
Article:
Park, H.-Y. orcid.org/0000-0001-8348-0887, Kwon, J.-W. orcid.org/0000-0003-3467-7805, Kim, H.-L. orcid.org/0000-0001-9091-8787 et al. (6 more authors) (2023) Cost-effectiveness of all-oral regimens for the treatment of multidrug-resistant tuberculosis in Korea: comparison with conventional injectable-containing regimens. Journal of Korean Medical Science, 38 (21). e167. ISSN 1011-8934

https://doi.org/10.3346/jkms.2023.38.e167

Mechanical Systems and Signal Processing 196 (2023) 110300

Contents lists available at ScienceDirect

Mechanical Systems and Signal Processing

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

Identification of piecewise-linear mechanical oscillators via Bayesian model selection and parameter estimation

R. Nayek [a,b], A.B. Abdessalem [c], N. Dervilis [b], E.J. Cross [b,*], K. Worden [b]

[a] Department of Applied Mechanics, Indian Institute of Technology Delhi, Hauz Khas, New Delhi 110016, India
[b] Dynamics Research Group, Department of Mechanical Engineering, University of Sheffield, Mapping Street, Sheffield S1 3JD, UK
[c] University of Angiers, 40 Rue de Rennes, 49100 Angers, France

ARTICLE INFO

Communicated by J. Noël

Keywords:
PWL systems
Approximate Bayesian computation
Model selection
Parameter estimation

ABSTRACT

The problem of identifying single degree-of-freedom (SDOF) nonlinear mechanical oscillators with piecewise-linear (PWL) restoring forces is considered. PWL nonlinear systems are a class of models that specify or approximate nonlinear systems via a set of locally-linear maps, each defined over different operating regions. They are useful in modelling hybrid phenomena common in practical situations, such as, systems with different modes of operation, or systems whose dynamics change because of physical limits or thresholds. However, identifying PWL models can be a challenging task when the number of operating regions and their partitions are unknown. This paper formulates the identification of oscillators with PWL restoring forces as a task of concurrent model selection and parameter estimation, where the selection of the number of linear regions is treated as a model selection task and identifying the associated system parameters as a task of parameter estimation. In this study, PWL maps in restoring forces with up to four regions are considered, and the task of model selection and parameter estimation task is addressed in a Bayesian framework. A likelihood-free Approximate Bayesian Computation (ABC) scheme is followed, which is easy to implement and provides a simplified way of doing model selection. The proposed approach has been demonstrated using two numerical examples and an experimental study, where ABC has been used to select models and identify parameters from among four SDOF PWL systems with different number of PWL regions. The results demonstrate the flexibility of using the proposed Bayesian approach for identifying the correct model and parameters of PWL systems, in addition to furnishing uncertainty estimates of the identified parameters.

1. Introduction

Linear systems are relatively easy to analyse and well-established theories exist for the identification and analysis of such systems. Linearity also means that local properties hold globally. This property, however, is not true for nonlinear systems, which make them hard to analyse. Hence, when confronted with nonlinear systems, a common technique is to approximate the nonlinear system with one or more linear systems. When a single linear approximation about an operating point is not accurate enough, a series of locally linear approximations may be defined over different regions of operation of the nonlinear system. Such a system with a set of locally-linear maps is called a *piecewise-linear* (PWL) system. These systems form an interesting class of models as they feature universal approximation properties [1], meaning that they can approximate any nonlinear function arbitrarily well, and they generalise the well-established theory of linear systems for analysing nonlinear systems.

* Corresponding author.
E-mail address: e.j.cross@sheffield.ac.uk (E.J. Cross).

https://doi.org/10.1016/j.ymssp.2023.110300
Received 13 May 2022; Received in revised form 3 November 2022; Accepted 14 March 2023
Available online 14 April 2023

Systems with PWL dynamics (or more generally, piecewise-affine (PWA) dynamics), have been studied for a long time in science and engineering. They are often used to model a variety of nonlinear phenomena occurring in many practical situations, such as systems with different modes of operation, or systems where the dynamics change due to thresholds, switches, and physical limits. One of the earliest practical implementation of PWL systems lies in control engineering, where a PWL control formulation was introduced in anticipatory actions of a servomechanism [2]. PWL systems are also quite useful in modelling rule-based intelligent controllers such as fuzzy-logic-based control systems [3]. In the 1970s, PWL models grew very popular in the circuit theory community for modelling large-scale electrical circuits with diodes [4], where nonlinear transfer functions of semiconductor diodes (used in analog computers) were realised with piecewise-continuous approximations. PWL models also have a long tradition of being used in modelling systems with saturation and relay nonlinearities [5]. In mechanical and structural engineering, PWL systems are extremely useful in representing oscillators with motion-limiting constraints [6]. Examples of such systems include shock absorbers in automotive suspension systems, where different damping characteristics are observed in compression and rebound, or vibrating components that make soft contact (or impact) with each other, giving rise to PWL stiffnesses [6]. The practical significance of these systems has resulted in several studies involving nonlinear SDOF mechanical oscillators where the restoring forces are modelled as PWL functions of displacement and/or velocity [7–10].

There are several results on the analysis, stability and computation of PWL systems in the literature [11–15], with the implicit assumption that a PWL model is available at hand. A PWL model can be derived from first principles by evaluating an explicit smooth nonlinear model (or a differential equation) at a number of points and using linear interpolation between those points. However, in most practical cases, an explicit system model is not available and hence there arises a need to identify a PWL model from experimental data. Identification of PWL models can be quite challenging depending on what information is known at hand.

The full set of parameters for a PWL model includes: (a) the number of operating regions, (b) the parameters of the partitions demarcating the different operating regions, and (c) the parameters of each local linear model [16]. When the number of operating regions and their partitions are known, parameter estimation of the local linear models is a relatively easy task and can be achieved with standard linear parameter identification methods [17]. However, when the partitions are not known, the problem becomes considerably more challenging, and even more difficult when the number of operating regions is also unknown. For a complete identification of a PWL model, one requires to simultaneously classify the operating regions and estimate the parameters of the partitions as well as those of the linear submodels.

A variety of frequentist and Bayesian approaches for identifying PWL/PWA systems has been proposed in the literature, see [16,18,19] for a review. In most works, a commonly-followed approach is to pose the identification of PWL models in a regression format, in the input–output space; this formatting allows the problem to be set up as an identification of piecewise AutoRegressive with eXogenous inputs (PWARX) models [20–22]. Another way is to treat the identification problem using state–space models [17]. When the number of operating regions is treated as known, the parameters of the partitions and those of the linear submodels can be estimated via optimisation [23,24]. However, the optimisation schemes, like gradient descent or Gauss–Newton search, may get stuck in local optima and an evolutionary optimisation scheme (such as a genetic algorithm), is better suited. As an extension to this problem with a fixed number of operating regions, a few studies proposed to add new partitions (i.e., new sub-regions) progressively using several steps in identification [25–27]. Another group of methods [20,21,28–31] treat the number of operating regions as unknown and start by classifying the data points according to a certain criterion and then estimating the local linear/affine submodels simultaneously or iteratively.

This paper deals with identification of SDOF vibratory oscillators where the restoring forces are piecewise-linear functions of the system displacements or velocities. Unlike the previous approaches that try to identify clusters of data points segregating the linear regions of operation, here the number of linear regions of operation is treated as a task of model selection, whereas that of determining the parameters of linear regions is solved via parameter estimation. Addressing model selection and parameter estimation in a Bayesian paradigm, Bayesian model selection is used to determine the PWL model with the most appropriate number of regions from among a set of PWL models with different number of regions of operation, and Bayesian parameter inference to estimate the distributions over the parameters of the linear regions. For conceptual simplicity, an ABC-based model selection and posterior parameter estimation procedure is followed [32,33]. Unlike previous Bayesian approaches [20–22] that developed specific Bayesian computation algorithms tailored to the formulation of PWARX systems, the ABC procedure can work with any formulation of the model as long as responses can be simulated from the model given the parameters. The simplicity of this approach lies in the relative ease of implementation, where one only needs to specify a set of PWL models with different model orders, i.e. different number of operating regions, and the ABC algorithm automatically determines the most suitable model, as well as estimates the posterior uncertainty over its parameters.

To demonstrate this concept, the ABC-based combined Bayesian model selection and parameter estimation is applied to two numerically-simulated examples of SDOF oscillators, one with PWL stiffness and other with PWL damping nonlinearities. The particular interest in these two types of PWL models is because they are quite useful in representing mechanical systems such as vibration sinks [34], vibration isolators [35], and freeplay nonlinearities that occur because of worn-out hinges and loose rivets [36]. What follows next, in Section 2, is a mathematical description of SDOF oscillators with PWL nonlinearities, specifically PWL stiffness and PWL damping models. Next, in Section 3, the ABC framework for Bayesian model selection and parameter inference is described, which is used for identification of the PWL systems. Section 4 outlines two numerical studies – one on a trilinear stiffness model and another on a trilinear damping model – to demonstrate the working of the proposed methodology. For experimental validation, the proposed procedure is applied to an experimental dataset from an automotive shock absorber in Section 5. Finally, Section 6 provides some discussion and concluding remarks.

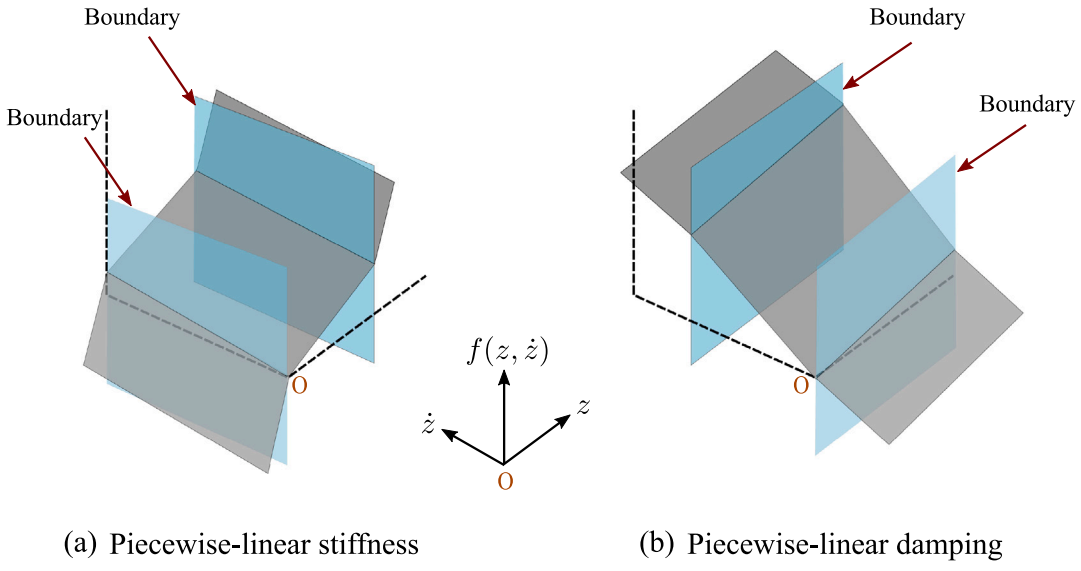


Fig. 1. Diagrammatic sketches of piecewise-linear restoring force surfaces for PWL stiffness and PWL damping models; the state–space is partitioned along displacement z in (a), and along velocity in $\dot{z}$ in (b).

2. Piecewise-linear model description

Consider an externally-forced nonlinear SDOF mechanical oscillator represented by the following equation of motion,

$$m\ddot{z} + f(z, \dot{z}) = u \tag{1}$$

where m is the mass of the oscillator, u is the external forcing function, and $z, \dot{z}, \ddot{z}$ are the displacement, velocity, and acceleration responses, respectively, and $f(z, \dot{z})$ is the internal restoring force due to damping and stiffness, which acts to bring the mass back to equilibrium position when disturbed. The nonlinear restoring force is assumed to be a continuous PWL function of the displacement z or velocity $\dot{z}$ (but not both at the same time), such that the state–space of the oscillator defined by the pair $\mathcal{X} = \{z, \dot{z}\}$, is partitioned into different operating regions $\{\mathcal{X}_p\}_{p=1}^{P}$. Each operating region is associated with a static linear relation describing the restoring force surface and the partition between two adjacent regions forms a plane. The region partitions can be considered as switches between different linear regions, the switches arising from the breakpoints in the PWL functions of the nonlinear restoring force f.

Here the focus is placed on the identification of two types of PWL systems that are quite useful for analysing nonlinear mechanical oscillators. These are PWL stiffness and PWL damping oscillators [6]. They differ from each other in the sense of which state variable – displacement or velocity – of the system admits a predominant PWL representation; their corresponding restoring force surface plots are shown in Fig. 1. When a linear behaviour is observed in the velocity variable $\dot{z}$ and a piecewise-linear behaviour occurs only as a function of the displacement variable z, a PWL stiffness system is obtained. For such a system, the system equation of motion can be compactly represented by,

$$m\ddot{z} + c_0\dot{z} + k_{\text{PWL}}(z) = u, \quad \text{for} \quad \{z, \dot{z}\} \in \mathcal{X}_p \tag{2}$$

Here, c_0 denotes the linear damping parameter and the term $k_{\text{PWL}}(z)$ represents the PWL stiffness function. For such a system, the stiffness-based restoring force $k_{\text{PWL}}(z)$ has P linear operating regions and the regional partitions are planes that occur parallel to the $f - \dot{z}$ plane as shown in Fig. 1(a). Nonlinearities in the form of PWL stiffness can arise in aircraft ground vibration tests from pylon–store–wing assemblies or at pre-loading bearing locations [6]. Freeplay nonlinearity is another form of PWL stiffness that can arise in aeroelastic systems because of worn-out hinges and loose rivets [36]. Contrary to a PWL stiffness system, the restoring force in a PWL damping system is dominated by a PWL function of velocity $\dot{z}$; the equation of motion of such a system would read as,

$$m\ddot{z} + c_{\text{PWL}}(\dot{z}) + k_0 z = u, \quad \text{for} \quad \{z, \dot{z}\} \in \mathcal{X}_p \tag{3}$$

where k_0 is the linear stiffness parameter and $c_{\text{PWL}}(\dot{z})$ represents the PWL damping function. For such a system, it is the damping parameter that varies across different regions of operation, and the regional partitions manifest as planes parallel to the $f - z$ plane as shown in Fig. 1. An example of a PWL damping system is a standard automotive damper or shock absorber which is designed to have different damping constants in compression and rebound [6]. It must be mentioned that piecewise linearity could also occur simultaneously in both displacement and velocity; however, such scenarios are less common in practice and hence not considered in this paper.

It is now worth looking at the piecewise-linear forms of k_{PWL} and c_{PWL}. As mentioned before, a dynamical system with PWL stiffness (or damping), is obtained by partitioning along the displacement (or velocity) field to obtain a finite set of linear operating

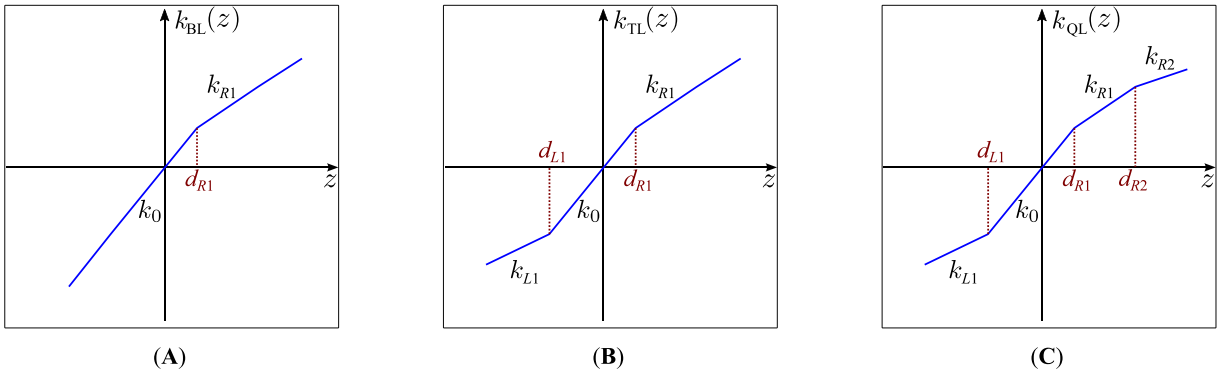


Fig. 2. SDOF mechanical models with (A) bilinear, (B) trilinear, and (C) quadlinear stiffnesses.

regions and then a linear restoring force map is considered for each region. When there exists only a single region of operation, i.e. no partitions, the system is a linear system. A PWL system results when there is more than a single linear region of operation, the number of regions of operation is considered here as the model order of the PWL system. For example, a second-order PWL system will comprise two regions of operation, and is referred to as a *bilinear* system, with $k_{\mathrm{BL}}(z)$ representing a bilinear function of displacement. In this study, PWL systems up to the fourth order are considered, since regions of operation beyond four are rarely seen in practice. Fig. 2 illustrates three PWL mechanical oscillators with bilinear (BL), trilinear (TL), and quadlinear (QL) stiffnesses and depicts their k_{PWL} maps as a function of z. For example, Fig. 2(A) illustrates a BL system with two linear regions of operation separated by a breakpoint, each region characterised by its own linear stiffness parameter; k_0 and k_{R1}, here. The breakpoint between the two adjacent regions of operation represents a displacement partition and its location on the displacement axis is denoted by d_{R1}. For the purpose of convenience, the stiffness and partition parameters are denoted by subscripts $L\#$ or $R\#$ to refer to which side (left or right) of the mass they appear. If the partition for the BL stiffness system appears to the left of the mass (instead of right, as shown in Fig. 2(A)), the stiffness parameter would be denoted by k_{L1} and the partition parameter by d_{L1}. It must be emphasised that two adjacent stiffness and partition parameters for any PWL-stiffness system should be distinct from each another. For example, the two adjacent stiffnesses k_{R1} and k_{R2} of a QL stiffness system (see Fig. 2(C)) cannot be the same, else it would resemble a TL stiffness model. In inverse identification, such redundancies may be avoided by enforcing the adjacent stiffness and partition parameters to be different from each other.

This study considers PWL systems that have up to four linear regions of operation (i.e. fourth order), since the mechanical oscillators that can be approximately modelled as PWL systems usually have less than four regions of operation. For a mechanical oscillator with PWL stiffness, the PWL maps for bilinear (BL), trilinear (TL), and quadlinear (QL) stiffnesses can be described by Eqs. (4), (5), and (6), respectively.

Bilinear stiffness:

$$k_{\mathrm{BL}}(z) = \begin{cases} k_0 z, & \text{for } z \le d_{R1} \\ k_0 d_{R1} + k_{R1}\left(z - d_{R1}\right), & \text{for } z > d_{R1} \end{cases} \tag{4}$$

Trilinear stiffness:

$$k_{\mathrm{TL}}(z) = \begin{cases} k_0 d_{L1} + k_{L1}\left(z - d_{L1}\right), & \text{for } z \le d_{L1} \\ k_0 z, & \text{for } d_{L1} < z \le d_{R1} \\ k_0 d_{R1} + k_{R1}\left(z - d_{R1}\right), & \text{for } z > d_{R1} \end{cases} \tag{5}$$

Quadlinear stiffness:

$$k_{\mathrm{QL}}(z) = \begin{cases} k_0 d_{L1} + k_{L1}\left(z - d_{L1}\right), & \text{for } z \le d_{L1} \\ k_0 z, & \text{for } d_{L1} < z \le d_{R1} \\ k_0 d_{R1} + k_{R1}\left(z - d_{R1}\right), & \text{for } d_{R1} < z \le d_{R2} \\ k_0 d_{R1} + k_{R1}\left(d_{R2} - d_{R1}\right) + k_{R2}\left(z - d_{R2}\right), & \text{for } z > d_{R2} \end{cases} \tag{6}$$

As for oscillators with PWL damping, their mathematical descriptions are very similar to those of the PWL stiffness systems, with the only difference that the PWL regions here are functions of the velocity $\dot{z}$ (instead of displacement z), as described in Eq. (3). Therefore, oscillators with bilinear, trilinear, and quadlinear damping, illustrated by Fig. 3, are described by the following relations.

Bilinear damping:

$$c_{\mathrm{BL}}(\dot{z}) = \begin{cases} c_0 \dot{z}, & \text{for } \dot{z} \le v_{R1} \\ c_0 v_{R1} + c_{R1}\left(\dot{z} - v_{R1}\right), & \text{for } \dot{z} > v_{R1} \end{cases} \tag{7}$$

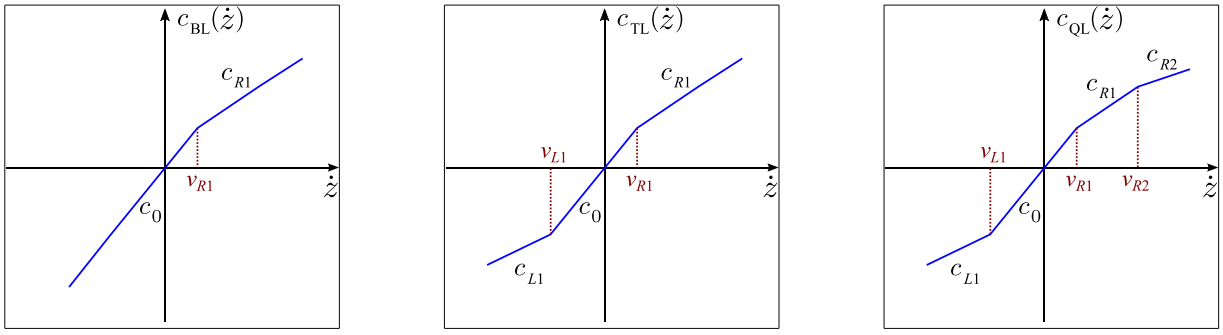


Fig. 3. SDOF mechanical models with (A) bilinear, (B) trilinear, and (C) quadlinear damping.

Trilinear damping:

$$c_{\mathrm{TL}}(\dot{z}) = \begin{cases} c_0 v_{L1} + c_{L1}\left(\dot{z} - v_{L1}\right), & \text{for } \dot{z} \le v_{L1} \\ c_0 \dot{z}, & \text{for } v_{L1} < \dot{z} \le v_{R1} \\ c_0 v_{R1} + v_{R1}\left(\dot{z} - v_{R1}\right), & \text{for } \dot{z} > v_{R1} \end{cases} \tag{8}$$

Quadlinear damping:

$$c_{\mathrm{QL}}(\dot{z}) = \begin{cases} c_0 v_{L1} + c_{L1}\left(\dot{z} - v_{L1}\right), & \text{for } \dot{z} \le v_{L1} \\ c_0 \dot{z}, & \text{for } v_{L1} < \dot{z} \le v_{R1} \\ c_0 v_{R1} + c_{R1}\left(\dot{z} - v_{R1}\right), & \text{for } v_{R1} < \dot{z} \le v_{R2} \\ c_0 v_{R1} + c_{R1}\left(v_{R2} - v_{R1}\right) + c_{R2}\left(\dot{z} - v_{R2}\right), & \text{for } \dot{z} > v_{R2} \end{cases} \tag{9}$$

Note that the variation of damping parameters across different regions is denoted by c_0, $c_{L\#}$ and $c_{R\#}$, and the velocity partitions between two adjacent regions of operation are denoted on the velocity axis by either $d_{R\#}$ or $d_{L\#}$.

For the purpose of identification, it is worth looking at the parameter set for a PWL stiffness (or damping) model, it consists of the mass, the linear damping (or stiffness) parameter and the series of stiffness (or damping) and partition parameters. If the model-order (i.e. the number of linear regions) and the parameters of a PWL model are known, one can provide a known input excitation to the model and simulate forward responses, such as displacement, velocity, or acceleration, using a suitable numerical time-integration scheme, such as MATLAB's `ode45` [37]. The output measurements from such a system, denoted by y, could consists of displacement, velocity, and/or acceleration, or some of their derivatives.

3. Bayesian model selection and parameter estimation using ABC

The goal in this study is to identify the model-order and the parameters of an SDOF PWL stiffness/damping model, subject to a given external excitation u, based on some measured output data in discrete-time, $\mathcal{D} = \left\{ y_1, \ldots, y_{N_t} \right\}$. The idea is to select a best-fit model from a pool of models – comprising linear, bilinear, trilinear and quadlinear models – and estimate its parameters. Such an identification problem is more commonly addressed as the task of model selection and parameter estimation. A Bayesian inference framework is employed here for performing this task, as it naturally incorporates the principle of parsimony and provides uncertainty quantification in the form of probability distributions. In Bayesian inference, one seeks to compare the quality of fit of different models according to posterior probability distributions (after observing evidence) over them. The Bayesian inference approach has been successfully applied in various domains ranging from structural dynamics [38–41], degradation and defect modelling [42,43], genetics [44], finance [45,46], ecology [47,48], etc.

In Bayesian parameter estimation, one seeks to estimate the posterior probability distribution over the parameter set for a specified model, given some measured data $\mathcal{D}$. The posterior distribution of parameters $\boldsymbol{\theta}^{(i)}$ for model $\mathcal{M}_i$, is given by Bayes' rule,

$$p\left(\boldsymbol{\theta}^{(i)} \mid \mathcal{D}, \mathcal{M}_i\right) = \frac{p\left(\mathcal{D} \mid \boldsymbol{\theta}^{(i)}, \mathcal{M}_i\right) p\left(\boldsymbol{\theta}^{(i)} \mid \mathcal{M}_i\right)}{p\left(\mathcal{D} \mid \mathcal{M}_i\right)} \tag{10}$$

where $p\left(\mathcal{D} \mid \boldsymbol{\theta}^{(i)}, \mathcal{M}_i\right)$ is the likelihood of the dataset, $p\left(\boldsymbol{\theta}^{(i)} \mid \mathcal{M}_i\right)$ is the parameter prior distribution, and $p\left(\mathcal{D} \mid \mathcal{M}_i\right)$ is the marginal likelihood (also called the model evidence). The model evidence is a multi-dimensional integral over the model-parameter space,

$$p\left(\mathcal{D} \mid \mathcal{M}_i\right) = \int p\left(\mathcal{D} \mid \boldsymbol{\theta}^{(i)}, \mathcal{M}_i\right)\ p\left(\boldsymbol{\theta}^{(i)} \mid \mathcal{M}_i\right) d\boldsymbol{\theta}^{(i)} \tag{11}$$

and is typically analytically intractable, especially when the likelihood and/or prior distribution is non-Gaussian and the dimension of the parameter space $\boldsymbol{\theta}^{(i)}$ is high. When one is only concerned with parameter posterior estimation, computation of the model

evidence is not essential, and it can be bypassed using standard Markov Chain Monte Carlo (MCMC) methods [49] – they only need the form of the unnormalised parameter posterior and can provide stationary samples from the parameter posterior distribution. Nonetheless, the model evidence is important for Bayesian model selection, where given a plausible set of K models, $\boldsymbol{\mathcal{M}} = \{\mathcal{M}_1, \ldots, \mathcal{M}_K\}$, the best model is determined as the one with the highest relative model posterior probability among all models in the set. The model posterior for the ith model $\mathcal{M}_i$ is proportional to the model evidence,

$$p(\mathcal{M}_i \mid \mathcal{D}) = \frac{p(\mathcal{D} \mid \mathcal{M}_i)\ p(\mathcal{M}_i)}{p(\mathcal{D})} \propto p(\mathcal{D} \mid \mathcal{M}_i)\ p(\mathcal{M}_i) \tag{12}$$

Several methods have been proposed to compute the model evidence, see review paper [50] for more details. With the advancement of MCMC algorithms, model evidence is more commonly estimated using MCMC samples from the parameter posterior distribution. Examples of such methods include Chib's algorithm [51], and transitional MCMC [52]. However, the implementation of standard MCMC methods to compute model evidence is not straightforward, as they require several additional steps and their efficacy depends on proper tuning of algorithmic hyperparameters. A specific MCMC algorithm that allows simultaneous model selection and parameter estimation in 'one-go' is Reversible-jump MCMC (RJMCMC) [53], which is quite popular for Bayesian model selection and parameter estimation. However, RJMCMC has its own drawbacks: each time an RJMCMC jumps between models, the parameter values sampled in the previous model are lost and subsequent jumps into that model must start afresh, which can make the RJMCMC quite inefficient.

Instead of following an MCMC-based approach, an ABC approach is adopted here, as it is conceptually intuitive and quite straightforward to implement. Moreover, ABC constitutes a class of simulation-based likelihood-free methods that has the advantage of being able to handle problems where the likelihood is not exactly known or is difficult to compute [54,55]. However, a drawback of ABC methods is that they can be computationally very slow. All ABC-based methods operate by generating simulated datasets and comparing them to the observed data. To conduct posterior parameter estimation, the generated simulated data $\mathcal{D}^*$ is compared to the observed data $\mathcal{D}$, and the corresponding input parameter set is accepted if a suitable distance measure between them $\rho(\mathcal{D}, \mathcal{D}^*)$ is less than a specified threshold ε defined by the user. The ABC algorithm thus provides samples from the approximate parameter posterior of the form

$$p(\boldsymbol{\theta}^{(i)} \mid \mathcal{D}, \mathcal{M}_i) \approx p_\varepsilon(\boldsymbol{\theta}^{(i)} \mid \mathcal{D}, \mathcal{M}_i) \propto \int p(\boldsymbol{\theta}^{(i)}, \mathcal{D}^* \mid \mathcal{M}_i, \rho(\mathcal{D}, \mathcal{D}^*) < \varepsilon)\ d\mathcal{D}^* \tag{13}$$

where $p_\varepsilon(\boldsymbol{\theta}^{(i)} \mid \mathcal{D}, \mathcal{M}_i)$ is the ABC-based approximate parameter posterior for model $\mathcal{M}_i$ and it becomes increasingly close to the true parameter posterior as ε gets smaller. In this study, the distance measure has been chosen to be the Normalised Mean Squared Error (NMSE) between the simulated data $\mathcal{D}^*$ and observed data $\mathcal{D}$,

$$\rho(\mathcal{D}, \mathcal{D}^*) = \frac{100}{N_t \sigma_{\mathcal{D}}^2} \sum_{i=1}^{N_t} (y_i^* - y_i)^2 \tag{14}$$

where $\sigma_{\mathcal{D}}^2$ is the variance of the observed dataset. Along with parameter estimation, ABC methods can also perform model selection based on relative acceptance frequencies of parameter samples for different models in a set of models, thus avoiding the direct need for model evidence computation. The algorithmic implementation of the ABC framework is quite simple and straightforward, which makes it an attractive alternative to the conventional Bayesian approaches, particularly when performing model selection. There are several ABC algorithms that have been proposed in the last decade, such as ABC rejection sampling [56], ABC Sequential Monte Carlo [32], ABC-MCMC [57], ABC Nested Sampling [33], ABC Subset Simulation [58]; see [54] for details.

In this study, the ABC-NS [33] algorithm is applied for Bayesian model selection and parameter estimation, albeit with some modifications tailored for faster convergence for the application at hand. The procedure of ABC-NS algorithm starts by generating a set of parameter samples, referred to as *particles*, from different models in the model set. The collection of all particles is called a *population*. A set of "active" particles are retained from the population, selected based on their relative NMSE values. Sampling distributions are defined using the active particles, which are then employed to generate (or sample) a new set of particles; together, the active particles and the newly-generated set of particles constitute the next population of particles. This process of generating new populations of particles is repeated until a convergence/stopping criteria is met. The relative proportion of particles belonging to each model from the last population serves to determine the relative model posterior, whereas the distribution of the model-specific particles themselves determines the parameter posterior distributions.

The ABC-NS algorithm implemented in this study consists of the following main steps:

1. *Create an initial population of particles*: This step generates an initial population of N_s particles from K models using model priors and parameter priors. First, the models are sampled from model priors, then model-specific parameters are sampled from parameter priors. For each sampled parameter $\boldsymbol{\theta}^*$ belonging to model $\mathcal{M}_k$, a dataset $\mathcal{D}^*$ is simulated and compared with the observed dataset $\mathcal{D}$, using NMSE given in Eq. (14). A number of particles N_s, that have their NMSE values lower than an initial user-defined threshold ε_1 are accepted and saved as the initial population of particles. The lines 2–12 in Algorithm 1 detail the generation of initial population of particles.
2. *Define the next threshold*: The NMSE values of all particles of the initial population are sorted (or ranked) in a descending order, and the next threshold ε_2 is defined as equal to the $(\alpha_0 N_s)$th element of the ordered NMSE values. See lines 13–15 in Algorithm 1.

3. *Assign weights to particles*: The particles with NMSE values above threshold ϵ_2 are dropped and the rest of the particles – the "active" particles – are assigned weights $\{\boldsymbol{w}_k\}_{k=1}^K$ based on their NMSE values; a particle with a lower NMSE value receives a higher weight. See lines 16–17 of Algorithm 1 and the weighting module in Algorithm 3.
4. *Construct ellipsoids for sampling*: The model-specific active particles, saved in $\{\boldsymbol{\mathcal{A}}_k\}_{k=1}^K$, are used to construct model-specific ellipsoids $\{\mathcal{E}_k\}_{k=1}^K$ defined by their weighted means $\{\boldsymbol{\mu}_k\}_{k=1}^K$ and weighted covariances $\{\boldsymbol{\Sigma}_k\}_{k=1}^K$ respectively,

$$\boldsymbol{\mu}_k = \frac{1}{\sum_i w_{k,i}} \sum_i w_{k,i} \boldsymbol{\theta}_{k,i} \tag{15a}$$

$$\boldsymbol{\Sigma}_k = \frac{1}{\sum_i w_{k,i}} \sum_i w_{k,i} \left(\boldsymbol{\theta}_{k,i} - \boldsymbol{\mu}_k\right) \left(\boldsymbol{\theta}_{k,i} - \boldsymbol{\mu}_k\right)^T \tag{15b}$$

   Here $\boldsymbol{\theta}_{k,i}$ denotes the ith active parameter particle from model $\mathcal{M}_k$ and $W_{k,i}$ is its corresponding scalar weight. Furthermore, the best parameter particles (with lowest NMSE values) from all models in the current population are saved as $\{\boldsymbol{\theta}_{k,\text{best}}\}_{k=1}^K$. See lines 18–19 in Algorithm 1 and the ellipsoid construction module in Algorithm 4.
5. *Sample from the ellipsoids*: Define the next population of N_s particles by adding particles to the existing set of active particles. These added particles are obtained by sampling from the ellipsoids $\{\mathcal{E}_k\}_{k=1}^K$. During this step, based on a certain prefixed probability p_{best}, either standard ellipsoids with mean $\{\boldsymbol{\mu}_k\}_{k=1}^K$ and covariances $\{\boldsymbol{\Sigma}_k\}_{k=1}^K$ are used for sampling, or 'shrunken' ellipsoids with reduced covariances $\{\gamma \boldsymbol{\Sigma}_k\}_{k=1}^K$ $(0 < \gamma < 1)$ and mean shifted to the location of the best particles $\{\boldsymbol{\theta}_{k,\text{best}}\}_{k=1}^K$ are used. The shrunken ellipsoids serve the purpose of sampling particles that have greater chances of acceptance and are useful in scenarios where the acceptance rates of particles from standard ellipsoidal sampling become diminishingly small. However, sampling from shrunken ellipsoids is expected to occur only occasionally and is controlled by p_{best}, which is usually given a small value such as 0.05, meaning that shrunken ellipsoids may be used for sampling only 5% of the total sampling iterations on an average. Note that, the sampled particles are accepted only if they satisfy the constraint conditions (if any), and when their corresponding NMSE values fall below the previously-generated threshold.
6. *Computing the next threshold automatically*: The next threshold is selected adaptively, based on NMSE values of particles of the current population, as well as those of the previous population. The idea is to compare the relative concentration of the distributions of model-specific NMSE values between two populations of particles, and then gauge how much the tolerance should be decreased to: (a) prevent diminishingly-small acceptance rates (when the tolerance change is too big) and/or (b) prevent the algorithm getting stuck around the same tolerance (when the tolerance change is too small and the samples are dominated by a single model class). For comparing concentrations, a kernel density estimator is used to estimate the NMSE densities of two successive populations of particles for each model class, and the ratios of the maximum values of the densities for the two successive populations are computed. The ratio r_k for a model class $\mathcal{M}_k$ is computed as,

$$r_k = 1 - \frac{f_{k,p-1}}{f_{k,p}} \tag{16}$$

   where $f_{k,p-1}$ and $f_{k,p}$ are the maximum values of the densities of NMSE values for the successive populations $p-1$ and p, respectively. Fig. 4 provides an example illustration of the idea for ratio calculations. In the illustration, four model classes are considered and density estimates over the NMSE values for each model class are obtained for the current population p and the previous population $p-1$. These density estimates are computed using the NMSE values of the corresponding parameter particles in the respective populations. For each model class, the maximum values of the density estimates (denoted by stars in the figure) are used to calculate the ratio given in Eq. (16). A higher concentration of NMSE values in the current population than the previous one will produce values of $\frac{f_{k,p-1}}{f_{k,p}}$ less than 1. In the illustration, the greatest increase in the concentration of NMSE values from previous to current population is seen for Model 1, with a prominent difference in the maximum values of its density estimates between two successive populations; and hence, it produces the largest ratio among all model classes. For Models 3 and 4, the difference in the maximum values of the densities are very small and their corresponding ratios would be small. Note that negative values of r_k may also manifest when values of $\frac{f_{k,p-1}}{f_{k,p}}$ greater than 1 occur. This case is illustrated for Model 2, where the maximum value of density estimate for the current population is lower than that of the previous population.
   A quantile α is assigned based on the maximum of the ratios $\{r_1, \ldots, r_K\}$, however, α is set to never go below a minimum of 0.1 to prevent stagnation of the population. Once the value of α is determined, the NMSE values of all the particles of the current pth population are sorted in a descending order, and the next threshold ϵ_{p+1} is defined equal to the (αN_s)th element of the ordered NMSE values. See lines 27–28 in Algorithm 1 and the automatic threshold selection module in Algorithm 2.
7. *Repeat iterations until stopping criterion is met*: The steps of weight assignment (Step 3), ellipsoid construction (Step 4), sampling from ellipsoids (Step 5), and automatic calculation of next threshold (Step 6) are repeated until a stopping criteria is met. When the relative difference between NMSE thresholds between two successive populations is less than a preset tolerance ϵ_{tol}, the ABC algorithm is deemed to have converged and the iterations are stopped.

Post convergence of the ABC algorithm, the posterior model probability is approximated by the proportion of particles in each model class from the last population,

$$p\left(\mathcal{M}_k \mid \mathcal{D}\right) \approx \frac{\text{Number of particles in } \mathcal{M}_k}{\text{Total number of particles } N_s} \tag{17}$$

Algorithm 1 Modified ABC-NS sampler

Input: Observed output data $\mathcal{D} = \{y_1, \dots, y_n\}$, input excitation u, model set $\boldsymbol{\mathcal{M}} = \{\mathcal{M}_1, \dots, \mathcal{M}_K\}$, total number of particles N_s, initial proportion of dropped particles α_0, initial threshold ε_1, importance-sampling probability p_{best}, shrink factor γ, stopping criterion
procedure GENERATE INITIAL POPULATION OF PARTICLES ▷ For first population
  $p = 1$, iter = 0
  while iter $< N_s$ **do**
    Sample a model from model set: $\mathcal{M}_k \sim p(\mathcal{M} \mid \boldsymbol{\mathcal{M}})$
    Sample a parameter from the model: $\boldsymbol{\theta}^* \sim p(\boldsymbol{\theta} \mid \mathcal{M}_k)$
    Simulate data: $\mathcal{D}^* \sim \mathcal{M}_k(\boldsymbol{\theta}^*)$ subject to excitation u
    Calculate NMSE value: $\rho(\mathcal{D}, \mathcal{D}^*)$
    if $\rho(\mathcal{D}, \mathcal{D}^*) < \varepsilon_1$ **then**
      Save parameter sample $\boldsymbol{\theta}^*$ in $\boldsymbol{\Theta}_{k,1}$
      Save NMSE value $\rho(\mathcal{D}, \mathcal{D}^*)$ in $\boldsymbol{e}_{k,1}$
      iter = iter + 1
  Concatenate the NMSE values for all models: $\boldsymbol{e}_{\text{all},1} = [\boldsymbol{e}_{1,1}, \dots, \boldsymbol{e}_{K,1}]$, $\boldsymbol{e}_{\text{all},1} \in \mathbb{R}^{N_s}$
  Sort $\boldsymbol{e}_{\text{all},1}$ in descending order, and get an integer index $i_x = \text{floor}(\alpha_0 N_s)$
  Define next threshold as $\varepsilon_2 = e_{\text{all},1}(i_x)$
procedure COMPUTE WEIGHTS OF PARTICLES($\{\boldsymbol{e}_{k,1}\}_{k=1}^K$, ε_1, ε_2)
  Output: $\{\boldsymbol{w}_k\}_{k=1}^K$
procedure CONSTRUCT ELLIPSOIDS FOR SAMPLING($\{\boldsymbol{\Theta}_{k,1}, \boldsymbol{e}_{k,1}, \boldsymbol{w}_k\}_{k=1}^K$)
  Output: $\{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \boldsymbol{\theta}_{k,\text{best}}, \boldsymbol{\mathcal{A}}_k, \boldsymbol{e}_k^{\mathcal{A}}\}_{k=1}^K$, N_a
==========
while stopping criterion is satisfied **do**
  Increment population $p = p + 1$
  procedure SAMPLE FROM ELLIPSOIDS($\{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \boldsymbol{\theta}_{k,\text{best}}\}_{k=1}^K$, p_{is}, γ, $\mathcal{D}$, ε_p, N_a, N_s, u)
    Outputs: $\{\boldsymbol{\mathcal{S}}_k, \boldsymbol{e}_k^{\mathcal{S}}\}_{k=1}^K$
  Define current population of particles: $\boldsymbol{\Theta}_{k,p} = \{\boldsymbol{\mathcal{A}}_k, \boldsymbol{\mathcal{S}}_k\}$ for $k = 1, \dots, K$
  NMSE values for current population: $\boldsymbol{e}_{k,p} = \{\boldsymbol{e}_k^{\mathcal{A}}, \boldsymbol{e}_k^{\mathcal{S}}\}$ for $k = 1, \dots, K$
  procedure COMPUTE THE NEXT THRESHOLD($\{\boldsymbol{e}_{k,p-1}, \boldsymbol{e}_{k,p}\}_{k=1}^K$)
    Output: ε_{p+1}
  procedure COMPUTE WEIGHTS OF PARTICLES($\{\boldsymbol{e}_{k,p}\}_{k=1}^K$, ε_p, ε_{p+1})
    Output: $\{\boldsymbol{w}_k\}_{k=1}^K$
  procedure CONSTRUCT ELLIPSOIDS FOR SAMPLING($\{\boldsymbol{\Theta}_{k,p}, \boldsymbol{e}_{k,p}, \boldsymbol{w}_k\}_{k=1}^K$)
    Output: $\{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \boldsymbol{\theta}_{k,\text{best}}, \boldsymbol{\mathcal{A}}_k, \boldsymbol{e}_k^{\mathcal{A}}\}_{k=1}^K$, N_a
  Check stopping criterion

Algorithm 2 Automatic threshold selection module

Inputs: NMSE values of previous population $\{\boldsymbol{e}_{k,p-1}\}_{k=1}^K$ and current population $\{\boldsymbol{e}_{k,p}\}_{k=1}^K$
Outputs: Threshold ε_{p+1}
procedure COMPUTE A NEW NMSE THRESHOLD ▷ For the $(p+1)$th population
  Set K equal to the total number of models in model set
  for each model $k = 1 : K$ **do**
    if $\boldsymbol{e}_{k,p-1}$ and $\boldsymbol{e}_{k,p}$ are non-empty **then**
      Estimate pdf, $\hat{p}(\boldsymbol{e}_{k,p-1})$, and find its maximum value: $f_{k,p-1} = \max(\hat{p}(\boldsymbol{e}_{k,p-1}))$
      Estimate pdf, $\hat{p}(\boldsymbol{e}_{k,p})$, and find its maximum value: $f_{k,p} = \max(\hat{p}(\boldsymbol{e}_{k,p}))$
      Store the ratio: $r_k = 1 - (f_{k,p-1}/f_{k,p})$
  Find the maximum ratio among all models: $r_{\max} = \max\{r_1, \dots, r_K\}$
  Set quantile value $\alpha = \max$ of $\{r_{\max}, 0.1\}$
  Concatenate NMSE values from all models: $\boldsymbol{e}_{\text{all},p} = [\boldsymbol{e}_{1,p}, \dots, \boldsymbol{e}_{K,p}]$, $\boldsymbol{e}_{\text{all},p} \in \mathbb{R}^{N_s}$
  Sort $\boldsymbol{e}_{\text{all},p}$ in descending order, and get the integer index $i_x = \text{floor}(\alpha N_s)$
  Define temporary threshold as $\varepsilon_{p+1} = e_{\text{all},p}(i_x)$

Algorithm 3 Weight assignment module

Inputs: NMSE values $\{\boldsymbol{e}_{k,p}\}_{k=1}^{K}$, current threshold ϵ_p, next threshold ϵ_{p+1}
Outputs: Weights $\{\boldsymbol{w}_k\}_{k=1}^{K}$ ▷ Weights for each sample in each model
procedure COMPUTE WEIGHTS FOR PARTICLES ▷ For the pth population
  for each model $k = 1 : K$ **do**
    if $\boldsymbol{e}_{k,p}$ is non-empty **then**
      Get the total number of particles: $n_k = \text{length}\left(\boldsymbol{e}_{k,p}\right)$
      Initialise weight vector to zeros: $\boldsymbol{w}_k = \text{zeros}(n_k, 1)$
      Find the set of indices, $\mathcal{I}$, of elements of $\boldsymbol{e}_{k,p}$ with NMSE values lower than ϵ_{p+1}
      for each index i in $\mathcal{I}$ **do**
        Set the corresponding weight, $w_k(\mathcal{I}_i) = \frac{1}{\epsilon_p}\left(1 - \left(\frac{e_{k,p}(\mathcal{I}_i)}{\epsilon_p}\right)^2\right)$
      The weights for the rest of the elements remain zero
      Normalise the weights such that they add upto unity

Algorithm 4 Ellipsoid construction module

Inputs: Population of particles $\{\boldsymbol{\Theta}_{k,p}\}_{k=1}^{K}$, NMSE values $\{\boldsymbol{e}_{k,p}\}_{k=1}^{K}$, weights $\{\boldsymbol{w}_k\}_{k=1}^{K}$
Outputs: Mean vectors $\{\boldsymbol{\mu}_k\}_{k=1}^{K}$, covariances $\{\boldsymbol{\Sigma}_k\}_{k=1}^{K}$, best particles $\{\boldsymbol{\theta}_{k,\text{best}}\}_{k=1}^{K}$, active particles $\{\boldsymbol{\mathcal{A}}_k\}_{k=1}^{K}$, NMSE values of active particles $\{\boldsymbol{e}_k^{\mathcal{A}}\}_{k=1}^{K}$, count of active particles N_a
procedure CONSTRUCT ELLIPSOIDS FOR SAMPLING
  Set the count of active particles to zeros: $\boldsymbol{n}_a = \text{zeros}(K, 1)$
  for each model $k = 1 : K$ **do**
    if $\boldsymbol{\Theta}_{k,p}$ is non-empty **then**
      Calculate the number of non-zero weights: $n_a(k) = \text{length}\left(\text{find}(\boldsymbol{w}_k > 0)\right)$
      Save $n_a(k)$ number of active particles from $\boldsymbol{\Theta}_{k,p}$ in $\boldsymbol{\mathcal{A}}_k$
      Select the corresponding NMSE values of the active particles from $\boldsymbol{e}_{k,p}$ and save them in $\boldsymbol{e}_k^{\mathcal{A}}$
      Save the parameter sample of $\mathcal{M}_k$ with the highest weight as $\boldsymbol{\theta}_{k,\text{best}}$
      Define ellipsoid $\mathcal{E}_k$ with weighted mean $\boldsymbol{\mu}_k$ and weighted covariance $\boldsymbol{\Sigma}_k$ using active particles in $\boldsymbol{\mathcal{A}}_k$
      Enlarge the ellipsoid by a factor f_0
  Save the total count of active particles, $N_a = \sum_{k=1}^{K} n_a(k)$

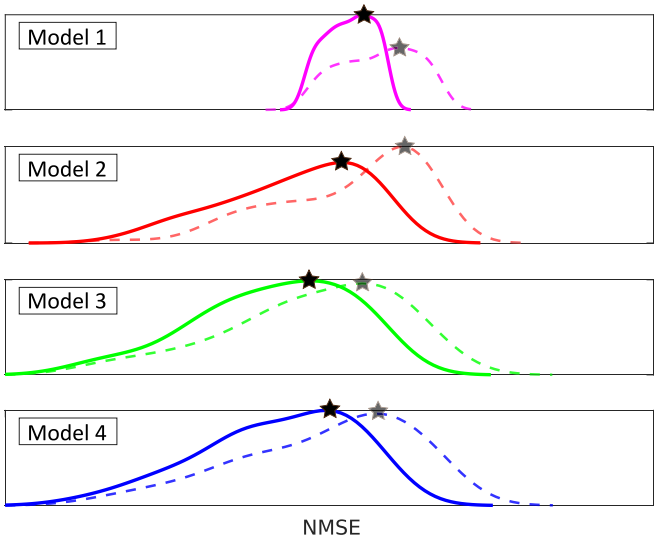


Fig. 4. Illustration of the idea of calculation of ratio r_k. Each plot shows density estimates for current population p (shown in bold line) and previous population $p-1$ (shown in dotted line) for a certain model class. The stars denote the maximum values of the density estimates.

Algorithm 5 Sampling module

Inputs: Ellipsoids $\{\mathcal{E}_k\}_{k=1}^{K}$ with means $\{\boldsymbol{\mu}_k\}_{k=1}^{K}$ and covariances $\{\boldsymbol{\Sigma}_k\}_{k=1}^{K}$, best particles $\{\boldsymbol{\theta}_{k,\text{best}}\}_{k=1}^{K}$, importance-sampling probability p_{best}, shrink factor γ, observed data D, current threshold ϵ_p, count of active particles N_a, total number of particles N_s, excitation u
Outputs: Sampled population $\{\boldsymbol{S}_k\}_{k=1}^{K}$, NMSE values for sampled population $\{\boldsymbol{e}_k^S\}_{k=1}^{K}$
Dropped number of particles: $N_d = N_s - N_a$
procedure SAMPLE PARTICLES FROM ELLIPSOIDS
    `iter = 0, satisfiedConstraint = false`
    while iter $< N_d$ **do**
        Sample a model from model set; $\mathcal{M}_k \sim p(\mathcal{M} \mid \boldsymbol{\mathcal{M}})$
        Sample a uniform random number: $v \sim U(0,1)$ ▷ $U(0,1)$ is a uniform distribution between (0,1)
        if $p_{\text{is}} > v$ **then** ▷ Condition for sampling around best particle
            Sample a particle θ^{**} from ellipse $\mathcal{E}_k$ with center $\boldsymbol{\theta}_{k,\text{best}}$ and shrinked covariance $\gamma\boldsymbol{\Sigma}_k$
        else
            Sample a particle θ^{**} from ellipse $\mathcal{E}_k$ with center $\boldsymbol{\mu}_k$ and covariance $\boldsymbol{\Sigma}_k$
        Check if parameter sample satisfies constraints: `satisfiedConstraint = true/false` ?
        if `satisfiedConstraint = true` **then**
            Simulate data: $D^* \sim \mathcal{M}_k(\theta^{**})$ subject to excitation u
            Calculate NMSE value: $\rho(D, D^*)$
            if $\rho(D, D^*) < \epsilon_p$ **then**
                Save parameter sample $\boldsymbol{\theta}^{**}$ in $\boldsymbol{S_k}$
                Save NMSE value $\rho(D, D^*)$ in $\boldsymbol{e}_k^S$
                iter = iter + 1

The model class that has the largest proportion of particles from the population of N_s particles is taken to be the "best" model, and the posterior distributions over its parameters are approximated by the particles belonging to the model.

To summarise, this section outlined an ABC algorithm to serve the purpose of identifying the correct model-order of a PWL system and estimate the posterior distribution of the model parameters. Identifying the correct model order entails first defining a set of PWL models with different model orders (such as bilinear, trilinear, quadlinear) and then using ABC to select the best-fit model-order, according to the relative proportion of model-specific particles from the last population of ABC. For the selected model, the posterior distributions over its parameters are approximated using the particles that belong to the selected model.

4. Numerical demonstrations

In this section, two numerical studies are conducted to demonstrate the working of the proposed methodology for identifying SDOF PWL oscillators, one with PWL stiffness and the other with PWL damping. The intention behind using these two different studies is to look at how the performance of the proposed methodology differs across these two types of PWL nonlinearities. Since the "true" data-generating models are known in these studies, it would be easier to understand how the results from the proposed methodology compare with the true system. The mathematical descriptions of both numerical examples conform to those defined in Section 2. Further, a third numerical study is considered, where a PWL model is fitted to a system with cubic restoring force using the proposed procedure. The intention there is to understand the extent to which fitted PWL models may be useful, especially for cases where the restoring forces may be continuous in variation. In all cases, the systems are subjected to an input excitation u, that is modelled as a zero-mean Gaussian bandlimited noise sequence, with a standard deviation of 2 and passband of $[0, 20]$ Hz; the time-series of the input excitation is provided in Fig. 5. Both systems are simulated using the `ode45` function in MATLAB, with a sampling rate of 100 Hz, for a time span of 10 s. The properties of each system are defined in Section 4.1, 4.2, and 4.3.

It must be emphasised once again, that the identification of a PWL stiffness (or damping) system involves determining the number of linear operating regions, i.e. the model order, the parameters associated with each linear region, i.e. the partition and regional stiffness (or damping) parameters, and the mass and damping (or stiffness) parameters. For model selection, a maximum of up to four linear regions (or fourth model-order) are guessed, which translates to performing model selection with a set of four models: a linear model, a bilinear model, a trilinear model, and a quadlinear model. Note that the number of parameters to be identified vary with each model class: three parameters for a linear model, five for a bilinear model, seven for a trilinear model, and nine for a quadlinear model. It must be remarked at this point, that the stiffness (or damping) and partition parameters of two neighbouring linear regions must differ from each other to avoid redundancies. Therefore, two constraint conditions are imposed while sampling the parameter particles using the ABC algorithm: (a) the partition parameters $d_{L\#}/d_{R\#}$ (or $v_{L\#}/v_{R\#}$) of a particle must be unique, and (b) the adjacent local stiffness (or damping) parameters of a particle must differ by a minimum of 5% from each other. Parameter particles that are accepted in an ABC population must satisfy these two constraints. In Bayesian inference, one requires to specify prior beliefs over parameters as well as over models. The model priors considered for the two simulation studies are the same, and a uniform discrete prior is assumed over all four models in the model set, that is, $p(\mathcal{M}_1) = p(\mathcal{M}_2) = p(\mathcal{M}_3) = p(\mathcal{M}_4) = \frac{1}{4}$. The

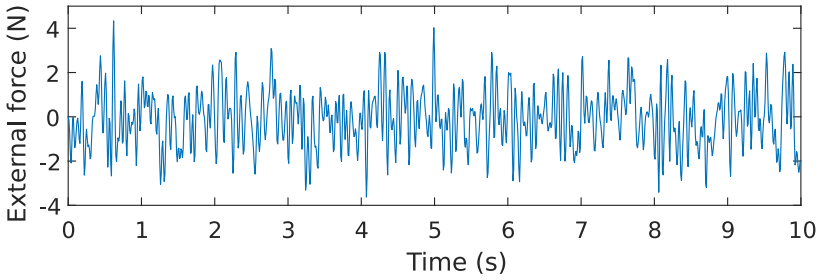


Fig. 5. External input forcing used to excite SDOF systems with PWL stiffness and damping.

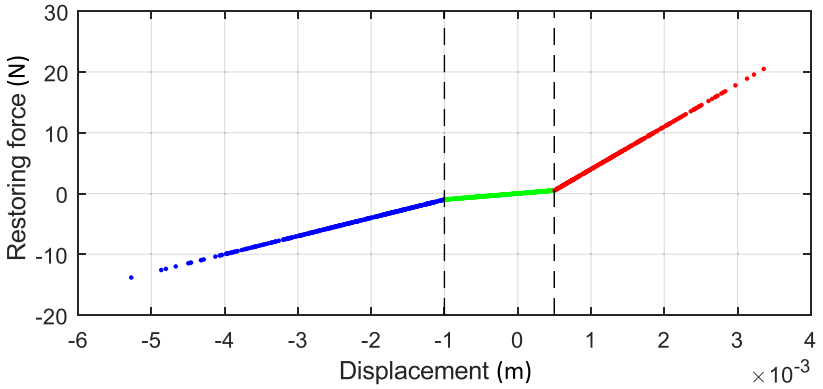


Fig. 6. Trilinear stiffness system with three different regions of the restoring force separated by two displacement partitions.

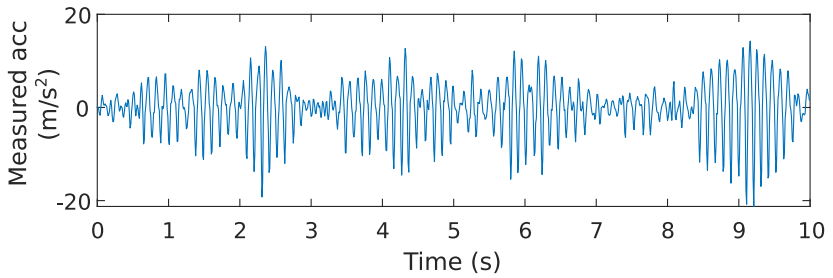


Fig. 7. Output acceleration used for identification of PWL stiffness dynamical system.

parameter-specific prior distributions for each simulation study are provided in the respective sections. The ABC algorithm is run with a total of $N_s = 1000$ particles, and the other algorithmic parameters are set to $\varepsilon_1 = 200$, $\alpha_0 = 0.4$, $p_{\text{best}} = 0.05$, $\gamma = 0.1$, and $\epsilon_{tol} = 0.005$. The values of these ABC algorithmic parameters remain the same for all following studies, unless mentioned otherwise. The simulations are run on a Windows-10 machine with Intel i5-4210M processor 2.60 GHz and 8 GB of RAM using MATLAB [37]. The next two subsections discuss the results of ABC inference on two SDOF dynamical systems, one with trilinear stiffness and the other with trilinear damping, respectively.

4.1. Identification of a dynamic model with trilinear stiffness

A dynamic SDOF spring–mass–damper with trilinear stiffness is considered for identification,

$$m\ddot{z} + c_0\dot{z} + k_{\text{TL}}(z) = u \tag{18}$$

with the true parameters of the model set to: $m = 1$ kg, $c_0 = 2$ Ns/m, $k_0 = 10^3$ N/m, $k_{L1} = 3 \times 10^3$ N/m, $k_{R1} = 7 \times 10^3$ N/m, $d_{L1} = -1 \times 10^{-3}$ m, and $d_{R1} = 0.5 \times 10^{-3}$ m. The trilinear map of the restoring force with respect to the displacement of the system is shown in Fig. 6. For the sake of measurements, the input force is assumed to be noise-free while the acceleration output is corrupted with 5% zero-mean Gaussian white noise; the output acceleration used for identification is shown in Fig. 7.

Table 1
Uniform parameter priors for linear (L), bilinear (BL), trilinear (TL), and quadlinear (QL) stiffness models.

Model	Parameter	Prior	Lower bound	Upper bound
L, BL, TL, QL	m (kg)	Log-uniform	0.5	1.5
L, BL, TL, QL	c_0 (Ns/m)	Log-uniform	0.1	4
L, BL, TL, QL	k_0 (N/m)	Log-uniform	0.5×10^3	1.5×10^3
BL, TL, QL	$k_{L\#}/k_{R\#}$ (N/m)	Log-uniform	0.5×10^3	10×10^3
BL, TL, QL	$d_{L\#}/d_{R\#}$ (m)	Uniform	-10×10^{-3}	10×10^{-3}

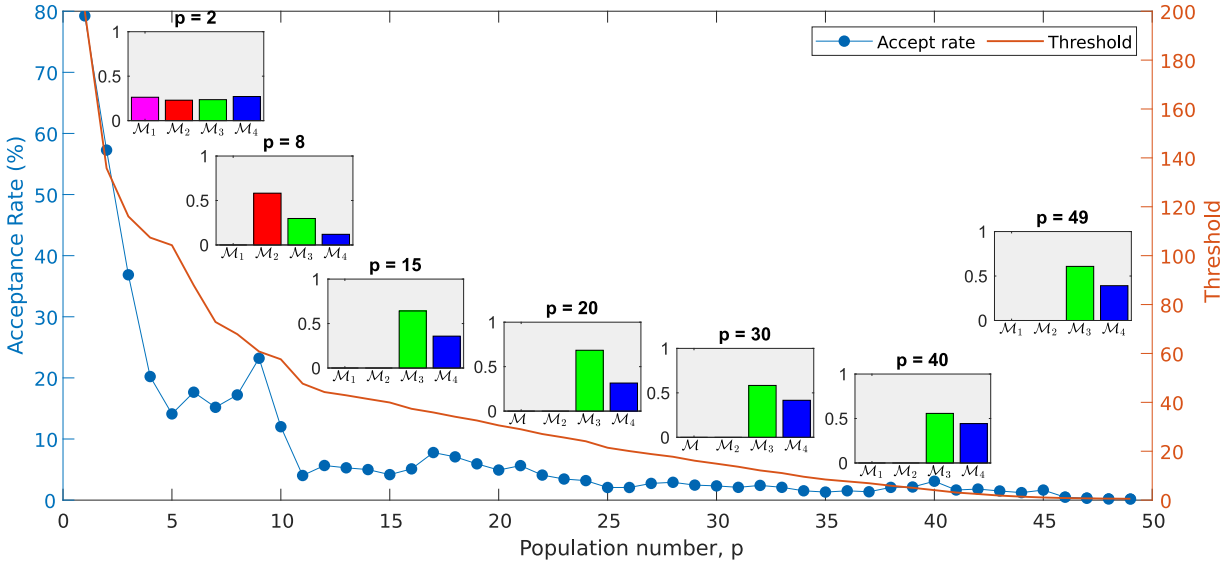


Fig. 8. Plot of acceptance rate and threshold across ABC populations. Small windows on the top of the plot depict model posterior probabilities for a selected number of populations.

Parameter prior distributions specified for all parameters of the models are tabulated in Table 1. Log-uniform priors are employed for positively-constrained parameters, while uniform priors are defined on partition parameters which can take non-positive values as well. The ABC algorithm took around 50 populations to converge, which used up three hours of computational time. The variation of acceptance rates and decrement of thresholds over several populations of ABC are shown in Fig. 8. The figure also shows model posterior probabilities for a selected number of populations. It can be seen that the acceptance rate decreases with populations as the simpler models exit and the parameter search-space becomes larger for trilinear and quadlinear models, while the thresholds becomes more tight. After the 10th population, the linear and bilinear models exit, as they are not able to produce datasets that are closer (in sense of NMSE) to the observed data than the NMSE thresholds of the next population. With the remaining trilinear and quadlinear models in the subsequent ABC populations, the parameter search spaces become relatively more complex, making it difficult to sample good particles – particles with NMSE lower than thresholds – and hence a decline in the acceptance rates is observed post population 10. Nonetheless, the ABC algorithm is able to clearly select the trilinear-stiffness model over the quadlinear-stiffness model at the end of convergence, demonstrating the principle of parsimony in Bayesian model selection. The marginal posterior distributions of parameters for the trilinear are shown in Fig. 9, in the form of histograms. The true values of the parameters are observed to lie closely around the peaks of the histograms, which shows the proposed methodology is not only able to encompass the true parameters within the posterior distribution but also the modes of the distribution would match closely with the true parameters. The parameter particles of the selected trilinear system can be further used to generate the several realisations of the restoring-force map. A plot of all such realisations, along with the mean of the restoring forces versus mean displacement is shown in Fig. 10, which gives a sense of uncertainty over the posterior of the trilinear restoring force map.

4.2. Identification of a dynamic model with trilinear damping

Having looked at identification of a PWL stiffness system in the previous subsection, this subsection considers a PWL damping system using a dynamic SDOF spring–mass–damper with trilinear damping,

$$m\ddot{z} + c_{\mathrm{TL}}(\dot{z}) + k_0 z = u \tag{19}$$

The model parameters for the forward simulation of the true model are considered as: $m = 1$ kg, $c_0 = 2$ Ns/m, $k_0 = 10^3$ N/m, $c_{L1} = 0.5$ Ns/m, $c_{R1} = 0.2$ Ns/m, $v_{L1} = -0.05$ m/s, and $v_{R1} = 0.025$ m/s. The corresponding trilinear map of the restoring force with

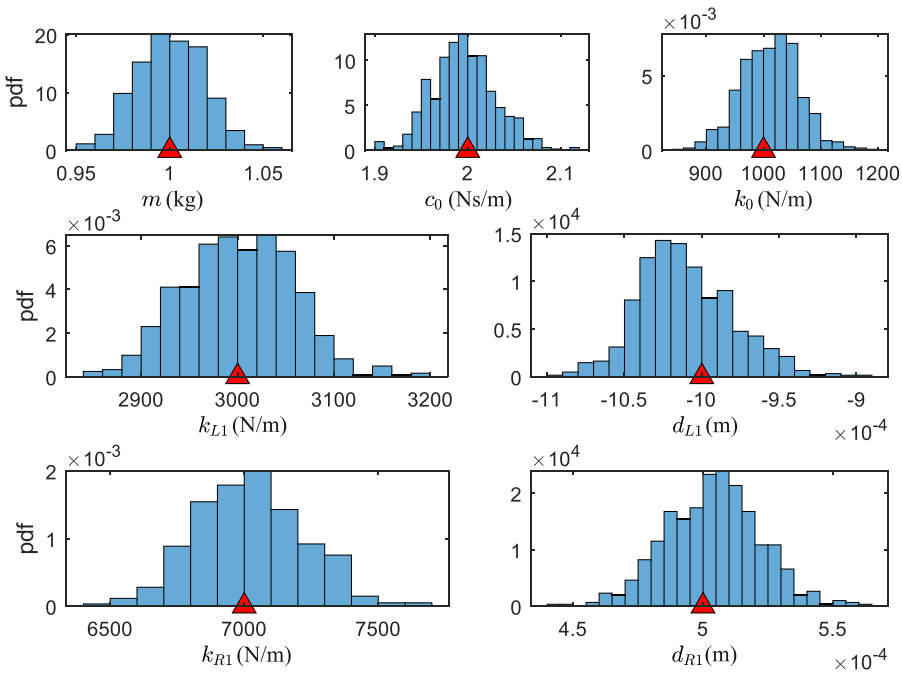


Fig. 9. ABC-based marginal posterior distributions over parameters of the selected trilinear stiffness model; the true values of parameters are shown in red triangle markers.

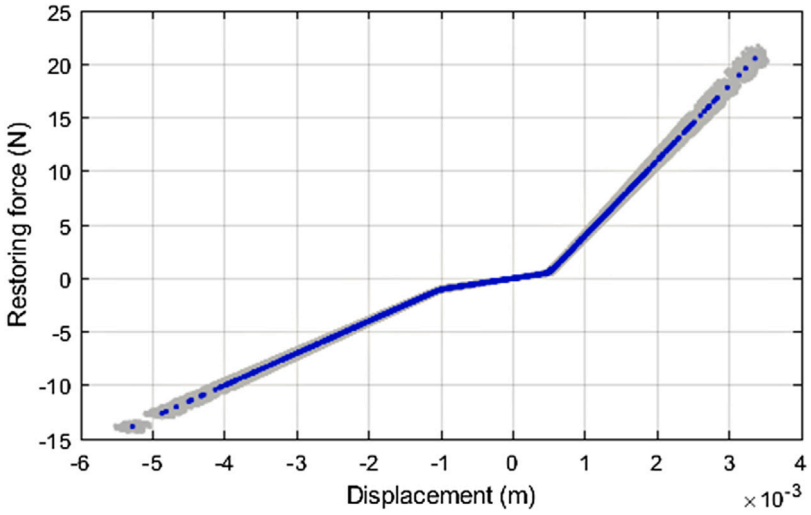


Fig. 10. Restoring force vs displacement map of the estimated trilinear stiffness system; the grey points in the background are realisations, and the blue points on the top represent the mean of the realisations.

respect to the velocity of the system is shown in Fig. 11. The noisy acceleration output used for identification is shown in Fig. 12. Similar to the previous study, for parameter priors, log-uniform distributions are specified over positively-constrained parameters and uniform distributions over velocity partition parameters; they are tabulated in Table 2.

Upon running, the ABC algorithm converged within 60 populations, taking up around 1 h and 15 min of computational time. A plot illustrating the acceptance rates and decrement of thresholds over the populations of ABC is provided in Fig. 13, with model posterior probabilities for a selected number of populations shown alongside. The linear model can be seen to survive for many more ABC populations (up to the 30th population), than what was seen in the PWL stiffness case. Such a behaviour is deemed to be caused because of the much lower sensitivity of response to damping, evident by the fact that the linear model with a single region of damping produces a response that is comparatively close (in the sense of NMSE), to the measured response, and hence the linear model is able to survive until the threshold drops quite low. Moreover, as the linear model is a simple model, its parameter space is

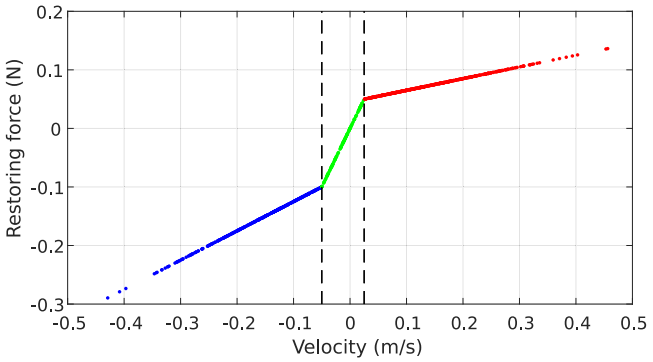


Fig. 11. Trilinear damping restoring force with three different damping regions separated by two velocity partitions.

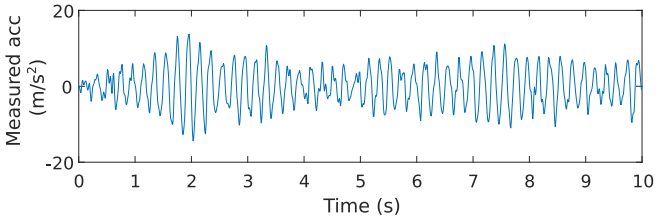


Fig. 12. Output acceleration used for identification of PWL damping dynamical system.

Table 2
Parameter priors for linear (L), bilinear (BL), trilinear (TL), and quadlinear (QL) damping models.

Model	Parameter	Prior	Lower bound	Upper bound
L, BL, TL, QL	m (kg)	Log-uniform	0.5	1.5
L, BL, TL, QL	c_0 (Ns/m)	Log-uniform	0.1	4
L, BL, TL, QL	k_0 (N/m)	Log-uniform	0.5×10^3	1.5×10^3
BL, TL, QL	$c_{L\#}/c_{R\#}$ (Ns/m)	Log-uniform	0.1	4
BL, TL, QL	$v_{L\#}/v_{R\#}$ (m/s)	Uniform	−0.5	0.5

small and it is easy to sample good particles; hence the ABC populations with the linear model have higher acceptance rates. Post departure of the linear model, the acceptance rate drops with the parameter space becoming more complex for the remaining PWL models. The bilinear damping model also departs at the 43rd population. Among the remaining trilinear and quadlinear damping models, the trilinear model has larger number of particles than the quadlinear model at convergence. Hence, based on the relative frequency of particles, the trilinear damping model is a better choice among the two surviving models. Nevertheless, the quadlinear damping model is also not a bad choice since the proportions of particles between the trilinear and quadlinear model do not differ by a large margin.

The marginal posterior parameter distributions for the trilinear damping model are shown in Fig. 14, in the form of histograms. The true values of the parameters are observed to lie around the peaks of the histograms, especially for mass and stiffness. However, the means of the posterior distributions associated with the regional damping parameters are a bit far off from the true values; the main reason behind such behaviour is once again attributed to the low sensitivity of response to damping. Nonetheless, it is satisfactory to note that their true values are captured well within the spread of the distributions. The posterior uncertainty over the trilinear restoring force map is shown by a scatter plot in Fig. 19; the realisations and the mean of the realisations are obtained by simulating the restoring forces for each posterior parameter particle of the selected trilinear system.

4.3. Identification of PWL dynamic model of a system with cubic restoring force

In the previous two numerical studies, the model class of the true systems and the estimated model were same, i.e., they were all PWL models. Different from the two previous studies, this study focuses on fitting PWL models to responses generated from model class that has continuous restoring force variation. In particular, an SDOF Duffing oscillator model is used for true response generation (with cubic restoring force variation):

$$m\ddot{z} + c_0\dot{z} + k_0 z + k_3 z^3 = u \tag{20}$$

The parameters used to simulate the acceleration response of the Duffing oscillator system are: $m = 1$ kg, $c_0 = 2$ Ns/m, $k_0 = 10^3$ N/m, and $k_3 = 10^8$ N/m^3. The time span of simulation and the input excitation sequence remains the same as used in

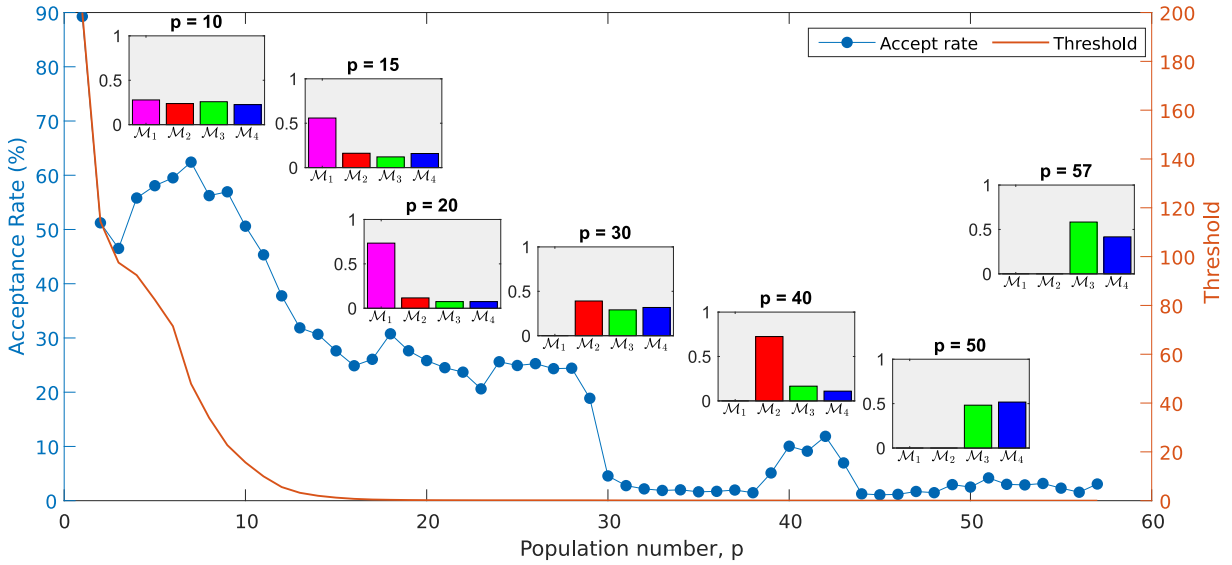


Fig. 13. Plot of acceptance rate and threshold across ABC populations. Small windows on the top of the plot depict model posterior probabilities for a selected number of populations.

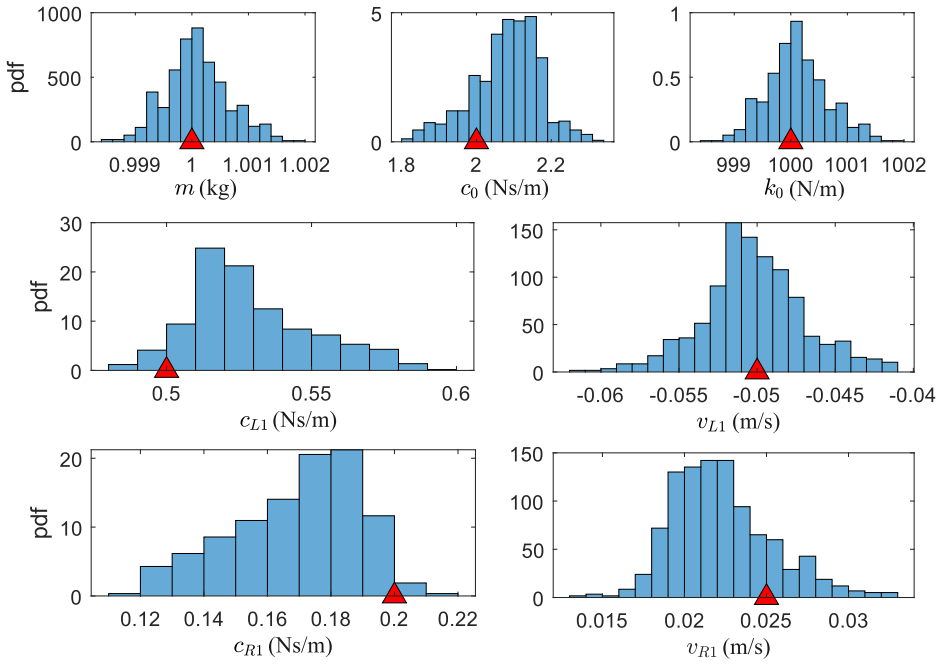


Fig. 14. ABC-based marginal posterior distributions over parameters of the selected trilinear damping model; the true values of parameters are shown in red triangle markers.

the previous two cases. Noiseless acceleration generated from the simulation is used as response data for PWL model selection and parameter estimation using the proposed procedure. As before, a set of four PWL stiffness models – linear, bilinear, trilinear, and quadlinear – are chosen and the proposed algorithm is run with $N = 1000$ particles until convergence.

From Fig. 15, it is found that the algorithm quickly eliminates the linear and bilinear stiffness models by the 10th generation, as these 'lower-order' models are unable to represent the cubic variation with decreasing tolerance. Note that with the linear and bilinear models exit around the tolerance threshold of 70 NMSE, leaving behind particles representing only the 'higher-order' trilinear and quadlinear stiffness models. Below an NMSE tolerance of 10, the trilinear model is unable to fit the data any better and exits the population at the 21st population, resulting in the quadlinear model as the only model choice for fitting the data. From the

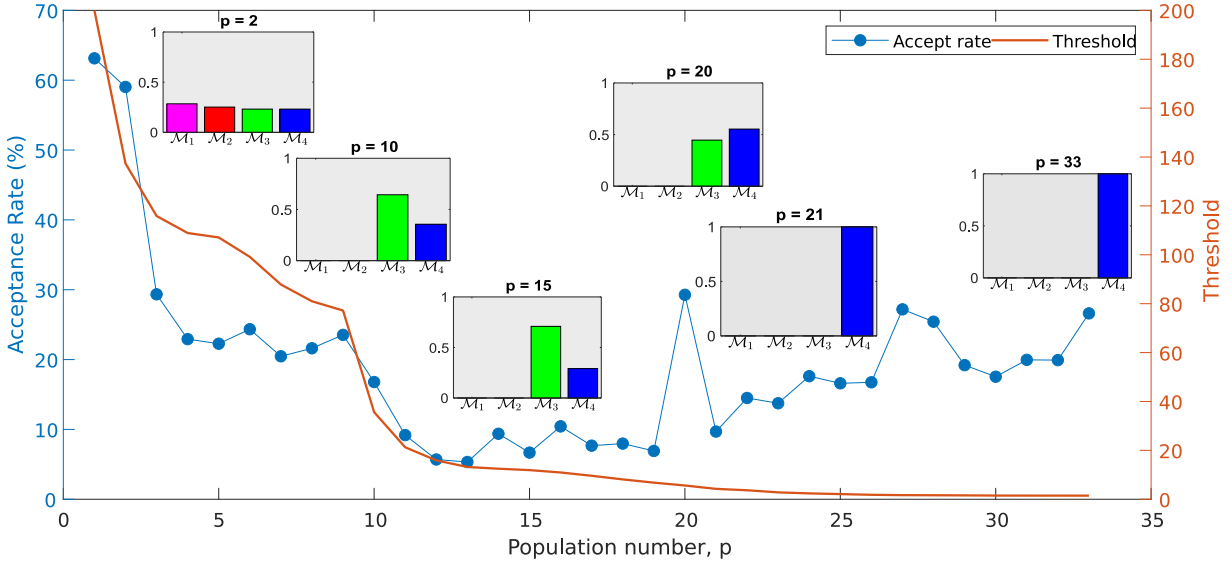


Fig. 15. Plot of acceptance rate and threshold across different ABC populations.

21st population until convergence, the iterations only tuned the parameter estimates of the quadlinear to better fit the data. The resulting parameter posterior distribution of the resulting quadlinear model is shown in Fig. 16. A mean-parameter-fitted quadlinear model is constructed by using the sample means of the parameters of the quadlinear model. The stiffness-based restoring force from the quadlinear model is then superimposed on the cubic stiffness restoring force from the Duffing oscillator for a comparison, in Fig. 17. Evidently, the quadlinear model produces a best-fit piecewise linear approximation of the continuous cubic restoring force. The discrepancy appears more near the extreme tails where the data points are few and it almost disappears in the middle regions. It is, therefore, expected that the response from the inferred PWL model would match the Duffing oscillator response very well in the regions of low values of acceleration and would differ only when the acceleration responses are high. This phenomenon can be seen clearly in Fig. 18.

To summarise the important points from the three numerical case studies presented in Section 4, a distinctive trend in the progression of ABC populations for PWL model selection is noticed. Starting with a uniform distribution of parameter particles from different models, the simplest PWL model among the surviving models tends to dominate the population of particles until the time it exits the population, due to unforgivingly low threshold for the simplest model. With the exit of the simplest model, the lost particles are replenished with parameter particles from the surviving higher-order PWL models (i.e. models having greater complexity than the exiting simpler model), and this behaviour is repeated until the algorithm converges. With the exit of a simpler model, the population has to replenished with parameter particles from higher-order PWL models. When the parameter space becomes larger for the remaining models, sampling good particles takes more effort and hence the acceptance ratio decreases. The time required for the algorithm to converge depends critically on the acceptance ratio (given a stopping tolerance); larger acceptance ratios lead to faster convergence. Another trend observed from multiple runs of the ABC algorithm is that a lower-order PWL model exits before a higher-order PWL from the ABC population. This behaviour makes sense as a lower-order PWL model is just a more constrained version of a higher-order PWL model, and hence it is only reasonable to eliminate a lower-order PWL before removing a higher-order PWL model. From an algorithmic perspective, a higher-order PWL model could be eliminated earlier than a lower-order PWL if the total number of particles used in the population is low; with a lower number of particles and a larger parameter space, it becomes more difficult to explore and sample good particles in future ABC populations and eventually poor exploration causes early exit of a higher-order PWL model. Finally, from the first two numerical studies, it was found that the proposed methodology was able to select the true data-generating trilinear model, based on the relative frequency of particles in the final ABC population. From the third numerical study, it was seen that in the case of continuous restoring force, the PWL model fit would be approximate but still useful.

5. Experimental study: Identification a shock absorber

This section considers an experimental case study on the nonlinear behaviour of an automotive shock absorber, where the objective is to fit a piecewise-linear relation to the restoring force. The shock absorber presents a more challenging case compared to the previous two simulation-based studies, mainly because the exact governing equation is not available and the relation is not exactly PWL.

The automotive shock absorber (or damper), is a fundamental part of the automobile suspension system, whose characteristics contribute greatly to the handling properties and ride comfort of a vehicle. Traditional industrial models tend to treat the mechanism

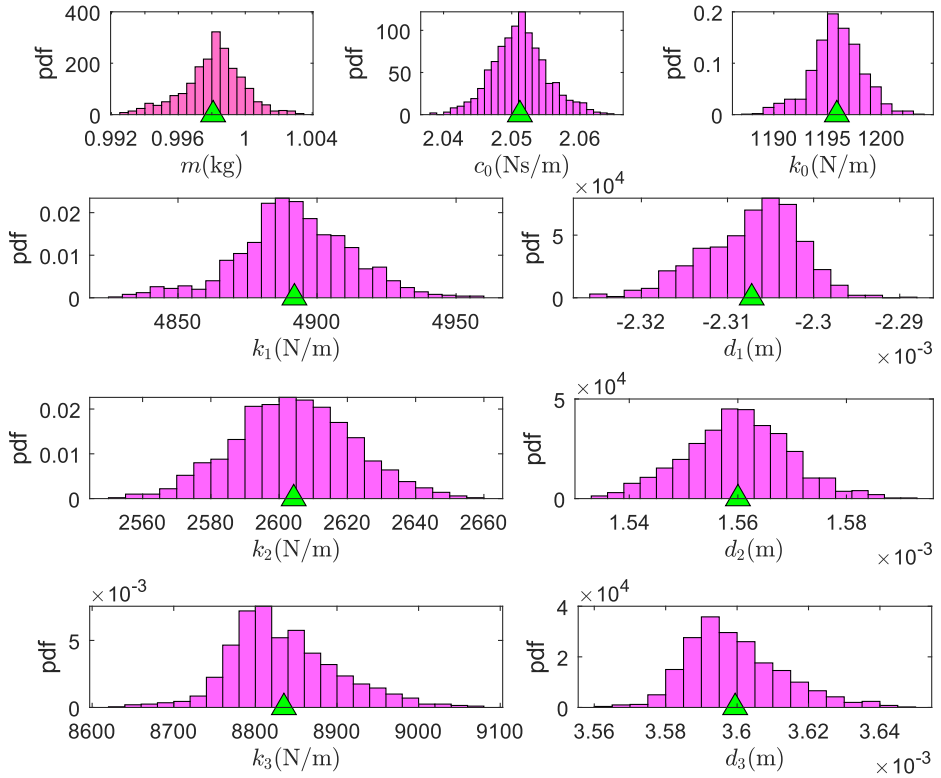


Fig. 16. ABC-based marginal posterior distributions of the parameters of the selected quadlinear stiffness model; the green triangles denote the mean values of the parameters.

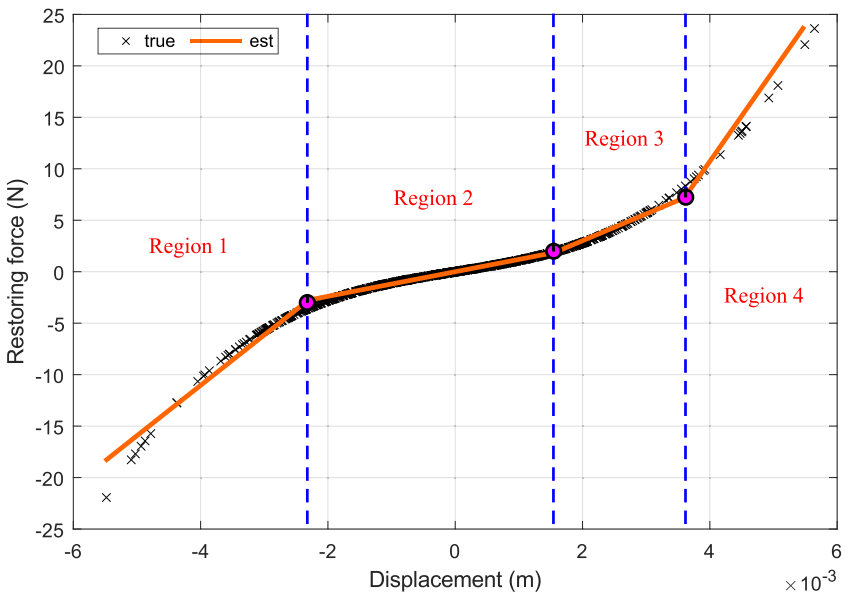


Fig. 17. Comparison of stiffness-based restoring forces for Duffing oscillator (true) with mean-parameter-fitted quadlinear stiffness model (estimated).

of a shock absorber using a simple linear spring-damper model. However, certain experimental studies [59–62] performed on the dynamics of isolated shock absorbers, show that a linear assumption is not entirely justified. This is not very surprising, as automotive dampers are designed to have different damping properties when in compression or rebound.

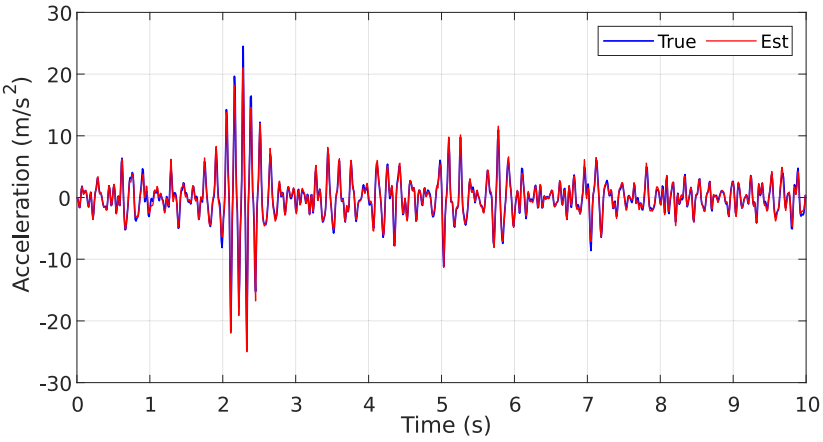


Fig. 18. Comparison of acceleration responses; true response from Duffing oscillator vs estimated response from mean-parameter-fitted quadlinear stiffness model.

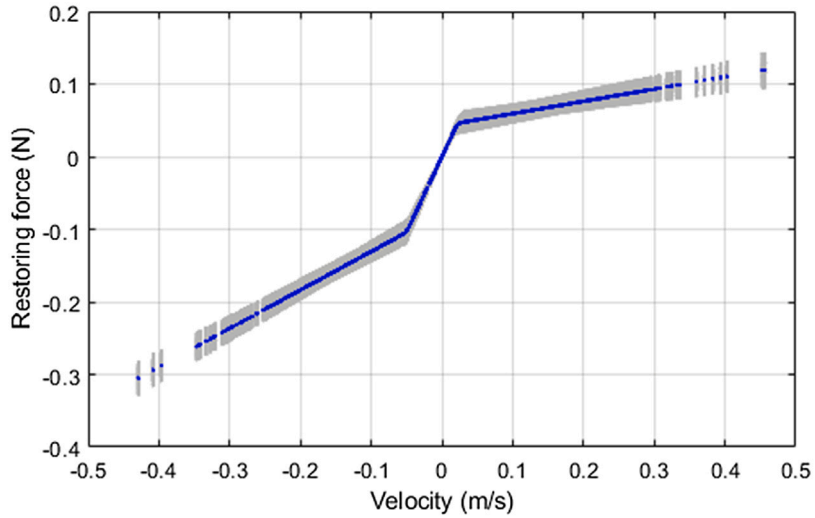


Fig. 19. Scatter plot of the restoring force vs velocity map for the estimated trilinear damping system; the grey points in the background are realisations, and the blue points on the top represent the mean of the realisations.

Recognising that shock absorbers are nonlinear in their damping behaviour, some means are required to characterise the nonlinearity and to represent their behaviour in computer simulations. Lang [59] developed a rigorous analytical model for an absorber by accounting for its internal mechanisms; the model consisted of 87 parameters. Although the results showed good agreement with the experiment, the model is far from general, as it can only be applied to that particular absorber and cannot be applied to one of a different design. Instead of trying to develop a parametric model based on the internal physics of the absorber, a rather straightforward approach is to identify an experimental characterisation of an absorber by fitting the restoring force to the velocity and displacement. Such a characterisation can be done by taking repeated measurements of the restoring force, the velocity and the displacement at different levels of excitation frequency and amplitude, and then plotting the restoring force data against the corresponding velocity and displacement values for fixed frequencies of excitation. The principal benefit from using this type of restoring-force characterisation is that it is a non-parametric in nature and is independent of the *a priori* model of the structure. Identification procedures based on restoring forces have been previously applied to the identification of automotive shock absorbers in a number of publications [63–65].

The experimental data considered in this paper come from a test carried out on a FIAT vehicle shock absorber. The test details involving the apparatus and experimental strategy can be found in [6]. Basically, the shock absorber was constrained to move in one direction to allow an SDOF assumption. The absorber was held fixed at one end to a load cell and a given velocity profile was imposed at the other end; the load cell at the fixed end provided measurements of the internal force. It was found that the contribution of inertial forces were negligible, and hence it was enough to assume a static behaviour for the shock absorber under the testing conditions.

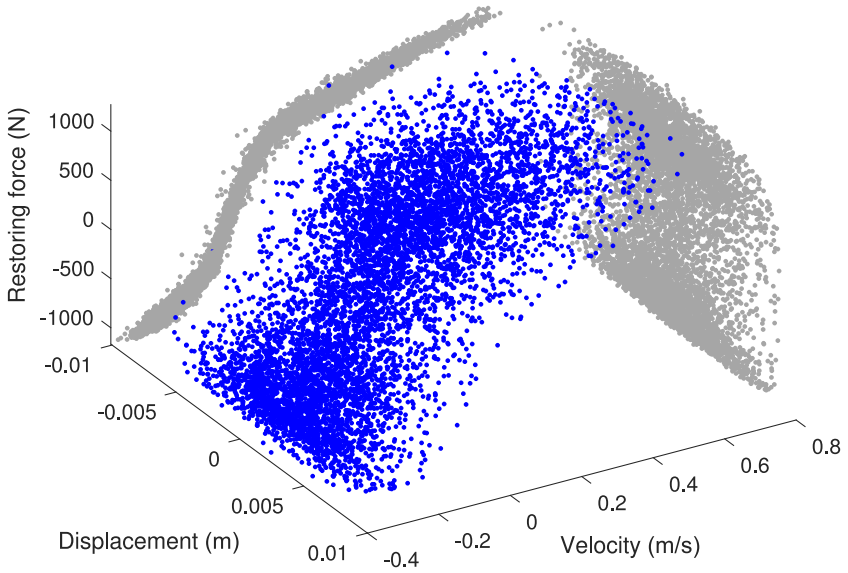


Fig. 20. Original data space of the shock absorber in three dimensions.

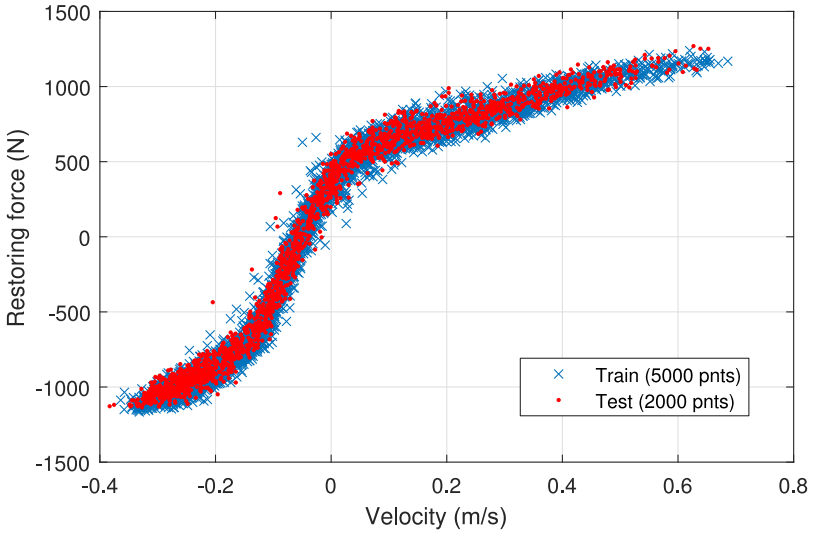


Fig. 21. Training and testing data for shock absorber.

The dataset considered in this study was produced under the influence of a random input profile. As such, the frequency effects can be considered to be 'averaged out' throughout the data. The dataset has 7000 data points featuring three variables: the restoring force, velocity, and displacement; these are presented as a three-dimensional scatter plot in Fig. 20, with two-dimensional projections illustrated alongside to show displacement and velocity dependence of the restoring force. It can be seen from the two-dimensional projections that the restoring force has a fairly distinct dependence on the velocity, while it seems to depend very little on the displacement. Negligible stiffness for the shock absorber used in this case study has been previously reported in [66]. Hence, the displacement variable is ignored and the restoring force is treated solely as a static function of the velocity in this study. The data projected on the velocity axis can be visibly seen (from Fig. 21) to have three (almost linear) regions of operation: the upper tail region represents a compression zone where the damper compresses as a result of (say) braking action, the lower tail region represents rebound where the damper rebounds to its original state after the braking action has ceased, and the mid-body region represents damping behaviour in the normal operating range. Because of these different regions of operation, it is deemed that the shock absorber can be modelled well by piecewise-linear damping.

To commence identification, the entire dataset is segregated in to two parts, a training set of 5000 points for learning an appropriate PWL map, and a test set of 2000 points for performing predictive checks. The training and testing data are plotted

Table 3
Parameter priors for linear (L), bilinear (BL), trilinear (TL), and quadlinear (QL) maps assumed for shock absorber experimental study.

Model	Parameter	Prior	Lower bound	Upper bound
L, BL, TL, QL	c_0 (N s/m)	Log-uniform	10^2	10^4
BL, TL, QL	$c_{L\#}/c_{R\#}$ (N s/m)	Log-uniform	10^2	10^4
BL, TL, QL	$v_{L\#}/v_{R\#}$ (m/s)	Uniform	−0.5	0.8

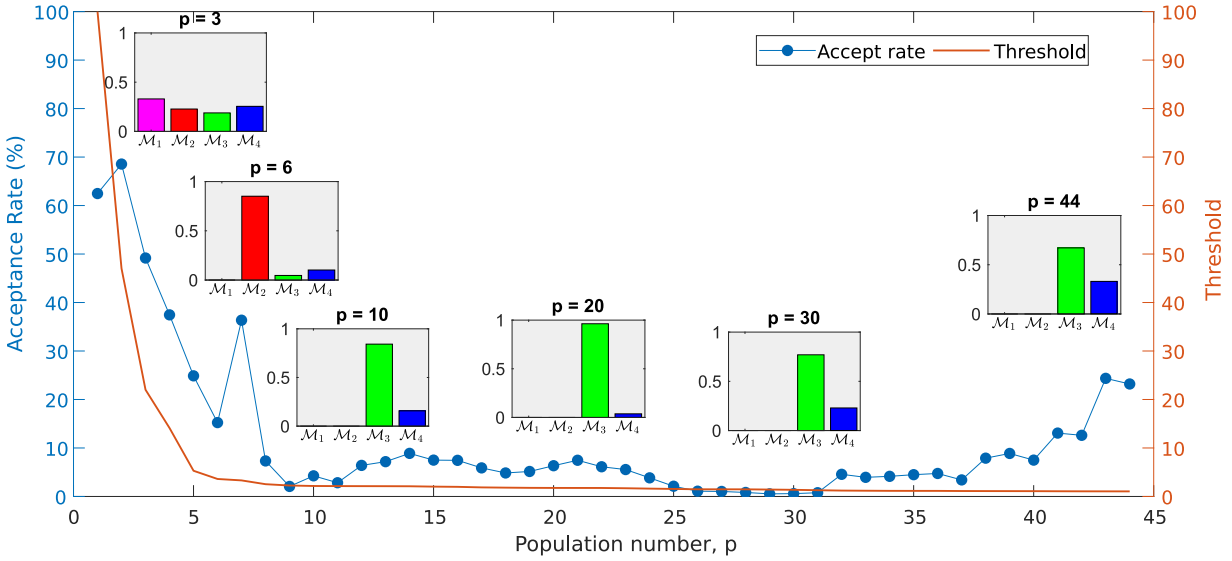


Fig. 22. Convergence of ABC for shock absorber data.

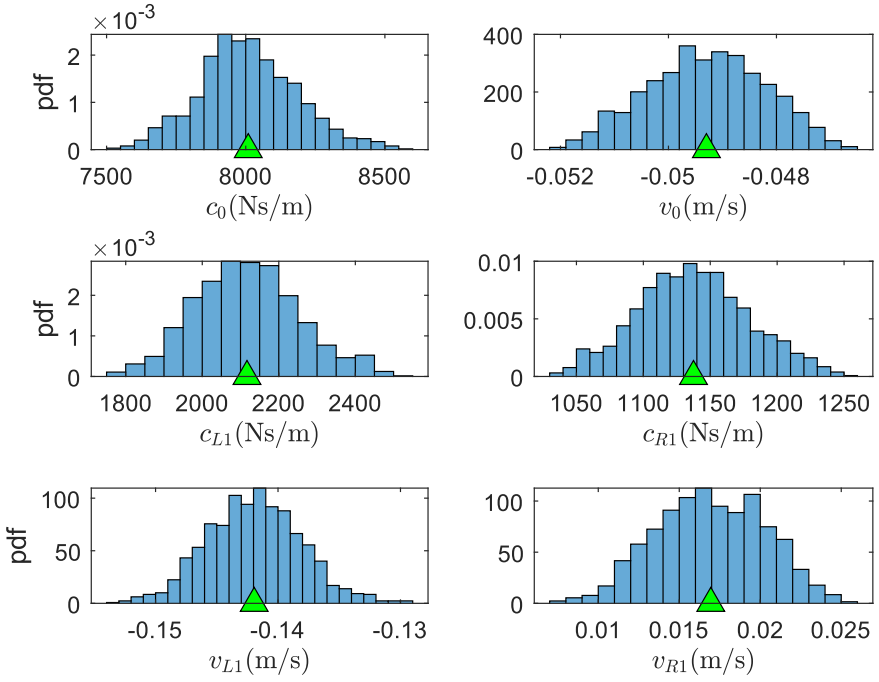
in Fig. 21. Unlike the previous numerical studies that dealt with identification of PWL restoring forces in dynamic SDOF models, in this case study, a 'static' PWL function of velocity is sought for the restoring force of the SDOF shock absorber. In other words, the following process is modelled here,

$$y = f_{\text{PWL}}\left(\dot{z} - v_0\right) \tag{21}$$

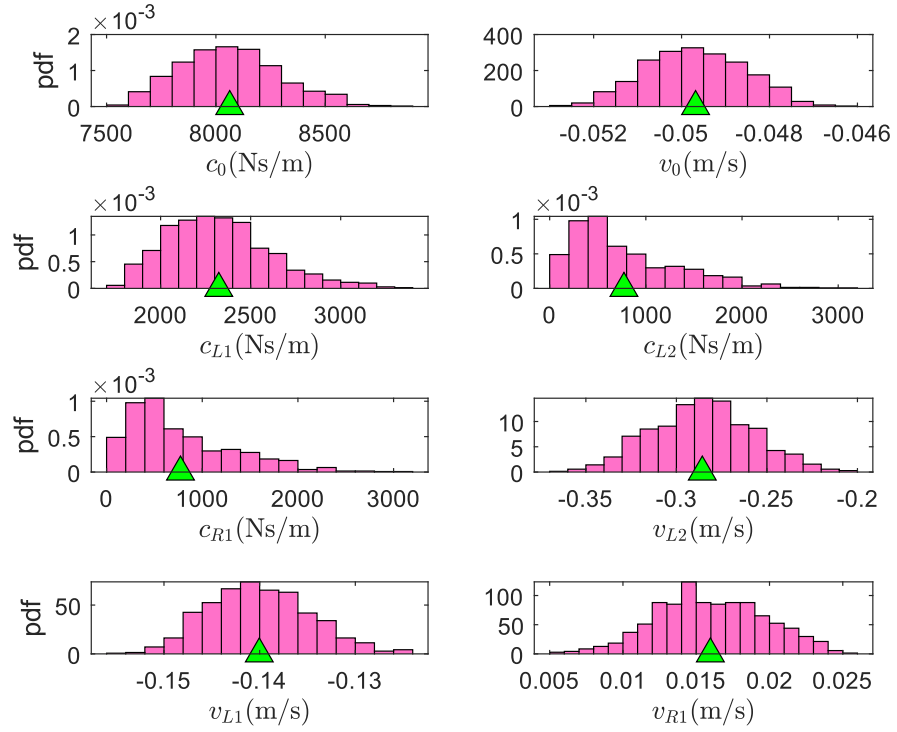
where y is the measured restoring force taken as the output variable, $\dot{z}$ is the measured velocity considered as the input variable, and v_0 is a constant offset variable. The offset variable is introduced to capture the offset of the data from the origin. The parameters to be determined for the static model involve the model order of the PWL map along with the region-specific linear damping coefficients, region partitions, and the constant offset variable. Once again, a pool of four candidate models is assumed, comprising linear, bilinear, trilinear and quadlinear maps of velocity. A uniform model prior was assumed over the four models, while uniform or log-uniform prior distributions were specified for parameters, as are tabulated in Table 3. The best model, along with the posterior distribution over its parameters was inferred using ABC (see Fig. 22).

A total of 2000 particles were used in the ABC algorithm, and the tolerance threshold was initialised at $\epsilon_1 = 100$. The ABC algorithm took 44 populations (and around 35 min) to converge to a trilinear model for the shock absorber, after the linear and bilinear models were eliminated within the first ten populations of the ABC algorithm. Between the two surviving models at convergence, the trilinear model was selected with a larger probability over the quadlinear model. For a better comparison, the posterior distributions of the parameters of both the trilinear and quadlinear models are shown in Fig. 23. Right away, one can see that non-zero offset parameters are obtained which is not surprising as the data do not pass through the origin; this is because of a preload in the experiment. The selected trilinear model has one partition occurring on the negative velocity axis and the other on the positive side, these have been denoted by v_{L1} and v_{R1} respectively. The damping parameter of the middle region c_0 is the highest, followed by that of the rebound region c_{L1} and the compression region c_{R1}; this inference agrees well with the visual depiction of the data. The quadlinear model begets an extra partition along the negative half of the velocity axis, denoted by v_{L2}, with damping parameter c_{L2}. All other parameters of the quadlinear model have very similar mean values as the trilinear model.

It is better to look at the data-fit qualities of the trilinear and quadlinear models for easier understanding of the results. To compare the data-fit, the model-predicted mean restoring forces are plotted against the experimentally-measured restoring forces for the training and test data set in Fig. 24; the model predicted means are obtained via Monte Carlo average of the model predicted responses simulated using posterior parameter samples. It can be seen that the quadlinear model introduces an extra linear region (see the bottom tail portion of Fig. 24(b)). The NMSE values of the corresponding predicted mean restoring forces are also provided on the plots, and they are found to be very close to each other for both the models, with those for the quadlinear model being

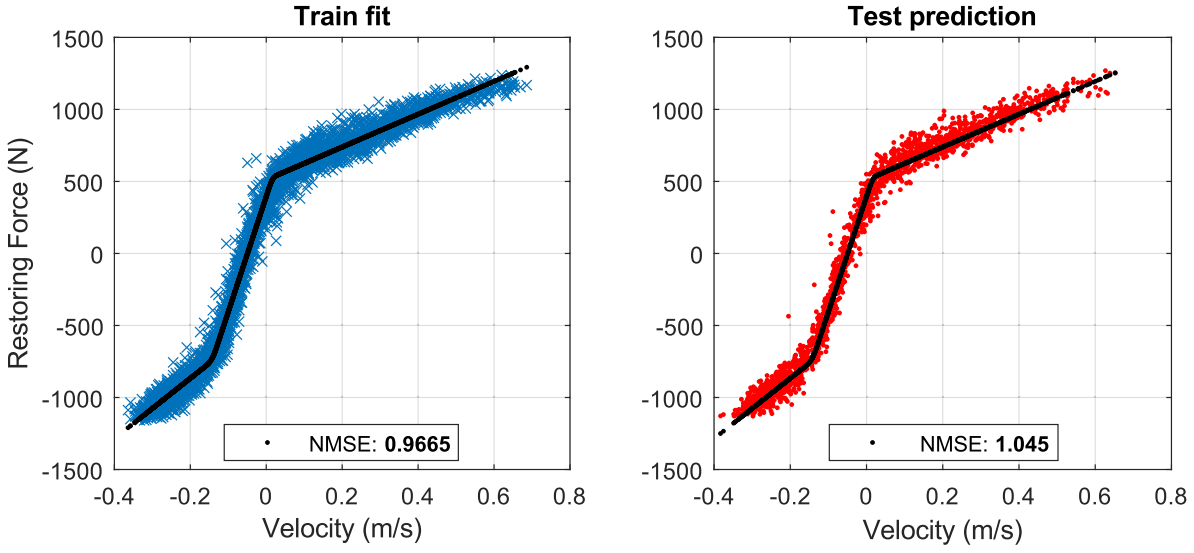


(a) Trilinear model parameters.

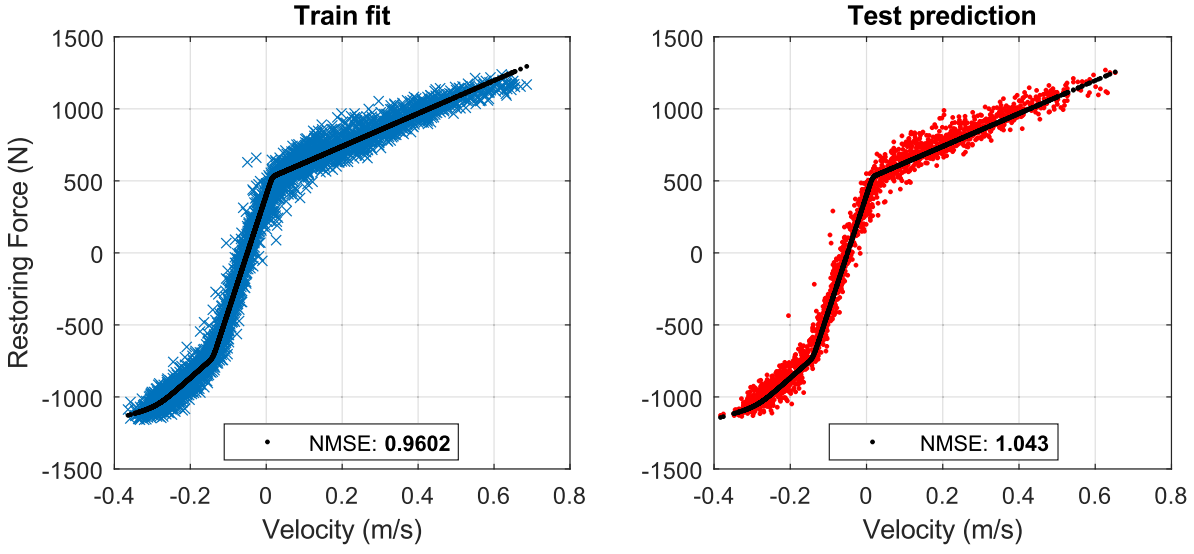


(b) Quadlinear model parameters.

Fig. 23. Histograms of marginal posterior distributions of the parameters of trilinear and quadlinear model for shock absorber; the means of the distributions are depicted by green triangles.



(a) Trilinear model fit.



(b) Quadlinear model fit.

Fig. 24. Training and test data fit for trilinear and quadlinear models.

marginally lower (and hence better) than the trilinear model. Interestingly, with the data-fit being very similar, the Bayesian algorithm selects a lower-order trilinear model over the higher-order quadlinear model, demonstrating once again the principle of parsimony built within the Bayesian inference paradigm.

6. Discussion and conclusion

This paper proposed a conceptually straightforward and easy-to-implement Bayesian procedure for identification of mechanical systems with PWL restoring forces. The nonlinear restoring forces of such systems are characterised by a series of distinct linear regions, the numbers and parameters of which must be determined from the measured data. The number of linear regions is determined using Bayesian model-order selection, whereby an appropriate model order for the PWL model is decided from a set of PWL models with different model orders, and the posterior distributions over the model parameters are determined using Bayesian parameter estimation. These two tasks are performed using a convenient ABC procedure. In particular, the paper uses SDOF systems with PWL nonlinearities in either stiffness or damping to demonstrate the working of the concept. Two numerical examples and an experimental case study are used for demonstrating the performance of the proposed procedure. In all cases, a modified ABC-NS

algorithm has been used to identify a PWL model from among four models with linear, bilinear, trilinear, and quadlinear nature (in either stiffness or damping). The results show that the ABC algorithm is able to select the correct model as well as identify the partition parameters and the regional stiffness/damping parameters. The results show the simplicity of the proposed approach and demonstrate its potential for identifying PWL systems in a Bayesian framework.

In passing, it must be mentioned that all studies considered in this paper dealt with PWL nonlinearity resulting from either stiffness or damping. In principle, the idea can be extended to systems having PWL nonlinearity in both stiffness and damping; in that case, one needs to also include models that have various combinations of stiffness and damping piecewise linearities in the model set. Needless to mention, the parameter search space in such cases would be become larger, with more models and multiple DOFs, leading to slow convergence and increased computational time for the ABC algorithm.

CRediT authorship contribution statement

R. Nayek: Investigation, Methodology, Software, Formal analysis, Writing – original and revised draft. **A.B. Abdessalem:** Software. **N. Dervilis:** Software. **E.J. Cross:** Review and editing, Supervision, Funding acquisition. **K. Worden:** Conceptualisation, Review and editing, Supervision, Funding acquisition.

Declaration of competing interest

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

Data availability

Data will be made available on request.

Acknowledgements

The authors gratefully acknowledge the support of the UK Engineering and Physical Sciences Research Council (EPSRC) via Grant references EP/R006768/1, EP/N018427/1, EP/J016942/1, and EP/S001565/1.

References

[1] J.-N. Lin, R. Unbehauen, Canonical piecewise-linear approximations, IEEE Trans. Circuits Syst. I 39 (8) (1992) 697–699.
[2] W.J. Schwartz, Piecewise linear servomechanisms, Trans. Amer. Inst. Electr. Eng. II: Appl. Ind. 71 (6) (1953) 401–405.
[3] G. Feng, Controller synthesis of fuzzy dynamic systems based on piecewise Lyapunov functions, IEEE Trans. Fuzzy Syst. 11 (5) (2003) 605–612.
[4] W.M.G. van Bokhoven, Piecewise-Linear Modelling and Analysis, Kluwer Technische Boeken Deventer, The Netherlands, 1981.
[5] A.A. Andronov, A.A. Vitt, S.E. Khaikin, Theory of Oscillators: Adiwes International Series in Physics, Vol. 4, Elsevier, 2013.
[6] K. Worden, G.R. Tomlinson, Nonlinearity in Structural Dynamics: Detection, Identification and Modelling, CRC Press, 2019.
[7] S.F. Masri, Analytical and experimental studies of a dynamic system with a gap, J. Mech. Des. 100 (3) (1978) 480–486, http://dx.doi.org/10.1115/1.3453952.
[8] D. Nguyen, S. Noah, C. Kettleborough, Impact behaviour of an oscillator with limiting stops, part I: a parametric study, J. Sound Vibr. 109 (2) (1986) 293–307.
[9] S. Natsiavas, Periodic response and stability of oscillators with symmetric trilinear restoring force, J. Sound Vib. 134 (2) (1989) 315–331.
[10] M.S. Allen, H. Sumali, D.S. Epp, Piecewise-linear restoring force surfaces for semi-nonparametric identification of nonlinear systems, Nonlinear Dynam. 54 (1) (2008) 123–135.
[11] V.M. Veliov, M.I. Krastanov, Controllability of piecewise linear systems, Systems Control Lett. 7 (5) (1986) 335–341.
[12] S. Natsiavas, Stability of piecewise linear oscillators with viscous and dry friction damping, J. Sound Vib. 217 (3) (1998) 507–522.
[13] E. Chicurel-Uziel, Exact, single equation, closed-form solution of vibrating systems with piecewise linear springs, J. Sound Vib. 245 (2) (2001) 285–301.
[14] J.M. Gonçalves, A. Megretski, M.A. Dahleh, Global analysis of piecewise linear systems using impact maps and surface Lyapunov functions, IEEE Trans. Automat. Control 48 (12) (2003) 2089–2106.
[15] J.-C. Ji, A.Y.T. Leung, Periodic and chaotic motions of a harmonically forced piecewise linear system, Int. J. Mech. Sci. 46 (12) (2004) 1807–1825.
[16] S. Billings, W. Voon, Piecewise linear identification of non-linear systems, Internat. J. Control 46 (1) (1987) 215–235.
[17] V. Verdult, M. Verhaegen, Subspace identification of piecewise linear systems, in: 2004 43rd IEEE Conference on Decision and Control, Vol. 4, CDC, IEEE, 2004, pp. 3838–3843.
[18] S. Paoletti, A.L. Juloski, G. Ferrari-Trecate, R. Vidal, Identification of hybrid systems: A tutorial, Eur. J. Control 13 (2–3) (2007) 242–260.
[19] A. Garulli, S. Paoletti, A. Vicino, A survey on switched and piecewise affine system identification, IFAC Proc. Vol. 45 (16) (2012) 344–355.
[20] A.L. Juloski, S. Weiland, W. Heemels, A Bayesian approach to identification of hybrid systems, IEEE Trans. Automat. Control 50 (10) (2005) 1520–1533.
[21] J. Wågberg, F. Lindsten, T.B. Schön, Bayesian nonparametric identification of piecewise affine ARX systems, IFAC-PapersOnLine 48 (28) (2015) 709–714.
[22] D. Piga, A. Bemporad, A. Benavoli, Rao-Blackwellized sampling for batch and recursive Bayesian inference of Piecewise Affine models, Automatica 117 (2020) 109002.
[23] K.S. Chan, H. Tong, On estimating thresholds in autoregressive models, J. Time Series Anal. 7 (3) (1986) 179–190.
[24] P. Julián, M. Jordán, A. Desages, Canonical piecewise-linear approximation of smooth functions, IEEE Trans. Circuits Syst. I 45 (5) (1998) 567–571.
[25] L. Breiman, Hinging hyperplanes for regression, classification, and function approximation, IEEE Trans. Inform. Theory 39 (3) (1993) 999–1013.
[26] T.A. Johansen, B.A. Foss, Identification of non-linear system structure and parameters using regime decomposition, Automatica 31 (2) (1995) 321–326.
[27] E.A. Heredia, G.R. Arce, Piecewise linear system modeling based on a continuous threshold decomposition, IEEE Trans. Signal Process. 44 (6) (1996) 1440–1453.
[28] A. Bemporad, A. Garulli, S. Paoletti, A. Vicino, A bounded-error approach to piecewise affine system identification, IEEE Trans. Automat. Control 50 (10) (2005) 1567–1580.

[29] G. Ferrari-Trecate, M. Muselli, D. Liberati, M. Morari, A clustering technique for the identification of piecewise affine systems, Automatica 39 (2) (2003) 205–217.
[30] H. Nakada, K. Takaba, T. Katayama, Identification of piecewise affine systems based on statistical clustering technique, Automatica 41 (5) (2005) 905–913.
[31] V. Breschi, D. Piga, A. Bemporad, Piecewise affine regression via recursive multiple least squares and multicategory discrimination, Automatica 73 (2016) 155–162.
[32] T. Toni, D. Welch, N. Strelkowa, A. Ipsen, M.P. Stumpf, Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems, J. R. Soc. Interface 6 (31) (2009) 187–202.
[33] A.B. Abdessalem, N. Dervilis, D. Wagg, K. Worden, Model selection and parameter estimation of dynamical systems using a novel variant of approximate Bayesian computation, Mech. Syst. Signal Process. 122 (2019) 364–386.
[34] O.V. Gendelman, Targeted energy transfer in systems with non-polynomial nonlinearity, J. Sound Vib. 315 (3) (2008) 732–745.
[35] X. Wang, G. Zheng, Two-step transfer function calculation method and asymmetrical piecewise-linear vibration isolator under gravity, J. Vib. Control 22 (13) (2016) 2973–2991.
[36] A. Abdelkefi, R. Vasconcellos, F. Marques, M. Hajj, Modeling and identification of freeplay nonlinearity, J. Sound Vib. 331 (8) (2012) 1898–1907.
[37] MATLAB version 9.9.0.1570001 (R2020b), The Mathworks, Inc., Natick, Massachusetts, 2020.
[38] J.L. Beck, K.-V. Yuen, Model selection using response measurements: Bayesian probabilistic approach, J. Eng. Mech. 130 (2) (2004) 192–203.
[39] K.-V. Yuen, Bayesian Methods for Structural Dynamics and Civil Engineering, John Wiley & Sons, 2010.
[40] K. Worden, J. Hensman, Parameter estimation and model selection for a class of hysteretic systems using Bayesian inference, Mech. Syst. Signal Process. 32 (2012) 153–169.
[41] R. Nayek, R. Fuentes, K. Worden, E.J. Cross, On spike-and-slab priors for Bayesian equation discovery of nonlinear dynamical systems via sparse linear regression, Mech. Syst. Signal Process. 161 (2021) 107986.
[42] I. Hazra, M.D. Pandey, N. Manzana, Approximate Bayesian computation (ABC) method for estimating parameters of the Gamma process using noisy data, Reliab. Eng. Syst. Saf. 198 (2020) 106780.
[43] P. Mason, Approximate Bayesian Computation of the occurrence and size of defects in advanced gas-cooled nuclear reactor boilers, Reliab. Eng. Syst. Saf. 146 (2016) 21–25.
[44] M.A. Beaumont, B. Rannala, The Bayesian revolution in genetics, Nature Rev. Genet. 5 (4) (2004) 251–261.
[45] K.M. Cremers, Stock return predictability: A Bayesian model selection perspective, Rev. Financ. Stud. 15 (4) (2002) 1223–1249.
[46] C. Mark, C. Metzner, L. Lautscham, P.L. Strissel, R. Strick, B. Fabry, Bayesian model selection for complex dynamic systems, Nature Commun. 9 (1) (2018) 1–12.
[47] W.A. Link, R.J. Barker, Bayesian Inference: With Ecological Applications, Academic Press, 2009.
[48] J.B. Johnson, K.S. Omland, Model selection in ecology and evolution, Trends Ecol. Evol. 19 (2) (2004) 101–108.
[49] S. Brooks, A. Gelman, G. Jones, X.-L. Meng, Handbook of Markov Chain Monte Carlo, CRC Press, 2011.
[50] S. Oladyshkin, W. Nowak, The connection between Bayesian inference and information theory for model selection, information gain and experimental design, Entropy 21 (11) (2019) 1081.
[51] S. Chib, I. Jeliazkov, Marginal likelihood from the Metropolis–Hastings output, J. Amer. Statist. Assoc. 96 (453) (2001) 270–281.
[52] J. Ching, Y.-C. Chen, Transitional Markov chain Monte Carlo method for Bayesian model updating, model class selection, and model averaging, J. Eng. Mech. 133 (7) (2007) 816–832.
[53] P.J. Green, Reversible jump Markov chain Monte Carlo computation and Bayesian model determination, Biometrika 82 (4) (1995) 711–732.
[54] S.A. Sisson, Y. Fan, M. Beaumont, Handbook of Approximate Bayesian Computation, CRC Press, 2018.
[55] J.-M. Marin, P. Pudlo, C.P. Robert, R.J. Ryder, Approximate Bayesian computational methods, Stat. Comput. 22 (6) (2012) 1167–1180.
[56] J.K. Pritchard, M.T. Seielstad, A. Perez-Lezaun, M.W. Feldman, Population growth of human Y chromosomes: a study of Y chromosome microsatellites, Mol. Biol. Evol. 16 (12) (1999) 1791–1798.
[57] P. Marjoram, J. Molitor, V. Plagnol, S. Tavaré, Markov chain Monte Carlo without likelihoods, Proc. Natl. Acad. Sci. 100 (26) (2003) 15324–15328.
[58] M. Chiachio, J.L. Beck, J. Chiachio, G. Rus, Approximate Bayesian computation by subset simulation, SIAM J. Sci. Comput. 36 (3) (2014) A1339–A1358.
[59] H.H. Lang, A Study of the Characteristics of Automotive Hydraulic Dampers at High Stroking Frequencies, University of Michigan, 1977.
[60] P. Hagedorn, J. Wallaschek, On equivalent harmonic and stochastic linearization for nonlinear shock-absorbers, in: Nonlinear Stochastic Dynamic Engineering Systems, Springer, 1988, pp. 23–32.
[61] G. Genta, P. Campanile, An approximated approach to the study of motor vehicle suspensions with nonlinear shock absorbers, Meccanica 24 (1) (1989) 47–57.
[62] J. Wallaschek, Dynamics of non-linear automobile shock-absorbers, Int. J. Non-Linear Mech. 25 (2–3) (1990) 299–308.
[63] G. Belingardi, P. Campanile, Improvement of the shock absorber dynamic simulation by the restoring force mapping method, in: Proceedings of the 15th International Modal Analysis Conference, Leuven, Belgium, 1990, pp. 441–454.
[64] C. Surace, K. Worden, G. Tomlinson, On the non-linear characteristics of automotive shock absorbers, Proc. Inst. Mech. Eng. D 206 (1) (1992) 3–16.
[65] S. Duym, R. Stiens, K. Reybrouck, Fast parametric and nonparametric identification of shock absorbers, in: Proceedings of the International Seminar on Modal Analysis, Katholieke Uuniversiteit Leuven, 1996, pp. 1157–1170.
[66] T. Zhang, R. Barthorpe, K. Worden, On treed Gaussian processes and piecewise-linear NARX modelling, Mech. Syst. Signal Process. 144 (2020) 106877.