinear structural dynamics: New optimisation perspectives

by

Max David Champneys

November 2022

Prof K. Worden, Prof N. Dervilis

Thesis submitted to the Department of Mechanical Engineering, University of Sheffield for the degree of Doctor of Philosophy

Acknowledgements

First and foremost I would like to thank my supervisors, Professors Keith Worden and Nikos Dervilis. When I first knew that I wanted to pursue a PhD, I knew that I wanted to work with them both, and I was not mistaken. I thank them for their belief in me, for their incredible insight, and for keeping me on track, despite my sometimes capricious research interests. I feel incredible gratitude to have worked within such a supportive and unfalteringly knowledgeable team. Nikos, your visit while I was in hospital buoyed me greatly at a difficult time and I shall not forget it. Keith, your remarkable breadth of knowledge ensured that our meetings have never once been boring, in the pub or otherwise.

As well as my supervisors, I would like to thank Dr Tim Rogers for his mentorship and guidance throughout my academic development. I thank you for the many bugs you have found in my code and for your passion for text-based racing simulation competitions.

If it is said that it takes a village to raise a child, then most certainly it takes a whole research group to grow a PhD. I could not imagine a more supportive, knowledgeable and downright *fun* environment in which to have been raised than the DRG. In particular, I thank Aidan, Chandy, Chris, George, Julian, Marcus, Matt, Matty, Tristan and the many others who have become close friends as well as colleagues. I thank you all for your friendship and for lunchtime conversations on everything from the topology of pasta to the global supply of baked beans.

I also would like to thank the IDC in machining science for their support and encouragement. I thank Fernando, Gloria, Jack, Oscar, Pete and all the others who have made my hours both in and out of the office a pleasure. A special thanks goes to Matt Tipuric for his unwavering tolerance of dance music during many 'power hours' at work.

I must also acknowledge the profound and unfaltering support offered by my parents, siblings, cousins (who really are like brothers), family, friends and pets. I would not be who I am today without you all and I count myself extremely lucky.

Finally to Alice, your love and support means everything to me. You have kept my head level, when so often it has seemed overwhelming.

Abstract

Analysis of structural dynamics is of fundamental importance to countless engineering applications. Analyses in both research and industrial settings have traditionally relied on linear or close to linear approximations of the underlying physics. Perhaps the most pervasive framework, modal analysis, has become the default framework for consideration of linear dynamics. Modern hardware and software solutions have placed linear analysis of structural dynamics squarely in the mainstream. However, as demands for stronger and lighter structures increase, and as advanced manufacturing enables more and more intricate geometries, the assumption of linearity is becoming less and less realistic. This thesis envisages three grand challenges for the treatment of nonlinearity in structural dynamics. These are: nonlinear system identification, exact solutions to nonlinear differential equations, and a nonlinear extension to linear modal analysis. Of these challenges, this thesis presents results pertaining to the latter two.

The first component of this thesis is the consideration of methods that may yield exact solutions to nonlinear differential equations. Here, the task of finding an exact solution is cast as a heuristic search problem. The structure of the search problem is analysed with a view to motivate methods that are predisposed to finding exact solutions. To this end, a novel methodology, the affine regression tree, is proposed. The novel approach is compared against alternatives from the literature in an expansive benchmark study.

Also considered, are nonlinear extensions to linear modal analysis. Historically, several frameworks have been proposed, each of which is able to retain only a subset of the properties of the linear case. It is argued here that retention of the utilities of linear modal analysis should be viewed as the criteria for a practical nonlinear modal decomposition. A promising direction is seen to be the recently-proposed framework of Worden and Green. The approach takes a machine-learning viewpoint that requires statistical independence between the modal coordinates. In this thesis, a robust consideration of the method from several directions is attempted. Results from several analyses demonstrate that statistical-independence and other inductive biases can be sufficient for a meaningful nonlinear modal decomposition, opening the door to a practical, nonlinear extension to modal analysis.

The results in this thesis take small but positive steps towards two pressing challenges facing nonlinear structural dynamics. It is hoped that further work will be able to build upon the results presented here to develop a greater understanding and treatment of nonlinearity in structural dynamics and elsewhere.

Contents

Chapter 1

Introduction

It is perhaps a fact of nature, that all interesting things seem to *change*.[1] Dynamics, or the study of things that change with time, has fascinated humanity since time immemorial. This fascination has lead researchers of posterity to study the motions of the planets, electromagnetism, the mind, and almost everything in-between. Understanding how and why physical quantities change is of fundamental interest.

Of particular importance to engineers is the consideration of the dynamics of *structures*, combinations of distinct elements each with their own dynamic properties. The analysis of structural dynamics pervades engineering and beyond. Although bridges, buildings and wind turbines may appear static, no real structure exists free from dynamic loading. Bridges are buffeted by the wind, buildings are shaken by earthquakes, and wind turbines are rocked by howling wind and sea. Designing, building and monitoring engineering structures necessarily requires an understanding of how they will respond to their loading conditions.

The most eminent type of structural motion is vibration. These periodic motions of structures can be advantageous; musical instruments make use of resonances to amplify pleasing frequencies and energy harvesters scavenge power from ambient motion. However, unplanned or excessive vibrations can have deleterious effects ranging from unpleasant noise pollution to structural collapse.

Vibration in structural dynamics is inescapable. Mathematically, harmonic motion arises directly from the physical laws that govern the dynamics. Forces acting on a structure induce a response via Newton's second law of motion,

$$F = m\ddot{y}$$

[1] That is, given sufficiently broad definitions of 'interesting' and 'change'.

where m is a structure's inertia, and y the displacement, and each over-dot represents a time derivative. Plainly, the application of a load on a structure causes it to accelerate in the direction of the force. In reality, most structures tend to resist being deformed and a restoring force acts to pull the structure back to equilibrium. For linear elasticity, Hooke's law describes this restoring force,

$$F = -ky$$

Equating these forces, results in a second-order homogeneous *differential equation* that describes the dynamics of a simple oscillator. The simple-harmonic equation of motion (EOM) is,

$$m\ddot{y} + ky = 0$$

The solutions to this equation are well known. All non-trivial solutions for physically realistic values of m and k are periodic, the structure will oscillate in free-vibration with a fixed period.

As the complexity of engineering structures increases, so too does the complexity of analysis. However, even as complexity grows, the objectives of the structural dynamicist remain constant. The work of the engineer can be broadly summarised by three tasks.[2]

- Identification: Specification of the equations of motion, from data or otherwise.
- Prediction: Time series solutions to the equations of motion for new loading or environmental conditions.
- Simplification: Decoupling of the equations of motion to facilitate simpler analysis.

In modern engineering, the role of structural dynamics is more important than ever. The design of increasingly-complex structures in order to fulfil ever-more demanding requirements has been a driving factor in human progress. However, these advancements come at a cost. Materials and geometries are increasingly being pushed into more demanding regimes, wherein a straightforward analysis starts to break down.

[2]Other important tasks remain such as input reconstruction and parameter identification but these rely on identification of a good model and a method of producing predictions.

1.1 The linear elephant and the nonlinear zoo

The fundamental laws that govern simple structural dynamics are linear equations. Newton's second law, viscous damping and Hooke's law of elastic deformation all exhibit this property. The presence of linearity in the equations of motion give rise to two properties *homogeneity* and *additivity*. Collectively these properties are known as the *principle of superposition*. For some linear dynamic system S,

$$y = S(x_1 + ax_2) = S(x_1) + aS(x_2)$$

The response from multiple inputs can be reconstructed via the sum of the individual contributions. Structural dynamic analysis of linear systems is by now an extremely advanced field. Since the 1970s, modal analysis [1], has emerged as *the* framework for conducting linear analysis and has enjoyed enormous popularity. In Chapter 2 a detailed account of the method will be introduced, but for now it will be sufficient to say that the tasks of identification, prediction and simplification all yield to linear modal analysis. Unfortunately, modal analysis is only valid for linear structures and cannot address nonlinearity.

To describe the dynamics of a structure as *nonlinear* is a somewhat negative definition of reality.[3] No single text could possibly describe all the myriad ways in which the conditions of linearity breakdown.

The effect of nonlinearity on structural dynamics ranges from the inconvenient to the profound. For small nonlinear contributions, the breakdown of superposition leads to an amplitude dependence in the resonance frequencies of harmonic systems. For more significant nonlinearities, diverse phenomena are observed, including jump-phenomena [2], bifurcations [3] and chaos [4] (extreme sensitivity to initial conditions). In some cases, nonlinear effects can even be harnessed to produce smaller and more efficient sensors [5] or energy harvesters [6].

In engineering, nonlinearity can enter the dynamics from many directions. A common loss of linearity arises via the material properties. As the demand for ever lighter and stronger materials drives material selection in engineering, the presence of nonlinearity is becoming more and more difficult to ignore. A common example is the elastic limit. When loaded, many materials will initially deform proportionally (i.e linearly). However, past a certain level of deflection (termed the elastic limit), the stiffness begins to depend

[3] "Using a term like nonlinear science is like referring to the bulk of zoology as the study of non-elephant animals." —Stanislaw Ulam.

nonlinearly on the level of forcing, as changes are made to the atomic structure of the material.

Another type of material nonlinearity is hysteresis. Hysteretic nonlinearities occur because of irreversible thermodynamic changes in the material and result in a lag in the relationship between loading and response. Because of this 'memory' effect in the dynamics, hysteretic systems are particularly challenging to model [7].

As well as via material properties, linear assumptions can also be violated by geometry. A classic example is the fixed-fixed beam. This idealised structure exhibits geometric nonlinearity for deflections on the order of only the thickness of the beam at the midpoint [2]. Another common example is the simple pendulum, which is patently nonlinear for only moderate angular displacements.[4]

Driven by computer-aided design and advanced manufacturing techniques such as additive manufacturing and multi-axis machining, the geometries available to the design engineer are today more complex than they have ever been. This complexity brings advantages in terms of performance, but also leads inevitably to increased nonlinearity.

As well as materials and geometries, the interfaces between components are another common source of nonlinearity. Looseness in bolted joints gives rise to multi-linear stiffness regimes that violate superposition. Common conditions such as friction and impact lead to discontinuities in the restoring forces and thus to nonlinearity in the dynamics.

As well as the nonlinearities presented above, linear analysis is similarly incapable of dealing with other complex yet common phenomena such as buckling [8, 9], rocking [10], and structural failure [11].

1.2 Motivation: Three challenges in nonlinear structural dynamics

A thorough assessment of structural dynamics is critical to the development of current and future engineering structures and vehicles; from larger wind turbines,[5] to future low-carbon modes of transportation. For the author, it is impossible to imagine that the new materials, geometries and assemblies that will enable these advancements will result in anything other than an increase in structural nonlinearity.

Nonlinearity breaks the analysis of linear structural dynamics in all three places.

[4] The linear assumption breaks down for restoring force from gravity $F = mg\sin(\theta) \approx mg\theta$.
[5] The capacity of a wind turbine can be related directly to its blade length.

- Identification: Nonlinear phenomena such as amplitude dependence make identification challenging.
- Prediction: Exact solutions are unavailable for nonlinear differential equations in general.
- Simplification: Decoupling of the equations of motion is not possible in general because of the breakdown of the principle of superposition.

These analyses are the key tasks that an engineer must complete in order to design, control, predict and monitor structures from a dynamical point of view. It is asserted here, that addressing each of these undertakings in the nonlinear case constitutes a *grand challenge* in structural dynamics research.

The remainder of this section will introduce three grand challenges for nonlinear structural dynamics as the motivation for the results of this thesis. The following chapter will proceed by analysing each challenge in detail and so only an overview is included here.

Challenge I: Nonlinear system identification

Central to scientific analysis is the connection of input actions to output results. Understanding (and describing mathematically), the complex relationship between cause and effect is perhaps the most fundamental building block of engineering research.

In dynamic systems, the connection between time-varying input and time-varying output data is the problem of *system identification*. The structural dynamics of linear systems[6] can be cogently understood through the lens of modal analysis [1]. However, the zoo of nonlinearity is resistant to a single all-encompassing approach.

The core challenge of nonlinear system identification (NLSI), is that nonlinear phenomena are incommensurable compared to the linear case. The diverse phenomena that a nonlinear system can exhibit does not lend itself to be encompassed by a single, physically meaningful description.

NLSI in structural dynamics has resulted in significant research effort in recent decades, reviews of the state of the art can be found in [12, 13]. Despite significant progress, the challenge is not solved. Major avenues for amelioration remain.

The first area in which NLSI methods stand to be improved is interpretability. Modern machine-learning techniques such as neural networks [14] are able to learn complex

[6] Subject to some restrictions as shall be seen in the following chapter.

dynamic relationships and give excellent performance in terms of error metrics. However, the neural network does not afford the dynamicist any insight as to the nature of the dynamics that have been learnt.

Another direction in which NLSI stands to make progress is the treatment of noise and uncertainty. Data from real structures is inescapably corrupted by noise and errors. In the presence of high levels of noise, the identification task is particularly challenging and great care must be taken to avoid over-fitting and bias. Faced with noise and uncertainty, the engineer must be upfront about the extent of their confidence.

Of the three challenges, NLSI is perhaps the most well-treated at the time of writing. Although significant work remains to be completed, research over several decades has generated a toolbox of methods for analysis.

Challenge II: Exact solutions to nonlinear differential equations

If one were tasked with enumerating the most pressing challenges in modern science, a better *understanding* of nonlinear phenomena would feature among the most prevalent. The mathematical object that encodes the relationships between physics and these phenomena is usually the differential equation. These equations describe analytically, the evolution of quantities in time and space.[7] Specifying and solving differential equations is the kernel problem in everything from the prediction of weather and stock market fluctuations to the motions of the planets and particles. Indeed, one of the prestigious Clay Institute Millennium Prizes [15] is concerned with solutions to a system of nonlinear differential equations (the acclaimed Naiver-Stokes equations [16]).

Linear dynamics are governed by differential equations that can be solved exactly and in closed-form.[8] These closed-form solutions offer significant interpretability into the nature of the dynamics. If the engineer wishes to investigate the effect of a particular parameter on the overall response of the structure, the effect is directly discernible from the closed-form solution.

Nonlinear differential equations can be found everywhere in engineering. At sufficiently small (or large) scales, almost everything ceases to obey a linear relationship. Despite their plurality, nonlinear differential equations are almost exclusively without closed-form solution. It is enticing to imagine the potential insight into nonlinear phenomena that might be afforded by nonlinear solutions to differential equations. Even an explicitly parametric solution to a nonlinear differential equation may afford key insights into the behaviour in the general case.

[7] And indeed many other things besides.
[8] For constant-coefficients and some cases of varying coefficients.

Challenge II is perhaps the most blue-sky of those considered in this thesis. It is far from certain that such exact, closed-form solutions even exist in general. Such solutions have thus far evaded researchers employing traditional techniques. However, it is interesting to imagine how recently-developed heuristic optimisation methods might offer a new perspective.

Challenge III: A nonlinear extension to linear modal analysis

As has been stated many times already in this introduction, linear modal analysis is pervasively useful for the analysis and interpretation of linear structural dynamics. Frequency-based methods [17] and hardware[9] have proliferated to the extent that the method is practically standard in an industrial setting.

Modal analysis decomposes the structural dynamics into independent single degree-of-freedom systems from which the overall response can be calculated by the principal of superposition. Better yet, this decomposition can be expressed in terms of physically-interpretable properties of the dynamics; the modeshapes, natural frequencies and damping ratios.

Nonlinear systems do not obey the principal of superposition and cannot be decoupled by a linear transformation, linear modal analysis cannot be applied. The traditional approach to addressing this shortcoming was to *linearise*, to consider only small amplitudes of vibration for which the structure could be considered approximately linear. However, as stated already, the increasing demands of modern engineering structures mean that the nonlinearity can no-longer be ignored. As nonlinearity increases, what will happen to the modal testing that is so prevalent in industrial applications?

There is an urgent need to extend the utilities of modal analysis to the nonlinear case. Indeed, several frameworks [18–20] have been proposed over the last fifty years that are each only able to preserve a subset of the properties of the linear case.

The challenge for structural dynamicists going forward will be to find frameworks for nonlinear normal modes (NNM), that can be applied in a practical industrial setting, that are robust to noise, and that capture as much utility as possible from the linear analysis.

[9] The Leuven Measurement Systems ('LMS box') has become a standard commercial solution for experimental testing.

1.3 Objectives of this thesis

These grand challenges are not intended to constitute an exhaustive list. Other important areas of nonlinear structural dynamics research exist; including multi-physics interactions, nonlinear control and a more thorough treatment of damping in real-world structures. However, the challenges presented here arise directly from the fundamental tasks of the structural dynamist.

No thesis (or book, or library) could be sufficient to address these grand challenges completely or in isolation. It is inevitable that a more narrow view must be taken.

In this thesis, the objective will be to present work pertaining to challenges II and III. It will not be expected that these challenges will be solved (how could it?), but the intention will be to take meaningful steps towards addressing them.

Engineering research has recently seen something of a paradigm shift [21], away from small-scale laboratory testing and modelling, and into a more data-driven view of analysis. Techniques developed in statistics and computer science (collectively *machine learning*), have enabled engineers to make rapid progress in previously-intractable problems. It is the viewpoint of the author that these techniques offer an important opportunity to address longstanding challenges in engineering. It will therefore be the ethos of this thesis to attempt to approach the challenges above from a machine learning point of view.

In particular, the objectives of this thesis will be:

- To formulate the problem of finding exact solutions to nonlinear ODE problems as a search problem.
- To seek methods of specifying such solutions in a *heuristic* manner.
- To motivate methods that might find solutions to as-yet unsolved nonlinear differential equations that describe nonlinear structural dynamics.
- To establish a criteria for a useful nonlinear extension to modal analysis.
- To examine frameworks for nonlinear modal analysis in the light of these criteria.
- To investigate the recently-proposed statistically-independent framework for NNMs, both qualitatively and quantitatively against the criteria.

As well as these major objectives, some ancillary results are also presented. The novel contributions of this thesis are summarised in Chapter 10.

1.4 Thesis structure

For the reader's convenience, a brief summary of each of the following chapters is included here.

Chapter 2: Presents a summary of the relevant literature and background pertaining to each of the grand challenges identified above. The chapter concludes with the specification of two research questions that the remainder of the thesis will attempt to address.

Chapter 3: The first of two chapters that considers heuristic approaches to the exact solutions of nonlinear differential equations. The task of finding an exact solution is recast as a search problem with a view to motivate techniques that might be tailored towards exact solutions. Also presented, is a novel benchmark suite of ODE problems that has several advantages over those seen in the literature.

Chapter 4: Presents a novel heuristic methodology for exact solutions to ODE problems, the *affine regression tree*. The new methodology is motivated by the results of the previous chapter and is compared to results from an expression tree method and to results from the literature. The methods are compared on two benchmark sets of ODE problems, one from the literature and also the new benchmark suite proposed in the previous chapter.

Chapter 5: The first chapter concerned with a practical nonlinear modal analysis. In this chapter, the recently-proposed statistically-independent NNM framework is introduced as a promising direction for investigation. The criteria for a practical framework for NNMs are reviewed. A number of metrics are proposed for the assessment of these criteria.

Chapter 6: Approaches the statistically-independent NNMs from a theoretical standpoint. The Fokker-Plank-Kolmogorov (FPK) equation is used to generate an exact modal decomposition in the limited single-degree-of-freedom (SDOF) case for a Duffing oscillator. An extension to the multiple-degree-of-freedom (MDOF) case is envisaged as a nonlinear optimisation problem.

Chapter 7: Constructs NNMs from the statistically-independent framework by two methods; a multinomial transformation [22] and a cycle-GAN [23]. The utility of each of the modal decompositions is assessed in a qualitative manner.

Chapter 8: Presents results pertaining to the specification of higher-order frequency-response functions (HFRFs). Convenient closed form expressions are derived for an entire class of NLSI models that are shown to have several convenient properties including

the ability to interpret black-box models and act as an extremely stringent test of model fit. HFRFs are motivated as a convenient way to assess the statistically-independent NNMs generated in the previous chapter in a qualitative manner.

Chapter 9: Considers at last, the statistically-independent framework for nonlinear modal analysis with regard to the criteria of Chapter 2. The principal angle of analysis is to perform NLSI on the modal dynamics. For each of the criteria, several observations are presented, demonstrating the utility of the framework.

Chapter 10: Some concluding remarks are offered as well as a summary of the novel contributions of this thesis. Finally, some directions for further investigation are described.

Chapter 2

Background

In Chapter 1, three grand challenges in nonlinear structural dynamics were identified and their impact on modern engineering was discussed. This chapter will now present the background that pertains to each of the challenges. The aim here, will be to introduce sufficient background to arrive in each case at the state-of-the-art. This chapter will conclude by formulating two research questions that the remainder of the thesis will be occupied in addressing.

2.1 Nonlinear system identification

Access to mathematical models of dynamical processes is a valuable endeavour that enables the engineer to interpret and predict their future behaviour. These processes are often given the extremely general term *system*, and approaches that attempt to find mathematical descriptions of them as tools for *system identification* (SI). In the world of dynamics, a system is simply a functional relationship between some time-varying inputs and outputs. In its most general form then, *system identification* is simply the task of specifying a functional map F[1], that describes the relationship between the time-varying input variables $\mathbf{x}(t)$ and the time-varying output variables $\mathbf{y}(t)$.

$$F : \mathbf{x}(t) \to \mathbf{y}(t) \tag{2.1}$$

In structural dynamics, the output variables $\mathbf{y}$ could be any one of the physical quantities of the system. For example, the forces, displacements, velocities and accelerations might all be considered. The second-order nature of Newton's second law means that

[1] In this thesis the terms *function* and *functional* will be used strictly. The former shall refer to static maps and the latter shall refer to dynamic maps i.e differential equations.

for structural dynamic analysis it is often sufficient to consider only one of these, and convert to the others by numerical or physical reasoning. Owing to the prevalence of low-cost piezo-electric accelerometers, most practical dynamic testing is conducted via collection of acceleration data [1]. Modern laser and scanning-laser vibrometers provide an exception to this trend and can provide accurate measurements of velocities and displacements without the integration error associated with the conversion from acceleration data. However, laser-vibrometers add both cost and complexity and are therefore often unsuitable for large-scale deployment. A yet more general formulation is the state-space model [24], a common formulation seen in the control community. For a state-space model, the functional F connects the observed inputs to a number of state variables that are not observed directly. The system outputs are then recovered by the application of a static observation function.

When the dynamics of the functional F are linear, the task of identification is well understood[2], and several powerful techniques are available depending on the application. In structural dynamics, linear modal analysis (LMA), has emerged as the lingua-franca for system identification of linear dynamics. LMA has a number of highly-useful features that provide both a compact, decoupled representation of the dynamics and physical insight in terms of natural frequencies and modeshapes. A full discussion of LMA is reserved for a following section. However, the reader is directed to the comprehensive monograph [1] for a full treatment of linear modal analysis in an identification context.

For a nonlinear functional F, identification is more challenging. So-called nonlinear system identification (NLSI) is a highly-active area of investigation that has attracted much attention in recent decades.

Although NLSI is a significant challenge in structural dynamics, it is not a principal aim of the current thesis and so the author feels that an extravagant treatment of the relevant literature lies outside the remit of this chapter. For a more complete view of the field, extensive surveys of the literature can be found over several decades in the review papers [12, 13, 25]. Instead, this section will provide an elucidatory categorisation of NLSI methods and introduce mathematically some models that will be required for analysis later in this thesis.

2.1.1 The axes of NLSI

Categorisation of NLSI methods is not a straightforward task. The very general nature of the problem (to find a functional mapping F between some input and output data), leads

[2] This is not to say solved, as meaningful work remains to be completed in the treatment of output-only identification, uncertainty quantification, treatment of sensor error and other areas.

FIGURE 2.1: The axes of NLSI methods.

naturally to a very general suite of techniques that can be applied. Indeed, researchers from control [26–28], machine learning [29], mechanical engineering [30–32], biology [33] and many others have developed and applied NLSI methods, and a growing interest has been shown in 'cross-pollinating' research from these fields [34].

Evidently, NLSI encompasses a broad church of techniques and tools. In place of a traditional hierarchical categorisation of NLSI methods it is perhaps more illustrative to consider only the 'axes'. In Figure 2.1 a number of attributes are considered for the categorisation of NLSI methods. It should be noted that these attributes describe the models and techniques used for conducting identification of a nonlinear system rather than the properties of the structure of interest.

As alluded to above, perhaps the most fundamental distinction that can be made between system identification approaches is that made between linear and nonlinear models. Although this seems a dichotomy at first (clearly linear models can only represent linear dynamics), techniques such as best-linear approximation (BLA) [35], quasi-linear methods [36] and Koopman operators [37] (infinite-dimensional linear expansions of nonlinear functionals) serve to blur the line from an identification point of view.

Perhaps the next biggest distinction that can be drawn between NLSI techniques is time-domain resolution. Continuous methods seek to infer a representation of the nonlinear functional F that can be evaluated to arbitrary precision. Interest in continuous-time identification methods spans several decades [38]. Methods of this type include techniques based on wavelet transforms [39, 40], whereby a finite-basis of kernels in the frequency domain are used to build a representation of the nonlinear functional. More recently, neural-network models have been proposed to approximate the functional directly [14] in a time-continuous fashion. Methods based on the Volterra series [41] can also be considered time-continuous as they give access to a functional series representation of F. Indeed, any method that specifies an analytic continuous representation for F (parameter-estimation, linear modal analysis) can likewise be considered time-continuous.

In [12], the process of NLSI is broken down into three phases:

1. Detection of nonlinearity.
2. Characterisation of nonlinearity.
3. Parameter estimation.

This type of analysis is very much rooted in gaining a physically-meaningful representation of the dynamics (linear and otherwise), that are present in the system. So-called *white-box* models rely on finding analytical descriptions of the types of nonlinearity (friction, hysteresis, impact) that are present in structural dynamics from engineering or physical insight. Conducting NLSI in this fashion permits access to the model structure of F directly. Thus, the NLSI problem is reduced to one of parameter estimation, for which a plethora of applications can be found in the literature using tools such as evolutionary optimisation [32, 42–44], particle-filters [45, 46] and several others.

In contrast to white-box models, *Black-box* approaches to NLSI make no attempt to preserve physical insight in favour of better model performance. Classical model formulations including the linear auto-regressive models with exogeneous inputs (ARX), the nonlinear extension (NARX) and the nonlinear moving-average extension (NARMAX) are examples of black-box techniques. For a comprehensive reference on the subject, the reader is directed to [30]. Developments in machine learning have lead to an explosion of black-box models including neural-network NARX (NN-NARX) [47] and kernel-NARX models (of which an important member is the Gaussian process NARX (GP-NARX) model [21, 48, 49]).

As well as these two extremities, there is considerable interest in so-called *grey-box* approaches in NLSI [50, 51]. The ethos here is to marry the powerful representation performance of black-box models with the physical insight of white-box models. A full treatment of the many shades[3] of grey that have been proposed is beyond the scope of the current chapter; however, the interested reader is directed to [52] for an instructive reference.

Another important delineation between methods is whether or not the model used is *parametric*. A parametric model requires the specification of a number of parameters, whereas in a non-parametric model, the model is fixed entirely by the observed data. In practice, most NLSI approaches rely on some explicit parametrisation of either physical quantities or black-box parameters. However some methods that permit a least squares

[3] Groans from the audience.

TABLE 2.1: A categorisation of some common NLSI methods.

Model class	Linearity	Resolution	Box	Prob.	Param.
LMA	Linear	Continuous	White	No	Yes
SSI	Linear	Continuous	White	Yes	Yes
BLA	Linear	Discrete	Black	No	No
ARX	Linear	Discrete	Black	No	No
White-box	Nonlinear	Continuous	White	No	Yes
NARX	Nonlinear	Discrete	Black	No	Yes
NARMAX	Nonlinear	Discrete	Black	No	Yes
NN-NARX	Nonlinear	Discrete	Black	No	Yes
NN-WT	Nonlinear	Continuous	Black	No	Yes
NN-ODE	Nonlinear	Continuous	Black	No	Yes
Kernel-NARX	Nonlinear	Discrete	Black	No	Semi
GP-NARX	Nonlinear	Discrete	Black	Yes	Semi
SMC2	Nonlinear	Discrete	Black	Yes	Yes

solution (such as ARX, BLA or the approach of Bai [53, 54]) can be considered non-parametric.

Driven by a desire to quantify uncertainty in mathematical modelling in general, the field of NLSI has seen a great deal of interest in probabilistic methods. Probabilistic interpretations of many classical algorithms have been proposed. Bayesian approaches to NLSI have likewise gained a lot of traction including reference-based subspace identification (SSI) [55] for linear modal dynamics and powerful sampling-based approaches such as Stochastic Monte-Carlo squared (SMC2) [56]. The rise of Gaussian-process regression [57] techniques has also been felt in the NLSI community in the form of GP-NARX models [21, 48, 49] that combine the flexibility of kernel methods with the uncertainty quantification of the Gaussian process.

Table 2.1 positions several common NLSI methods on the axes of Figure 2.1. The entries listed here are far from exhaustive (indeed one could imagine many more categories and subcategories), but hopefully serve to give an indication as to the breadth of tools that are available.

2.1.2 Some NLSI models used in this thesis

It is useful at this stage to introduce some NLSI models that will form part of the analyses presented later in this thesis. Both linear and nonlinear models will be used and so this section begins with a description of the ARX model.

ARX models are linear, discrete-time black-box models that predict the value of the output at a future time point based on observations from the input and output at previous time points. The explicit form of the model is,

$$y_t = \sum_{i=1}^{n_y} a_i y_{t-i} + \sum_{i=0}^{n_x-1} b_i x_{t-i} \tag{2.2}$$

where n_x and n_y refer to the number of lagged inputs and outputs respectively (i.e the amount of 'memory' in the model) and $\mathbf{a} = [a_1, a_2, \ldots, a_{ny}]$ and $\mathbf{b} = [b_1, b_2, b_{nx}]$ are vectors of model parameters. For a given lag structure, ARX models also permit a vectorised formulation,

$$\mathbf{y} = H \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix} \tag{2.3}$$

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

$$H = \begin{bmatrix} y_{t-1} & y_{t-2} & \cdots & y_{t-n_y} & x_t & x_{t-1} & x_{t-2} & \cdots & x_{t-n_x+1} \\ y_{t-2} & y_{t-3} & \cdots & y_{t-n_y-1} & x_{t-1} & x_{t-2} & x_{t-3} & \cdots & x_{t-n_x} \\ y_{t-3} & y_{t-4} & \cdots & y_{t-n_y-2} & x_{t-2} & x_{t-3} & x_{t-4} & \cdots & x_{t-n_x-1} \\ \vdots & & & & \vdots & & & & \\ y_{t-N+p} & & & & x_{t-N+p-1} & & & & \end{bmatrix} \tag{2.4}$$

where N is the total number of temporal points in the training data and $p = \max(n_y, n_x)$. ARX models are completely defined by the model weights a_i and b_i which can be simply fitted to input output data by first constructing the Hankel matrix and then finding the least-squares solution of the linear equation in (2.3).

Extending the ARX class to nonlinear models gives rise to the NARX formulation. Rather than a linear expansion of the lagged variables, a static map is selected that predicts the value of the output at the next time step.

$$y_t = \Phi(y_{t-1}, y_{t-2}, \ldots, y_{t-n_y}, x_t, x_{t-1}, x_{t-2}, \ldots, x_{t-n_x+1}) \tag{2.5}$$

where Φ 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,

$$\mathbf{y} = \Phi(H) \tag{2.6}$$

In practice, Φ can be selected based on domain knowledge or by a sufficiently flexible set of basis functions (polynomials or radial basis functions (RBFs) are common choices). Training of NARX models can proceed in a number of ways depending on the choice of the nonlinearity. For choices of Φ that are linear in the parameters (i.e polynomials, RBF, etc), the vectorised formulation can be factorised as,

$$\mathbf{y} = \alpha^{\mathsf{T}}\Phi(H) \tag{2.7}$$

where α is a vector of model weights that can be solved by a least-squares procedure[4].

A highly-flexible specific case of the NARX model is the GP-NARX model. The GP-NARX formulation assumes that the nonlinear map Φ 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[5], and the non-parametric form permits a high degree of model flexibility. A full review of Gaussian process and GP-NARX literature is beyond the scope of this paper and so only a description of the approach is included here; for excellent references on the topics, the reader is directed to [21, 49, 57].

A Gaussian process (GP) is 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[6],

$$y_i = f(\mathbf{x}) + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0, \sigma_n^2) \tag{2.8}$$

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,

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x})) \tag{2.9}$$

where k is a positive semi-definite covariance kernel $k(\mathbf{x}, \cdot) \rightarrow \mathbb{R}^+$ and $m(\mathbf{x}) \rightarrow \mathbb{R}$ is the mean function. Several choices for both of these functions are applicable and their selection is often driven by domain knowledge [57]. In the absence of any prior

[4] It is important to note that any hyperparameters in the nonlinear function (model order, length-scales, exponents) must be optimised separately.

[5] Uncertainty quantification for GP-NARX models is straightforward for *one-step-ahead* prediction, for other prediction types the situation is (to put it lightly) more complex. The reader is directed to [21] for a discussion.

[6] The Gaussian process model by itself is a static map and cannot represent functional objects. In the notation that follows, inputs and outputs to the static map are indexed with i.

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 real beauty of the GP formulation is that it 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 $\{X^*, \mathbf{y}^*\}$.

$$\begin{pmatrix} \mathbf{y} \\ \mathbf{y}^* \end{pmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} K(X,X) + \sigma_n^2 I & K(X, X^*) \\ K(X^*, X) & K(X^*, X^*) + \sigma_n^2 I \end{bmatrix}\right) \tag{2.10}$$

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,

$$\mathbf{y}^* \sim \mathcal{N}(\mathbb{E}[\mathbf{y}*], \mathbb{V}[\mathbf{y}*]) \tag{2.11}$$

where the posterior mean prediction is given by,

$$\mathbb{E}[\mathbf{y}*] = K(X^*, X)\left[K(X,X) + \sigma_n^2 I\right]^{-1}\mathbf{y} \tag{2.12}$$

and the posterior variance is given by,

$$\mathbb{V}[\mathbf{y}*] = K(X^*, X^*) - K(X^*, X)\left[K(X,X) + \sigma_n^2 I\right]^{-1} K(X, X^*) + \sigma_n^2 \tag{2.13}$$

Substituting the zero-mean function and the posterior variance into the formulation of a NARX model in (2.6) 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,

$${\mathbf{y}^*}_{\text{OSA}} = \mathbb{E}[y*] = K(H^*, H)\left[K(H,H) + \sigma_n^2 I\right]^{-1}\mathbf{y} \tag{2.14}$$

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. 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,

$$y_t^* = k\left(\mathbf{h}_t^*, H\right)\left[K(H,H) + \sigma_n^2 I\right]^{-1}\mathbf{y} \tag{2.15}$$

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

$$\mathbf{h}_t^* = \begin{bmatrix} \hat{y}_{t-1}^* & \hat{y}_{t-2}^* & \dots & \hat{y}_{t-n_y}^* & x_t & x_{t-1} & x_{t-2} & \dots & x_{t-n_x+1} \end{bmatrix} \tag{2.16}$$

where the $\hat{y}^*$ are model predictions from previous time steps that are fed back into the model.

2.2 Solutions to differential equations

When the form of the functional F is known, the problem of NLSI is reduced to one of parameter estimation. If one is able to identify the parameters (by physical insight or otherwise), the equations of motion (EOMs) can be recovered in closed form. It may at this stage be tempting to feel that the hard part is over. However, significant analytical challenges remain.

Mathematically speaking, the equations of motion of a nonlinear structure in continuous time are simply differential equations. Encoded within them is all the required information necessary to make predictions and understand the dynamics. How then can this pertinent information be extracted? In order to progress from the EOM to a time domain prediction, the differential equation must first be *solved*.

As shall be seen, for linear dynamics (subject to proportional or Rayleigh-type damping), this can be done in closed-form with little difficulty. However, when the dynamics are nonlinear, this represents a significant challenge. For the nonlinear EOMs that are commonly seen in structural dynamics, almost no closed-form solutions have been discovered.

2.2.1 Differential equations disambiguation

Since the early descriptions of dynamic systems by Leibniz and Newton, it has been apparent that not all systems are equivalent. The behaviour of structural dynamic systems alone includes a highly diverse set of observable phenomena including periodicity,

decay, resonance and chaos. Given the wide variety of properties, it is unsurprising that significant effort has been directed towards classification of differential equations.

The most immediate classification available to the researcher is based on the structure of the differential equation itself. In order to facilitate the discussion that follows (and as a useful nomenclature for the reader), some major definitions are included here.

Any general differential equation can, without loss of generality, be transformed into its implicit form,

$$\Psi = 0 \tag{2.17}$$

Where Ψ is a static function that maps a number of independent variables x_i, unknown functions of those variables f_i (the dependent variables), and the derivatives and partial derivatives of the f_i to a real value[7].

The highest order derivative r found in the arguments of Ψ is known as the *order* of the differential equation. More than one such equation Ψ over the same set of independent and dependent variables (coupled or otherwise) is termed a *system* of d differential equations,

$$\begin{aligned} \Psi_1 &= 0 \\ \Psi_2 &= 0 \\ &\vdots \\ \Psi_d &= 0 \end{aligned} \tag{2.18}$$

A differential equation is *ordinary* (an ODE) if the terms of Ψ contain only the derivatives of a single independent variable x,

$$\Psi(x; f(x), f^{(1)}(x), \ldots, f^{(r)}(x)) = 0 \tag{2.19}$$

where superscripted indices refer to derivative orders. Or equivalently for systems of ordinary differential equations (SODE),

[7]Here it is assumed that the differential equation is defined in continuous time. For a translation to discrete time dynamics, the derivatives and partial derivatives can be replaced by lagged values of the of the dependent variables.

$$\begin{aligned}
\Psi_1\left(x, \mathbf{f}(x), \mathbf{f}^{(1)}(x), \ldots, \mathbf{f}^{(r)}(x)\right) &= 0 \\
\Psi_2\left(x, \mathbf{f}(x), \mathbf{f}^{(1)}(x), \ldots, \mathbf{f}^{(r)}(x)\right) &= 0 \\
&\vdots \\
\Psi_d\left(x, \mathbf{f}(x), \mathbf{f}^{(1)}(x), \ldots, \mathbf{f}^{(r)}(x)\right) &= 0
\end{aligned} \tag{2.20}$$

where vector quantities are given boldface notation. Conversely, a differential equation is *partial* (a PDE), if the unknown function is in several independent variables $f(x_1, \ldots x_n)$ and their partial derivatives,

$$\Psi\left(x_1, \ldots x_n; f(x_1, \ldots x_n), \frac{\partial f}{\partial x_1}, \ldots \frac{\partial f}{\partial x_n}; \frac{\partial^2 f}{\partial x_1 \partial x_1}, \cdots \frac{\partial^2 f}{\partial x_1 \partial x_n}; \ldots\right) = 0 \tag{2.21}$$

As before, a system of PDEs (SPDE) is defined as a number of such equations over the same basis of independent and dependent variables.

A very fundamental concept in the categorisation of differential equations is again *linearity*. A linear differential equation (be it ODE or PDE) is one that can be expressed as a purely linear combination of the derivatives of the independent variables,

$$\Psi = b(x) + \sum_{k=0}^{r} a_k(x) f^{(k)}(x) = 0 \tag{2.22}$$

Where the $a(x), b(x)$ are functions of x. In the case where all $a_k(x)$ are constant, the equation is referred to as having constant parameters or constant coefficients. This property generalises to PDEs and SPDEs where the sum includes the partial derivative terms. Any differential equation that does not have this property is nonlinear (NLODE, NLPDE, etc.).

A *homogenous* differential equation, contains for every term an unknown function or one of its derivatives. for structural systems, homogenous problems arise in the absence of external excitation i.e free-vibration. Equations without this property are termed *inhomogeneous*.

Alongside the structure of the differential equation, there are often restrictions placed on the form of the dependent variables. These restrictions arise in physically interesting problems because of the geometric and physical context being described. There are conventionally two classes of problem.

TABLE 2.2: Summary of properties of differential equations. Notation is in reference to (2.23).

Property	Description	Has property	Else
System	$d > 1$	SDE	DE
Ordinary	$\frac{\partial^{i+j} f}{\partial x_i \partial x_j} = 0$	ODE	PDE
Linear	$\Psi = b(x) + \sum_{i=0}^{r} a_i(x) f^{(i)}(x) = 0$	Linear DE	NLDE
Homogenous	$g(x_1, \dots x_n) = 0$	Homogenous	Inhomogeneous
Parameter type	$a_i(x)$ constant	Constant	Variable
Problem type	$\Psi = 0, \quad f^{(i)}(x_0) = u_i$	IVP	BVP

An *initial-value problem* (IVP) fixes the values of the dependent variables f_i at some point in their domain. Solution of the IVP requires specification of a functional form of the dependent variables that satisfies both the differential equation and the initial condition. IVPs arise most frequently in physical time-varying problems where the state of a system at some initial time is known[8]. Examples of typical IVPs include vibration analysis, orbital mechanics and the motion of projectiles.

A *boundary-value-problem* (BVP) specifies the values of the dependent variables at the boundary of some interval of interest. A solution to the BVP is defined as above, where the values of the f_i are given functional forms that fulfil both the differential equation and the constraints. BVPs arise most often in physical PDE problems whereby a time-varying quantity defined on a continuous domain is studied. For example, problems in heat transfer, fluid dynamics and wave propagation are most often posed as BVPs.

Because most of the properties described thus far are independent, it is difficult to summarise them within a hierarchical figure. One can more or less choose any properties from above and describe a meaningful subset of differential equations. Instead, Table 2.2 provides a summary of abbreviations. Descriptions of properties in the table refer to the general differential equation,

$$\Psi_i \left(f(x_1, \dots x_n); \frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_n}; \frac{\partial^2 f}{\partial x_1 \partial x_1}, \cdots \frac{\partial^2 f}{\partial x_1 \partial x_n}; \dots \right) = g(x_1, \dots x_n), \quad i = 1, \dots, d \tag{2.23}$$

Ever more general and granular classifications can be made to include objects such as algebraic (including integrals) and stochastic (non-deterministic terms) differential equations. The author feels that solution of nonlinear ODEs is itself a significantly challenging task and so analysis will proceed for ODE problems only.

[8] This itself could be considered a boundary.

Existence and uniqueness of solutions

Given the physical implications of solutions to differential equations it is perhaps not surprising that much interest has been directed towards determining exactly when a solution exists and whether or not it is unique. The answer to this question affords significant philosophical insight. Non-unique solutions to physically expressive problems carries implications for determinism, the non-existence of solutions may inform us that our models might not be rich enough to describe the dynamics that can be observed.

By far the majority of differential equations that yet possess exact solutions are linear. As luck would have it, many of the physical laws that govern the universe can be adequately described by linear differential equations, Newton's motion of solids, Faraday's induction of electric fields and Fourier's heat flow are notable examples[9]. Concrete understanding of the linear differential equation has provided the fundamental framework for much of modern analysis into structural dynamics and control systems. Linear analysis has become so pervasive that even in the presence of known nonlinear phenomena, it is often the approach of the engineer to *linearise* [28, 35, 37]; to try to find the closest approximation to a linear system such that the analysis techniques to which they are familiar remain available.

The resistance to addressing the nonlinearity directly is understandable. Regrettably, exact solutions exist for only a tiny subset of differential equations. Exact solutions to many interesting and important problems remain unyielding. There are notable exceptions, for example Duffing's oscillator (in its unforced and undamped form) was solved (by Duffing [58]) in terms of elliptic functions. The homogenous nonlinear IVP is,

$$f'' + kf + k_d f^3 = 0, \quad f(0) = x_0, \quad f'(0) = 0 \tag{2.24}$$

The solution in terms of Jacobi elliptic functions is,

$$f(x) = x_0 \,\mathrm{cn}\left(\sqrt{k + k_d x_0^2}t, \sqrt{\frac{k_d x_0^2}{2\left(k + k_d x_0^2\right)}}\right), \quad k + k_d x_0^2 \neq 0 \tag{2.25}$$

where $\mathrm{cn}(x, m)$ is the Jacobi elliptic cosine function [59].

Several important contributions have been made regarding the existence and uniqueness of solutions to differential equations. The first such theorem, published (with the correct proof), by Peano and Cauchy in 1890 [60] applies to initial-value problems. In the work,

[9] It is important to note that all these examples are themselves approximations.

TABLE 2.3: Summary of existence and uniqueness theorems for different classes of differential equation.

Theorem	Applicable DEs	Conditions	Exist.	Unique.
Peano	ODE IVPs, SODE IVPs	Ψ, g continuous	Yes	No
Piccard-Lindelöf	ODEs IVPs, SODE IVPs	Ψ, g Lipchitz	Yes	Yes
Carathéodory	Inhomogeneous ODEs	Ψ continuous	Yes	Yes
Cauchy-Kowalevski	Cauchy PDE IVPs	Ψ, g analytic	Yes	Yes

the authors are able to prove that any first order ODE IVP has a solution provided Ψ is continuous over the support of the independent variable.

A more powerful but restrictive result was later proved for first-order ODE IVPs. The so-called Piccard-Lindelöf theorem proves both the existence and uniqueness of solutions provided that Ψ is Lipchitz continuous. An even more general result still is proved by Carathéodory, requiring only that the homogenous terms of an ODE be continuous. The result proves both existence and uniqueness for a number of non-smooth differential equations. For PDEs with specific initial Cauchy boundary constraints, the Cauchy-Kowalevski theorem proves uniqueness and existence of analytic PDE IVPs.

Table 2.3 contains the existence and uniqueness theories described above. The table is only intended to illustrate the scope of what is and is not known. For a more detailed reference, the reader is directed to [61].

It is one thing to be able to prove the existence or uniqueness of a solution, but it is entirely different to be able to find that solution. Of the types of equations covered by the theorems in Table 2.3, only a further subset are able to be solved directly. There are scarce examples of general solutions to differential equations outside of the linear ODE class.

2.2.2 So you want to solve a differential equation?

So far in this chapter, much has been stated about the existence and uniqueness of solutions and the methods and challenges contingent in identifying them. But what objects can constitute a solution? It has been stated already that the solution is the unknown function f or functions $\mathbf{f}$ that satisfy both the differential equation and any constraints placed upon it. Clearly there are a multitude of objects that might have this property.

The term 'solution' has been applied to a great many objects in the differential equation literature. Analytical methods provide functional forms for f as expressions in mathematical notation. Such objects can represent an exact solution (satisfying the

DE and its conditions exactly across the entire domain), or approximately (providing a close approximation in some subset of the domain). Numerical methods specify only the values of f at a finite subset of the domain. Heuristic and hybrid methods vary in their approaches, often combining a mixture of exact functional and numerical terms (for example a vector of basis functions with numerically-optimised weights).

Approximate methods

A great number of tools are at the disposal of the engineer for providing solutions to nonlinear ODEs in an approximate manner. Perhaps the most familiar approach is that of *numerical integration*. Methods such as Euler's method and Runge-Kutta schemes, specify approximate solutions to differential equations by iteratively advancing the dynamics forwards in time. At each time point, approximations of the derivatives are taken and used to predict a system trajectory over some time interval h. For many such methods, theorems have been proved placing upper bounds on the amount of absolute and relative error accrued during integration for various sizes of h. Recently, a probabilistic view of numerical integration has been taken [62, 63], that includes a quantification of the uncertainty introduced by the integrator. A full review of numerical methods lies squarely outside the remit of this chapter, the interested reader is directed to [64] for a jumping off point.

As well as numerical methods, there are also a number of analytical approaches that specify approximate solutions to differential equations. Examples include perturbation methods [65] and approaches based on harmonic balance [66].

For many problems in structural dynamics, a numerical-integration solution is sufficiently useful. However, approximate solutions can only provide a narrow view of the response of a structure. What happens if a parameter changes value? What happens if the excitation to the system is radically different? Approximate solutions cannot approach these questions without recomputing the dynamics.

Although pervasive and useful, it will not be an objective of this thesis to produce approximate solutions to differential equations. Instead the focus will be on methods that might yield *exact* analytical solutions.

Exact methods

In order to specify an exact solution to a differential equation, one must primarily rely on exact analytical tools. The general approach depends highly on the form of the problem, but can be summarised as a search for a transformation into a canonical form for which

the solution is known up to some parameterisation or change of variable. Although the problem of knowing generally if one can make such a transformation is likely unknowable, strong heuristic methods have been developed and widely implemented in computer algebra systems (CAS). CAS software is, at the time of writing, highly proficient at locating the known solutions to differential equations, with leading packages such as Mathematica [67], Maple [68] and Sympy [69] able to specify exact forms for many known solutions in seconds.

Despite their proficiency in solving known problems, CAS suites are completely unable to approach problems with no known solution. In such cases, the researcher has essentially two options; the first is to use some insight to the problem from physics or elsewhere in order to glean insight into the form of the solution. The success of this approach is far from guaranteed, with luck seeming to play as important a role as intuition.

Alternatively, one might try to leverage computational might in order to approach the solution. Such an approach could be undertaken in the brute-force sense, sampling at random from trial functions until one that satisfies the differential equation is found, or conducted in a more principled manner using heuristic methods.

It is the heuristic approach that offers the most potential for exact solutions to as-yet unsolved differential equations. Using computational effort to heuristically solve differential equations exactly is not a new proposal. In fact, Koza highlighted the viability of *symbolic regression* in the widely-cited book [70] as early as 1992. It is the opinion of the author that the computational resources available to the modern researcher combined with recent advancements in the power of CAS tools provide an exciting opportunity. If these components can be combined in a new framework for exact solutions, the author believes that some previously-intractable problems may finally yield solutions.

2.2.3 Symbolic regression

One of the most pervasive approaches thus far considered for heuristic solution of differential equations is symbolic regression; the advantages are several. The approach has no ties to linearity or any other type of differential equation, a generic representation can be constructed that can encode a broad array of functional forms and no restriction is placed on the form of the objective function.

Symbolic regression performs optimisation over mathematical expressions. It is most useful when neither the underlying model or its parameterisation are known to the investigator or when an analytic (white-box) model is required. Most commonly, symbolic regression is conducted as an application of *genetic programming* [70], but this is

not necessarily the case. Several alternative symbolic regression frameworks have been proposed including approaches based on deep learning [71] and Bayesian inference [72].

Despite these alternate frameworks, genetic programming remains the most common tool. A major advantage is the degree of flexibility afforded by the approach. Authors are able to select encodings, search operations and objective functions in order to most suitably approach the problem at hand.

A great number of problems have been approached by authors using symbolic regression. In addition to solving generic differential equations, authors have considered control problems [73], financial time-series forecasting [74], chemical process modelling [75] and many other problems.

As well as countless application papers, substantial academic effort has been invested in developing ever more effective methodologies for performing symbolic regression. A full review of the state of the art is beyond the scope of this chapter, but a recent review in [76] summarises the relative benchmark performance of the field well. Important contributions include regularisation based on order of nonlinearity in [77] and semantic structure-preserving crossover and mutation operations for expressions in [78].

Symbolic regression of solutions to differential equations

The first serious consideration is given by Koza in his well-cited 1992 monograph [70]. In the book, a Lisp-program encoded symbolic regression genetic program is presented that is able to find approximate and exact solutions to several symbolic regression problems including the solution of three differential equations with relatively compact solutions. The features of the genetic program used (prefix encoding, tree crossover, subtree mutation, numerical differentiation, L^2-norm error for the fitness function), have come to be recognised as the standard methodology. This approach is often used as the baseline to which subsequent methods are compared.

After Koza, several authors presented variations on the method including string encodings [79], reverse-Polish notation [80], a hybrid-analytical approach [81] and an extension to systems of differential equations with separate parameter estimation in [82].

The next important result was generated by Tsoulos and contributors in 2006 [83]. In the paper, the authors propose a grammar-based encoding combined with elementary single-point crossover and mutation operations. Fitness is calculated using an L^2 metric over the right-hand side remainder of the target differential equation with additional terms included for specifying boundary conditions. Differentiation of candidate expressions is conducted numerically using automatic differentiation [84] at a number of pre-specified

sampling points. In the opinion of the author, the most valuable contribution of the article is the benchmark suite of differential equations included in the work. The authors present nine linear ODE problems and four nonlinear ODE problems. An extended method, capable of dealing with SODEs and PDEs, is also presented and applied to four and seven such problems respectively. A problem with no closed form solution is even attempted, returning an approximate solution. The article reports the performance of their method on each of these problems recovering the exact solution in most cases.

A serious limitation of the work of Tsoulos et al. is the compactness of the solutions to the differential equations considered. In almost every problem, the solution can be represented by a compact expression tree, severely limiting the effective search space. One imagines that the method presented would struggle to solve differential equations with less compact solutions. Despite this, the problems presented in the paper have been used subsequently as a benchmark, for example in [85].

A 2008 contribution in [86] presents an approach based on L^2 optimisation of a number of co-location points for the solution of PDEs. On cases where the PDE boundary conditions have a geometrically simple structure, the authors propose a method for appending terms to the candidate solutions such that the boundary conditions are satisfied by construction. In non-simple cases, the expressions are given extra terms from the output of a radial basis function network.

In 2010 the authors of [85] proposed that Cartesian genetic programming (CGP) be applied to the solution of differential equations. In the paper, the authors demonstrate an improved benchmark performance on a suite of problems similar to the ODEs used in [83], compared to a standard tree-based genetic programming approach.

A 2015 paper [87] utilises the idea of grammar encodings and uses techniques from continuous space optimisation (namely differential evolution) in order to produce solutions. Elements of the encoding vector are rounded to the nearest integer after genetic operations to ensure validity. The results of this paper appear promising and a highly accurate approximation to a form of Duffing's equation is claimed. However, fitness is only calculated on a very limited domain of the functions and in most cases this domain is small enough that the solution defined on it is monotonic. This combined with unrealistic solutions (involving many nested transcendental functions or dramatically complex structures) are significant limitations of this study.

A paper published in 2016 presents genetic programming and automatic differentiation (GPAD) [88]. The method uses a tree encoding and numerical automatic differentiation to evaluate fitness. The approach is demonstrated on a number of classes of differential equation with promising results. As far as the author is aware this contribution is the

first to leverage an object-oriented approach to tree representation. The approach was later extended in 2018 to the domain of stochastic differential equations [89], employed both in the specification and solution of models.

A recent contribution in [71] proposes a very different approach to those thus far considered. In the paper the authors recast the problem in a deep learning framework using a sequence-to-sequence model [90] trained on a synthetic dataset comprised of equation-solution pairs. The results of the study are promising, although there are significant doubts about the ability of the proposed model to generalise beyond the training data. A response to the paper is presented in [91] that outlines several weaknesses of the approach.

Despite steady progress, no single approach has yet proved decisive in driving heuristic solution of differential equations into new territory. Despite the significance of positive results, the conditions for research in this direction are harsh, and not without risk. However, the author retains a positive outlook. For a given nonlinear ODE problem, essentially three scenarios can be envisaged:

1. Closed form solutions do not exist in terms of known transcendental functions and objects.
2. Closed form solutions do exist, but heuristic methods are insufficiently powerful to locate them.
3. Closed form solutions are available and can be found by heuristic means.

Of these scenarios, the author feels that the second is the least probable. If a closed-form solution is available, then it stands to reason that given enough computational effort, it may be found by heuristic means.

The first scenario raises some interesting questions. The lack of a solution in terms of known transcendental functions, motivates a search for new objects that encode the behaviour of nonlinear dynamic processes. It is often the case in mathematics that the discovery of fundamental objects is driven by a need to explain phenomena which can be readily observed, the famous Bessel functions discovered while investigating Kepler's planetary motion (and later applied to many other physical systems) are pertinent examples.

The absence of any solution carries interesting philosophical ramifications. In a structural dynamics context, this could be interpreted as evidence that the differential equation is an inadequate description of the dynamics. To arrive at this conclusion one must

only reason that a 'solution' to the true EOM can always be observed experimentally. To quote the author's supervisor "When I turn on the tap, water comes out!".

It is the third scenario from which the author draws optimism. Considering only the nonlinear ODEs that are commonly seen in nonlinear structural dynamics, it is hard to imagine that there is not a single one that is inexpressible in terms of the transcendental objects that are already known.

2.3 Nonlinear normal modes

In the absence of a solution to a nonlinear ODE, the engineer does not give up hope that the dynamics might still be understood. For structural dynamics problems, linear modal analysis has come to define the way that engineers interpret system behaviour. Tracking of modal properties has enabled considerable further analysis in fields such as engineering design, machining science [92] and structural health monitoring (SHM) [93].

In this section, the procedure of linear modal analysis is introduced. It is shown that LMA has a great number of properties that make it ideal for practical analysis of structural dynamics. It is clear that a great deal of research effort has been expended in the search for a nonlinear extension or alternative to LMA that is able to preserve these desirable qualities. Some criteria for a useful nonlinear modal decomposition are motivated.

2.3.1 Linear modal analysis

Before the nonlinear approaches can be reviewed, it will first be useful to revise the linear theory.[10] A multiple degree-of-freedom (MDOF) linear structural dynamical system (spatially discretised), can without loss of generality be described entirely by the linear ODE,

$$M\ddot{\mathbf{y}} + C\dot{\mathbf{y}} + K\mathbf{y} = \mathbf{x}(\mathbf{t}) \tag{2.26}$$

where the M, C, K are $n \times n$ parameter matrices of the n-DOF system, $\mathbf{y}$ are the system displacements and $\mathbf{x}$ is the system excitation. In order to derive the modal properties, one considers initially the undamped, homogenous case $C = \mathbf{0}$, $\mathbf{x} = 0$,

[10] The theory here is well covered in many textbooks ([1] for example), but is reproduced here in the interest of consistent notation if nothing else.

$$M\ddot{\mathbf{y}} + K\mathbf{y} = 0 \tag{2.27}$$

The above is now a homogenous linear ODE with nontrivial trial solution,

$$\mathbf{y} = \phi \cos(\omega t) \tag{2.28}$$

where ϕ is a *modeshape* vector. Substituting the above into the ODE,

$$-M\omega^2\phi \cos(\omega t) + K\phi \cos(\omega t) = 0 \tag{2.29}$$

and then cancelling out the cosines and rearranging gives,

$$K\phi = \omega^2 M\phi \tag{2.30}$$

which can be recognised as a generalised eigenvalue problem[11], the nonzero solution of which leads to n eigenvalue-eigenvector pairs (modes); the natural frequencies ω_i^2 and modeshapes ϕ_i corresponding to the i^{th} mode.

The significance of these quantities cannot be overstated. Apart from acting as a reduced order model of the dynamics (naïvely n^2+n parameters instead of $2n^2$ in the undamped case)[12], the properties of the eigen-decomposition allow the dynamics to be decoupled. Consider now the effect of transforming the original coordinate system by the modeshape matrix Φ. Let the modal coordinate space be $\mathbf{y} = \Phi\mathbf{u}$. Then the equation of motion becomes,

$$M\Phi\ddot{\mathbf{u}} + K\Phi\mathbf{u} = \mathbf{x}(\mathbf{t}) \tag{2.31}$$

Pre-multiplying now by Φ^{T},

$$\Phi^{\mathsf{T}}M\Phi\ddot{\mathbf{u}} + \Phi^{\mathsf{T}}K\Phi\mathbf{u} = \Phi^{\mathsf{T}}\mathbf{x}(\mathbf{t}) \tag{2.32}$$

[11] In practice for well-defined structural dynamics systems, the mass matrix M is invertible and so a standard eigenvalue problem can be solved instead.

[12] There is some additional complexity here that arises from the fact that the modeshape vectors are an orthogonal basis. Geometrically, the modeshapes lie on the compact Stiefel manifold in $\mathcal{R}^{n\times n}$, which has dimension $\mathcal{R}^{n^2-\frac{1}{2}n(n+1)}$. The overall number of free parameters is therefore $\frac{1}{2}(n^2+n)$.

Since from the properties of the eigen-decomposition, the parameter matrices are diagonalised by the eigenvector matrix.

$$\Phi^{\mathsf{T}} M \Phi = [M] = I \tag{2.33}$$

$$\Phi^{\mathsf{T}} K \Phi = [K] = \omega_n^2 I \tag{2.34}$$

whereby bracketed quantities refer to diagonal matrices. The equations of motion in the modal space are completely uncoupled. In the case of proportional damping, the damping parameter matrix C can be expressed as a linear combination of the mass and stiffness matrices.

$$C = \alpha M + \beta K \tag{2.35}$$

substituting this into the equation of motion and once again taking the modal coordinate transform,

$$M\Phi\ddot{\mathbf{u}} + C\Phi\mathbf{u} + K\Phi\mathbf{u} = \mathbf{x}(\mathbf{t}) \tag{2.36}$$

$$\Phi^{\mathsf{T}} M\Phi\ddot{\mathbf{u}} + \Phi^{\mathsf{T}}(\alpha M + \beta K)\Phi\dot{\mathbf{u}} + \Phi^{\mathsf{T}} K\Phi\mathbf{u} = \Phi^{\mathsf{T}}\mathbf{x}(\mathbf{t}) \tag{2.37}$$

$$[M]\ddot{\mathbf{u}} + (\alpha[M] + \beta[K])\dot{\mathbf{u}} + [K]\mathbf{u} = \Phi^{\mathsf{T}}\mathbf{x}(\mathbf{t}) \tag{2.38}$$

$$[M]\ddot{\mathbf{u}} + [C]\dot{\mathbf{u}} + [K]\mathbf{u} = \Phi^{\mathsf{T}}\mathbf{x}(\mathbf{t}) \tag{2.39}$$

and the equations of motion are still uncoupled. For most structural systems, the quantity of damping is closely related to the stiffness (and not the mass, i.e $\alpha = 0$) and so the parameters of $[C]$ can be extracted as the damping ratios,

$$[C] = \beta[K] = 2\omega_n\zeta I \tag{2.40}$$

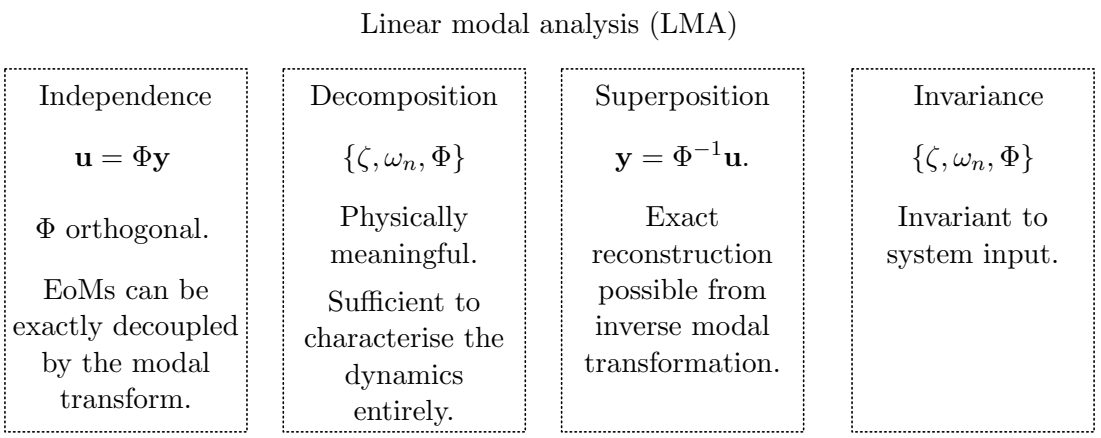


FIGURE 2.2: The useful properties of linear modal analysis.

There are yet more advantages of the modal decomposition. As well as having a more compact representation in terms of the natural frequencies, damping ratios and modeshapes, these properties also afford significant physical intuition. The natural frequencies of a system inform the engineer about the locations of resonances. The damping ratios give insight into how quickly the response of the system will decay close to each resonance. The modeshapes define hyper-planes in the configuration space that describe the synchronous motion of the system at each mode. It would be easy to argue that armed with only these three quantities, the structural dynamicist has enough information to inform design choices, manage operating conditions and forecast performance at new operating points. The set $\{\omega_n^2, \zeta, \Phi\}$ is invariant given the system parameters, and sufficient to completely describe the dynamics for all possible inputs [2].

Perhaps most miraculous of all, is that the physical coordinates can be exactly reconstructed by an inverse transformation, $\mathbf{y} = \Phi^{-1}\mathbf{u}$. This allows the contribution of each mode to be analysed independently and then recombined via the principle of superposition. For structures of many DOFs (and therefore many modes), this permits a considerable computational saving, and the potential to neglect modes that are far from the operating condition, for a reduced-order representation.

In summary, linear modal analysis is so pervasive because it is so useful in practice. Figure 2.2 depicts the principal utilities of LMA.

2.3.2 What is a mode anyway?

Given the advantages of LMA in practice, it is utterly unsurprising that considerable research effort has been expended in the search for an extension to the nonlinear case; a decomposition into *nonlinear normal modes* (NNM) of vibration.

Before jumping straight into the deep end, it is useful at this stage to hold some discussion into exactly what is meant by a 'mode' of vibration and how this might be extended into the nonlinear case[13]. In the published literature, there has been some contention as to what might constitute a nonlinear 'mode' of vibration. To illustrate, listed below are some objects that have been described as nonlinear modes:

1. Nonlinear decompositions into modal coordinates $u_i = f_i(\mathbf{y})$: The f_i are the modes.
2. Synchronous motions of a structure all passing through equilibrium at the same point: $y_i = f_j(y_1)$: The f_j are the modes.
3. Manifolds (in configuration or phase space) upon which modal dynamics remain $y_i = f_j(y_1, \dot{y}_1)$: The f_j are the modes.
4. Latent directions that minimise the statistical correlation between displacements $u_i = f_i(\mathbf{y})$: The f_i are the modes.

Rather than offer yet another definition, the author prefers instead to consider the *useful properties* that a nonlinear normal mode might retain from the linear analysis. Taking this viewpoint encourages a natural categorisation of approaches that have been thus far proposed. In this section, NNM approaches are grouped by the qualities of linear modal analysis that are preserved. In Figure 2.2, the properties of LMA are grouped under four categories.

Independence

Independence[14] is defined here as the ability of the decomposition to render the dynamics into a basis of modal displacements that cannot excite each other in isolation. Should motion be initiated in a single mode, the others should not be excited. In the linear case, Φ represents an orthogonal basis for describing the dynamics (given damping or proportional or Rayleigh type). Theoretically, many frameworks for NNMs appeal to the results of Lyapunov that show that there exist at least n families of periodic solutions around stable equilibria in a n-DOF nonlinear system. It stands to reason that any nonlinear extension to modal analysis should therefore define at least n NNMs.

[13]The discussion in this section must necessarily deal with semantics, which is rarely a precise endeavour. It is worth pointing out at this stage that the categorisation of NNMs presented in this chapter are derived from the authors opinion as to the meaning of vague notions such as 'modal', 'decomposition' and 'nonlinear modeshape'. It is the opinion of the author that use dictates the meaning of such words.

[14]In other studies [94], this property has been termed *invariance*, however, the author prefers independence as it does not give the false impression that the NNMs do not depend on the energy present in the system.

Decomposition

As noted in [94], the utility of any framework for nonlinear modal analysis is firmly rooted in the ability of the method to explain the underlying dynamics in a way that is intuitive. As has been shown in the linear case, the decomposition into natural frequencies, damping ratios and modeshapes is appealing because these objects offer immediate insight into the behaviour of the system. Further, in the linear case, the modal properties are sufficient to form and complete description of the dynamics, subject to arbitrary. Decomposition is therefore defined as the ability of the framework to present a physically-meaningful explanation of the dynamics. A related quality is that the nonlinear framework should reduce correctly to LMA in the limit of linearity.

Superposition

A great advantage of LMA is that the physical dynamics of the system can be reconstructed in terms of a superposition of modal contributions. Regrettably, the principle of superposition is predominantly absent in the nonlinear case. However, inverse nonlinear transformations (approximate or otherwise) that enable the total system response to be expressed in terms of nonlinear modal contributions are a highly desirable feature of a nonlinear framework.

Invariance

In LMA, the modal properties are invariant to the excitation of the system and are therefore sufficient to characterise the dynamics across the entire operating range. It is well understood, that nonlinearity introduces energy dependence into the dynamics [2]. Techniques that are able to *generalise* between input types and excitation levels are therefore desirable in any nonlinear extension to LMA.

2.3.3 Frameworks for nonlinear modal analysis

In the last century, several frameworks have been proposed as nonlinear extensions to linear modal analysis. Theoretic appeals have been made to Koopman theory [95, 96], group theory [97], normal forms [98], complex manifolds [99], and spectral submanifolds [100] in the search for a practical extension to LMA. Of the proposed frameworks, there are two that have garnered particular attention; that of Rosenberg, and that of Shaw and Pierre. These two frameworks are reviewed in the following sections.

Rosenberg: Coherent motions

The earliest formulation for NNMs is that of Rosenberg [18, 101]. The formulation is derived from the observation that in the linear case, for each mode, the displacements can be written as a linear multiple of some arbitrary driving displacement DOF (chosen in the notation that follows to be y_1).

$$y_j^{(r)} = \phi_{rj} y_1^{(r)} \tag{2.41}$$

Geometrically, this can be interpreted as a straight line in the configuration space. These linear modal motions are visualised for a 2-DOF linear system in Figure 2.3 as dashed lines. Rosenberg noticed that these straight lines in the linear case result in *coherent motions* of the structure at each mode; displacements at every degree of freedom pass through the equilibria and reach their extreme values at the same point in time. In order to facilitate the nonlinear nature of the motions, Rosenberg proposed an extension to equation (2.41), that permits the displacements at each mode to depend on a nonlinear map from the driving coordinate. The new ansatz is therefore,

$$y_j^{(r)} = f_{rj}(y_1^{(r)}) \tag{2.42}$$

where the f_{rj} describe the nonlinear motions at the r^{th} nonlinear mode. In this way, the NNMs in the Rosenberg framework are defined as the coherent motions of the structure. In Figure 2.3, the in-phase (both masses translate in the same direction at the same time), and out-of-phase (masses translate in opposite directions), NNMs of a nonlinear 2-DOF system are visualised by solid lines in the configuration space of the system. The Rosenberg definition has several advantages, such as a convenient visualisation in the configuration space as well as the property that if any single mode is excited then all others will remain quiescent for all time. Evidently, for a nonlinear system that does not obey the principle of superposition, the f_{rj} (and therefore the NNMs), will depend on the energy present in the system. Tracking these NNMs through different energy levels gives rise to a convenient visualisation strategy [94] in a frequency-energy plot (FEP). Proponents of Rosenberg's formulation of an NNM argue that the FEP enables the structural dynamicist to interpret how the systems resonances (for all modes), depend on the total system energy, which is a useful tool in both design and testing scenarios.

However, the original framework has a number of disadvantages that limit its utility in practice. Chief among these is the limitation to conservative (i.e undamped), systems. Although some extensions to the Rosenberg framework have been proposed to address

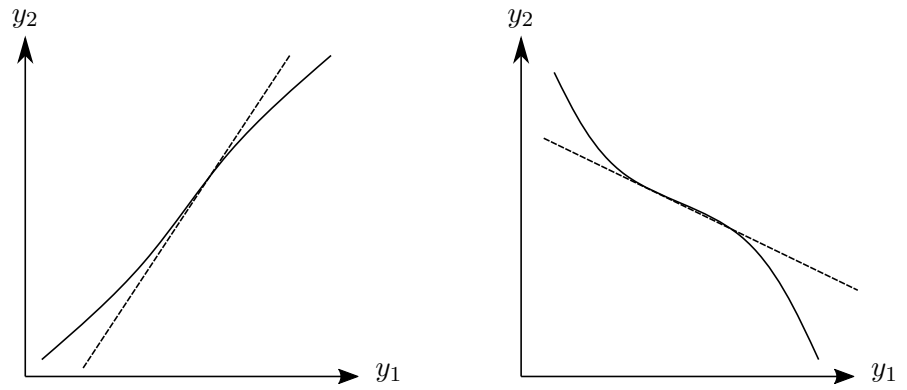


FIGURE 2.3: The Rosenberg formulation of an NNM describes coherent motions of a nonlinear system. Dashed line: LNM, solid line: NNM.

Rosenberg NNM

independence	Decomposition	Superposition	Invariance
$y_i = f_i(y_1)$	f_i	Not explicitly possible.	$f_i = f_i(y_1, E)$
y_i cannot excite each other.	Coherent motions of the structure. Can track NNMs through bifurcation and modal interactions.		Expressly dependent on energy in system.

FIGURE 2.4: The useful properties of the Rosenberg NNM framework.

this shortfall [102–104], it is largely argued that the damped dynamics are often close enough to the undamped dynamics for the approach to remain useful. The practical utilities of the Rosenberg NNMs are summarised relative to those of LMA in Figure 2.4.

Another limitation of the Rosenberg NNM framework is that it is not expressly possible to reconstruct the overall physical displacements from the contributions of the individual NNMs [103]. This limits the utility of the approach from a modelling point of view whereby the dynamicist wishes to conduct analysis in a simplified modal space and then reconstruct the overall dynamics.

Since the original proposition of the method [18, 101], considerable attention has been paid to extending the framework. Important results include the extensions to *non-necessarily synchronous motions* [94] in order to account for internal resonances, whereby one DOF oscillates at a different frequency (a 1:3 harmonic as seen in Duffing oscillators for example) and so the motion is no longer synchronous.

In early contributions, the Rosenberg NNMs were calculated by analytical means, directly from the EOMs [105]. However, in the last decade, several data-driven approaches have gained popularity that can infer the Rosenberg NNMs from measured data [106, 107]. Important results include an experimental scheme that can infer the NNM from broadband excitation [108] and a method based on deep learning and autoencoder networks that derives the modal responses from output data [109]. In terms of applications, there have been a number of contributions that demonstrate the Rosenberg method. Examples include a light nonlinear attachment in [102], and a full-scale aircraft in [110].

There are a number of excellent review texts that detail the developments in the Rosenberg NNM literature. The early developments are well summarised in references [111, 112], whereas more recent contributions can be found in references [94, 105].

Shaw-Pierre: Invariant manifolds

Several decades after the Rosenberg definition was proposed, an alternative formulation based on the concept of invariant manifolds was introduced by Shaw and Pierre [19, 113–115]. The Shaw-Pierre NNM was inspired by results from centre-manifold theory, and extends the Rosenberg framework by lifting the ansatz into the phase space of the dynamic system. As in the Rosenberg case, the dynamics of each mode are expressed in terms of a single driving DOF. However, in the Shaw-Pierre framework, both displacements and velocities are considered. The ansatz is now,

$$y_i^{(r)} = f_r(y_1, \dot{y}_1) \tag{2.43}$$

$$\dot{y}_i^{(r)} = g_r(y_1, \dot{y}_1) \tag{2.44}$$

Geometrically, the Shaw-Pierre formulation leads to the the specification of *invariant manifolds* in the phase space. It is known that the dynamics of linear systems also lie on invariant manifolds in the phase space. These linear planar manifolds arise from the eigenspace of the modal decomposition. In the nonlinear case, these manifolds are non-planar but remain tangent to the linear modal plane at the equilibrium. This effect is visualised in Figure 2.5 for a 2-DOF system possessed of two modes, one whereby the motions of the DOFs are in phase and another where they are out of phase.

As well as the geometrically more general formulation, the Shaw-Pierre approach carries a number of distinct advantages over that of Rosenberg. Principally, the inclusion of the

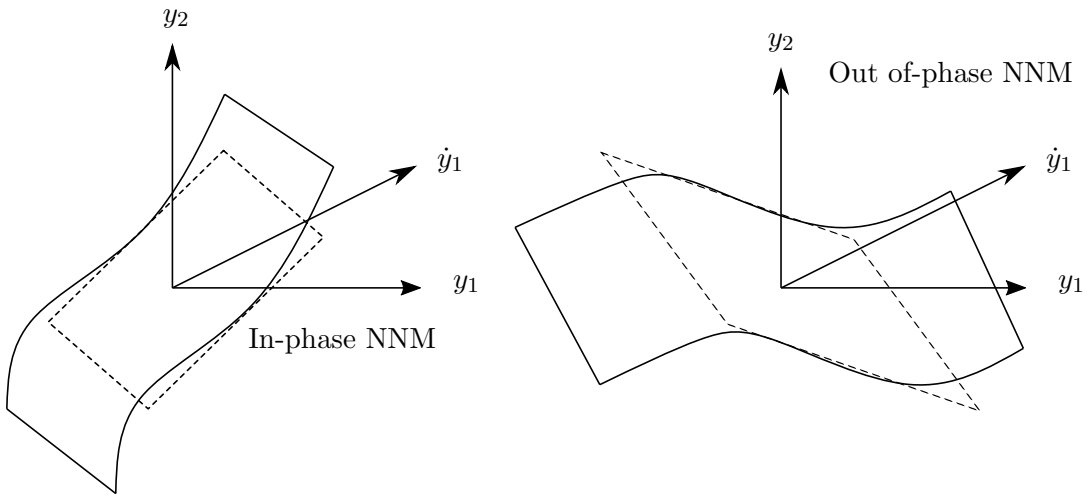


FIGURE 2.5: The Shaw-Pierre formulation of an NNM; a curved manifold in the phase space of the dynamics that is tangent at the equilibrium to the underlying linear mode. Dashed line, LNM, solid line: NNM.

Shaw-Pierre NNM

Independence	Decomposition	Superposition	Invariance
$y_i = f_i(y_1, \dot{y}_1)$ $\dot{y}_i = g_i(y_1, \dot{y}_1)$ System dynamics are SDOF on the manifolds.	$\mathbf{u} = M(\mathbf{y})$ Invariant manifold in phase space Tangent to linear modes in the phase space.	$\mathbf{u} \approx M^{-1}(\mathbf{y})$ Approximate results for inverse map. Cannot always invert M.	Motion initiated on the manifold remains there for all time. Modes are valid in the neighbourhood of equilibria.

FIGURE 2.6: The useful properties of the Shaw-Pierre NNM

velocity terms in the phase-space representation, allows a proxy for system energy and therefore non-conservative systems are handled naturally.

Another advantage is that any motion initiated on one of the modal manifolds remains there for all time. This is akin to the independence property of Rosenberg, but naturally accommodates decaying orbits in the phase-space. The formulation also permits the dynamics to be expressed in terms of modal coordinates that decouple the dynamics into SDOF oscillators on the manifolds. The authors also suggest a method for obtaining a forward transformation, from the physical to the modal dynamics on the manifold. Also proposed is an approximate inverse transformation that can be used to reconstruct the physical dynamics. The practical utilities of the Shaw-Pierre NNM are summarised relative to those of LMA in Figure 2.6.

The core of the Shaw-Pierre framework relies on an appeal to centre manifold theory [116]. A limitation of the Shaw-Pierre framework is therefore that the ethos of the

method is *local* in nature, and the behaviour of the dynamics far from the equilibria may not be well represented.

Another limitation is that the Shaw-Pierre framework as first described [19] is unable to accommodate internal resonances, and other types of system including situations when the underlying system cannot be decoupled. However the authors note that the method can be extended to accommodate some types of internal resonances by including multiple pairs of state variables as the driving DOFs.

Although originally defined for spatially discretised systems, the Shaw-Pierre framework was extended to continuous systems in [117]. In the work, the authors give results for several beam case studies and derive the governing equations of motions on the invariant manifolds.

In [118], a Galerkin approach is taken to specifying the invariant manifolds. The authors argue that taking this approach improves the range over which the calculated manifolds are valid, as they are no longer restricted by a continuation approach close to the linearised system at the equilibrium.

More recently, the method was applied as part of a deep learning framework [109] that specifies the NNMs (of both Rosenberg and Shaw-Pierre type) by considering the bottleneck layer of an auto-encoder network to be *intrinsic directions* in the modal space and then interpreting the modes by projecting single driving DOFs through the decoder layer. The approach operates in a data-driven manner that adds practical utility to the method.

A recent generalisation of both the Rosenberg and Shaw-Pierre frameworks to the case of non-conservative, period systems has been proposed by Haller and co-authors in [100]. Motivated by a desire to provide a consistent methodology, the frameworks are united under the umbrella of spectral-submanifolds (SSM). The authors are able to provide existence and uniqueness results for the underlying SSMs and motivate their use as a tool for model order reduction. The computation of the SSMs is analytically very demanding, however a recent contribution envisages a data-driven scheme [119].

Nonlinear decomposition methods

As well as theoretical frameworks for the consideration of nonlinear normal modes, several approaches have been applied in a purely data-driven setting and carry many of the utilities of LMA. At the core of such approaches is the nonlinear decomposition,

$$\mathbf{u} = F(\mathbf{y}) \tag{2.45}$$

that decomposes the physical displacements into an equal number of modal displacements, wherein the dynamics are simplified (by some measure). Methods of this type include principal orthogonal decomposition (POD) [120–123] and independent component analysis (ICA) [124]. Despite a data-driven formulation, some of these methods are able to interface with vibration theory. For example, in the linear case, the POD modes are shown in [20] to converge to the linear modes[15] as the amount of data increased.

Inspired by ICA and the ansatz of Shaw-Pierre leading to a multinomial expression for the forward modal transformation, a recent approach based on statistical-independence was proposed [22, 125]. The approach casts the problem of specifying a nonlinear decomposition as one in machine learning. The approach proceeds by specifying a number of inductive biases within an objective function. The method was shown to produce a decomposition into a modal basis wherein the power spectral densities (PSDs) have distinct peaks on a number of case study examples. Since the original proposition, several contributions have extended the method by considering alternative forms for the nonlinear decomposition function and the inductive biases used. In [126], a number of non-parametric learners are considered for the decomposition. In [127] some nonlinear correlation functions are considered. The form of the nonlinear decomposition was further extended to a normalising-flow based method in [128] and variational auto-encoder in [129]. In [23], a cycle-consistent generative adversarial network (cycle-GAN) was employed. The cycle-GAN was shown to have a number of distinct advantages over other approaches including the automatic specification of an inverse transformation and the enforcement of *conformality* via the mapping.

The statistically-independent framework for modal analysis shows significant promise, although there is a need to go beyond the visual assessments of [22, 23, 126] and consider robustly the nature of the modal transformation in the context of LMA. This framework will form the focus of the work presented in this thesis. To this end, a full description of the approach is saved for a following chapter.

2.3.4 Towards a practical NNM for structural dynamics

It is clear that a great deal of research effort has been expended in the development of nonlinear extensions to modal analysis. With the relevant literature reviewed, attention can now be directed to the specification of some criteria for a practical NNM. Given the many utilities of LMA it is a challenge to arrive at a compact set of criteria.

[15] Provided the mass matrix is proportional to the identity matrix.

There are some properties that all sensible approaches to NNMs must obey. Firstly, it is clear that any nonlinear method should specify[16] at least as many nonlinear modes of vibration as there are DOFs in the structural system. This follows directly from the results of Lyapunov and is observed by all frameworks that have been explored in this chapter. Another important property is that the NNMs can be derived from measured data as well as from the EOMs of the structure. In practice, one does not always have access to the EOMs of a structure and so nonlinear modal analysis in a data-driven manner is essential. Once again, the author observes that data-driven formulations are available for all frameworks so-far considered.

Given the immense utility of LMA in practice, it stands to reason that these utilities should also be desirable in the nonlinear case. The following criteria are therefore defined as the useful properties of a nonlinear extension to modal analysis.

1. *Independence*: The ability of the decomposition to render the dynamics into an independent modal basis, preferably SISO.
2. *Decomposition*: The extent to which the decomposed modal dynamics represent a physically-meaningful (by some measure), basis for understanding the structural dynamics.
3. *Superposition*: The extent to which the original dynamics can be recovered from the decomposition.

These criteria are essentially identical to the useful properties of LMA, as described in Figure 2.2, with the exclusion of invariance. While validity at all excitation levels is a useful property of the linear modal case, in the nonlinear case it is necessary to model directly the energy dependence of the NNM. In Chapter 5, the above criteria will be examined in detail with a view to specify metrics that might be used to assess them more rigorously in a qualitative manner.

2.4 Summary and research questions

With the relevant literature considered, the aims of this thesis can now be stated more formally into research questions. Of the three grand challenges in nonlinear structural dynamics, identified in Chapter 1, it is the aim of this thesis to present novel contributions pertaining to challenges II and III. The following summarises the literature pertaining to these challenges.

[16] In the full-fidelity representation, i.e reduced order models notwithstanding.

Exact solutions to nonlinear differential equations

The consideration of solutions of differential equations as a heuristic search problem is not in itself a novel idea. For example, the work of Seaton and collaborators [85] presents some convenient definitions. However, it is clear that in order to be successful, a thorough consideration of the underlying search problem is critical.

Although a considerable amount of attention has been paid to approximate solutions of differential equations, it appears that there are comparatively few heuristic methods constructed within an ethos where only exact solutions will be accepted. A promising tool for achieving this appears to be symbolic regression. Although a mature field of research in its own right, applications of symbolic regression to the solution of differential equations are sparse in the literature. Of particular interest is the combination of symbolic regression with CAS software for symbolic equality testing within the heuristic framework. The combination of these two elements with a specific focus on exact solutions constitutes a research gap within the published literature.

A great deal of important structural dynamics are governed by nonlinear ODEs and PDEs. Exact, closed-form solutions to both types have the potential to make a profound impact. However, given the exploratory nature of the investigation, an initially narrowed view must be taken, and so the analysis presented in this thesis will consider ODE problems only. An appropriate research question is therefore:

Can a heuristic search methodology be found that leverages CAS and techniques from symbolic regression with the potential to find exact solutions to as-yet unsolved nonlinear ODEs?

Chapters 3 and 4 of this thesis will be dedicated to answering this research question.

A practical nonlinear alternative to linear modal analysis

A nonlinear alternative to LMA has been an active area of research now for more than fifty years. During this time, many frameworks for nonlinear modal analysis have been proposed, each of which is able to preserve only a subset of the features of linear modal analysis. Of the proposed frameworks, those which have received the most attention are that of Rosenberg and the geometrically more general approach of Shaw and Pierre. Despite steady progress in both camps, the practical use of both methods is hamstrung by somewhat extended technicalities, limitations on types of excitation and the lack of an inverse modal transformation.

Of particular interest are NNM frameworks that take a *machine learning* viewpoint towards NNMs. Of the frameworks taking this approach, the recently proposed statistically-independent framework has shown promise. The approach permits a functional decomposition into independent DOFs and constructs an inverse map. However, the approach is still fairly new, and a consideration of the effect of the modal transformation on the modal dynamics has scarcely been considered. In this thesis, the statistically-independent framework will be examined in detail and an attempt made to connect the promising results obtained via the application of *inductive biases* and machine learning to the underlying dynamics. The research question can be posed:

To what extent does the recently proposed statistically-independent framework for nonlinear modal analysis constitute a practical nonlinear extension to LMA?

The approach taken in this thesis shall be to measure practical utility by the consideration of the criteria defined above. Chapters 5-9 inclusive will be dedicated to consideration of this research question.

Chapter 3

Towards exact solutions of nonlinear ODEs

This chapter and the one that follows, are concerned with the second of the grand challenges identified in this thesis, that of finding truly exact solutions to nonlinear ordinary differential equations.

The ethos here will be to approach the problem of finding solutions to differential equations in the same manner that one might approach any other task in machine learning or optimisation. Critically however, approximate solutions will not be accepted. Because of this strict outlook, many conventional techniques and wisdoms are not applicable. For example, floating point arithmetic is too imprecise to evaluate the exact equalities that are required. Similarly, methods that rely on meta-regression steps to specify the values of constants will not be useful as they too are inexact.

In place of conventional techniques, new and existing approaches must be considered specifically in the context of their ability to identify exact solutions to differential equations. To this end, the following chapter will present the problem of finding solutions to ODEs as a heuristic *search problem*, whereby solutions may be found by the application of heuristic optimisation.

Casting the problem of finding solutions to differential equations as one of optimisation is not by itself a novel approach, several authors [71, 81, 83, 130, 131] have attempted this task with varying degrees of success. However, formal study of the search problem is limited in the literature. One such contribution [85], presents some discussion of the relevant search spaces and defines a measure of search complexity based on the sizes of the representation space of expressions and solutions. For compatibility, their notation for the representation and solution spaces is adopted in the definitions developed here.

As well as the search problem formulation, this chapter also considers two other important aspects of the symbolic regression implementation. An enumeration of the various search spaces and a consideration of suites of benchmark problems. After analysing the properties of a useful benchmark, a novel benchmark suite of problems is proposed that has several advantages over a commonly used problem set in the literature.

3.1 The needle in the infinite haystack

Solving differential equations by conventional means is *hard*.[1] Solving them heuristically is by no means an easy feat either. The core issue is that the underlying search problem is extremely difficult. Even under strong assumptions, there is an uncountably infinite number of possible trial functions that one might consider; blindly searching within them for a solution would be akin to searching for a single needle in an infinite haystack.

Faced with such a daunting task, how can one hope to make progress? The approach in this chapter, is to break down the search problem by considering the structure of the various spaces in which the search takes place.

Function space

So, where must the search for solutions begin? The solution to an ordinary differential equation is (by definition) a function. The set of functions is highly diverse and encompasses forms from simple polynomials to nonsmooth monstrosities. Consider the following function attributed to Dirichlet for example,

$$f(x) = \begin{cases} 1/b & \text{if } x = \frac{a}{b} \text{ with } a, b \in \mathbb{Z} \\ 0 & \text{otherwise} \end{cases} \tag{3.1}$$

The above function is continuous for every irrational number and discontinuous everywhere else[2]. Fortunately, in order for a function to be the solution to a differential equation, it must obey certain restrictions. In order to be a solution, a trial function must satisfy both the differential equation itself as well as any boundary conditions that may be imposed upon it. For this to be the case, the function must have defined derivatives up to the order of the differential equation. Let,

$$\Psi(x, f(x), f^{(1)}(x), ..., f^{(r)}(x)) = 0 \tag{3.2}$$

[1] PSPACE-hard, for a numerical solution given some smoothness requirements [132].
[2] Even thinking about it is enough to make the author shudder.

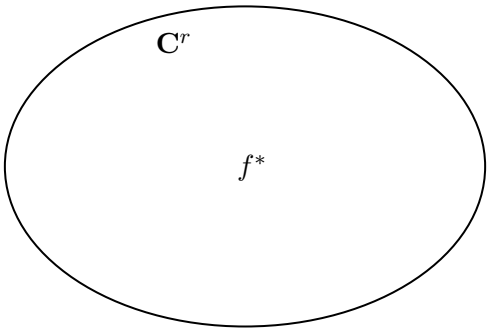


FIGURE 3.1: The function space $\mathbf{C}^r(\mathbb{R})$, containing f^*.

describe a r^{th} order ODE, subject to boundary constraints sufficient to make the solution unique. Denoting the solution to Ψ as f^*, and the function space over the real numbers with r continuous derivatives as $\mathbf{C}^r(\mathbb{R})$, it is required that $f^* : \mathbb{R} \to \mathbb{R} \in \mathbf{C}^r(\mathbb{R})$. Furthermore, let $\omega_{\mathbf{C^r}}$ be the set of all functions $f \in \mathbf{C}^r$ such that $\Psi(f) = 0$ and all boundary constraints are satisfied. If the differential equation satisfies the conditions for the solution f^* to be unique, one has,

$$\omega_{\mathbf{C^r}} = \{f^*\} \tag{3.3}$$

Or equivalently, the haystack contains only a single needle.

Figure 3.1 depicts $\mathbf{C}^r(\mathbb{R})$ and f^* graphically. Suppose now that one is able to sample at random from $\mathbf{C}^r(\mathbb{R})$; inspired by [85], the probability of finding f^* in any given sample is,

$$\mathrm{P}(f = f^*) = \frac{\mu(\omega_{\mathbf{C^r}})}{\mu(\mathbf{C}^r(\mathbb{R}))} \tag{3.4}$$

Where $\mu(S)$ is the counting measure defined on some set S. This (potentially naïve) line of reasoning finds that since there is only a single needle,

$$\mu(f^* \in \mathbf{C^r}) = \mu(\omega_{\mathbf{C^r}}) = 1 \tag{3.5}$$

and a whole lot of hay,

$$\mu(\mathbf{C^r}) = \infty \tag{3.6}$$

the probability of finding the solution vanishes.

$$\mathrm{P}(f = f^*) \approx 0 \tag{3.7}$$

It is clear that any attempt to search for solutions within $\mathbf{C}^r(\mathbb{R})$ by sampling directly is doomed to find only hay.

Expression trees

In reality, it is not even straightforward to (uniformly) sample from $\mathbf{C}^r(\mathbb{R})$.[3] In practice, one must adopt a notational scheme in order to express functions as expressions. Mathematical expressions are able to encode a broad church of function objects, including: derivatives, conditionals, integrals, series and many other complex forms.

With the intent to reduce the complexity of the search problem somewhat, a strong assumption is made here; that the solutions to some (interesting) nonlinear differential equations are representable as closed-form analytical expressions. Although this may seem a bold assumption, the author believes that there are certainly some interesting solutions available as closed form expressions in terms of a finite basis of operators and transcendental functions.

A closed-form analytical expression is a far easier object to work with. Structurally, a closed-form analytical expression can always be represented as a directed acyclic graph (a tree). The exact definition of a closed-form analytic expression, and what structures it is permitted to contain is rather vague and varies from author to author.

However, by considering expressions as trees, a natural definition arises. A closed-form analytic expression is defined as a tree, whereby each node represents a member of either a function set $F = \{f_1, ..., f_m\}$ (in which case the node is internal and must be the root of a subtree), or a terminal set $T = \{t_1, ..., t_n\}$. The overall search basis (simply denoted F hereafter), is given by the elements of these two sets.

This definition affords a degree of flexibility, as one is free to select the elements of F to suit the problem domain.

Consider the following function,

$$f(x) = x^2 - 1 \tag{3.8}$$

[3] There are some methods that can draw samples from functions spaces, for example the Gaussian process. However, closed form expressions for the samples are unavailable.

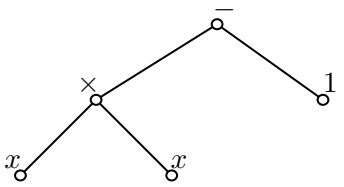


FIGURE 3.2: A tree corresponding to the expression $x^2 - 1$.

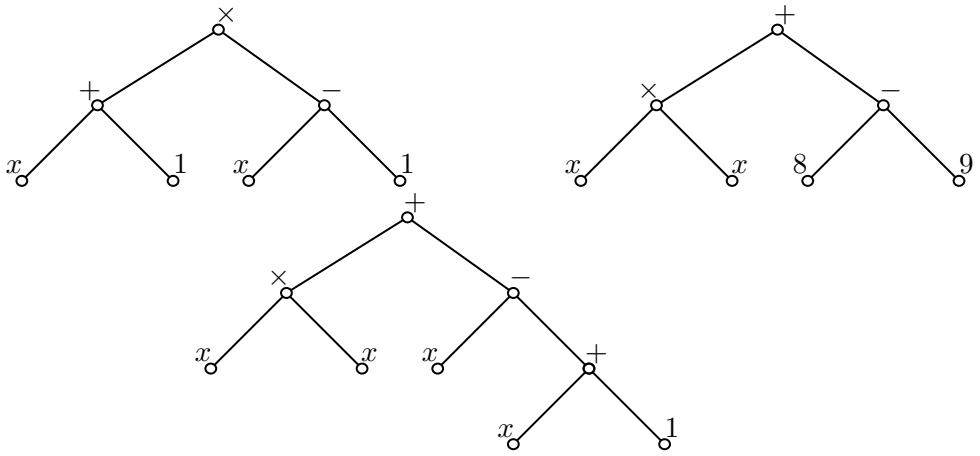


FIGURE 3.3: Several other trees corresponding to $x^2 - 1$.

The tree corresponding to this function might take the form of Figure 3.2; an issue is immediately apparent. The tree in Figure 3.2 might well have taken a different form. Figure 3.3 depicts several other trees that yield the same function.

In the absence of a limit on tree dimensions, there are an infinite number of trees that correspond to any given function[4].

Let $\mathbf{T}(F)$ be the space of expression trees representing closed-form analytical expressions containing functions and terminals from F. It is important to note here that $\mathbf{T}$ and $\mathbf{C}^r$ have very different structures. There exists a many-to-one relationship between expression trees in $\mathbf{T}$ and functions in $\mathbf{C}^r$. Furthermore, not every expression tree in $\mathbf{T}$ corresponds to a function in $\mathbf{C}^r$. In order to directly evaluate functional similarity, one must be able to project expressions into the function space via a mapping $S : \mathbf{T} \to \mathbf{C}^r$. The search problem in expression tree space can therefore be stated,

$$\exists t \in \mathbf{T}(F), \quad S(t) = f^* \tag{3.9}$$

Put simply, it is assumed that within a finite tree space $\mathbf{T}(F)$ there exists a tree t, that represents a solution to the differential equation of interest.

[4]This is easily proved by the consideration of any subtree that represents an identity mapping, $t(a) = a + a - a$ for example. Given no limit on tree dimensions, these identity mappings might be recursively applied *ad infinitum*.

Encoding space

In practice, additional restrictions are often levied on the elements of $\mathbf{T}(F)$, to ensure that trees are not arbitrarily large and do not contain expressions with undesirable properties such as complex terms, asymptotes or excessive compositions of transcendental functions. After [85], this reduced space is termed the *space of representable expressions* $\mathbf{\Omega} \subseteq \mathbf{T}$.

Sampling (uniformly) from $\mathbf{\Omega}$ given arbitrary constraints on the form and dimensionality of the expressions is non-trivial. Practically, one often selects an *encoding space* from which to draw samples instead. A decoding function $P : \mathbf{E} \to \mathbf{\Omega}$ is then used to recover expression representations. There are a number of representational encodings that have been investigated in the literature. A brief categorisation of approaches thus far considered is presented below.

- Naïve tree encodings i.e. $\mathbf{\Omega} = \mathbf{T}$.
- Tree-interpretation methods (grammars, strings).
- Tree-representation methods (infix, prefix, Reverse Polish Notation (RPN)).
- Graph-based methods (Cartesian Genetic Programming (CGP)).

The search space can now be defined as the space of representable expressions, given some encoding space $\mathbf{E}$ and a mapping $P : \mathbf{E} \to \mathbf{\Omega}$. In the language of genetic programming, this mapping is the phenotyping function and elements of $\mathbf{E}$ and $\mathbf{\Omega}$ are the genotype and phenotype respectively.

Returning to the reasoning of (3.4), one recovers the so-called *unguided complexity* k [85] of the search problem as,

$$k = \frac{1}{\mathrm{P}(S(t \in \mathbf{\Omega}) = f^*)} = \frac{\mu(\mathbf{\Omega})}{\mu(\omega)} \tag{3.10}$$

where ω is the set of expressions $t \in \mathbf{\Omega}$ such that $S(t) = f^*$. Unlike in (3.4), this value is tractable. Both $\mu(\mathbf{\Omega})$ and $\mu(\omega)$ can be assumed to be finite and fixed by the selection of Ψ, P and $\mathbf{E}$. In [85], the authors make meaningful contributions to estimating the value of k (by experimentally fitting (3.10) to a binomial distribution) for a number of different problems and two encoding spaces.

A further abstraction is possible; the total complexity K is defined as,

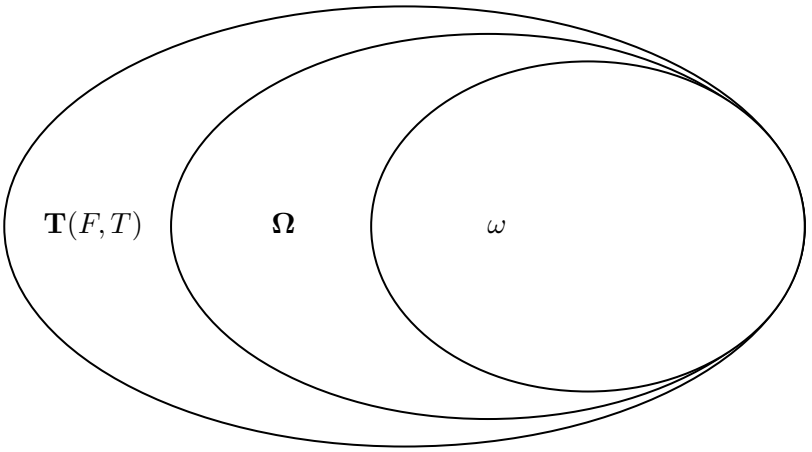


FIGURE 3.4: Hierarchy of expression spaces.

$$K = \frac{1}{\mathrm{P}(S(P(e \in \mathbf{E})) = f^*)} = \frac{\mu(\mathbf{E})}{\mu(\omega_E)} \tag{3.11}$$

where ω_E is the set of encodings that represent the solutions to the differential equations. In some instances it will more straightforward to work with this quantity, as $\mu(\mathbf{E})$ can be more straightforward to evaluate than $\mu(\mathbf{\Omega})$.

Fitness space

In order to translate a search problem into an optimisation problem, there must be an objective function to be guided towards an optimum; a mapping from the function space to the real line,

$$J : \mathbf{C}^r \to \mathbb{R} \tag{3.12}$$

In the language of evolutionary computing, this mapping is referred to as the *fitness function*, with its value for any argument referred to as the fitness of that argument. The selection of the fitness function is motivated in practice by a desire to guide the optimisation procedure smoothly towards an optimum. For a given trial function, there are several criteria that must be considered.

- If a trial solution agrees with a numerical solution of the ODE ($f = f*$).
- If a trial solution represents a solution to the ODE ($\Psi(f) = 0$).
- Whether or not a trial solution respects any boundary/initial conditions.

Because one has access to a numerical form for f, each of the above may be assessed by the application of an error metric, for example the $\mathcal{L}^2$-norm. The overall optimisation problem is therefore multi-objective and relative weights between the terms must be selected.

In addition to the numerical terms above, in order to check for exact solutions, a symbolic measure of the solution is also required. This can be implemented by substituting the trial function into the ODE *symbolically* to assess agreement. If by this method, a true solution to the ODE is found, then the optimisation run can be terminated.

Search operations

With the haystack assembled, how best to move around within it to search for a needle? As with any optimisation scheme, methods for exploring the search space must be identified. It is clear that the choice of search operations will be strongly affected by the choice of encoding scheme. For example, if an encoding space is comprised of vectors of integers, then vector crossover and integer perturbation can be applied. If instead, strings are used then different methods must be specified.

A common choice of representation scheme in the literature is a tree encoding [70, 88, 133]. Within such a space, common choices for search operations in this space are *tree crossover* and *subtree mutation*. For a full treatment of search operations in tree search spaces, the interested reader is directed to [70].

3.2 Heuristics

Considering the above, any optimisation-based approach to searching for solutions to a differential equation must contain certain fundamental elements:

- An encoding space (Genotype): $\mathbf{E}$.
- A decoding function (Phenotyping function): $P : \mathbf{E} \to \mathbf{\Omega}$.
- A functional interpretation scheme: $S : \mathbf{\Omega} \to \mathbf{C^r}$.
- An objective function (Fitness function): $J : \mathbf{C^r} \to \mathbb{R}$.
- One or several perturbing search operations (Genetic operations): $M_i : \mathbf{E} \to \mathbf{E}$.

What properties must these elements have in order for the search to have the best chance of success in finding the solution to Ψ? This is an open question in the literature that

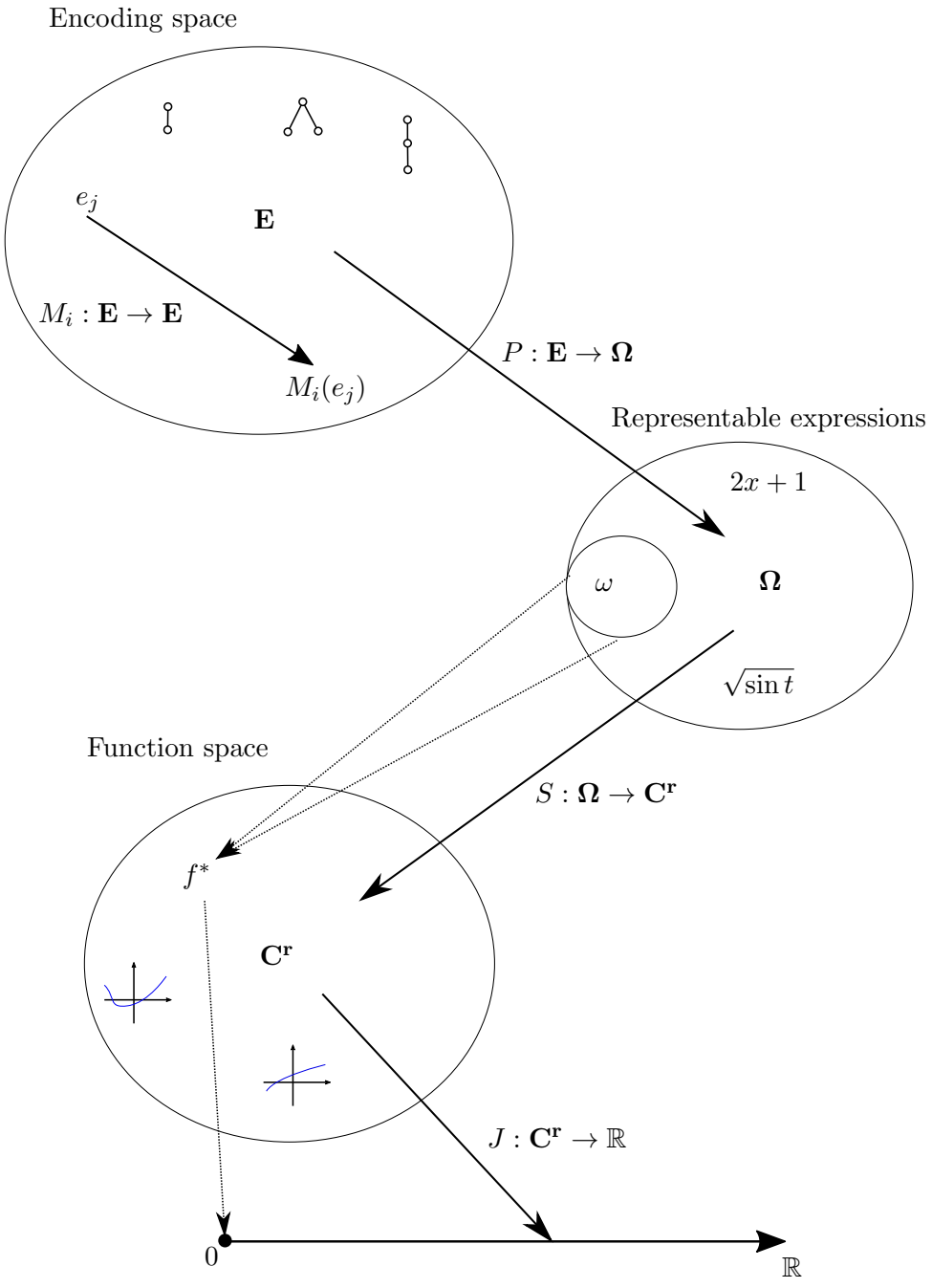


FIGURE 3.5: Visualising the overall optimisation problem.

has received little attention. The author is aware of two criteria that has been considered in the context of differential equation solutions; those of complexity [85], and the notion of locality [134].

Complexity

The unguided and total complexity of the search problem are defined in (3.10) and (3.11) respectively. For a given Ψ, this involves the selection of the encoding space and phenotyping function such that the quotients,

$$k = \frac{\mu(\mathbf{\Omega})}{\mu(\omega)} \tag{3.13}$$

and,

$$K = \frac{\mu(\mathbf{E})}{\mu(\omega)} \tag{3.14}$$

are minimised. Intuitively, this is akin to maximising the number of needles and minimising the amount of hay. Practically, this can be achieved by selecting a compact search basis containing as few operations as possible. This, inevitably leads to the paradox of knowing which search terms to include without knowing the form of the solution *a priori*. However, engineering judgment is applicable here.[5] If one is searching for the solution for a nonlinear ODE with oscillatory behaviour, then it is reasonable to include sinusoids. If there are trigonometric constants in the ODE then these should be included in the search basis. ODEs of the Duffing-type should include the Jacobi elliptic functions, and so on.

Locality

Locality is a measure of continuity of search space that was introduced in the evolutionary computation literature by Rothlauf [135]. Inspired by continuous-space optimisation, the idea of locality is to ensure that nearby points in the encoding space are mapped to nearby points in the fitness domain; this means that if e and e' are adjacent points in the encoding space $\mathbf{E}$, then in mapping these points to the fitness domain $\mathbb{R}$, they return similar objective scores. Formally,

[5] It might seem that returning to engineering judgement here runs the risk of confining the search to problems for which the solution is already known. The author would make the argument that the choice of a basis in symbolic regression still affords extreme flexibility in terms of the expressions that can be represented.

$$|J(e) - J(e')| \leq \epsilon \tag{3.15}$$

where ϵ is some small value. The motivation for this requirement is that it naturally promotes smoothness in the geometry of the search space. Smoothness is desirable in an optimisation context as it reduces sharp local minima (in which routines can become trapped) and promotes easier optimisation surfaces. These properties enable the search operations to out-perform random search.

Several formal definitions of locality have been provided (the interested reader is directed to [134] for a thorough treatment). A pervasive issue is how to measure the notion of adjacency in the encoding space.

One approach might be to use a tree-distance metric [136]. However, two very similar trees might represent two very different functions (consider the effect of replacing a sine node with an exponential, say). Instead, the approach here will be to consider the adjacency in the encoding space in terms of the perturbing search operations (M_i).

Thus, locality can be measured by the application of a single perturbing search operation in the encoding space. For a search operation M_i the locality of that operation in the search space ($\ell(M_i)$) is given by,

$$\ell(M_i) = \frac{\sum_{\mathbf{E}} \left(P(e_i) - M_i(P(e_i))\right)^2}{\mu(\mathbf{E})} \tag{3.16}$$

In practice, enumerating this quantity is not feasible as it requires sampling from every point in the encoding space. In addition, the search operations M_i are often stochastic in nature and so the expectation of the numerator must be approximated. Some statistical results in this vein are available in [134], but these are not considered further. Owing to the difficulty in calculation, the locality heuristic is employed hereafter in a qualitative manner.

Encouraging locality motivates the selection of search operations that are able to make small continuous perturbations to the underlying function. Clearly this is not possible for all cases (it is difficult to imagine how a sine might be continuously mapped to an exponential), but for some sub-structures and constants the idea of a *local* perturbation is intuitive. The notion of locality in the encoding space will be used as motivation for a new encoding scheme in the following chapter.

3.3 Enumerating search spaces

Both of the heuristics identified above rely on enumeration of the search space sizes $\mu(\mathbf{E})$ and $\mu(\mathbf{\Omega})$ in their computation. Indeed, the size of the search space is itself a useful heuristic for understanding the difficulty of a search problem. Because of this, it is useful to enumerate and compare the sizes of some common search spaces.

A common choice [70, 88, 133] for $\mathbf{E}$ is to use a tree structure. This approach has the convenient property that all mathematical expressions can be mapped in an isomorphic way to a tree object,

$$\mathbf{E}_{\text{tree}} \simeq \mathbf{\Omega}_{\text{tree}} \tag{3.17}$$

and therefore the spaces have the same measure,

$$\mu(\mathbf{E}_{\text{tree}}) = \mu(\mathbf{\Omega}_{\text{tree}}) \tag{3.18}$$

The size of the search space of trees is discrete but naturally not finite. Practically however, one places a restriction on the size of tree structures that are permitted. Several approaches are possible including limits on tree depth or internal nodes [71]. However, the approach here will be to restrict the total number of nodes in the expression tree. This has the advantage of not subjecting any bias towards either very deep or very wide trees. Another advantage of using the total number of nodes is that it is equivalent to maximum tree depth by the relation $n = m^h$ where n is the number of nodes, m is the maximum number of child nodes connected to any given node (hereafter referred to as the *arity* of that node) and h is the tree depth.

The maximum arity of any node defines the arity of the tree structure. The number of unlabelled m-ary trees with exactly n nodes is given by the Fuss-Catalan numbers [137],

$$C_n = \frac{1}{(m-1)n+1}\binom{mn}{n} \tag{3.19}$$

However, the nodes in an expression tree representation are not unlabelled. Instead, each node of arity $i \in [0, \ldots, m]$ derives a label from set f_i, the edges are unlabelled. Thus, the overall label set,

$$F = \{f_0, f_1, \ldots, f_i, \ldots, f_m\} \tag{3.20}$$

is the *search basis* as above. In the literature, the case $i = 0$ is sometimes considered separately with 0-ary nodes referred to as *terminals* or *leaves* and $f_0 = T$ referred to as the terminal set. For compactness, this notation will not be used here.

Implementations of the search problem must necessarily select a basis that contains all the mathematical objects needed to describe the solution, or only approximate solutions will be possible.

The size of the tree space clearly depends on the maximum arity m, the maximum number of nodes n and the basis set F. Let,

$$T_j = \mu\left(\mathbf{T}(m, j, F)\right) \tag{3.21}$$

be the number of possible node-labelled m-ary trees with basis F comprised of exactly j nodes. By considering trees of all sizes up to n, one has,

$$\mu(\mathbf{E}_{\text{tree}}(m, n, F)) = \sum_{j=0}^{n} T_j \tag{3.22}$$

In order to derive this quantity, consider the case $n = 1$. Since this can only be a single node with no children, the number of possible labels (and therefore trees) is,

$$T_1 = f_0 \tag{3.23}$$

Next, consider the case $n = 2$; there is still only a single possible tree structure (one with one root and one child). The number of such trees is equal to the number of label combinations,

$$T_2 = f_1 f_0 \tag{3.24}$$

This process is displayed graphically up to $n = 4$ in Figure 3.6. A recurrence relation can now be derived for the case $n = k$. In order to simplify the notation, the following derivation will continue with the case $m = 2$. For basis functions of analytical expressions this is a realistic restriction as there are few common analytical expression operations with an arity greater than two (an example might be a summation with limits on indices - technically a trinary operation but these are not often seen in exact solutions to differential equations).

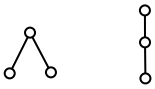
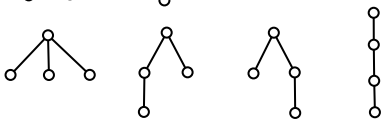
n	m		Possible trees
1	0		f_0
2	1		$f_1 f_0$
3	2		$f_2 f_0^2 + f_1^2 f_0$
4	3		$f_3 f_0^3 + f_2 f_1 f_0^2 + f_2 f_1 f_0^2 + f_1^3 f_0$

FIGURE 3.6: Possible labelled trees with up to $n = 4$ nodes.

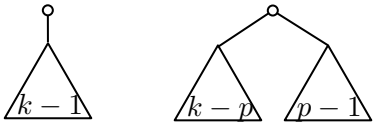


FIGURE 3.7: Possible trees with $m = 2, n = k$, $p \in [2, k-1]$.

In the case $m = 2, n = k$ there are two important cases to consider, as shown in Figure 3.7. In the first, the root node is unary. Its single argument is a subtree with $k-1$ nodes. In this case there are,

$$T_{k,\text{unary}} = f_1 T_{k-1} \tag{3.25}$$

possible trees. In the other case the root node is binary and the number of nodes in its arguments sum to $n-1$. In performing this sum one has,

$$T_{k,\text{binary}} = f_2\{T_{k-2}T_1, T_{k-3}T_2, \ldots, T_1 T_{k-2}\} = f_2 \sum_{p=2}^{k-1} T_{k-p}T_{p-1} \tag{3.26}$$

The required relation is now the sum of these two possibilities,

$$T_k = T_{k,\text{unary}} + T_{k,\text{binary}} = f_1 T_{k-1} + f_2 \sum_{p=2}^{k-1} T_{k-p}T_{p-1} \tag{3.27}$$

Given $T_1 = f_0$ and $T_2 = f_1 f_0$ from above, one can construct a recursive scheme to calculate any $\mu(\mathbf{E}_{\text{tree}}(2, n, F))$, and thus the number of structurally-unique expressions $\mu(\mathbf{\Omega}_{\text{tree}})$,

$$\mu(\mathbf{E}_{\text{tree}}) = \mu(\mathbf{\Omega}_{\text{tree}}) = \sum_{k=1}^{n} \left[f_1 T_{k-1} + f_2 \sum_{p=2}^{k-1} T_{k-p} T_{p-1} \right] \tag{3.28}$$

Another common representation scheme is a grammar-based structure [83, 87]; these are employed by several authors alongside techniques such as modular arithmetic to encode expressions as vectors of integers. The search space of grammar-based representations can conveniently be enumerated by considering the number of unique tree structures. This process gives a better estimate of $\mu(\mathbf{\Omega}_{\text{grammar}})$ than simply raising the maximum integer value to the power of the vector length (i.e $\mu(\mathbf{E}_{\text{grammar}})$). This advantage arises because the modular arithmetic removes the need for a maximum integer value resulting in a many-to-one mapping via the decoding function. Practically, this means that unlike the expression tree representation,

$$\mu(\mathbf{E}_{\text{grammar}}) \gg \mu(\mathbf{\Omega}_{\text{grammar}}) \tag{3.29}$$

In terms of heuristics, comparing equivalent tree space sizes permits a better comparison between representation schemes. Since grammar-based structures sometimes include trinary operations, an additional term in equation (3.27) is required. The number of distinct trees with k nodes is now,

$$\mu(\mathbf{T}_{\text{grammar}}(3, k, F)) = f_1 T_{k-1} + f_2 \sum_{p=2}^{k-1} T_{k-p} T_{p-1} + f_3 \sum_{q=2}^{k-2} \left[T_{q-1} \sum_{p=q+1}^{k-1} T_{k-p} T_{p-q} \right] \tag{3.30}$$

However, there is some subtlety here; several of the nodes included in the grammar do not alter the underlying function mathematically and only act as placeholders for the decoding function P. Such nodes include the 'expression' and 'operation' nodes. In considering $\mu(\mathbf{\Omega}_{\text{grammar}})$, these meta-nodes can be safely collapsed (i.e ignored) in the computation.

Figure 3.8 is a plot of $\mu(\mathbf{\Omega}_{\text{tree}})$, $\mu(\mathbf{T}_{\text{grammar}})$ and $\mu(\mathbf{\Omega}_{\text{grammar}})$ with increasing n. The tree-space data are generated with a basis set of operations given by,

$$\begin{aligned} F_{\text{tree}} = \{ & \{x, [0, 9]\}, \\ & \{\sin, \cos, \log, \exp\}, \\ & \{+, -, \times, \div\}\} \end{aligned} \tag{3.31}$$

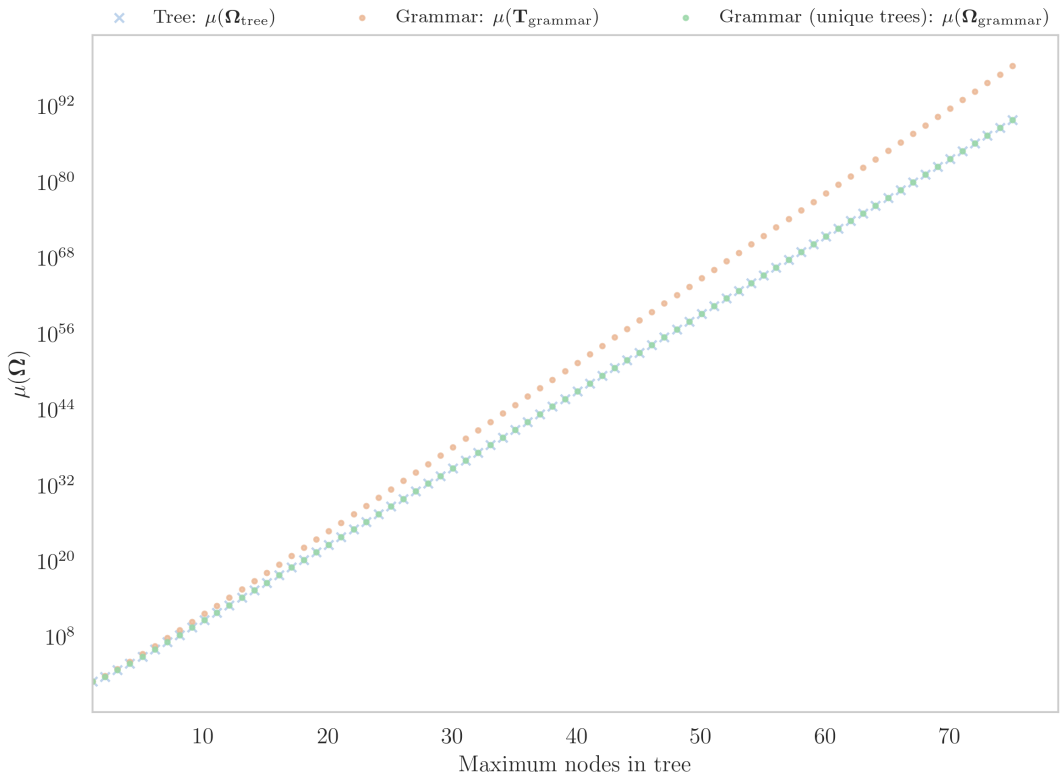


FIGURE 3.8: Size of various spaces with increasing number of nodes n.

for the expression trees (indicative of the approach suggested by [88]). The grammar-space data are generated with $m = 3$ and basis set,

$$\begin{aligned} F_{\text{grammar}} = \{&\{x, [0, 9]\}, \\ &\{\sin, \cos, \log, \exp, < \text{expr} >, < \text{digit} >\}, \\ &\{+, -, \times, \div, < \text{func} >\}, \\ &\{< \text{op} >\}\} \end{aligned} \tag{3.32}$$

for the grammar representation (indicative of the study by Tsoulos et al. in [83]). The meta-operations (in angle brackets) are ignored in the computation of $\mu(\mathbf{\Omega}_{\text{grammar}})$, resulting in an identical basis. As can be seen in Figure 3.8, the size of these spaces grows extremely quickly with the number of nodes in the tree[6]. The number of trees in grammar-based approaches appears to grow more quickly because of the trinary operation. However, it can be seen that the number of expression trees representable by these encodings is equivalent.

[6]Quickly exceeding both estimated number of particles in the observable universe and the recursion limit of the authors laptop!

TABLE 3.1: ODE problems of Tsoulos and Lagaris [83].

#	ODE	Domain	Subject to
1	$f' - \frac{2t-f}{t} = 0$	$t \in [0.1, 1]$	$f(0.1) = 20.1$
2	$f' - \frac{1-f\cos(t)}{\sin(t)} = 0$	$t \in [0.1, 1]$	$f(0.1) = \frac{2.1}{\sin(0.1)}$
3	$f' + \frac{1}{5}f - e^{\frac{-t}{5}}\cos(t) = 0$	$t \in [0, 1]$	$f(0) = 0$
4	$f'' + 100f = 0$	$t \in [0, 1]$	$f(0) = 0, f'(0) = 10$
5	$f'' - 6f' + 9f = 0$	$t \in [0, 1]$	$f(0) = 0, f'(0) = 2$
6	$f'' + \frac{1}{5}f' + f + \frac{1}{5}e^{\frac{-t}{5}}\cos(t) = 0$	$t \in [0, 2]$	$f(0) = 0, f'(0) = 1$
7	$f'' + 100f = 0$	$t \in [0, 1]$	$f(0) = \sin(10), f'(0) = 0$
8	$tf'' + (t-1)f' + f = 0$	$t \in [0, 1]$	$f(0) = 1, f(1) = 0$
9	$f'' + \frac{1}{5}f' + f + \frac{1}{5}e^{\frac{-t}{5}}\cos(t) = 0$	$t \in [0, 1]$	$f(1) = \frac{\sin(0.1)}{e^{\frac{1}{5}}}, f'(0) = 1$

TABLE 3.2: Exact solutions to the ODE problems of Tsoulos and Lagaris [83].

#	Exact solution
1	$f = t + \frac{2}{t}$
2	$f = \frac{t+2}{\sin(t)}$
3	$f = e^{\frac{-t}{5}}\sin(t)$
4	$f = \sin(10t)$
5	$f = 2te^{3t}$
6	$f = e^{\frac{-t}{5}}\sin(t)$
7	$f = \sin(10t)$
8	$f = 1 - t$
9	$f = e^{\frac{-t}{5}}\sin(t)$

3.4 On benchmarks for exact symbolic regression

An important consideration when assessing potential methods for performing exact symbolic regression is a suite of benchmark problems upon which methodologies can be compared. In the literature, there is little in the way of standardised benchmark problems for comparison. Perhaps the closest available is the suite of problems from Tsoulos and Lagaris [83]. In their paper, the authors propose several suites of problems including ODEs, SODEs and PDEs.

Considered here is the suite of nine ODE problems reproduced in Table 3.1. The solutions for the Tsoulos and Lagaris ODE problems (TL benchmark) are given in Table 3.2.

Several studies have presented results on the TL benchmark (including [85]). However, it is the opinion of the author that there are several limitations to this benchmark. The first is that the ODE problems vary dramatically in their difficulty (as shall be demonstrated in a later chapter). This limitation is problematic because there is no insight provided into exactly what it is that makes one ODE more difficult to solve than another in the context of symbolic regression.

Another problem with the TL benchmark is that there is no unified metric of performance. In their paper, Tsoulos and Lagaris [83] report the average number of iterations used by their method, in [85], the results (of a slightly altered benchmark set) are reported in terms of the unguided and guided complexities.

These limitations are hereby used to motivate the specification of a new benchmark for solving ODE problems by the method of symbolic regression. To begin, a consistent metric of performance is established. One issue in reporting the average number of function evaluations (NFEs) is how to treat runs where no exact solution is found. To overcome this, the right-censored maximum likelihood estimate of the average number of NFEs is used. Denoting this quantity as Λ, and assuming R runs of an algorithm with N function evaluations per run,

$$\Lambda = \frac{1}{R^*} \sum_{i}^{R} n_i \tag{3.33}$$

where R^* is the number of times the exact solution is found in R trials and,

$$n_i = \begin{cases} n_i^*, & \text{if exact solution found in } n_i^* < N \text{ NFEs} \\ N, & \text{otherwise} \end{cases} \tag{3.34}$$

It is noted here that the one major factor in the difficulty of the solution is the size of the minimum tree that can represent the solution. For example, the equation $f = 1 + x$ is much easier to identify than $f = e^{-5x} \sin(3x - 2)$. For this reason, the proposed benchmark will feature a suite of ODE problems, the solution to each of which corresponds to a minimum tree representation of n nodes, where $n \in [1, 8]$. As shall be seen, choosing solutions with tree sizes in this range will provide a better gradation of difficulty. The computational difficulty of finding exact solutions with tree representations larger than 8 nodes grows very quickly. For this reason, no larger solutions are considered in this benchmark.[7] It is also important to define domains that capture all the pertinent dynamics of the function of interest. With these considerations in mind, a suite of problems is proposed in Table 3.3.

An interesting consideration is the effect of the form of the solution versus the form of the ODE in the search problem. In order to control for this effect, a number of ODE formulations are presented in Figure 3.4.

[7] A natural question is how many nodes might be required for the solution of as-yet unsolved nonlinear ODEs. At this stage it is hard to envisage a satisfactory answer. However, a natural starting place is to consider tree sizes that can be found with the computational resources available to the author.

TABLE 3.3: Function solutions for proposed ODE benchmark.

n	f^*	Domain	Subject to
1	t	$t \in [0, 10]$	$f(0) = 0, f'(0) = 1$
2	$\sin(t)$	$t \in [0, 12]$	$f(0) = 0, f'(0) = 1$
3	$5t$	$t \in [0, 5]$	$f(0) = 0, f'(0) = 5$
4	$\cos(t - 1)$	$t \in [0, 12]$	$f(0) = \cos(1), f'(0) = \sin(1)$
5	$e^t cos(t)$	$t \in [0, 5]$	$f(0) = 1, f'(0) = 1$
6	$8t/e^t$	$t \in [0, 10]$	$f(0) = 1, f'(0) = 8$
7	$2t/9t$	$t \in [0, 8]$	$f(0) = 0, f'(0) = 2/9$
8	$2\sin(t) + t/3$	$t \in [0, 8]$	$f(0) = 0, f'(0) = 2 + 1/3$

TABLE 3.4: ODEs used in the proposed benchmark.

ODE	Homogenous ODE $\Psi(f)$
Trivial linear	f'
Second order linear	$f'' + f' + 2f$
Duffing-type	$f'' + f' + 2f + 8f^3$

Note that any homogenous ODE of the form $\Psi(f) = 0$, can be transformed into an inhomogeneous form Ψ' with an arbitrary solution f^* as,

$$\Psi'(f) = \Psi(f) - \Psi(f^*) \tag{3.35}$$

In this manner, the proposed benchmark now consists of 24 ODE problems, of controlled difficulty via the complexity of the solution and the complexity of the ODE.

3.5 Conclusions

This chapter has presented the problem of finding solutions to ODEs as a search problem. Here, the notation of [85] has been extended to consider formally the various search spaces and the maps between them.

Also considered, are a number of heuristics that encode information pertaining to the difficulty of the underlying search problem. Each of these heuristics is used to motivate practical choices that can be made when designing an optimisation procedure. The first of these—complexity—is used to motivate a compact search basis. The second—locality—is used to motivate encoding spaces and search operations that promote a continuous optimisation surface. A limitation of the analysis presented here is that these quantities are not enumerated for any concrete problems. Several works [85, 134] present numerical calculations of these metrics in contexts including symbolic regression and beyond. The interested reader is directed to those works for more details.

This chapter has also presented an enumeration of the search space sizes for two commonly-used encoding spaces in symbolic regression; expression grammars and expression trees. It is shown that, for an identical search basis, the two encodings are able to represent the same number of expressions, however, the expression grammar has a larger encoding space because of the presence of auxiliary nodes in the trees.

Finally, symbolic regression benchmarks are discussed. One such benchmark found in the literature (that of Tsoulos and Lagaris) is considered, and several weaknesses are identified. In the light of these limitations, a new benchmark suite of ODE problems is proposed, alongside a consistent metric of performance for symbolic regression algorithms.

Chapter 4

Affine-regression trees for exact symbolic regression

The objective of this chapter is to evaluate the performance of several symbolic regression approaches, to solving nonlinear ODEs on benchmark problems. This analysis is conducted with a view to identifying promising techniques that might be able to search heuristically for the solution to pertinent nonlinear ODEs typical of those seen in structural dynamics.

Inspired by the notion of *locality* (a local map between genotypes and objective scores), this chapter also presents a novel encoding scheme, the *affine symbolic regression tree*. Here, the affine-regression tree encoding is developed and the sizes of the relevant search spaces are calculated and compared to existing approaches.

The new encoding scheme is introduced with with a specific focus on specifying exact solutions to ODE problems, in order to assess this the performances of the novel method is compared to two common expression representations; the tree based encoding and to the grammar based approach of Tsoulos and Lagaris in [83]. The approaches are compared on two benchmark suites of ODE both linear and nonlinear ODE problems.

4.1 Affine-regression trees

A novel encoding scheme for symbolic regression is proposed here. This encoding is essentially an extension of the expression-tree encoding space with additional structure at each node.

The new approach is inspired by two observations. Firstly, the exact discovery of constants in symbolic regression schemes is an open problem that has not been significantly explored in the symbolic regression literature[1]. The second observation is the promotion of locality. For many existing encoding schemes, adjacent points in $\mathbf{E}$ are unlikely to be adjacent in $\mathbb{R}$ once projected there via the objective function. Put plainly, it is often not trivial for the optimisation algorithm to increment towards the true solution despite a high degree of *semantic similarity* to the solution. An intuitive way to understand semantic similarity is the following. Consider an ODE with the solution,

$$f^* = \sin(2x) \tag{4.1}$$

Now consider two trial expressions,

$$f_1 = 2x - \frac{4}{3}x^3 + \frac{4}{15}x^5 - \frac{8}{315}x^7$$

$$f_2 = \sin(3x)$$

Evaluated over a domain of $x \in [-1, 1]$, an objective function (the $\mathcal{L}_2$ norm for example), will prefer f_1 over f_2 despite the fact that *semantically*, f_2 is far closer to the true solution.

The affine symbolic regression tree is defined in the same manner as the expression tree. The distinction is that each node η is now a 3-tuple,

$$\eta = \{a, f, b\} \tag{4.2}$$

Where a and b are termed *constants*. During evaluation, nodes take the affine form,

$$\eta = af + b \tag{4.3}$$

In this regard, the representation bears some similarity to the multiple regression approach in [138], whereby linear combinations of all tree subexpressions are used during a meta-optimisation step. However, the current approach differs both in that an affine combination is used and in that the constants are included as a part of the tree structure itself.

[1]There are several notable techniques that specify constants approximately - the reader is directed to [79, 138, 139] for examples.

TABLE 4.1: Constant mutation operations proposed by the current study.

Operation	Action	Description
$m_0(\theta)$	$\theta \to 0$	Zero constant
$m_1(\theta)$	$\theta \to 1$	Unit constant
$m_i(\theta)$	$\theta \to \theta \pm 1$	Increment/decrement
$m_{r1}(\theta)$	$\theta \to \frac{p\pm1}{q}$	Increment/decrement numerator
$m_{r2}(\theta)$	$\theta \to \frac{p}{q\pm1}$	Increment/decrement denominator
$m_s(\theta)$	$\theta \to s\theta, s \in S$	Multiply by transcendental constant
$m_r(\theta)$	$\theta \to \theta^r, r \in R$	Raise to exponent
$m_{pm}(\theta)$	$\theta \to -\theta$	flip sign of constant

Constants have their own structure and are represented by a 4-tuple,

$$\theta = \{s, p, q, r\} \tag{4.4}$$

where p and q are integers in the range $[0, z]$ and $[1, z]$ respectively. r is a member of the set of permitted exponents R with a default value of 1. s is an element from the permitted transcendental constants S, also with a default value of 1. Upon execution constants are evaluated as,

$$\theta = s\left(\frac{p}{q}\right)^r \tag{4.5}$$

Upon initialisation, a constants are given a value of 1 ($\{1, 1, 1, 1\}$) and b constants a value of zero ($\{0, 1, 1, 1\}$). Initialising the constants in this way provides a bias towards sparse solutions. Care is taken to ensure that $q \neq 0$ so that illegal divisions are avoided by design.

Constants are mutated during the run of the search by a number of constant mutation operations that are defined in addition to more orthodox tree-based search operations (the reader is directed to [133] for a reference). These constant mutation operations are described in Table 4.1.

The advantages of this representation are several; consider the expression tree in Figure 4.1. The affine tree is far more compact, requiring only a fraction of the number of nodes to represent the same expression. In fact, with affine trees there is no need to include integers or other constants in the basis set F at all. Constant discovery is handled entirely by the affine constant objects.

The size of the resulting search space can readily be estimated by extending the analysis of expression trees in the previous chapter. The first step is to enumerate the number

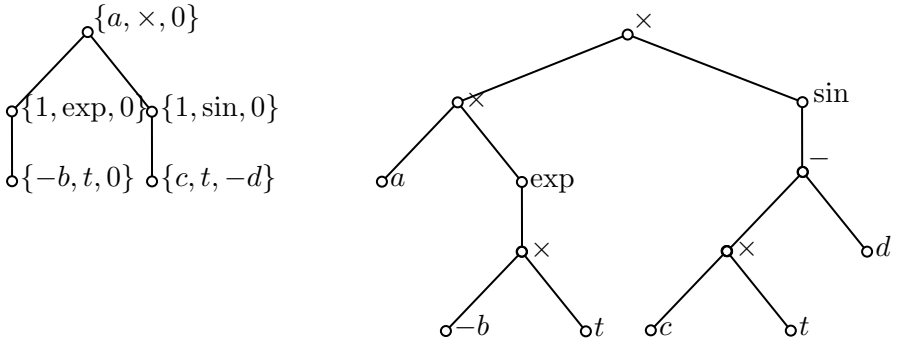


FIGURE 4.1: Comparison between compact affine and expression trees for the expression $f = ae^{-bt}\sin(ct - d)$.

of possible values that an affine constant $\theta \in \mathbf{A}$ can take; this is achieved by considering the elements in the tuple,

$$\mu(\mathbf{A}) = \mu(P)\mu(Q)\mu(R)\mu(S) = z(z+1)\mu(R)\mu(S) \tag{4.6}$$

Since there are two constants per node, the expression for $\mu(\mathbf{\Omega}_{\text{affine}})$ can be written,

$$\mu(\mathbf{\Omega}_{\text{affine}}) = \sum_{k=0}^{n} \left[(z(z+1)\mu(R)\mu(S))^{2k} T_j \right] \tag{4.7}$$

or explicitly,

$$\mu(\mathbf{\Omega}_{\text{affine}}) = \sum_{k=0}^{n} \left[(z(z+1)\mu(R)\mu(S))^{2k} \left[f_1 T_{k-1} + f_2 \sum_{p=2}^{k-1} T_{k-p} T_{p+1} \right] \right] \tag{4.8}$$

Does size matter?

Comparing the above result to the results of Chapter 3 seems to suggest that this approach is strictly worse (i.e larger in terms of search space size) than expression trees. However, this is a false equivalence. While the affine tree representation is at least as compact as the expression tree representation, in many cases it will be significantly more so. In practice, one is able to select a lower value of n, when working with the affine representation, and still maintain the same coverage of $\mathbf{C}^r$. Figure 4.2 compares several scenarios in which the affine representation is more compact and plots them against the number of nodes required for an affine representation. The scenarios shown are illustrated in Figure 4.3.

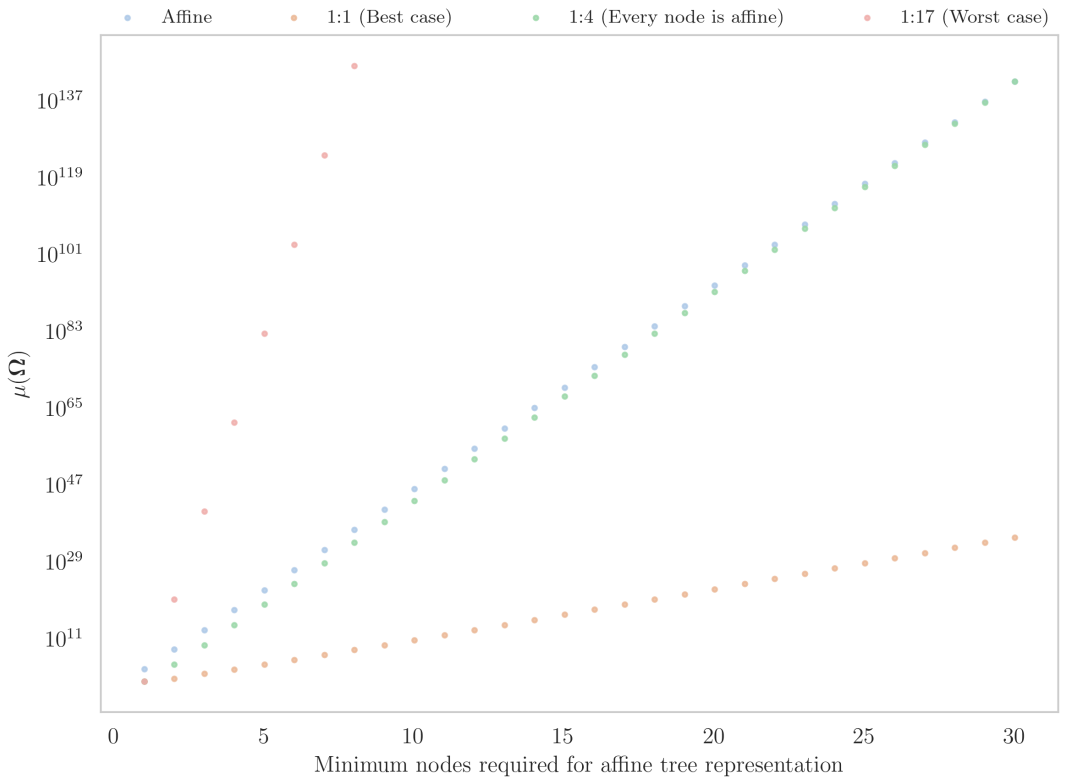


FIGURE 4.2: Comparison of affine and expression trees in terms of search space size versus minimum number of nodes required for affine representation.

The results of Figure 4.2 indicate that the affine representation has a smaller search space for expressions that require four times as many nodes to represent as an expression tree. For trivial expressions with only a few elementary integer constants, such a ratio is perhaps unrealistic. However, the author would argue that for the type of expressions seen in the solutions to ODEs in structural dynamics (such as the one depicted in Figure 4.1), a 1:4 ratio is more likely. The author would note that the tree representations in Figure 4.1 assume that the constants $a, -b, c, -d$ are specified in the search basis. In reality, an expression tree would require further subtrees to represent these values (and their negatives) adding yet more complexity to the expression tree.

The advantages of the affine tree representation extend beyond the compactness of the search space. The constant-mutations described in Table 4.1, permit a continuous optimisation surface between expressions of the correct form (only errors in constants) and the true solution. An informal explanation is thus. Consider any two affine trees of the same structure α_1 and α_2 (but with different values of the constants a and b) with objective function values j_1 and j_2 respectively such that $j_2 < j_1$. There must then be a chain of constant mutation operations $[m_1, \ldots, m_v]$, the successive application of which will map α_1 onto α_2. It is furthermore argued that each application of the mutation

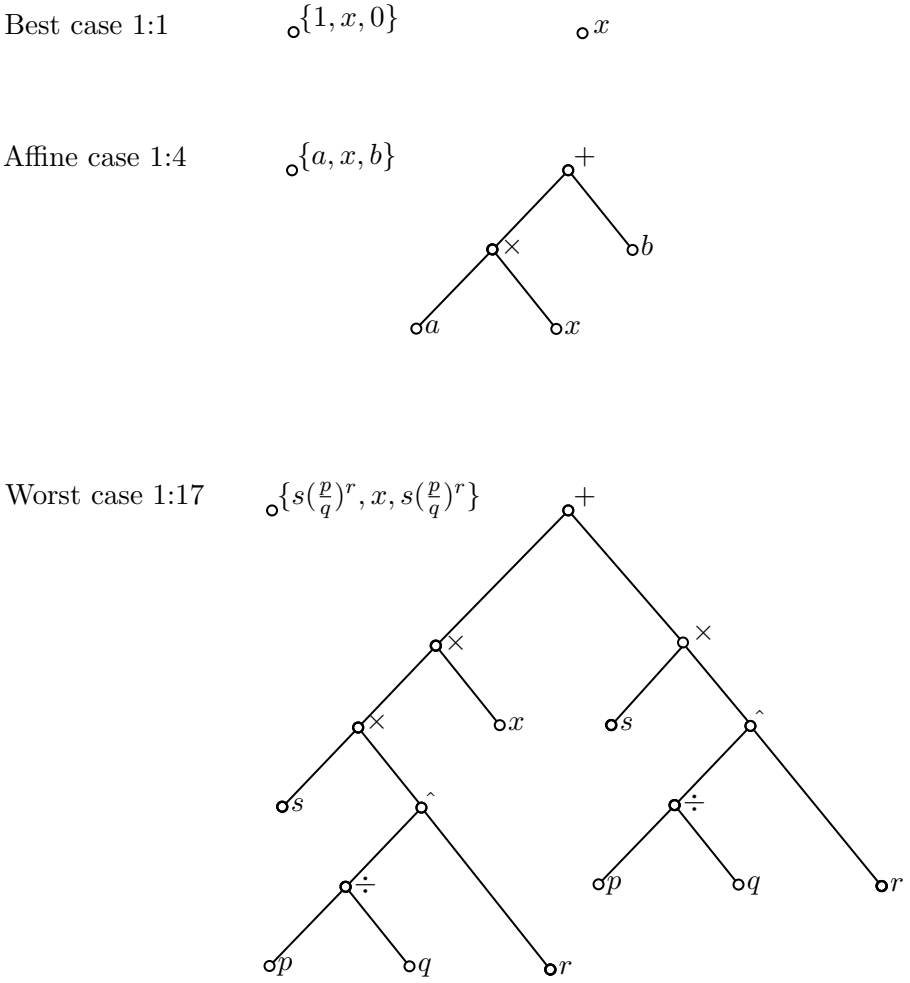


FIGURE 4.3: Comparison of the representation power of affine nodes compared to standard expression trees.

operations in the chain is more likely to result in a monotonically-decreasing objective score than an equivalent path based on orthodox search operations.

4.2 Methodology

In order to perform symbolic regression as optimisation, a suitable optimisation algorithm must be chosen. For search problems defined on discrete encoding spaces, several options are available, although a common choice is the genetic algorithm (GA) (also referred to as genetic programming [70, 133]). These algorithms are inspired by the biological process of evolution. The GA is a very flexible approach that encompasses a great deal of methodologies that can be tailored to the search problem at hand. As such, the GA approach will be adopted here. The algorithm consists of a population of candidate solutions (genotypes) to a search problem are iterated over a number of

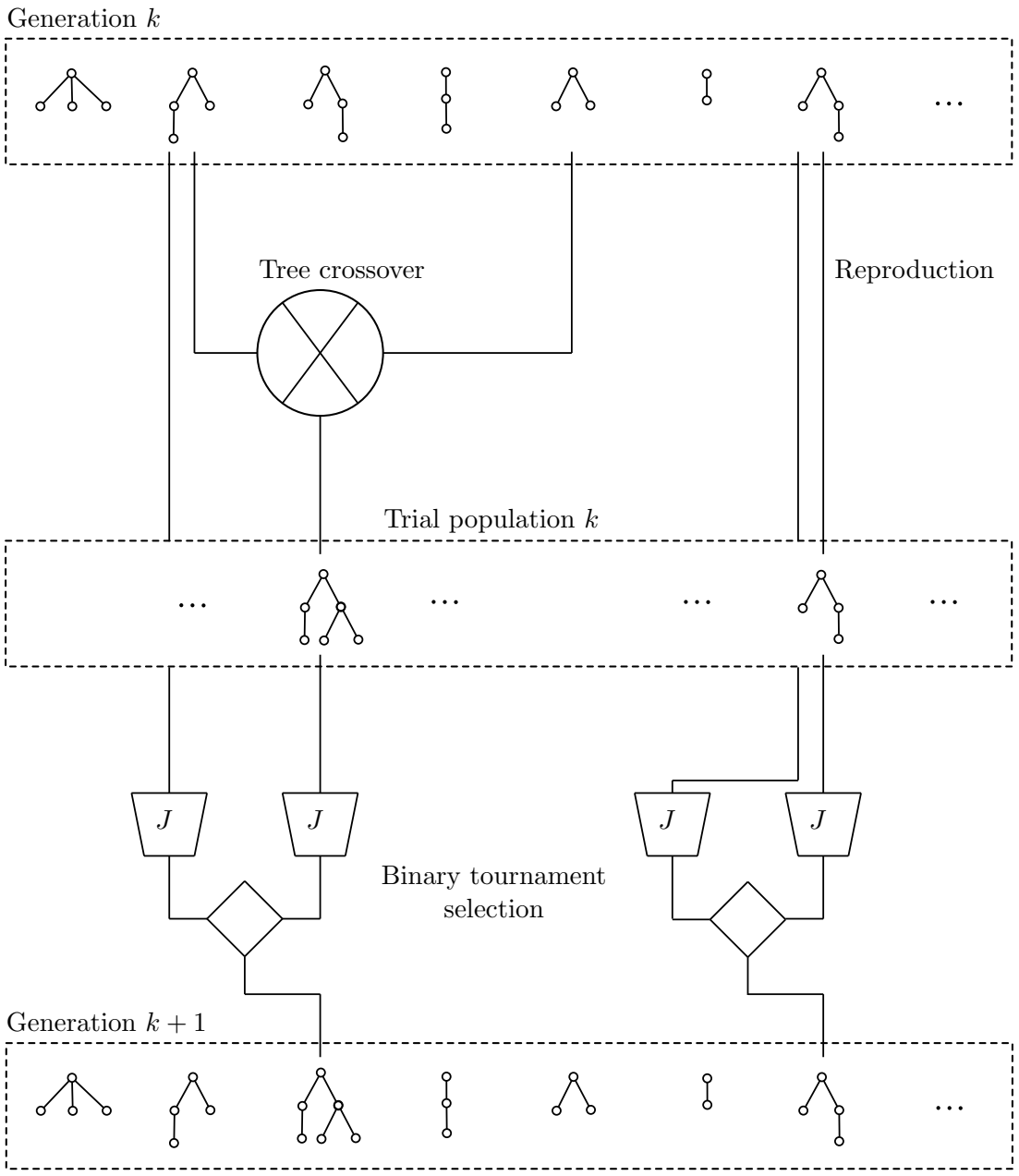


FIGURE 4.4: Overview of the genetic algorithm implemented for this chapter.

generations towards a global optimum by the application of search methods (genetic operations). During the optimisation, candidate members of the populations are allowed to advance to the next generation depending on their objective scores (phenotype).

A standard GA is comprised of an initialisation step, a reproduction step and a selection step. An overview of the GA used in this study is given in Figure 4.4.

Initialisation

During initialisation, an initial population of candidate solutions is sampled from the encoding space. Initialisation is implemented here by the recursive 'grow' method [133]. Nodes are sampled at random from the search basis F. If a node with an arity greater than zero is sampled, then additional nodes are sampled as the arguments of that node. This process is reclusively applied until either all terminal nodes have zero arity or until a pre-specified number of nodes have been selected. In the latter case, all terminal nodes are selected from the set of nodes with arity zero.

Reproduction

During reproduction, a number of genetic search operations are applied and a second population of solutions is produced. This second population is referred to as the *trial population*. The approach used here is to generate the trial population by sampling from a number of genetic operations. The operations considered in this investigation are:

- Reproduction: A randomly-selected member of the current population is added to the trial population.
- Tree crossover: Two randomly-selected parent trees are crossed over at random nodes, and the resultant tree is added to the trial population.
- Random tree: A random tree (generated by the 'grow' method above), is added to the trial population.
- Subtree mutation: A random node from a random parent is replaced with a random subtree.

In order to further aid the search, the probability of selecting one of the above search operations is adapted alongside the optimisation run. Adaption is a common approach in evolutionary optimisation to overcome the problem of hyperparameter selection [140]. The methods identified above are initially assigned equal probabilities and then after a period of generations L_p (referred to as the learning period), the (un-normalised) probability of selecting search method m_i is updated as,

$$p(m = m_i) = \frac{w_i + 1}{h_i + 1} \tag{4.9}$$

where h_i is the number of times search operation m_i was applied in the previous learning period of L_p generations and w_i is the number of successful entrants to the next generation in the same interval.

In the affine case, the constant mutation operations are applied to the trial population at this stage. Up to $n_{\text{mutations}}$ are applied with probability P_m. As above, the probability of applying any one of the methods detailed in Table 4.1 is adapted during the course of the optimisation run.

Selection

The trial vector is now established and the selection step is applied. In many applications of genetic algorithms seen in the literature, this is achieved by tournament selection [70, 133]. Here, binary tournament selection is used, whereby the objective scores of the current and trial populations are compared elementwise with the lower objective score being permitted to enter the next generation. One risk with tournament selection is that promising solutions (i.e with semantic similarity) might be discarded in favour of 'greedy' approximate solutions (series approximations etc.). However, it is hoped that this effect will be counteracted by the reproduction operation allowing some semantically prominsing less-optimal solutions to progress to subsequent generations.

In order to reduce the complexity of the search problem, some inductive biases are injected at this stage; trivial checks for illegal operations such as infinite terms or division by zero are employed before the evaluation of the objective function. In addition, to restrict the size of the search space, nested transcendental functions (sines, cosines and exponentials) are disallowed. In the case that such a nesting is produced by the action of the search operations, the trial expression is prevented from advancing to the next generation.

An objective function is selected with four terms. The function is defined as the weighted sum of the mean-squared error (MSE) over the ODE, the trial solution f, its first derivative f' and initial or boundary conditions. The target quantities $\hat{f}$ and $\hat{f}'$ are computed in advance of the run by a fixed-step fourth-order Runge-Kutta scheme with a step size determined by the domain of the target problem. For first-order problems, the MSE over f' is not used. The overall objective function is therefore,

$$J(f) = \frac{1}{d}\sum_{i}^{d}\Big[\lambda_1(\Psi(\hat{f}) - \Psi(f))^2 + \lambda_2(\hat{f}_i - f_i)^2 + \lambda_3(\hat{f}'_i - f'_i)^2 + \Big] + \lambda_4 \sum_j (I_j(\hat{f}) - I_j(f))^2 \tag{4.10}$$

TABLE 4.2: Parameters pertaining to the genetic algorithm used in the case study.

Parameter	Symbol	Value
Population size	N	400
Maximum generation	G	500
Initialisation procedure		'grow' [70]
Selection procedure		Tournament ($k = 2$)
Initial maximum tree depth	d_0	3
Maximum nodes in tree	n	10
Learning period	L_p	5

where $\Psi(f)$ is the ODE of interest and the I_j are the initial or boundary conditions applicable to that problem. The λ_i are the weights and are all set to unity with the exception of λ_4 which is set to 100. A computer algebra engine (in this case the symbolic library available in the python language *sympy* [69]), is used for the exact evaluation of derivatives and assessment of exact solutions.

Parameterisation

Parameters pertaining to the GA are collected for the convenience of the reader in Table 4.2. For both representations, a basis set given by,

$$\begin{aligned} F = \{&\{t,\}, \\ &\{\sin, \cos, \log, \exp\}, \\ &\{+, -, \times, \div\}\} \end{aligned} \tag{4.11}$$

is used. For the expression tree representations, the digits $[0, 9]$ are added to f_1. Initialisation, mutation and application of affine constants is as described in the previous section. In order to simplify the search, the sets of exponents R and transcendental constants S are set to $\{1\}$ and the corresponding affine mutations are excluded.

4.3 Results

Comparison of encoding schemes - TL benchmark

With the GA established, the results on the benchmark studies are presented here. The figures in this section, are formatted in the following manner. Dots in the plots, represent successful runs of the optimisation algorithm, whereas crosses represent runs

of the algorithm that did not find an exact solution. Diamonds in the figure represent the right-censured maximum likelihood estimates of the expected number of function evaluations required to find an exact solution, calculated as per (3.33) The uncertainty bars in the plots are the 90% confidence intervals of the above statistic, estimated by a bootstrapping method.

For a more rigorous comparison, the data from the Tsoulos and Lagaris study is included in the figures. For this dataset, only maximum, minimum and average values were recorded in the paper and so the confidence intervals can be interpreted as the 100% confidence intervals.[2] Results for individual runs from [83] are not reported and therefore these are omitted from the figure.

To aid with visualisation, the ODE problems have also been re-ordered in terms of search difficulty, as measured by the number of nodes in the maximally compact expression tree.

The first comparison made in this chapter is between encoding spaces. In Figure 4.5, the expression tree, affine-regression tree and the grammar-based encoding of Tsoulos and Lagaris [83] are compared in terms of the number of function evaluations required to find an exact solution. As can be seen in the figure, there are considerable differences between the encoding schemes. The affine encoding significantly outperforms the others on a subset of the easiest ODE problems. However, both affine and tree encodings quickly degrade with problem difficulty. The Affine tree approach is unable to find the exact solution in any of the runs for the three most difficult ODE problems, potentially indicating a poor convergence towards larger tree sizes. This degradation with problem difficulty is paradoxically not seen in the expression grammar approach, and it considerably outperforms the other methods on the harder ODE problems. Interestingly, Figure 4.5 does not show any evidence that the maximum number of nodes in the tree (i.e search space size) affects the MLE estimates of Λ.

Figure 4.6 depicts the same runs, compared to those where tree crossover is removed. This study investigates the extent to which the GA population is evolving cooperatively (whereby useful subtrees are shared between members of the populations). In the figure, there is no discernible difference between the runs with and without crossover indicating that each member of the population is evolving in isolation.

One issue with the TL benchmark, is that the ODE solutions are largely monotonic within the chosen domains. This is problematic because smooth monotonic functions can be approximated in myriad ways, leading to poor locality in the objective function. It is desirable that the solution domain captures the salient dynamics of the problem so

[2]There is additional uncertainty as to the number of function evaluations used in the Tsoulos and Lagaris study, as only the number of generations is reported and the population size is only given as an interval. These uncertainties have been included in the error-bars in the figures.

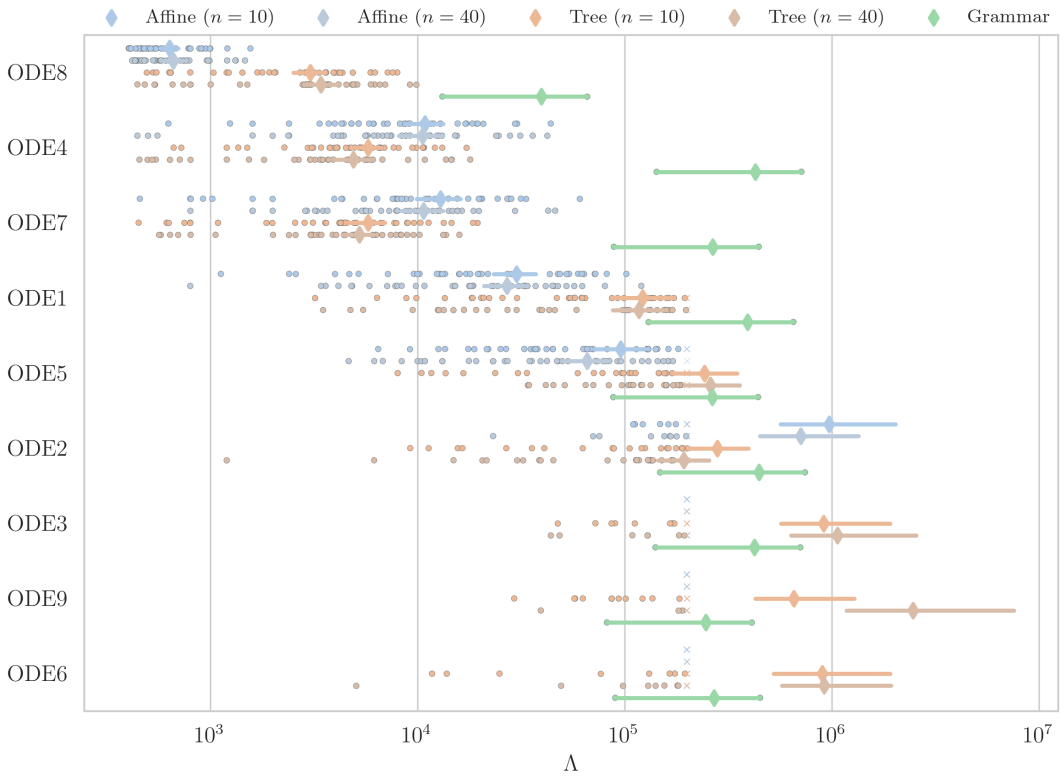


FIGURE 4.5: Comparison between expression and affine tree encodings.

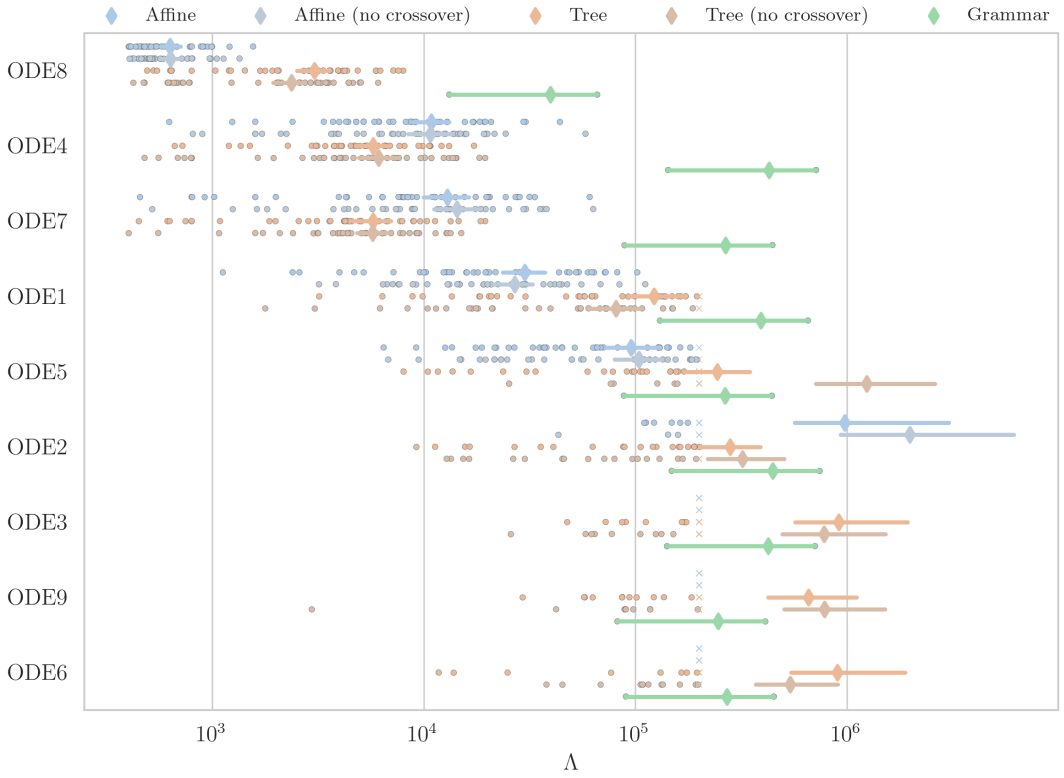


FIGURE 4.6: Investigation into the effect of tree crossover operations.

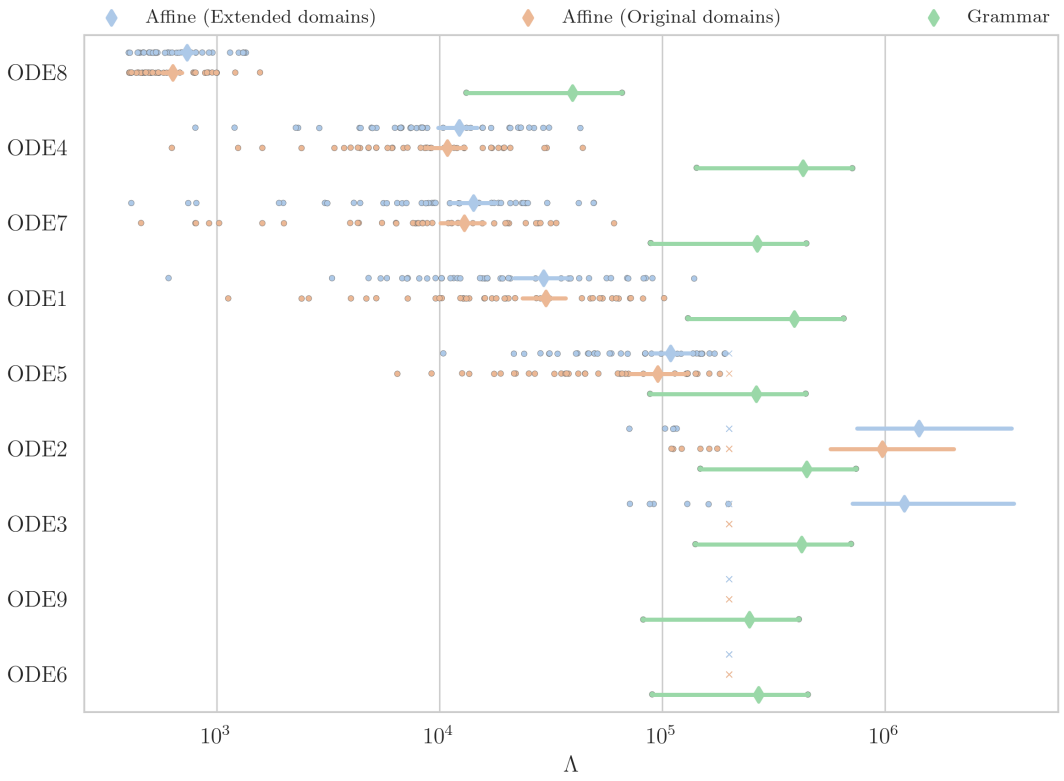


FIGURE 4.7: Comparing the effect of increased domains sizes for the ODEs in the TL benchmark set.

that the number of inexact solutions is minimised. To investigate this issue in the TL benchmark, additional optimisation runs are conducted for the affine-encoding scheme with extended domains. These are compared to the initial runs in Figure 4.7. Other than a few successful runs in ODE3, there is little discernible difference between the performances. This result indicates that the optimisation procedure is not hamstrung by the limited domain sizes.

In the investigations presented thus far, there has been little observed difference between the performances. A potentially-concerning explanation is that the dominant search mechanism is random sampling. To assess the extent to which this is the case, Figure 4.8 presents the results of the previous runs against additional data collected from a random-sampling approach (simply sampling a random expression tree with the 'grow' method). Alarmingly, the random search runs appear to perform similarly to the other approaches.

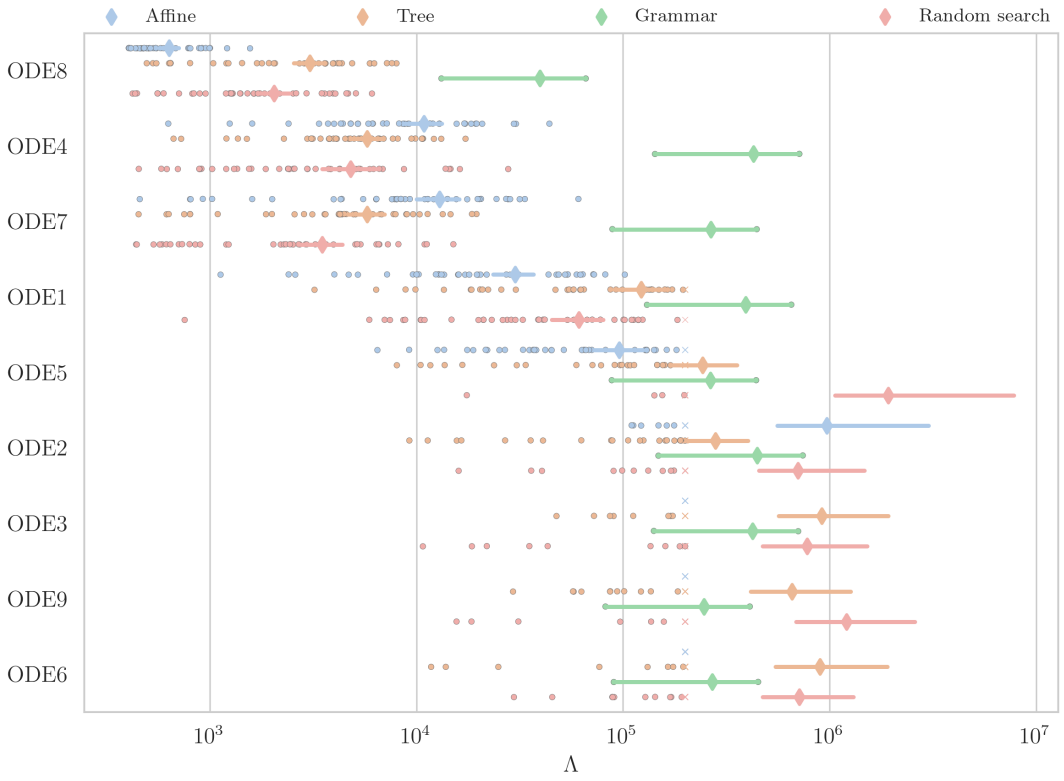


FIGURE 4.8: Performance comparison versus random search.

New benchmark

In Chapter 3, several limitations of the TL benchmark were identified, and a new benchmark was proposed for the assessment of optimisation-based solution of ODE problems. In this section, results for the affine-regression and tree expression encoding schemes are presented on the benchmark suite of problems proposed in Table 3.3. The first investigation, shown in Figure 4.9 compares performance on the encoding schemes. In the figure, the performance of the optimisation (measured in terms of Λ) degrades significantly with the number of nodes in the solution tree. An interesting exception to this trend is that ODE4 seems to have been more straightforward to solve than ODE3. One explanation for this is the presence of the comparatively-larger constant in ODE3, leading to less compact than optimal representations.

As seen in the TL benchmark ODE problems, there is little significant difference between the performances of the affine and tree-encoding schemes, aside from the more complex nodes, whereby the tree encoding appears to perform marginally better. This result is interpreted here as more evidence that the dominant search mechanism is random sampling of the encoding space.

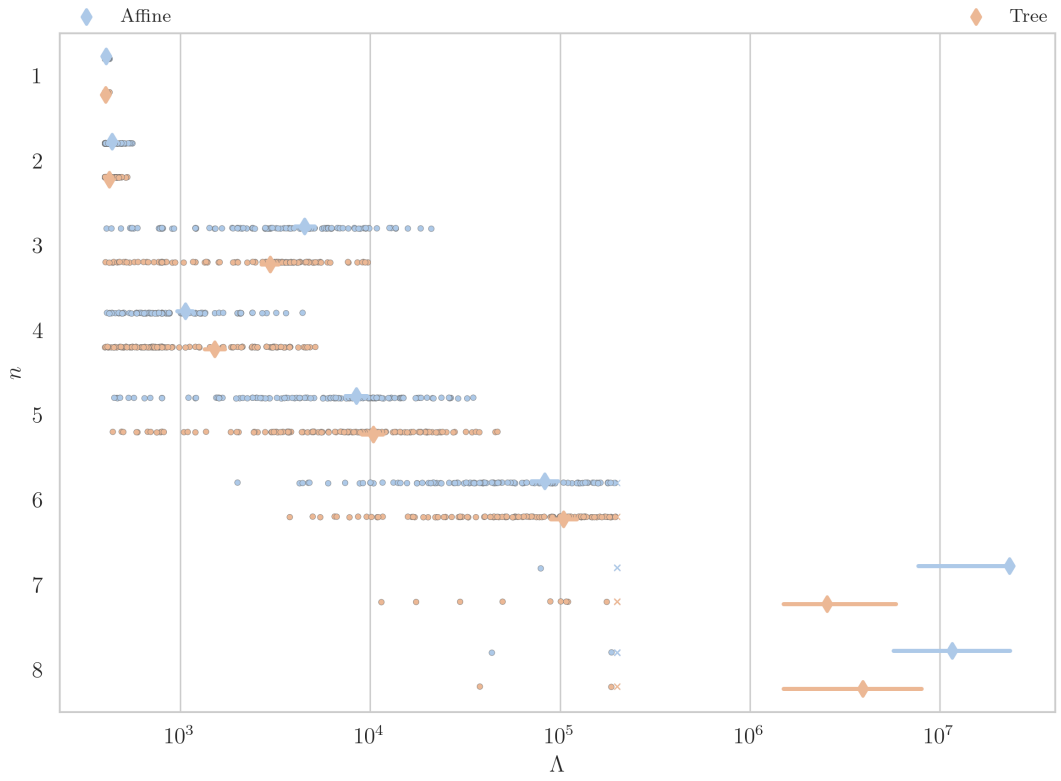


FIGURE 4.9: Performance comparison between affine and expression tree encodings.

An interesting ancillary investigation is presented here. Optimisation runs (using the affine-encoding scheme), were conducted for each of the homogeneous ODE problems in the new benchmark. Each of the homogenous ODE problems considered are then transformed into the inhomogeneous form by the procedure described in Chapter 3. Each of the resulting optimisation problems are presented in Figure 4.10. Interestingly, the different ODE problems do not appear to affect the performance of the optimisation run. The author sees two possible explanations for this result; the first is that the form of the homogeneous ODE does not affect the objective function in a meaningful way. This explanation is supported by the fact that terms in the objective function are measured in the function domain by an $\mathcal{L}^2$ metric that is insensitive to the algebraic form of the solution. Another explanation is that once again, random search is the dominant search modality and so the form of the ODE is irrelevant.

An aside on the claim - 'Dsolve/Mathematica/Maple was unable to solve this problem but the current approach can'

Several authors have made the claim stated above regarding one approach or another in reference to some differential equation of interest. The author feels that this claim

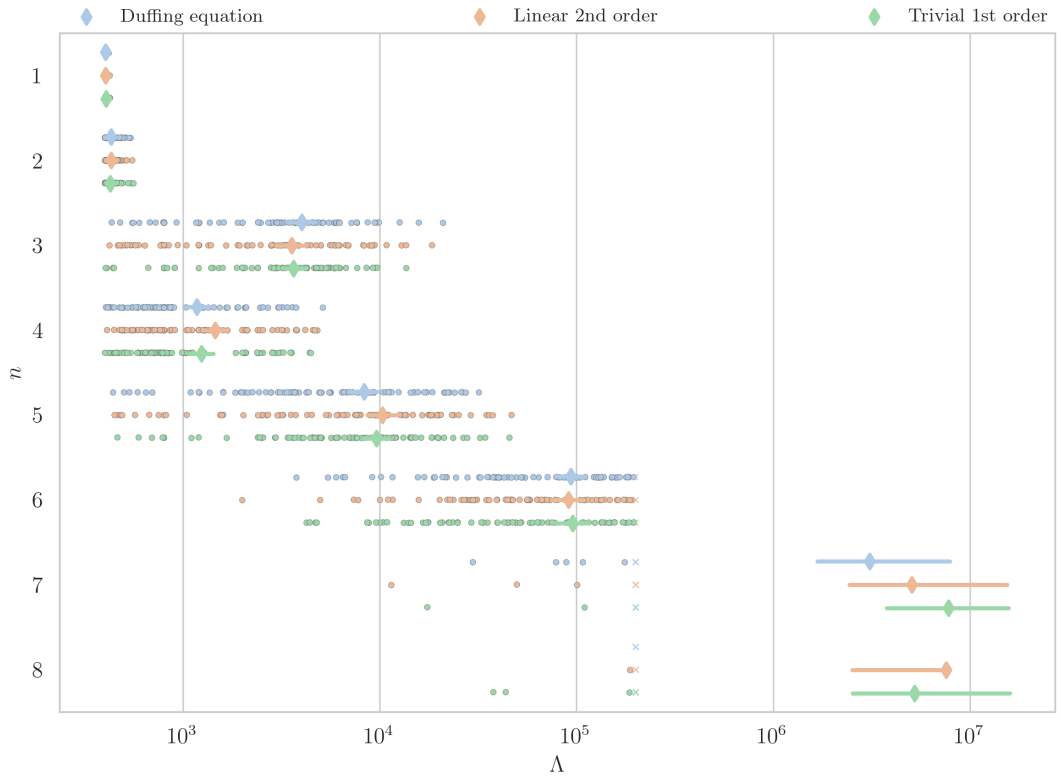


FIGURE 4.10: Comparison between solution size (nodes in tree) vs. ODE used in objective function.

(or rather the implied equivalence) is somewhat disingenuous. The difficulty of solving a differential equation in a purely analytical sense (the approach taken by computer algebra packages) is entirely related to the form of the differential equation. In the former case, the problem can largely be restated as 'can the differential equation of interest be transformed into one of several canonical forms for which the solution is prescribed?'. Conversely, in a heuristic approach, the difficulty is largely prescribed by the form of the solution. Figure 4.10 demonstrates this effect clearly. There is no perceptible difference between the difficulty (as measured by Λ) between different ODE problems which have the same functional form for the solution, despite wildly different inhomogeneous ODE problems.

Optimisation of the Affine-regression method: TL benchmark

The results presented thus far depict mixed performance of the affine method that is comparable to that of the tree encoding scheme. However, performance on the harder ODE problems (as measured by solution-tree size), appears to be significantly worse. At this stage, it is useful to examine the effects of some of the hyperparameters of the affine-regression tree approach. For brevity, the presented results are given on a subset

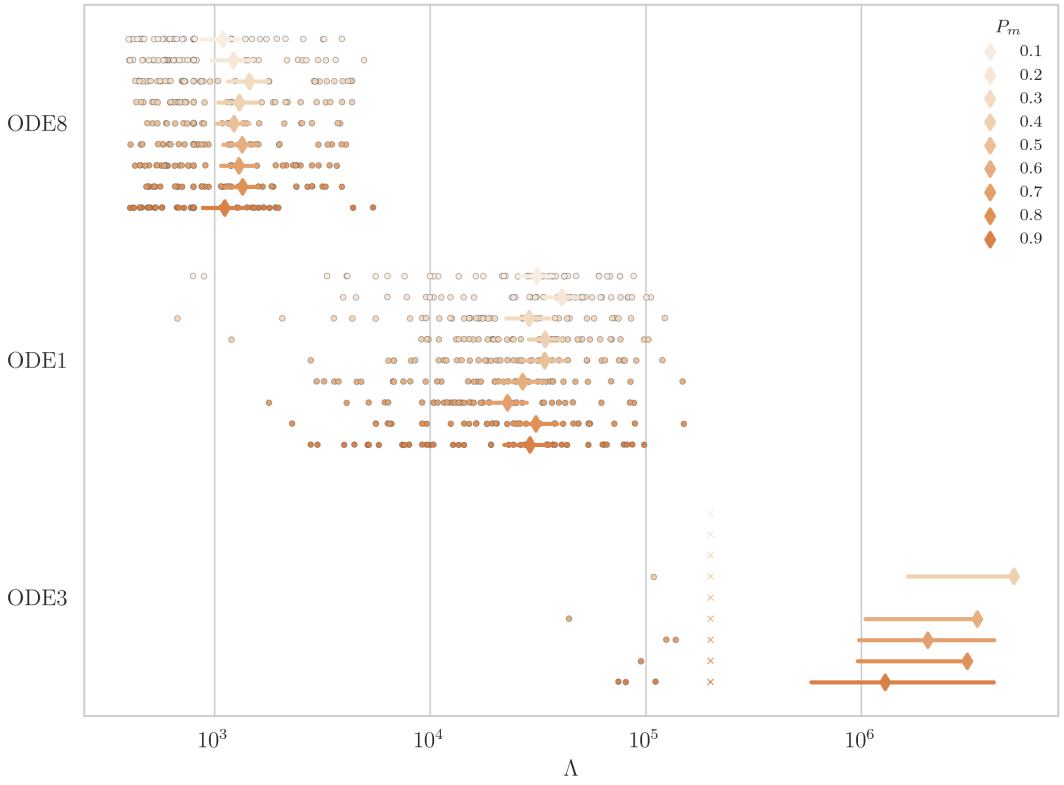


FIGURE 4.11: Investigating the effect of the mutation probability p_m.

of the TL problems, namely OD8, ODE1 and ODE3. These problems are selected to represent both the easiest and most difficult problems in the TL benchmark.

The first parameter considered here is the effect of the constant mutation probability P_m. The parameter was varied between values of 0.1 and 0.9 and the results of the optimisation runs are presented in Figure 4.11. In the figure, there is little effect from variation in the parameter on problems ODE8 and ODE1. However, there appears to be a slight preference for larger values of P_m on the more difficult ODE3 problem. One interesting possibility is that the values of P_m might be adapted as per some meta-heuristic hyperparameter optimisation scheme [140]. In this way the likelihood of applying the mutation operation might be varied during the run of the algorithm to suit different search regimes.

Figures 4.12 and 4.13, depict similar investigations on the values of the initial constant values a_0 and b_0 respectively. As might be expected, there is a clear advantage to selecting $a_0 = 1$ and $b_0 = 0$. This makes sense given the observation that the majority of ODE solutions seen 'in the wild' admit an affine representation with a sparse constant structure (i.e $a = 1$ and $b = 0$ for most nodes).

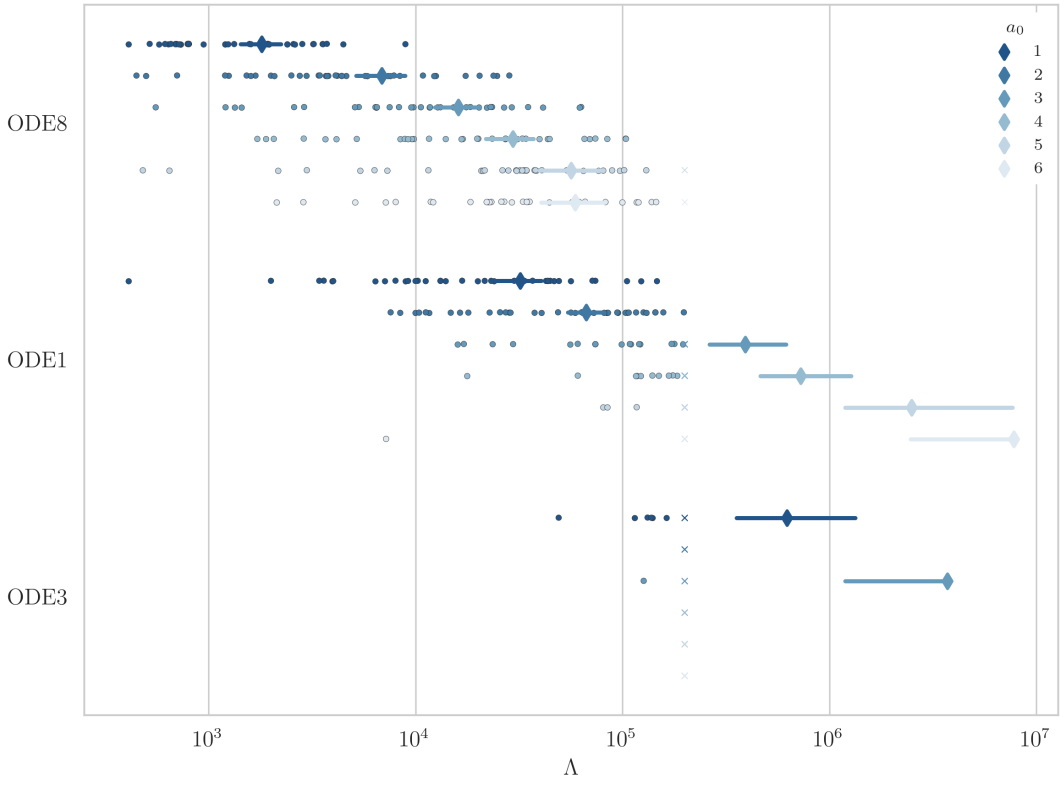


FIGURE 4.12: Investigating the effect of the initial linear constant value a_0.

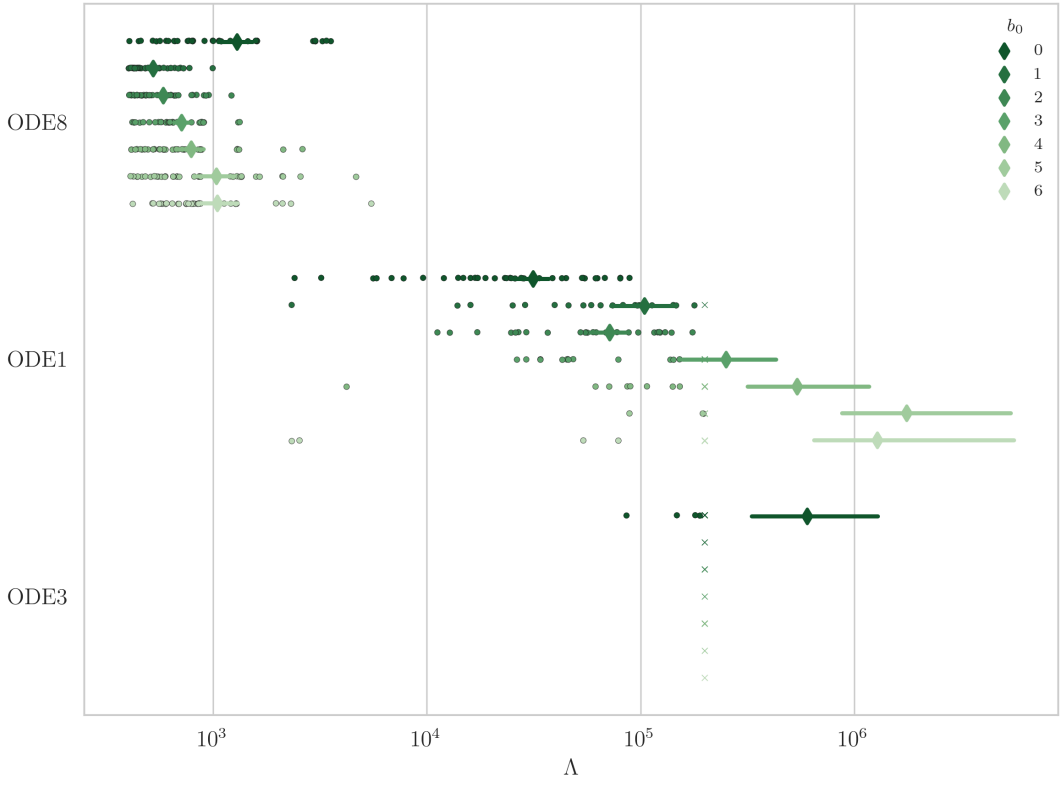


FIGURE 4.13: Investigating the effect of the initial constant parameter b_0.

The results presented thus far indicate that a dominant search mechanism (for tree structures at least) is random sampling. A potential reason for this is that it is very difficult to move locally within the encoding space without drastically altering the functional form (and therefore objective score), of an expression. Inspired by this limitation a novel constant specification scheme is introduced here.

Constant snap is a new mutation method that operates on the values of constants. The constant snap method proceeds as follows. First, a random constant (a or b) is selected from a random node in the tree. Next, a simulated annealing heuristic-optimisation routine [141] is used to specify an approximate value of the constant, treating the selected constant as an unknown and all other parameters in the tree as fixed. The resulting meta-optimisation task is a 1-dimensional nonlinear optimisation problem and the objective function is given by the $\mathcal{L}^2$ distance between the trial solution and the target values (obtained by numerically integrating the ODE forward in time).

Finally, an exact rational approximation (up to some numerical tolerance), of the constant value is taken by a numerical simplification algorithm. Here, the 'n-simplify' routine from Sympy [69] is used.

Figure 4.14 depicts the results of applying the constant snap mutation with probability P_{snap} to each node in the trial population. In the figure, there is clear evidence that the constant snap mutation improves search performance. On the harder ODE3 problem, the optimisation runs with $P_{\text{snap}} = 0.9$ give the best performance of any approach yet considered.

4.4 Conclusions

This chapter has presented a novel encoding scheme for expressions—the affine-regression tree. The proposed approach has a number of potential advantages over traditional tree-encoding schemes, including increased locality in the representation of constants, and a more compact search space. Also presented are a number of mutation operations for affine-regression trees that are designed to take small steps in the objective function, permitting a continuous optimisation surface. The search space size of the affine tree approach is computed and compared to those enumerated in the previous chapter. It is found that, while on the surface, the affine approach results in larger search spaces, much of this effect can be attributed to the specification of constants. For many real world expressions, the affine approach leads to considerably more compact expression representations.

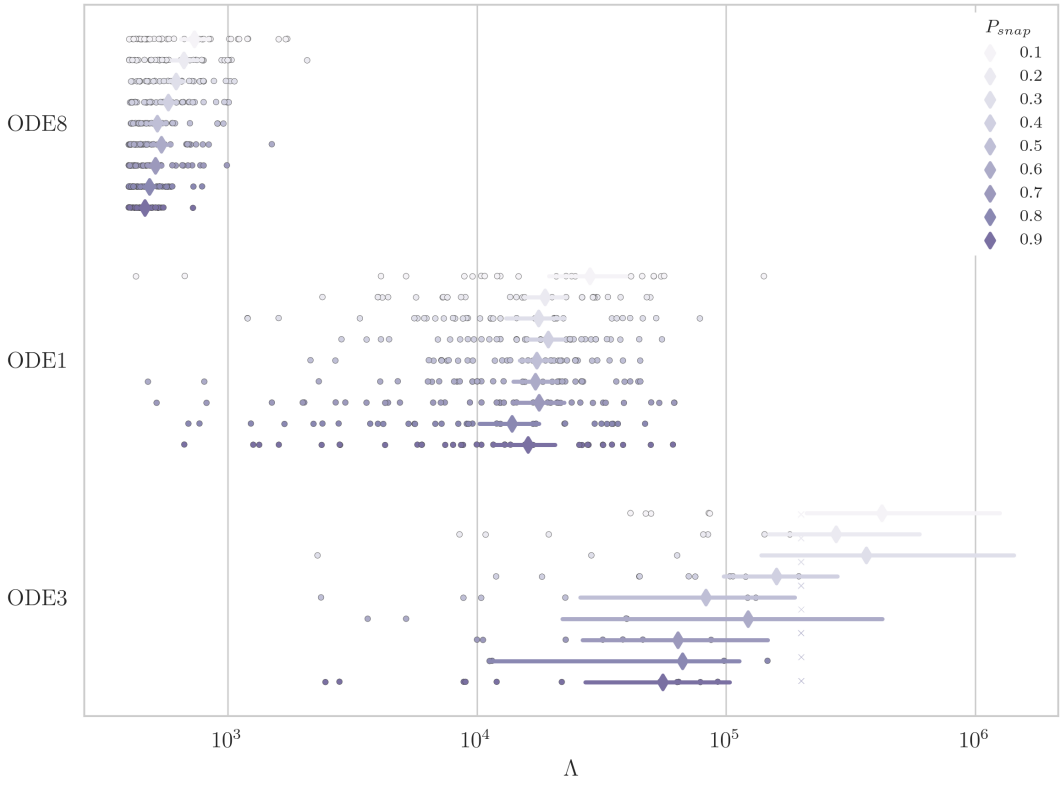


FIGURE 4.14: Investigating the effect of the probability of employing the snapping method p_{snap}.

The method of exact symbolic regression (using a GA) is demonstrated for affine and expression trees on two benchmark suites of ODE problems. On the first, from Tsoulos and Lagaris [83], mixed performances are seen. On ODE problems with the most compact solutions, the affine approach is seen to perform best. However, for ODE problems with the largest solutions, the affine tree encoding (as presented), failed to find any exact solutions. Interestingly, the data from the grammar-based approach of Tsoulos and Lagaris [83] does not seem to suffer from degradation with increasing search space dimension.

Perhaps most concerningly, none of the approaches considered in this chapter are able to significantly outperform random search (as seen in Figure 4.8). The obvious reasoning here is that there is still insufficient locality between the encoding space and the objective function for meaningful heuristic optimisation. It is the opinion of the author that further work in this direction must prioritise methods that promote this locality, if meaningful improvement over random sampling is to be found.

On the second benchmark, proposed in Chapter 3, the results are similar. There is little discernible difference between the affine-regression tree and expression tree encodings, and a strong degradation of performance with increasing numbers of nodes in the solution

tree. An interesting result obtained on this benchmark is that there does not appear to be a strong correlation between the homogenous ODE problem and the difficulty of the optimisation problem. Instead, (as has been seen empirically on the TL benchmark problems), the difficulty of the optimisation problem seems to be controlled entirely by the size of the solution.

A final set of results presented in this chapter consider the parameterisation of the affine-regression approach. Several hyperparameters of the affine-regression method are considered in the study. As might be expected, it is found that the initial values of the constant coefficients in the affine formulation $a_0 = 1$ and $b_0 = 0$ give the best performance. It is also shown that there is little sensitivity to the value of the mutation probability P_m.

Also considered here, is an exact meta-optimisation mutation—constant snap. The performance of this procedure is examined, and initial results are promising. When the operation is applied to new entrants to the trial population with probability $P_{\text{snap}} = 0.9$, the overall performance of the optimisation runs on ODE3 of the TL benchmark were better than any approach thus far considered. The author believes that constant snapping is a promising avenue for the exact specification of constants in symbolic regression, and several refinements to the method can be envisaged at this stage, including multiple heuristic regression of several constants at once, more powerful heuristic optimisation tools (such as differential evolution [142]) and the inclusion of a basis set of transcendental constants.

Overall, the performance of the affine-regression approach presented in this chapter has been varied. Although the method shows promise in several areas (including constant snapping and compactness of representation), the method has yet to provide a significant improvement over the standard expression-tree approaches or even random search. However, there yet remains fertile ground for investigation. The author considers several directions as promising areas for further work: One interesting observation of Seaton [85] is that the solutions to ODE problems tend to contain repeated sub-structures because of the application of the differential operator. No method so far considered in this chapter has this functionality. To approach this, Seaton and contributors have applied the method of *Cartesian genetic programming* (CGP) [85, 143] to the solution of ODEs by symbolic regression. It is interesting to imagine a hybrid approach that combines the structure detection of CGP with the constant specification of the affine tree.

4.4.1 From here to the moon

Where do the current results leave the search for exact solutions for nonlinear ODE problems? Although there has been little in the way of positive results, the author believes that some interesting conclusions can be reached. Certainly, analytical solution of as yet unsolved nonlinear ODE problems is a severe task, that has (by definition) evaded solution. Motivated by a desire to circumvent this analytical difficulty, it seems that symbolic regression may just exchange a difficult problem with one of comparable difficulty.

In Chapter 2, the following line of reasoning was established. For a given unsolved nonlinear ODE problem, either:

1. A closed form solution does not exist in terms of known transcendental functions and objects.
2. A closed form solution does exist, but heuristic methods are insufficiently powerful to locate it.
3. A closed form solution is available and can be found by heuristic means.

The author would argue that nothing presented thus far is sufficient to prefer any one of these statements in general. As has been shown several times in the results of this chapter, the difficulty in locating an ODE solution by heuristic means lies with the complexity of the solutions and not the ODE. What this means in practice, is that, should an exact solution to an as-yet unsolved ODE exist with a compact formulation (in terms of tree size), then there is a strong chance that it can be found by heuristic means.

Despite the negative results of this chapter, the author sees no reason to pull their hand out of the haystack.

Chapter 5

Statistically-independent NNMs for decoupling of nonlinear structural dynamics

The previous chapters have been concerned with the second of the grand challenges in nonlinear structural dynamics identified in this thesis; solutions to nonlinear ODEs. The remaining chapters will now address the third challenge, a nonlinear extension to modal analysis.

Although pervasively useful in the linear case, it has been shown that the effect of nonlinearity on modal analysis is very destructive. Retention of the practical properties of modal analysis in the nonlinear case is an open area of research, that has attracted attention for many decades after the original works of Rosenberg [18]. Since Rosenberg, many quantities have been introduced in the literature that have been described as a NNM of vibration. Among these, the recently-proposed statistically independent framework [22] shows particular promise. The approach differs from many other NNM frameworks, by placing its focus on constructing a decomposition that is of practical utility to the engineer. In Chapter 2, a number of criteria were established for a practical nonlinear extension to linear modal analysis:

1. *Independence*: The ability of the decomposition to render the dynamics into an independent modal basis, preferably SISO.

2. *Decomposition*: The extent to which the decomposed modal dynamics represent a physically meaningful (by some measure) basis for understanding the structural dynamics.

3. *Superposition*: The extent to which the original dynamics can be recovered from the decomposition.

The objective of this chapter will be to introduce the statistically independent framework and consider how the criteria of Chapter 2 might be employed both as metrics of utility and also as *inductive biases* that can be incorporated into the machine learning framework to induce a useful decomposition by construction.

5.1 The statistically-independent framework

In 2017, Worden and Green [22], proposed a new class of NNM, based on a nonlinear decomposition into uncorrelated time series. A motivation for these NNMs was that often during analysis, the engineer does not have access to the underlying EOM of their system, and so methods that rely on harmonic balance or shooting approaches cannot be applied.

The mathematical structure of the framework is very simple. 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 are uncorrelated time series,

$$f(\mathbf{y}) = \mathbf{u} \tag{5.1}$$

Figure 5.1 depicts the statistically-independent 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,

$$f^{-1}(\mathbf{u}) = \mathbf{y} \tag{5.2}$$

At first sight, this approach may seem to bear close resemblance to the NNM framework introduced by Rosenberg. However, it is structurally distinct. In the statistically-independent framework, each of the physical displacements are used to build a map onto each of the modal displacements 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. Instead, the modal ansatz is,

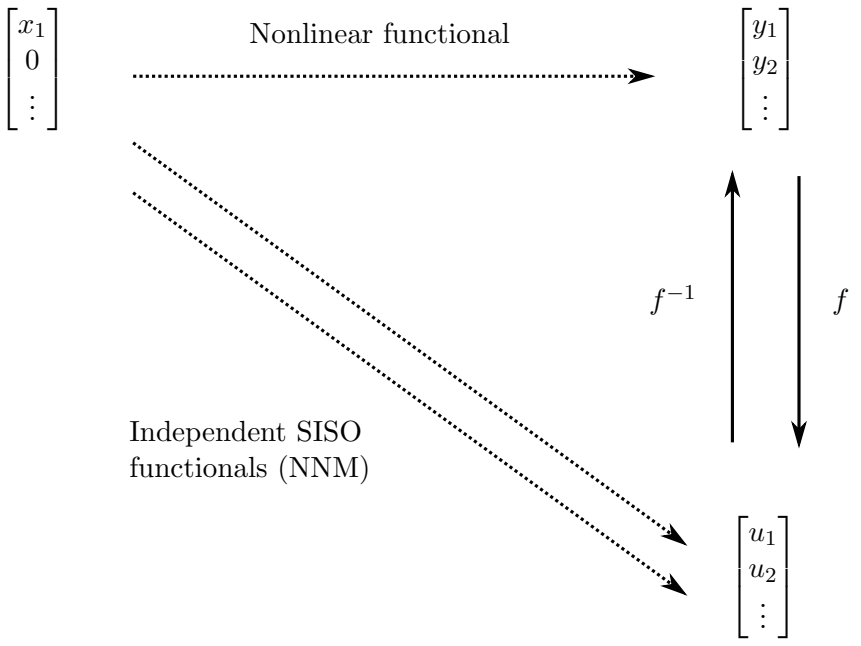


FIGURE 5.1: Overview of the novel NNM approach.

$$u_i = f_i(y_1, y_2, \ldots, y_n) \quad \forall i = 1, \ldots, n \tag{5.3}$$

Several contributions have extended and implemented the framework. These include nonparametric learners in [126], consideration of nonlinear correlation metrics in [127] and a neural network formulation in [23].

A motivation for the investigation of the statistically-independent framework is that it can be readily applied in a similar approach to a linear modal testing campaign. Faced with the task of nonlinear system identification on an unknown structurally-nonlinear system, the framework of statistically-independent NNMs remains an attractive alternative to a purely black-box SIMO identification scheme. A practical workflow might be:

1. Capture some displacement data from some SIMO structure of interest, subject to broadband excitation.
2. Presume or establish the presence of nonlinearity (for example by assessment of the coherence).
3. Conduct statistically-independent modal analysis on the data and retrieve the modal dynamics.

4. Finally perform NLSI on the modal dynamics as SISO functionals.

What is possible with statistical independence?

At first sight, the statistically-independent framework may not appear sufficiently powerful to produce useful decompositions for all of innumerable ways that a system can be nonlinear. Indeed it is not claimed here that the framework can be applied in all settings. At this stage, no formal proof can be offered that limits the approach to a certain class of systems and excitation types. However it will be useful at this stage to hold some discussion over when the approach might be applied in practice.

In order for the outputs of the nonlinear dynamic system to be projected into statistically-independent coordinates, it must be required that the outputs are in some sense random signals. It therefore follows that excitation signals should be sufficiently broadband such that there is little to no contribution from coherent nonlinear effects such as stochastic resonance, bifurcation and chaos. In terms of the classes of nonlinearities that the method can be applied to, it is hard to imagine any limit on the functional form that can be considered.

At its core, the statistically-independent framework relies on the specification of a number of cognitive biases that encourage a practical decomposition into modal coordinates. It can therefore be imagined that the types of nonlinearity and excitation for which the approach is valid will depend entirely on the choices of these biases.

In Chapter 2, it was argued that the criteria for a practical nonlinear modal decomposition could be expressed as the following,

1. *Independence*: The ability of the decomposition to render the dynamics into an independent modal basis, preferably SISO.
2. *Decomposition*: The extent to which the decomposed modal dynamics represent a physically meaningful (by some measure), basis for understanding the structural dynamics.
3. *Superposition*: The extent to which the original dynamics can be recovered from the decomposition.

The remaining sections of this chapter present an investigation into practical metrics that can be used to evaluate these criteria within the context of the statistically-independent framework. This is done with a view to identifying metrics that can be used as *objective functions* within a machine-learning approach.

5.2 On measures of independence

The first criteria introduced in Chapter 2 is *independence*. In much of the work that has so far been completed surrounding the statistically-independent framework, this has been evaluated by a simple linear correlation metric [22]. However, as identified in [127], this is a restricted view that does not account for the presence of nonlinear correlations.

Correlation is not the only measure of independence as defined above. As an additional *inductive bias*, the original statistically-independent framework [22], sought to retain the idea of orthogonality that is present in linear modal analysis (the mode shapes are a weighted orthogonal basis). In [22], orthogonality was introduced by evaluation of the pairwise inner products of the columns of the parameter matrices in the multinomial expansion of the forward transformation.

In [23], the idea of orthogonality was extended by requiring that the forward and inverse transformations be *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 $\mathbf{x_i}, \mathbf{x_j} \in X$, f is conformal if,

$$\frac{\mathbf{x}_i \cdot \mathbf{x}_j}{\|\mathbf{x}_i\|\|\mathbf{x}_j\|} = \frac{f(\mathbf{x}_i) \cdot f(\mathbf{x}_j)}{\|f(\mathbf{x}_i)\|\|f(\mathbf{x}_j)\|} \quad \forall \mathbf{x}_i, \mathbf{x}_j \in X \tag{5.4}$$

where $\cdot$ is the usual scalar product. Conformality is included as a term in the objective function of [23], by the application of an orthogonality-assembly (essentially a perturbation based test of conformality in the forward transformation) with a neural-network structure. For additional detail on the implementation of the orthogonality-assembly, the interested reader is directed to the original publication [23].

5.2.1 Metrics for analysis of correlation

An essential ingredient of this data-driven approach to nonlinear modal analysis is the selection of a correlation metric. The notion of correlation between two time series can be understood intuitively in a number of ways. For example, correlation can be understood to be the extent to which one signal is predictable from the other (and vice versa), or the amount of information (by some measure) that one signal contains about another.

A common choice for the evaluation of correlation is the *Pearson product moment correlation coefficient* (often simply referred to as the correlation coefficient). In fact, many

such objects exist, each with a different subset of desirable properties. A number of such metrics are compared here.

In the notation that follows, α and β are jointly-distributed signals, each of length n with finite second moment.

Pearson's product moment correlation

The first correlation metric considered here is Pearson's product moment correlation coefficient (Cor); it is a linear metric that is described for signals α and β by,

$$\rho_p = \frac{K(\alpha, \beta)}{\sigma_\alpha \sigma_\beta} \tag{5.5}$$

where the covariance function K is given by,

$$K(\alpha, \beta) = \mathrm{E}\left[(\alpha - \mathrm{E}[\alpha])(\beta - \mathrm{E}[\beta])^{\mathsf{T}}\right] \tag{5.6}$$

where E represents the expectation operator. However, for sample populations, the sample Pearson correlation coefficient can be used,

$$\rho_p(\alpha, \beta) \approx \frac{\sum_{i=1}^{n}(\alpha_i - \bar{\alpha})(\beta_i - \bar{\beta})}{\sum_{i=1}^{n}(\alpha_i - \bar{\alpha})^2 \sum_{i=1}^{n}(\beta_i - \bar{\beta})^2} \tag{5.7}$$

This representation is convenient in that the mean residuals can be efficiently computed by vectorisation, allowing for a fast implementation. Additionally, because of the scaling on the denominator, ρ_p is bounded on the interval $[-1, 1]$, with a value of one indicating two perfectly linearly correlated signals and a value of negative one indicating perfect anti-correlation. A zero value indicates statistical independence up to second order.

In order for this metric to be useful for assessing modal correlation, the absolute value of the coefficient is taken, so that it is mapped into the range $[0, 1]$. Because the objective of the optimisation is to achieve statistical independence, it will be inconsequential that correlations and anti-correlations are not distinguished. Although there are several advantages in terms of ease and speed of implementation of this metric, it is limited in that is is only able to encode linear correlations.

Spearman's rank monotonicity test

Spearman's rank monotonicity test (ρ_s) is another correlation metric similar to that of Cor, except that the correlation test is preformed on the indices of the sorted data (i.e ranks) rather than on the values of the data directly. As such, the metric does not test for correlation in the signal, but monotonicity.

Spearman's rank monotonicity test is evaluated for two signals by first replacing each signal point α_i with it's rank r_{α_i}. This can be achieved with any efficient sorting algorithm. The Spearman coefficient can then be evaluated by taking the correlation of these ranks as in,

$$\rho_s(r_\alpha, r_\beta) = \frac{\rho_p(r_\alpha, r_\beta)}{\sigma_{r_\alpha}\sigma_{r_\beta}} \tag{5.8}$$

In the absence of repeated ranks (no 'ties' in the signal values) the Spearman coefficient can be efficiently computed by the approximation,

$$\rho_s(r_\alpha, r_\beta) = 1 - \frac{6\sum_i D_i^2}{n(n^2-1)} \tag{5.9}$$

where n is the length of each signal and D is the difference in rank between each observation in the signal,

$$D = r_\alpha - r_\beta \tag{5.10}$$

As with the Pearson coefficient, the Spearman metric returns values on the interval $[-1, 1]$. As such, the absolute value is taken to ensure that independence represents a minimal value.

Computationally, the Spearman rank coefficient is at least as expensive as the Pearson correlation coefficient, as the expression is identical, apart from the calculation of the ranks. However the Spearman rank coefficient is not limited to linear correlations as it is a test for monotonicity. Significant savings are available in the absence of repeated ranks, as the above approximation is computable in linear time; however, the process of validating the lack of repeated ranks can add to the complexity.

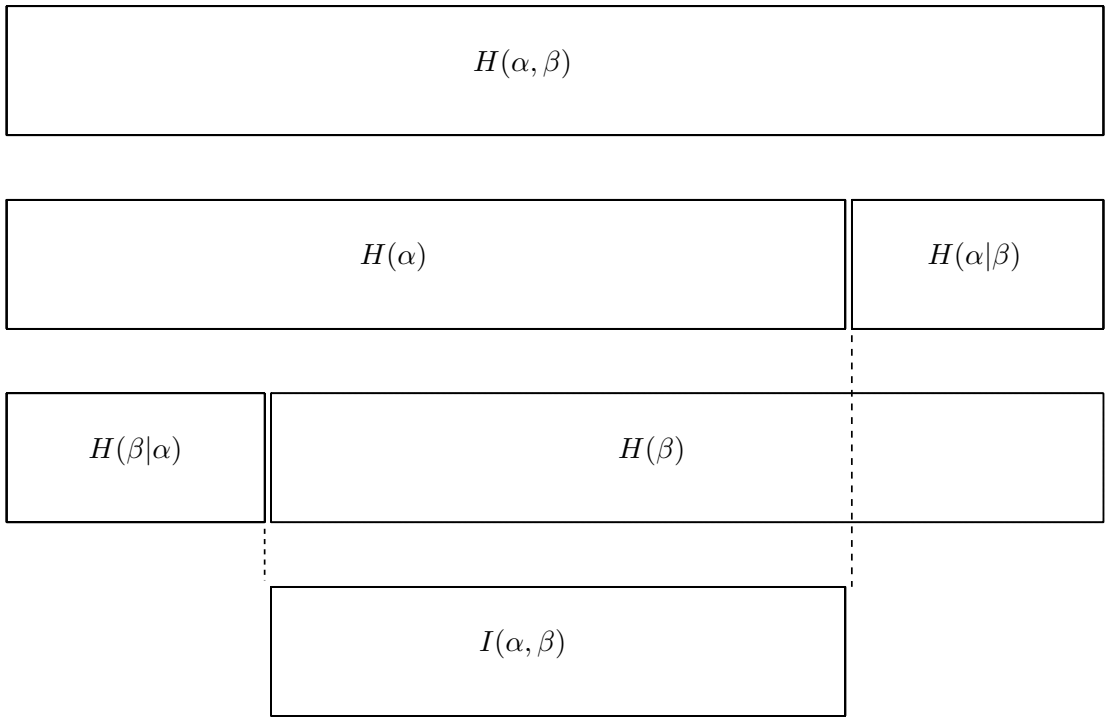


FIGURE 5.2: Graphical representation of mutual information and its relationship with other information theoretic quantities.

Mutual information

The mutual information is another measure of dependence based on the ideas of information and entropy. Put simply, it is the amount of information (by some measure) that can be inferred about a signal by observing another. It is closely related to the idea of Shannon entropy [144]. Figure 5.2 depicts a graphical representation of the mutual information between two signals. The figure is based on the representation provided by Mackay in [145].

The mutual information can be calculated form the joint and marginal entropies of the signals α and β, given by,

$$H(\alpha, \beta) = -\sum_{i}^{d}\sum_{j}^{d} p(\alpha_i, \beta_j) \log(p(\alpha_i, \beta_j)) \tag{5.11}$$

$$H(\alpha) = -\sum_{i}^{d} p(\alpha_i) \log(p(\alpha_j)) \tag{5.12}$$

The mutual information can then be calculated as,

$$I(\alpha;\beta) = H(\alpha) + H(\beta) - H(\alpha,\beta) \tag{5.13}$$

As can be inferred from Figure 5.2, the standard metric of mutual information is bounded on the interval $[0, H(\alpha,\beta)]$. As such, it is necessary to divide through by the joint entropy $H(\alpha,\beta)$ in order to recover a metric that is bounded on the interval $[0,1]$.

$$I_{\text{metric}}(\alpha;\beta) = \frac{I(\alpha;\beta)}{H(\alpha,\beta)} \tag{5.14}$$

The computation of the mutual information is theoretically inexpensive, but in practice one does not usually have access to the probability distributions of α and β and so these must be expensively estimated using a binning method. As such, the mutual information is the most expensive of the metrics thus far considered.

Distance correlation

The distance correlation (dCor) can be thought of as a generalisation of the Pearson product moment correlation coefficient. Whereas the Pearson coefficient is only able to detect linear correlations between signals of scalar variables, the distance correlation is able to detect both linear and nonlinear correlations between pairs of random vector-valued signals, that need not necessarily have the same dimension.

The distance correlation is defined for pairs of vector-valued signals **a** and **b** by first calculating the distance matrices A and B, specified by,

$$a_{kj} = ||\{\alpha_j\} - \{\alpha_k\}||, \qquad k,j = 1,2,..,l \tag{5.15}$$

$$b_{kj} = ||\{\beta_j\} - \{\beta_k\}||, \qquad k,j = 1,2,..,l \tag{5.16}$$

where $||...||$ is taken to be Euclidean norm given by,

$$||X|| = \sqrt{\sum_i x_i^2} \tag{5.17}$$

In fact, any distance metric is appropriate. One is free to use instead the Mahalanobis or rectilinear norms. In practice, the specification of the distance measure will depend on the application and the dimensionality of the data.

Next, the doubly-centred distance matrices A' and B' are computed,

$$A'_{kj} = a_{kj} - \bar{a}_j \bar{a}_k + \bar{A}, \qquad k, j = 1, 2, .., d \tag{5.18}$$

$$B'_{kj} = b_{kj} - \bar{b}_j \bar{b}_k + \bar{B}, \qquad k, j = 1, 2, .., d \tag{5.19}$$

In these equations, the bar notation indicates the mean value with the subscript indicating whether it is over a row or column of the distance matrices A and B. the parameter $\bar{A}$ represents the grand mean (mean of all elements) of the distance matrix A. The distance covariance can now be recovered from,

$$\mathrm{dCov}(\alpha, \beta) = \sqrt{\frac{1}{n^2} \sum_j \sum_k A_{kj} B_{kj}} \tag{5.20}$$

By considering the distance covariance of a signal with itself, the distance variance (dVar) can be similarly calculated from,

$$\mathrm{dVar}(\alpha) = \sqrt{\frac{1}{n^2} \sum_j A_{kj}^2} \tag{5.21}$$

Finally the distance correlation can be evaluated from equation (5.15),

$$\mathrm{dCor}(\alpha, \beta) = \frac{\mathrm{dCov}(\mathrm{M}, \mathrm{N})}{\sqrt{\mathrm{dVar}(\mathrm{M})\, \mathrm{dVar}(\mathrm{N})}} \tag{5.22}$$

Unlike other correlation metrics considered for this investigation, the distance correlation is already bounded on the interval $[0, 1]$ and so no re-scaling is required. In terms of computational cost, dCor is by far the most expensive of the metrics considered. This is because of the expense involved in the computation of the distance matrices which quickly grows in cost with the length of the signals. However, this increase in computational cost is offset by the ability of the distance correlation to detect nonlinear correlations in the data.

5.2.2 Comparison on data with known correlations

The statistically-independent framework relies at its core on the inductive bias of uncorrelated modal displacements. Selection of an appropriate metric of correlation is therefore an integral component in the framework. Given the correlation metrics identified above, it is useful to experimentally validate their performance on a number of signals with known correlations. The correlations considered here are, independence (no

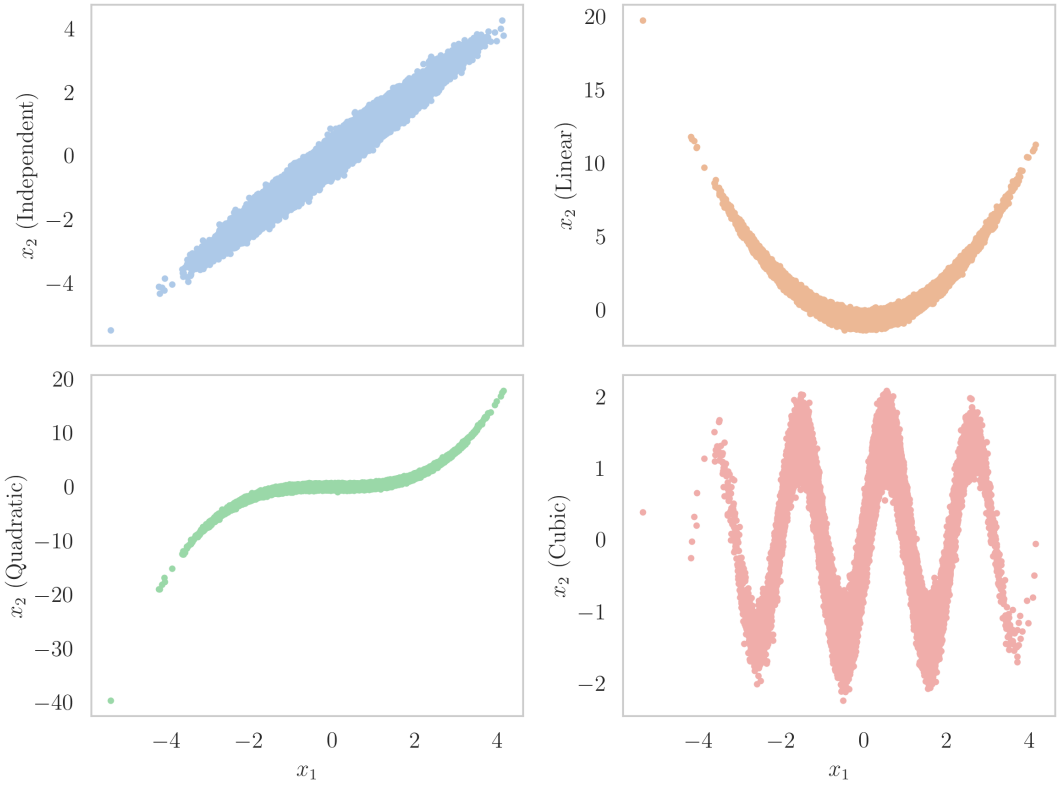


FIGURE 5.3: Correlated data (x_1 vs x_2) considered in this investigation at a measurement noise level of 20%.

correlation), linear correlation and nonlinear correlations including quadratic, cubic and a sinusoidal relationship. For each test correlation signal, 10^5 points are sampled from a standard normal distribution. The correlating signals are then generated and normalised to zero-mean, unit standard deviation. Finally, some Gaussian measurement noise is applied (except in the independent case) with unit variance and relative magnitudes in the range $[0, 0.9]$ with an increment of 0.1. Overall, 41 test signals were generated. Figure 5.3 depicts test signals at a measurement noise level of 20%.

With the test correlation data established, the correlation metrics considered above can be compared. Each of the dependence measures described earlier are applied to the each of the 41 test cases. The results of the investigation, averaged over all noise levels are shown in Figure 5.4. As can be seen from the figure, all measures were able to correctly give a low score on the independent set. Performance on the linear correlation is mostly good with all metrics except mutual information correctly attributing a high level of correlation. In the nonlinear correlation tests, the performance is more mixed. As is to be expected, the linear correlation metrics (ρ_p and ρ_s) perform well on the odd nonlinearity but less well on the even functions (quadratic and sinusoidal). Overall, the periodic nonlinear correlation proves the most difficult to detect, with the distance correlation performing the best overall.

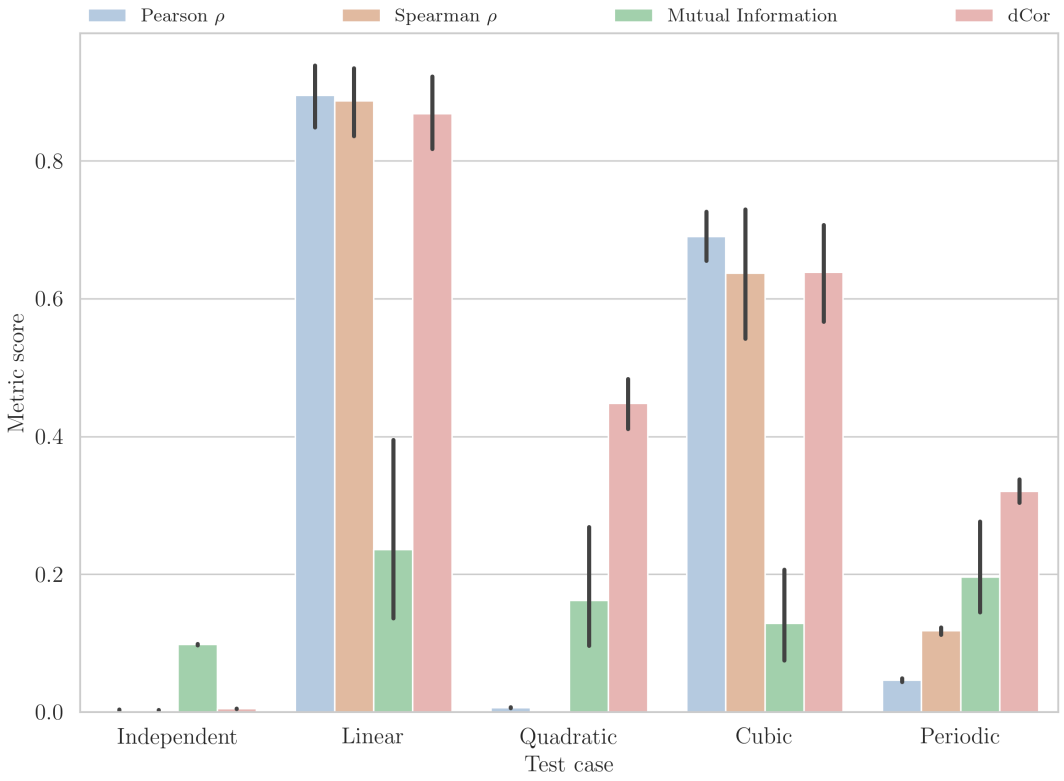


FIGURE 5.4: Performance of the measures of statistical independence at every noise level, error bars depict 90% confidence intervals estimated by bootstrapping.

The results in Figure 5.4 give the impression that the mutual information metric performs strictly worse than the other metrics considered in all cases; however, this is an artefact of the sensitivity to the level of measurement noise. Figure 5.5 depicts the degradation of the mutual information metric with increasing measurement noise. In the absence of any corruption, the mutual information metric performs extremely well and attributes high levels of correlation between datasets. However, the performance can be seen to degrade quickly as the noise increases.

Measurement noise is an inevitable part of engineering dynamics analysis. A measure of statistical independence that depends so strongly on the level of measurement noise is unlikely to be useful in detecting correlations in real-world datasets, and so the mutual information metric is not considered further.

Overall, the most promising metrics for the analysis of correlations in the context of the statistically-independent NNM frameworks are the Pearson product moment correlation coefficient (ρ_p) and the distance correlation (dCor). Of these, dCor is by far the better at detecting nonlinear correlations in data. However, this comes at the price of an increased computational cost. In contrast, ρ_p is limited in its ability to only detect linear correlations (although odd-type nonlinearities are well approximated). However,

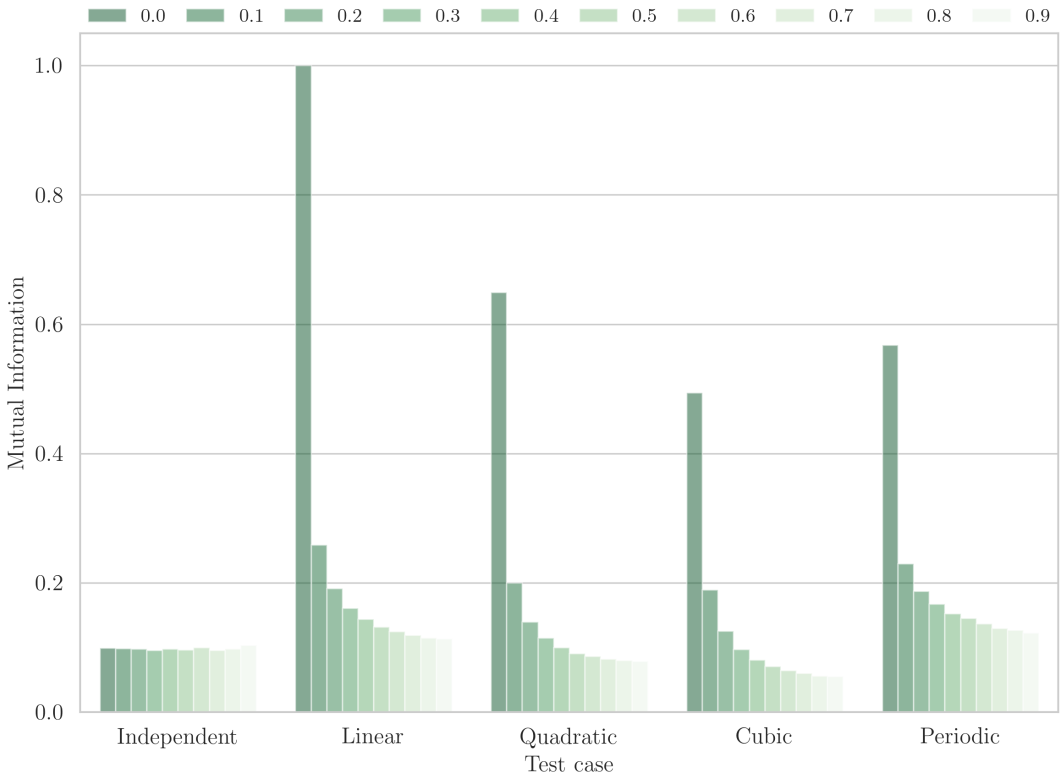


FIGURE 5.5: Degradation of the mutual information metric with increasing measurement noise.

the computational cost of the linear metric is much lower than the nonlinear one[1] and this is an important consideration in a machine learning context, where the metric may be required to be evaluated many thousands of times.

5.3 On measures of modal decomposition

Of perhaps equal importance to independence is the extent to which the decomposed signals represent a practical decomposition that facilitates a simplified analysis. In [22], it is argued that an intuitive requirement of a useful nonlinear modal decomposition is one that results in distinct resonance peaks in the power spectral densities (PSDs) of the modal dynamics. This idea of *unimodality* takes motivation from the observation that the modal dynamics in the linear case are necessarily single peaks. In the paper, the unimodality of the NNM PSDs was judged visually 'by eye'. While this approach is useful for a qualitative argument, it is useful to imagine methods that might include

[1] Pearson's product moment coefficient can be computed in $\mathcal{O}(n)$ time whereas the distance correlation requires $\mathcal{O}(n^2)$ for the formulation given here. An alternate faster $\mathcal{O}(n \log n)$ algorithm is given in [146] which is the implementation used in the numerical study.

modal unimodality as an additional inductive bias in the framework. To this end, some additional measures of modal decomposition are included here.

Convolution unimodality test

One issue with the 'by-eye' approach used in [22], is the lack of repeatability of the method. To address this, a measure of modal separation is presented based on the convolution operation. The proposed measure is motivated by the unimodality-preserving properties of the convolution operation. it is therefore reasoned that if the PSDs of the modal coordinates are convolved, the resultant signal can only be unimodal if each of the modal directions was itself unimodal. The unimodality of the convolved signal can then be assessed by a peak-picking algorithm or another method. Another advantage of the convolution-unimodality test is that a highly-efficient implementation is available. Leveraging the fact that the convolution operation in the frequency domain is equivalent to a multiplication in the time domain, one is able to compute the convolved signal as,

$$U_{\text{convolution}} = U_1 * U_2 * \cdots * U_n = \mathcal{F}\left[\prod_i^n u_i\right] \tag{5.23}$$

where $\mathcal{F}$ is the Fourier transform operator. The proposed approach is depicted in Figure 5.6 where some test unimodal and bimodal signals, corrupted by noise, are convolved. As can be seen, only the unimodal signals produce a unimodal convolution.

The convolution measure is limited in that it does not provide a strict measure of decomposition, and is therefore still limited to the qualitative type of analysis considered in [22].

Spearman rank unimodality test

Given the weaknesses of the method proposed above, a strict metric definition of unimodality is proposed here, motivated by the Spearman rank correlation coefficient described above. For a true unimodal signal $f(x)$, the following conditions must hold,

$$f(x_1) \leq f(x_2), \quad x_1 < x_2, \quad x_1, x_2 < \operatorname{argmax} f(x) \tag{5.24}$$

$$f(x_1) \geq f(x_2), \quad x_1 < x_2, \quad x_1, x_2 > \operatorname{argmax} f(x) \tag{5.25}$$

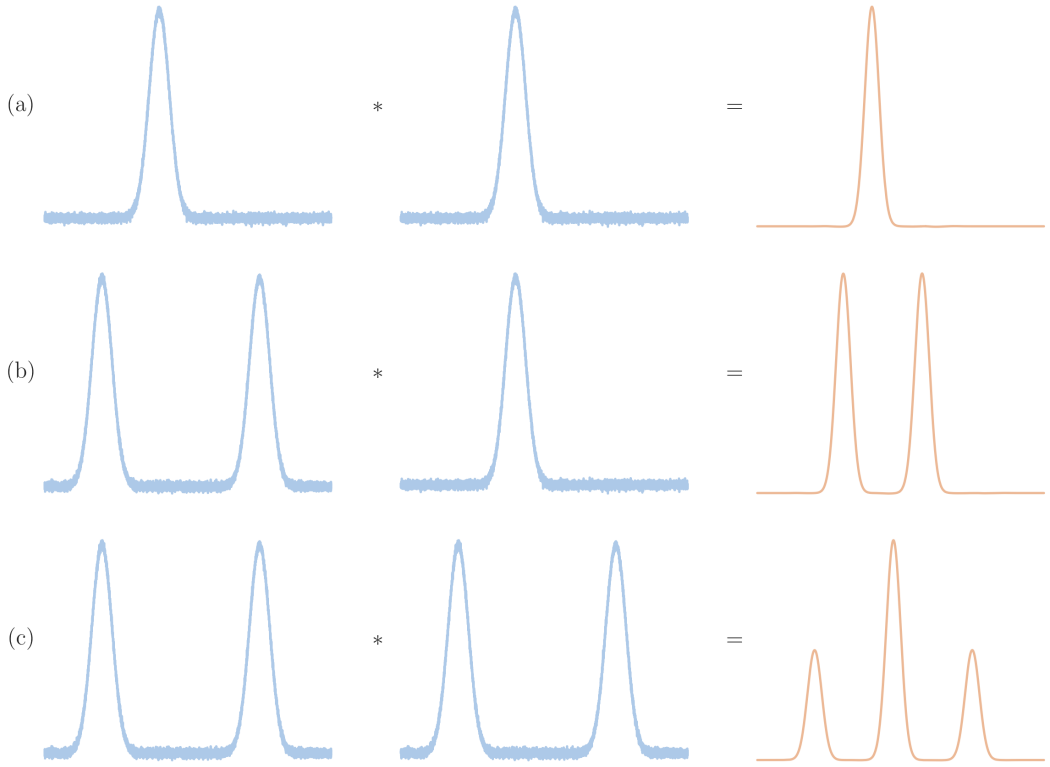


FIGURE 5.6: Result of convolving toy PSD signals; (a) Unimodal with unimodal, (b) Unimodal and bimodal, (c) Bimodal and bimodal.

In other words, if one were to divide f into two signals by splitting the series about the maximum point of f, the resulting signals must be monotonically increasing (left of the peak) and monotonically decreasing (right of the peak). For some trial signal $g(x)$, the unimodality can be measured by evaluating the normalised-product of a monotonicity test on each of the two signals. For the left signal,

$$m_{\text{lhs}} = \rho_p(x^-, g(x^-)), \quad x < \operatorname{argmax} g(x) \tag{5.26}$$

and the the right signal,

$$m_{\text{rhs}} = -\rho_p(x^+, g(x^+)), \quad x > \operatorname{argmax} g(x) \tag{5.27}$$

The overall metric can then be calculated as,

$$m_{\text{unimodality}}(g) = \sqrt{m_{\text{lhs}} m_{\text{rhs}}} \tag{5.28}$$

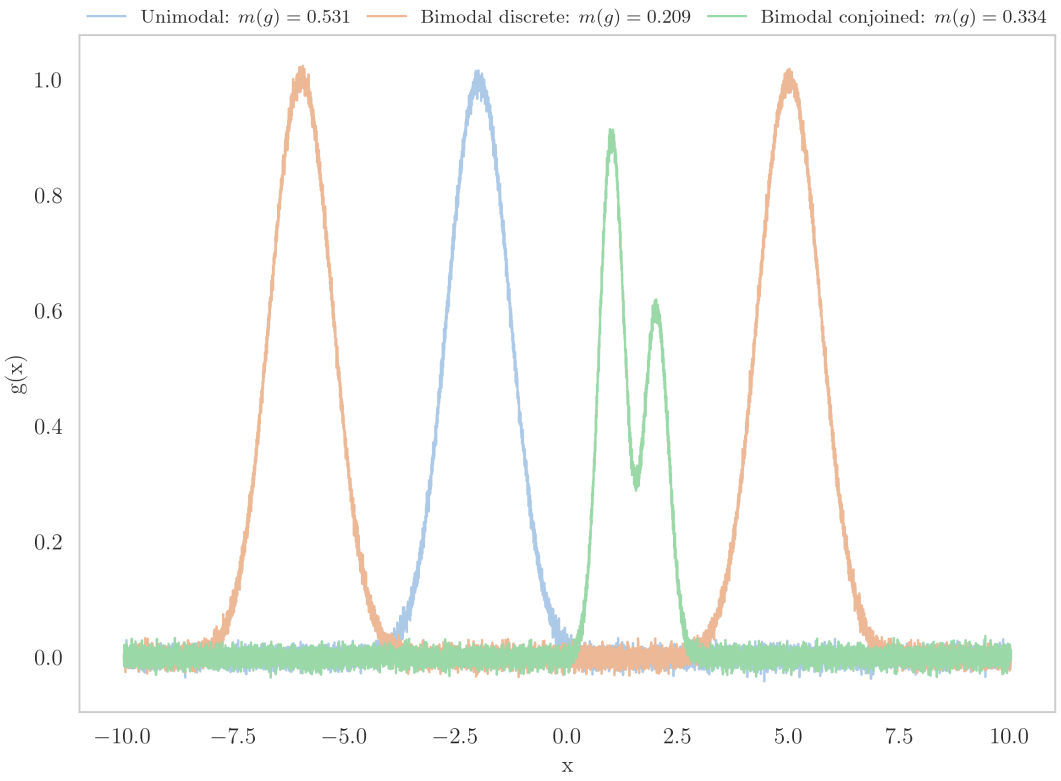


FIGURE 5.7: Results of the proposed unimodality metric based on Spearman rank monotonicity test on toy PSD data

The above metric is bounded on the interval $[0, 1]$ for signals with a single maxima. only a truly unimodal signal can produce a unity score. The proposed metric is demonstrated on a small number of test cases in Figure 5.7. The test cases are chosen to simulate the types of structures seen in modal PSD data; these are: a unimodal signal, a bimodal signal with well separated peaks and a bimodal signal with considerable overlap above the noise floor. As an additional challenge, the signals are corrupted by a Gaussian noise with zero-mean and variance of 10^{-2}.

As can be seen in the figure, the proposed metric performs well at detecting unimodality in the presence of noise; correctly assigning the highest metric score to the unimodal signal and lower scores to the others. The metric proposed in this study is a promising tool for the assessment of modal decomposition; however, there are still limitations. For example, the approach is still unable to account for the extent to which the resonance peaks present in the physical displacements have been decomposed into an independent single-peak basis. The current approach is limited in that a decomposition into modal coordinates that each only contained resonance peaks at the same frequencies would be attributed an equal score to one that produces an independent basis.

A metric based on the cosine distance

In [23], a third metric of modal decomposition is introduced based on the idea of the inner product between PSDs. The inspiration behind the metric is that for an independent modal decomposition, each modal degree of freedom should have a PSD with a single distinct peak that is not present in any of the others.

To achieve this, the authors in [23] calculate the average pairwise cosine distance between the modal PSDs. The equation describing this quantity is,

$$m_{\text{cosine}} = \frac{1}{N}\sum_{i,j}^{N} \frac{\mathbf{U_i} \cdot \mathbf{U}_j}{||\mathbf{U}_i||||\mathbf{U}_j||} \quad i \neq j \tag{5.29}$$

for a decomposition into N modal coordinates. The proposed metric is conveniently bounded between $[0, 1]$. In practice, it is not possible to achieve a unity value, as the effects of damping broaden the resonance peaks in the PSDs. However, the metric still offers a useful measure of the modal separation. In [23], the measure is used as a post processing step to select from a number of candidate decompositions, the one that best decomposes the dynamics into single resonance peaks.

5.4 On metrics of modal superposition

Another important aspect of the statistically-independent framework is the notion of modal superposition. This can be measured in the time domain by the evaluation of the reconstruction error after the application of the forward and inverse modal mappings. For example, taking a normalised mean-square error (NMSE),

$$J_{\text{reconstruction}} = \frac{100}{N\sigma_y}\sum_{j}^{N}(\hat{y}_j - y_j)^2 \tag{5.30}$$

where,

$$\hat{y} = f^{-1}(f(y)) \tag{5.31}$$

5.5 Conclusions

In this chapter, the statistically-independent framework for nonlinear modal analysis has been introduced. The framework is seen to offer a promising practical nonlinear extension to liner modal analysis.

Despite its promise, there are a number of limitations to the framework as it is currently posed. One such limitation is that it only considers displacements, and as such is theoretically unable to handle some nonlinear phenomena such as internal resonances [94]. Remedies to this situation are simple to envisage. For example, cross terms could be added as extra dimensions to the modal transformation (as is done within a Rosenberg framework [94]). Alternatively, the modal maps could be lifted into the phase space of the dynamics (as in the Shaw-Pierre framework), such that velocity terms are decoupled alongside displacements.

It is also interesting to imagine if any equivalence could be established linking the statistically-independent NNMs to previous theoretical approaches such as that of Rosenberg [101] or Shaw-Pierre [19]. However, neither the author here nor those of [22] are able to claim any such equivalence at this time.

Also in this chapter, the use of machine learning in the construction of the modal maps has been used to motivate the selection of performance metrics. The idea is that these metrics may be included as objective functions that encourage the specification of useful modal decompositions as measured against the criteria established in Chapter 2.

For each of the criteria, a number of potential metrics have been introduced, including two new methods for the assessment of unimodality in the PSDs of the decomposed modal displacements.

For the measurement of independence, several statistical measures have been introduced, and their performances compared, on a toy dataset with known linear and nonlinear correlations. Of these, the most promising are the linear correlation measure and the nonlinear distance correlation metric. Also considered is the idea of conformal mappings and their use in specifying orthogonal nonlinear mappings.

A limitation of the measures of correlation proposed in this chapter is the lack of a time-dependent measure of correlation. The issue is, that several of the correlation measures considered do not factor in the temporal nature of the data. As such, the measures may therefore fail to detect the presence of periodic dependencies in the data. To overcome this limitation, one could make use of a cross-correlation metric; however, this would incur additional computational expense.

Another limitation is the limited view of what constitutes a good decomposition, as per the criteria of Chapter 2. In this chapter, as in [22], decomposition has been measured in terms of the unimodality of the decomposed PSDs. A potential shortcoming of this approach is that nonlinear systems often display secondary (and tertiary etc.) resonances at harmonics of the principal resonant frequencies. When adopting the statistically-independent approach, the hope is that these harmonics will, in some sense, be 'cancelled out' by the cognitive biases placed on the nonlinear decomposition. This assumption may seem excessive, but it is not a new proposition. The related approach of 'simplifying transformations' [147, 148] takes a similar viewpoint.

Although unimodality of the decomposed spectra is not a bad measure, (certainly, single resonances in the decomposed spectra represent a useful basis for analysis), it is the opinion of the author that this is a somewhat limited view of decomposition. Other factors of at least equal importance include;

- The preservation of resonances present in the physical displacements.
- The extent to which the transformation correctly reduces to the linear modes in the absence of nonlinearity.
- The extent to which the underlying linear dynamics are preserved.
- The extent to which the modal transformation is able to generalise between excitation levels (locally or globally).

These factors are key challenges that any nonlinear extension to modal analysis must overcome. However, measurement of these properties requires some additional analysis which shall be explored in the chapters that follow.

Chapter 6

Statistically independent NNMs: Towards exact NNMs

The statistically-independent NNM framework relies at its core on a nonlinear static map from the physical displacements to the uncorrelated modal coordinates. Thus far in the literature, a machine-learning approach has been taken to learn the modal transformation [22]. However, it is of interest to ask at this stage; can the transformation be derived directly from the equations of motion?

There are several motivations for attempting to compute the nonlinear modal mapping directly from the equations of motion. A key motive is that it is manifestly possible in the linear case. It is therefore argued here, that access to a modal mapping in the case of perfect system knowledge (even in the SISO case), is an important theoretical result in support of the statistically-independent framework. Another important motivation is the reduction in computational cost that can be made by avoiding a machine learning approach.

This chapter examines one method for the construction of analytic modal transformations from the equations of motion. The proposed approach is demonstrated for a single degree of freedom, and challenges pertaining to the MDOF case are presented.

6.1 Direct results from the FPK equation

The approach undertaken here is inspired by the Fokker-Plank-Kolmogorov (FPK) equation. The FPK equation is a partial differential equation, the solution of which is the so-called *stationary probability density*, $p(\mathbf{y}, \dot{\mathbf{y}}|\mathbf{x})$ of a dynamic system. The stationary probability density can be interpreted as the probability of observing a given system

state over the entire time history of the system response, subject to some input $\mathbf{x}$. In the context of statistically-independent NNMs, the stationary probability density is of particular interest. With access to $p(\mathbf{y}, \dot{\mathbf{y}}|\mathbf{x})$, the problem of specifying the modal transformation becomes the specification of a map f such that,

$$f(\mathbf{y}) = \mathbf{u} \tag{6.1}$$

$$\mathbf{y} \sim p(\mathbf{y}, \dot{\mathbf{y}}|\mathbf{x}) \tag{6.2}$$

$$\mathbf{u} \sim \prod_{i}^{n} \alpha_i(u_i|\mathbf{x}) \tag{6.3}$$

where the α_i are target distributions that can be selected arbitrarily to promote independence. For example, one might require that the modal map decomposes displacements into independent Gaussian distributions. The map can then specified either by exact reasoning, or by the application of the change-of-variables equation. For example, in the one-dimensional case, the change-of-variables equation is given by,

$$p_y(y) = p_u(u) \left| \frac{\mathrm{d}u}{\mathrm{d}y} \right| \tag{6.4}$$

Now, substituting in the required form of the transformation,

$$p_y(y) = p_u(f(y)) \left| \frac{\mathrm{d}f(y)}{\mathrm{d}y} \right| \tag{6.5}$$

$$\left| \frac{\mathrm{d}f(y)}{\mathrm{d}y} \right| = \frac{p(u)}{p(f(y))} \tag{6.6}$$

One now has an ODE that, if solved (directly or otherwise), yields the nonlinear transformation from the stationary density to some arbitrary target distribution as required.

Important results regarding the derivation of stationary probability densities for a class of second-order dynamic systems with polynomial-stiffness nonlinearities were developed by Caughey in [149]. The full derivation of the results used here goes beyond the scope of the current investigation but the interested reader is directed to [149] for a thorough reference.

The salient result in the context of the present investigation is that for a nonlinear system of the form,

$$\ddot{\mathbf{y}} + [C]\dot{\mathbf{y}} + \mathbf{F}(\mathbf{y}) = \mathbf{x}(t) \tag{6.7}$$

where C is a diagonal matrix of damping coefficients c, and where $\mathbf{x}(t) \sim \mathbb{N}(\mathbf{0}, \sigma_x^2 I)$. The stationary probability density is given by,

$$p_s(\mathbf{y}, \dot{\mathbf{y}}) = A \times \exp\left[-\frac{2c}{D}\left[\frac{1}{2}\dot{\mathbf{y}}^2 + \int_0^y \mathbf{F}(\zeta)\,\mathrm{d}\zeta\right]\right] \tag{6.8}$$

Where D is the spectral density level of the white-noise Gaussian input, A is the normalising constant and ζ is a dummy variable. For discrete data, D can be related to the variance of the input and the sampling period (Δ_t) by,

$$D = \sigma_x^2 \Delta_t \tag{6.9}$$

6.2 Direct transformations in a single dimension

It may seem bizarre to perform nonlinear modal analysis on a SISO system (and indeed it is!), but the author makes the arguments that the results established here provide an important theoretical illustration for the statistically-independent NNM framework.

Before blindly applying machine learning to the problem of learning $f(\mathbf{y})$, it is a worthwhile endeavour to consider which inductive biases are sufficient to induce the desired properties of a nonlinear modal decomposition. Certainly, naïve application of the modal transformation in a single dimension is insufficient to provide insight into the decomposition and independence criteria, yet there are still important properties that can be investigated.[1] One such property is the extent to which the stationary distributions of nonlinear dynamic systems can be altered by static maps. Another is the way in which these static maps vary with nonlinear intensity and excitation level. Understanding these properties is key to understanding which inductive biases should be built into the machine-learning framework for nonlinear modal analysis.

[1] It is perhaps not even particularly meaningful to describe the transformed coordinates of a SISO oscillator as 'modal' in this context. The term 'transformed nonlinear displacements' will therefore be adopted in this section.

The proposed approach is to learn directly the nonlinear transformation from the FPK and a target distribution (chosen arbitrarily to be a Gaussian). The approach is demonstrated here in the SISO case for a Duffing-type nonlinear system with the equation of motion,

$$\ddot{y} + c\dot{y} + ky + \epsilon y^3 = x(t) \tag{6.10}$$

where $x(t) \sim \mathbb{N}(0, \sigma_x^2)$ is Gaussian, white-noise excitation, with variance σ_x^2.

Starting with the stationary density for a second-order dynamic system with a stiffness nonlinearity (6.8), and substituting in,

$$F = y + \epsilon y^3 \tag{6.11}$$

one finds,

$$p_s(y, \dot{y}) = A \times \exp\left[-\frac{2c}{D}\left[\frac{1}{2}\dot{y}^2 + \int_0^y \zeta + \epsilon\zeta^3 \,\mathrm{d}\zeta\right]\right] \tag{6.12}$$

After performing the integration in the exponent, one has,

$$p(y, \dot{y}) = A \times \exp\left[-\frac{c}{D}\left[\dot{y}^2 + ky^2 + \frac{\epsilon}{2}y^4\right]\right] \tag{6.13}$$

which is the joint stationary probability density of the displacement and velocity up to some constant A.[2]

To recover the density of the displacement only, one notices that the above can be re-written as,

$$p(y, \dot{y}) = p_s(y)p_s(\dot{y}) \tag{6.14}$$

where the densities of the velocities and displacements are independent. Now marginalising over the velocities, one has,

$$p_s(y) = \int_{\dot{y}} p_s(\dot{y}, y) \,\mathrm{d}\dot{y} \tag{6.15}$$

[2] A closed form expression for A is available in terms of parabolic cylinder functions and other higher-order transcendental functions. However in practice, a numerical approximation can be taken—here, a quadrature-based numerical solver is employed—with no damage to the results presented here.

or equivalently,

$$p_s(y) = A \times \exp\left[-\frac{c}{D}\left[ky^2 + \frac{\epsilon}{2}y^4\right]\right] \int_{-\infty}^{\infty} \exp\left[-\frac{c}{D}\dot{y}^2\right] \mathrm{d}\dot{y} \tag{6.16}$$

Because the integral in the above has the form of a Gaussian distribution, one can use the result that the infinite integral over the distribution is equal to unity to recover,

$$p_s(y) = A \times \exp\left[-\frac{c}{D}\left[ky^2 + \frac{\epsilon}{2}y^4\right]\right] \sqrt{\frac{D\pi}{c}} \tag{6.17}$$

Now let,

$$A' = A \times \sqrt{\frac{D\pi}{c}} \tag{6.18}$$

Then the stationary probability density of the displacement of the nonlinear Duffing equation can be recovered as,

$$p_s(y) = A' \times \exp\left[-\frac{c}{D}\left[ky^2 + \frac{\epsilon}{2}y^4\right]\right] \tag{6.19}$$

A similar approach can be taken to recover an expression for the stationary probability density of the displacements, yielding,

$$p_s(\dot{y}) = A'' \times \exp\left[-\frac{c}{D}\dot{y}^2\right] \tag{6.20}$$

Which is clearly a Gaussian distribution. In order to validate the expressions derived here, they are compared to the stationary distributions estimated empirically from simulated data. The data are generated using a fixed-step fourth-order Runge-Kutta scheme. Parameters pertaining to the simulation are collected in Table 6.1. Figure 6.1 depicts the PSDs (estimated by the Welch method), for the linear (obtained by setting $\epsilon = 0$) and nonlinear data generated for this investigation. In the figure, there is clear evidence of nonlinearity, including a hardening effect on the resonance peak and a visible harmonic at three times the principal resonance.

The stationary distribution for the displacements derived above, is plotted against numerical results in Figure 6.2 for the Duffing equation and the equivalent linear system.

As can be seen from the figures, there is excellent agreement between the numerical results and those obtained from the FPK equation.

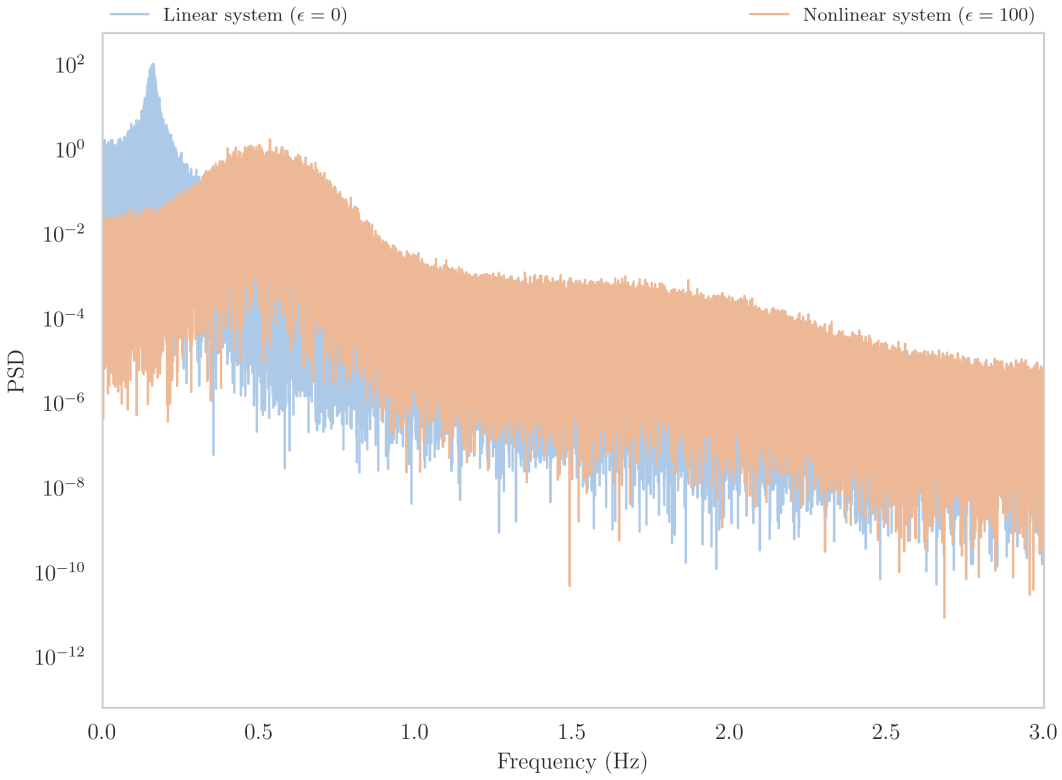


FIGURE 6.1: PSD of linear versus nonlinear system.

TABLE 6.1: Parameters used to simulate the toy dynamics data.

Parameter	Symbol	Value
Mass	m	1
Viscous damping	c	20
Linear stiffness	k	1
Cubic stiffness	ϵ	100
Input excitation level	σ_x^2	1
Runge-Kutta method order		4
Sampling frequency (Hz)	f_s	10
Dataset size	N	10^6

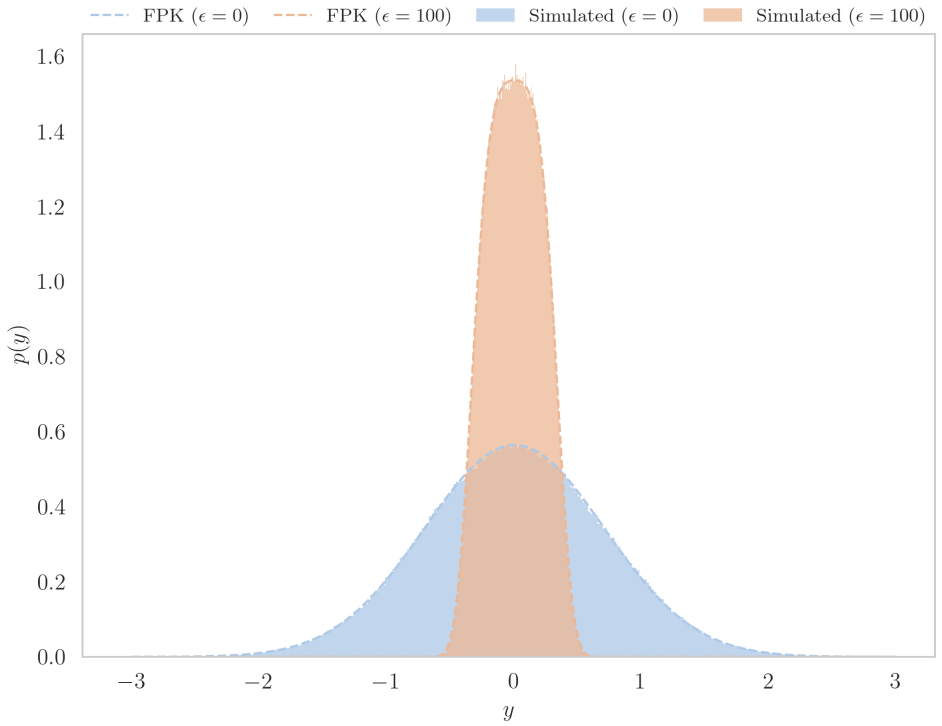


FIGURE 6.2: Stationary densities of simulated data versus predictions from the FPK.

It is now interesting to consider a static nonlinear map that renders the transformed displacements Gaussian. In the context of the statistically-independent framework, this is akin to the statistical-independence criterion (albeit trivially in a single dimension for now). The desired forward nonlinear transformation has the form,

$$f(y) = u, \quad u \sim \mathbb{N}(\mu_u, \sigma_u^2) \tag{6.21}$$

It is required that $f : y \to u$ be a static, continuous and invertible map; one can thus apply the change-of-variables equation and write,

$$\left| \frac{\mathrm{d}f(y)}{\mathrm{d}y} \right| = \frac{p(u)}{p(f(y))} \tag{6.22}$$

If f is additionally constrained to be a monotonic function,

$$\frac{\mathrm{d}f(y)}{\mathrm{d}y} > 0 \quad \forall y \tag{6.23}$$

then,

$$\frac{\mathrm{d}f(y)}{\mathrm{d}y} = \frac{p(u)}{p(f(y))} \tag{6.24}$$

Substituting in the definition of $p(y)$ and replacing $p(u)$ with the required form of a Gaussian distribution gives,

$$\frac{df(y)}{dy} = \frac{A'}{\sqrt{2\sigma_u^2\pi}} \times \exp\left[-\frac{(f(y)-\mu_u)^2}{2\sigma_u^2} + \frac{c}{D}\left[ky^2 + \frac{\epsilon}{2}y^4\right]\right] \tag{6.25}$$

Where the parameters of the target distribution are set to,

$$\sigma_u^2 = \frac{ck}{2D} \tag{6.26}$$

$$\mu_u = 0 \tag{6.27}$$

The above is a first-order nonlinear ODE in $f(y)$. Although the ODE is separable in f and y, the integral in f is intractable analytically and so a numerical solution must be adopted. Since all the distributions considered here are symmetric, it is sufficient to only consider the half-plane IVP with $y > 0$ and $f(0) = 0$ as an initial condition. The full map can then be recovered by reflecting the positive solution about $y = 0$. In this investigation, a fixed-step Runge-Kutta IVP algorithm is used to solve (6.25). The integration provides a discrete representation of f, that is interpolated linearly to form a continuous mapping. Finally, the nonlinear displacements are transformed onto the modal displacements u that now obey the target distribution.

Remark 6.1. Although a Gaussian target distribution has been selected for $p(u)$, in practice there is no restriction on the form of the target (besides independence). The author envisages that the choice of $p(u)$ will depend on physical insights (i.e. bounding, distribution support) that one might be able to make about the nonlinear system under investigation.

Figure 6.3 depicts histograms of the simulated nonlinear and transformed data against the original and target distributions. As can be seen in the figure, the quality of the transformation is excellent, the modal data fits the target distribution exactly.

It is of interest at this stage to consider the effects of varying nonlinearity and excitation level on the form of the nonlinear transformation. In Figure 6.4, the physical and transformed data are plotted and compared over a range of values for the nonlinear stiffness parameter. Also plotted are third-order polynomial fits to the forward nonlinear

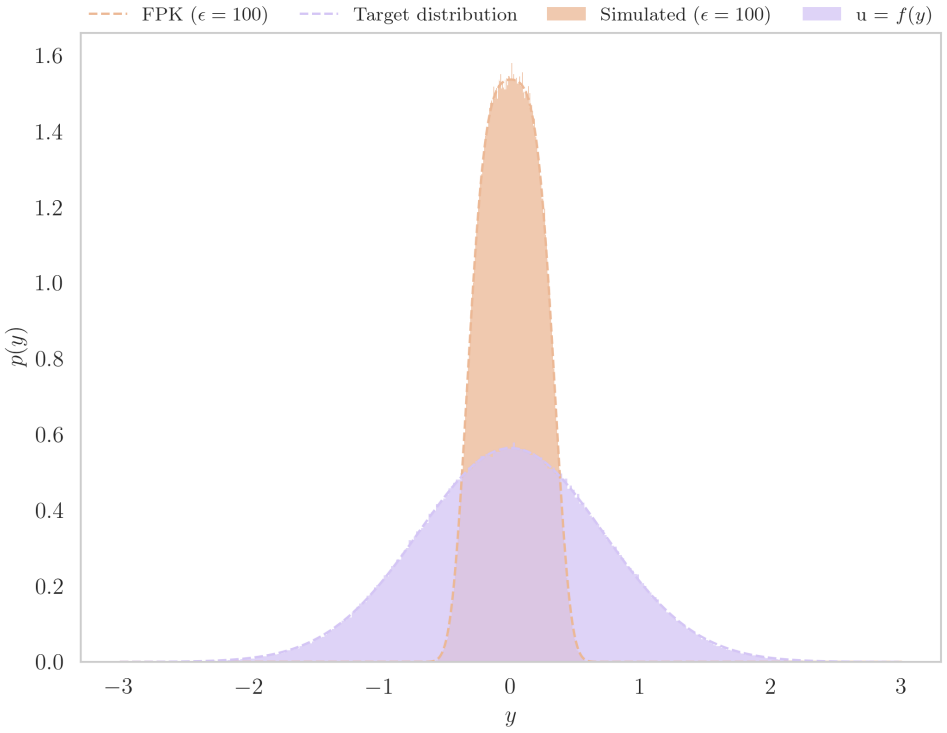


FIGURE 6.3: Histograms of physical and transformed displacements compared to FPK prediction and target distribution.

transformations in each case. As expected, with $\epsilon = 0$ the nonlinear transformation is trivially the unity map. As the level of nonlinearity increases, the cubic structure of the required transformation becomes more pronounced. The data (obtained from integrating equation (6.25) forward in time) are used to fit polynomial models of the nonlinear transformation. Examination of the least-squares polynomial models of these data is of interest as some insight can be gleaned as to the form of the mapping.

In several cases, towards the edges of the distributions, the measured samples are seen to deviate from the least-squares polynomial fit. The author reasons that these discrepancies are caused by numerical issues in the evaluation of (6.25). As the magnitude $|y| \to \infty$, the numerator and denominator of the change-of-variables equation both rapidly approach zero, leading to instability in the numerical integration procedure. Note however, that these errors do not cause any visible damage to the quality of the fit in Figure 6.3; this is because the instability only affects the very extremes of the distributions, where very few samples are observed.

Another important consideration at this stage is the amplitude invariance of the proposed approach. Figure 6.5 depicts the forward nonlinear transformation and the corresponding least-squares polynomial fit, calculated at a range of excitation levels. It is

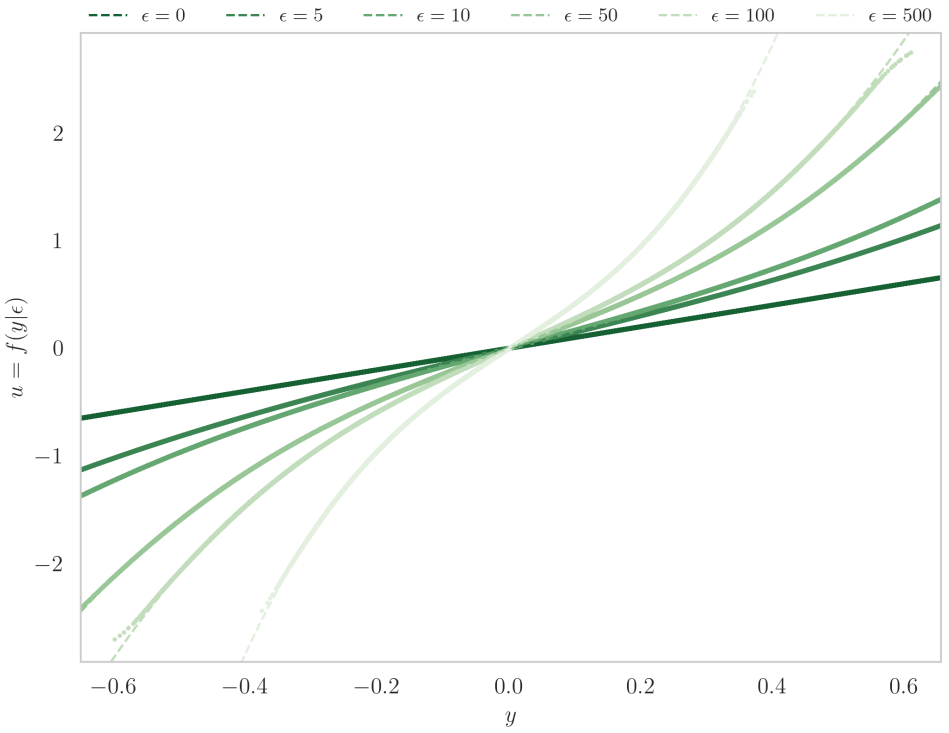


FIGURE 6.4: Comparison of the nonlinear mapping $f(y)$ with varying nonlinear stiffness ($\sigma_x^2 = 1$).

interesting to note that the nonlinear transformations are not independent of amplitude. This result implies that amplitude invariance cannot be guaranteed under the statistically independent framework using only a Gaussian target distribution. In order to achieve a modal transformation with amplitude invariance, additional inductive biases must be introduced to the framework.

6.2.1 A cautionary result

The choices of $\sigma_u^2 = \frac{ck}{2D}$ and $\mu_u = 0$ in the analysis above were not arbitrary, but deliberate. Consider the effect of taking $\epsilon = 0$ in equation (6.19).

$$p_s(y) = A' \times \exp\left[-\frac{c}{D}\left[ky^2 + \frac{0}{2}y^4\right]\right]$$

$$p_s(y) = A' \times \exp\left[-\frac{ck}{D}y^2\right] \tag{6.28}$$

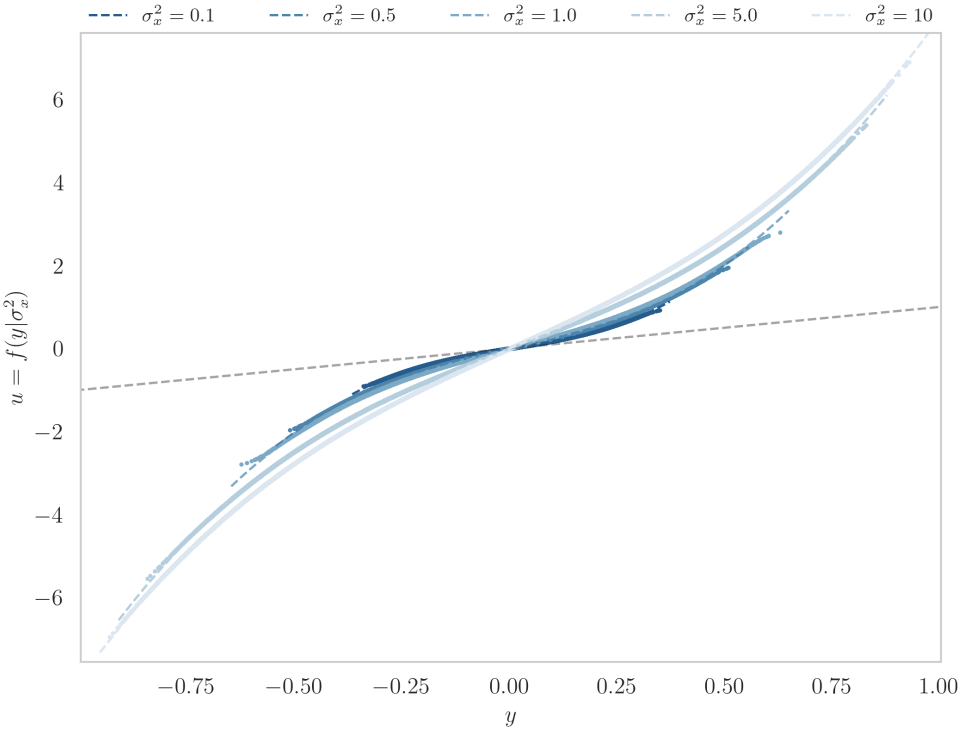


FIGURE 6.5: Comparison of the nonlinear mapping $f(y)$ with varying excitation level ($\epsilon = 100$).

which is a Gaussian distribution with mean 0 and variance $\frac{ck}{2D}$. In effect, the map f projects the density of the nonlinear displacements onto a distribution wherein there is no effect from the nonlinearity.

The following line of reasoning is now a tempting (but erroneous) one. Because the stationary probability density of $u = f(y)$ is the same as that of the underlying linear system (with $\epsilon = 0$) given the same white-noise input, the dynamics of $x \to u$ should therefore be linear.

Certainly, the converse argument is true. However a disproof of the above is straightforward. One begins by substituting the map f into the equation of motion for the Duffing oscillator.

$$\frac{\mathrm{d}^2}{\mathrm{d}t^2}g(u) + c\frac{\mathrm{d}}{\mathrm{d}t}g(u) + kg(u) + \epsilon g(u)^3 = x(t) \tag{6.29}$$

where,

$$g(u) = f^{-1}(u) = y \tag{6.30}$$

Evaluating the time derivatives of $g(u)$,

$$\frac{\mathrm{d}}{\mathrm{d}t}g(u) = \frac{\mathrm{d}g}{\mathrm{d}u}\frac{\mathrm{d}u}{\mathrm{d}t} = g'(u)\dot{u} \tag{6.31}$$

$$\frac{\mathrm{d}^2}{\mathrm{d}t^2}g(u) = \frac{\mathrm{d}}{\mathrm{d}t}\left[g'(u)\dot{u}\right] = g''(u)\dot{u}^2 + g'(u)\ddot{u} \tag{6.32}$$

and substituting into the equation of motion, gives,

$$g''(u)\dot{u}^2 + \ddot{u}g'(u) + cg'(u)\dot{u} + kg(u) + \epsilon g(u)^3 = x(t) \tag{6.33}$$

which, despite a Gaussian stationary density for x and u is clearly a nonlinear ODE.

6.3 Towards direct MDOF NNMs

Although it has been shown that a modal transformation (of sorts) is available in the SISO case, it is of no practical interest for performing nonlinear modal analysis. Unfortunately, the method presented above cannot be trivially extended to the higher-dimensional case. The principal issue is the form of the change-of-variables equation for higher-dimensional mappings. For n degrees of freedom, the static forward modal transformation $f(\mathbf{y}) = \mathbf{u}$, is a function with n inputs and n outputs. The change of variables equation is thus,

$$|J_f| = \frac{p(\mathbf{u})}{p(f(\mathbf{y}))} \tag{6.34}$$

where J_f is the Jacobian of f, and $|\cdot|$ denotes the determinant operation. The above expression cannot be uniquely solved for f and once again a machine learning-approach must be adopted. However, for nonlinear systems where the equations of motions are known exactly, the above expression can be used as an objective function whereby the quantity,

$$J_{\mathrm{FPK}}(f, \mathbf{y}) = |J_f| - \frac{p(\mathbf{u})}{p(f(\mathbf{y}))} \tag{6.35}$$

can be minimised within an optimisation framework to enforce an arbitrary target distribution in the modal space. Formulating the machine-learning problem in this way bears close resemblance to the method of normalising flows (NF) [150]. Indeed, a NF

approach to generating the statistically-independent NNMs has been presented in the literature [128], with comparable results to other approaches.

Given the difficulty in solving (6.34), direct access to the modal transformation is not considered further at this stage. However, there are still promising avenues for further investigation, including the inclusion of the change-of-variables equation in the objective function when the stationary density of the physical displacements can be accessed (analytically from the FPK equation or otherwise).

6.4 Conclusions

In this chapter, the problem of specifying exact modal transformations under the statistically-independent framework has been treated. An approach to constructing the maps using the FPK equation and the change-of-variables equation is presented. In the limited SISO case, it is shown that a nonlinear transformation can be specified by the numerical solution of a nonlinear ODE. This has furthermore been demonstrated on a simulated benchmark dataset. It is shown that the modal transformations specified in this way depend locally on the level of input excitation.

It is interesting to imagine whether there might exist a choice of $p(u)$ (parametrised by the input excitation) that permits an amplitude-invariant forward nonlinear transformation (excluding of course the trivial $p(u) = p(y)$). It is certainly hard to imagine that this would be possible in general for nonlinear dynamic systems. However this is an interesting avenue for further investigation.

Also considered in this chapter is the extension of the method to the multiple degree-of-freedom case. It is shown that the differential equation arising from the change of variables equation in this case cannot be directly solved for the modal transformation f. Instead, a machine-learning approach is envisaged, that uses the Jacobian of the modal transformation in the objective function to encourage conformance with a target distribution in the modal coordinates.

Although the analysis in this chapter has produced some interesting results in the context of the statistically-independent NNM framework, some important limitations remain. Chief among these, is the lack of insight into the transformed modes. Of particular interest in practice are the following questions;

- What effect does f have on the resonances of the response?
- Have the underlying linear dynamics been altered?

- Does the modal transformation introduce additional nonlinearities?

Although some reasoning might be made in the context of the transformed equation of motion in (6.33), a full investigation of these is saved for later chapters, once some further machinery for analysis is established.

Chapter 7

Generating statistically-independent NNMs

This chapter will present some case study examples of generating nonlinear normal modes under the statistically-independent framework. Two approaches from the literature are considered for generating the forward and inverse modal maps f and f^{-1}.

Although no resolutely novel methodology is presented here for specifying the NNMs[1], the novel analysis of the statistically-independent NNMs (in a following chapter), requires that some NNMs be constructed. The objective of this chapter is therefore twofold. Principally, NNMs are constructed for some case-study nonlinear systems for later analysis. Additionally, the opportunity is taken to evaluate the effect that the different inductive biases have on the qualitative performance of the modal transformations.

The first method for generating NNMs considered in this chapter is the original approach of Worden and Green in [22]. Here, the forward modal mapping is learned by an explicitly-parametrised multinomial model of fixed order. The parameters of this multinomial transformation are specified by the application of a heuristic optimisation algorithm. Cognitive biases are introduced by an objective function that considers pairwise correlations between the decomposed displacements.

The second approach demonstrated in this chapter is derived from the recently proposed work of [23]. In the paper, the authors propose a neural formulation of the forward and inverse maps. By the application of a cycle-consistent generative adversarial network (cycle-GAN) [151], the authors are able to jointly learn both the forward and inverse

[1] The consideration of the distance correlation metric from Chapter 5 in the multinomial decomposition may be viewed as a minor yet interesting contribution.

modal transformations. In addition, the inductive bias of *conformality* (as per the cosine distance metric introduced in Chapter 5), encourages a practical decomposition into independent resonances.

There are in fact several other approaches that have been presented in the literature for the generation of NNMs in the statistically-independent framework that are not considered in this chapter. These include approaches based on non-parametric methods [126], normalising flows [128] and another based on an auto-encoder model [129]. Although these techniques have some desirable properties, the author feels that the best comparison for the sake of this chapter is between the original formulation of [22] and the state of the art neural architecture method of [23].

In the literature, statistically-independent NNMs have already been generated for a range of nonlinear structural 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 [22, 23]. It is the opinion of the author that the efficacy of the statistically-independent framework is now well-established. It is not of particular interest to explore additional types of nonlinearity or experimental case studies in this thesis. Although such work is certainly fertile ground for future consideration, the author feels that a more pressing requirement is the reconciliation of the data-driven framework with physical insight.

With the above objectives in mind, the two approaches to generating the NNMs are each demonstrated on two benchmark case studies. A 2-DOF and a 3-DOF nonlinear system with a cubic stiffness element. Although is might seem that consideration of only simulated case-studies is a weakness of the results here, direct access to the equations of motion will prove valuable for the analysis that is to follow. The quality of the nonlinear modal decompositions are judged visually (a more robust analysis is presented in a later chapter), in terms of the extent to which the decomposed power spectral densities (PSDs) show distinct resonance peaks.

7.1 Benchmark nonlinear dynamical systems

In this chapter, the NNMs of two systems with a static cubic stiffness nonlinearity will be extracted. Figure 7.1 gives a schematic view of the 2-DOF system under investigation.

For 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,

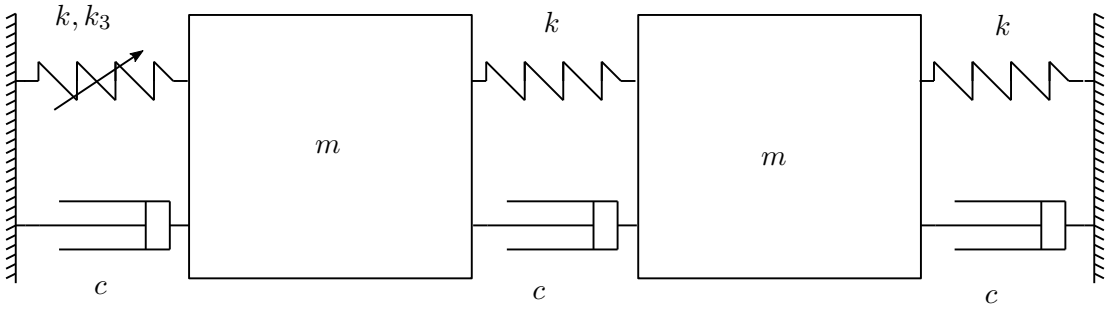


FIGURE 7.1: Schematic of the 2-DOF nonlinear system.

TABLE 7.1: 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

$$M\ddot{\mathbf{y}} + C\dot{\mathbf{y}} + K\mathbf{y} + K_3\mathbf{y}^3 = \mathbf{x}(t) \tag{7.1}$$

where, M, C, K, K_3 are square parameter matrices corresponding to the nonlinear system depicted in Figure 7.1, and have values as given by Table 7.1.

To generate the time-series data, 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. For excitation, white Gaussian noise $x \sim N(0, \sigma_x)$ is applied to the first degree-of-freedom of each system. An additional low-pass filter is applied to the excitations with a cut-off frequency of 50Hz. 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 7.1.

In order to verify that the dynamics are indeed nonlinear, the spectral densities (estimated by the Welch method [152]) of the nonlinear data are compared to the equivalent system with the nonlinear elements removed. Nonlinear cubic hardening is demonstrated for the 2-DOF system and $\sigma_x = 20$ in Figure 7.2. In the plot there is clear evidence of hardening of the resonance peaks from the inclusion of the cubic stiffness. In the modal

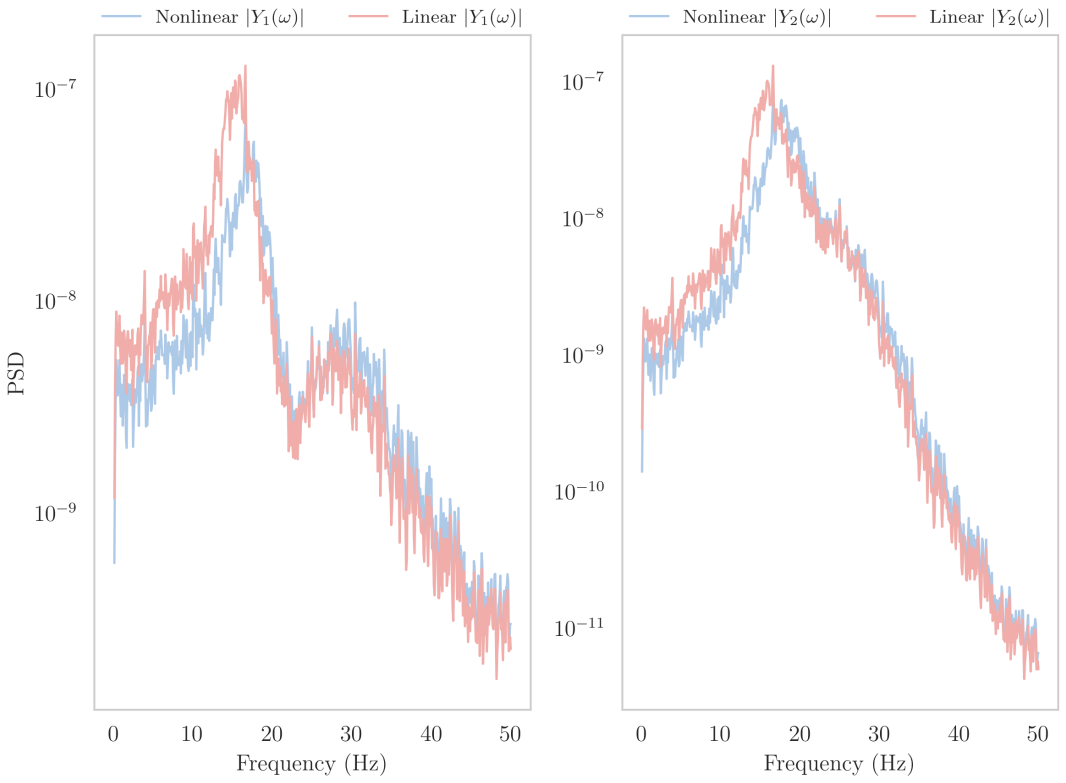


FIGURE 7.2: PSDs (estimated by the Welch method [152]) for the 2-DOF nonlinear and underlying linear system, the nonlinear data shows a clear hardening via shifts in the principal resonance frequencies.

analyses that follow, the data from the highest level of excitation (i.e most nonlinear) will be used.

7.2 Meta-heuristic optimisation

With the nonlinear benchmark data established, the NNMs can be generated. The first approach presented in this chapter is derived from the multinomial expansion of [22]. In the paper, the authors choose a parametric form for the forward modal transformation f. The parameters of the transformation are then optimised according to an objective function that encodes the inductive biases of the NNM framework. In [22], a multinomial expansion of f is taken,

$$\mathbf{u} = A_1\mathbf{y} + A_2\Pi_2(\mathbf{y}) + A_3\Pi_2(\mathbf{y}) + \dots \tag{7.2}$$

where the A_i are parameter matrices and where $\Pi_i(\mathbf{y})$ represents a vector of products of length m combinations of the elements of $\mathbf{y}$. Clearly, the number of free parameters in

the above equation grows combinatorically[2] with the order of the multinomial and the number of degrees of freedom. In order to reduce the difficulty of the resulting optimisation problem somewhat, the authors of [22] reason that for a cubic-type nonlinearity, only the linear and third-order terms need be included. The resultant transformation is thus,

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} + \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \end{bmatrix} \begin{bmatrix} y_1^3 \\ y_1^2 y_2 \\ y_1 y_2^2 \\ y_2^3 \end{bmatrix} \tag{7.3}$$

for the 2-DOF system and,

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} + \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1;10} \\ a_{21} & a_{22} & \dots & a_{2;10} \\ a_{31} & a_{32} & \dots & a_{3;10} \end{bmatrix} \begin{bmatrix} y_1^3 \\ y_1^2 y_2 \\ y_1^2 y_3 \\ y_2^3 \\ y_2^2 y_1 \\ y_2^2 y_3 \\ y_3^3 \\ y_3^2 y_1 \\ y_3^2 y_2 \\ y_1 y_2 y_3 \end{bmatrix} \tag{7.4}$$

for the 3-DOF system.

For the objective function, the inductive biases of independence and orthogonality are enforced. For the former, the pairwise correlations between the transformed modal coordinates are calculated. Here, two metrics for the assessment of correlation are considered. Following the analysis presented in Chapter 5, optimisation runs are conducted using both the Pearson product correlation coefficient and the distance correlation metrics.

In order to encourage orthogonality, an additional objective term is included that penalises the cosine distance between the columns of the linear part of the transformation. This approach (taken from [22]), is motivated by the fact that in linear modal analysis the columns of the modal matrix are all orthogonal to each other. The overall objective function can be written,

[2]The number of free parameters in a multinomial transformation up to order k in n variables (i.e degrees of freedom) is given by $\sum_{i=1}^{k} \frac{(i+2n-1)!}{i!(2n-1)!}$.

$$J = \lambda_1 J_{\text{orthogonality}} + \lambda_2 J_{\text{correlation}} \tag{7.5}$$

$$J_{\text{orthogonality}} = \sum_i \sum_j |a_1^{(i)} \cdot a_1^{(j)}| + |a_2^{(i)} \cdot a_2^{(j)}| + \dots \quad i \neq j \tag{7.6}$$

$$J_{\text{correlation}} = \sum_i \sum_j |\operatorname{cor}(u_1, u_2)| + |\operatorname{cor}(u_1, u_3)| + \dots \quad i \neq j \tag{7.7}$$

Where λ_1, λ_2 are weighting terms that are each (arbitrarily) set to unity.[3] It should be noted here that the objective function presented above differs slightly from that which was employed in [22]. In the paper, the authors explicitly include third-order correlations (i.e. $\operatorname{cor}(u_1, u_2^3)$ etc.) in the objective function. These terms are neglected here with the reasoning that distance correlation metric should be able to handle the nonlinear correlations implicitly.

Differential evolution

With the objective function established, all that remains is to perform the optimisation of the multinomial coefficients. Because the gradients of the above objective function with respect to the parameters are not (trivially) available in closed-form, the optimisation will be performed by heuristic means.

Heuristic optimisation encompasses a class of algorithms that are able to perform optimisation in the absence of gradients or in the presence of challenging optimisation environments (multiple objectives, local minima, nonsmooth search spaces). Many heuristic optimisation algorithms are derived from the more general class of genetic algorithms (GAs), that are introduced in Chapter 4. Although a great number of algorithms have been presented in the literature (a subset of popular algorithms can be found in [141, 153–155]), a popular choice is methods based on *differential evolution* (DE) [142].

The structure of DE is essentially identical to that of a genetic algorithm. The algorithm is comprised of an initialisation step, followed by a number of generations during which a population of trial solutions is optimised towards a global optimum. During each generation, new candidate vectors are produced by taking the numerical difference between parent vectors (from the previous generation), and adding them to a third parent vector. This genetic reproduction procedure imitates the evaluation of gradient information

[3] It is argued here that *a priori* there is no reason to weight one of these terms more strongly than the other in the objective function.

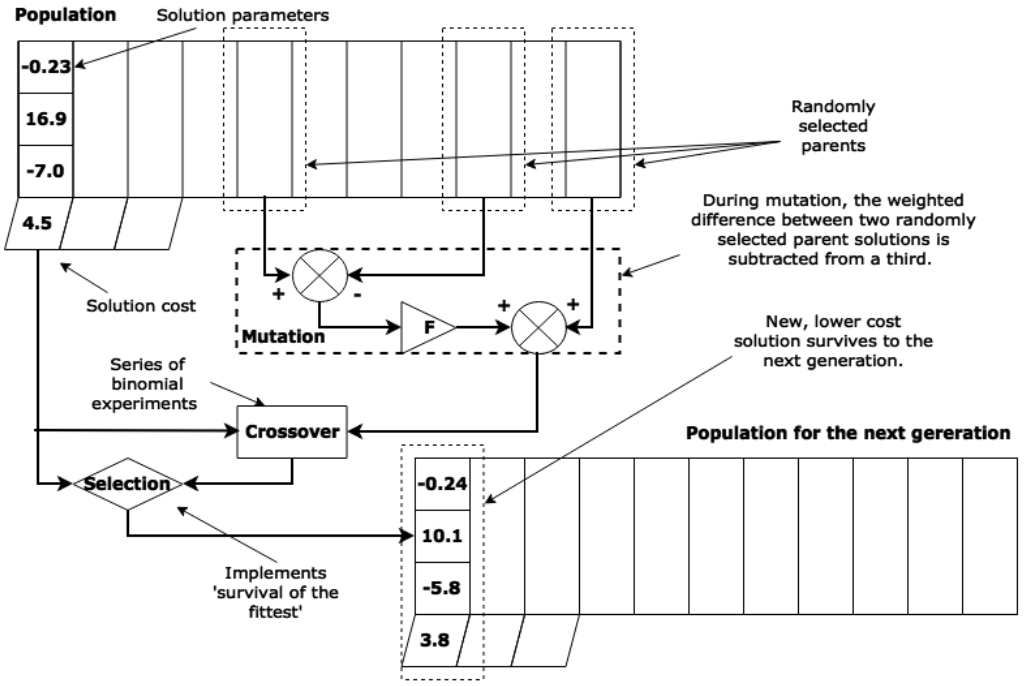


FIGURE 7.3: Schematic depiction of a single generation of the DE algorithm.

and makes the DE algorithm particularly effective at optimisation within smooth local minima.

Several extensions to the original DE algorithm have been proposed in the literature, including a greedy-search approach (JADE) [156]. The DE algorithm was extended to the class of meta-heuristic optimisers (whereby hyperparameters are optimised during the run) by Qin and Suganthan in [140]. So-called *Self-adapting differential evolution* (or SADE to her mates), has been applied to a great number of problems in engineering and beyond [31, 32, 42, 157, 158] and so is an excellent candidate for the optimisation task at hand.

A full description of the DE/SADE algorithm is somewhat beyond the scope of the current chapter (and indeed is well treated elsewhere), the interested reader is directed to the original papers [142, 159] for additional detail on implementation or to [160] for a benchmark comparison of several heuristic optimisation methods. The implementation used in the studies presented here is taken from the python package 'FreeLunch', developed by the author, which is available on the python package index[4]. However, a schematic of the procedure of a single generation of the DE algorithm is depicted for the readers convenience in Figure 7.3.

[4] `https://pypi.org/project/freelunch/`

TABLE 7.2: Parameters pertaining to the SADE optimisation procedure, note that smaller optimisation sizing parameters are used to account for the significantly increased computational complexity of the distance correlation method.

Parameter	Description	Value
G	Number of generations (per run)	$\{10^4, 2 \times 10^3\}$
N	Population size	$\{10^4, 2 \times 10^3\}$
F_μ	Learning rate initial mean	0.5
F_σ	Learning rate initial standard deviation	0.2
Cr_μ	Crossover rate initial mean	0.2
Cr_σ	Crossover rate initial standard deviation	0.1
L_p	Learning rate	10

To specify parameters to the nonlinear map, the SADE algorithm is employed here. First, an initial population of candidate parameters is generated. In order to assist the search, the linear parameters (the elements of the A_1 matrix), are initialised from a Gaussian distribution centred around the PCA components of the physical displacements and with standard deviation $\sigma_a = 0.5$. Parameters of the nonlinear terms are initialised uniformly on the interval $[-0.5, 0.5]$. During each optimisation run, parameter values are prevented from exceeding an absolute value of 100.

For each system the optimisation was re-initialised and run 100 times. Both the Pearson correlation coefficient and the distance correlation metrics are considered. For the former, 10^4 points were used to evaluate the correlation and the population size was set to 10^3 individuals optimised over 10^3 generations. Because of the increased computational complexity of the distance correlation metric[5], only 2×10^3 points were used to evaluate the metric, and population size was set to 500 individuals optimised over only 500 generations. Parameters pertaining to the optimisation problem are collected for the readers convenience in Table 7.2.

Results

Once each of the optimisation runs are completed, the best transformation is selected as the parameter set that provides the lowest objective function score. These parameter vectors are then used to generate the modal displacements for the entire time-series. PSDs of the physical and modal coordinates are estimated by the Welch method and compared in the following figures.

Figures 7.4 and 7.5 depict the original and transformed PSDs for the 2-DOF system for the Pearson and distance correlation metrics respectively. In the former, the quality of

[5]Pearson product moment correlation coefficient can be computed in $\mathcal{O}(n)$ time whereas the distance correlation metric requires nominally $\mathcal{O}(n^2)$. A $\mathcal{O}(n \log n)$ algorithm was also recently proposed [146]. Results in this chapter use the implementation available as part of the 'dcor' python library [161].

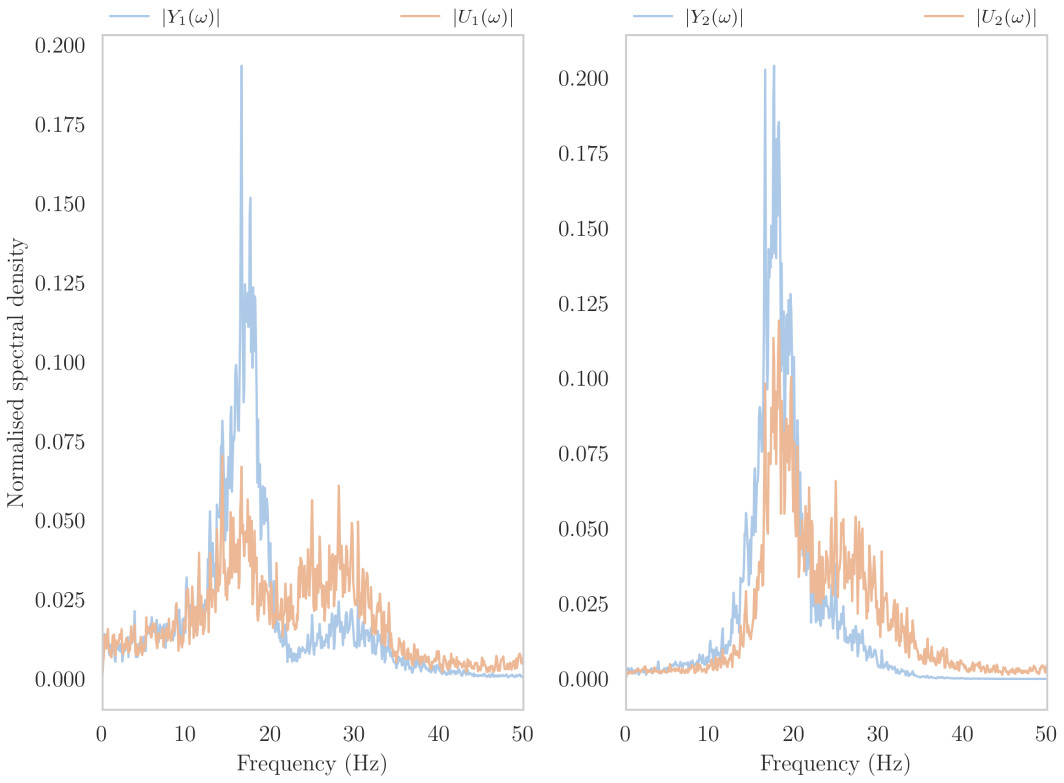


FIGURE 7.4: 2-DOf system: Physical vs. modal PSDs - multinomial expansion with cor.

the separation (as judged 'by-eye' for independent resonances) is poor. In comparison, the decomposed spectra generated by minimisation of the distance correlation in Figure 7.5 shows a far greater level of separation. At first sight it might appear that the improved separation can be attributed to the nonlinear correlation metric. However, visual assessment of each of the 100 best solutions from each run, finds examples from both metrics that provide a good decomposition by-eye. It should be noted here that the ordering of the modal decomposition is essentially random, this is because the columns of the transformation matrices can arise in arbitrary order without affecting the value of the objective function.

The resultant nonlinear maps for the 2-DOF system are plotted in Figures 7.6 and 7.7. As expected, both figures depict a linear region close to the origin and cubic structure in the extremities.

Results from the 3-DOF investigation are depicted in Figure 7.8 for the Pearson correlation coefficient and Figure 7.9 for the distance correlation metric. As before, the best performing transformation trained to minimise the pairwise cross-correlations measured by Pearson's correlation coefficient has failed to produce a visually-distinct set of resonant peaks. The performance of the best performing distance correlation has likewise

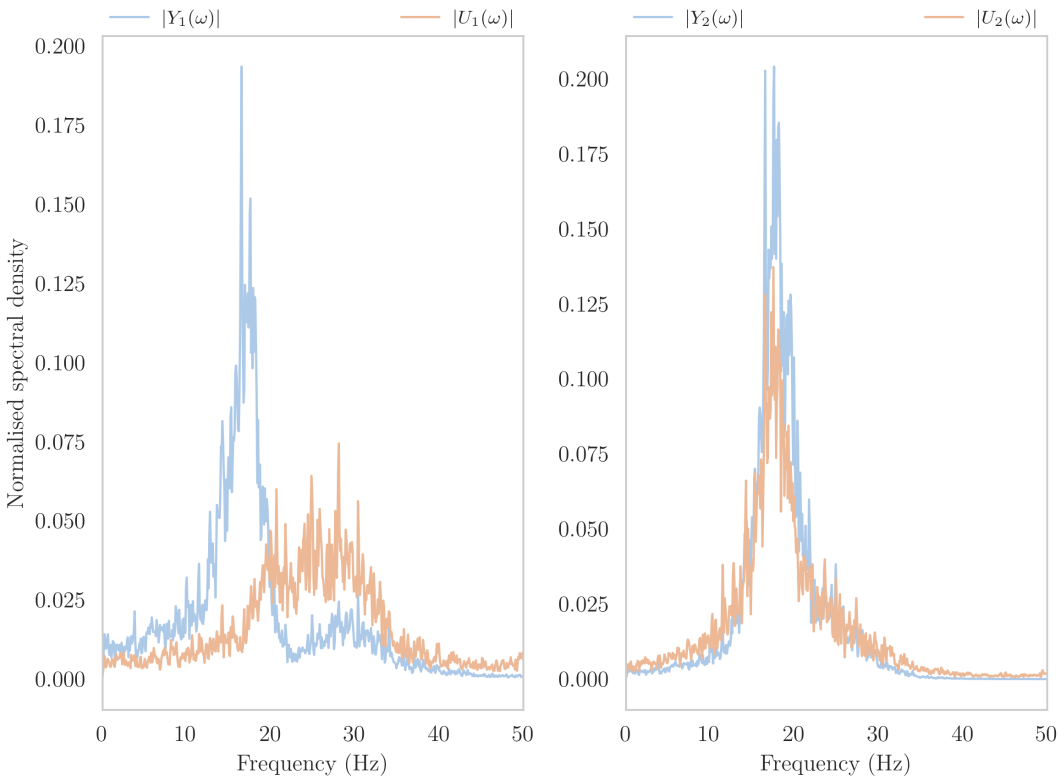


FIGURE 7.5: 2-DOf system: Physical vs. modal PSDs - multinomial expansion with dcor.

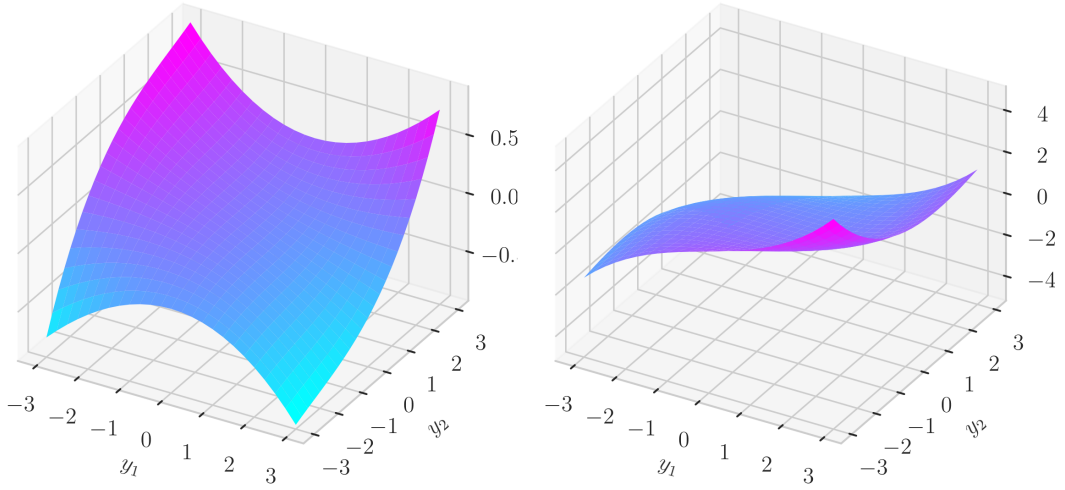


FIGURE 7.6: Forward nonlinear map - multinomial expansion with cor.

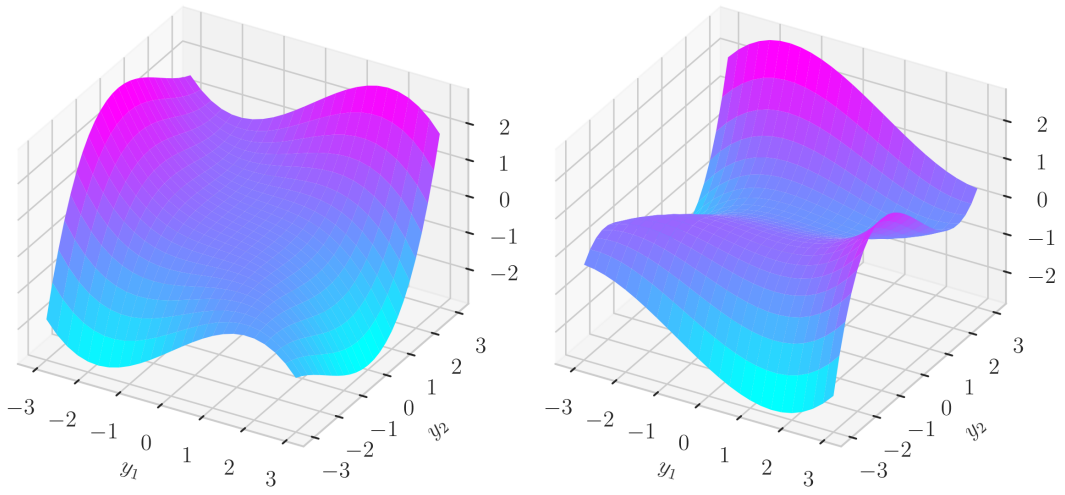


FIGURE 7.7: Forward nonlinear map - multinomial expansion with dcor.

failed to produce visually-distinct resonances, indicating that elimination of the correlations (linear or otherwise) is insufficient to guarantee a decomposition into distinct resonance peaks.

Once again, examples of transformations that perform visually better (and worse), can be found amongst the best transformations from each of the 100 runs of the optimiser, adding further evidence that an additional inductive bias is required to achieve a consistent decomposition into distinct resonances.

7.3 Cycle-GAN

Although there is evidence that good modal decompositions can be found by the heuristic optimisation approach of [22], several disadvantages remain. A recent contribution [23], demonstrates that *cycle-consistent generative adversarial networks* (cycle-GAN) can provide several advantages. The inverse mapping is handled naturally by the neural architecture 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.

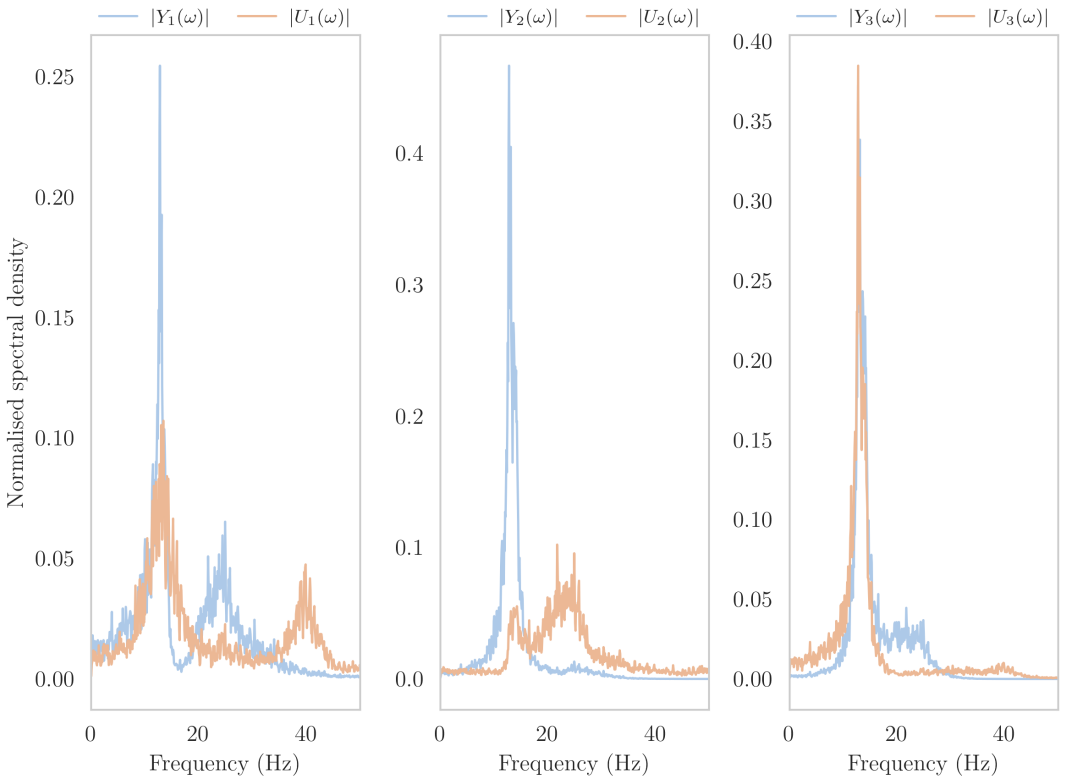


FIGURE 7.8: 3-DOf system: Physical vs. modal PSDs - multinomial expansion with the cor.

The notion of using a neural architecture to learn a decomposition mapping is not a new one. There are several techniques such as auto-encoders [162] or their variational extensions [163, 164] that attempt to build latent representations of data based on inductive biases. However, any reasonable framework for NNMs should have as many modal directions as there are physical degrees-of-freedom.[6] Network architectures that rely on a bottleneck layer become problematic in this regard. If the dimensionality of the latent space is selected to be equal to that of the input, the network becomes susceptible to the degenerate unity-map case. The cycle-GAN does not have such a limitation, the specification of a target latent dimension forces the network to learn a meaningful decomposition.

The structure of the cycle-GAN model is essentially comprised of two generative adversarial networks (GANs) [165], 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[7] each possessed of a single hidden layer. Of these two networks, the first, -

[6] Even if a reduced-order representation is desirable, it is natural that the full-fidelity representation should have as many modes as degrees-of-freedom. The author notes that this must be the case if a reduction to linear modal analysis is desired in the limit of linearity.

[7] Historically referred to as a *multi-layer perceptron* (MLP).

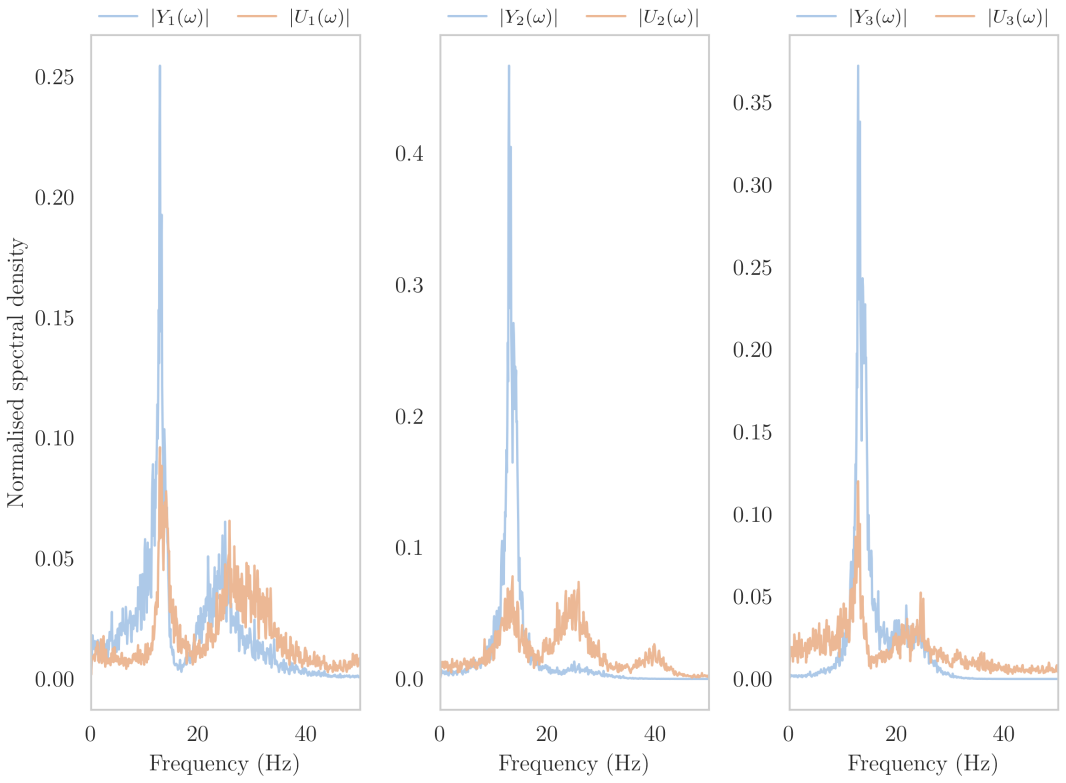


FIGURE 7.9: 3-DOf system: Physical vs. modal PSDs - multinomial expansion with dcor.

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 [23], 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 [151],

$$\begin{aligned} \mathcal{L}_1^{\mathbf{y}} &= \mathbb{E}[\log D_f(\mathbf{y})] + \mathbb{E}[\log(1 - D_f(f(\mathbf{u})))] \\ \mathcal{L}_1^{\mathbf{u}} &= \mathbb{E}[\log D_{f^{-1}}(\mathbf{u})] + \mathbb{E}[\log(1 - D_{f^{-1}}(f^{-1}(\mathbf{u})))] \end{aligned} \tag{7.8}$$

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,

$$\begin{aligned}\mathcal{L}_2^{\mathbf{y}} &= \|\mathbf{y} - f(f^{-1}(\mathbf{y}))\|_2 \\ \mathcal{L}_2^{\mathbf{u}} &= \|\mathbf{u}^* - f^{-1}(f(\mathbf{u}^*))\|_2\end{aligned} \tag{7.9}$$

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 [23]. The networks are trained over a number of epochs consisting of 2048 samples per epoch using an ADAM [166], stochastic gradient-descent algorithm. The loss from a given training epoch is given by,

$$\mathcal{L} = \lambda_1(\mathcal{L}_1^{\mathbf{y}} + \mathcal{L}_1^{\mathbf{u}}) + \lambda_2(\mathcal{L}_1^{\mathbf{y}} + \mathcal{L}_1^{\mathbf{u}}) + \lambda_3\mathcal{L}_3 \tag{7.10}$$

where the λ_i are weights in the objective function. Following the approach of [23] 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 Figure 7.10.

Following [23], 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.

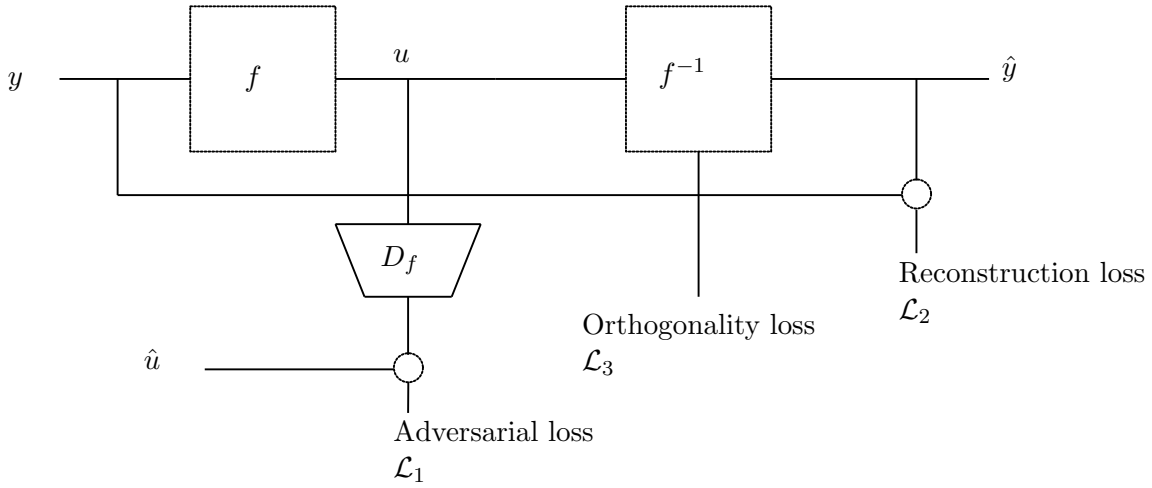


FIGURE 7.10: Overview of the cycle-GAN structure for the forward pass.

TABLE 7.3: 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

The network structures and parameters are detailed in Table 7.3. 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 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 (PSDs) of modal coordinates $P\mathbf{u}$ and is given by,

$$\mathcal{L}_{\text{inner}} = \sum_{i=1,j=i+1}^{n_{\text{dof}}} \frac{Pu_i \cdot Pu_j}{\|Pu_i\| \, \|Pu_j\|} \tag{7.11}$$

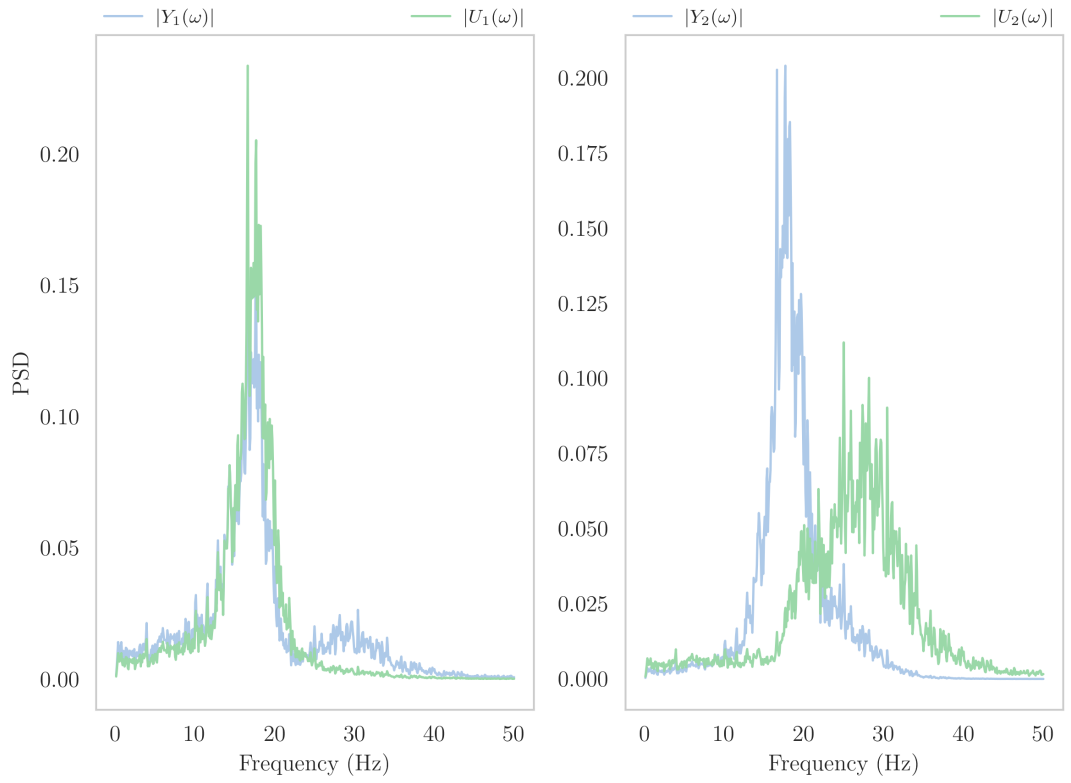


FIGURE 7.11: 2-DOf system: Physical vs. modal PSDs - cycle-GAN decomposition.

In the above, the Pu_i and Pu_j are estimated by a Welch method [152]. This choice leads to the introduction of some additional hyperparameters arising from the window length and overlap size in the Welch method. The author's 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 2-DOF and 3-DOF systems in Figures 7.11 and 7.12. When judged visually, the peaks in the PSDs are well separated, indicative of a good decomposition.

7.4 Conclusions

In this chapter, two methods for generating nonlinear normal modes under the statistically-independent framework have been presented and compared. The quality of the modal decomposition has been evaluated qualitatively in terms of the separation of the resonance structures in the decomposed PSDs.

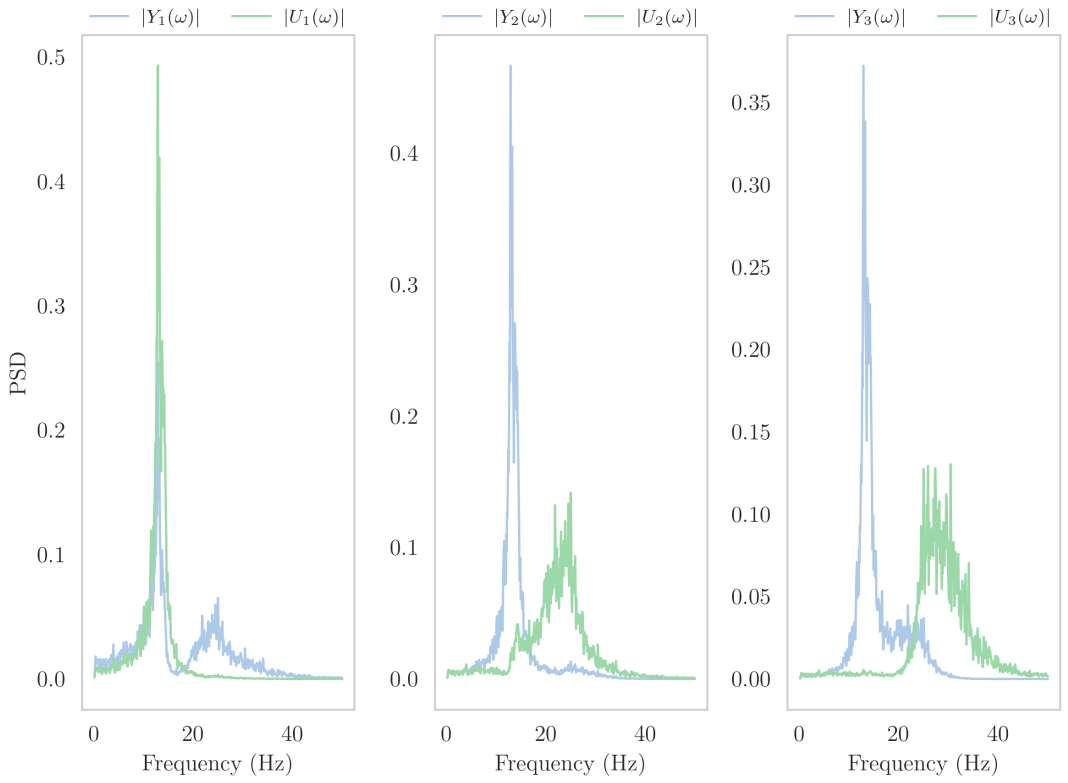


FIGURE 7.12: 3-DOf system: Physical vs. modal PSDs - cycle-GAN decomposition.

Judged visually, the modal decompositions constructed by the heuristic decomposition method exhibit mixed performance. Although there is some evidence that the inclusion of the distance correlation improves the performance, it is evident that the correlation metric and column-wise orthogonality of the linear part of the multinomial transformation are insufficient inductive biases to guarantee visually distinct peaks in every run.

In comparison, the cycle-GAN approach has performed significantly better. A likely explanation for this observation is that the inclusion of the additional inductive biases of conformality via the orthogonality assembly and independence through the inner product metric used to select the best overall transformation. In addition to the increased performance, the cycle-GAN approach also offers several desirable properties including a non-parametric formulation, concurrent specification of the inverse transformation and optimisation by gradient descent. Given the considerable advantages of the cycle-GAN approach, only the NNMs generated by this method will be considered in the analysis that follows

Although the 'by-eye' assessment of the modal decompositions is convenient, there remains an urgent need to quantitively evaluate the NNMs in terms of the criteria of Chapter 2.

1. *Independence*: The ability of the decomposition to render the dynamics into an independent modal basis, preferably SISO.

2. *Decomposition*: The extent to which the decomposed modal dynamics represent a physically meaningful (by some measure) basis for understanding the structural dynamics.

3. *Superposition*: The extent to which the original dynamics can be recovered from the decomposition.

It is certainly the case that the visual assessment of the modes in this chapter gives an indication of both independence and decomposition, but concrete conclusions cannot be reached before the completion of more robust analysis in the following chapters.

Chapter 8

Higher-order FRFs for analysis of nonlinear modal dynamics

The previous three chapters have considered in detail, aspects of the statistically-independent framework for nonlinear modal analysis. In the previous chapter, the nonlinear modal transformations were established in a data-driven setting and it was qualitatively shown that the modes had desirable properties. However, questions remain as to the nature of the decomposed modal dynamics.

In this chapter, a powerful tool for understanding and visualising nonlinear dynamics is explored, *higher-order frequency response functions* (HFRFs). These objects have several advantages for understanding the dynamics of nonlinear normal modes. HFRFs arise from a frequency-domain representation of the Volterra series [41]. Each HFRF encodes the dynamic properties of a system independently at a given order, in much the same way as Taylor series coefficients of static functions. In this way, the linear and nonlinear components of the dynamics can be evaluated separately. In general, there are an infinite number of HFRFs for a given nonlinear system (since the Volterra series is an infinite series). However, meaningful assessment of the salient aspects of the dynamics (underlying linear component, presence of odd and even nonlinearities) can be obtained by considering only the first few HFRFs.

As well as the present requirement for interpreting the dynamics of the nonlinear modes generated under the statistically-independent framework, HFRFs have considerable utility for the validation of black-box techniques in nonlinear system identification (NLSI). One area of particular interest is the interpretability of models that are constructed in the black-box setting. To this end, a number of approaches have been proposed in the NLSI literature [167]. A common approach to gleaning physical insight from black-box models is to use the Volterra series [2].

In this chapter, closed form expressions for the HFRFs of kernel-NARX models are developed by the application of the method of harmonic probing [168]. These expressions are furthermore validated against a numerical case study where the HFRFs are extracted from a Gaussian process NARX model (for a variety of common kernel choices), trained on data from a nonlinear dynamic system with polynomial stiffness elements.

8.1 The Volterra series

The Volterra series [41], can be thought of as a generalisation of the Taylor series to the case of functionals, that allows for 'memory' effects to be captured. Whereas the Taylor series encodes information about a function at a given position in the input space,

$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!}(x-a)^n + \dots \tag{8.1}$$

The Volterra series is able to capture the effect of previous values of the input. In continuous time, the series is given by,

$$y(t) = y_1(t) + y_2(t) + y_3(t) + \cdots + y_n(t) \tag{8.2}$$

where,

$$y_1(t) = \int_{-\infty}^{+\infty} h_1(\tau)x(t-\tau)\mathrm{d}\tau \tag{8.3}$$

$$y_2(t) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} h_2(\tau_1,\tau_2)\, x(t-\tau_1)\,\mathrm{d}\tau_1\mathrm{d}\tau_2 x(t-\tau_2) \tag{8.4}$$

$$y_n(t) = \int_{-\infty}^{+\infty}\cdots\int_{-\infty}^{+\infty} h_n(\tau_1,\cdots,\tau_n)\, x(t-\tau_1)\cdots x(t-\tau_n)\,\mathrm{d}\tau_1\cdots\mathrm{d}\tau_n \tag{8.5}$$

In the above, the h_i are known as the *Volterra kernels*. These kernels are akin to the Taylor coefficients of a static function. The Volterra kernels encode the nonlinearity of the functional at various orders. Of particular interest in an NLSI context is h_1, which encodes the linear impulse response of the dynamics. Just as in the linear case with Duhamel's integral, the Volterra series admits a frequency-domain representation. The expression for arbitrary functionals is beyond the scope of the results in this chapter, but

for the single-input single-output (SISO) dynamic system subject to harmonic excitation, taking the Fourier transform of the series yields [2],

$$y(t) = H_1(\omega)\mathrm{e}^{i\omega t} + H_2(\omega,\omega)\mathrm{e}^{i2\omega t} + H_3(\omega,\omega,\omega)\mathrm{e}^{\mathrm{i}3\omega t} + \cdots \tag{8.6}$$

In the above, the Volterra kernels in the time domain become the higher-order frequency response functions H_i. These HFRFs have a number of desirable properties in terms of interpreting the dynamics of the nonlinear system. For example, just as h_1 encodes the linear impulse response of the system, H_1 encodes the FRF of the linear dynamics. Clearly, this is a useful tool for model interpretation in an NLSI context. With access to the HFRFs of a model, an engineer is able to evaluate common modelling pitfalls, such as bias in the linear dynamics[1] and spurious higher-order behaviour, for example, the presence of even nonlinearities in systems that are known to only contain odd nonlinearities.

8.2 Harmonic probing

HFRFs can be derived analytically from the equations of motion of a dynamic system using the harmonic-probing algorithm [168, 169]. For simple dynamics, this is straightforward, but quickly grows in complexity with model order. Given the utility of the HFRFs in interpreting NLSI models, it is unsurprising that a fair amount of attention has been generated in deriving closed-form expressions for the HFRFs for various model types. For example, expressions are available for a simple neural-network nonlinear-autoregressive model with exogeneous inputs (NARX) in [170], and a Gaussian-process NARX (GP-NARX) model with a squared exponential kernel in [49, 171] (and extended to the multi-lengthscale case in [172]).

In order to demonstrate the method of harmonic probing when the equations of motion are known, the HFRFs of a nonlinear quadratic-cubic Duffing equation are derived here. The derivations established here, largely follow the work of Gifford et al. [168]. The system under investigation obeys the equation of motion,

$$m\ddot{y} + c\dot{y} + ky + k_2y^2 + k_3y^3 = x(t) \tag{8.7}$$

In order to extract $H_1(\omega)$, a harmonic probing input is assumed of the form,

[1] In the case that the true underlying linear stiffness is available.

$$x(t) = e^{i\omega t} \tag{8.8}$$

From the Volterra series, the appropriate probing output is given by,

$$y(t) = e^{i\omega t} H_1(\omega) + 2e^{i(\omega+\omega)t} H_2(\omega, \omega) + \cdots \tag{8.9}$$

It is important to note that all higher-order terms (containing H_2, H_3 etc.) can be neglected here. This is because of the orthogonality property of the harmonic terms. Notice that once the harmonic probing input and output are substituted into the equation of motion, the only term in (8.9) that can generate coefficients of $e^{i\omega t}$ is the first term, $e^{i\omega t} H_1(\omega)$. This truncation is important, because by equating coefficients of $e^{i\omega t}$, one can retrieve an equation that can be solved for H_1. Thus, for compactness the harmonic probing response is written as,

$$y(t) = e^{i\omega t} H_1(\omega) \tag{8.10}$$

Substituting this and the probing input into the equation of motion above, one has,

$$\begin{aligned} k e^{i\omega t} H_1(\omega) + c \frac{\mathrm{d}}{\mathrm{d}t} \left[e^{i\omega t} H_1(\omega) \right] + m \frac{\mathrm{d}^2}{\mathrm{d}t^2} \left[e^{i\omega t} H_1(\omega) \right] \\ + k_2 \left(e^{i\omega t} H_1(\omega) \right)^2 + k_3 \left(e^{i\omega t} H_1(\omega) \right)^3 = e^{i\omega t} \end{aligned}$$

$$\begin{aligned} k \left[e^{i\omega t} H_1(\omega) \right] + ci\omega e^{i\omega t} H_1(\omega) + m(i\omega)^2 e^{i\omega t} H_1(\omega) \\ + k_2 e^{2i\omega t} H_1(\omega)^2 + k_3 e^{3i\omega t} H_1(\omega)^3 = e^{i\omega t} \end{aligned} \tag{8.11}$$

Extracting only the coefficients of $e^{i\omega t}$,

$$H_1(\omega) \left[k + (i\omega)c + (i\omega)m \right] = 1 \tag{8.12}$$

is in an expression that can be solved trivially for H_1.

$$H_1(\omega) = \left[-\omega^2 m + i\omega c + k \right]^{-1} \tag{8.13}$$

It is interesting to note that H_1 encodes the *underlying* linear dynamics of the system, or the FRF of the linear system if all nonlinear elements were removed. This is a valuable property of the HFRFs, as it allows the engineer to assess the extent to which a black-box model may have biased the underlying linear system. It is argued here that this represents a more stringent method of model validation than assessment of $\mathcal{L}^2$ error metrics alone.

Continuing the method of harmonic probing for H_2, the appropriate input is now,

$$x(t) = e^{i\omega_1 t} + e^{i\omega_2 t} \tag{8.14}$$

for which the response from the Volterra series is,

$$y(t) = e^{i\omega_1 t}H_1(\omega_1) + e^{i\omega_2 t}H_1(\omega_2) + e^{i(\omega_1+\omega_2)t}H_2(\omega_1,\omega_2) + \cdots \tag{8.15}$$

where once again, the higher-order terms can be neglected on the basis that they cannot produce coefficients of $e^{i(\omega_1+\omega_2)t}$. Substituting these as before into the equations of motion and extracting all coefficients of $e^{i(\omega_1+\omega_2)t}$, yields an expression which may be solved for $H_2(\omega_1,\omega_2)$; the result is,

$$H_2(\omega_1,\omega_2) = -k_2 H_1(\omega_1)H_1(\omega_2)H_1(\omega_1+\omega_2) \tag{8.16}$$

Notice that this expression contains the quadratic stiffness as a scaling factor and so has useful interpretability in terms of the extent to which the quadratic stiffness has been recovered by the model. The process can be repeated in a similar (read: tedious) fashion for H_3, by considering three independent harmonics in the probing input. The result is,

$$\begin{aligned} H_3(\omega_1,\omega_2,\omega_3) = -\frac{1}{3}H_1(\omega_1+\omega_2+\omega_3)\big[&3k_3 H_1(\omega_1)H_1(\omega_2)H_1(\omega_3) \\ &+ 2k_2(H_1(\omega_1)H_2(\omega_2,\omega_3) + H_1(\omega_2)H_2(\omega_1,\omega_3) + H_1(\omega_3)H_2(\omega_1,\omega_2))\big] \end{aligned} \tag{8.17}$$

Once again, notice that the expression for H_3 can be constructed entirely from H_1 and the nonlinear stiffness coefficients, affording important insight into the characteristics of the nonlinear system. Note also the effect of taking $k_2 = 0$, the entire expression is parameterised by k_3 and H_1.

8.2.1 The kernel-NARX model

The principal contributions of this chapter are general expressions for the HFRFs of an entire class of stationary kernel-NARX models, parameterised entirely in terms of the training data and the choice of kernel $\kappa(d)$, where d is some metric that describes the distance between two points on the input space. An important member of the stationary-kernel NARX class is the GP-NARX model [21, 48]. These models have a number of advantages including a non-parametric form, a small number of hyperparameters and uncertainty quantification.

The models in this class have the explicit predictive form given by,

$$y_t = \sum_{j}^{N} \alpha_j \kappa \left(\sum_{k=1}^{n_y} ||y_{t-k} - v_{jk}|| + \sum_{m=0}^{n_x} ||x_{t-m} - u_{jm}|| \right) \tag{8.18}$$

where α_i are the learned model weights, x_i, y_i are the unseen lagged inputs and outputs to the system and u, v are the training data; n_x and n_y are the maximum numbers of lags in the inputs and outputs. The above expression also permits a vectorised form,

$$y_t = \alpha^{\mathsf{T}} \kappa \left(||\mathbf{h} - H|| \right) \tag{8.19}$$

where $\mathbf{h}$ are the unseen lagged inputs and outputs to the model,

$$\mathbf{h} = \begin{bmatrix} \mathbf{y} \\ \mathbf{x} \end{bmatrix} = \begin{bmatrix} y_{t-1}, & y_{t-2}, & \cdots, & y_{t-n_y} \\ x_t, & x_{t-1}, & \cdots, & x_{t-n_x-1} \end{bmatrix} \tag{8.20}$$

H is the Hankel matrix of training inputs and outputs and $||x - x'||$ is a vector norm,

$$H = \begin{bmatrix} V \\ U \end{bmatrix} = \begin{bmatrix} y_{t-1} & y_{t-2} & \cdots & y_{t-n_y} & x_t & x_{t-1} & x_{t-2} & \cdots & x_{t-n_x+1} \\ y_{t-2} & y_{t-3} & \cdots & y_{t-n_y-1} & x_{t-1} & x_{t-2} & x_{t-3} & \cdots & x_{t-n_x} \\ y_{t-3} & y_{t-4} & \cdots & y_{t-n_y-2} & x_{t-2} & x_{t-3} & x_{t-4} & \cdots & x_{t-n_x-1} \\ \vdots & & & & \vdots & & & & \\ y_{t-N+p} & & & & x_{t-N+p-1} & & & & \end{bmatrix}^{\mathsf{T}} \tag{8.21}$$

The derivations presented here make the assumption that $||\cdot||$ is the $\mathcal{L}^2$ norm. Rewriting the above expression, one has

$$y_t = \alpha^{\mathsf{T}} \kappa \left(\sqrt{\mathbf{d}^2} \right) \tag{8.22}$$

where,

$$\mathbf{d}^2 = \sum_j^N \left(\begin{bmatrix} \mathbf{y} \\ \mathbf{x} \end{bmatrix} - \begin{bmatrix} \mathbf{v}_j \\ \mathbf{u}_j \end{bmatrix} \right)^2 = (\mathbf{y} - U)^{\mathsf{T}}(\mathbf{y} - U) + (\mathbf{x} - V)^{\mathsf{T}}(\mathbf{x} - V) \tag{8.23}$$

Now let,

$$\mathbf{q} = \mathbf{y}^{\mathsf{T}}\mathbf{y} - 2\mathbf{y}V + \mathbf{x}^{\mathsf{T}}\mathbf{x} - 2\mathbf{x}U \tag{8.24}$$

and,

$$\gamma^2 = -\sum_j^N \mathbf{v}_j^2 - \sum_k^N \mathbf{u}_k^2 \tag{8.25}$$

Equation (8.23) can then be rewritten,

$$\mathbf{d}^2 = \mathbf{q} + \gamma^2 \tag{8.26}$$

The trick here is to now expand the kernel function κ as a Taylor series around $\mathbf{d}^2 = \gamma^2$, resulting in,

$$\kappa \approx A_0 + A_1(\mathbf{d}^2 - \gamma^2) + A_2(\mathbf{d}^2 - \gamma^2)^2 + \dots \tag{8.27}$$

Or equivalently,

$$\kappa \approx A_0 + A_1\mathbf{q} + A_2\mathbf{q}^2 + \dots \tag{8.28}$$

Where the A_i are the Taylor series coefficients, computed in the usual manner, that now depend on the norm of training data, γ^2. Thus, equation (8.22) becomes,

$$y_t = \alpha^{\mathsf{T}} \left(A_0 + A_1\mathbf{q} + A_2\mathbf{q}^2 + \dots \right) \tag{8.29}$$

which is essentially a discrete-time polynomial NARX model in y with input x. The method of harmonic probing can now be applied, following the approach of [173]. However, it is useful to first establish some notation to ensure the compactness of the forthcoming expressions. First, let,

$$E_j = e^{i\omega_j t} \tag{8.30}$$

be a harmonic signal consisting of a single frequency. Now define,

$$\mathbf{\Delta}_y(\omega) = e^{-ik\omega\Delta_t} \quad k \in 1, \dots, n_y \tag{8.31}$$

$$\mathbf{\Delta}_x(\omega) = e^{-im\omega\Delta_t} \quad m \in 1, \dots, n_x \tag{8.32}$$

as vectors of lag operators that apply the discrete-time lag operation, where $\Delta_t = f_s^{-1}$ is the sampling period of the discrete-time method.

The appropriate probing signals for the discrete time model are now,

$$\mathbf{x} = E_1\mathbf{\Delta}_x(\omega) \tag{8.33}$$

$$\mathbf{y} = E_1\mathbf{\Delta}_y(\omega)H_1(\omega) \tag{8.34}$$

As before, H_1 can be evaluated by substituting the above into equation (8.29) and pulling out the coefficient of E_1. One might be tempted to think that because equation (8.29) is an infinite series, there will be an infinite number of terms that produce coefficients of E_1. However, this is not the case. In fact, the only term that can generate the required coefficients is $A_1\mathbf{q}$. The algebra is not included explicitly here for reasons of space but the result can be found as,

$$H_1(\omega) = \frac{2A_1\mathbf{\Delta}_x(\omega)U\alpha}{1 + 2A_1\mathbf{\Delta}_y(\omega)V\alpha} \tag{8.35}$$

Continuing in this fashion for H_2 and H_3 yields (arduously), to the following expressions,

$$
\begin{aligned}
H_2(\omega_1,\omega_2) = \left[1 + 2A_1\boldsymbol{\Delta}_y(\omega_1+\omega_2)V\alpha\right]^{-1} \times \Big[\\
+H_1(\omega_1)H_1(\omega_2)\left[4A_2(\boldsymbol{\Delta}_y(\omega_1)V)\circ(\boldsymbol{\Delta}_y(\omega_2)V) + A_1\boldsymbol{\Delta}_y(\omega_1)^{\mathsf{T}}\boldsymbol{\Delta}_y(\omega_2)\right]\alpha \\
+H_1(\omega_1)\left[4A_2(\boldsymbol{\Delta}_y(\omega_1)V)\circ(\boldsymbol{\Delta}_x(\omega_2)U)\right]\alpha \\
+H_1(\omega_2)\left[4A_2(\boldsymbol{\Delta}_y(\omega_2)V)\circ(\boldsymbol{\Delta}_x(\omega_1)U)\right]\alpha \\
+\left[4A_2(\boldsymbol{\Delta}_x(\omega_1)U)\circ(\boldsymbol{\Delta}_x(\omega_2)U) + A_1\boldsymbol{\Delta}_x(\omega_1)^{\mathsf{T}}\boldsymbol{\Delta}_x(\omega_2)\right]\alpha\Big]
\end{aligned}
\tag{8.36}
$$

$$
\begin{aligned}
H_3(\omega_1,\omega_2,\omega_3) = &\Big[1 + 2A_1\boldsymbol{\Delta}_y(\omega_1+\omega_2+\omega_3)V\alpha\Big]^{-1} \times \Big[\\
&+H_1(\omega_1)H_1(\omega_2)H_1(\omega_3)\times \\
&\quad\Big[-12A_3(\boldsymbol{\Delta}_y(\omega_1)V)\circ(\boldsymbol{\Delta}_y(\omega_2)V)\circ(\boldsymbol{\Delta}_y(\omega_3)V) - 6A_2\boldsymbol{\Delta}_y(\omega_1)^{\mathsf{T}}\boldsymbol{\Delta}_y(\omega_2)\circ\boldsymbol{\Delta}_y(\omega_3)V\Big]\,\alpha \\
&+H_1(\omega_1)H_1(\omega_2) \\
&\quad\Big[-12A_3(\boldsymbol{\Delta}_y(\omega_1)V)\circ(\boldsymbol{\Delta}_y(\omega_2)V)\circ(\boldsymbol{\Delta}_x(\omega_3)U) - 2A_2\boldsymbol{\Delta}_y(\omega_1)^{\mathsf{T}}\boldsymbol{\Delta}_y(\omega_2)\circ\boldsymbol{\Delta}_x(\omega_3)U\Big]\,\alpha \\
&+H_1(\omega_2)H_1(\omega_3) \\
&\quad\Big[-12A_3(\boldsymbol{\Delta}_y(\omega_2)V)\circ(\boldsymbol{\Delta}_y(\omega_3)V)\circ(\boldsymbol{\Delta}_x(\omega_1)U) - 2A_2\boldsymbol{\Delta}_y(\omega_2)^{\mathsf{T}}\boldsymbol{\Delta}_y(\omega_3)\circ\boldsymbol{\Delta}_x(\omega_1)U\Big]\,\alpha \\
&+H_1(\omega_1)H_1(\omega_3) \\
&\quad\Big[-12A_3(\boldsymbol{\Delta}_y(\omega_1)V)\circ(\boldsymbol{\Delta}_y(\omega_3)V)\circ(\boldsymbol{\Delta}_x(\omega_2)U) - 2A_2\boldsymbol{\Delta}_y(\omega_1)^{\mathsf{T}}\boldsymbol{\Delta}_y(\omega_3)\circ\boldsymbol{\Delta}_x(\omega_2)U\Big]\,\alpha \\
&+H_1(\omega_1)H_1(\omega_2,\omega_3) \\
&\quad[4A_2(\boldsymbol{\Delta}_y(\omega_1)V)\circ(\boldsymbol{\Delta}_y(\omega_2+\omega_3)V)]\,\alpha \\
&+H_1(\omega_2)H_1(\omega_1,\omega_3) \\
&\quad[4A_2(\boldsymbol{\Delta}_y(\omega_2)V)\circ(\boldsymbol{\Delta}_y(\omega_1+\omega_3)V)]\,\alpha \\
&+H_1(\omega_3)H_1(\omega_2,\omega_3) \\
&\quad[4A_2(\boldsymbol{\Delta}_y(\omega_3)V)\circ(\boldsymbol{\Delta}_y(\omega_1+\omega_2)V)]\,\alpha \\
&+H_1(\omega_1,\omega_2) \\
&\quad[4A_2(\boldsymbol{\Delta}_y(\omega_1+\omega_2)V)\circ(\boldsymbol{\Delta}_x(\omega_1)U)]\,\alpha \\
&+H_1(\omega_2,\omega_3) \\
&\quad[4A_2(\boldsymbol{\Delta}_y(\omega_2+\omega_3)V)\circ(\boldsymbol{\Delta}_x(\omega_2)U)]\,\alpha \\
&+H_1(\omega_1,\omega_3) \\
&\quad[4A_2(\boldsymbol{\Delta}_y(\omega_1+\omega_3)V)\circ(\boldsymbol{\Delta}_x(\omega_3)U)]\,\alpha \\
&+H_1(\omega_1) \\
&\quad\Big[-12A_3(\boldsymbol{\Delta}_y(\omega_1)V)\circ(\boldsymbol{\Delta}_x(\omega_2)U)\circ(\boldsymbol{\Delta}_x(\omega_3)U) - 2A_2\boldsymbol{\Delta}_x(\omega_2)^{\mathsf{T}}\boldsymbol{\Delta}_x(\omega_3)\circ\boldsymbol{\Delta}_y(\omega_1)V\Big]\,\alpha \\
&+H_1(\omega_2) \\
&\quad\Big[-12A_3(\boldsymbol{\Delta}_y(\omega_2)V)\circ(\boldsymbol{\Delta}_x(\omega_3)U)\circ(\boldsymbol{\Delta}_x(\omega_1)U) - 2A_2\boldsymbol{\Delta}_x(\omega_1)^{\mathsf{T}}\boldsymbol{\Delta}_x(\omega_3)\circ\boldsymbol{\Delta}_y(\omega_2)V\Big]\,\alpha \\
&+H_1(\omega_3) \\
&\quad\Big[-12A_3(\boldsymbol{\Delta}_y(\omega_1)V)\circ(\boldsymbol{\Delta}_x(\omega_3)U)\circ(\boldsymbol{\Delta}_x(\omega_2)U) - 2A_2\boldsymbol{\Delta}_x(\omega_2)^{\mathsf{T}}\boldsymbol{\Delta}_x(\omega_3)\circ\boldsymbol{\Delta}_y(\omega_3)V\Big]\,\alpha \\
&+\Big[-12A_3(\boldsymbol{\Delta}_x(\omega_1)U)\circ(\boldsymbol{\Delta}_x(\omega_2)U)\circ(\boldsymbol{\Delta}_x(\omega_3)U) - 6A_2\boldsymbol{\Delta}_x(\omega_1)^{\mathsf{T}}\boldsymbol{\Delta}_x(\omega_2)\circ\boldsymbol{\Delta}_x(\omega_3)U\Big]\,\alpha\Big]
\end{aligned}
\tag{8.37}
$$

where $\circ$ denotes an elementwise product between vectors. It is noteworthy that these expressions are far more complicated than those of the continuous-time systems considered earlier in this chapter. This dramatic increase in complexity can be attributed to the far more flexible nature of these expressions. They are able to encode a far more rich set of functional that the polynomial stiffness models considered above.

Extension to the GP-NARX case

Note that the above equations can be extended to include the GP-NARX models considered in this work by including the signal variance parameter within the kernel weights,

$$\alpha' = \alpha\sigma_f^2 \tag{8.38}$$

Lengthscales, including ARD can be also be included by elementwise scaling of the data matrices and lag operator vectors,

$$U' = \frac{U}{\ell} \tag{8.39}$$

$$V' = \frac{V}{\ell} \tag{8.40}$$

$$\mathbf{\Delta}_x(\omega)' = \frac{\mathbf{\Delta}_x(\omega)}{\ell} \tag{8.41}$$

$$\mathbf{\Delta}_y(\omega)' = \frac{\mathbf{\Delta}_y(\omega)}{\ell} \tag{8.42}$$

Note also, that the noise variance parameter is included by construction from the definition of the kernel weights in the GP-NARX model.

$$\alpha = \left[\kappa(H,H) + \sigma_n^2 I\right]^{-1}\mathbf{y} \tag{8.43}$$

Substituting these values into the equations above returns the HFRFs of the GP-NARX model. Additional hyperparameters can be explicitly included when computing the Taylor series of the kernel.

TABLE 8.1: Parameters pertaining to the simulation of the nonlinear system.

Parameter	Symbol	Value
Mass	m	1
Viscous damping	c	20
Linear stiffness	k	10^4
Quadratic stiffness	k_2	10^7
Cubic stiffness	k_3	5×10^9
Input excitation level	σ_x^2	2
Low-pass cut-off frequency (Hz)	f_c	50
Filter order		4
Sampling frequency (Hz)	f_s	1024
RMS noise level		0.1%
Dataset size	N	10^3

8.3 Case study

To illustrate the effectiveness of the HFRFs as a tool for verification in NLSI, a case study example is developed here. For the present study, the system under investigation will be a single degree-of-freedom (SDOF) nonlinear quadratic-cubic Duffing equation. The equation of motion for this dynamic system is given by,

$$m\ddot{y} + c\dot{y} + ky + k_2y^2 + k_3y^3 = x(t) \tag{8.44}$$

For the simulation data, the equations of motion are integrated forward in time by a fixed-step fourth-order Runge-Kutta method with a sampling frequency of 1024 Hz. For excitation, a white Gaussian signal low-pass filtered onto the interval $[0, 50]$ Hz is applied with a mean of zero and a standard deviation of 2N. All parameters pertaining to the benchmark system and the simulation are collected for the convenience of the reader in Table 8.1.

Overall, 10^5 points are simulated. After simulation, the first 2×10^4 points are discarded in order to remove the effects of transients from the initial conditions or integrator. In order to simplify the training of nonlinear models, the remaining data were then standardised to zero-mean and unit variance. To stabilise the numerical methods, a small amount of process noise was added at 0.1% of the root-mean-squared (RMS) signal level. Finally, a training, validation and testing set are extracted from the scaled data, each comprised of 10^3 points.

Although the expressions for HFRFs developed in this work are valid for the class of stationary kernel-NARX models generally, the model chosen for this case-study is the GP-NARX model.

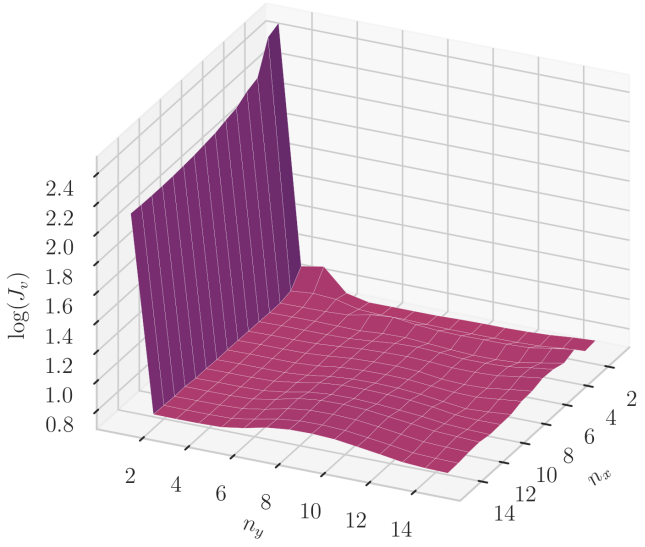


FIGURE 8.1: Lag structure cost surface on unseen validation data for the ARX model.

An open problem in the GP-NARX literature (and in NARX models more generally) is the efficient and robust selection of the lag structure (also know as lag-structure optimisation (LSO)). Many approaches based on the Akaike information criteria [174] or a Bayesian extension [175] have been proposed, but these all suffer from expensive computational cost, as a new model must be trained for each lag structure under consideration (a problem that scales combinatorically with input and output dimension). In the present study, LSO is simply conducted by a naïve grid search on the maximum lags n_x and n_y using a surrogate ARX model.

Maximum lags up to a value of 15 are considered and the resultant simulation NMSE of the ARX model on the unseen validation data is presented in Figure 8.1. As can be seen from the figure, the performance of the ARX model appears to be largely insensitive to the lag structure above some (low) threshold number of lags in the output. With this in mind, the lag structure used for all GP-models is therefore chosen (somewhat arbitrarily) as $n_x = 3$ and $n_y = 5$.

With the lag structure established, the GP-NARX models can be trained. An important choice when working with any kernel-NARX method is the choice of kernel function. Much has been written on the subject of kernel selection for GPs, and the interested reader is directed to reference [57] for additional detail. The focus of the current study

TABLE 8.2: Covariance kernel functions used in this study.

Kernel	Function	Hyperparameters
RBF	$\sigma_f^2 e^{-\frac{1}{2}d^2}$	$\{\sigma_n^2, \ell, \sigma_f^2\}$
Rational Quadratic	$\sigma_f^2(1+\frac{1}{\alpha}d^2)^{-\alpha}$	$\{\sigma_n^2, \ell, \sigma_f^2, \alpha = 2\}$
Matérn($\frac{1}{2}$)	$\sigma_f^2 e^{-d}$	$\{\sigma_n^2, \ell, \sigma_f^2\}$
Matérn($\frac{3}{2}$)	$\sigma_f^2(1+\sqrt{3}d)e^{-\sqrt{3}d}$	$\{\sigma_n^2, \ell, \sigma_f^2\}$
Matérn($\frac{5}{2}$)	$\sigma_f^2(1+\sqrt{5}d+\frac{5}{3}d^2)e^{-\sqrt{5}d}$	$\{\sigma_n^2, \ell, \sigma_f^2\}$

TABLE 8.3: Taylor series expansions of the kernels used in this study.

Kernel	Taylor series about γ^2
RBF	$e^{-\frac{1}{2}\gamma^2}(1-\frac{\mathbf{q}}{2}+\frac{\mathbf{q}^2}{8}-\frac{\mathbf{q}^3}{48}+\dots)$
Rational Quadratic	$\frac{4}{(\gamma^2+2)^2}-\frac{8\mathbf{q}}{(\gamma^2+2)^3}+\frac{12\mathbf{q}^2}{(\gamma^2+2)^4}-\frac{16\mathbf{q}^3}{(\gamma^2+2)^5}+\dots$
Matérn($\frac{1}{2}$)	$e^{-\gamma}(1-\frac{\mathbf{q}}{2\gamma}+\frac{(1+\gamma)\mathbf{q}^2}{8\gamma^3}-\frac{(3+3\gamma+\gamma^3)\mathbf{q}^3}{48\gamma^5}+\dots)$
Matérn($\frac{3}{2}$)	$e^{-\sqrt{3}\gamma}((\sqrt{3}\gamma+1)-\frac{3\mathbf{q}}{2}+\frac{3\sqrt{3}\mathbf{q}^2}{8\gamma}-\frac{(\sqrt{3}+3\gamma)\mathbf{q}^3}{16\gamma^5}+\dots)$
Matérn($\frac{5}{2}$)	$e^{-\sqrt{5}\gamma}((\frac{1}{3}(5\gamma_2+3\sqrt{5}\gamma+3))-\frac{5(\sqrt{5}\gamma+1)\mathbf{q}}{6}+\frac{25\mathbf{q}^2}{24\gamma}-\frac{25\sqrt{5}\mathbf{q}^3}{144\gamma}+\dots)$

is to extract HFRFs from a number of stationary kernels. To this end, a number of common kernels have been selected. Details of the kernels considered in this investigation are given in Table 8.2. For compactness, the kernel functions are given in terms of the stationary scaled $\mathcal{L}_2$ distance d,

$$d = \sqrt{\left(\frac{x-x'}{\ell}\right)^2} \tag{8.45}$$

where ℓ is the *lengthscale* hyperparameter. In order to simplify the optimisation, the models trained in the present study do not have an ARD lengthscale structure.

For each kernel in 8.2, the GP still has a small number of hyperparameters that must be optimised. Optimisation is conducted by the minimisation of the negative log marginal likelihood,

$$\log P(\theta|H) = -\frac{1}{2}\mathbf{y}^{\mathsf{T}}\left[\kappa(H,H)+\sigma_n^2 I\right]\mathbf{y} - \frac{1}{2}\log\left|\kappa(H,H)+\sigma_n^2 I\right| \tag{8.46}$$

where θ is the vector of hyperparameters. Since the derivative of the above is available in closed form, the optimisation of θ is conducted by gradient descent. For each kernel, the optimisation procedure is repeated for 25 random initialisations, each proceeding until a convergence criteria was met. The lowest overall objective score is then selected. With the models trained, the quality of the model fit is assessed by a normalised mean-square error metric given by,

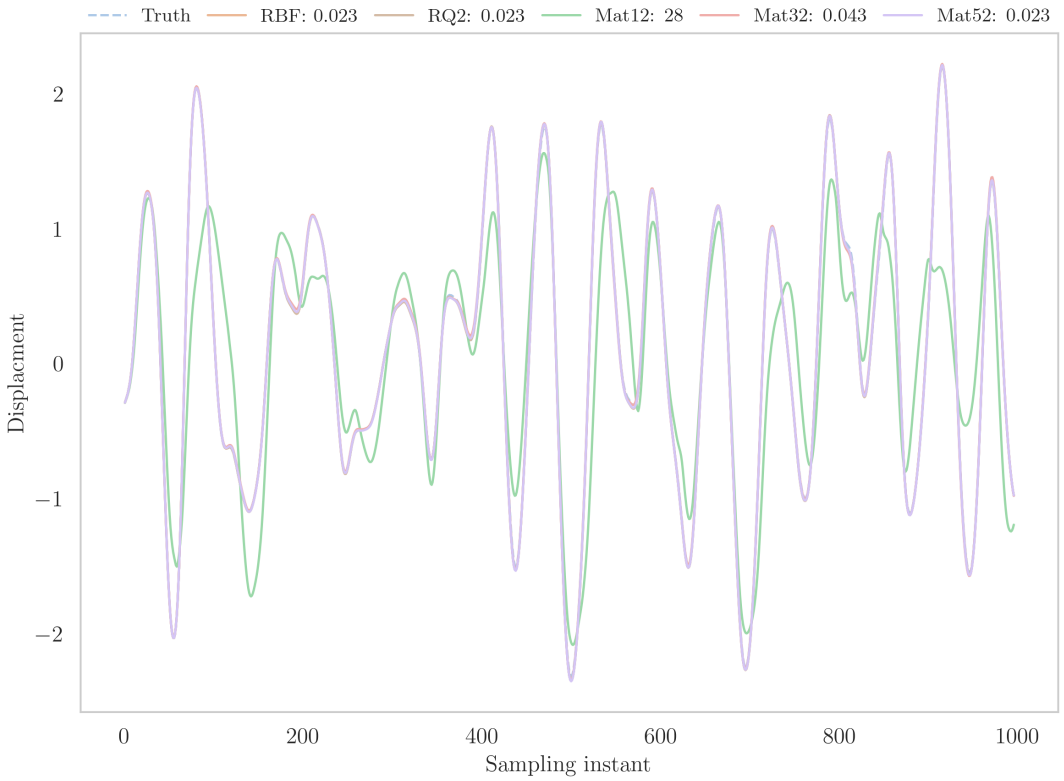


FIGURE 8.2: Simulation performance of the GP-NARX models on the unseen testing data.

$$J_{\text{NMSE}} = \frac{100}{N\sigma_y}\sum_{j}^{N}(y_j - \hat{y}_j)^2 \tag{8.47}$$

where y_j are unseen testing data and $\hat{y}_j$ are the simulations from the model. The NMSE performances of each kernel are collected in Table 8.4 and the predicted time series are plotted in Figure 8.2.

The quality of the model fit is excellent for every kernel except the Matérn($\frac{1}{2}$). This is not unexpected. The Matérn kernel classes is known to produce very non-smooth models [57] for low values of the free parameter. Given that the true dynamics are governed by a very smooth differential equation with a smoothed input (filtered Gaussian noise) it is unsurprising that the Matérn($\frac{1}{2}$) kernel produces a poor fit.

With the expressions for the HFRF developed, these are now used to assess the quality of the GP-NARX models that where fitted in the previous section. The HFRFs for the stationary kernel NARX models require Taylor expansions of the kernels about the training data. These were computed using the computer algebra software MAPLE 2018 [68]. The results are collected in Table 8.3.

TABLE 8.4: NMSE simulation performance of the GP-NARX models on the unseen testing data.

Kernel	Simulation NMSE
RBF	0.0229 %
Rational Quadratic	0.0233 %
Matérn($\frac{1}{2}$)	27.8 %
Matérn($\frac{3}{2}$)	0.0435 %
Matérn($\frac{5}{2}$)	0.0234 %

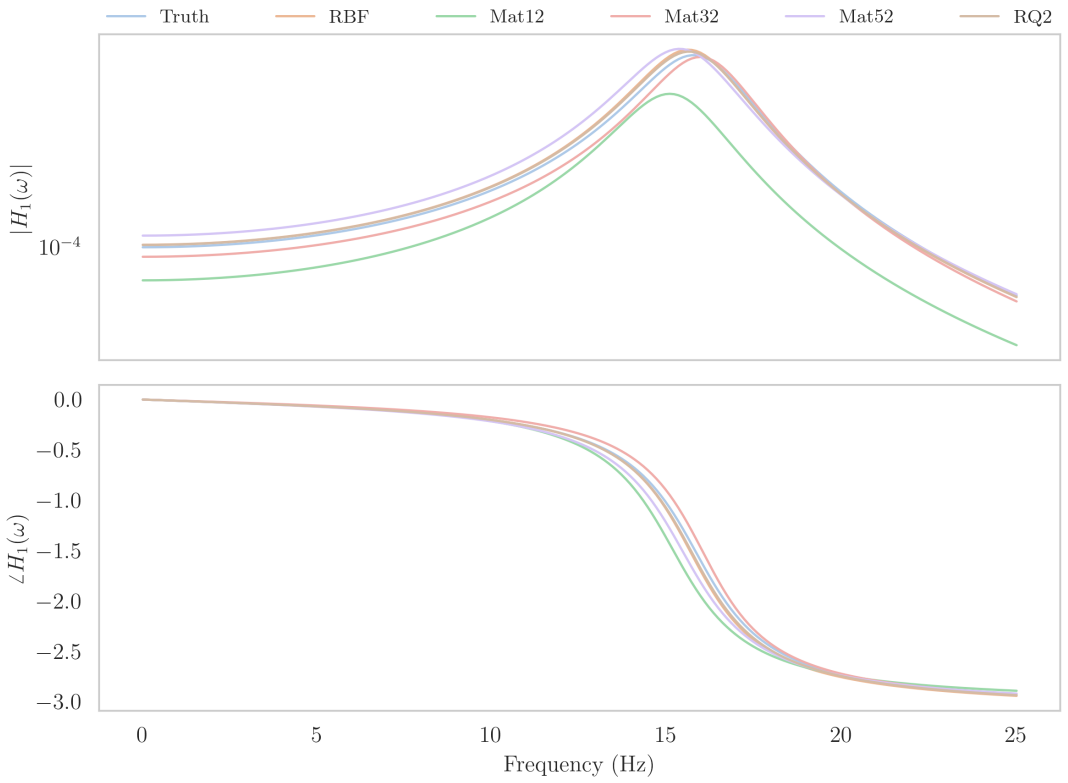


FIGURE 8.3: $H_1(\omega)$ calculated from the GP-NARX models compared to theoretical values.

Finally, the HFRFs can be computed and compared to the theoretical values. Figure 8.3 depicts the estimates of H_1 against the theoretical values. The results show strong agreement in almost every case indicating that the models have learnt an unbiased representation of the underlying linear dynamics. As before, the GP-NARX model with the Matérn($\frac{1}{2}$) kernel has performed poorly. In this case, the linear dynamics are clearly biased to compensate for the effect of the smooth nonlinearities.

Figure 8.6 depicts the main diagonal of the calculated and theoretical H_2; here, the agreement is still strong for the smooth kernels but each of the Matérn kernels, show some degradation in capturing the quadratic stiffness nonlinearity. This can also be seen in the H_2 surfaces plotted in Figures 8.4 and 8.5.

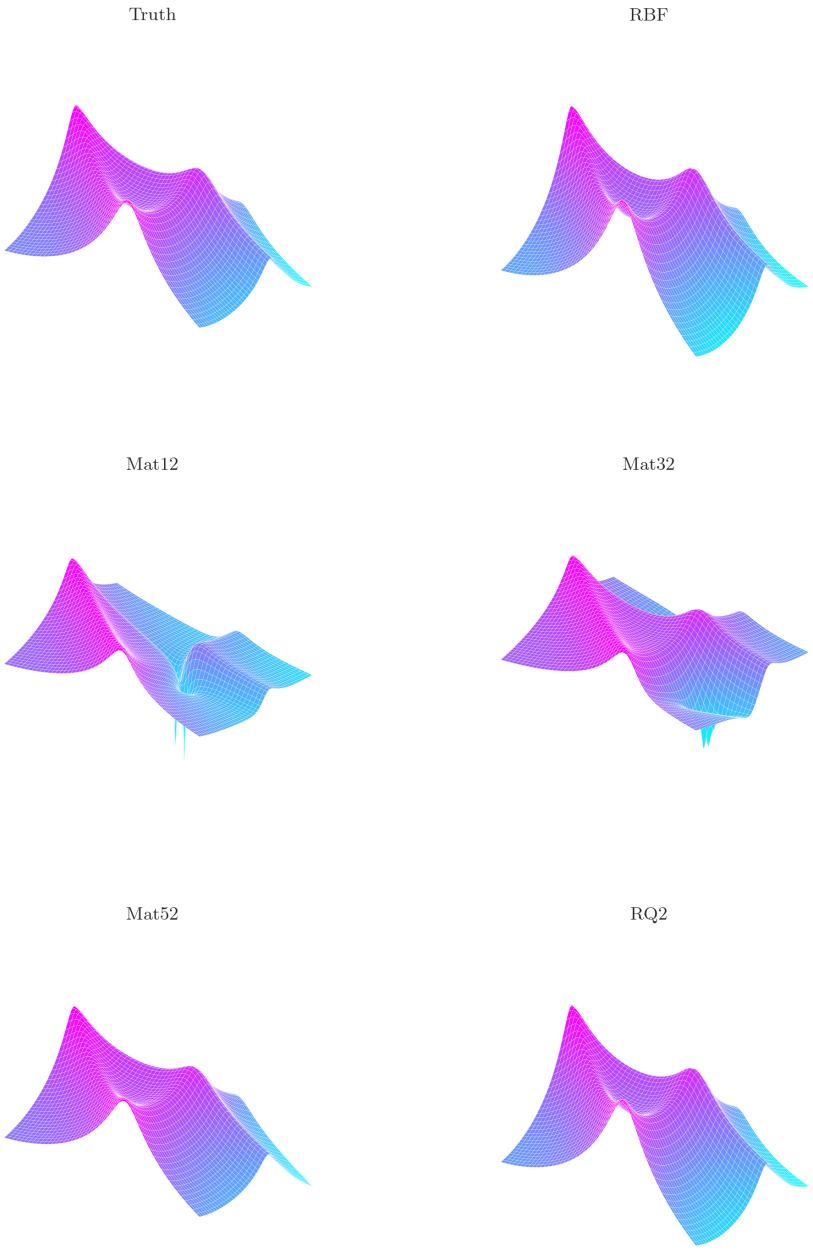


FIGURE 8.4: Surface plots of $|H_2(\omega_1, \omega_2)|$ calculated from the GP-NARX models compared to theoretical values.

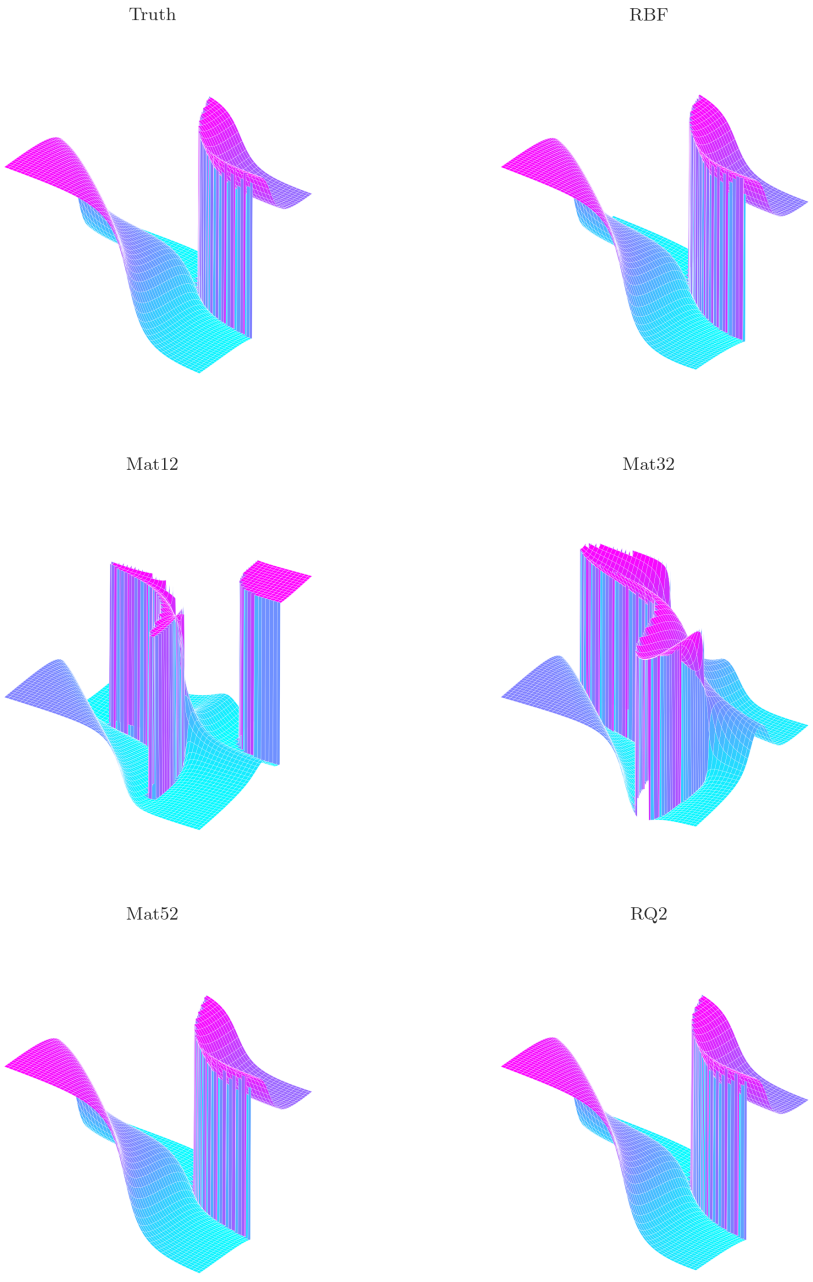


FIGURE 8.5: Surface plots of $\angle H_2(\omega_1, \omega_2)$ calculated from the GP-NARX models compared to theoretical values.

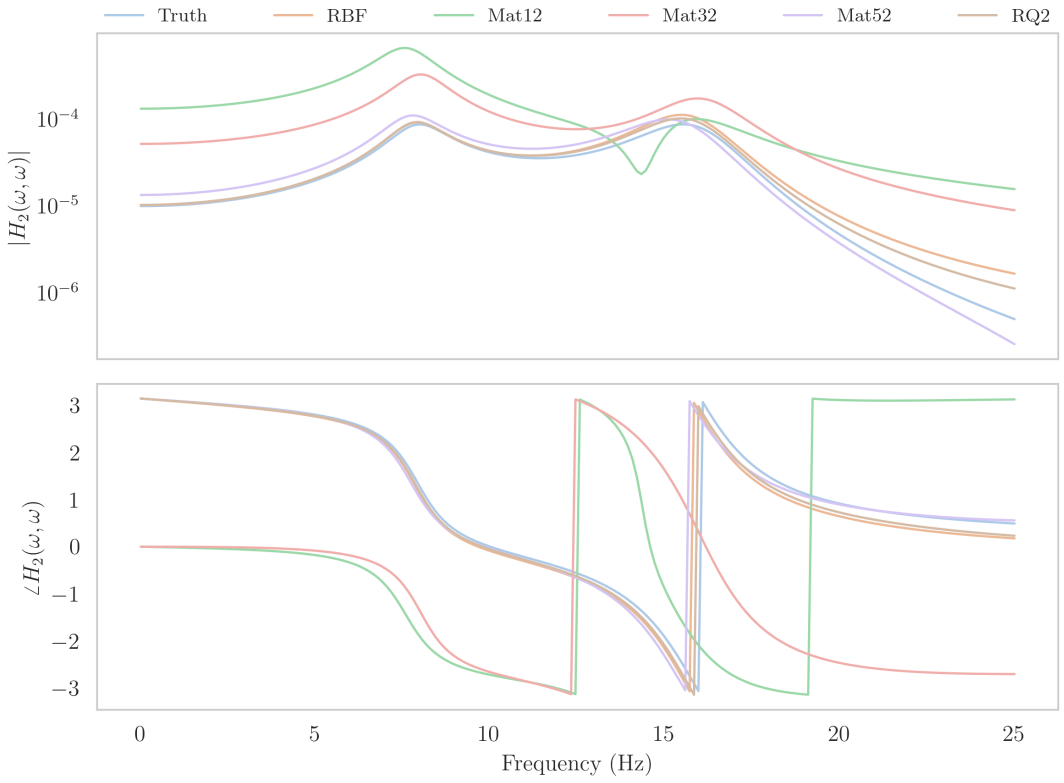


FIGURE 8.6: Main diagonal of $H_2(\omega_1, \omega_2)$ calculated from the GP-NARX models compared to theoretical values.

The results for H_3 are plotted in Figure 8.7. In the plot, the main diagonals (i.e. $H_3(\omega, \omega, \omega)$) of the three-dimensional manifolds are plotted. Once again, the smooth kernels have achieved a good reconstruction of the true dynamics, with some noise visible in the phase information. As before, the non-smooth Matérn kernels have struggled to reconstruct the higher-order nonlinear terms accurately.

8.4 Conclusions

This chapter presents closed-form expressions for the HFRFs of stationary kernel-NARX models, of which an important member is the GP-NARX model. The development of these expressions is motivated by a desire to use NLSI and the HFRFs to interpret the dynamics of the nonlinear normal modes presented in the previous chapters. The developed expressions are validated here on a numerical case study, whereby GP-NARX models are trained to represent the dynamics of a nonlinear Duffing-type equation with quadratic and cubic stiffness elements.

In the numerical case study, the quality of the model agreement on the unseen testing data is excellent (except in the case of the very rough Matérn($\frac{1}{2}$) kernel). The model

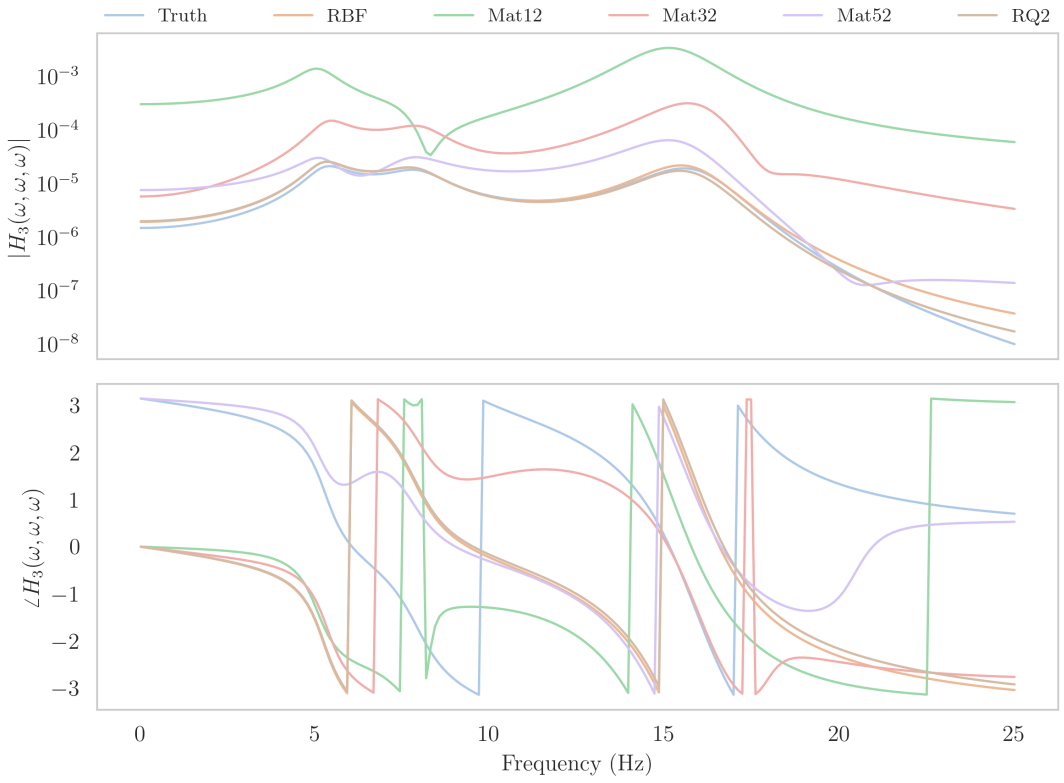


FIGURE 8.7: Main diagonal of $H_3(\omega_1, \omega_2, \omega_3)$ calculated from the GP-NARX models compared to theoretical values.

agreement with H_1 indicates that all other models have done a reasonable job of learning unbiased estimates of the nonlinear dynamics. As expected, the smooth kernels (RBF, rational quadratic) have excellent agreement, whereas the non-smooth kernels (Matérn class) have performed less well. This trend only grows more pronounced as the order of the HFRF increases. By H_3, the model agreement with kernels from the Matérn class is quite poor as seen in Figure 8.7.

These results clearly indicate the utility of HFRFs as a stringent test of model validation and bias. The poor quality of the HFRF agreement, despite strong NMSE scores, for the Matérn kernels is of particular interest. Ultimately, bias such as this will manifest as poor generalisation; the training data have been explained by spurious higher-order nonlinearities and biased linear terms.

Access to the HFRFs of powerful NLSI models opens the door to powerful analysis in the context of the NNMs generated in this thesis. In particular, it will now be possible to examine to what extent the underlying linear dynamics of the modes are the same as the linear modes of the underlying nonlinear system (modal decomposition). Additionally, the conservation of odd and even nonlinearities can be investigated through the consideration of the magnitudes of the higher-order HFRFs.

A current limitation of the results in this chapter is the reliance on the $\mathcal{L}_2$ norm in the expansion of the Taylor series. In practice, many other norms are appropriate. Although the $\mathcal{L}_2$ is a popular choice when implementing stationary kernels, it would be to extend the results in this chapter to other possible distance metrics.

Efficient computation of HFRFs is also an open question. The convenient vectorised notation used here offers a considerable saving over the explicit notation used in [49]. However, a tensor formulation can be envisaged that may offer a dramatic speedup on modern computing architecture.

Another avenue for investigation is the quantification of uncertainty of the HFRF models. In [49], a data-driven approach is adopted that estimates the uncertainty on the HFRFs by making draws from the posterior distribution of the GP and then estimates the HFRFs via a Monte-Carlo method. It is interesting to imagine if the uncertainty on the HFRFs might be calculated in closed form. However, this is left as further work at this stage and no assessment of the uncertainty in the HFRFs is included here.

Chapter 9

Quantitative analysis of statistically-independent NNMs with NLSI

With the nonlinear modes established by the cycle-GAN approach in Chapter 7 and all the machinery for analysis in place; the attention of this chapter can be turned to a qualitative assessment of the criteria of Chapter 2.

1. *Independence*: The ability of the decomposition to render the dynamics into an independent modal basis, preferably SISO.
2. *Decomposition*: The extent to which the decomposed modal dynamics represent a physically-meaningful (by some measure), basis for understanding the structural dynamics.
3. *Superposition*: The extent to which the original dynamics can be recovered from the decomposition.

The structure of this chapter as as follows: Firstly, the NLSI procedure for fitting GP-NARX models to the physical and modal displacements will be developed. Next, for each of the criteria above, a number of quantitative observations are presented, with a view to understand the nature of the NNM under a statistically-independent framework.

9.1 Nonlinear system identification

In this chapter, the approach will be to fit both linear and nonlinear models to the physical and modal displacements that were calculated in Chapter 7. The reasoning for employing both linear and nonlinear models in the identification task is to enable a comparison between both the quality of fit and quality of generalisation between excitation levels. In this way it will be possible to examine the extent to which the decomposed basis is governed by nonlinear functionals.

For the linear models, an ARX formulation will be employed as described in Chapter 2. For the nonlinear models, a GP-NARX formulation will be employed. Although practically, a large number of black-box nonlinear system identification models are applicable to this task, the GP-NARX model offers several advantages including a low number of hyperparameters and a kernel-based formulation[1].

The training approach for GP-NARX models is the following: First, a training, validation and testing dataset is taken from both the simulated physical and transformed modal data, each comprising 10^3 points. In the author's 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 of the discrete-time models; here, this is achieved by performing a grid search over the maximum number of lags in the input n_x and output n_y respectively.

A more thorough approach might be to consider a fully combinatorial lag selection scheme, whereby all possible combinations of lagged inputs (up to some maximum value) are considered. However, a full combinatorial lag selection is not deemed efficient as it can be 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 any hyperparameters $(\sigma_n^2, \sigma_f^2, \ell)$ 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,

$$J_{\text{LSO}}(\sigma_n^2, \sigma_f^2, \ell) = -\frac{1}{2}\mathbf{y}^T \left[K(X,X) + \sigma_n^2 I\right] \mathbf{y} - \frac{1}{2}\log\left|K(X,X) + \sigma_n^2 I\right| \tag{9.1}$$

[1] Quantification of uncertainty is another advantage, but that is not the focus of the work conducted in this chapter, and so is not considered.

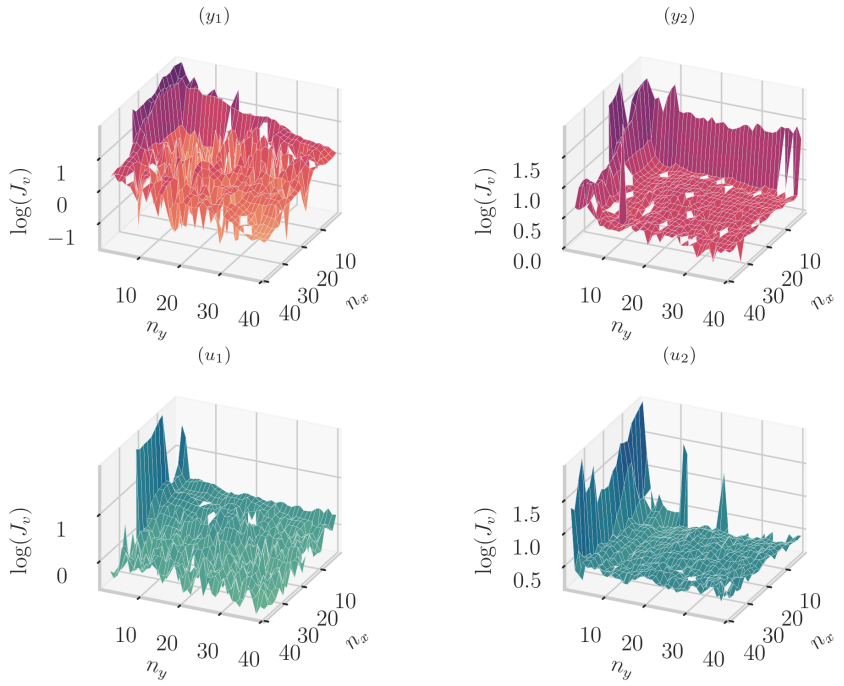


FIGURE 9.1: GP-NARX NMSE surface during LSO training. NMSE scores exceeding 100 are excluded from the plot.

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,

$$J_{\text{NMSE}} = \frac{100}{\sigma_y^2 N_J} \sum_{i}^{N_J} (y_i - \hat{y}_i)^2 \tag{9.2}$$

where $N_J = N_p - (n_x + n_y)$. A visualisation of the resultant cost surfaces is given for the GP-NARX models of the 2-DOF system in Figure 9.1. 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 has 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 more strongly encourage the models to learn the underlying dynamics of the data rather than simply providing good prediction performance.

All GP-NARX hyperparameter optimisations are completed using a quantum particle swarm optimisation (QPSO) algorithm [176]. The optimisation and training parameters

TABLE 9.1: Parameters used in the optimisation of the GP-NARX models.

Parameter	Description	Value
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
n_G	Number of generations in the QPSO optimiser	200
n_P	Population size in the QPSO optimiser	200

are collected in Table 9.1 for the reader's convenience.

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

9.2 Independence

With the NLSI models trained, attention can be turned to analysis. The first criteria of Chapter 2 is *independence*. Independence is measured here in terms of the extent to which the modal basis can be modelled as a discrete set of SISO functionals.

Observation: The modal transformation permits a high-fidelity SISO model basis

As might be expected from the nonlinear systems considered in this thesis, a good SISO representation is possible and this is seen in the quality of the model fit. Indeed, both the physical and modal displacements admit a high-fidelity SISO model basis.

For the 2-DOF system, the MPO predictions on the unseen training data are depicted in Figures 9.2 and 9.3 for the physical and modal displacements respectively. MPO NMSE scores and lag structures are collected in Table 9.2. Overall, the GP-NARX models are outperforming the linear ARX models in every case except u_2, where the performance is similar. This is quantitative evidence that the decomposed SISO basis is comprised of independent nonlinear functionals. This finding is in line with the observations of Chapter 6, where even in the SDOF case, mapping to a target Gaussian distribution induces a nonlinear functional.

As can be seen from the results, the quality of fit is fair 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

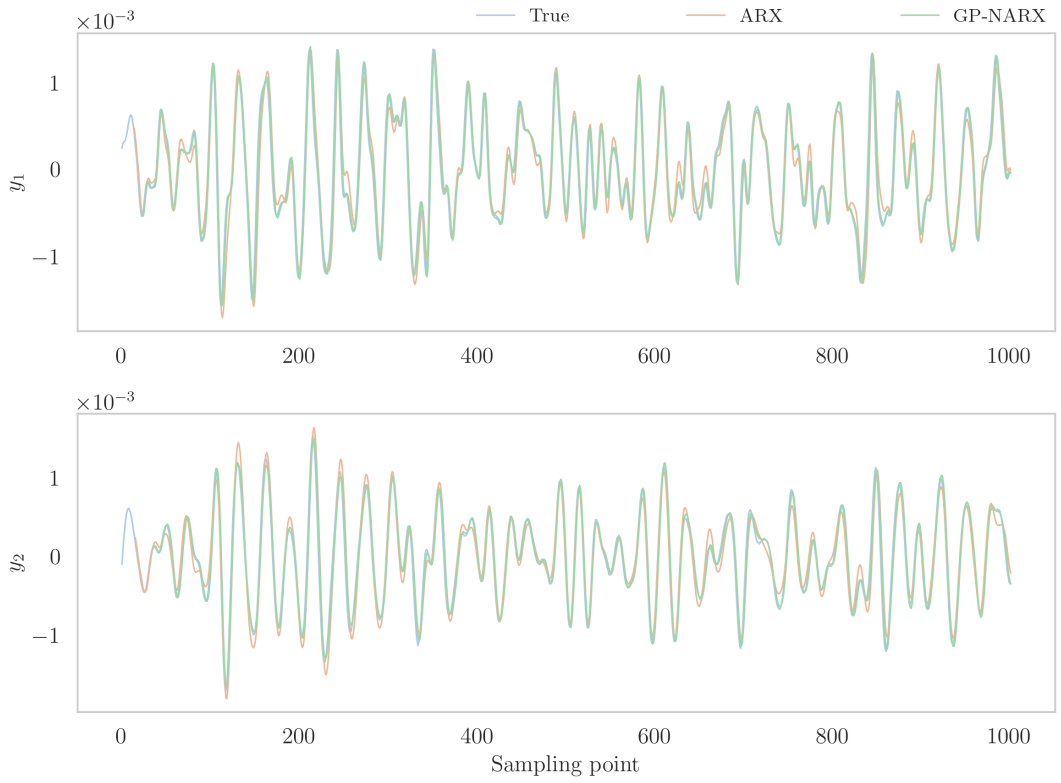


FIGURE 9.2: MPO predictions on the unseen testing data for the NLSI models of the functional $\mathbf{x} \rightarrow \mathbf{y}$ for the 2-DOF system.

TABLE 9.2: Lag structures and MPO NMSE scores for the trained models on the unseen testing data for the 2-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

leading to very rough LSO surfaces like the one seen in Figure 9.1. 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. Adding to the difficulty is the difference in signal to noise ratios between the modal dynamics, increasing the optimisation difficulty and further leading to rough LSO cost surfaces.

For the 3-DOF system, the model predictions on y and u are shown in Figures 9.4 and 9.5 respectively. Lag structures and NMSE scores are collected in Table 9.3.

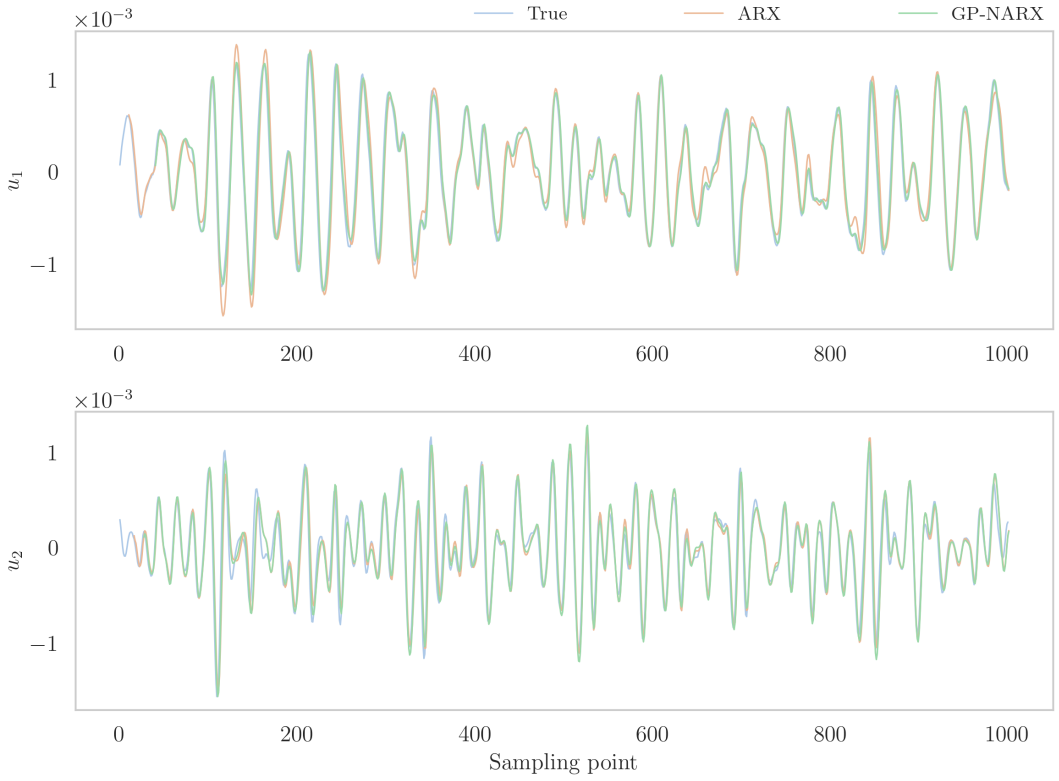


FIGURE 9.3: MPO predictions on the unseen testing data for the NLSI models of the functional $\mathbf{x} \rightarrow \mathbf{u}$ for the 2-DOF system.

TABLE 9.3: Lag structures and MPO NMSE scores for the trained models on the unseen testing data for the 3-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

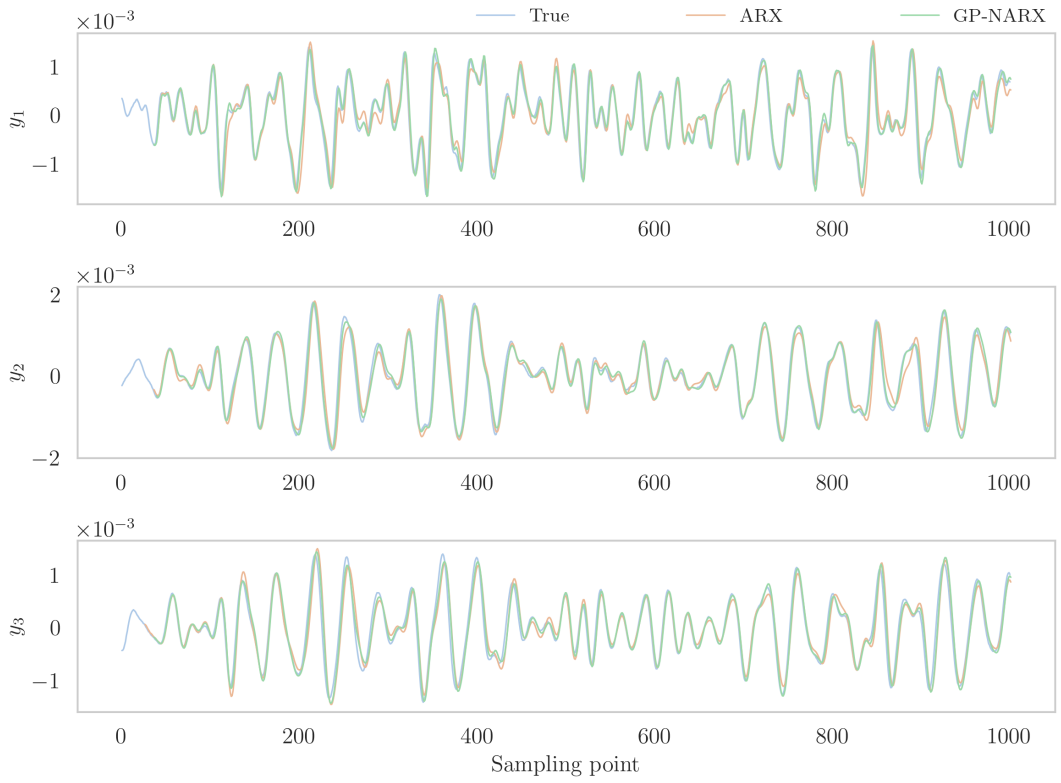


FIGURE 9.4: MPO predictions on the unseen testing data for the NLSO models of the functional $\mathbf{x} \to \mathbf{y}$ for the 3-DOF system.

Although it is unsurprising that a SISO model basis is possible for the modal displacements (indeed it is also possible for the physical displacements), it is promising that the modal decomposition does not introduce any coupling between coordinates that might affect the quality of fit of the SISO GP-NARX models used here.

Observation: The decomposed SISO basis shows approximate generalisation between excitation levels

As well as the three criteria of Chapter 2, linear modal analysis is also invariant to the excitation level.

Although it was shown in Chapter 6 that a Gaussian target corresponding to the underlying linear system is insufficient to produce amplitude invariance in general, Figure 6.5 shows that the exact transformation varies only weakly with changing amplitudes. It is therefore interesting to examine the extent to which the SISO model basis that has been established is able to generalise (locally), between excitation levels.

In order to assess approximate generalisation, additional nonlinear data are generated for both the 2-DOF and 3-DOF systems at the lower excitation levels of $\sigma_x = 14$ and

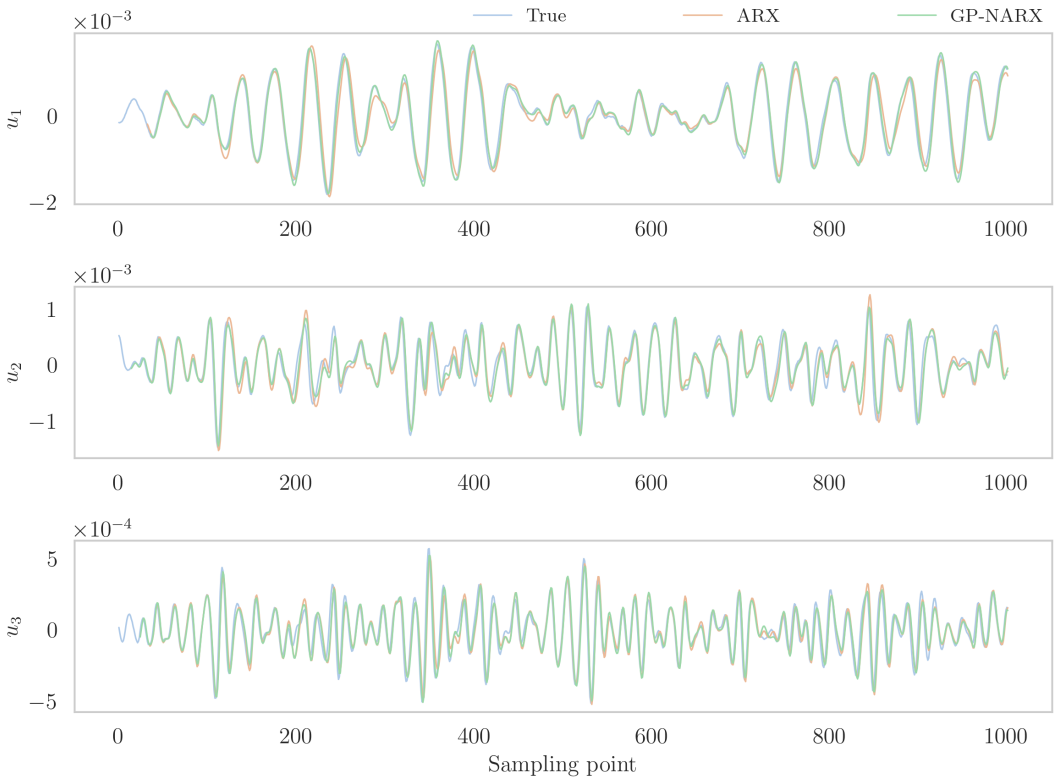


FIGURE 9.5: MPO predictions on the unseen testing data for the NLSO models of the functional $\mathbf{x} \rightarrow \mathbf{u}$ for the 3-DOF system.

$\sigma_x = 8$. 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 resultant MPO prediction NMSE scores (on the unseen testing data), from the NLSI models trained above are then collected for the modal displacements in Figures 9.6 and 9.7.

The figures show good generalisation of the nonlinear models compared to linear models which constitutes further evidence of nonlinearity in the functional $\mathbf{x} \rightarrow \mathbf{u}$. As one would expect, the best generalisation performance for linear and nonlinear models is present when the training excitation level is close to the testing level. For the 2-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 3-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 author offers two explanations for this phenomena. The first is that the NLSI may be entirely limited by

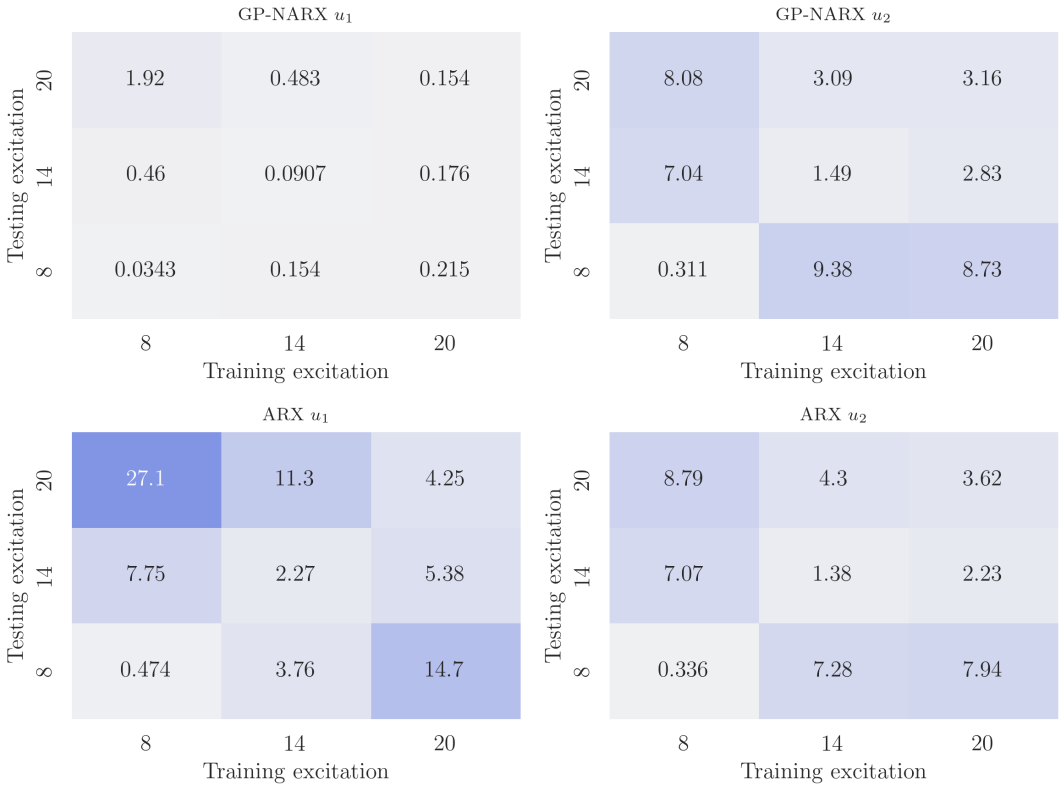


FIGURE 9.6: Cross-validation NMSE of GP-NARX and ARX models of modal displacements at different excitation levels for the 2-DOF system.

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. Although unlikely to be possible in general, for some types of nonlinear systems[2], 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, there is certainly insufficient proof at this stage.

One possible hindrance to the generalisation performance here 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

[2]For example, systems with a single nonlinear element such as those considered here.

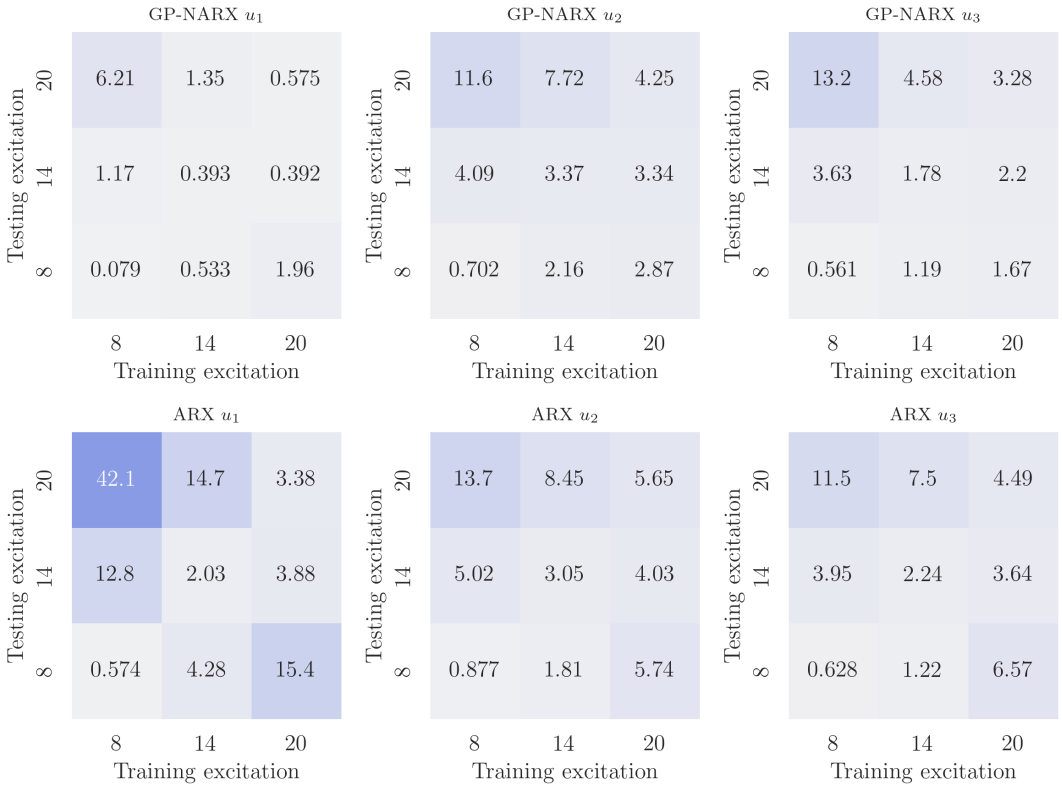


FIGURE 9.7: Cross-validation NMSE of GP-NARX and ARX models of modal displacements at different excitation levels for the 3-DOF system.

despite having potentially very dissimilar dynamics. A more sophisticated normalisation scheme (or even an approach based on second order statistical moments), is an interesting avenue for additional work and may enable a greater generalisation of the modal transformation between excitation levels.

9.3 Decomposition

In Chapter 5, several metrics for assessment of decomposition of modes were presented; however, in the conclusions of that chapter it was argued that the following properties must be approached for a better assessment:

- The preservation of resonances present in the physical displacements.
- The extent to which the transformation correctly reduces to the linear modes in the absence of nonlinearity.
- The extent to which the underlying linear dynamics are preserved.

- The extent to which the modal transformation is able to generalise between excitation levels (locally or globally).

Thus far, it is only really the first and last of these that have been evaluated. The inductive bias of independent resonances has been applied within the machine learning-framework and this has successfully generated a single resonance structure as can be seen in the results of Chapter 7.

Now that access to both NLSI models of the functionals $\mathbf{x} \rightarrow \mathbf{y}$ and $\mathbf{x} \rightarrow \mathbf{u}$ and the HFRFs of those models have been established, quantitative assessment of the remaining properties can at last proceed.

Observation: The underlying linear dynamics correspond closely to the linear modes

The second and third properties proposed in Chapter 5 can really be thought of as two sides of the same coin. If the underlying linear modal structure is preserved by the modal transformation then in the limit of linearity (for example close to the equilibrium), it is those dynamics that will be present. In order to assess the underlying linear structure, it is sufficient to consider the $\mathbf{H}_1$ of the NLSI models. $\mathbf{H}_1(\omega)$ is plotted for the $\mathbf{x} \rightarrow \mathbf{y}$ and $\mathbf{x} \rightarrow \mathbf{u}$ functionals in Figure 9.8 for the 2-DOF system and in Figure 9.9 for the 3-DOF system. Also plotted in each figure is the true $\mathbf{H}_1(\omega)$ as derived from the equations of motion see Appendix A.

The first observation from the figures is that the agreement between the $\mathbf{H}_1(\omega)$ for the original system $\mathbf{x} \rightarrow \mathbf{y}$ and the theoretical values from harmonic probing are excellent. This is a promising result as it indicates that the NLSI scheme employed here is sufficiently powerful to permit good agreement with the HFRFs without bias in the linear components. This result is a good indication that the HFRFs of $\mathbf{x} \rightarrow \mathbf{u}$ can be trusted to be accurate even though no theoretical comparison is available.

Further indications of a good decomposition into SISO functionals are the largely single-peak structures of the $\mathbf{H}_1$ for the modal coordinates, as might be expected of the underlying linear modal decomposition. 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 2-DOF system and $[0.33\%, 1.73\%, 6.85\%]$ for the 3-DOF system.

These results indicate that the underlying linear modal structure has been largely retained by the modal transformation. The discrepancies above can likely be explained

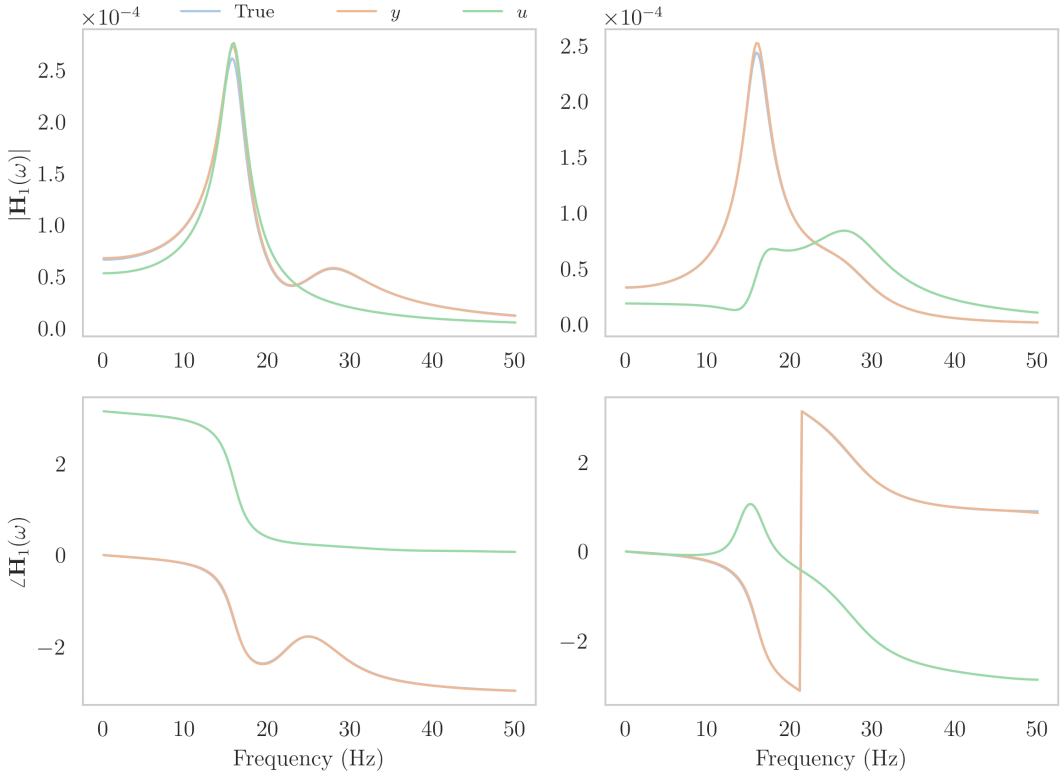


FIGURE 9.8: Comparison of the $\mathbf{H}_1$s for the 2-DOF nonlinear system.

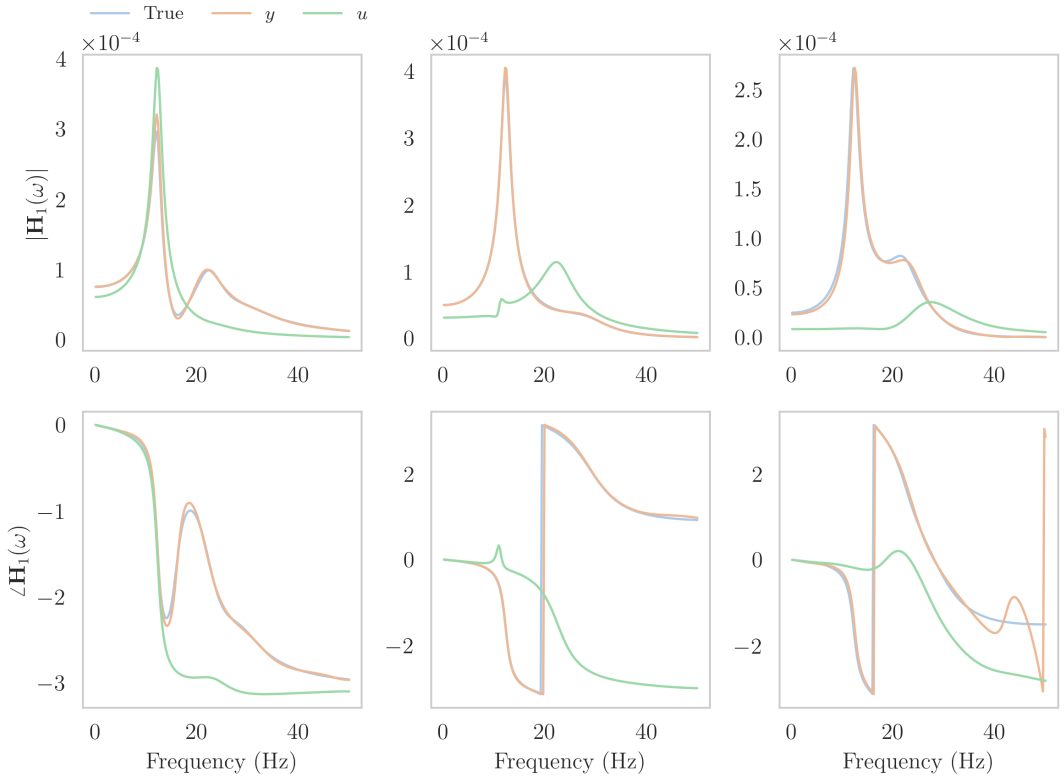


FIGURE 9.9: Comparison of the $\mathbf{H}_1$s for the 3-DOF nonlinear system.

by errors accumulated during the training of the nonlinear models. That the percentage error increases with increasing resonance frequency is perhaps unsurprising. Considering the PSDs of the physical displacements, the relative magnitudes of the resonances in both systems decreases dramatically. Practically, this effect results in drastically-different signal-to-noise ratios that negatively affects the quality of nonlinear models. This effect is furthermore visible in the deterioration of the MPO NMSE scores of the nonlinear models of the modal dynamics.

The retention of the underlying linear structure (as measured in terms of the location of the underlying linear natural frequencies), is an encouraging result in favour of statistically-independent NNMs.

Observation: The modal transformation induces a non-physical quadratic nonlinear structure in the modal dynamics

The magnitudes and phases of the $\mathbf{H}_2(\omega, \omega)$ for the modal dynamics are plotted in Figure 9.10. Neither of the benchmark systems considered here have any even nonlinearities present and so it might be expected that the $\mathbf{H}_2(\omega, \omega)$ should all be strictly zero in the modal transformation. However, this is not what is observed. Figure 9.10 depicts evidence of structure in the $\mathbf{H}_2$. How might this spurious quadratic structure be explained?

The author believes that there are two factors that contribute to the presence of quadratic nonlinearity in the modal dynamics. The first can be attributed to the flexibility of the GP-NARX model and the form of $\mathbf{H}_2$. Consider the form of the second order HFRF[3],

$$\mathbf{H}_2(\omega_1, \omega_2) = -K_2 \mathbf{H}_1(\omega_1) \circ \mathbf{H}_1(\omega_2) \circ \mathbf{H}_1(\omega_1 + \omega_2) \tag{9.3}$$

Because the above term is constructed from a product of $\mathbf{H}_1$ terms, any quadratic structure in the nonlinear model (even noise) will manifest as peaks at ω_1, ω_2 and $\omega_1 + \omega_2$. Given the extreme flexibility of the GP-NARX model with a squared-exponential kernel it is perhaps unsurprising that some spurious quadratic component is generated.

The second factor contributing to the nonlinear structure of Figure 9.10 is the nature of the nonlinear transformation itself. Consider again the result of Chapter 6 whereby the modal transformation is applied directly to the equation of motion for the SDOF case,

$$g''(u)\dot{u}^2 + \ddot{u}g'(u) + cg'(u)\dot{u} + kg(u) + \epsilon g(u)^3 = x(t) \tag{9.4}$$

[3] A full derivation of these HFRFs is available in Appendix A.1.

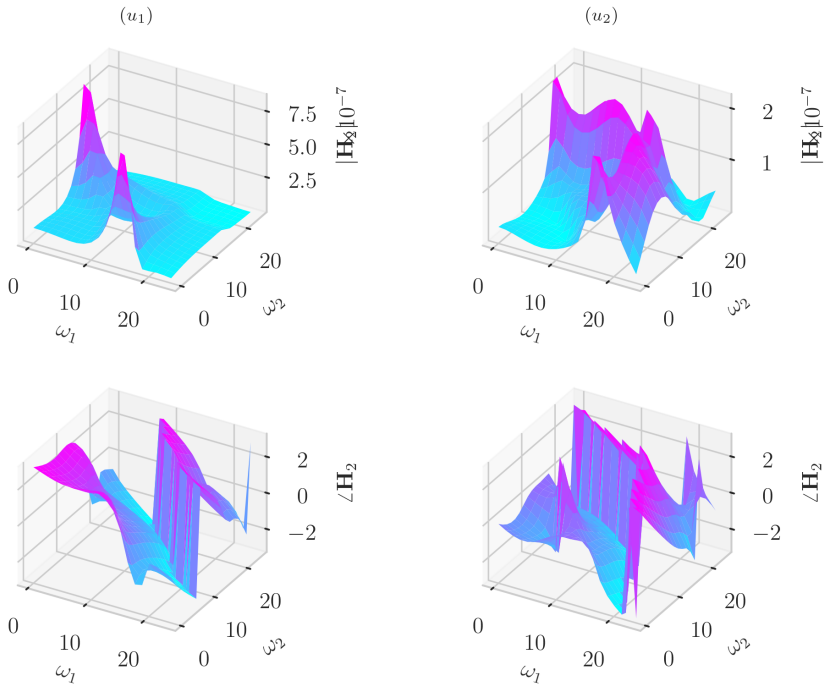


FIGURE 9.10: $\mathbf{H}_2$ of the GP-NARX model of $\mathbf{x} \rightarrow \mathbf{u}$.

Calculating now the HFRFs of this modal equation of motion reveals that quadratic structure has been introduced[4].

$$H_2(\omega_1,\omega_2) = H_1(\omega_1)H_1(\omega_2)H_1(\omega_1{+}\omega_2)\left[a_2(ci(\omega_1+\omega_2)-\omega_1^2-\omega_2^2+k)+3a_3i(\omega_1+\omega_2)\right] \tag{9.5}$$

Now, for a nonlinear mapping that is purely-odd (as might be expected for the odd-type nonlinearities that are seen in the systems considered here), the second order Taylor series coefficient must be zero ($a_2 = 0$). The above can now be re-written,

$$H_2(\omega_1,\omega_2) = 3a_3i(\omega_1+\omega_2)H_1(\omega_1)H_1(\omega_2)H_1(\omega_1+\omega_2) \tag{9.6}$$

The effect of the $g'(u)$ terms in the equation of motion is such that even for a purely-cubic system with a purely odd nonlinear mapping, some non-physical quadratic structure is still introduced to the modal dynamics.

[4]For a full derivation of these HFRFs, the author is directed to Section A.2 of the appendices of this thesis.

Although these effects are perhaps not a desirable property of the statistically-independent NNMs, it is an important realisation that must be considered when applying the methodology of statistically-independent NNMs in practice.

9.4 Superposition

Another important measure of the quality of the modal decomposition is superposition. In Chapter 5, it was argued that an appropriate measure of modal superposition is the extent to which the original dynamics can be reconstructed via the inverse mapping. Quantitative assessment of superposition will be measured in this chapter using the familiar normalised mean-square error metric,

$$J_{\text{NMSE}} = \sum_{j}^{n} \left[\frac{100}{\sigma^2_{\mathbf{y}_j} N} \sum_{i}^{N} (y_{ji} - \hat{y}_{ji})^2 \right] \tag{9.7}$$

where $\hat{\mathbf{y}}_\mathbf{j}$ are the reconstructed physical displacements from the j^{th} degree of freedom.

Observation : The decomposition permits a highly accurate inverse transformation

The superposition performance of the transformations learned by the cycle-GAN are excellent. An advantage of the cycle-GAN is that the inverse mapping is constructed alongside the forward modal transformation. Practically, this structure results in excellent reconstruction performance. For the 2-DOF system, the overall reconstruction introduces NMSEs of 0.0262% and 0.0133% for each DOF respectively over the entire 10^5 simulated points. Reconstruction performance for the 2-DOF dataset is depicted in Figure 9.11 for a subset of 10^3 points. As can be seen in the figure, the agreement is excellent, and the reconstructed displacements are visually indistinguishable from the true displacements.

Performance on the 3-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. A subset of the reconstructed signal is plotted in Figure 9.12 for visualisation.

Once the cycle-GAN is trained, the overall mapping from $\mathbf{y}$ to $\hat{\mathbf{y}}$ or $f(f^{-1}(\mathbf{y}))$, is itself a static map. If the the modal inverse function f^{-1} is a true inverse, the map from $\mathbf{y}$ to $\hat{\mathbf{y}}$ should be close to the identity. This map is plotted in Figure 9.13 for the 2-DOF

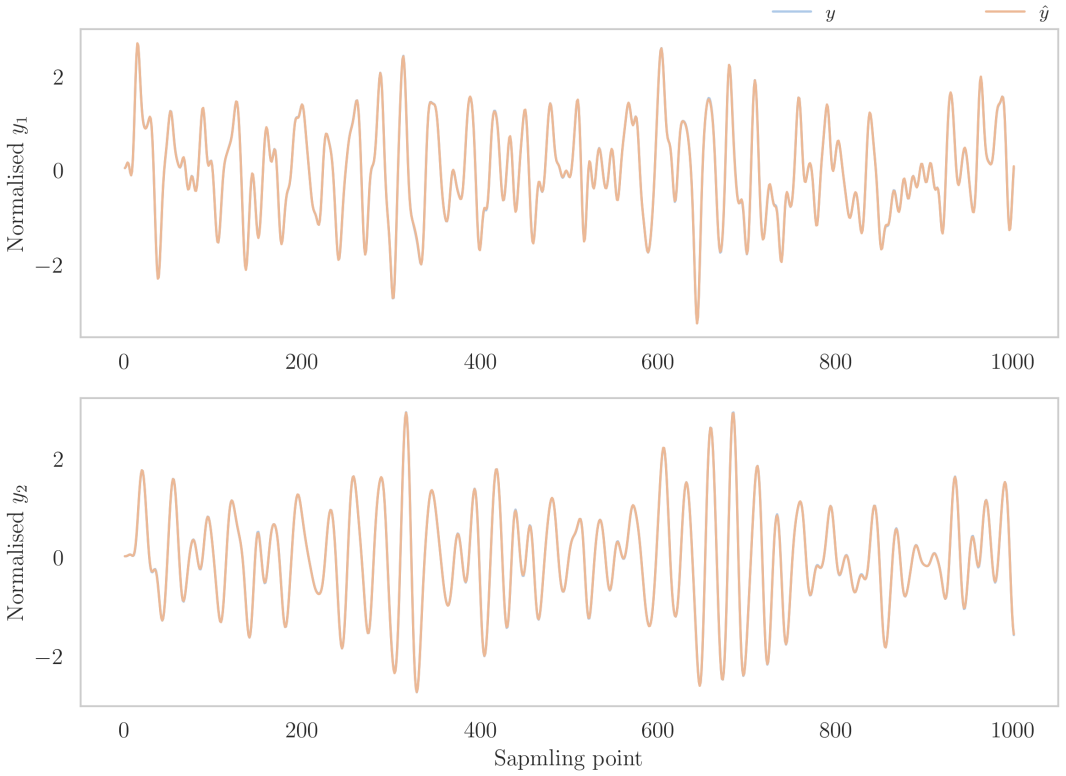


FIGURE 9.11: Original vs. reconstructed displacements for the 2-DOF system.

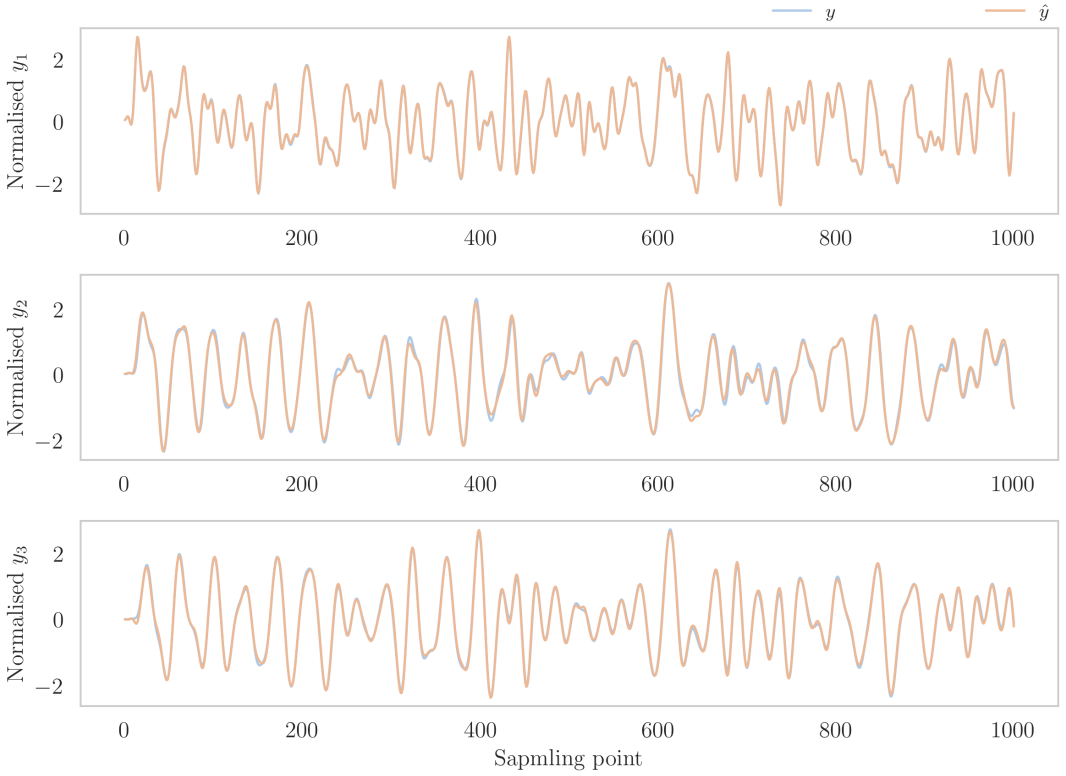


FIGURE 9.12: Original vs. reconstructed displacements for the 3-DOF system.

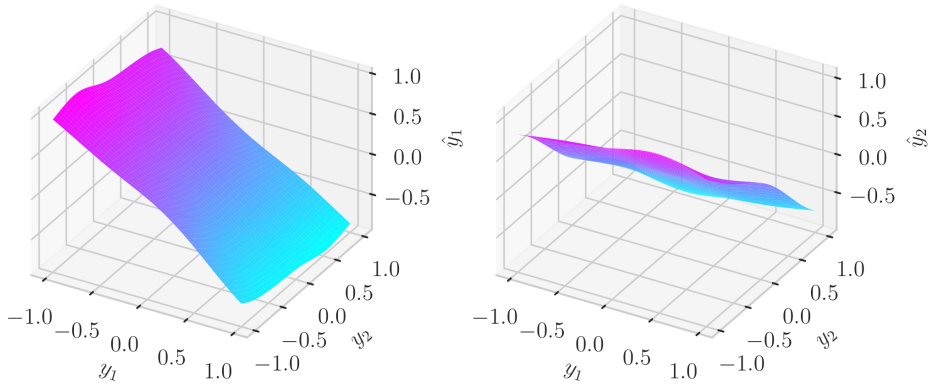


FIGURE 9.13: Overall functional reconstruction of the forward and inverse modal transformations for the 2-DOF system.

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 Figure 9.13 that depart from unity are the regions of the input-space corresponding to high amplitudes in the displacements. Because of the nature of the Duffing-like equations of motion, 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.

Observation: The decomposed SISO model basis reconstructs the dynamics accurately

Perhaps a more 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 the NMSE score above. 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

TABLE 9.4: 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

of such an investigation are presented in Table 9.4. In the table, the SISO model reconstruction (the y_i reconstructed from the NLSI predictions of the u_i), are compared to the reconstructions from the SIMO case ($\hat{\mathbf{y}} = f^{-1}(f(\mathbf{y}))$) 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 9.4, the SISO decomposition produces excellent reconstructions for the 2-DOF and good reconstructions for the 3-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 NMSE MPO errors on the corresponding modal coordinates. An example is present for the 3-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 because of 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 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,

$$d \times \binom{r + n_x + n_y}{r} \tag{9.8}$$

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.

9.5 Conclusions

This chapter has introduced for the first time some quantitative assessment of the dynamics of the modes constructed under the statistically-independent framework. The approach has been to first model the dynamics of the physical and modal displacements using ARX and GP-NARX models. Once fitted, the quality of fit, generalisation performance, HFRFs and reconstruction performance of these models has been tested with a view to offer practical observations of the criteria of Chapter 2.

In terms of *independence*, the quality of the SISO model fit to the modal coordinates was evaluated and found to be excellent for the 2-DOF system and good for the 3-DOF system. It was also observed that the model performances of the nonlinear GP-NARX models were far stronger than those of the linear ARX models, adding further evidence to the analysis in Chapter 6 that the modal dynamics are governed by nonlinear functionals. The approximate generalisation of the SISO models has been assessed and shown to be favourable when nonlinear models are used. This too, is in line with analysis of the SISO case in Chapter 6 where the modal transformation is shown to only vary weakly with changing amplitudes when a Gaussian target distribution is used.

Modal *decomposition* is assessed by analysis of the HFRFs of the nonlinear models of the physical and modal dynamics. It is shown that the underlying linear components of the physical displacements has reconstructed the true linear dynamics with good accuracy. The underlying linear component of the modal dynamics is shown to have a largely single-peak structure that corresponds closely to that of the linear modes. These results indicate that the NNMs are able to retain the underlying linear modes in a purely data-driven manner. It is also shown by analysis of the $\mathbf{H}_2$ that the application of the nonlinear map necessarily introduces a quadratic structure into the modal dynamics, even when the nonlinear map is purely odd.

The *superposition* performance of the modes is assessed for both the SIMO (whereby $\hat{y} = f(f^{-1}(\mathbf{y}))$), and SISO reconstruction cases (whereby the SISO models are used to perform the reconstructions), by an NMSE metric. It is found in both cases that the reconstruction performance is strong, and that the degradation in moving to the SISO reconstruction is not extreme. This result indicates that the overall nonlinear map $f(f^{-1}(\mathbf{y}))$ is close to the identity, and that the SISO basis offers a complete picture of the dynamics (within some local amplitude regime).

As well as the criteria of Chapter 2, the statistically-independent NNM framework has other desirable properties that make it an attractive option in a practical setting. Principally, the NNMs are constructed in an output-only and data driven way with very little

prior information required to produce the decomposition. Additionally, the neural architecture permits both forward and inverse transformations to be learnt simultaneously in such a way that ensures invertibility of the modal dynamics.

Despite the propitious results of this chapter, several limitations of the statistically-independent framework persist. Perhaps the most pressing of these is that thus far consideration has only been given to SIMO structural systems. Although this type of system is common in real-world structural dynamics (multi-storey buildings subject to ground vibrations, monopiles subject to wave loading, modal tests etc.), it is an interesting open question how the framework could be extended to the fully MIMO case. Some discussion in this vein is saved for the following chapter.

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 [23] already represents a significant computational saving over the approach of [22]. However it interesting to imagine other methods that might lessen the computational cost.

At this stage, no formal equivalence can be currently claimed between the present framework and the previous approaches of Rosenberg and Shaw-Pierre, because of 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.

A motivation for the investigation of the statistically-independent framework was that it can be readily applied in a similar approach to a linear modal testing campaign. An indicative workflow was:

1. Capture some displacement data from some SIMO structure of interest, subject to broadband excitation.
2. Presume or establish the presence of nonlinearity (for example by assessment of the coherence).
3. Conduct statistically-independent modal analysis on the data and retrieve the modal dynamics.

4. Finally perform NLSI on the modal dynamics as SISO functionals.

When applied successfully, a compact set of modal coordinates are established that have been shown here to reduce (in the limit of linearity), to the underlying linear modes.

Chapter 10

Conclusions and further work

10.1 Concluding remarks

As the demand for lighter, stronger and high-performance structures increases, nonlinearity is becoming an inescapable facet of the modern engineering design landscape. Cutting edge materials, geometries and assemblies break the assumptions of linearity and linear analyses can no longer be applied.

In this thesis, three significant challenges in nonlinear structural dynamics have been identified.

- Challenge I: Nonlinear system identification.
- Challenge II: Exact solutions to nonlinear differential equations.
- Challenge III: A nonlinear extension to linear modal analysis.

These challenges are motivated by the pressing need for advanced treatment of nonlinearity in structural dynamics.

After a review of the background theory and literature in Chapter 2, the objectives of the thesis were established. The objectives were to conduct novel analyses pertaining to two of the three challenges; exact solutions to nonlinear ODEs and a nonlinear alternative to linear modal analysis. For each of these challenges, some concluding remarks are offered here.

Towards exact solutions to nonlinear ODEs

Nonlinearity arises in structural dynamics from the physical laws that govern structures. Applying the physical conditions and appealing to conservation laws, leads to nonlinear differential equations that encode the dynamics. Whereas in the linear case exact closed form-solutions are available, precious few solutions are available in the nonlinear case.

Exact, closed form solutions to nonlinear differential equations have the potential to revolutionise understanding of nonlinear phenomena. It could be argued that even particular solutions to some differential equations could give vital insight into the behaviour of nonlinearities in real-world structures.

In Chapter 2, some background to the task of solving differential equations was presented. Some structural classifications of differential equations were reviewed and some theorems relating to existence and uniqueness were revised. Discussion was held over what exactly constitutes a *solution* to a differential equation. Both approximate and analytical methods were considered, with a particular focus on methods that are able to specify exact, closed-form solutions by heuristic search. Of such methods, a promising technique was found to be symbolic regression. Several authors have applied symbolic regression techniques to solving differential equations with varying degrees of success. However, in the considered literature, almost no authors have placed the emphasis of their efforts on exact solutions only.

A research gap was therefore identified; to combine results from symbolic regression, computer algebra software and insights from heuristic search problems to attempt to find a novel methodology with the potential to address as-yet unsolved nonlinear differential equations.

With the research question established, Chapter 3 began the consideration of exact solutions by investigating the task of solving a differential equation as a heuristic search problem. The aspects of the search spaces present in the problem were considered and some heuristic measures for search-problem difficulty (complexity and locality). were reviewed. Expressions were also derived for the search space sizes of several common expression encoding schemes. The chapter concluded with a consideration of benchmark suites of differential equation problems seen in the literature. A common benchmark suite (that of Tsoulos and Lagaris), was found to confound results by not separating the difficulty in solving the differential equation from the complexity of the solution. A novel suite of benchmark problems was proposed to address this issue.

Chapter 4 presented a novel encoding scheme for symbolic regression of differential equations – the affine regression tree. The new approach combines the structure detection

of a traditional expression tree encoding with additional structure at each node, motivated by the idea of locality in the constants. The sizes of the search spaces arising from the new encoding scheme were calculated and compared to traditional approaches. Although larger, the search spaces of the affine regression tree were found to encourage a more compact representation for sparsely-parameterised equations.

The remainder of Chapter 4 was concerned with a comprehensive benchmark test of the affine regression tree and two other approaches; an expression tree scheme implemented by the author and the expression grammar approach in [83]. The methods were compared on two benchmark suites of linear and nonlinear ODE problems, that of Tsoulos and Lagaris, and the new benchmarks proposed in Chapter 3.

Results from the benchmarking tests were mixed. It was shown that for the most compact expressions, (representable with just one or two nodes), the affine scheme requires fewer objective function evaluations on average to find the exact solution. However, as the complexity (i.e representation size), of the solution increases, there is significant degradation. For the most complex solutions (requiring a greater number of nodes to represent), almost no exact solutions were found. Perhaps most concerning, is that none of the approaches considered were able to significantly outperform a random-sampling based approach. Several additional investigations were presented that demonstrated the effects of hyperparameters in the affine encoding scheme. In particular it is found that a sparse initialisation of the constants ($a_0 = 1$ and $b_0 = 0$) improves performance.

Particularly interesting are the results from the proposed 'parameter snap' meta regression procedure. The method, whereby values of parameters are first specified approximately by a simulated annealing procedure and then 'snapped' to nearby exact integers and constants, gave the best overall performance on the benchmark problems. Especially encouraging was that the inclusion of the procedure increased the number of runs that found the exact solution to one of the more challenging ODE problems.

A trend seen throughout the benchmark study was that the homogenous structure of the ODE had almost no bearing on the difficulty when measured in terms of number of function evaluations required on average to find the exact solution. Instead, the complexity of any given ODE problem under the heuristic search problem is governed entirely by the form of the solution itself. This is evidenced by the results on the new benchmark suite in Figure 4.10. The author draws encouragement from these results. It is clear that the proposed methodology has no ties to linearity and the traditional metrics of what makes a differential equation easy (or difficult) to solve. One might therefore imagine that if the methods proposed in this thesis could be improved sufficiently there is no reason that a solution could not be found given a suitable basis set of functions and constants.

In Chapters 2 and 4, the following line of reasoning was established. For a given nonlinear differential equation, either:

1. Closed form solutions do not exist in terms of known transcendental functions and objects.
2. Closed form solutions do exist, but heuristic methods are insufficiently powerful to locate them.
3. Closed form solutions are available and can be found by heuristic means.

In the opinion of the author, the results of this thesis do not give cause to prefer any one of these statements in general. Certainly, symbolic regression of differential equations has been shown to be an arduous task. However, the key advantage here is that symbolic regression does not hold any preference for the form of the differential equation under consideration, be it linear or otherwise.

The author sees at this point no cause to abandon hope that closed-form solutions can be found. Even in the absence of general solutions, lesser results may yet be instrumental in developing understanding of nonlinear phenomena.

Statistically-independent nonlinear normal modes

Linear modal analysis has become the dominant framework for the identification, modelling and simplification of linear structural dynamics. Modern hardware and software tools have rendered LMA highly accessible in both industrial and research environments. Lamentably, LMA cannot be applied in the case of nonlinear dynamics. The presence of nonlinearity in cutting-edge engineering design is only increasing. There is therefore an urgent need to develop methods that can act as a practical nonlinear alternative to LMA.

In Chapter 2, the LMA theory was reviewed with a focus on the aspects of the method that make it useful in practice. These were found to be;

- *Independence*: The modal transformation decouples the equations of motion exactly.
- *Decomposition*: The modal properties (natural frequencies, damping ratios and modeshapes), are a physically meaningful description of the dynamics.
- *Superposition*: The physical displacements can be exactly reconstructed from the modal contributions.

- *Invariance*: The modal parameters are invariant to changes in the level of excitation.

In the nonlinear literature, a great many objects have been described as an NNM, each able to preserve only a subset of the useful properties of the linear case. The author presents the argument that the criteria for a practical nonlinear extension to nonlinear modal analysis should be defined in terms of these useful properties. The following definitions were offered;

1. *Independence*: The ability of the decomposition to render the dynamics into an independent modal basis, preferably SISO.
2. *Decomposition*: The extent to which the decomposed modal dynamics represent a physically meaningful (by some measure), basis for understanding the structural dynamics.
3. *Superposition*: The extent to which the original dynamics can be recovered from the decomposition.

The above criteria were then used to assess the utility of NNM frameworks that have been suggested in the literature. The two major viewpoints to have gained traction are that of Rosenberg [18, 101], and that of Shaw and Pierre [19, 113, 114, 117]. Both approaches to NNMs have attracted a great deal of development, and the major waypoints were reviewed. Despite their popularity, the methods are both shown to have practical limitations, including an inability to model the dynamics away from resonances and a somewhat technical formulation.

Data-driven approaches to NNMs were also considered. Of these approaches, the recently proposed statistically-independent framework [22] is viewed as a promising approach. However, additional investigation is required to establish the extent to which the properties of LMA are retained. Although several studies [23, 126–129], have demonstrated the method on both simulated and experimental datasets, analysis of the modal decomposition had thus far only been presented in a qualitative manner.

A research opportunity was therefore identified; to assess the statistically-independent framework for nonlinear modal analysis, quantitively, in terms of the criteria identified in Chapter 2.

Consideration of the problem here commences in Chapter 5. The chapter began by introducing the background for the statistically-independent framework. Furthermore, for each of the criteria of Chapter 2, a number of quantitative metrics were introduced.

The motivation for this was two-fold. The metrics would be able to serve as a measure of the utility of the method and as *inductive biases* within a machine-learning framework that might be used to ensure a useful decomposition by construction. Of the proposed metrics, two novel measures of unimodality are proposed; one based on the convolution operation and another that appeals to monotonicity tests. The new metrics are shown to be effective in a simplified test comprised of toy data corrupted by Gaussian noise. Despite the utility of the proposed metrics, the need to gain a greater understanding of the modal dynamics is motivated.

In Chapter 6, an attempt is made to learn the nonlinear modal transformation directly from the equations of motion. An approach is suggested based on the application of the Fokker-Plank-Kolmogorov (FPK) equation. Results are demonstrated for a trivial single degree-of-freedom system. It is shown that, via the application of the FPK equation, the stationary density of a nonlinear SISO functional can be mapped arbitrarily into a different probability distribution by the numerical solution of an ODE problem. Although simplistic, the SISO results lend important theoretical support to the statistically-independent framework. The chapter continues by demonstrating that even if the stationary distribution of a system is Gaussian, it is not necessarily governed by linear dynamics despite the converse being true. Exact modal decompositions from MDOF systems are also considered. It is shown that the FPK and change-of-variables equations alone are insufficient to derive the modal transformation. Instead, a machine-learning approach based on the residuals between the ratio of the physical and target modal distributions and the determinant of the Jacobian of the nonlinear modal transformation is envisaged.

Chapter 7 introduces two benchmark nonlinear systems that are used as a case-study for the rest of the thesis. For these two systems, NNMs are constructed by two methods: The multinomial approach of [22] (modified to consider alternative nonlinear measures of correlation), and the neural approach of [23]. The performance of the nonlinear decomposition arising from the two methods is compared in a qualitative manner. It is found that the cycle-GAN approach produces a better decomposition as measured by the extent to which the modal PSDs have single-frequency peaks. It is reasoned that the inclusion of several inductive biases (including conformality and orthogonality of the convolved spectra), lead to the increased performance. The chapter concludes by motivating analysis that is able to directly assess the nature of the decomposed modal dynamics.

Machinery for the assessment of the modal dynamics is established in Chapter 8. The use of NLSI as a method of investigating the modal dynamics is introduced. However, it is recognised that interpretation of the NLSI models is an open problem. A convenient

method for the interpretation of black-box NLSI models is the Volterra series. It is shown that, in the frequency domain, the Volterra kernels become higher-order FRFs that encode the frequency responses (HFRFs) of dynamic systems at higher orders. The chapter proceeds by developing closed-form expressions for the HFRFs for an entire class of black-box NLSI models with a stationary kernel-NARX formulation. The derived expressions are demonstrated in a case-study example of GP-NARX models fitted to data generated from a simulated nonlinear system with quadratic and cubic stiffness nonlinearities. Five stationary kernels are selected and the HFRFs of the trained models are calculated. It is shown that the HFRFs act as both a convenient way to interpret the lower order dynamics and as an extremely stringent test of model fit that goes far beyond validation-error metrics.

With all the required tools in place, quantitative assessment of the modal dynamics is conducted at last in Chapter 9. The chapter considers a series of analyses corresponding to each of the criteria of Chapter 2. To begin, NLSI models are fitted to both the physical and modal coordinates of the nonlinear dynamics from both case studies. It is shown that for both physical and modal displacements, a high-quality SISO representation is possible. Via a generalisation study, it is also shown that the decomposed SISO basis enables approximate generalisation between local excitation levels.

With access now to the HFRFs of the modal dynamics, it is demonstrated that the *underlying* linear components of the nonlinear SISO models of the modal dynamics align closely with the linear modes of the physical system. This result is a promising indication that the inductive biases employed are sufficient to retain a physically-meaningful modal transformation. An interesting ancillary result is that the modal transformation introduces spurious second-order structure. This effect is demonstrated empirically by consideration of the second-order HFRFs of the modal dynamics and analytically for the exact map of a SDOF system in the appendices.

Finally, the superposition performance of the nonlinear transformations is considered. It is observed that the overall map $f^{-1}(f(\mathbf{y}))$ is very close to unity and that the SISO model basis can be projected back onto the physical coordinates without accumulating much error.

Overall, the statistically-independent framework shows significant promise as a nonlinear extension to linear modal analysis, as measured by the extent to which the *useful* aspects of LMA are preserved. In particular:

- The method can be applied in a **data-driven** and **output-only** manner.

- The transformation is shown to permit an **independent**, **high-fidelity** SISO model basis.
- The decomposed SISO modal basis shows **fair generalisation** between local excitation levels.
- The **underlying** linear part of the nonlinear modal dynamics correspond closely to the **true linear modes**.
- The physical displacements can be recovered accurately from the modal contributions by a **nonlinear superposition**.

It cannot be claimed here that the statistically-independent approach is a drop-in replacement for LMA in the case of nonlinearity. Certainly, it is difficult to imagine that any such method could be able to preserve all of the features of the linear analysis. Faced with this challenge, the engineer must seek methods that are able to preserve only the most desirable aspects of a modal decomposition. In this regard, the statistically-independent framework for NNMs shows considerable promise.

10.2 Summary of novel contributions

The novel contributions presented in this thesis are summarised here.

Towards exact solutions to nonlinear ODEs

- Heuristic solution to differential equations is considered as a multi-level search problem, extending the description of Seaton et. al [85].
- The sizes of search spaces corresponding to several expression encoding schemes are enumerated.
- A novel benchmarking suite of ODE problems is proposed that separates the effects of expression size and homogenous ODE problem.
- A novel expression representation scheme is proposed – the affine regression tree.
- A number of mutation operations are proposed for the affine regression tree, including an exact parameter meta-optimisation scheme.
- The new approach is compared to two traditional approaches (tree-based and grammar-based encodings) in a comprehensive benchmark study on two suites of ODE problems.

Statistically independent nonlinear normal modes

- Several criteria are established for the assessment of a practical nonlinear extension to linear modal analysis.
- With a view to promote independence in the modal decomposition, two metrics of monotonicity in the decomposed PSDs are proposed.
- A nonlinear correlation metric (distance correlation), is used within the framework of [22], extending the approach.
- The FPK equation is employed to generate a modal transformations for the SISO case, directly from the equations of motion.
- The HFRFs of the transformed dynamics (via the FPK equation) are constructed in SISO case, it is shown that a non-physical quadratic structure is introduced to the modal dynamics by the mapping.
- NLSI models are fitted to the NNMs (constructed using the neural approach of [23]) in order to assess qualitatively the practical utility of the statistically-independent framework for NNMs.
- The modal dynamics are analysed by extracting HFRFs from the NLSI models of the modal displacements.
- The generalisation performance of the SISO modal representation is investigated by consideration of prediction performance at several excitation levels.

Other contributions

- Closed-form expressions are derived for the HFRFs of an entire class of NLSI models – the stationary kernel-NARX model.
- The utility of the derived expressions for the HFRFs are demonstrated in a case study example for five different choices of the kernel function.

10.3 Further work

Nonlinear system identification

Although consideration of nonlinear system identification was not a principal aim of this thesis, some interesting results have still arisen. In particular, work presented in Chapter

8 has yielded novel closed-form expressions for the HFRFs of an entire class of NLSI model – the stationary kernel-NARX model. These results offer a convenient tool for the engineer to interpret the dynamics of NLSI models trained in an entirely data-driven manner. The HFRFs have also been demonstrated to be an extremely stringent[1] test of model adherence when the true dynamics are known; thus opening the door to more robust analysis of generalisation.

Some limitations of the derived expressions persist. Some avenues for future investigation can therefore be considered. For one, the expressions derived currently rely on the distance metric in the stationary kernel to be the $\mathcal{L}_2$ norm. Further work might investigate alternative metrics. Furthermore, the expressions for the HFRFs are derived for stationary kernels only. Extensions to alternative kernels (based on inner products for example), are therefore of interest.

Although HFRF expressions for other types of model are to be found in the literature [49, 170, 171], these are largely for specific model types and not for generic model classes. Recent interest in neural and other architectures for NLSI motivates the derivation of yet more generic expressions for the HFRFs of other such classes of black-box NLSI models.

Another open problem is computational efficiency. As the order of the HFRF increases, the number of computed points that must be specified increases geometrically. Numerically, this relies on a series of tensor operations that have been (for the purposes of computer algebra assistance), recast as matrix operations. Returning to a full tensor formulation of the Volterra series expansion could open the door to significant computational advantages.

Exact solutions of nonlinear ODEs

In this thesis, a heuristic search approach to identifying exact solutions to nonlinear differential equations was presented. Although the results of the benchmark investigation were mixed, the author envisages several directions for future investigation. A common finding throughout the research presented here was that, during a heuristic search, it is the *complexity* of the solution that controls the search difficulty rather than the form of the differential equation itself. This theme runs opposite to traditional approaches to solving differential equations that rely on canonical forms or computer algebra assistance. This observation motivates the need to identify methods that are able to sparsely represent solutions in the search space. The heuristic metric of locality is a related topic

[1] Far more so than a mean-squared error statistic on validation data.

[134] in evolutionary optimisation from which techniques might be borrowed in order to make progress.

The specification of exact constants in the heuristic search approach is an open problem. The affine regression tree approach presented in this thesis attempts to separate the twin tasks of structure detection and parameterisation. Although several methods for parameter mutation are proposed, consideration of further methods is of interest.

Of all the parameter mutation operations considered in the benchmark investigation of Chapter 4, the constant-snap algorithm showed the greatest potential. the implementation of the method used in this thesis is a very early prototype and a great deal more investigation into the approach is warranted. Of particular interest is the choice of heuristic optimisation algorithm (here simulated annealing was used but it is clear that many such algorithms would be appropriate[2]), and the 'snapping' method. For the latter, the field of integer-relation algorithms [178] may provide a coherent method of specifying constants in terms of rational and transcendental objects.

Another interesting observation is that the structure of parameters in the affine representation is largely sparse for most differential equation problems. Most of the a coefficients are seen to be equal to unity and many of the b coefficients equal to zero. It was shown in Chapter 4 that performance increases as these values are initialised close to these sparse values. It is of interest then, to consider methods that might take further advantage of this apparent sparsity.

A related phenomena (noticed also by Seaton [85]), is that the action of the differential operator on composed functions (i.e the chain rule for differentiation), tends to lead to repeated sub-structures in the form of the solution (for example the natural frequency appears in both the decaying and harmonic terms for a damped linear oscillator). In [85], this structure was exploited using Cartesian genetic programming (CGP) [143, 179]. Further effort would be well spent in assessing if this type of repeated structure could be incorporated into the affine regression tree framework.

Statistically independent nonlinear normal modes

The statistically-independent framework for nonlinear modal analysis has been shown in this thesis to offer a practical nonlinear extension to linear modal analysis. Despite the promising results, several limitations of the method remain. So far, the method has only been applied in a SIMO setting. Although this is a common scenario in practice, it is unrealistic to not consider the fully MIMO case. Modal testing in practice can, for

[2]There is no `https://github.com/MDCHAMP/FreeLunch` in optimisation after all [177].

the linear case, be done in a SIMO manner and then use the principal of reciprocity to obtain a full model of the dynamics. Regrettably, the principle of reciprocity cannot be relied upon in the nonlinear case [2]. As such, it is of significant interest to consider the application of the method in the MIMO case.

A similar consideration is the effect of non-Gaussian input signals. In Chapter 6, the analysis from the FPK equation relies on a Gaussian excitation signal. Although many real-world loading scenarios might be approximately Gaussian, there remains a significant number that are patently non-Gaussian. Extensions to the method should therefore be envisaged that are able to accommodate other excitation types. A first port of call in this direction might be to investigate how the performance of the method degrades as the input is driven further and further from a Gaussian distribution. This approach would assess robustness of the method and may also give insights into what a non-Gaussian modal decomposition might look like.

The application of the method in a practical setting is also of interest. A campaign of nonlinear testing on real-world engineering structures with nonlinearity would provide a solid demonstration of the practical utility of the approach.

One area in which the statistically-independent framework lags behind other NNM approaches is visual interpretation. The modes of the Rosenberg type are neatly summarised by the frequency-energy plot. Those of Shaw and Pierre are conveniently visualised (for 2-DOF systems), by the invariant manifolds upon which the modal dynamics persist. In order for the method to offer practical interpretability, similar methods of visualisation should be sought for the statistically-independent framework.

The data-driven approach of the statistically-independent framework is remarkable, in that the modes are constructed form the data and inductive biases only. Although at this stage no formal equivalence between NNMs from the present approach and those of traditional approaches can be claimed, it is very interesting to imagine which inductive biases would be required to align the approach with that of Rosenberg or Shaw and Pierre. In [96, 109], a deep-learning data-driven approach is presented that constructs modes aligned with both such frameworks. It is of particular interest to investigate how the inductive biases used in that study might be included within the statistically-independent framework.

Modal invariance is defined here as the property of linear modes that means they are independent of the level (or type) of excitation. This property is unavailable in the nonlinear case and so approaches to nonlinear modal analysis must either ignore or explicitly model the energy dependence of the modes. In this work, the SISO modal basis is shown to generalise approximately between local excitation levels. However, the

explicit parameterisation of the energy level in the system is an interesting avenue for investigation. This parameterisation could be implemented within the neural architecture, or by consideration of velocities in the decompositions as in the Shaw and Pierre framework.

A final point for consideration is the full decoupling of the nonlinear dynamics. Consider the form of the decoupled equations of motion in the linear case,

$$\Phi^{\mathsf{T}} M \Phi \ddot{\mathbf{u}} + \Phi^{\mathsf{T}} K \Phi \mathbf{u} = \Phi^{\mathsf{T}} \mathbf{x}(\mathbf{t}) \tag{10.1}$$

Notice that the input has also been transformed for the EOMs to be fully decoupled. Although this might not affect the SIMO systems considered thus far, it might be the case that in the general MIMO formulation it will also be required to take a nonlinear transformation on the input as well as the output for the dynamics to be fully decoupled.

It is also interesting to envisage applications in which the principal research directions from this thesis might be combined. For this, the author imagines two possible approaches.

One possible direction is a slight relaxation of the requirement that the symbolic regression scheme should find exact solutions. Instead, the heuristic search could target a decoupled set of approximate solutions to a nonlinear functional such that they represent a useful (as per the criteria of Chapter 2) set of 'modal' SISO solutions to the coupled nonlinear SODE of interest. The decoupled solutions may be targeted in a number of ways. For example, an emphasis might be placed on compact solutions (by some limitation on tree dimensions) or near-linear equations (by imposing a prior structure on the trees). Access to closed-form expressions for approximate decoupled solutions to a nonlinear ODE may open the door for greater understanding of nonlinear phenomena in structural dynamics and permit insight into new methods for performing nonlinear decomposition and reduced-order modelling.

Another enticing possibility is to target the nonlinear decomposition function f with the methodology developed in this thesis for symbolic regression. In this way, the nonlinear function may be specified in closed-form directly from the collected nonlinear data. Access to a closed-form expression for the transformation may offer key insights into which biases are required to produce a practical nonlinear decomposition. This approach could also be applied to other frameworks for the simplification of nonlinear SODEs such as normal form analysis and Poincaré problems of exact linearisations.

Epilogue

Must find that needle,

that elephant in the zoo;

but do they exist?

Appendix A

Derivations

A.1 Harmonic probing of a quadratic-cubic SIMO Duffing oscillator

The derivations presented in this section follow the results derived in [168] 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 k^{th}-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,

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

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,

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

A.2 Harmonic probing of the NNM of a SISO Duffing oscillator

In Chapter 6 the following SISO equation of motion was considered.

$$\ddot{y} + c\dot{y} + ky + \epsilon y^3 = x(t) \tag{A.18}$$

Subjecting the dynamics of the above to a general forward modal transformation $u = f(y)$, with inverse given by $y = g(u)$, it was shown that the equation of motion in terms of the modal coordinate u is,

$$g''(u)\dot{u}^2 + \ddot{u}g'(u) + cg'(u)\dot{u} + kg(u) + \epsilon g(u)^3 = x(t) \tag{A.19}$$

Now expanding the inverse modal transformation out as a Taylor series,

$$g(u) \approx a_0 + a_1 u + a_2 u^2 + a_3 u^3 \dots \tag{A.20}$$

where the a_i are the coefficients of the Taylor series expansion of $g(u)$. In the analysis that follow, it is assumed that $a_0 = 0$ as the constant offset term in the map will produce infinitely many terms in the HFRF expressions. The author does not view this a limitation of the results however as $g(0) = 0$ (and therefore $a_0 = 0$) was in fact a requirement of the analysis in Chapter 6.

Considering terms now up to third order[1] and substituting the usual probing inputs for $H_1(\omega)$ results in a large expression from which the coefficient of $e^{i\omega t}$ can be extracted to yield,

$$H_1(\omega) = iH_1(\omega)ca_1\omega + 2ia_2\omega H_1(\omega) - H_1(\omega)a_1\omega^2 + H_1(\omega)ka_1 \tag{A.21}$$

Which can be solved for H_1 to reveal,

$$H_1(\omega) = \frac{1}{2a_2 i\omega + a_1(ci\omega - \omega^2 + k)} \tag{A.22}$$

[1] As the reader is probably well aware at this stage, the series truncation can be done with no loss to the generality to the expressions derived here, owing to the extraction of low-order harmonics from the general output of the Volterra series.

Note in the above that for the case $a_1 = 1$ and $a_2 = 0$ (i.e unit linearity and no quadratic component), the underlying linear dynamics of the system are exactly preserved. In the case that there is a quadratic component to the nonlinear inverse map, this acts to bias the damping only, the biased damping can be expressed,

$$c^* = a_1 c + 2a_2 \tag{A.23}$$

The underlying natural frequency is unaffected.

Continuing in the usual fashion for $H_2(\omega_1, \omega_2)$, with some effort[2], the result can be found as,

$$H_2(\omega_1, \omega_2) = H_1(\omega_1)H_1(\omega_2)H_1(\omega_1+\omega_2)\left[a_2(ci(\omega_1+\omega_2) - \omega_1^2 - \omega_2^2 + k) + 3a_3 i(\omega_1+\omega_2)\right] \tag{A.24}$$

An interesting observation from the above is that even if $a_2 = 0$, structure is introduced in the H_2. The higher order HFRFs can be accessed in the familiar manner, this is left as an exercise (in futility) to the reader.

[2] And with some assistance from a computer algebra solver! [68]

Appendix B

Publications

Listed here are the publications arising from work completed during the PhD leading to this thesis. Many of the results presented in this thesis can be found in the publications below.

Journal papers

M. D. Champneys, G. Tsialiamanis, T. J. Rogers, N. Dervilis, K. Worden (2022). On the dynamic properties of statistically-independent nonlinear normal modes. *Mechanical Systems and Signal Processing*, 181, 109510.

G. Tsialiamanis, **M. D. Champneys**, N. Dervilis, J. D. Wagg, K. Worden (2022). On the application of generative adversarial networks for nonlinear modal analysis. *Mechanical Systems and Signal Processing*, 166, 108473.

M. D. Champneys, A. Green, J. Morales, M. Silva, D. Mascarenas (2021). On the vulnerability of data-driven structural health monitoring models to adversarial attack. *Structural Health Monitoring*, 20(4), 1476-1493.

Conference papers

M. D. Champneys, G. Tsialiamanis, T. J. Rogers, N. Dervilis, K. Worden (2023). Towards Exact Statistically Independent Nonlinear Normal Modes via the FPK Equation. (In press)

M. D. Champneys, G. Tsialiamanis, T. J. Rogers, N. Dervilis, K. Worden (2023). On higher-order frequency response functions from kernel-NARX methods. (In press)

J. D. Longbottom, **M. D. Champneys**, E. J. Cross, U. T. Tygesen, T. J. Rogers (2023). Probabilistic numerical integration and sequential Monte Carlo for online identification of nonlinear dynamic systems. (In press)

G. Tsialiamanis, **M. D. Champneys**, J. D. Wagg, N. Dervilis, K. Worden (2023). On the Use of Cycle-Consistent Generative Adversarial Networks for Nonlinear Modal Analysis. In *Topics in Modal Analysis and Parameter Identification*, Volume 8 (pp. 45-57). Springer, Cham.

M. D. Champneys, G. Tsialiamanis, T. J. Rogers, N. Dervilis, K. Worden (2023). On Modelling Statistically Independent Nonlinear Normal Modes with Gaussian Process NARX Models. In *Nonlinear Structures and Systems*, Volume 1 (pp. 135-147). Springer, Cham.

M. D. Champneys, N. Dervilis, K. Worden (2022). On Affine Symbolic Regression Trees for the Solution of Functional Problems. In *Nonlinear Structures and Systems*, Volume 1 (pp. 95-108). Springer, Cham.

M. D. Champneys, K. Worden, N. Dervilis (2019). Nonlinear modal analysis based on complete statistical independence. In *Nonlinear Vibrations, Localization And Energy Transfer*, 4.

Appendix C

Additional projects

As well as the results presented in this thesis, the author has contributed work to a number of other projects which are summarised here.

Adversarial machine learning and structural health monitoring

In the first year of the PhD the author was fortunate enough to have the opportunity to visit Los Alamos national laboratories (LANL) and collaborate with researchers there over the course of four weeks in the autumn of 2019. During that time the author was able to collaborate on a number of interesting projects.

Most notably, a body of work was completed on the vulnerability of data-driven algorithms in the field of Structural Health Monitoring (SHM) to adversarial attack. The work applied some of the emerging trends in the computer science and machine learning communities to a SHM framework and found that even in the naïve case there were significant vulnerabilities and life-safety implications. The work went on to develop a threat model pertaining directly to adversarial attacks in SHM applications.

The principle output of the collaboration was a journal paper [180] written by the author and collaborators at the lab which was published in the journal of structural health monitoring in May 2020.

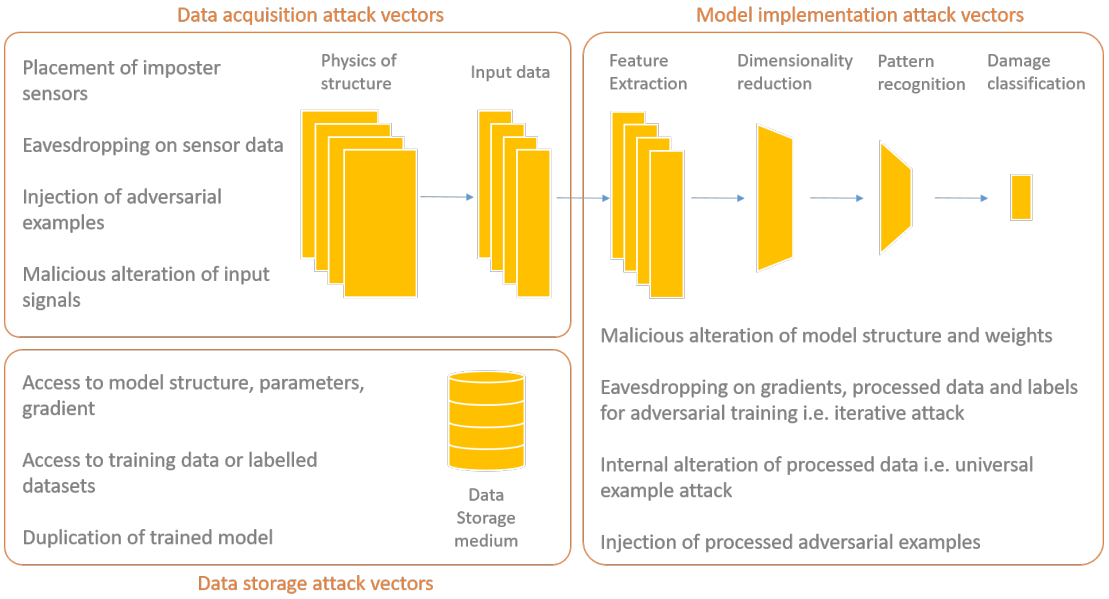


FIGURE C.1: Consideration of attack vectors from the work on adversarial robustness.

Meta-heuristic optimisation

As well as academic outputs, the author has also invested time in the development of an open-source library of meta-heuristic optimisation algorithms developed for the python language. The package features several common algorithms for heuristic optimisation including simulated annealing, differential evolution and particle swarm methods. The library is developed with ease of use and extensibility in mind and has been used for several of the authors projects as well as projects of colleagues and others. The package also supports benchmarking and parallel optimisations for significant computational advantage.

The FreeLunch package is freely available on the python package index `https://pypi.org/project/freelunch/`.

Probabilistic parameter estimation

In the fourth year of the doctoral program, the author collaborated with colleagues from the dynamics research group (DRG) on a probabilistic parameter identification scheme that incorporates a full account of the uncertainty contingent in the time integration step. The approach employed recent results from the field of probabilistic numerics [63] in order to identify parameters of a nonlinear hysteretic Bouc-Wen system. The results of the project were presented by colleagues at both the workshop on nonlinear system identification and uncertainty quantification in structural dynamics conferences.

Bibliography

[1] D. J. Ewins. *Modal Testing: Theory, Practice and Application*. 2009.

[2] K. Worden and G. R. Tomlinson. *Nonlinearity in Structural Dynamics: Detection, Identification and Modelling*. IOP, 2001.

[3] J. Guckenheimer and P. Holmes. *Nonlinear oscillations, dynamical systems, and bifurcations of vector fields*, volume 42. Springer Science & Business Media, 2013.

[4] T. Parker and L. Chua. Chaos: A tutorial for engineers. *Proceedings of the IEEE*, 1987.

[5] J. F. Rhoads, S. W. Shaw, and K. L. Turner. Nonlinear dynamics and its applications in micro-and nanoresonators. *Journal of dynamic systems, measurement, and control*, 2010.

[6] F. Cottone, H. Vocca, and L. Gammaitoni. Nonlinear energy harvesting. *Physical review letters*, 2009.

[7] R. Bouc. Forced vibrations of mechanical systems with hysteresis. In *Proc. of the Fourth Conference on Nonlinear Oscillations*, Prague, 1967.

[8] B. Budiansky and J. W. Hutchinson. A survey of some buckling problems. *Journal of Spacecraft and Rockets*, 2003.

[9] A. Gomes and J. Appleton. Nonlinear cyclic stress-strain relationship of reinforcing bars including buckling. *Engineering Structures*, 1997.

[10] M. N. Chatzis and A. W. Smyth. Robust modeling of the rocking problem. *Journal of Engineering Mechanics*, 2012.

[11] H. Ozdemir. Nonlinear transient dynamic analysis of yielding structures, 1976.

[12] J. Noël and G. Kerschen. Nonlinear system identification in structural dynamics: 10 more years of progress. *Mechanical Systems and Signal Processing*, 2017.

[13] G. Kerschen, K. Worden, A. F. Vakakis, and J.-C. Golinval. Past, present and future of nonlinear system identification in structural dynamics. *Mechanical Systems and Signal Processing*, 2006.

[14] R. T. Q. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud. Neural Ordinary Differential Equations. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[15] A. M. Jaffe. The millennium grand challenge in mathematics. *Notices of the AMS*, 2006.

[16] P. Constantin and C. Foias. *Navier-Stokes equations*. University of Chicago Press, 1988.

[17] B. Peeters, H. Van Der Auweraer, P. Guillaume, and J. Leuridan. The polymax frequency-domain method: A new standard for modal parameter estimation? *Shock and Vibration*, 2004.

[18] R. M. Rosenberg. The normal modes of nonlinear n-degree-of-freedom systems. *Journal of Applied Mechanics*, 1962.

[19] S. W. Shaw and C. Pierre. Normal modes for non-linear vibratory systems. *Journal of Sound and Vibration*, 1993.

[20] B. Feeny and R. Kappagantu. On the physical interpretation of proper orthogonal modes in vibrations. *Journal of Sound and Vibration*, 1998.

[21] T. J. Rogers. *Towards Bayesian System Identification: With Application to SHM of Offshore Structures*. PhD thesis, The University of Sheffield, 2019.

[22] K. Worden and P. L. Green. A machine learning approach to nonlinear modal analysis. *Mechanical Systems and Signal Processing*, 2017.

[23] G. Tsialiamanis, M. D. Champneys, N. Dervilis, D. J. Wagg, and K. Worden. On the application of generative adversarial networks for nonlinear modal analysis. *Mechanical Systems and Signal Processing*, 2022.

[24] J. D. Hamilton. Chapter 50 state-space models. In *Handbook of Econometrics*. Elsevier, 1994.

[25] J. P. Noël and G. Kerschen. 10 years of advances in nonlinear system identification in structural dynamics: A review. In *Proceedings of ISMA 2016-International Conference on Noise and Vibration Engineering*, 2016.

[26] L. Ljung. Perspectives on system identification. *Annual Reviews in Control*, 2010.

[27] J. Decuyper, P. Dreesen, J. Schoukens, M. C. Runacres, and K. Tiels. Decoupling multivariate polynomials for nonlinear state-space models. *IEEE Control Systems Letters*, 2019.

[28] V. Cerone, V. Razza, and D. Regruto. A unified framework for the identification of a general class of multivariable nonlinear block-structured systems. *International Journal of Robust and Nonlinear Control*, 2021.

[29] A. Chiuso and G. Pillonetto. System identification: A machine learning perspective. *Annual Review of Control, Robotics, and Autonomous Systems*, 2019.

[30] S. A. Billings. *Nonlinear System Identification*. John Wiley & Sons, Ltd, 2013.

[31] M. Schoukens and K. Worden. Evolutionary identification of block-structured systems. In *Dynamics of Coupled Structures, Volume 4*. Springer, 2017.

[32] K. Worden, R. Barthorpe, E. Cross, N. Dervilis, G. Holmes, G. Manson, and T. Rogers. On evolutionary system identification with applications to nonlinear benchmarks. *Mechanical Systems and Signal Processing*, 2018.

[33] M. J. Korenberg and I. W. Hunter. The identification of nonlinear biological systems: Wiener kernel approaches. *Annals of Biomedical Engineering*, 1990.

[34] J. Noël and M. Schoukens. Cross-fertilising research in nonlinear system identification between the mechanical, control and machine learning fields. *Mechanical Systems and Signal Processing*, 2019.

[35] J. Schoukens, M. Vaes, and R. Pintelon. Linear system identification in a nonlinear setting: Nonparametric analysis of the nonlinear distortions and their impact on the best linear approximation. *IEEE Control Systems*, 2016.

[36] Y. Cheng and J. Hu. Nonlinear system identification based on SVR with quasi-linear kernel. In *The 2012 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2012.

[37] A. Mauroy and J. Goncalves. Koopman-based lifting techniques for nonlinear systems identification. *IEEE Transactions on Automatic Control*, 2020.

[38] H. Unbehauen and G. Rao. Continuous-time approaches to system identification—a survey. *Automatica*, 1990.

[39] D. Coca and S. A. Billings. Continuous-time system identification for linear and nonlinear systems using wavelet decompositions. *International Journal of Bifurcation and Chaos*, 1997.

[40] S. Billings and H.-L. Wei. A new class of wavelet networks for nonlinear system identification. *IEEE Transactions on Neural Networks*, 2005.

[41] V. Volterra and E. T. Whittaker. *Theory of Functionals and of Integral and Integro-differential Equations*. Dover publications, 1959.

[42] K. Worden, I. Antoniadou, O. Tiboaca, G. Manson, and R. Barthorpe. Linear and nonlinear system identification using evolutionary optimisation. In *Simulation-Driven Modeling and Optimization*. Springer, 2016.

[43] K. Worden and G. Manson. On the identification of hysteretic systems. part i: Fitness landscapes and evolutionary identification. *Mechanical Systems and Signal Processing*, 2012.

[44] K. Worden, O. Tiboaca, I. Antoniadou, and R. Barthorpe. System identification of an mdof experimental structure with a view towards validation and verification. In *Topics in Modal Analysis, Volume 10*. Springer, 2015.

[45] J. Ching, J. L. Beck, and K. A. Porter. Bayesian state and parameter estimation of uncertain dynamical systems. *Probabilistic Engineering Mechanics*, 2006.

[46] T. J. Rogers, T. B. Schön, A. Lindholm, K. Worden, and E. J. Cross. Identification of a Duffing oscillator using particle Gibbs with ancestor sampling. *Journal of Physics: Conference Series*, 2019.

[47] S. A. Billings, H. B. Jamaluddin, and S. Chen. Properties of neural networks with applications to modelling non-linear dynamical systems. *International Journal of Control*, 1992.

[48] J. Kocijan. *Modelling and control of dynamic systems using Gaussian process models*. Springer, 2016.

[49] K. Worden, W. E. Becker, T. J. Rogers, and E. J. Cross. On the confidence bounds of Gaussian process NARX models and their higher-order frequency response functions. *Mechanical Systems and Signal Processing*, 2018.

[50] T. Bohlin. A case study of grey box identification. *Automatica*, 1994.

[51] J. P. Noël and J. Schoukens. Grey-box state-space identification of nonlinear mechanical vibrations. *International Journal of Control*, 2017.

[52] T. Rogers, G. Holmes, E. Cross, and K. Worden. On a grey box modelling framework for nonlinear system identification. In *Special Topics in Structural Dynamics, Volume 6*. Springer, 2017.

[53] E.-W. Bai. Non-parametric nonlinear system identification: A data-driven orthogonal basis function approach. *IEEE Transactions on Automatic Control*, 2008.

[54] E.-W. Bai. Non-parametric nonlinear system identification: An asymptotic minimum mean squared error estimator. *IEEE Transactions on Automatic Control*, 2010.

[55] B. Peeters and G. de Roeck. Reference-based stochastic subspace identification for output-only modal analysis. *Mechanical Systems and Signal Processing*, 1999.

[56] N. Chopin, P. E. Jacob, and O. Papaspiliopoulos. SMC squared: an efficient algorithm for sequential analysis of state space models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 2012.

[57] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. 2006.

[58] G. Duffing. *Erzwungene schwingungen bei veränderlicher eigenfrequenz und ihre technische bedeutung. Braunschweig*. Vieweg & Sohn, 1918.

[59] P. F. Byrd and M. D. Friedman. *Handbook of Elliptic Integrals for Engineers and Scientists*. Springer Berlin Heidelberg, 1971.

[60] G. Peano. Démonstration de l'intégrabilité des équations différentielles ordinaires. *Mathematische Annalen*, 1890.

[61] D. Betounes. *Differential equations: Theory and applications*. Springer Science & Business Media, 2009.

[62] H. Kersting and P. Hennig. Active uncertainty calibration in bayesian ode solvers. 2016.

[63] O. Teymur, H. C. Lie, T. Sullivan, and B. Calderhead. Implicit Probabilistic Integrators for ODEs. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[64] P. J. Davis and P. Rabinowitz. *Methods of Numerical Integration*. Dover Publications, 2007.

[65] A. H. Nayfeh, C.-M. Chin, and J. Pratt. Perturbation methods in nonlinear dynamics—applications to machining dynamics. *Journal of Manufacturing Science and Engineering*, 1997.

[66] R. Mickens. Comments on the method of harmonic balance. *Journal of Sound and Vibration*, 1984.

[67] W. R. Inc. Mathematica, Version 12.0. Champaign, IL, 2019.

[68] Maplesoft, a division of Waterloo Maple Inc.. Maple.

[69] A. Meurer, C. P. Smith, M. Paprocki, O. Čertík, S. B. Kirpichev, M. Rocklin, A. Kumar, S. Ivanov, J. K. Moore, S. Singh, T. Rathnayake, S. Vig, B. E. Granger, R. P. Muller, F. Bonazzi, H. Gupta, S. Vats, F. Johansson, F. Pedregosa, M. J. Curry, A. R. Terrel, v. Roučka, A. Saboo, I. Fernando, S. Kulal, R. Cimrman, and A. Scopatz. Sympy: symbolic computing in python. *PeerJ Computer Science*, 2017.

[70] J. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT press, 1992.

[71] L. Guillaume and F. Charton. Deep learning for symbolic mathematics. In *Proceedings of the 2019 International Conference on Learning Representations*, 2019.

[72] Y. Jin, W. Fu, J. Kang, J. Guo, and J. Guo. Bayesian symbolic regression. *arXiv preprint arXiv:1910.08892*, 2019.

[73] B. Howley. Genetic programming of near-minimum-time spacecraft attitude maneuvers. In *Proceedings of the 1st annual conference on genetic programming*, 1996.

[74] H. Iba and T. Sasaki. Using genetic programming to predict financial data. In *Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406)*, volume 1. IEEE, 1999.

[75] B. McKay, M. Willis, and G. Barton. Steady-state modelling of chemical process systems using genetic programming. *Computers & chemical engineering*, 1997.

[76] P. Orzechowski, W. La Cava, and J. H. Moore. Where are we now? a large benchmark study of recent symbolic regression methods. In *Proceedings of the Genetic and Evolutionary Computation Conference*, 2018.

[77] E. J. Vladislavleva, G. F. Smits, and D. Den Hertog. Order of nonlinearity as a complexity measure for models generated by symbolic regression via pareto genetic programming. *IEEE Transactions on Evolutionary Computation*, 2008.

[78] A. Moraglio, K. Krawiec, and C. G. Johnson. Geometric semantic genetic programming. In *International Conference on Parallel Problem Solving from Nature*. Springer, 2012.

[79] D. Diver. Applications of genetic algorithms to the solution of ordinary differential equations. *Journal of Physics A: Mathematical and General*, 1993.

[80] G. Burgess. Finding approximate analytic solutions to differential equations using genetic programming. Technical report, Defense science and technology organisation, Canbera, Australia, 1999.

[81] S. Kirstukas, K. Bryden, and D. Ashlock. A hybrid genetic programming approach for the analytical solution of differential equations. *International Journal of General Systems*, 2005.

[82] H. Cao, L. Kang, Y. Chen, and J. Yu. Evolutionary modeling of systems of ordinary differential equations with genetic programming. *Genetic Programming and Evolvable Machines*, 2000.

[83] I. Tsoulos and I. Lagaris. Solving differential equations with genetic programming. *Genetic Programming and Evolvable Machines*, 2006.

[84] L. B. Rall and R. LB. *Automatic differentiation: Techniques and applications*. Springer, 1981.

[85] T. Seaton, G. Brown, and J. Miller. Analytic solutions to differential equations under graph-based genetic programming. In *European Conference on Genetic Programming*. Springer, 2010.

[86] A. Sóbester, P. B. Nair, and A. J. Keane. Genetic programming approaches for solving elliptic partial differential equations. *IEEE transactions on evolutionary computation*, 2008.

[87] L. Mex, C. Cruz-Villar, and F. Peñuñuri. Closed-form solutions to differential equations via differential evolution. *Discrete Dynamics in Nature and Society*, 2015.

[88] W. Lobão, D. Dias, and M. Pacheco. Genetic programming and automatic differentiation algorithms applied to the solution of ordinary and partial differential equations. In *2016 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 2016.

[89] W. J. de Araujo Lobão, M. A. C. Pacheco, D. M. Dias, and A. C. A. Abreu. Solving stochastic differential equations through genetic programming and automatic differentiation. *Engineering Applications of Artificial Intelligence*, 2018.

[90] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[91] E. Davis. The use of deep learning for symbolic integration: A review of (lample and charton, 2019). *arXiv preprint arXiv:1912.05752*, 2019.

[92] G. Stépán. Modelling nonlinear regenerative effects in metal cutting. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 2001.

[93] C. R. Farrar and K. Worden. *Structural Health Monitoring: A Machine Learning Perspective*. John Wiley & Sons, 2012.

[94] G. Kerschen, M. Peeters, J.-C. Golinval, and A. F. Vakakis. Nonlinear normal modes, part I: A useful framework for the structural dynamicist. *Mechanical Systems and Signal Processing*, 2009.

[95] G. I. Cirillo, A. Mauroy, L. Renson, G. Kerschen, and R. Sepulchre. Global parametrization of the invariant manifold defining nonlinear normal modes using the koopman operator. In *Volume 6: 11th International Conference on Multibody Systems, Nonlinear Dynamics, and Control*. American Society of Mechanical Engineers, 2015.

[96] A. Rostamijavanani, S. Li, and Y. Yang. A Study on Data-Driven Identification and Representation of Nonlinear Dynamical Systems with a Physics-Integrated Deep Learning Approach: Koopman Operators and Nonlinear Normal Modes. In M. R. Brake, L. Renson, R. J. Kuether, and P. Tiso, editors, *Nonlinear Structures & Systems, Volume 1*, Conference Proceedings of the Society for Experimental Mechanics Series, Cham, 2023. Springer International Publishing.

[97] A. F. Vakakis, L. I. Manevitch, Y. V. Mikhlin, V. N. Pilipchuk, and A. A. Zevin. *Normal modes and localization in nonlinear systems*. Springer, 2001.

[98] C. Touzé. *A normal form approach for nonlinear normal modes*. PhD thesis, LMA, 2003.

[99] A. H. Nayfeh, C. Chin, and S. A. Nayfeh. On nonlinear normal modes of systems with internal resonance. *Journal of Vibration and Acoustics*, 1996.

[100] G. Haller and S. Ponsioen. Nonlinear normal modes and spectral submanifolds: existence, uniqueness and use in model reduction. *Nonlinear Dynamics*, 2016.

[101] R. Rosenberg. On nonlinear vibrations of systems with many degrees of freedom. In *Advances in applied mechanics*, volume 9. Elsevier, 1966.

[102] Y. S. Lee, G. Kerschen, A. F. Vakakis, P. Panagopoulos, L. Bergman, and D. M. McFarland. Complicated dynamics of a linear oscillator with a light, essentially nonlinear attachment. *Physica D: Nonlinear Phenomena*, 2005.

[103] M. Krack. Nonlinear modal analysis of nonconservative systems: extension of the periodic motion concept. *Computers & Structures*, 2015.

[104] L. Renson and G. Kerschen. Nonlinear Normal Modes of Nonconservative Systems. In G. Kerschen, D. Adams, and A. Carrella, editors, *Topics in Nonlinear Dynamics, Volume 1*, Conference Proceedings of the Society for Experimental Mechanics Series, New York, NY, 2013. Springer.

[105] M. Peeters, R. Viguié, G. Sérandour, G. Kerschen, and J.-C. Golinval. Nonlinear normal modes, part II: Toward a practical computation using numerical continuation techniques. *Mechanical systems and signal processing*, 2009.

[106] L. Renson, G. Kerschen, and B. Cochelin. Numerical computation of nonlinear normal modes in mechanical engineering. *Journal of Sound and Vibration*, 2016.

[107] J. C. Slater. A numerical method for determining nonlinear normal modes. *Nonlinear Dynamics*, 1996.

[108] J.-P. Noël, L. Renson, C. Grappasonni, and G. Kerschen. Identification of nonlinear normal modes of engineering structures under broadband forcing. *Mechanical Systems and Signal Processing*, 2016. Special Issue in Honor of Professor Simon Braun.

[109] S. Li and Y. Yang. Data-driven identification of nonlinear normal modes via physics-integrated deep learning. *Nonlinear Dynamics*, 2021.

[110] G. Kerschen, M. Peeters, J. C. Golinval, and C. Stéphan. Nonlinear modal analysis of a full-scale aircraft. *Journal of Aircraft*, 2013.

[111] A. F. Vakakis. *Analysis and identification of linear and nonlinear normal modes in vibrating systems.* PhD thesis, California Institute of Technology, United States – California, 1991.

[112] A. Vakakis. Non-linear normal modes (NNMs) and their applications in vibration theory: an overview. *Mechanical Systems and Signal Processing*, 1997.

[113] S. W. Shaw. An invariant manifold approach to nonlinear normal modes of oscillation. *Journal of Nonlinear Science*, 1994.

[114] S. Shaw and C. Pierre. Non-linear normal modes and invariant manifolds. *Journal of sound and Vibration*, 1991.

[115] N. Boivin, C. Pierre, and S. W. Shaw. Non-linear normal modes, invariance, and modal dynamics approximations of non-linear systems. *Nonlinear Dynamics*, 1995.

[116] J. Carr. *Applications of centre manifold theory*, volume 35. Springer Science & Business Media, 2012.

[117] S. W. Shaw and C. Pierre. Normal modes of vibration for non-linear continuous systems. 1994.

[118] E. Pesheck, C. Pierre, and S. Shaw. A new galerkin-based approach for accurate non-linear normal modes through invariant manifolds. *Journal of Sound and Vibration*, 2002.

[119] M. Cenedese, J. Axås, B. Bäuerlein, K. Avila, and G. Haller. Data-driven modeling and prediction of non-linearizable dynamics via spectral submanifolds. *Nature Communications*, 2022.

[120] K. Lu, Y. Jin, Y. Chen, Y. Yang, L. Hou, Z. Zhang, Z. Li, and C. Fu. Review for order reduction based on proper orthogonal decomposition and outlooks of applications in mechanical systems. *Mechanical Systems and Signal Processing*, 2019.

[121] Y. Liang, H. Lee, S. Lim, W. Lin, K. Lee, and C. Wu. Proper orthogonal decomposition and its applications—part i: Theory. *Journal of Sound and Vibration*, 2002.

[122] K. Kunisch and S. Volkwein. Galerkin proper orthogonal decomposition methods for a general equation in fluid dynamics. *SIAM Journal on Numerical analysis*, 2002.

[123] K. Kunisch and S. Volkwein. Galerkin proper orthogonal decomposition methods for parabolic problems. *Numerische mathematik*, 2001.

[124] S. Roberts and R. Everson. *Independent Component Analysis: Principles and Practice*. Cambridge University Press, 2001.

[125] K. Worden and P. L. Green. A machine learning approach to nonlinear modal analysis. In F. N. Catbas, editor, *Dynamics of Civil Structures, Volume 4*, Cham, 2014. Springer International Publishing.

[126] N. Dervilis, T. E. Simpson, D. J. Wagg, and K. Worden. Nonlinear modal analysis via non-parametric machine learning tools. *Strain*, 2018.

[127] M. D. Champneys, K. Worden, and N. Dervilis. Nonlinear modal analysis based on complete statistical independence. In *Nonlinear Vibrations, Localisation and Energy Transfer*, volume 160, 2019.

[128] L. A. Bull, P. A. Gardner, N. Dervilis, and K. Worden. Normalising flows and nonlinear normal modes. In *IFAC-PapersOnLine*, volume 54. Elsevier, 2021.

[129] T. Simpson, G. Tsialiamanis, N. Dervilis, K. Worden, and E. Chatzi. On the use of variational autoencoders for nonlinear modal analysis. In *IMAC 2022*, 2022.

[130] I. G. Tsoulos, D. Gavrilis, and E. Glavas. Solving differential equations with constructed neural networks. *Neurocomputing*, 2009.

[131] D. Howard and S. C. Roberts. Genetic programming solution of the convection-diffusion equation. In *Proceedings of the 3rd Annual Conference on Genetic and Evolutionary Computation*, 2001.

[132] A. Kawamura, H. Ota, C. Rösnick, and M. Ziegler. Computational complexity of smooth differential equations. In *International Symposium on Mathematical Foundations of Computer Science*. Springer, 2012.

[133] R. Poli, W. Langdon, N. McPhee, and J. Koza. *A Field Guide to Genetic Programming*. Lulu Enterprises, 2008.

[134] T. Seaton. *Locality in the Evolutionary Optimisation of Programs*. PhD thesis, University of York, 2013.

[135] F. Rothlauf. Representations for genetic and evolutionary algorithms. In *Representations for Genetic and Evolutionary Algorithms*. Springer, 2006.

[136] P. Bille. A survey on tree edit distance and related problems. *Theoretical computer science*, 2005.

[137] R. Graham, D. Knuth, O. Patashnik, and S. Liu. Concrete mathematics: a foundation for computer science. *Computers in Physics*, 1989.

[138] I. Arnaldo, K. Krawiec, and U. O'Reilly. Multiple regression genetic programming. In *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*, 2014.

[139] H. Iba, H. deGaris, and T. Sato. A numerical approach to genetic programming for system identification. *Evolutionary computation*, 1995.

[140] A. Qin and P. Suganthan. Self-adaptive differential evolution algorithm for numerical optimization. In *2005 IEEE congress on evolutionary computation*, volume 2. IEEE, 2005.

[141] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *science*, 1983.

[142] R. Storn and K. Price. Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 1997.

[143] J. F. Miller and S. L. Harding. Cartesian genetic programming. In *Proceedings of the 10th annual conference companion on Genetic and evolutionary computation*, 2008.

[144] C. E. Shannon. A mathematical theory of communication. *Bell system technical journal*, 1948.

[145] D. J. MacKay and D. J. Mac Kay. *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

[146] X. Huo and G. J. Székely. Fast computing for distance covariance. *Technometrics*, 2016.

[147] N. Dervilis, K. Worden, D. Wagg, and S. Neild. *Simplifying Transformations for Nonlinear Systems: Part I, An Optimisation-Based Variant of Normal Form Analysis*, volume 1. Springer International Publishing, 2015.

[148] N. Dervilis, K. Worden, D. Wagg, and S. Neild. *Simplifying Transformations for Nonlinear Systems: Part II, Statistical Analysis of Harmonic Cancellation*, volume 1. Springer International Publishing, 2015.

[149] T. K. Caughey. Derivation and application of the fokker-planck equation to discrete nonlinear dynamic systems subjected to white random excitation. *Citation: The Journal of the Acoustical Society of America*, 1963.

[150] I. Kobyzev, S. J. Prince, and M. A. Brubaker. Normalizing flows: An introduction and review of current methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[151] J. Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE International Conference on Computer Vision*, 2017.

[152] P. Welch. The use of fast fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms. *IEEE Transactions on Audio and Electroacoustics*, 1967.

[153] R. Eberhart and J. Kennedy. A new optimizer using particle swarm theory. In *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, 1995.

[154] S. Mirjalili. Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. *Knowledge-based systems*, 2015.

[155] D. Karaboga and B. Basturk. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *Journal of Global Optimization*, 2007.

[156] J. Zhang and A. C. Sanderson. Jade: adaptive differential evolution with optional external archive. *IEEE Transactions on evolutionary computation*, 2009.

[157] D. R. Polson. *Assessment of applications of optimisation to building design and energy modelling*. PhD thesis, UCL (University College London), 2020.

[158] M. Neshat, M. M. Nezhad, E. Abbasnejad, S. Mirjalili, D. Groppi, A. Heydari, L. B. Tjernberg, D. A. Garcia, B. Alexander, Q. Shi, *et al.* Wind turbine power output prediction using a new hybrid neuro-evolutionary method. *Energy*, 2021.

[159] A. K. Qin, V. L. Huang, and P. N. Suganthan. Differential Evolution Algorithm With Strategy Adaptation for Global Numerical Optimization. *IEEE Transactions on Evolutionary Computing*, 2009.

[160] S. Das, S. S. Mullick, and P. N. Suganthan. Recent advances in differential evolution–an updated survey. *Swarm and evolutionary computation*, 2016.

[161] C. Ramos-Carreño. dcor: distance correlation and related E-statistics in Python, 2022.

[162] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 2006.

[163] D. P. Kingma and M. Welling. An introduction to variational autoencoders. *Foundations and Trends in Machine Learning*, 2019.

[164] A. Larsen, S. S. Kaae, and O. Winther. Autoencoding beyond pixels using a learned similarity metric. In *International conference on machine learning*. PMLR, 2016.

[165] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, volume 27, 2014.

[166] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[167] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi. A survey of methods for explaining black box models. *ACM computing surveys (CSUR)*, 2018.

[168] S. J. Gifford and G. R. Tomlinson. Recent advances in the application of functional series to non-linear structures. *Journal of Sound and Vibration*, 1989.

[169] K. Worden, G. Manson, and G. Tomlinson. A harmonic probing algorithm for the multi-input volterra series. *Journal of Sound and Vibration*, 1997.

[170] J. E. Chance, K. Worden, and G. R. Tomlinson. Frequency domain analysis of narx neural networks. *Journal of Sound and Vibration*, 1998.

[171] K. Worden, G. Manson, and E. J. Cross. On gaussian process narx models and their higher-order frequency response functions. In *Solving Computationally Expensive Engineering Problems*. Springer, 2014.

[172] M. D. Champneys, G. Tsialiamanis, T. J. Rogers, N. Dervils, and K. Worden. On the dynamic properties of statistically-independent nonlinear normal modes. *Mechanical*, 2022. Manuscript submitted for publication in: *Mechanical Systems and Signal Processing*.

[173] S. A. Billings and K. M. Tsang. Spectral analysis for non-linear systems, part i: Parametric non-linear spectral analysis. *Mechanical Systems and Signal Processing*, 1989.

[174] H. Akaike. Factor analysis and AIC. In *Selected papers of Hirotugu Akaike*. Springer, 1987.

[175] G. Schwarz. Estimating the dimension of a model. *The annals of statistics*, 1978.

[176] J. Sun, C. H. Lai, and X. J. Wu. *Particle Swarm Optimisation: Classical and Quantum Perspectives*. CRC Press, 2016.

[177] D. Wolpert and W. Macready. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1997.

[178] H. Ferguson, D. Bailey, and S. Arno. Analysis of pslq, an integer relation finding algorithm. *Mathematics of Computation*, 1999.

[179] J. F. Miller. Cartesian genetic programming: its status and future. *Genetic Programming and Evolvable Machines*, 2019.

[180] M. D. Champneys, A. Green, J. Morales, M. Silva, and D. Mascarenas. On the vulnerability of data-driven structural health monitoring models to adversarial attack. *Structural Health Monitoring*, 2020.