PHILOSOPHICAL TRANSACTIONS A

rsta.royalsocietypublishing.org

Opinion piece

**Cite this article:** Green PL, Worden K. 2015 Bayesian and Markov chain Monte Carlo methods for identifying nonlinear systems in the presence of uncertainty. *Phil. Trans. R. Soc. A* **373**: 20140405.
http://dx.doi.org/10.1098/rsta.2014.0405

Accepted: 22 May 2015

One contribution of 11 to a theme issue 'A field guide to nonlinearity in structural dynamics'.

**Subject Areas:**
mechanical engineering

**Keywords:**
nonlinear, system identification, model updating, Bayesian

**Author for correspondence:**
P. L. Green
e-mail: p.l.green@liverpool.ac.uk

†Present address: Institute for Risk and Uncertainty, Centre for Engineering Sustainability, School of Engineering, University of Liverpool, Liverpool L69 3GQ, UK.

Electronic supplementary material is available at http://dx.doi.org/10.1098/rsta.2014.0405 or via http://rsta.royalsocietypublishing.org.

THE ROYAL SOCIETY PUBLISHING

# Bayesian and Markov chain Monte Carlo methods for identifying nonlinear systems in the presence of uncertainty

P. L. Green† and K. Worden

Department of Mechanical Engineering, University of Sheffield, Mappin Street, Sheffield S1 3JD, UK

In this paper, the authors outline the general principles behind an approach to Bayesian system identification and highlight the benefits of adopting a Bayesian framework when attempting to identify models of nonlinear dynamical systems in the presence of uncertainty. It is then described how, through a summary of some key algorithms, many of the potential difficulties associated with a Bayesian approach can be overcome through the use of Markov chain Monte Carlo (MCMC) methods. The paper concludes with a case study, where an MCMC algorithm is used to facilitate the Bayesian system identification of a nonlinear dynamical system from experimentally observed acceleration time histories.

## 1. Introduction

System Identification (SI) is a technique of considerable importance within the discipline of structural dynamics. In the absence of a complete physics-based description of a system or structure, SI can provide the missing pieces of information that allow the formulation of a descriptive or predictive model. When the structure of interest has linear dynamical behaviour, the problem of SI is well established, to the extent that authoritative text books and monographs exist [1,2]. In the case of linear dynamical systems, it is usually sufficient to consider sets of linear second-order differential equations (*modal* models) or first-order differential equations (*state-space* models) as the appropriate mathematical model structure. In that case, the SI problem is largely reduced to determining the correct number of equations and

the numerical parameters in the model. Unfortunately, most structures will, in reality, display nonlinear characteristics to some extent, and the SI problem for nonlinear structures and systems is by no means solved. One of the main problems in nonlinear SI is the number and variety of possible model structures that arise once the variety of possible nonlinearities is taken into account [3,4].

It is not necessary here to provide a detailed classification of nonlinear SI models and approaches; however, it will prove useful to give a higher level breakdown of model structures based on their motivation. Predictive models can be divided into three classes: *white*, *grey* and *black-box* models.

> *White-box* models are taken here to be those whose equations of motion have been derived completely from the underlying physics of the problem of interest and in which the model parameters have direct physical meanings. Finite-element models constitute one sub-class of such models.
> *Black-box* models are, by contrast, usually formed by adopting a parametrized class of models with some universal approximation property and learning the parameters from measured data; in such a model, like a neural network, the parameters will not generally carry any physical meaning.
> *Grey-box* models, as the name suggests, are usually a hybrid of the first two types above. They are commonly formed by taking a basic core motivated by known physics and then adding a black-box component with approximation properties suited to the problem of interest. A good example of a grey-box model is the Bouc–Wen model of hysteresis. In the Bouc–Wen model, a mass-spring-damper core is supplemented by an extra state-space equation which allows versatile approximation of a class of hysteresis loops [5,6].

In all of these cases, measured data from the system or structure of interest can be used in order to determine any unknown aspects of the model, e.g. any necessary undetermined parameters can be estimated. The use of measured data often means that uncertainty is introduced into the problem. There are two main sources of uncertainty caused by consideration of measured data. The first source is measurement noise; in general, other sources (*noise*) will contribute to measurements of the variable of interest and the direct distinction between signal and noise will be impossible. The second problem is encountered when a measured variable is itself a random process. In this case, only specific finite realizations of the process of interest can be measured; variability between realizations leads to variability between parameter estimates and thus gives rise to uncertainty.

In the past, the SI practitioner would generally implement the classical algorithms (i.e. least-squares minimization) as an exercise in linear algebra and would usually treat the resulting set of crisp parameter estimates as determining 'the model'. Even if a covariance matrix were extracted, the user would usually use this only to provide confidence intervals or 'error bars' on the parameters; predictions would still be made using the crisp parameters produced by the algorithm. Such approaches do not fully accommodate the fact that a given set of measured data, subject to the sources of uncertainty discussed above, may be consistent with a number of different parametric models. It is now becoming clear—largely as a result of the pioneering work of James Beck and colleagues and more recently from guidance from the machine learning community—that a more robust approach to parameter estimation, and also model selection, can be formulated on the basis of Bayesian principles for probability and statistics. Among the potential advantages offered by a Bayesian formulation are the estimation procedure will return parameter distributions rather than parameters; predictions can be made by integrating over all parameters consistent with the data, weighted by their probabilities; evidence for a given model structure can be computed, leading to a principled means of model selection.

Adoption of Bayesian methods first became widespread in the context of the identification of black-box models; the methods have recently begun to occupy a central position within the machine learning community [7,8]. Bayesian methods for training multi-layer perceptron neural

networks are a good example of this trend [9]; the Gaussian process model is also achieving wide popularity [10]. Most machine learning algorithms, like the neural networks and Gaussian processes already mentioned, are used to learn static relationships between variables; however, they can easily be used to model dynamical processes by assuming a NARX or NARMAX form for the mapping of interest [11,12]. A recent example of nonlinear system identification using Gaussian process NARX models can be found in [13]; this study is of interest because it shows how physical insight might be gained from the black-box GP NARX models. There has also been a body of work concerned with Bayesian parameter estimation for polynomial NARMAX models, a recent contribution can be found in [14].

In the context of white-box models, and in particular within the nonlinear SI community, the use of Bayesian methods has not been so widespread; however, their pedigree is as long. One can find references to Bayesian methods in a monograph on parameter estimation from 1974 [15], and dating from the same year is perhaps the first paper on Bayesian methods for structural dynamic SI [16]. To date, the most systematic and extensive development of Bayesian SI is the result of the work of James Beck and his various collaborators. Beck's early work on statistical system identification is summarized in [17] and his transition to a Bayesian framework is given in [18]. This paper uses a Laplace approximation to remove the need to evaluate intractable high-dimensional integrals. Later, Beck & Au [19] introduce a *Markov chain Monte Carlo* (MCMC) method as a more general means of computing response quantities of interest represented by high-dimensional integrals. Bayesian methods of model selection are discussed in [20], and the paper also discusses the possibility of marginalizing over different model *classes*. A recent contribution [21] discusses identification and model selection for a type of hysteretic system model—the Masing model. Staying with hysteresis models, the paper [22] considers how MCMC can be used for Bayesian estimation of Bouc–Wen models and discusses a simple model selection statistic. Two recent developments which are of interest are the introduction of probability logic for Bayesian SI [23] and a method for potentially reducing computational expense for MCMC by selecting the most informative training data [24]. Bayesian methods for the system identification of differential equations have also been the subject of recent interest in the context of *systems biology* [25,26] and show considerable promise in the context of structural dynamics.

At this point, it is appropriate to define some notation. Here $\mathcal{M}$ is used to represent a model structure. $\boldsymbol{\theta} \in \mathbb{R}^{N_\theta}$ is then used to represent the vector of parameters within that model which requires estimation. Finally, $\mathcal{D}$ is used to denote a set of observations which one has made about the system of interest, i.e. the measured data. As an example, one may consider the case study which is shown in §5, where one is attempting to create a white-box model of a dynamical system whose response is thought to be greatly influenced by friction effects. In this case, $\mathcal{M}$ represents the hypothesized equation of motion of the system. Here $\boldsymbol{\theta}$ represents the parameters within the equation of motion which require estimation—in the current example, this includes terms which modulate the level of viscous damping and friction in the system. The data, $\mathcal{D}$, consist of a time history of acceleration measurements which have been taken during a dynamic test. The basic idea of the Bayesian approach to identification is that, by repeatedly applying Bayes' theorem, one can assess the probability of a set of parameters $\boldsymbol{\theta}$ as well as a model structure $\mathcal{M}$ conditional on the data $\mathcal{D}$ using

$$p(\boldsymbol{\theta} \mid \mathcal{D}, \mathcal{M}) = \frac{p(\mathcal{D} \mid \boldsymbol{\theta}, \mathcal{M}) p(\boldsymbol{\theta} \mid \mathcal{M})}{p(\mathcal{D} \mid \mathcal{M})} \tag{1.1}$$

and

$$P(\mathcal{M} \mid \mathcal{D}) = \frac{p(\mathcal{D} \mid \mathcal{M}) P(\mathcal{M})}{p(\mathcal{D})}, \tag{1.2}$$

respectively, where

$$p(\mathcal{D} \mid \mathcal{M}) = \int p(\mathcal{D} \mid \boldsymbol{\theta}, \mathcal{M}) p(\boldsymbol{\theta} \mid \mathcal{M}) \, \mathrm{d}\boldsymbol{\theta} \tag{1.3}$$

is a normalizing constant which ensures that $p(\boldsymbol{\theta} \mid \mathcal{D}, \mathcal{M})$ integrates to unity. This is referred to here as the 'marginal likelihood' but can also be described as the 'model evidence' (because, as

is shown in §2, it can provide evidence for candidate model structures). With equation (1.1), one converts an *a priori* probability density for the parameters $\boldsymbol{\theta}$ into a posterior density having seen the data $\mathcal{D}$. If one desires a point estimate of the parameters, the usual course of action is to choose that which maximizes the posterior probability $p(\boldsymbol{\theta} \mid \mathcal{D}, \mathcal{M})$. Now, as the data $\mathcal{D}$ is a constant of the identification problem, one is reduced to maximizing $p(\mathcal{D} \mid \boldsymbol{\theta}, \mathcal{M})p(\boldsymbol{\theta} \mid \mathcal{M})$. It is often the case at this point, that an uninformative constant (and hence improper) prior $p(\boldsymbol{\theta} \mid \mathcal{M})$ is chosen, and this reduces the problem to that of maximizing $p(\mathcal{D} \mid \boldsymbol{\theta}, \mathcal{M})$, which is simply the likelihood of the data. The *maximum a posteriori* (MAP) estimate thus becomes maximum likelihood. If one were to further assume that the distribution of any measurement noise was Gaussian (with some extra conditions), the problem essentially becomes one of minimizing a least-squares cost/error function. The sequence of assumptions and approximations discussed above clearly loses much of the benefit of adopting a Bayesian approach in the first place. This paper will emphasize the benefits of a 'full' Bayesian methodology.

The layout of the paper is as follows. In §2, the fundamental principles behind a Bayesian approach to system identification are described and the benefits of using MCMC algorithms within a Bayesian framework are emphasized. Sections 3 and 4 are devoted to the description of various MCMC algorithms which can be used to address the issues of parameter estimation and model selection, respectively. These sections are not intended to be a thorough review but, instead, focus on those algorithms which have proved to be particularly useful and/or are based on unique concepts and methodologies.[1] Finally, in §5, a case study is used to demonstrate how MCMC can be used within a Bayesian framework to generate robust models of nonlinear dynamical systems.

## 2. Bayesian system identification

The problem of SI is easily stated: given measured data from a structure or system, how does one infer the equations of motion which 'generated' the data. This problem is not at all easy to solve; it is essentially an inverse problem of the second kind and can be extremely ill-posed even if the underlying equations are assumed to be linear in the parameters of interest [3]. Furthermore, the 'solution' may not even be unique. If the equations of motion are not linear in the parameters of interest, the difficulties multiply. Another issue is concerned with *confidence* in derived parameter estimates. This issue is a result of the fact that measurements or data from a system will, in reality, almost always be contaminated by random noise. Given a set of data $\mathcal{D} = \{(x_i, y_i), i = 1, \ldots, N\}$ of sampled system inputs $x_i$ and outputs $y_i$, if there is no measurement noise, an identification algorithm should yield a deterministic estimate of the system parameters $\boldsymbol{\theta}$,

$$\boldsymbol{\theta} = id(\mathcal{D}), \tag{2.1}$$

where the function *id* represents the application of the identification algorithm to the data $\mathcal{D}$. Now, if noise $\epsilon(t)$ is present on the input or output data (or both), $\boldsymbol{\theta}$ will become a random vector conditioned on the data. In this context, one no longer wishes to find an *estimate* of $\boldsymbol{\theta}$, but rather to specify one's belief in its value. If it is assumed that the noise is Gaussian with (unknown) standard deviation $\sigma_\epsilon$, then the parameter $\sigma_\epsilon$ can be subsumed into $\boldsymbol{\theta}$, and inferred along with the model parameters. In probabilistic terms, instead of equation (2.1) one now has

$$\boldsymbol{\theta} \sim p(\boldsymbol{\theta} \mid \mathcal{D}, \mathcal{M}), \tag{2.2}$$

where $\mathcal{M}$ represents the choice of model.

The usual objective of system identification is to provide a predictive model, i.e. one which can estimate or predict system outputs if a different system input is provided. In the probabilistic context described above, the best that one could do is to determine a predictive distribution.

[1]For a more detailed description of various MCMC algorithms, the technical report by Neal is recommended [27] while, for the interested reader, it is worth noting that [28] is an impressive Python resource for coding MCMC schemes.

Suppose a new input sequence $\boldsymbol{x}^*$ were applied to the system, one would wish to determine the density for the predicted outputs

$$\boldsymbol{y}^* \sim p(\boldsymbol{y}^* \mid \boldsymbol{x}^*, \boldsymbol{\theta}, \mathcal{D}, \mathcal{M}) \tag{2.3}$$

noting all the dependencies.[2] The mean of this distribution would give the 'best' estimates for the predictions and the covariance would allow one to establish confidence intervals for them. However, one notes the presence of the parameter vector $\boldsymbol{\theta}$. In practice, one might use the $\boldsymbol{\theta}$ value corresponding to the mean or the mode of the posterior parameter distribution; however, a truly Bayesian viewpoint on the prediction would require one to marginalize over the parameter estimates, i.e. to derive

$$p(\boldsymbol{y}^* \mid \boldsymbol{x}^*, \mathcal{D}, \mathcal{M}) = \int p(\boldsymbol{y}^* \mid \boldsymbol{x}^*, \boldsymbol{\theta}, \mathcal{M}) p(\boldsymbol{\theta} \mid \mathcal{D}, \mathcal{M}) \, \mathrm{d}\boldsymbol{\theta}, \tag{2.4}$$

where $p(\boldsymbol{\theta} \mid \mathcal{D}, \mathcal{M})$—the posterior parameter distribution—is given by equation (1.1).

This is a very powerful idea: allowing for a fixed model structure, *one is making predictions using an entire set of parameters consistent with the training data*, with each point in the space of parameters weighted according to its probability given the data. In practice, there are considerable problems in implementing the full Bayesian approach, i.e. performing the intractable integral (2.4). One of the main advantages of using MCMC algorithms is that they allow one to *generate samples* from the posterior parameter distribution, even when the geometry of $p(\boldsymbol{\theta} \mid \mathcal{D}, \mathcal{M})$ is complex and its probability density is very concentrated relative to the prior. These samples can then be used as part of Monte Carlo simulations, allowing one to propagate one's parameter uncertainties without evaluating equation (2.4).

Another potential advantage of a Bayesian approach is that it may be possible to assess the relative evidence for a number of competing model structures. Suppose one believes that the true model structure is one of a finite number $\{\mathcal{M}_i, i = 1, \ldots, M\}$ (the discussion here will closely follow [25]). In principle, one could imagine computing the probability of observing the data $p(\mathcal{D} \mid \mathcal{M}_i)$, given the particular model structure. If this quantity were available then, by defining a prior probability $P(\mathcal{M}_i)$ on each model structure, one could use equation (1.2) to select the model with the highest probability. Even more in the spirit of Bayesian inference, one could marginalize over *all possible* model structures weighted according to their probability; in terms of prediction, one would have

$$p(\boldsymbol{y}^* \mid \boldsymbol{x}^*, \mathcal{D}) = \sum_{i=1}^{M} p(\boldsymbol{y}^* \mid \boldsymbol{x}^*, \mathcal{M}_i, \mathcal{D}) P(\mathcal{M}_i \mid \mathcal{D}). \tag{2.5}$$

Furthermore, if one appeals to Bayes theorem in the form of equation (1.2) and assumes equal priors on the models, one arrives at a comparison ratio or *Bayes factor*

$$B_{ij} = \frac{P(\mathcal{M}_i \mid \mathcal{D})}{P(\mathcal{M}_j \mid \mathcal{D})} = \frac{p(\mathcal{D} \mid \mathcal{M}_i)}{p(\mathcal{D} \mid \mathcal{M}_j)}, \tag{2.6}$$

which weights the evidence for two models in terms of marginal likelihoods of the data given the models.

The Bayesian approach to model selection is particularly attractive as the marginal likelihood rewards models for being high fidelity while also penalizing them for being overly complex. By automatically embodying Ockham's Razor with regard to model selection, it follows that the adoption of a Bayesian approach can help to prevent overfitting. An intuitive explanation of this property is provided by MacKay [7], where it is suggested that a complex model will be capable of replicating a larger range of predictions than a simple model with relatively few parameters. As the probability density function $p(\mathcal{D} \mid \mathcal{M})$ must always be normalized it follows that, in a region where both models are able to replicate the same data, the marginal likelihood will be larger for the simpler model (figure 1). It is in this way that $p(\mathcal{D} \mid \mathcal{M})$ can be used to *provide evidence* for candidate model structures.

[2]Without loss of generality, the reader can regard $\boldsymbol{x}^*$ and $\boldsymbol{y}^*$ as a set of samples, a set of variables or both.

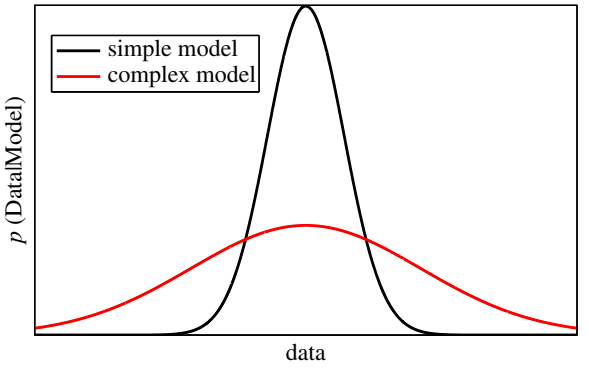


**Figure 1.** The embodiment of Ockham's Razor in the marginal likelihood (original explanation described in [7]).

For a more detailed explanation, an information theoretic analysis of the marginal likelihood was originally discussed by Beck & Yuen [29] before being generalized in [21]. Noting that $\int p(\boldsymbol{\theta} \mid \mathcal{D}, \mathcal{M}_i)\, \mathrm{d}\boldsymbol{\theta} = 1$, it follows that the logarithm of the marginal likelihood can be written as

$$\ln[p(\mathcal{D} \mid \mathcal{M}_i)] = \ln[p(\mathcal{D} \mid \mathcal{M}_i)] \int p(\boldsymbol{\theta} \mid \mathcal{D}, \mathcal{M}_i)\, \mathrm{d}\boldsymbol{\theta} \tag{2.7}$$

$$= \int \ln[p(\mathcal{D} \mid \mathcal{M}_i)] p(\boldsymbol{\theta} \mid \mathcal{D}, \mathcal{M}_i)\, \mathrm{d}\boldsymbol{\theta} \tag{2.8}$$

$$= \int \ln\left[\frac{p(\mathcal{D} \mid \boldsymbol{\theta}, \mathcal{M}_i) p(\boldsymbol{\theta} \mid \mathcal{M}_i)}{p(\boldsymbol{\theta} \mid \mathcal{D}, \mathcal{M}_i)}\right] p(\boldsymbol{\theta} \mid \mathcal{D}, \mathcal{M}_i)\, \mathrm{d}\boldsymbol{\theta} \tag{2.9}$$

therefore

$$\ln[p(\mathcal{D} \mid \mathcal{M}_i)] = \int \ln[p(\mathcal{D} \mid \boldsymbol{\theta}, \mathcal{M}_i)] p(\boldsymbol{\theta} \mid \mathcal{D}, \mathcal{M}_i)\, \mathrm{d}\boldsymbol{\theta} - \int \ln\left[\frac{p(\boldsymbol{\theta} \mid \mathcal{D}, \mathcal{M}_i)}{p(\boldsymbol{\theta} \mid \mathcal{M}_i)}\right] p(\boldsymbol{\theta} \mid \mathcal{D}, \mathcal{M}_i)\, \mathrm{d}\boldsymbol{\theta}. \tag{2.10}$$

The first term in the above equation is the posterior mean of the log-likelihood which is a measure of the average data fit of model $\mathcal{M}_i$. It follows that achieving a good fit to the training data will provide evidence for a candidate model structure. The second term in equation (2.10) represents the relative entropy between the prior and posterior. The marginal likelihood therefore penalizes models which are 'complex' where, a complex model is defined as that which is able to extract large amounts of information about the parameters $\boldsymbol{\theta}$ from the data $\mathcal{D}$. It is important to note that the marginal likelihood is a function of the prior $p(\boldsymbol{\theta} \mid \mathcal{M}_i)$—it is possible to alter the evidence for $\mathcal{M}_i$ by altering the prior distribution while maintaining the same model structure.

Unfortunately, the marginal likelihoods themselves (equation (1.3)) are often analytically intractable and numerically challenging because of their high-dimensional nature [26]. While one can resort to less informative model selection indicators which are simpler to compute (for example, as used in [22], a Bayesian generalization of the Akaike Information Criterion (AIC) [30] known as the *Deviance Information Criterion* (DIC)), it will be shown in §4 that there now exist MCMC methods which can be used to estimate the marginal likelihoods of different models/generate samples directly from $P(\mathcal{M} \mid \mathcal{D})$.

As a final comment on the issue of model selection, it should be noted that there are already examples of the use of Bayesian strategies for model selection in the structural dynamics literature, the 'Ockham factor' defined in [19] being one of these. In [31], the authors use a Bayesian model screening approach in order to determine the appropriate nonlinear terms to include in a system model. The book [32] discusses Bayesian model selection in some detail.

## 3. Markov chain Monte Carlo for the posterior parameter distribution

The first set of algorithms reviewed here are those which are designed to generate samples from the posterior parameter distribution (equation (1.1)), while circumventing the need to evaluate the marginal likelihood (equation (1.3)). These methods involve the creation of an ergodic Markov chain—a Markov chain whose probability distribution tends to a functional form which is independent of time—which evolves through the parameter space (see [33] for a comprehensive discussion on the convergence of Markov chains). Simply stated, MCMC involves 'forcing' the stationary distribution of the Markov chain to be equal to (or at least proportional to) some target distribution such that, having allowed the chain to become stationary, it is effectively generating samples from the target. In the context of this section, the target distribution is $p(\boldsymbol{\theta} \mid \mathcal{D}, \mathcal{M})$.

Throughout the following text, one's target distribution is written as $\pi(\boldsymbol{\theta}) = \pi^*(\boldsymbol{\theta})/Z$ where $\pi^*$ is used to denote the unnormalized distribution and $Z$ is the corresponding normalizing constant.

### (a) The Metropolis algorithm

The Metropolis algorithm was originally proposed in [34], was later generalized by Hastings in [35] and is one of the most established MCMC algorithms. With a Markov chain whose current state is $\boldsymbol{\theta}^{(i)}$, the first step of the Metropolis algorithm involves probabilistically proposing a new state $\boldsymbol{\theta}'$. This proposal is generated from a probability density function $q(\boldsymbol{\theta}' \mid \boldsymbol{\theta}^{(i)})$ which is conditional on the current state of the chain and, for the sake of simplicity, will be assumed to be symmetrical and centered on $\boldsymbol{\theta}^{(i)}$. This proposal then becomes the new state of the Markov chain *with probability*

$$\min\left\{1, \frac{\pi(\boldsymbol{\theta}')}{\pi(\boldsymbol{\theta}^{(i)})}\right\} = \min\left\{1, \frac{\pi^*(\boldsymbol{\theta}')}{\pi^*(\boldsymbol{\theta}^{(i)})}\right\}. \tag{3.1}$$

If accepted, the new state of the Markov chain is $\boldsymbol{\theta}^{(i+1)} = \boldsymbol{\theta}'$ otherwise $\boldsymbol{\theta}^{(i+1)} = \boldsymbol{\theta}^{(i)}$. The probability of making the transition from some state $\boldsymbol{\theta}$ to the region $\boldsymbol{\theta}' \, \mathrm{d}\boldsymbol{\theta}'$ can be written as

$$T(\boldsymbol{\theta}' \mid \boldsymbol{\theta}) \, \mathrm{d}\boldsymbol{\theta}' = q(\boldsymbol{\theta}' \mid \boldsymbol{\theta}) \, \mathrm{d}\boldsymbol{\theta}' \min\left\{1, \frac{\pi^*(\boldsymbol{\theta}')}{\pi^*(\boldsymbol{\theta}^{(i)})}\right\}. \tag{3.2}$$

The first point to note is that, by using such a transition, one satisfies the condition known as *detailed balance*:

$$\pi(\boldsymbol{\theta}) T(\boldsymbol{\theta}' \mid \boldsymbol{\theta}) = \pi(\boldsymbol{\theta}') T(\boldsymbol{\theta} \mid \boldsymbol{\theta}') \implies \int \pi(\boldsymbol{\theta}) T(\boldsymbol{\theta}' \mid \boldsymbol{\theta}) \, \mathrm{d}\boldsymbol{\theta} = \pi(\boldsymbol{\theta}'), \tag{3.3}$$

(noting that $\int T(\boldsymbol{\theta} \mid \boldsymbol{\theta}') \, \mathrm{d}\boldsymbol{\theta} = 1$) which shows that the stationary distribution of the Markov chain is equal to the target. The second point to note is that, to evaluate the acceptance probability (equation (3.1)), one does not need to know the normalizing constant $Z$. This makes the Metropolis algorithm particularly well suited to Bayesian inference problems as it allows one to sample from $p(\boldsymbol{\theta} \mid \mathcal{D}, \mathcal{M})$ without having to evaluate the marginal likelihood.

Figure 2*a* shows an example of the Metropolis algorithm generating samples from a two-dimensional target distribution. It is clear that the Markov chain must go through a transitionary period (known as the 'burn in') before it converges to its stationary distribution—the samples generated during this time will need to be discarded. Figure 2*b*,*c* shows what can happen if one selects proposal densities which are have too small or too large variance. With a small proposal density, the Markov chain will take a very long time to converge to its stationary distribution while, with a large proposal density, the majority of the proposed states are rejected and the resulting samples from the Markov chain are highly correlated. The efficiency of the Metropolis algorithm is therefore highly dependent on the tuning of the proposal density $q$. A final point worth noting is that, when using the Metropolis algorithm to generate samples from $p(\boldsymbol{\theta} \mid \mathcal{D}, \mathcal{M})$, the use of large data sets may cause numerical issues when one is evaluating the acceptance probability. This can easily be overcome by simply using the logarithm of equation (3.1), such that one then only needs to evaluate the logarithm of the posterior parameter distribution.

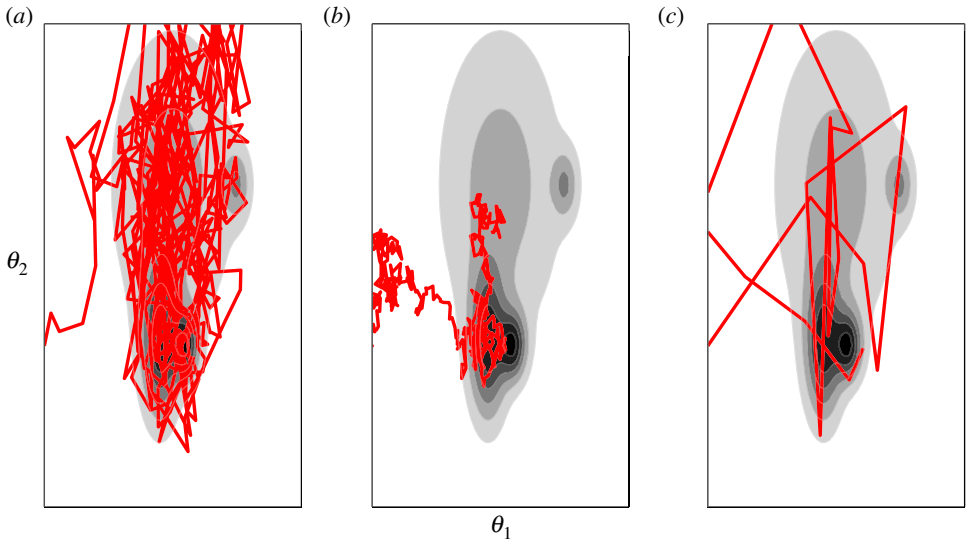


**Figure 2.** (*a–c*) Sampling from a two-dimensional distribution using the Metropolis algorithm.

## (b) Hybrid Monte Carlo

Hybrid Monte Carlo (also referred to as Hamiltonian Monte Carlo (HMC)) [36] is an MCMC method which is designed to explore parameter spaces more efficiently that the Metropolis algorithm. To facilitate an understanding of HMC, one must envisage that the $i$th element of the parameter vector $\boldsymbol{\theta}$ represents the displacement of a particle in the $i$th direction. One then introduces a vector of momenta $\boldsymbol{p} \in \mathbb{R}^{N_\theta}$ (recalling that $N_\theta$ is the number of parameters to be estimated) such that the Hamiltonian of the system is $H = K(\boldsymbol{p}) + V(\boldsymbol{\theta})$ (where $K$ and $V$ are the kinetic and potential energies, respectively). Introducing a fictitious 'time' variable $\tau$, the dynamics of the system can then be evolved through $\tau$ according to

$$\frac{\mathrm{d}\boldsymbol{\theta}}{\mathrm{d}\tau} = \boldsymbol{p} \quad \text{and} \quad \frac{\mathrm{d}\boldsymbol{p}}{\mathrm{d}\tau} = -\nabla V. \tag{3.4}$$

Writing $p = |\,\boldsymbol{p}\,|$, the key here is to define the kinetic and potential energies as

$$K = \frac{p^2}{2} \quad \text{and} \quad V = -\ln(\pi^*(\boldsymbol{\theta})) \tag{3.5}$$

such that

$$H = \frac{p^2}{2} - \ln(\pi^*) \quad \text{and} \quad \exp(-H) = \exp\left(-\frac{p^2}{2}\right)\pi^*. \tag{3.6}$$

As a result, if one targets the distribution $\exp(-H)$ and then simply omits the samples of $p$, one will be left with samples of $\boldsymbol{\theta}$ from the target $\pi$.

To generate a candidate state $\{\boldsymbol{p}', \boldsymbol{\theta}'\}$ from the current state $\boldsymbol{\theta}^{(i)}$, one must first generate an initial momenta $\boldsymbol{p} \sim \mathcal{N}(0, 1)$ (noting that this is actually a direct sample from $\exp(-H)$). The Hamiltonian of this current state is written as $H^{(i)}$. The system is then allowed to evolve according to equation (3.4) for a certain amount of 'time', until it reaches some state $\{\boldsymbol{p}', \boldsymbol{\theta}'\}$ which has Hamiltonian $H'$. As with the Metropolis algorithm, this state is then accepted *with probability*

$$\min\left\{1, \frac{\exp(-H')}{\exp(-H^{(i)})}\right\}. \tag{3.7}$$

From the above equation, it is clear that, if the dynamics of the system are modelled perfectly, then the new state will always be accepted (as the Hamiltonian must remain constant). In practice,

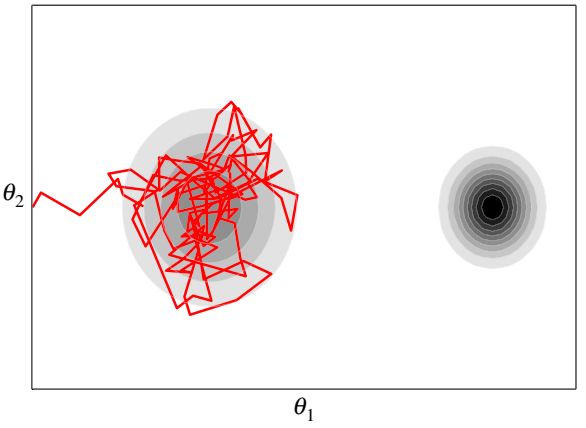


**Figure 3.** MCMC becoming stuck in a 'local trap'.

however, the evolution of the system according to equation (3.4) must usually be conducted numerically (usually using finite difference estimates of $\nabla V$), and so the Hamiltonian will alter as a result of numerical error. In [36], it is shown that one can still obey detailed balance (and therefore generate samples from $\exp(-H)$) so long as the dynamics of the system are reversible. This can be guaranteed by using the 'leapfrog' numerical integration technique (see [27] for more details).

The ability of Hybrid Monte Carlo to 'generate momentum' during the proposal process can allow it to conduct efficient explorations of the parameter space relative to the Metropolis algorithm (reference [27] gives a clear explanation of this physical analogy). However, its successful implementation sometimes requires careful tuning of parameters in the leapfrog algorithm, as well as the parameters which dictate how long the system must evolve before a proposal is generated. Furthermore, the need to repeatedly evaluate the posterior distribution to obtain estimates of $\nabla V$ can make the algorithm computationally expensive. It has however, been successfully applied to various structural dynamics problems in [37–39].

## (c) Simulated annealing

Figure 3 shows a Markov chain which has become 'stuck' in a local region of high probability density which has prevented (or at least reduced the probability of) it reaching the globally highest region of probability density. Such local regions are referred to as 'local traps'. Simulated annealing, which was originally proposed as an optimization algorithm in [40], is a powerful method which not only provides information which can be used to tune the Metropolis algorithm, but which also reduces the probability of local trapping. This is because it initially begins with proposals which allow large steps in the parameter space—the proposal variance is then reduced, thus allowing refinement within a given mode.

Consider the situation where, using the Metropolis algorithm, one aims to generate samples from the target $\pi^* = \exp(-h(\boldsymbol{\theta}))$. With simulated annealing, one proceeds by targeting the sequence of distributions

$$\pi_j^* = \exp(-\beta_j h(\boldsymbol{\theta})), \quad j = 1, 2, \ldots \quad \text{where } \beta_1 < \beta_2 < \cdots. \tag{3.8}$$

The parameter $\beta$ is referred to as the inverse 'temperature' (thus drawing an analogy between $\pi_j$ and a Boltzmann distribution). One begins by targeting a distribution with a low value of $\beta$ (high temperature) before steadily 'cooling' the system until $\beta = 1$, simulating the process of annealing. This essentially means that the 'fine details' of the target distribution are introduced gradually and that, at high temperatures, the Markov chain is able to traverse the parameter space

relatively freely compared to when $\beta = 1$. It is this which allows the Markov chain to easily escape local traps and converge to the globally optimum region of the parameter space.

When attempting to generate samples from the posterior parameter distribution specifically, a more sophisticated version of simulated annealing involves targeting

$$\pi_j^* = p(\mathcal{D} \mid \boldsymbol{\theta}, \mathcal{M})^{\beta_j} p(\boldsymbol{\theta} \mid \mathcal{M}), \quad 0 = \beta_0 < \beta_1 < \cdots < \beta_M = 1, \tag{3.9}$$

which allows one to facilitate a smooth transition from the prior to the posterior parameter distributions. The strictly increasing sequence of $\beta$ vales (the 'annealing schedule') is crucial to the success of the algorithm. Annealing too fast can result in the Markov chain becoming stuck in local traps while annealing to slowly will incur unnecessarily large computational costs. While there are algorithms which feature adaptive annealing schedules (e.g. [27,41–43]), they are not reviewed in this paper.

Simulated annealing has proved to be a very successful methodology and has directly influenced the development of algorithms such as Simulated Tempering [44,45], Exchange Monte Carlo [46], adaptive variants of the Metropolis algorithm [19], Transitional MCMC [42], AIMS [43] and Data Annealing [47] (where the annealing process is instigated through the gradual introduction of data points into the likelihood).

## 4. Markov chain Monte Carlo for the posterior model distribution

The algorithms described in §3 are all designed to generate samples from the posterior parameter distribution while circumventing the need to evaluate the marginal likelihood. While these methods are undoubtedly powerful, they do not allow one to evaluate the posterior model distribution and so can only be used to evaluate what is commonly referred to as the 'first level of Bayesian inference'. Here, three algorithms are described which, using quite different methods, can be used to address *both* levels of inference—parameter estimation and model selection.

### (a) Transitional Markov chain Monte Carlo

The Transitional MCMC (TMCMC) algorithm was presented in [42]. As with simulated annealing, it involves targeting the sequence of distributions defined by equation (3.9).

Consider the case where one has $N$ samples from $\pi_j$, which are denoted $\boldsymbol{\theta}_j^{(i)}$, $i = 1, \ldots, N$ (when TMCMC is initiated these would be samples from the prior). One then uses a technique very similar to importance sampling to target the next distribution $\pi_{j+1}$. Specifically, one calculates the 'importance weights' and 'normalized importance weights' of each sample using

$$w_j^{(i)} = \frac{\pi_{j+1}^*(\boldsymbol{\theta}_j^{(i)})}{\pi_j^*(\boldsymbol{\theta}_j^{(i)})} = p(\mathcal{D} \mid \boldsymbol{\theta}_j^{(i)}, \mathcal{M})^{\beta_{j+1}-\beta_j} \quad \text{and} \quad \hat{w}_j^{(i)} = \frac{w_j^{(i)}}{\sum_i w_j^{(i)}}, \tag{4.1}$$

respectively. With standard importance sampling, one would then 'resample' by assigning $\theta_{j+1}^{(i)} = \theta_j^{(i)}$ with probability $\hat{w}_j^{(i)}$. If left to continue in this manner, the algorithm would suffer from the well-known degeneracy problem (a phenomenon often associated with the particle filter), and the set of samples would become dominated by relatively few, highly weighted samples.

To overcome this issue, TMCMC considers each resampled value $\theta_{j+1}^{(i)}$ as the starting point of a Markov chain. The Markov chains evolve according to the Metropolis algorithm, each targeting $\pi_{j+1}$. The probability that a Markov chain will 'grow' is determined by the normalized importance weight of its initial sample. The advantage of this approach is that, by simultaneously growing Markov chains in high probability regions of $\pi_{j+1}$, one is able to generate samples from distributions with multiple modes. Once the Markov chains have generated a sufficient number of samples from $\pi_{j+1}$, the process is simply repeated until one is left with samples from $p(\boldsymbol{\theta} \mid \mathcal{D}, \mathcal{M})$.

With regard to estimating the marginal likelihood, if one denotes $\boldsymbol{w}_j$ as a vector of importance weights, then from the property that

$$\mathrm{E}[\boldsymbol{w}_j] = \int \frac{\pi^*_{j+1}}{\pi^*_j}\pi_j \,\mathrm{d}\boldsymbol{\theta} = \int \frac{\pi^*_{j+1}}{\pi^*_j}\frac{\pi^*_j}{Z_j}\,\mathrm{d}\boldsymbol{\theta} = \frac{Z_{j+1}}{Z_j} \tag{4.2}$$

it follows that $p(\mathcal{D}\mid\mathcal{M}) = Z_0\ \mathrm{E}[\boldsymbol{w}_0]\mathrm{E}[\boldsymbol{w}_1]\cdots\mathrm{E}[\boldsymbol{w}_{M-1}]$. As a result, by estimating the expected value of the importance weights at each stage of the algorithm, one can approximate the marginal likelihood.

As well as being able to sample from distributions with multiple modes and estimate the marginal likelihood, its reliance on the simultaneous growth of multiple Markov chains makes TMCMC suitable for parallel processing [48]. Furthermore, it is also shown in [42] that, by selecting values of $\beta$ which ensure that the coefficient of variation of the importance weights remain within predefined limits, the algorithm is also able to generate an adaptive annealing schedule which prevents large changes in the geometry of the target distribution occurring.

As a result of these benefits, TMCMC has become a popular algorithm which has been applied to many engineering problems (e.g. [49–53]) and has helped to inspire other algorithms such as AIMS [43] (which is not discussed here).

## (b) Reversible jump Markov chain Monte Carlo

Reversible jump MCMC (RJMCMC) [54,55] is unique in that it does not attempt to evaluate $P(\mathcal{M}\mid\mathcal{D})$ separately for each individual model structure; instead, MCMC is used to target the distribution $\pi(\boldsymbol{\theta},\mathcal{M}) = P(\boldsymbol{\theta},\mathcal{M}\mid\mathcal{D})$. This 'joint posterior' is the product of the posterior parameter and model distributions and can be expanded using Bayes' theorem to gain

$$P(\boldsymbol{\theta},\mathcal{M}\mid\mathcal{D}) = \frac{P(\mathcal{D}\mid\boldsymbol{\theta},\mathcal{M})P(\boldsymbol{\theta},\mathcal{M})}{P(\mathcal{D})} = \frac{P(\mathcal{D}\mid\boldsymbol{\theta},\mathcal{M})P(\boldsymbol{\theta}\mid\mathcal{M})P(\mathcal{M})}{P(\mathcal{D})}. \tag{4.3}$$

In the following text, $\boldsymbol{x}$ is used to denote the current state of a Markov chain. In the context of algorithms which generate samples from the posterior parameter distribution only, $\boldsymbol{x}$ is simply equal to $\boldsymbol{\theta}$, the current position of the Markov chain in the parameter space. For RJMCMC—where the Markov chain is being used to generate samples from the joint posterior (equation (4.3))—the current state of the Markov chain is $\boldsymbol{x} = \{\boldsymbol{\theta}_k, k\}$, where $k$ indexes the model structures and $\boldsymbol{\theta}_k$ represents the current parameter estimates of the $k$th model.

Guaranteeing that the resulting Markov chain will obey detailed balance (and therefore have a stationary distribution equal to the target) is complicated by the fact that different model structures will usually feature different numbers of parameters and, as result, RJMCMC involves the propagation of a Markov chain across parameter spaces of *varying dimension*.

Consider the situation where the algorithm's current state is $\boldsymbol{x}\in\mathbb{R}^n$ and the state $\boldsymbol{x}'\in\mathbb{R}^{n'}$ is proposed via

$$\boldsymbol{x}' = h(\boldsymbol{x},\boldsymbol{u}), \tag{4.4}$$

where $h$ is a user-defined function and $\boldsymbol{u}$ is an auxiliary random variable sampled from a distribution $g(\boldsymbol{u})$; as will be shown, the variable $\boldsymbol{u}$ allows the preservation of dimension of the Markov chain and ultimately assures detailed balance.

In [54,55], it is demonstrated that detailed balance will hold if

$$\pi(\boldsymbol{x})g(\boldsymbol{u})\alpha(\boldsymbol{x},\boldsymbol{x}') = \pi(\boldsymbol{x}')g'(\boldsymbol{u}')\alpha(\boldsymbol{x}',\boldsymbol{x})\left|\frac{\partial(\boldsymbol{x}',\boldsymbol{u}')}{\partial(\boldsymbol{x},\boldsymbol{u})}\right|, \tag{4.5}$$

where $\boldsymbol{u}'$ is a random variable, sampled from $g'(\boldsymbol{u}')$, which will facilitate the proposal of $\boldsymbol{x}$ from $\boldsymbol{x}'$. Equation (3.3) highlights an issue which RJMCMC must overcome if it is to generate samples from the joint posterior. Specifically, for detailed balance to hold, it must be ensured that the mapping from $(\boldsymbol{x},\boldsymbol{u})$ to $(\boldsymbol{x}',\boldsymbol{u}')$ is diffeomorphic. To address this a 'dimension matching' procedure is employed. In the current example, where $\boldsymbol{x}\in\mathbb{R}^n$ and $\boldsymbol{x}'\in\mathbb{R}^{n'}$, this involves ensuring that $u\in\mathbb{R}^r$ and $u'\in\mathbb{R}^{r'}$ are chosen such that $n+r = n'+r'$.

It is then relatively easy to show that, to ensure that equation (4.5) is satisfied, one must set the acceptance probability, $\alpha$, equal to

$$\alpha(\boldsymbol{x}, \boldsymbol{x}') = \min\left\{1, \frac{\pi(\boldsymbol{x}')g'(\boldsymbol{u}')}{\pi(\boldsymbol{x})g(\boldsymbol{u})}\left|\frac{\partial(\boldsymbol{x}', \boldsymbol{u}')}{\partial(\boldsymbol{x}, \boldsymbol{u})}\right|\right\}. \tag{4.6}$$

Practically, before RJMCMC can be used one has to outline a set of possible 'moves' which collectively can allow the Markov chain to transition from any state $\boldsymbol{x}$ to any other state $\boldsymbol{x}'$ (perhaps in more than one step). This usually involves constructing a 'birth' and 'death' move which, respectively, allow the Markov chain to propose states in models with more or less parameters than the current model structure. Finally, one must also define an 'update' move which allows the Markov chain to explore the parameter space of the current model structure (this can be achieved simply by using the Metropolis algorithm). In each iteration of RJMCMC, each of these moves are attempted with a user-defined probability.

The obvious advantage of RJMCMC is that is allows one to investigate the probability of all the competing model structures simultaneously—one does not have to obtain estimates of $P(\mathcal{M} \mid \mathcal{D})$ for each model separately. Examples of RJMCMC being applied to mechanical engineering problems can be found in [56–58].

## (c) Nested sampling

Nested sampling [59,60] is unlike the other algorithms discussed here as, rather than generating samples from the posterior parameter distribution, it is specifically designed to estimate the marginal likelihood. It involves noting that the required integral (equation (1.3)) can be viewed as

$$\int_{\lambda} (\text{Likelihood} = \lambda) \times (\text{Prior mass associated with Likelihood} = \lambda). \tag{4.7}$$

One then defines $X$ as the being the prior mass enclosed within the contour where the likelihood is larger than $\lambda$:

$$X = \int_{p(\mathcal{D}|\boldsymbol{\theta},\mathcal{M})>\lambda} p(\boldsymbol{\theta} \mid \mathcal{M})\, \mathrm{d}\boldsymbol{\theta}. \tag{4.8}$$

It is then assumed that there exists a function $\lambda = L(X)$ which, if one is given $X$, will reveal the corresponding value of $\lambda$. When $X = 0$ it is clear that there will be no prior mass within the contour defined by $p(\mathcal{D} \mid \boldsymbol{\theta}, \mathcal{M}) = \lambda$ (implying that $\lambda$ must be larger than $\max_{\theta}\{p(\mathcal{D} \mid \boldsymbol{\theta}, \mathcal{M})\}$. $L(X)$ is then a decreasing[3] function of $X$ until, when $X = 1$, $\lambda$ must be equal to zero as the entire prior mass is now contained in the contour defined by $p(\mathcal{D} \mid \boldsymbol{\theta}, \mathcal{M}) = \lambda$.

From equation (4.8), it follows that $\mathrm{d}X$ represents the prior probability mass $p(\boldsymbol{\theta} \mid \mathcal{M})\, \mathrm{d}\boldsymbol{\theta}$ associated with the contour where the likelihood is equal to $\lambda$, ergo

$$P(\mathcal{M} \mid \mathcal{D}) = \int p(\mathcal{D} \mid \boldsymbol{\theta}, \mathcal{M}) p(\boldsymbol{\theta} \mid \mathcal{M})\, \mathrm{d}\boldsymbol{\theta} = \int_0^1 L(X)\, \mathrm{d}X \tag{4.9}$$

and a difficult multi-dimensional integral has been reduced to a simple one-dimensional integral.

The Nested sampling algorithm begins with $N$ samples $\{\boldsymbol{\theta}^{(1)}, \ldots, \boldsymbol{\theta}^{(N)}\}$ from the prior. One then locates the sample $\boldsymbol{\theta}^{(k)}$ which resulted in the lowest value of the likelihood (denoted $\lambda_{\text{min}}^{(1)}$). The corresponding value of $X$ (written as $X^{(1)}$) is estimated according to

$$X^{(1)} = \frac{N-1}{N}. \tag{4.10}$$

One then replaces $\boldsymbol{\theta}^{(k)}$ with a sample which has been generated from the prior and is subject to the constraint that the resulting likelihood is larger than $\lambda_{\text{min}}^{(1)}$ (one may try to achieve this using the Metropolis algorithm or other MCMC methods). This procedure is repeated until a series of

[3]Technically, it is assumed to be strictly decreasing such that there is a 1–1 relationship between $X$ and $\lambda$—see [60] for more details.

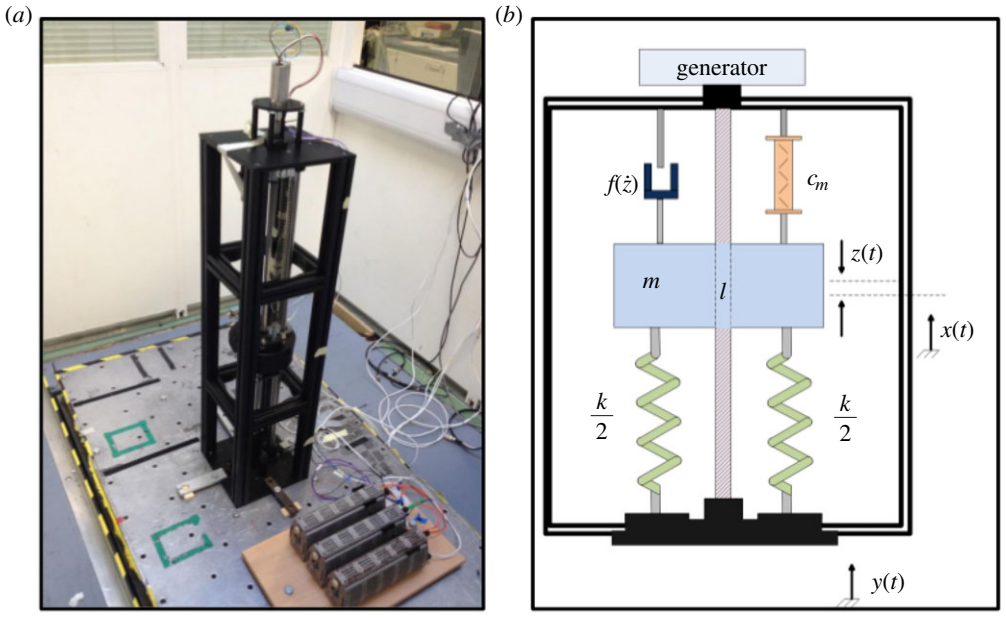


**Figure 4.** (*a*) Test rig and (*b*) schematic of rotational energy harvester at the University of Southampton Institute of Sound and Vibration.

$X$ values and the corresponding $L(X) = \lambda$ have been obtained—standard numerical methods can then be used to estimate $\int_0^1 L(X)\,\mathrm{d}X$.

While nested sampling is an elegant algorithm, the need to generate samples from the prior subject to constraints on the likelihood can be difficult and, as such, it has not been widely adopted within the context of structural dynamics (although it was used in [61]). It is included here as it provides an interesting method of estimating the marginal likelihood which is fundamentally different from TMCMC or RJMCMC and, with further development, may become more ubiquitous within structural dynamics.

## 5. Case study

The case study shown here was originally conducted as part of a collaborative project with the University of Southampton (full findings are published in [62]); it is included here as it clearly demonstrates how using MCMC methods within a Bayesian framework can be used to quantify and propagate the uncertainties involved in modelling nonlinear dynamical systems.

Figure 4*a* shows a rotational energy harvester—a device which, via a ball–screw mechanism, is designed to convert low-frequency translational motion into high-frequency rotational motion (which can then be transformed into electrical energy). The device is mounted on a electro-hydraulic shaker while accelerometers are attached to the shaker and the oscillating mass (for a more detailed description of the experiment, see [62,63]). With the measured inputs and outputs ($x$ and $y$) being the acceleration of the base and mass, respectively, the aim was to use a set of experimentally obtained data to infer a robust model of the device.

Referring to the schematic in figure 4*b*, the equation of motion of the energy harvester is

$$M\ddot{z} + b_{\mathrm{m}}\dot{z} + kz + f(\dot{z}) = -m\ddot{x}, \quad M = m + J\left(\frac{2\pi}{l}\right)^2, \quad b_{\mathrm{m}} = \left(\frac{2\pi}{l}\right)^2 c_{\mathrm{m}}, \tag{5.1}$$

where $z = y - x$ is the relative acceleration between the mass and base, $l$ is the ball screw lead, $c_{\mathrm{m}}$ is mechanical damping, $k$ is spring stiffness, $m$ is the oscillating mass and $J$ represents the system's moment of inertia. The function $f(\dot{z})$ is a friction model which is to be identified. In this case, two

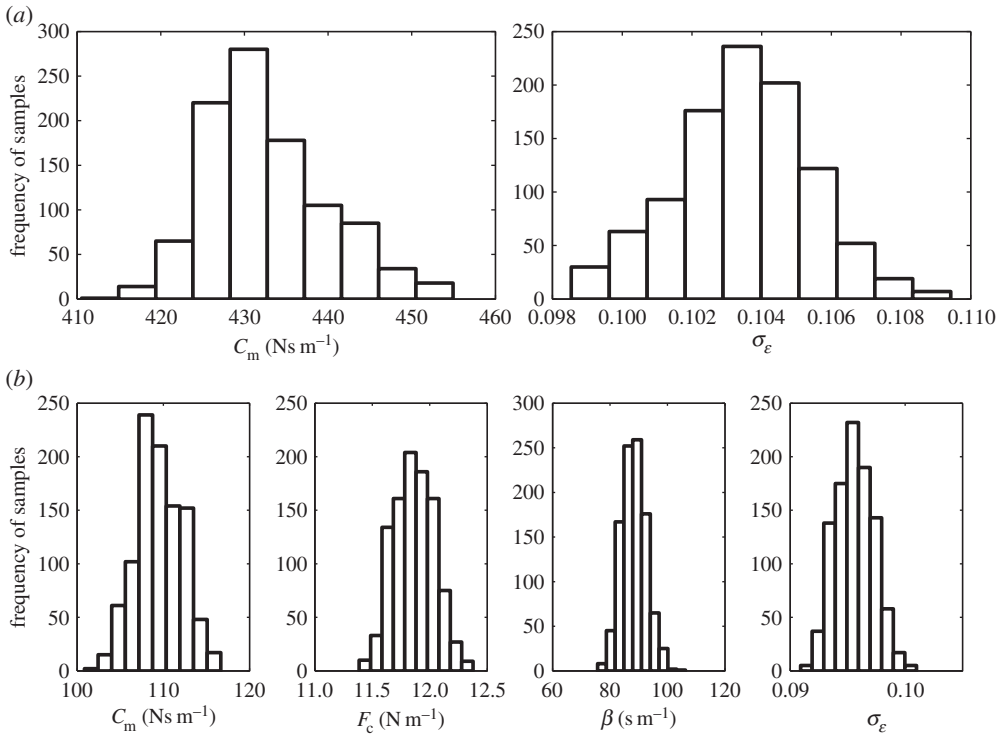


**Figure 5.** MCMC samples generated for model 1 (*a*) and model 2 (*b*).

models, $\mathcal{M}_1$ and $\mathcal{M}_2$, were considered. With the first, it was assumed that all of the parasitic losses in the device could be modelled using a linear viscous damper (which is equivalent to setting $f(\dot{z}) = 0$) while, with the second, a hyperbolic tangent friction model $f(\dot{z}) = F_c \tanh(\beta\dot{z})$ was hypothesized. Assuming that $M$, $k$ and $l$ were already estimated with sufficient accuracy and employing a Gaussian likelihood with standard deviation $\sigma_\epsilon$, the identification of models 1 and 2 involved estimating the parameter vectors $\{c_m, \sigma_\epsilon\}$ and $\{c_m, F_c, \beta, \sigma_\epsilon\}$, respectively. Aside from obtaining probabilistic estimates for the parameters in each model, the aim here was to establish whether it is worth including the additional complexity of the hyperbolic tangent friction model.

Using TMCMC to generate 1000 samples from the posterior parameter distribution of each model, figure 5 shows the histograms of the resulting samples (where row 1 is model 1 and row 2 is model 2). Using these samples as part of Monte Carlo simulations, figure 6*a*,*b* shows the ability of model 1 and model 2 to replicate the training data (the filled grey regions in figure 6 represent $3\sigma$ confidence bounds). These results seem to indicate that the additional complexity of model 2 has allowed it to form a more accurate representation of the training data. Using TMCMC to analyse the marginal likelihood, the finding that $P(\mathcal{M}_2 \mid \mathcal{D}) \approx 1$ confirms that model 2 is preferable. Finally, figure 6*c* shows the ability of model 2 to replicate a set of 'unseen' acceleration time history (data which were not used to train the model).

It is interesting to note that, although model 2 provides a better fit to the data, the confidence bounds on the predictions made by each model are of a similar magnitude. This is essentially due to a poor 'initial phrasing' of the problem. Specifically, when the likelihood was defined, it was assumed that the probability of witnessing each data point was a Gaussian distribution, centred on the prediction made by the model and with variance $\sigma_\epsilon^2$. The choice of a Gaussian distribution is justified somewhat by the Principle of Maximum Entropy [64] from which one finds that, having assumed the first 2 moments of the likelihood, the Gaussian distribution is that which minimizes the amount of additional information that must be assumed. However, having completed the analysis using such a likelihood, it can be observed that model 2 actually appears better able to replicate the experiment when at low amplitudes. From this, one may conclude

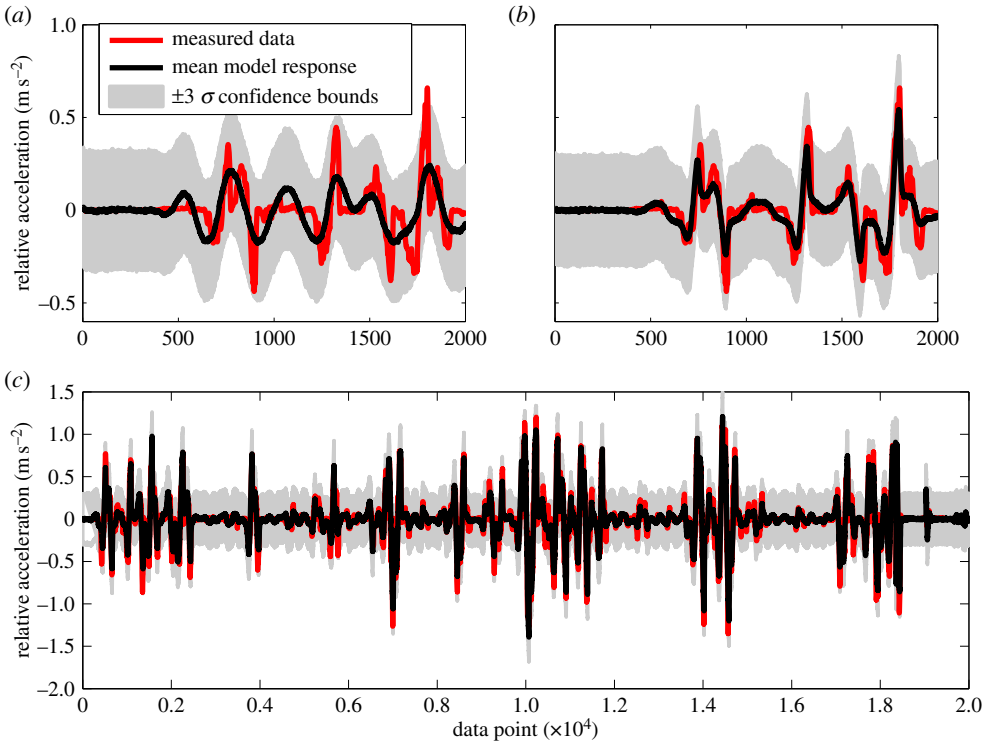


**Figure 6.** The ability of (*a*) model 1 and (*b*) model 2 to replicate the training data. (*c*) Shows predictions about previously 'unseen' data using model 2.

that the probability of witnessing a data point, conditional on the model, actually varies with amplitude. As a continuation to this study one could propose a more complex likelihood before repeating the analysis. Potentially, one could then adopt a Bayesian approach to the selection of different likelihoods, thus preventing the selection of overly complex 'error-prediction' models (e.g. [65]).

## 6. Future work

Ultimately, with each sample generated using MCMC requiring a model run, the applicability of MCMC to Bayesian system identification problems is limited by computational cost. This places several restrictions on the types of problems which can be addressed. For situations where one's model is expensive, a current stream of research is aimed towards the development of MCMC algorithms which are suitable for large-scale parallelization [66], and those which are able to reduce computational cost via the exploitation of interpolation methods (see [67] for example, where kriging is integrated into TMCMC). Further interest has been directed towards the scenario where one is confronted with large datasets from which to infer models. The work [24] proposes a method which allows the selection of small, highly informative subsets of data while, in [47,68], MCMC methods are proposed which allow the tracking of one's parameter estimates as more data are analysed (helping to establish when a sufficient amount of data has been used).

## 7. Conclusion

In this paper, the authors have presented arguments for the adoption of a Bayesian framework for the system identification of nonlinear dynamical systems in the presence of uncertainty. Specifically, it has been highlighted how a Bayesian approach allows one to realize probabilistic

parameter estimates in the presence of measurement noise, select high fidelity models which are not overfitted and make predictions which are marginalized over one's parameter estimates and, in some cases, over a set of candidate model structures. It is then shown how many of the potential difficulties with such an approach can be overcome through the use of Markov chain Monte Carlo (MCMC) algorithms. A brief tutorial/review of six different MCMC algorithms is then given, each of which has been chosen because it has either proved to be particularly useful and/or is based on unique concepts and methodologies. The paper finishes with a case study, where an MCMC algorithm is demonstrated within a Bayesian framework to realize a model of a nonlinear, rotational energy harvester.

Data accessibility. The training data are available in the electronic supplementary material.

Authors' contributions. P.L.G. contributed the material on Markov chain Monte Carlo methods and carried out the analysis shown in §5 (Case study). K.W. contributed the material giving a general overview of Bayesian system identification and aided the article's revision. Both authors contributed to the drafting of the manuscript and gave final approval for publication.

Competing interests. We declare we have no competing interests.

Funding. The authors would like to acknowledge the EPSRC Programme Grant 'Engineering Nonlinearity' EP/K003836/1 which funded the work in this paper as well as the collaborative project described in §5.

## References

1. Soderstrom T, Stoica P. 1994 *System identification*, New Edition. Englewood Cliffs, NJ: Prentice Hall.
2. Ljung L. 1999 *System identification: theory for the user*, 2nd edn. Englewood Cliffs, NJ: Prentice Hall.
3. Worden K, Tomlinson GR. 2001 *Nonlinearity in structural dynamics: detection, modelling and identification*. Bristol, UK: Institute of Physics.
4. Kerschen G, Worden K, Golinval J-C, Vakakis AK. 2006 Past, present and future of nonlinear system identification in structural dynamics. *Mech. Syst. Signal Process.* **20**, 505–592. (doi:10.1016/j.ymssp.2005.04.008)
5. Bouc R. 1967 Forced vibration of mechanical system with hysteresis. In *Proc. of 4th Conf. on Nonlinear Oscillation, Prague, Czechoslovakia*.
6. Wen Y. 1976 Method for random vibration of hysteretic systems. *ASCE J. Eng. Mech. Div.* **102**, 249–263.
7. Mackay MJC. 2003 *Information theory, inference and learning algorithms*. Cambridge, UK: Cambridge University Press.
8. Bishop CM. 2007 *Pattern recognition and machine learning*. Berlin, Germany: Springer.
9. Bishop CM. 1998 *Neural networks for pattern recognition*. Oxford, UK: Oxford University Press.
10. Rasmussen CE, Williams CKI. 2006 *Gaussian processes for machine learning*. Cambridge, MA: MIT Press.
11. Leontaritis IJ, Billings SA. 1985 Input–output parametric models for nonlinear systems, Part I: deterministic nonlinear systems. *Int. J. Control* **41**, 303–328. (doi:10.1080/0020718508961129)
12. Leontaritis IJ, Billings SA. 1985 Input-output parametric models for nonlinear systems, Part II: stochastic nonlinear systems. *Int. J. Control* **41**, 329–344. (doi:10.1080/0020718508961130)
13. Worden K, Manson G, Cross EJ 2012 Higher-order frequency response functions from Gaussian process NARX models. In *Proc. of 25th International Conference on Noise and Vibration Engineering, Leuven, Belgium*.
14. Baldacchino T, Anderson SR, Kadirkamanathan V. 2013 Computational system identification for Bayesian NARMAX modelling. *Automatica* **49**, 2641–2651. (doi:10.1016/j.automatica.2013.05.023)
15. Bard Y. 1974 *Nonlinear parameter estimation*. New York, NY: Academic Press.
16. Collins JD, Hart GC, Hasselman TK, Kennedy B. 1974 Statistical identification of structures. *AIAA J.* **12**, 185–190. (doi:10.2514/3.49190)
17. Beck JL. 1989 Statistical system identification of structures. In *Proc. of 5th Int. Conf. on Structural Safety and Reliability*, pp. 1395–1402. New York, NY: ASCE.
18. Beck JL, Katafygiotis LS. 1998 Updating models and their uncertainties. I. Bayesian statistical framework. *ASCE J. Eng. Mech.* **124**, 455–461. (doi:10.1061/(ASCE)0733-9399(1998)124:4(455))

19. Beck JL, Au SK. 2002 Bayesian updating of structural models and reliability using Markov chain Monte Carlo simulation. *J. Eng. Mech.* **128**, 380–391. (doi:10.1061/(ASCE)0733-9399(2002)128:4(380))
20. Beck JL, Yuen K-V. 2004 Model selection using response measurements: Bayesian probabilistic approach. *ASCE J. Eng. Mech.* **130**, 192–203. (doi:10.1061/(ASCE)0733-9399(2004)130:2(192))
21. Muto M, Beck JL. 2008 Bayesian updating and model class selection for hysteretic structural models using stochastic simulation. *J. Vib. Control* **14**, 7–34. (doi:10.1177/1077546307079400)
22. Worden K, Hensman JJ. 2012 Parameter estimation and model selection for a class of hysteretic systems using Bayesian inference. *Mech. Syst. Signal Process.* **32**, 153–169. (doi:10.1016/j.ymssp.2012.03.019)
23. Beck JL. 2010 Bayesian system identification based on probability logic. *Struct. Control Health Monit.* **17**, 825–847. (doi:10.1002/stc.424)
24. Green PL, Cross EJ, Worden K. 2015 Bayesian system identification of dynamical systems using highly informative training data. *Mech. Syst. Signal Process.* **56–57**, 109–122. (doi:10.1016/j.ymssp.2014.10.003)
25. Girolami M. 2008 Bayesian inference for differential equations. *Theoret. Comp. Sci.* **408**, 4–16. (doi:10.1016/j.tcs.2008.07.005)
26. Calderhead B, Girolami M, Higham DJ. 2010 Is it safe to go out yet? Statistical inference in a zombie outbreak model. University of Strathclyde, Department of Mathematics and Statistics.
27. Neal RM. 1993 *Probabilistic inference using Markov chain Monte Carlo methods*. Technical report. Toronto, ON: Dept of Computer Science, University of Toronto.
28. Patil A, Huard D, Fonnesbeck CJ. 2010 PyMC: Bayesian stochastic modelling in Python. *J. Stat. Software* **35**, 1–81.
29. Beck JL, Yuen KV. 2004 Model selection using response measurements: Bayesian probabilistic approach. *J. Eng. Mech.* **130**, 192–203. (doi:10.1061/(ASCE)0733-9399(2004)130:2(192))
30. Gelman A, Carlin JB, Stern HS, Rubin DB. 2004 *Bayesian data analysis*, 2nd edn. London, UK: Chapman and Hall.
31. Kerschen G, Golinval J-C, Hemez FM. 2003 Bayesian model screening for the identification of nonlinear mechanical structures. *ASME J. Vib. Acoust.* **125**, 389–397. (doi:10.1115/1.1569947)
32. Yuen K-V. 2010 *Bayesian methods for structural dynamics and civil engineering*. New York, NY: John Wiley and Sons.
33. Doob JL. 1953 *Stochastic processes*. Wiley publications in statistics. New York, NY: Wiley.
34. Metropolis N, Rosenbluth AW, Rosenbluth MN, Teller AH, Teller E. 1953 Equation of state calculations by fast computing machines. *J. Chem. Phys.* **21**, 1087–1092. (doi:10.1063/1.1699114)
35. Hastings WK. 1970 Monte Carlo sampling methods using Markov chains and their applications. *Biometrika* **57**, 97–109. (doi:10.1093/biomet/57.1.97)
36. Duane S, Kennedy AD, Pendleton BJ, Roweth D. 1987 Hybrid Monte Carlo. *Phys. Lett. B* **195**, 216–222. (doi:10.1016/0370-2693(87)91197-X)
37. Cheung SH, Beck JL. 2009 Bayesian model updating using hybrid Monte Carlo simulation with application to structural dynamic models with many uncertain parameters. *J. Eng. Mech.* **135**, 243–255. (doi:10.1061/(ASCE)0733-9399(2009)135:4(243))
38. Marwala T. 2001 Probabilistic fault identification using vibration data and neural networks. *Mech. Syst. Signal Process.* **15**, 1109–1128. (doi:10.1006/mssp.2001.1386)
39. Nakada Y, Matsumoto T, Kurihara T, Yosui K. 2005 Bayesian reconstructions and predictions of nonlinear dynamical systems via the hybrid Monte Carlo scheme. *Signal Process.* **85**, 129–145. (doi:10.1016/j.sigpro.2004.09.007)
40. Kirkpatrick S, Gelatt CD, Vecchi MP. 1983 Optimization by simulated annealing. *Science* **220**, 671–680. (doi:10.1126/science.220.4598.671)
41. Green PL 2014 Bayesian System Identification of Nonlinear Dynamical Systems using a Fast MCMC Algorithm. In *Proc. of ENOC 2014, European Nonlinear Dynamics Conference, Vienna, Austria, 6–11 July*.
42. Ching J, Chen YC. 2007 Transitional Markov chain Monte Carlo method for Bayesian model updating, model class selection, and model averaging. *J. Eng. Mech.* **133**, 816–832. (doi:10.1061/(ASCE)0733-9399(2007)133:7(816))
43. Beck JL, Zuev KM. 2013 Asymptotically independent Markov sampling: a new Markov chain Monte Carlo scheme for Bayesian inference. *Int. J. Uncertainty Quant.* **3**, 445–474. (doi:10.1615/Int.J.UncertaintyQuantification.2012004713)
44. Marinari E, Parisi G. 1992 Simulated tempering: a new Monte Carlo scheme. *Europhys. Lett.* **19**, 451–458. (doi:10.1209/0295-5075/19/6/002)

45. Geyer CJ, Thompson EA. 1995 Annealing Markov chain Monte Carlo with applications to ancestral inference. *J. Amer. Stat. Assoc.* **90**, 909–920. (doi:10.1080/01621459.1995.10476590)
46. Hukushima K, Nemoto K. 1996 Exchange Monte Carlo method and application to spin glass simulations. *J. Phys. Soc. Japan* **65**, 1604–1608. (doi:10.1143/JPSJ.65.1604)
47. Green PL. 2014 Bayesian system identification of a nonlinear dynamical system using a novel variant of simulated annealing. *Mech. Syst. Signal Process* **52–53**, 133–146. (doi:10.1016/j.ymssp.2014.07.010)
48. Angelikopoulos P, Papadimitriou C, Koumoutsakos P. 2012 Bayesian uncertainty quantification and propagation in molecular dynamics simulations: a high performance computing framework. *J. Chem. Phys.* **137**, 144103. (doi:10.1063/1.4757266)
49. Goller B, Broggi M, Calvi A, Schueller GI. 2011 A stochastic model updating technique for complex aerospace structures. *Finite Elements Anal. Design* **47**, 739–752. (doi:10.1016/j.finel.2011.02.005)
50. Goller B, Schueller GI. 2011 Investigation of model uncertainties in Bayesian structural model updating. *J. Sound Vibr.* **330**, 6122–6136. (doi:10.1016/j.jsv.2011.07.036)
51. Zheng W, Yu Y. 2013 Bayesian probabilistic framework for damage identification of steel truss bridges under joint uncertainties. *Adv. Civil Eng.* **2013**, 1–13. (doi:10.1155/2013/307171)
52. Zheng W, Chen YT. 2014 Novel probabilistic approach to assessing barge–bridge collision damage based on vibration measurements through transitional Markov chain Monte Carlo sampling. *J. Civil Struct. Health Monit.* **4**, 119–131. (doi:10.1007/s13349-013-0063-2)
53. Wang J, Katafygiotis LS. 2014 Reliability-based optimal design of linear structures subjected to stochastic excitations. *Struct. Safety* **47**, 29–38. (doi:10.1016/j.strusafe.2013.11.002)
54. Green PJ. 1995 Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika* **82**, 711–732. (doi:10.1093/biomet/82.4.711)
55. Green PJ, Hastie DI. 2009 Reversible jump MCMC. *Genetics* **155**, 1391–1403.
56. Zio E, Zoia A. 2009 Parameter identification in degradation modeling by reversible-jump Markov Chain Monte Carlo. *Reliab. IEEE Trans.* **58**, 123–131. (doi:10.1109/TR.2008.2011674)
57. Guan X, Jha R, Liu Y. 2011 Model selection, updating, and averaging for probabilistic fatigue damage prognosis. *Struct. Safety* **33**, 242–249. (doi:10.1016/j.strusafe.2011.03.006)
58. Tiboaca D, Green PL, Barthorpe RJ, Worden K. 2014 Bayesian system identification of dynamical systems using reversible jump Markov Chain Monte Carlo. In *Topics in modal analysis II, Orlando, FL*, vol. 8, pp. 277–284. Berlin, Germany: Springer.
59. Skilling J. 2004 Nested sampling. In *Bayesian inference and maximum entropy methods in science and engineering, Garching, Germany, 25–30 July. AIP Conf. Proc.* **735**, 395–405. (doi:10.1063/1.1835238)
60. Skilling J. 2006 Nested sampling for general Bayesian computation. *Bayesian Anal.* **1**, 833–859. (doi:10.1214/06-BA127)
61. Mthembu L, Marwala T, Friswell MI, Adhikari S. 2011 Model selection in finite element model updating using the Bayesian evidence statistic. *Mech. Syst. Signal Process.* **25**, 2399–2412. (doi:10.1016/j.ymssp.2011.04.001)
62. Green PL, Hendijanizadeh M, Simeone L, Elliott SJ. In press. Probabilistic modelling of a rotational energy harvester. *J. Intelligent Mater. Syst. Struct*. (doi:10.1177/1045389X15573343)
63. Hendijanizadeh M 2014 Design and optimisation of constrained electromagnetic energy harvesting devices. PhD thesis, University of Southampton, UK.
64. Jaynes ET 2003 *Probability theory: the logic of science*. Cambridge, UK: Cambridge University Press.
65. Simoen E, Papadimitriou C, Lombaert G. 2013 On prediction error correlation in Bayesian model updating. *J. Sound Vib.* **332**, 4136–4152. (doi:10.1016/j.jsv.2013.03.019)
66. Hadjidoukas PE, Angelikopoulos P, Papadimitriou C, Koumoutsakos P. 2015 $\pi$4u: A high performance computing framework for Bayesian uncertainty quantification of complex models. *J. Comput. Phys.* **284**, 1–21. (doi:10.1016/j.jcp.2014.12.006)
67. Angelikopoulos P, Papadimitriou C, Koumoutsakos P. 2015 X-TMCMC: Adaptive kriging for Bayesian inverse modeling. *Comp. Methods Appl. Mech. Eng.* **289**, 409–428. (doi:10.1016/j.cma.2015.01.015)
68. Green PL 2015 A MCMC method for Bayesian system identification from large data sets. In *Proc. IMAC XXXIII, Conf. and Exposition on Structural Dynamics. Model Validation and Uncertainty Quantification*, vol. 3, pp. 275–281. Berlin, Germany: Springer International Publishing.