

Bayesian Model Updating Using Hybrid Monte Carlo Simulation with Application to Structural Dynamic Models with Many Uncertain Parameters

Sai Hung Cheung¹ and James L. Beck²

Abstract: In recent years, Bayesian model updating techniques based on measured data have been applied to system identification of structures and to structural health monitoring. A fully probabilistic Bayesian model updating approach provides a robust and rigorous framework for these applications due to its ability to characterize modeling uncertainties associated with the underlying structural system and to its exclusive foundation on the probability axioms. The plausibility of each structural model within a set of possible models, given the measured data, is quantified by the joint posterior probability density function of the model parameters. This Bayesian approach requires the evaluation of multidimensional integrals, and this usually cannot be done analytically. Recently, some Markov chain Monte Carlo simulation methods have been developed to solve the Bayesian model updating problem. However, in general, the efficiency of these proposed approaches is adversely affected by the dimension of the model parameter space. In this paper, the Hybrid Monte Carlo method is investigated (also known as Hamiltonian Markov chain method), and we show how it can be used to solve higher-dimensional Bayesian model updating problems. Practical issues for the feasibility of the Hybrid Monte Carlo method to such problems are addressed, and improvements are proposed to make it more effective and efficient for solving such model updating problems. New formulae for Markov chain convergence assessment are derived. The effectiveness of the proposed approach for Bayesian model updating of structural dynamic models with many uncertain parameters is illustrated with a simulated data example involving a ten-story building that has 31 model parameters to be updated.

DOI: 10.1061/(ASCE)0733-9399(2009)135:4(243)

CE Database subject headings: Bayesian analysis; Identification; Simulation; Structural dynamics; Hybrid methods; Monte Carlo method.

Introduction

Model updating using measured system response, with or without measured excitation, has a wide range of applications in structural health monitoring, structural response prediction, reliability and risk assessment, and structural control. There always exist modeling errors and uncertainties associated with the process of constructing a mathematical model of a structure and its future excitation, whether it is based on physics or on a black-box “non-parametric” model. Being able to quantify the uncertainties accurately and appropriately is essential for a robust prediction of future response and reliability of structures (Beck and Katafygiotis 1991, 1998; Papadimitriou et al. 2001; Beck and Au 2002; Cheung and Beck 2007a). A fully probabilistic Bayesian model updating approach provides a robust and rigorous framework for these applications due to its ability to characterize modeling uncertainties associated with the underlying structural system and to

its exclusive foundation on the probability axioms.

In our applications of the Bayesian approach, we use the Cox–Jaynes interpretation of probability (Cox 1961; Jaynes 2003) as an extension of binary Boolean logic to a multivalued logic of plausible inference where the plausibility of each model within a class \mathcal{M} of models for a system, based on data \mathcal{D} , is quantified by the updated joint probability density function $p(\boldsymbol{\theta}|\mathcal{D},\mathcal{M})$ (posterior PDF); here, the model parameters $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^D$ define each model in \mathcal{M} . By Bayes’ theorem, this posterior PDF of $\boldsymbol{\theta}$ is given by

$$p(\boldsymbol{\theta}|\mathcal{D},\mathcal{M}) = c^{-1}p(\mathcal{D}|\boldsymbol{\theta},\mathcal{M})p(\boldsymbol{\theta}|\mathcal{M}) \quad (1)$$

where $c=p(\mathcal{D}|\mathcal{M})$ is the normalizing constant which makes the probability volume under the posterior PDF equal to unity; $p(\mathcal{D}|\boldsymbol{\theta},\mathcal{M})$, is the likelihood function based on the predictive PDF for the response given by model class \mathcal{M} ; and $p(\boldsymbol{\theta}|\mathcal{M})$ is the prior PDF selected for the model class \mathcal{M} which is used to quantify the initial plausibility of each predictive model defined by the value of the parameters $\boldsymbol{\theta}$ allowing prior information to be incorporated. Based on the data \mathcal{D} , it is useful to classify a Bayesian model class \mathcal{M} into one of three distinct categories (Beck and Katafygiotis 1991, 1998; Katafygiotis and Lam 2002): globally identifiable, locally identifiable, and unidentifiable, depending on whether the set of maximum likelihood estimates is a singleton, finite, or uncountable (continuum) respectively, in the parameter space Θ .

¹Ph.D. Candidate, Division of Engineering and Applied Science, California Institute of Technology, Mail Code 104-44, Pasadena, CA 91125 (corresponding author). E-mail: sai@caltech.edu

²Professor of Engineering and Applied Science, California Institute of Technology, Pasadena, CA 91125. E-mail: jimbeck@caltech.edu

Note. Associate Editor: Arvid Naess. Discussion open until September 1, 2009. Separate discussions must be submitted for individual papers. The manuscript for this paper was submitted for review and possible publication on May 7, 2007; approved on September 26, 2008. This paper is part of the *Journal of Engineering Mechanics*, Vol. 135, No. 4, April 1, 2009. ©ASCE, ISSN 0733-9399/2009/4-243–255/\$25.00.

One of the most useful applications of Bayesian model updating is to make predictions about future events based on past observations. Assume \mathcal{D} contains data (for example, output response and perhaps input data as well) from the measurements on the system. Let $\mathbf{X}_{1:n}$ denote n observations of some physical quantity of interest (for example, total accelerations or interstory drifts) at different discrete time instants which are contained in \mathcal{D} . Based on a choice of model class \mathcal{M} in which each value of $\boldsymbol{\theta}$ gives one model, all the probabilistic information for the prediction of the future responses $\mathbf{X}_{n+1:n+m}$ at m different time instants is contained in the *robust* predictive PDF for \mathcal{M} given by the Theorem of Total Probability

$$p(\mathbf{X}_{n+1:n+m}|\mathcal{D},\mathcal{M}) = \int p(\mathbf{X}_{n+1:n+m}|\boldsymbol{\theta},\mathcal{M})p(\boldsymbol{\theta}|\mathcal{D},\mathcal{M})d\boldsymbol{\theta} \quad (2)$$

where the predictive PDF of each model is weighted by its posterior probability.

The Bayesian approach requires the evaluation of multidimensional integrals, such as in Eq. (2), and this usually cannot be done analytically. Laplace's method of asymptotic approximation (Beck and Katafygiotis 1991, 1998) has been used in the past, which utilizes a Gaussian approximation to the posterior PDF which is valid when there is a large amount of data. However, application of this approximation faces difficulties when (1) the amount of data is small so its accuracy is questionable; or (2) the chosen class of models is unidentifiable based on the available data. Also, such an approximation requires a nonconvex optimization in what is usually a high-dimensional parameter space, which is computationally challenging, especially when the model class is not globally identifiable and so there may be multiple global maximizing points.

Thus, in recent years, focus has shifted from asymptotic approximations to using stochastic simulation methods in which samples consistent with the posterior PDF $p(\boldsymbol{\theta}|\mathcal{D},\mathcal{M})$ are generated. There are several difficulties related to this sampling: (1) the normalizing constant c in Bayes' theorem (1) is unknown a priori and its evaluation requires a high-dimensional integration over the uncertain parameter space; and (2) the high probability content of $p(\boldsymbol{\theta}|\mathcal{D},\mathcal{M})$ occupies a much smaller volume than that of the prior PDF, so samples in the high probability region of $p(\boldsymbol{\theta}|\mathcal{D},\mathcal{M})$ cannot be generated efficiently by sampling from the prior PDF using direct Monte Carlo simulation. To tackle the aforementioned difficulties, Markov chain Monte Carlo (MCMC) simulation methods (e.g., Robert and Casella 1999; Beck and Au 2002; Ching et al. 2006; Ching and Cheng 2007; Muto and Beck 2008) were proposed to solve the Bayesian model updating problem more efficiently.

Probably the most well-known MCMC method is the Metropolis–Hastings (MH) algorithm (Metropolis et al. 1953; Hastings 1970) which creates samples from a Markov chain whose stationary state is a specified target PDF. In principle, this algorithm can be used to generate samples from the posterior PDF but, in practice, its direct use is highly inefficient because the high probability content is often concentrated in a very small volume of the parameter space. Beck and Au (2000, 2002) proposed an approach which combines the idea from simulated annealing with the MH algorithm to simulate from a sequence of target PDFs, where each such PDF is the posterior PDF based on an increasing amount of data. The sequence starts with the spread-out prior PDF and ends with the much more concentrated posterior PDF. The samples from a target PDF in the sequence are used to con-

struct a kernel sampling density which acts as a global proposal PDF for the MH procedure for the next target PDF in the sequence. The success of this approach relies on the ability of the proposal PDF to simulate samples efficiently for each intermediate PDF. However, in practice, this approach is only applicable in lower dimensions since in higher dimensions, a prohibitively large number of samples are required to construct a good global proposal PDF which can generate samples with reasonably high acceptance probability. In other words, if the sample size for the particular level is not large enough, most of the candidate samples generated by the proposal PDF will be rejected by the MH algorithm, leading to many repeated samples, greatly slowing down the exploration of the high probability region of the posterior PDF.

Ching et al. (2006) adopted Gibbs sampling (Geman and Geman 1984) to solve high-dimensional model updating problems that use linear structural models and modal data. Ching and Cheng (2007) proposed the transitional markov chain monte carlo (TMCMC) algorithm and Muto and Beck (2008) applied it to the updating of hysteretic structural models. TMCMC adopts the idea as in Beck and Au (2002) of using a sequence of intermediate PDFs such that the last PDF in the sequence is $p(\boldsymbol{\theta}|\mathcal{D},\mathcal{M})$. The main difference is in the way samples are simulated: TMCMC uses reweighting and resampling techniques on the samples from a target PDF $\pi_i(\boldsymbol{\theta})$ in the sequence to generate initial samples for the next target PDF $\pi_{i+1}(\boldsymbol{\theta})$ in the sequence. A Markov chain of samples is initiated from each of these initial samples using the MH algorithm with stationary distribution $\pi_{i+1}(\boldsymbol{\theta})$: each sample is generated from a local random walk using a Gaussian proposal PDF centered at the current sample of the chain that has a covariance matrix estimated by importance sampling using samples from $\pi_i(\boldsymbol{\theta})$. TMCMC has several advantages over the previous approaches: (1) it is more efficient; (2) it allows the estimation of the normalizing constant c of $p(\boldsymbol{\theta}|\mathcal{D},\mathcal{M})$, which is important for Bayesian model class selection (Beck and Yuen 2004). However, TMCMC has potential problems in higher dimensions which need further attention: (1) the initial samples from reweighting and resampling of samples in $\pi_i(\boldsymbol{\theta})$, in general, do not exactly follow $\pi_{i+1}(\boldsymbol{\theta})$, so the Markov chains must “burn-in” before samples follow $\pi_{i+1}(\boldsymbol{\theta})$, requiring a large amount of samples to be generated for each intermediate level; (2) in higher dimensions, convergence to $\pi_{i+1}(\boldsymbol{\theta})$ can be very slow when using the MH algorithm based on local random walks, as in TMCMC. This adverse effect becomes more pronounced as the dimension increases, and it introduces more inaccuracy into the statistical estimates based on the samples.

In this paper, we show how the Hybrid Monte Carlo method, also known as Hamiltonian Markov chain method, can be used to solve higher-dimensional Bayesian model updating problems. Additional proof of the validity of the Hybrid Monte Carlo method using the Fokker–Planck equation is also provided. Features and parameters which affect the effectiveness of the Hybrid Monte Carlo method for higher-dimensional updating problems are discussed. Practical issues for feasibility of the method are addressed, and improvements are proposed to make it more effective and efficient for solving higher-dimensional model updating problems in structural dynamics. New formulae for Markov chain convergence assessment are derived. The effectiveness of the proposed approach for Bayesian model updating of structural dynamic models with many uncertain parameters is illustrated with a simulate data example involving a ten-story building that has 31 model parameters to be updated.

Hybrid Monte Carlo Method

Hybrid Monte Carlo method (HMCM) was first introduced by Duane et al. (1987) as a MCMC technique for sampling from complex distributions by combining Gibbs sampling, MH algorithm acceptance rule, and deterministic dynamical methods. By avoiding the local random walk behavior exhibited by the MH algorithm through the use of dynamical methods, HMCM can be much more efficient. The advantage of HMCM is even more pronounced when sampling the highly-correlated parameters from posterior distributions that are often encountered in Bayesian structural model updating. However, the potential of HMCM has not yet been explored in Bayesian structural model updating.

In HMCM, a fictitious dynamical system is considered in which auxiliary “momentum” variables $\mathbf{p} \in \mathbb{R}^D$ are introduced and the uncertain parameters $\boldsymbol{\theta} \in \mathbb{R}^D$ in the target distribution $\pi(\boldsymbol{\theta})$ are treated as the variables for the displacement. The total energy (Hamiltonian function) of the fictitious dynamical system is defined by $H(\boldsymbol{\theta}, \mathbf{p}) = V(\boldsymbol{\theta}) + W(\mathbf{p})$, where its potential energy $V(\boldsymbol{\theta}) = -\ln \pi(\boldsymbol{\theta})$ and its kinetic energy $W(\mathbf{p})$ depend only on \mathbf{p} and some chosen positive definite “mass” matrix $\mathbf{M} \in \mathbb{R}^{D \times D}$

$$W(\mathbf{p}) = \mathbf{p}^T \mathbf{M}^{-1} \mathbf{p} / 2 \quad (3)$$

Since \mathbf{M} can be chosen at our convenience, it is taken as a diagonal matrix with entries M_i , i.e., $\mathbf{M} = \text{diag}(M_i)$. A joint distribution $f(\boldsymbol{\theta}, \mathbf{p})$ over the phase space $(\boldsymbol{\theta}, \mathbf{p})$ is considered

$$f(\boldsymbol{\theta}, \mathbf{p}) = K \exp[-H(\boldsymbol{\theta}, \mathbf{p})] \quad (4)$$

where K = normalizing constant. Clearly

$$f(\boldsymbol{\theta}, \mathbf{p}) = K \pi(\boldsymbol{\theta}) \exp(-\mathbf{p}^T \mathbf{M}^{-1} \mathbf{p} / 2) \quad (5)$$

Note that $\pi(\boldsymbol{\theta})$ can be unnormalized (the usual situation that arises when constructing a posterior PDF) since its normalizing constant can be absorbed into K . Samples of $\boldsymbol{\theta}$ from $\pi(\boldsymbol{\theta})$ can be obtained if we can sample $(\boldsymbol{\theta}, \mathbf{p})$ from the joint distribution $f(\boldsymbol{\theta}, \mathbf{p})$ in Eq. (5). Note that Eq. (5) shows that \mathbf{p} and $\boldsymbol{\theta}$ are independent and the marginal distributions of $\boldsymbol{\theta}$ and \mathbf{p} are, respectively, $\pi(\boldsymbol{\theta})$ and $\mathcal{N}(\mathbf{0}, \mathbf{M})$, a Gaussian distribution with zero mean and covariance matrix \mathbf{M} .

Using Hamilton's equations, the evolution of $(\boldsymbol{\theta}, \mathbf{p})$ through fictitious time t is given by

$$\frac{d\mathbf{p}}{dt} = -\frac{\partial H}{\partial \boldsymbol{\theta}} = -\nabla V(\boldsymbol{\theta}) \quad (6)$$

$$\frac{d\boldsymbol{\theta}}{dt} = \frac{\partial H}{\partial \mathbf{p}} = \mathbf{M}^{-1} \mathbf{p} \quad (7)$$

There are four features worth noting regarding the above evolution:

1. The total energy H remains constant throughout the evolution;
2. The dynamics are time reversible, i.e., if a trajectory initiates at $(\boldsymbol{\theta}', \mathbf{p}')$ at time 0 and ends at $(\boldsymbol{\theta}'', \mathbf{p}'')$ at time t , then a trajectory starting at $(\boldsymbol{\theta}'', \mathbf{p}'')$ at time 0 will end at $(\boldsymbol{\theta}', \mathbf{p}')$ at time $-t$ (or, equivalently, a trajectory starting at $(\boldsymbol{\theta}'', -\mathbf{p}'')$ at time 0 will end at $(\boldsymbol{\theta}', -\mathbf{p}')$ at time t);
3. The volume of a region of phase space remains constant (by Liouville's theorem); and
4. The above evolution of $(\boldsymbol{\theta}, \mathbf{p})$ leaves $f(\boldsymbol{\theta}, \mathbf{p})$ in Eq. (5) as the stationary distribution (Duane et al. 1987); in particular, if $\boldsymbol{\theta}(0)$ follows the distribution $\pi(\boldsymbol{\theta})$, then after time t , $\boldsymbol{\theta}(t)$ also follows $\pi(\boldsymbol{\theta})$. Duane et al. (1987) proved this by showing

that the detailed balance condition for the stationarity of a Markov chain is satisfied. In Appendix I, we provide an alternative proof to show that $f(\boldsymbol{\theta}, \mathbf{p})$ is actually the stationary distribution using the diffusionless Fokker-Planck equation.

If we start with $\boldsymbol{\theta}(0)$ and draw a sample $\mathbf{p}(0)$ from $\mathcal{N}(\mathbf{0}, \mathbf{M})$, then solve the Hamiltonian dynamics (6) and (7) for some time t , the final values $[\boldsymbol{\theta}(t), \mathbf{p}(t)]$ will provide an independent sample $\boldsymbol{\theta}(t)$ from $\pi(\boldsymbol{\theta})$. In practice, Eqs. (6) and (7) have to be solved numerically using some time-stepping algorithm such as the commonly-used leapfrog algorithm (Duane et al. 1987). In this latter case, for time step δt , we have

$$\mathbf{p}\left(t + \frac{\delta t}{2}\right) = \mathbf{p}(t) - \frac{\delta t}{2} \nabla V[\boldsymbol{\theta}(t)] \quad (8)$$

$$\boldsymbol{\theta}(t + \delta t) = \boldsymbol{\theta}(t) + \delta t \mathbf{M}^{-1} \mathbf{p}\left(t + \frac{\delta t}{2}\right) \quad (9)$$

$$\mathbf{p}(t + \delta t) = \mathbf{p}\left(t + \frac{\delta t}{2}\right) - \frac{\delta t}{2} \nabla V[\boldsymbol{\theta}(t + \delta t)] \quad (10)$$

Eqs. (8)–(10) can be reduced to

$$\boldsymbol{\theta}(t + \delta t) = \boldsymbol{\theta}(t) + \delta t \mathbf{M}^{-1} \left\{ \mathbf{p}(t) - \frac{\delta t}{2} \nabla V[\boldsymbol{\theta}(t)] \right\} \quad (11)$$

$$\mathbf{p}(t + \delta t) = \mathbf{p}(t) - \frac{\delta t}{2} \{ \nabla V[\boldsymbol{\theta}(t)] + \nabla V[\boldsymbol{\theta}(t + \delta t)] \} \quad (12)$$

The gradient of V with respect to $\boldsymbol{\theta}$ needs to be calculated once only for each time instant since its value in the last step in the above algorithm at time t is the same as the first step at time $t + \delta t$.

HMCM Algorithm

The complete algorithm of HMCM can be summarized as follows (for some chosen \mathbf{M} , δt , and L):

1. Initialize $\boldsymbol{\theta}_0$ (discussion of the choice of this is presented in a later section) and simulate \mathbf{p}_0 such that $\mathbf{p}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{M})$;
2. Repeat the following for $i = 1, \dots, N$:

In iteration i , let the most recent sample be $(\boldsymbol{\theta}_{i-1}, \mathbf{p}_{i-1})$, then do the following to simulate a new sample $(\boldsymbol{\theta}_i, \mathbf{p}_i)$:

- a. Randomly draw a new momentum vector \mathbf{p}' from $\mathcal{N}(\mathbf{0}, \mathbf{M})$;
- b. Initiate the leapfrog algorithm with $[\boldsymbol{\theta}(0), \mathbf{p}(0)] = (\boldsymbol{\theta}_{i-1}, \mathbf{p}')$ and run the algorithm for L time steps to obtain a new candidate sample $(\boldsymbol{\theta}'', \mathbf{p}'') = [\boldsymbol{\theta}(t + L\delta t), \mathbf{p}(t + L\delta t)]$; and
- c. Accept $(\boldsymbol{\theta}_i, \mathbf{p}_i) = (\boldsymbol{\theta}'', \mathbf{p}'')$ with probability $P_{acc} = \min\{1, \exp(-\Delta H)\}$ where $\Delta H = H(\boldsymbol{\theta}'', \mathbf{p}'') - H(\boldsymbol{\theta}_{i-1}, \mathbf{p}_{i-1})$. If rejected, then $(\boldsymbol{\theta}_i, \mathbf{p}_i) = (\boldsymbol{\theta}_{i-1}, \mathbf{p}_{i-1})$, so $V(\boldsymbol{\theta}_i) = V(\boldsymbol{\theta}_{i-1})$ and $\nabla V(\boldsymbol{\theta}_i) = \nabla V(\boldsymbol{\theta}_{i-1})$.

Discussion of Algorithm

Step 2 a allows simulation of samples in regions with different H , thereby allowing the Markov chain to move to any point in the phase space of $(\boldsymbol{\theta}, \mathbf{p})$ via the deterministic step in 2 b. This is an important step since it allows a global exploration of the $\boldsymbol{\theta}$ space in contrast to the local random walk behavior of the MH algo-

rithm with a local proposal PDF. We can represent most integration algorithms used to solve Hamilton's equations by the following general iterative formula:

$$[\boldsymbol{\theta}(n\delta t), \mathbf{p}(n\delta t)] = \mathbf{h}\{\boldsymbol{\theta}[(n-1)\delta t], \mathbf{p}[(n-1)\delta t]\} \quad (13)$$

where \mathbf{h} corresponds to the mapping produced by the time-stepping algorithm, e.g., leap frog. The candidate sample $(\boldsymbol{\theta}_c, \mathbf{p}_c)$ is then the output of the following:

$$(\boldsymbol{\theta}_c, \mathbf{p}_c) = \underbrace{\mathbf{h}(\mathbf{h}(\cdots \mathbf{h}(\boldsymbol{\theta}(0), \mathbf{p}(0))))}_L = \underbrace{\mathbf{h}(\mathbf{h}(\cdots \mathbf{h}(\boldsymbol{\theta}(0), \mathbf{M}^{1/2}\mathbf{z}))}_L \quad (14)$$

where \mathbf{z} is a standard Gaussian vector with independent components $\mathcal{N}(0, 1)$. Thus, Steps 2 a and 2 b together can be viewed as drawing a candidate sample from a global transition PDF which is non-Gaussian if the mapping \mathbf{h} is nonlinear (the usual case). Applying mapping \mathbf{h} multiple times leads to the exploration of the phase space further away from the current point, toward the higher probability region, avoiding the local random walk behavior of most MCMC methods. Therefore, HMCM can be viewed as a combination of Gibbs sampling (Step 2 a) followed by a Metropolis algorithm step (Step 2 c) in an enlarged space with an implied complicated proposal PDF that enhances a more global exploration of the phase space than using a simple Gaussian PDF centered at the current sample, as adopted for the proposal PDF in the random walk Metropolis algorithm.

Although the leapfrog algorithm is volume preserving (symplectic) and time reversible, H does not remain exactly constant due to the systematic error introduced by the discretization of Eqs. (6) and (7) with the leapfrog algorithm. To keep $f(\boldsymbol{\theta}, \mathbf{p})$ as the invariant PDF of the Markov chain, and thus keep $\pi(\boldsymbol{\theta})$ invariant, this systematic error needs to be corrected through the Metropolis acceptance/rejection step in Step 2 c. The probability of acceptance, \mathbf{p}_{acc} , in Step 2 c depends only on the difference in energy ΔH between H for the candidate sample $(\boldsymbol{\theta}'', \mathbf{p}'')$ and H for $(\boldsymbol{\theta}_{i-1}, \mathbf{p}')$, which initiates the current leapfrog steps. The candidate sample $(\boldsymbol{\theta}'', \mathbf{p}'')$ with lower H is always accepted, while that with higher H is accepted with a probability of $\min\{1, \exp(-\Delta H)\}$.

It is worth noting that when $L=1$, HMCM is similar to an algorithm in which the evolution of $\boldsymbol{\theta}$ follows the following Itô stochastic differential equation:

$$d\boldsymbol{\theta}(t) = -\frac{1}{2}\mathbf{M}^{-1}\nabla V[\boldsymbol{\theta}(t)]dt + \mathbf{M}^{-1/2}d\tilde{\mathbf{W}}(t) \quad (15)$$

where $\tilde{\mathbf{W}}(t) \in \mathbb{R}^D$ is a standard Wiener process. The discretized version corresponding to Eq. (15) is

$$\boldsymbol{\theta}_c = \boldsymbol{\theta}(t) - \frac{1}{2}\mathbf{M}^{-1}\nabla V[\boldsymbol{\theta}(t)]\delta t + \mathbf{M}^{-1/2}\sqrt{\delta t}\mathbf{z} \quad (16)$$

where $\boldsymbol{\theta}_c$ =candidate sample and \mathbf{z} is a standard Gaussian vector with independent components that are $\mathcal{N}(0, 1)$. Thus, it is interesting to see that when $L=1$, the candidate sample of HMCM is drawn from the Gaussian proposal PDF

$$q[\boldsymbol{\theta}_c|\boldsymbol{\theta}(t)] = \frac{1}{(2\pi|\mathbf{C}|)^{D/2}} \times \exp\left(-\frac{1}{2}\{\boldsymbol{\theta}_c - \tilde{\boldsymbol{\mu}}[\boldsymbol{\theta}(t)]\}^T \mathbf{C}^{-1}\{\boldsymbol{\theta}_c - \tilde{\boldsymbol{\mu}}[\boldsymbol{\theta}(t)]\}\right) \quad (17)$$

where the mean $\tilde{\boldsymbol{\mu}}[\boldsymbol{\theta}(t)]$ and the covariance matrix \mathbf{C} are given by the following:

$$\tilde{\boldsymbol{\mu}}[\boldsymbol{\theta}(t)] = \boldsymbol{\theta}(t) + \frac{1}{2}\mathbf{M}^{-1}\delta t \nabla \ln \pi[\boldsymbol{\theta}(t)] \quad (18)$$

$$\mathbf{C} = \mathbf{E}\{[\mathbf{M}^{-1/2}\sqrt{\delta t}\mathbf{z}(t)][\mathbf{M}^{-1/2}\sqrt{\delta t}\mathbf{z}(t)]^T\} = \delta t\mathbf{M}^{-1} \quad (19)$$

It can be seen from Eq. (18) that the above algorithm can reduce the tendency to do a local random walk by having a drift term that tends to force the Markov chain samples toward the higher probability region of $\pi(\boldsymbol{\theta})$.

There are three parameters, namely \mathbf{M} , δt , and L , that need to be chosen before performing HMCM. If δt is chosen to be too large, the energy H at the end of the trajectory will deviate too much from the energy at the start of the trajectory which may lead to frequent rejections due to the Metropolis step in Step 2 c. Thus, δt should be chosen small enough so that the average rejection rate due to the Metropolis step is not too large but not too small that effective exploration of the high probability region is inhibited; a procedure for optimally choosing δt is presented later. For each dynamic evolution in the deterministic Step 2 b, L can be randomly chosen from a discrete uniform distribution from 1 to some preselected L_{max} to avoid getting into a resonance condition (Mackenzie 1989) (although it occurs rarely in practice) in which the trajectories from Step 2 b go around the same closed trajectory for a number of cycles. Matrix \mathbf{M} can be chosen to be a diagonal matrix $\text{diag}(M_1, \dots, M_D)$, where M_i is 1 for each i if the components of $\boldsymbol{\theta}$ are of comparable scale. This can be ensured by initially normalizing the uncertain parameters $\boldsymbol{\theta}$.

Proposed Improvements

Computation of Gradient of $V(\boldsymbol{\theta})$ in Implementation of HMCM

In general, $\nabla V(\boldsymbol{\theta}) = -\nabla \ln \pi(\boldsymbol{\theta})$ cannot be found analytically, so numerical methods must be used to find its value. The most common method uses finite differences. The computation of the gradient vector $\nabla V(\boldsymbol{\theta})$ using finite differences requires either D or $2D$ evaluations of V , where D =dimension of the uncertain parameters.

Here we propose to use "algorithmic differentiation" (Rall 1981; Kagiwada et al. 1986), in which a program code for sensitivity analysis (gradient calculation) can be created alongside the original program for an output analysis to form a combined code for both output analysis and sensitivity analysis. The program code for the output analysis can always be viewed as a composite of basic arithmetic operations and some elementary intrinsic functions. The main idea of algorithmic differentiation is to apply the chain rule for differentiation judiciously to the elementary functions, the building blocks forming the program for output analysis, and to calculate the output and its sensitivity with respect to the input parameters simultaneously in one code. Reverse differentiation is a form of algorithmic differentiation (Wolfe 1982; Griewank 1989) which starts with the output variables and computes the sensitivity of the output with respect to each of the intermediate variables. The biggest advantage of reverse differentiation is seen when the output variable is a scalar and the corresponding gradient, with respect to high-dimensional input parameters, is of interest. Under this circumstance, it has been shown (Griewank 1989) that the computational effort required by reverse differentiation to calculate the gradient accurately is only between 1 to 4 times of that required to calculate the output

function, regardless of the dimension of the input parameters. This situation applies to our problem since the output variable of interest is the scalar function V .

Structural analysis programs usually involve program statements which perform vector and matrix operations and solve implicit linear equations. Higher-dimensional implicit linear equations are involved and the number of elementary intermediate variables required to store information for differentiation is large. Thus, it is more efficient to perform differentiation at the vector or matrix levels. A very efficient reverse differentiation code has been derived for this case in this work. Due to the space limitations, the details of the derivations and implementation are not presented here.

Control of δt

The acceptance probability of a candidate sample at the end of the $(\boldsymbol{\theta}, \mathbf{p})$ trajectory for the Hamiltonian dynamics of Eqs. (6) and (7) is influenced by the discretization errors introduced by the integration algorithm. The distance d moved in the $(\boldsymbol{\theta}, \mathbf{p})$ space after one evolution depends on δt . In HMCM, δt should be chosen small enough so that the average rejection rate due to the Metropolis step is not too large. On the other hand, larger δt facilitates a bigger movement from the existing samples and so a better exploration of the phase space. Therefore, we want to choose δt which is as large as possible while at the same time maintaining a reasonable acceptance rate of the Metropolis step. This can be achieved by maximizing the expected distance $d(\delta t)$ moved by a sample with respect to δt

$$d(\delta t) = (\delta t) \bar{P}_{\text{acc}}(\delta t) \quad (20)$$

where the average acceptance probability in HMCM, \bar{P}_{acc} , can be estimated by counting the proportion of distinct samples out of the amount of samples simulated. To do the above maximization, one can use a small number of samples and empirically explore different δt 's to achieve maximum $d(\delta t)$ with δt chosen such that $\bar{P}_{\text{acc}} \geq p_0$ (say $p_0=0.1$).

Increasing the Acceptance Probability of Samples

If the acceptance probability is increased for a fixed δt , then it will produce a reduction in the repetition of samples, thus improving the efficiency of exploration of the posterior PDF by the HMCM samples. In very high dimensions, one way to further increase the acceptance probability is to use more accurate higher-order symplectic integrators, such as those in Forest and Ruth (1990), but at the expense of increased computational effort. Another variant is to utilize information in the trajectory samples when moving from $(\boldsymbol{\theta}_{i-1}, \mathbf{p}_{i-1})$ to $(\boldsymbol{\theta}_i, \mathbf{p}_i)$ in Step 2 of HMCM (Neal 1994; Cheung and Beck 2007b)

Starting Markov Chain in High Probability Region of Posterior PDF

Starting the Markov chain with an initial point $\boldsymbol{\theta}_0$ closer to the important region of the posterior PDF can lead to more efficient exploration of this region. The following has been found to be effective:

The optimization of $V(\boldsymbol{\theta})$ [equivalently $\pi(\boldsymbol{\theta})$] to select $\boldsymbol{\theta}_0$ can be performed using an efficient SPSA (simultaneous perturbation stochastic approximation) optimization algorithm (Spall 1998a) with the use of common random numbers (Kleinman 1999). $\boldsymbol{\theta}_0$ is taken as the approximate optimal solution $\boldsymbol{\theta}^*$ obtained by the optimization algorithm. This method relies on the approximation of $\nabla V(\boldsymbol{\theta})$ using a two-sided perturbation as follows:

$$\frac{\partial V}{\partial \theta_i} \approx \frac{V(\boldsymbol{\theta} + h\boldsymbol{\Delta}) - V(\boldsymbol{\theta} - h\boldsymbol{\Delta})}{2h\Delta_i} \quad (21)$$

where $\boldsymbol{\Delta}=[\Delta_1, \Delta_2, \dots, \Delta_D]$ is the perturbation vector, the distribution of which is user-specified and h is a scalar which dictates the size of the perturbation of $\boldsymbol{\theta}$. A simple and valid choice for $\boldsymbol{\Delta}$ (Spall 1998b; Sadegh and Spall 1998) is to use a symmetric Bernoulli distribution: $P(\Delta_i=1)=P(\Delta_i=-1)=0.5$, for $i=1, 2, \dots, D$.

In SPSA, all components of $\boldsymbol{\theta}$ are perturbed randomly and simultaneously, and only two evaluations of the function V are required (instead of $2D$ evaluations required in the finite central difference method) to estimate the whole gradient vector ∇V . The optimization algorithm for determining an optimal point $\boldsymbol{\theta}^*$ is done by running the following recursive equation, starting with some initial guess $\boldsymbol{\theta}_0$:

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - a_k g_k(\boldsymbol{\theta}_k) \quad (22)$$

where $g_k(\boldsymbol{\theta}_k)$ =estimate of the gradient of V evaluated at $\boldsymbol{\theta}_k$

$$g_k(\boldsymbol{\theta}_k) = \frac{V(\boldsymbol{\theta}_k + b_k \boldsymbol{\Delta}_k) - V(\boldsymbol{\theta}_k - b_k \boldsymbol{\Delta}_k)}{2b_k \Delta_k} \quad (23)$$

$\boldsymbol{\Delta}_k=[\Delta_{k1}, \Delta_{k2}, \dots, \Delta_{kD}]$ =perturbation vector generated in the k th iteration using the Bernoulli distribution as before; $a_k=a_0/(A+k+1)^\alpha$ and $b_k=b_0/(k+1)^\gamma$ are gain sequences which are critical to the performance of SPSA based optimization. Normalization of $\boldsymbol{\theta}$ is performed so that each component of $\boldsymbol{\theta}$ is of comparable scale. Some guidelines for the selection of the non-negative coefficients a_0 , b_0 , A , α , and γ are provided in Spall (1998b). Common random numbers can be used to further improve the convergence of the above SPSA optimization algorithm (Kleinman et al. 1999). Another improvement is to use a second-order stochastic algorithm analogous to the deterministic Newton–Raphson algorithm (Spall 1997).

It should be noted that the approach presented in this section cannot solve the case involving well-separated regions with high probability content of the posterior PDF. On the other hand, with enough samples in previous levels, TMCMC can potentially provide initial points in different regions of high probability content of the posterior PDF by making use of multiple chains. However, the inherent convergence and efficiency problems of the random walk MH algorithm in higher dimensions still exists. One can incorporate HMCM proposed in this paper into TMCMC by replacing the random walk Metropolis algorithm in simulating from the whole sequence of PDFs or just the last PDF in the sequence. In practice, the case involving well-separated regions with high probability content of the posterior PDF is relatively rare.

Assessment of Markov Chain Reaching Stationarity

Given a finite set of N samples $\boldsymbol{\theta}^{(k)}$, $k=1, 2, \dots, N$, from a Markov chain distributed according to its stationary PDF $\pi(\boldsymbol{\theta})$, the estimate for the expectation of any function $\mathbf{g}(\boldsymbol{\theta})$ of $\boldsymbol{\theta}$ is as follows:

$$E[\mathbf{g}(\boldsymbol{\theta})] = \int \mathbf{g}(\boldsymbol{\theta}) \pi(\boldsymbol{\theta}) d\boldsymbol{\theta} \approx \frac{1}{N} \sum_{k=1}^N \mathbf{g}(\boldsymbol{\theta}^{(k)}) \quad (24)$$

For example, if $\mathbf{g}(\boldsymbol{\theta})=\boldsymbol{\theta}$, then $E[\mathbf{g}(\boldsymbol{\theta})]$ will become the expected value of $\boldsymbol{\theta}$, i.e., $E[\boldsymbol{\theta}]$. If the Markov chain is ergodic, the right-hand side of Eq. (24) converges almost surely to the left-hand side for samples simulated using MCMC procedures such as the one presented in this paper (Tierney 1994). In this section, we first present a new approach to assess whether the samples $\boldsymbol{\theta}^{(k)}$, $k=1, 2, \dots, N$, simulated using an MCMC algorithm, have converged to samples from its stationary PDF $\pi(\boldsymbol{\theta})$. Then we exam-

ine how the accuracy of the estimator in Eq. (24) depends on the number of samples N .

A common existing approach for convergence assessment is based on observing whether the sample estimate of a certain $E[\mathbf{g}(\boldsymbol{\theta})]$ stabilizes for some chosen function \mathbf{g} . However, this can give misleading results since the stabilization can be a result of the chain of samples being trapped in some neighborhood of the parameter space (but the Markov chain has not yet converged to the stationary distribution). Another major drawback of this approach is that it is hard to judge how far the underlying Markov chain is away from reaching stationarity or convergence since one does not know a priori what value the estimate for $E[\mathbf{g}(\boldsymbol{\theta})]$ should converge to.

To solve the above issues, we establish a known quantity depending on $\pi(\boldsymbol{\theta})$ which can also be estimated from the samples, then we check how far the estimate is from the exact value of the chosen quantity. Consider the quantity

$$I_i = E[g(\theta_i)] = \int g(\theta_i)\pi(\boldsymbol{\theta})d\boldsymbol{\theta} \quad (25)$$

where $g(\theta_i)$ is such that there exists some differentiable function $G(\theta_i)$ with $G'(\theta_i)=g(\theta_i)$. Recall that $\pi(\boldsymbol{\theta})=c^{-1}\exp[-V(\boldsymbol{\theta})]$. Denote $\boldsymbol{\theta}_{-i}$ as a vector containing all elements of $\boldsymbol{\theta}$ except θ_i ; $\pi(\theta_i)$ as the marginal distribution of θ_i ; and θ_i^u and θ_i^l as the upper limit and lower limit of the domain of integration with respect to θ_i , respectively. After performing integration by parts on I_i with respect to θ_i , an alternative expression for $E[g(\theta_i)]$ can be obtained. If we divide this alternative expression by I_i as follows, Q_i should be equal to 1:

$$Q_i = \frac{\int G(\theta_i)\frac{\partial V(\boldsymbol{\theta})}{\partial \theta_i}\pi(\boldsymbol{\theta})d\boldsymbol{\theta} + \int [G(\theta_i)\pi(\boldsymbol{\theta})]_{\theta_i=\theta_i^l}^{\theta_i=\theta_i^u}d\boldsymbol{\theta}_{-i}}{\int g(\theta_i)\pi(\boldsymbol{\theta})d\boldsymbol{\theta}} = 1 \quad (26)$$

The second term in the nominator can be expressed in terms of $\pi(\theta_i)$ as follows:

$$\begin{aligned} \int [G(\theta_i)\pi(\boldsymbol{\theta})]_{\theta_i=\theta_i^l}^{\theta_i=\theta_i^u}d\boldsymbol{\theta}_{-i} &= \left[G(\theta_i) \int \pi(\boldsymbol{\theta})d\boldsymbol{\theta}_{-i} \right]_{\theta_i=\theta_i^l}^{\theta_i=\theta_i^u} \\ &= [G(\theta_i)\pi(\theta_i)]_{\theta_i=\theta_i^l}^{\theta_i=\theta_i^u} \end{aligned} \quad (27)$$

Thus, Eq. (26) becomes

$$Q_i = \frac{\int G(\theta_i)\frac{\partial V(\boldsymbol{\theta})}{\partial \theta_i}\pi(\boldsymbol{\theta})d\boldsymbol{\theta} + [G(\theta_i)\pi(\theta_i)]_{\theta_i=\theta_i^l}^{\theta_i=\theta_i^u}}{\int g(\theta_i)\pi(\boldsymbol{\theta})d\boldsymbol{\theta}} = 1 \quad (28)$$

Denote $\theta_i^{(k)}$ as the i th component of the k th sample $\boldsymbol{\theta}^{(k)}$ from $\pi(\boldsymbol{\theta})$. The sample estimate \tilde{Q}_i for Q_i is given by

$$\tilde{Q}_i = \frac{\frac{1}{N}\sum_{k=1}^N G(\theta_i^{(k)}) \left. \frac{\partial V(\boldsymbol{\theta})}{\partial \theta_i} \right|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{(k)}} + [G(\theta_i)\pi(\theta_i)]_{\theta_i=\theta_i^l}^{\theta_i=\theta_i^u}}{\frac{1}{N}\sum_{k=1}^N g(\theta_i^{(k)})} \quad (29)$$

It is convenient to choose $G(\theta_i)=\theta_i$ so that $g(\theta_i)=1$, and thus Eq. (29) becomes

$$\tilde{Q}_i = \frac{1}{N}\sum_{k=1}^N \theta_i^{(k)} \left. \frac{\partial V(\boldsymbol{\theta})}{\partial \theta_i} \right|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{(k)}} \quad (30)$$

where the second term in the numerator of Eq. (29) is dropped because usually, for model updating problems, $\pi(\theta_i)$ decays exponentially as θ_i approaches the limit of domain of integration. Asymptotically, all \tilde{Q}_i 's should converge to 1 with increasing N . With the above construction, we can define a quantity which averages over all Q_i 's

$$\bar{Q} = \sum_{i=1}^D Q_i/D \quad (31)$$

The exact value of \bar{Q} is 1. The estimate $\tilde{\bar{Q}}$ for \bar{Q} by simulation is obtained by averaging all \tilde{Q}_i 's

$$\tilde{\bar{Q}} = \sum_{i=1}^D \tilde{Q}_i/D \quad (32)$$

We assume that the Markov chain is close enough to stationarity if the error of $\tilde{\bar{Q}}$ is less than a certain acceptable threshold, i.e., $|\tilde{\bar{Q}}-1| < \varepsilon$.

Statistical Accuracy of Sample Estimator

Now, let $\tilde{E}[g(\boldsymbol{\theta})]$ denote the estimator of $E[g(\boldsymbol{\theta})]$ as in Eq. (24) for some function g . Let $\boldsymbol{\theta}^{(k)}$, $k=1, 2, \dots, N$, denote samples from the stationary PDF $\pi(\boldsymbol{\theta})$. The statistical accuracy of the sample estimator $\tilde{E}[g(\boldsymbol{\theta})]=1/N\sum_{k=1}^N g(\boldsymbol{\theta}^{(k)})$ can be assessed by evaluating the corresponding coefficient of variation (c.o.v.) $\tilde{\delta}_g$ which can be estimated using the following:

$$\tilde{\delta}_g = \frac{\sqrt{\text{Var}(\tilde{E}[g(\boldsymbol{\theta})])}}{E[\tilde{E}[g(\boldsymbol{\theta})]]} \quad (33)$$

where the mean $E[\tilde{E}[g(\boldsymbol{\theta})]]$ and variance $\text{Var}(\tilde{E}[g(\boldsymbol{\theta})])$ of the sample estimate can be estimated using the following (the derivation of $\text{Var}(\tilde{E}[g(\boldsymbol{\theta})])$ is shown in Appendix II):

$$E[\tilde{E}[g(\boldsymbol{\theta})]] = E[g(\boldsymbol{\theta})] \approx \tilde{E}[g(\boldsymbol{\theta})] = \frac{1}{N}\sum_{k=1}^N g(\boldsymbol{\theta}^{(k)}) \quad (34)$$

$$\text{Var}(\tilde{E}[g(\boldsymbol{\theta})]) = \frac{\rho(0)}{N}(1 + \lambda) \quad (35)$$

$$\lambda = 2\sum_{\tau=1}^{N-1} \left(1 - \frac{\tau}{N}\right) \frac{\rho(\tau)}{\rho(0)} \in [0, N-1] \quad (36)$$

$$\begin{aligned} \rho(\tau) &= E[(g(\boldsymbol{\theta}^{(k+\tau)}) - E[g(\boldsymbol{\theta})])(g(\boldsymbol{\theta}^{(k)}) - E[g(\boldsymbol{\theta})])] \\ &\approx \frac{1}{N-\tau}\sum_{k=1}^{N-\tau} (g(\boldsymbol{\theta}^{(k+\tau)}) - \tilde{E}[g(\boldsymbol{\theta})])(g(\boldsymbol{\theta}^{(k)}) - \tilde{E}[g(\boldsymbol{\theta})]) \end{aligned} \quad (37)$$

$\text{Var}(\tilde{E}[g(\boldsymbol{\theta})])$ is equal to the lower bound $\rho(0)/N$ (corresponding to $\lambda=0$) when the samples are independent (such as when using standard Monte Carlo simulation) while $\text{Var}(\tilde{E}[g(\boldsymbol{\theta})])$ is equal to

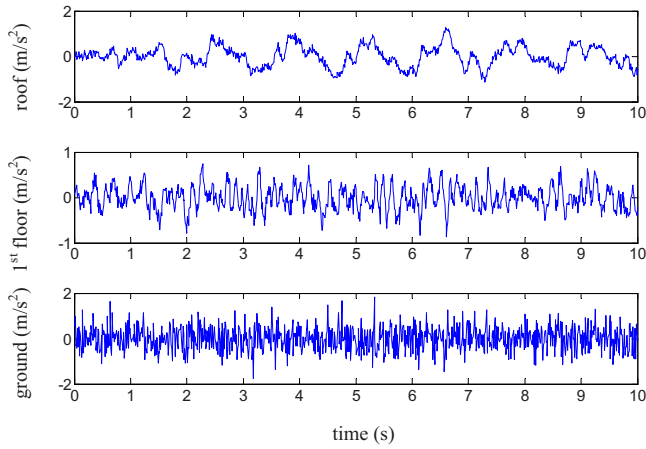


Fig. 1. Acceleration Data Set 1 in ten-story building

the upper bound $\rho(0)$ (corresponding to $\lambda=N-1$) when the samples are perfectly correlated. The closer the value of λ is to zero, the less correlated the samples are. In fact, $N/(1+\lambda)$ can be viewed as the effective number of independent samples. Eqs. (34)–(37) can be used to estimate the c.o.v. for the estimator of $E[g(\theta)]$ from N MCMC samples.

Illustrative Example: Ten-Story Building

Suppose that noisy accelerometer data (simulated here) are available from a ten-story building excited by an earthquake. Two sets of data are considered: Data Set 1 are the acceleration data that are contaminated by a typical amount of noise (10% rms noise-to-signal ratio) used in published simulated data studies; Dataset 2 are the acceleration data that are contaminated by a large amount of noise (100% rms noise-to-signal ratio) to examine the robustness of the Bayesian procedure to extreme noise levels. System identification is to be performed using a ten-story linear shear-building model and so we estimate the mass m_i , damping coefficient c_i , and stiffness parameter k_i for each story, $i=1, \dots, 10$. A duration of 10 s (with a sample interval of 0.01 s) of the total acceleration at the base, the first floor, and the roof are measured. The measurements corresponding to Dataset 1 and Dataset 2 are shown in Figs. 1 and 2, respectively. Let $N_o=2$ denote the number

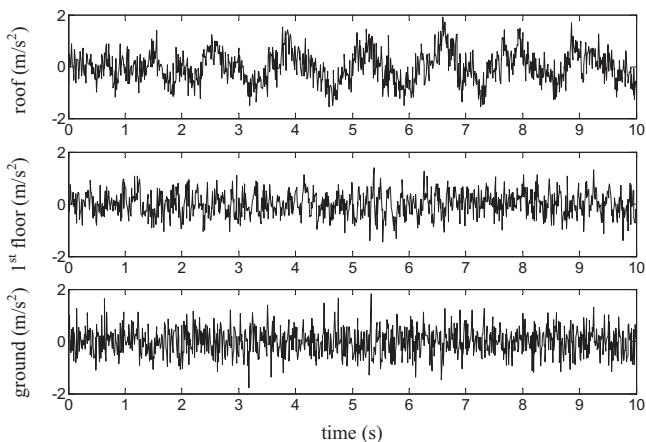


Fig. 2. Acceleration Data Set 2 in ten-story building

of observed degrees of freedom (first floor and roof) and $N_T=1000$ denote the length of the discrete time history data. Let $y_n(t_j; \theta)$ denote the output at time t_j at the n th observed degree of freedom predicted by the proposed structural model and $\hat{y}_n(t_j)$ denote the corresponding measured output. As in Beck and Katafygiotis (1991, 1998), the prediction and measurement errors $\varepsilon_n(t_j) = \hat{y}_n(t_j) - y_n(t_j; \theta)$ for $n=1, 2, \dots, N_o$ and $j=1, 2, \dots, N_T$, are modeled as independent and identically distributed Gaussian variables with mean zero and some unknown variance σ^2 , based on the Principle of Maximum Entropy (Jaynes 2003). Altogether, we need to estimate 31 model parameters with σ included.

The likelihood function $p(\mathcal{D}|\theta)$ for this problem is

$$p(\mathcal{D}|\theta) = \frac{1}{(2\pi\sigma^2)^{N_o N_T/2}} \exp\left(-\frac{1}{2\sigma^2} \sum_{n=1}^{N_o} \sum_{j=1}^{N_T} [\hat{y}_n(t_j) - y_n(t_j; \theta)]^2\right) \quad (38)$$

Note that this updating problem is unidentifiable because the mass, stiffness, and damping parameters can be uniformly scaled without changing the $y_n(t_j; \theta)$. The prior PDF for θ is chosen to be independent distributions, that is, m_i , c_i , k_i follow a Gaussian distribution with means equal to their nominal values $m_0=2 \times 10^4$ kg; $c_0=6 \times 10^4$ Nm⁻¹ s, $k_0=2 \times 10^7$ Nm⁻¹, and the corresponding coefficients of variation (c.o.v.) of 10%, 30%, 30%, and σ follows a lognormal distribution with median $\sigma_0=1.0$ ms⁻² and a logarithmic standard deviation of $s_0=0.3$ (the c.o.v. is about 30%). These nominal values are not equal to the exact values, which are assumed to be unknown. For the mass parameters, relatively smaller values of c.o.v. are assumed since these parameters can usually be more accurately determined from the structural drawings than the other parameters. For each of the other parameters that are not so well known a priori, a larger c.o.v. is assumed. It should be noted the objective of the prior PDFs is to allow prior information to be incorporated when performing model updating. For those parameters where there is little prior information, prior PDFs that reflect higher uncertainty (i.e., in this case, larger c.o.v.) are used. Under such circumstances, the updated uncertainties for these parameters depend mostly on the data and are often insensitive to the prior PDFs. Here we define the dimensionless uncertain parameters θ_i , $i=1, 2, \dots, 30$, as the original parameters divided by their nominal values: $\theta_i=m_i/m_0$ for $i=1, \dots, 10$; $\theta_i=c_{i-10}/c_0$ for $i=11, \dots, 20$; $\theta_i=k_{i-20}/k_0$ for $i=21, \dots, 30$; and $\theta_{31}=\sigma/\sigma_0$.

HMCM is applied by first doing 3000 evaluations of $\pi(\theta)$ for Dataset 1 and 4000 evaluations of $\pi(\theta)$ for Dataset 2 to find the initial point via the SPSA algorithm. The SPSA stopping criteria is such that each component of θ and $\ln \pi(\theta)$ of the current iteration and the previous iteration differ by less than a prescribed threshold of 1%. Then 3000 HMCM samples are generated which are sufficient to reduce the error of \bar{Q} time-stepping to less than $\varepsilon=0.1$, where \bar{Q} is evaluated using Eqs. (30) and (32). In the HMCM, L is chosen to be an integer selected from a uniform distribution over the interval $[0, 40]$ and $\delta t=0.0005$ for Dataset 1 and $\delta t=0.0075$ for Dataset 2 to give an average probability of accepting candidate samples of about 0.8–0.9. The upper limit of L is chosen such that the correlation between the neighboring samples for each component is small (in this case, the correlation coefficient of the neighboring samples is less than 0.2).

The partial derivative of $V(\theta)$ in HMCM with respect to θ_{31} can be determined analytically

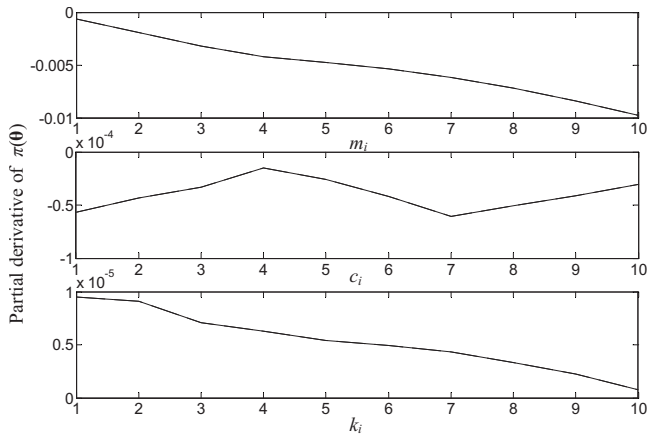


Fig. 3. Gradient using two different methods: reverse algorithmic differentiation and central finite difference for mass parameters (top figure), damping parameters (middle figure), and stiffness parameters (bottom figure); the curves are indistinguishable

$$\frac{\partial V}{\partial \theta_{31}} = \frac{\partial V}{\partial \sigma} \frac{\partial \sigma}{\partial \theta_{31}} = \sigma_0 \left\{ \frac{N_o N_T}{\sigma} - \frac{1}{\sigma^3} \sum_{n=1}^{N_o} \sum_{j=1}^{N_T} [\hat{y}_n(t_j) - y_n(t_j; \boldsymbol{\theta})]^2 + \frac{1}{\sigma} \left(1 + \frac{\ln \sigma - \ln \sigma_0}{s_0^2} \right) \right\} \quad (39)$$

The remaining 30 components of the gradient of V with respect to $\boldsymbol{\theta}$ are calculated using the efficient reverse algorithmic differentiation code which was developed in this study. Fig. 3 shows that the gradient computation using the reverse algorithmic differentiation (RD) overlaps with that obtained by central finite difference (CFD) with optimum perturbation size. It should be noted that the amount of computations required by CFD to calculate a gradient vector is 30 times that required by RD.

Table 1 shows the sample mean (column 3), sample c.o.v. (column 4), and estimation error (column 5) of the structural parameters, along with the exact values (column 2) of the parameters used to generate Data Set 1. Compared with the prior uncertainty in the parameters, the posterior (updated) uncertainty is reduced since the data provide information about these parameters. There is a smaller degree of reduction in the uncertainty in the mass parameters than that in the damping and stiffness param-

Table 1. Statistical Results for Structural Parameter Estimates for 10% Noise-to-Signal Ratio [Data Set 1]

Parameter	Exact value β_i	μ_i =mean estimate of parameter	σ_i/μ_i =c.o.v. estimate of parameter	Error= $ \beta_i - \mu_i /\beta_i$	$ \mu_i - \beta_i /\sigma_i$
1 m_1	1.92×10^4	2.00×10^4	3.2%	3.8%	1.16
2 m_2	1.97×10^4	2.06×10^4	5.2%	4.4%	0.82
3 m_3	1.95×10^4	1.95×10^4	7.2%	0.3%	0.04
4 m_4	2.06×10^4	2.00×10^4	5.9%	3.0%	0.52
5 m_5	2.05×10^4	2.02×10^4	5.4%	1.1%	0.21
6 m_6	1.98×10^4	2.01×10^4	6.3%	1.8%	0.29
7 m_7	1.94×10^4	1.91×10^4	6.8%	1.0%	0.14
8 m_8	2.06×10^4	2.00×10^4	9.1%	2.7%	0.30
9 m_9	1.90×10^4	2.08×10^4	7.3%	9.9%	1.23
10 m_{10}	2.01×10^4	2.18×10^4	5.4%	8.6%	1.47
11 c_1	7.70×10^4	8.62×10^4	5.9%	12.0%	1.81
12 c_2	7.78×10^4	8.20×10^4	7.9%	5.4%	0.66
13 c_3	7.86×10^4	7.70×10^4	12.0%	2.0%	0.17
14 c_4	7.28×10^4	7.46×10^4	8.8%	2.4%	0.27
15 c_5	7.19×10^4	8.18×10^4	5.6%	13.7%	2.15
16 c_6	7.37×10^4	7.07×10^4	8.4%	4.0%	0.50
17 c_7	7.10×10^4	7.77×10^4	10.4%	9.3%	0.82
18 c_8	7.11×10^4	6.20×10^4	10.1%	12.8%	1.46
19 c_9	6.90×10^4	6.93×10^4	13.8%	0.4%	0.03
20 c_{10}	7.57×10^4	6.63×10^4	7.2%	12.4%	1.97
21 k_1	2.16×10^7	2.24×10^7	3.4%	4.0%	1.14
22 k_2	1.74×10^7	1.76×10^7	4.6%	0.8%	0.16
23 k_3	2.04×10^7	2.07×10^7	7.4%	1.7%	0.22
24 k_4	1.99×10^7	2.09×10^7	4.7%	5.0%	1.00
25 k_5	1.74×10^7	1.86×10^7	5.5%	6.5%	1.11
26 k_6	1.68×10^7	1.74×10^7	6.8%	3.3%	0.48
27 k_7	1.87×10^7	1.89×10^7	7.3%	0.9%	0.12
28 k_8	1.77×10^7	1.89×10^7	9.8%	7.0%	0.66
29 k_9	1.84×10^7	1.86×10^7	8.7%	1.0%	0.11
30 k_{10}	1.72×10^7	1.64×10^7	5.3%	4.6%	0.92
31 σ	0.040	0.041	1.6%	2.5%	1.49

Table 2. Statistical Results for Structural Parameter Estimates for 100% Noise-to-Signal Ratio [Data Set 2]

Parameter	Exact value β_i	μ_i =mean estimate of parameter	σ_i/μ_i =c.o.v. estimate of parameter	Error= $ \beta_i-\mu_i /\beta_i$	$ \mu_i-\beta_i /\sigma_i$
1 m_1	1.92×10^4	1.95×10^4	7.3%	1.2%	0.17
2 m_2	1.97×10^4	2.02×10^4	9.3%	2.3%	0.24
3 m_3	1.95×10^4	1.95×10^4	9.0%	0.2%	0.02
4 m_4	2.06×10^4	2.07×10^4	9.5%	0.4%	0.04
5 m_5	2.05×10^4	1.95×10^4	9.4%	5.0%	0.53
6 m_6	1.98×10^4	2.04×10^4	9.5%	3.0%	0.31
7 m_7	1.94×10^4	2.00×10^4	9.6%	3.2%	0.32
8 m_8	2.06×10^4	1.98×10^4	10.3%	3.7%	0.37
9 m_9	1.90×10^4	1.91×10^4	10.1%	1.1%	0.08
10 m_{10}	2.01×10^4	2.05×10^4	10.1%	2.4%	0.23
11 c_1	7.70×10^4	7.45×10^4	20.0%	3.2%	0.17
12 c_2	7.78×10^4	6.86×10^4	22.3%	12.0%	0.61
13 c_3	7.86×10^4	6.82×10^4	23.7%	13.3%	0.65
14 c_4	7.28×10^4	5.92×10^4	27.9%	18.7%	0.83
15 c_5	7.19×10^4	5.96×10^4	30.3%	17.2%	0.68
16 c_6	7.37×10^4	6.13×10^4	27.4%	16.9%	0.74
17 c_7	7.10×10^4	7.14×10^4	25.6%	1.0%	0.02
18 c_8	7.11×10^4	6.67×10^4	26.2%	6.3%	0.25
19 c_9	6.90×10^4	6.06×10^4	28.5%	12.2%	0.49
20 c_{10}	7.57×10^4	6.79×10^4	24.6%	10.4%	0.47
21 k_1	2.16×10^7	2.05×10^7	9.1%	4.8%	0.56
22 k_2	1.74×10^7	1.50×10^7	10.3%	13.7%	1.54
23 k_3	2.04×10^7	1.97×10^7	14.9%	3.2%	0.22
24 k_4	1.99×10^7	2.29×10^7	14.5%	15.1%	0.90
25 k_5	1.74×10^7	2.24×10^7	17.4%	28.7%	1.28
26 k_6	1.68×10^7	1.99×10^7	20.0%	18.3%	0.77
27 k_7	1.87×10^7	1.93×10^7	21.0%	3.1%	0.14
28 k_8	1.77×10^7	1.99×10^7	19.5%	12.4%	0.57
29 k_9	1.84×10^7	1.82×10^7	20.3%	1.5%	0.07
30 k_{10}	1.72×10^7	1.74×10^7	31.8%	2.3%	0.02
31 σ	0.400	0.395	1.6%	1.1%	0.73

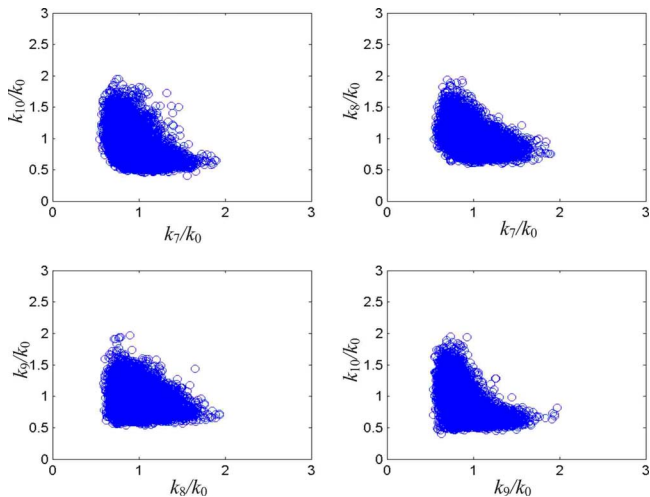


Fig. 4. Pairwise posterior sample plots for some stiffness parameters

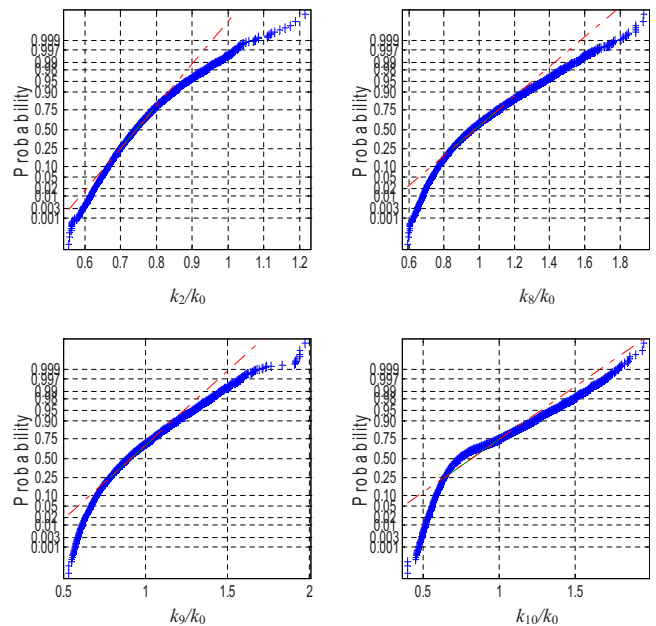


Fig. 5. Gaussian probability paper plots for some k_i

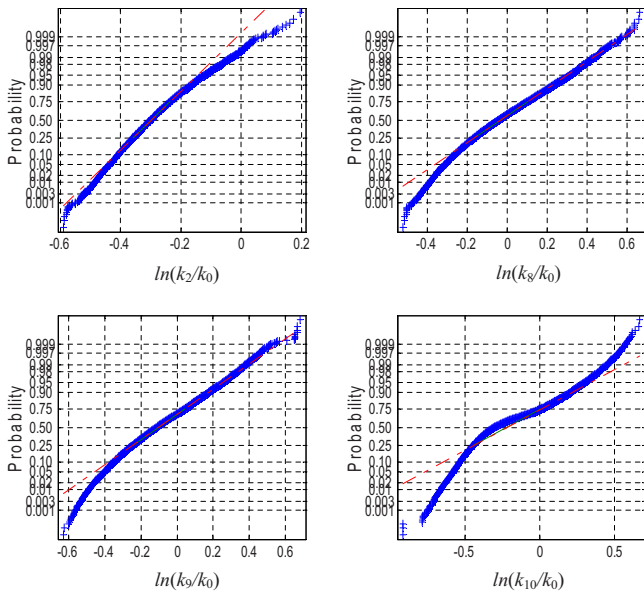


Fig. 6. Gaussian probability paper plots for some $\ln k_i$

eters. This is because the prior PDF for the mass parameters is closer to the corresponding posterior PDF than that for the other parameters. As expected, there is a higher uncertainty in the damping parameters than in the mass and stiffness parameters. This is because the modal contributions to the response are more sensitive to the mass and stiffness than to the damping. It can be seen that the estimation error is reasonably small: 0.3–10.0% for mass parameters; 0.4–13.7% for damping parameters; 0.75–7.0% for stiffness parameters. Column 6 shows the magnitude of the error in terms of the number of standard deviations. It can be seen that the magnitude of error is less than 2 standard deviations for almost all parameters.

Table 2 shows the results using Data Set 2, which is the large noise case. It can be seen that even in this case, the performance of Bayesian system identification is still good. In most cases, the errors in the stiffness parameters are significantly larger than

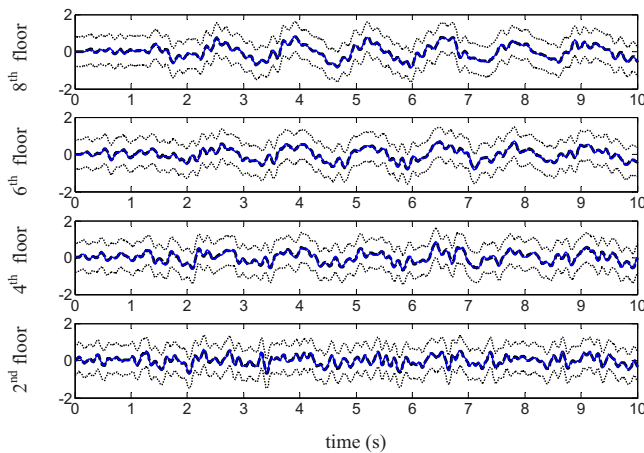


Fig. 7. Exact (solid) and mean predicted (dashed) time histories of the total acceleration (m/s^2) at some unobserved floors together with time histories of the total acceleration that are twice the standard deviation of the predicted robust response from the mean robust response (dotted) (Data Set 2)

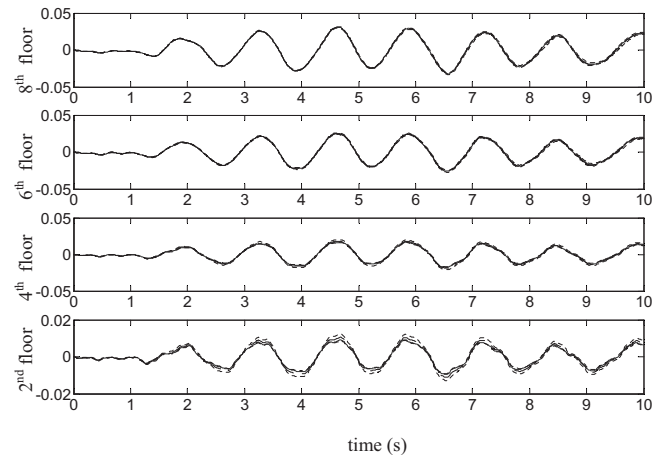


Fig. 8. Exact (solid) and mean (dashed) time histories of the displacement (m) at some unobserved floors together with time histories of the displacement that are twice the standard deviation of the predicted robust response from the mean robust response (dotted) (Data Set 2)

Dataset 1. The results for the stiffness parameters are highly correlated with one another and are not jointly Gaussian. Fig. 4 shows the samples plots for some pairs of θ_i corresponding to the stiffness. It can be seen clearly that the stiffness parameters are not jointly Gaussian. Figs. 5 and 6 show plots where posterior samples for some k_i/k_0 and $\ln(k_i/k_0)$, respectively, are plotted on Gaussian probability paper. If the samples essentially lie on a straight line in these plots, the posterior marginal distribution of θ_i can be taken to be approximately Gaussian or lognormal, respectively. From the figures, it can be seen that the marginal distribution for some stiffness parameters (for example, k_2 , k_8 , k_9 , k_{10} , etc.) are non-Gaussian and also not log-normal. The multivariate Gaussian approximation of the posterior PDF that is effectively assumed in Bayesian updating using Laplace's asymptotic approximation (Beck and Katafygiotis, 1991, 1998), is not so good here because the few observed locations ($N_o=2$), high noise-to-signal ratio (100%), and many parameters (31) make the problem

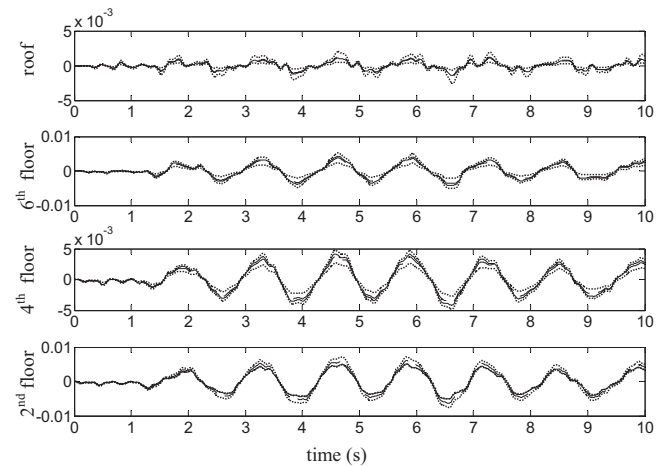


Fig. 9. Exact (solid) and mean (dashed) time histories of the interstory drift (m) at some unobserved floors s together with time histories of the interstory drift that are twice the standard deviation of the predicted robust response from the mean robust response (dotted) (Data Set 2)

Table 3. Exact Natural Frequency and Damping Ratio for Each Complex Mode (Data Set 2)

Complex mode	Natural frequency (Hz)	Damping ratio (%)	Natural frequency (Hz) from Bayesian updating	Damping ratio (%) from Bayesian updating
1	0.735	0.92	0.734(0.2%)	0.85(8.0%)
2	2.158	2.71	2.149(0.3%)	2.60(7.1%)
3	3.562	4.45	3.600(0.7%)	4.03(9.5%)
4	4.891	6.03	4.878(0.8%)	5.83(8.6%)
5	6.047	7.65	6.022(1.8%)	7.33(8.8%)
6	7.106	9.11	7.214(2.3%)	8.42(10.1%)
7	8.049	10.13	7.990(2.4%)	9.17(11.5%)
8	8.620	11.11	8.828(2.7%)	9.56(13.1%)
9	9.306	11.58	9.661(3.2%)	9.60(13.5%)
10	9.631	11.92	10.519(4.5%)	9.26(15.5%)

unidentifiable (e.g., see Fig. 4). Being able to capture the non-Gaussian behavior of the posterior PDF is essential for robust prediction of the future response and reliability of structures (Cheung and Beck 2007a).

To illustrate the predictive power and robustness of the Bayesian model updating approach using HMCM, we compare the exact time histories of the total acceleration (Fig. 7), the displacement (Fig. 8), and the interstory drift (Fig. 9) of some *unobserved* floors with the corresponding mean response from the robust predictive PDF given by Eq. (2). The solid curve shows the exact values of the response; the dashed curve shows the mean robust response estimated by averaging over the mean responses from each of the posterior samples. The two dotted curves give the responses that are twice the standard deviation of the predicted robust response from the mean robust response. The curves for the exact and the mean total acceleration, displacement, and drift responses are almost indistinguishable. Also, all figures show that the exact response lies almost always between the two dot-dashed curves. It can be seen that Bayesian robust analyses are able to give robust prediction of the response even at the unobserved degrees of freedom, despite the fact that the model is unidentifiable based on very noisy data. The total acceleration being a linear combination of displacements and velocities has its uncertainty contributed by both displacements and velocities, while the interstory drift being the difference of the displacement of the two neighboring floors has its uncertainty contributed by the displacement of the two floors. Thus, higher uncertainties can be found in predicting the total acceleration and interstory drift than predicting the floor displacement.

The building considered in this example has nonclassical damping and, thus, possesses complex modes. Table 3 gives the sample mean (with sample c.o.v. inside the parenthesis) of the natural frequency (column 4) and damping ratio (column 5) for each complex mode along with the exact values of the natural frequency and damping ratio (columns 2 and 3). It can be seen that the Bayesian analysis is able to give robust estimates for these modal parameters of the underlying structure despite the large noise and lack of identifiability in the structural parameters. As expected, the estimates for the lower modes are better than those for the higher modes (as can be seen from the higher sample c.o.v. for the parameters corresponding to higher modes) because only the first few complex modes of the structure are excited significantly by the earthquake ground motion, so it is the information from these modes that are primarily utilized in the estimation of the shear-building model parameters. However, the higher-mode frequencies and damping parameters are still quite

accurately estimated, presumably because the tridiagonal shear-building stiffness and damping matrices induce strong constraints on the modal parameters.

Concluding Remarks

The proposed methodology shows high potential for solving model updating problems in higher-dimensional parameter spaces, even unidentifiable ones, which are well known to present a challenging computational problem. Any type of model can be used: physics-based or blackbox, linear or nonlinear, without restriction on the type of data. Although our focus of application here is on system identification and model updating of structural dynamic systems, there are other possible areas of potential application such as Bayesian regression and classification problems (e.g., Oh et al. 2008).

The Hybrid Monte Carlo algorithm is presented and its features are discussed and reviewed in detail. Improvements are proposed to make the Hybrid Monte Carlo method more effective and efficient for solving higher-dimensional model updating problems in structural dynamics. New formulae for Markov chain convergence assessment are also derived. The illustrative numerical example shows that based on acceleration data from the structure, the proposed fully probabilistic Bayesian model updating approach is able to characterize modeling uncertainties associated with the underlying structural system and can provide robust estimation even when the model class is unidentifiable based on the recorded response.

The focus of this paper is to establish the tools that are suitable for solving Bayesian model updating problems involving structural dynamic models with many uncertain parameters. The Hybrid Monte Carlo method was successfully applied to updating of a linear structural dynamic model with 31 uncertain parameters. The next step is to apply these tools to Bayesian structural model updating of more complicated structural systems and to the case where data are collected from real structures.

Acknowledgments

The first writer would like to acknowledge the financial support from Caltech's George W. Housner Graduate Fellowship.

Appendix I

The Hamiltonian Eqs. (6) and (7) are equivalent to the following diffusionless Itô stochastic differential equation:

$$d\mathbf{x}(t) = \mathbf{v}[\mathbf{x}(t), t] dt \quad (40)$$

where the state is formed by augmenting the displacement vector with the momentum vector

$$\mathbf{x}(t) = \begin{bmatrix} \boldsymbol{\theta}(t) \\ \mathbf{p}(t) \end{bmatrix} \quad (41)$$

and the drift term $\mathbf{v}[\mathbf{x}(t), t]$ of the corresponding Fokker–Planck equation (FPE) is given by

$$\mathbf{v}[\mathbf{x}(t), t] = \begin{bmatrix} \frac{\partial H}{\partial \mathbf{p}} \\ -\frac{\partial H}{\partial \boldsymbol{\theta}} \end{bmatrix} \quad (42)$$

Here we will show that the probability density function $f(\boldsymbol{\theta}, \mathbf{p})$ as defined in Eq. (4) is the stationary distribution for the evolution in (40). Consider

$$\begin{aligned} \nabla \cdot (f\mathbf{v}(\mathbf{x}(t), t)) &= f \nabla \cdot (\mathbf{v}(\mathbf{x}(t), t)) + \mathbf{v}(\mathbf{x}(t), t) \cdot \nabla f \\ &= f \nabla \cdot (\mathbf{v}(\mathbf{x}(t), t)) - f\mathbf{v}(\mathbf{x}(t), t) \\ &\quad \cdot \nabla H (\because f = C^{-1} \exp(-H) \Rightarrow \nabla f = -f \nabla H) \\ &= f(\nabla \cdot (\mathbf{v}(\mathbf{x}(t), t)) - \mathbf{v}(\mathbf{x}(t), t) \cdot \nabla H) \\ &= f \left(\begin{bmatrix} \frac{\partial}{\partial \boldsymbol{\theta}} \\ \frac{\partial}{\partial \mathbf{p}} \end{bmatrix} \cdot \begin{bmatrix} \frac{\partial H}{\partial \mathbf{p}} \\ -\frac{\partial H}{\partial \boldsymbol{\theta}} \end{bmatrix} - \begin{bmatrix} \frac{\partial H}{\partial \mathbf{p}} \\ -\frac{\partial H}{\partial \boldsymbol{\theta}} \end{bmatrix} \cdot \begin{bmatrix} \frac{\partial H}{\partial \boldsymbol{\theta}} \\ \frac{\partial H}{\partial \mathbf{p}} \end{bmatrix} \right) \\ &= f \left[\sum_{i=1}^D \frac{\partial^2 H^2}{\partial \theta_i \partial p_i} - \frac{\partial^2 H^2}{\partial p_i \partial \theta_i} - \left(\sum_{i=1}^D \frac{\partial H}{\partial p_i} \frac{\partial H}{\partial \theta_i} \right. \right. \\ &\quad \left. \left. - \frac{\partial H}{\partial \theta_i} \frac{\partial H}{\partial p_i} \right) \right] = 0 \end{aligned}$$

Thus $f(\boldsymbol{\theta}, \mathbf{p})$ = stationary distribution for Eq. (40) [equivalently for Eqs. (6) and (7)] since it satisfies the corresponding stationary diffusionless FPE (Liouville's equation)

$$\nabla \cdot \{f\mathbf{v}[\mathbf{x}(t), t]\} = 0 \quad (43)$$

Appendix II

$$\begin{aligned} \text{Var}(\tilde{E}[g(\boldsymbol{\theta})]) &= E[(\tilde{E}[g(\boldsymbol{\theta})] - E[\tilde{E}[g(\boldsymbol{\theta})]])^2] = E \left[\left(\frac{1}{N} \sum_{k=1}^N g(\boldsymbol{\theta}^{(k)}) - E[g(\boldsymbol{\theta})] \right)^2 \right] = E \left[\left(\frac{1}{N} \sum_{k=1}^N g(\boldsymbol{\theta}^{(k)}) - E[g(\boldsymbol{\theta})] \right) \left(\frac{1}{N} \sum_{j=1}^N g(\boldsymbol{\theta}^{(j)}) - E[g(\boldsymbol{\theta})] \right) \right] \\ &= \frac{1}{N^2} E \left[\sum_{k=1}^N \sum_{j=1}^N (g(\boldsymbol{\theta}^{(k)}) - E[g(\boldsymbol{\theta})]) (g(\boldsymbol{\theta}^{(j)}) - E[g(\boldsymbol{\theta})]) \right] = \frac{1}{N^2} \sum_{k=1}^N \sum_{\tau=1}^N E[(g(\boldsymbol{\theta}^{(k)}) - E[g(\boldsymbol{\theta})]) (g(\boldsymbol{\theta}^{(k+\tau)}) - E[g(\boldsymbol{\theta})])] \\ &= \frac{1}{N^2} \left\{ \sum_{k=1}^N E[(g(\boldsymbol{\theta}^{(k)}) - E[g(\boldsymbol{\theta})])^2] + 2 \sum_{\tau=1}^{N-1} \sum_{k=1}^{N-\tau} E[(g(\boldsymbol{\theta}^{(k)}) - E[g(\boldsymbol{\theta})]) (g(\boldsymbol{\theta}^{(k+\tau)}) - E[g(\boldsymbol{\theta})])] \right\} \end{aligned}$$

where $E[(g(\boldsymbol{\theta}^{(k+\tau)}) - E[g(\boldsymbol{\theta})]) (g(\boldsymbol{\theta}^{(k)}) - E[g(\boldsymbol{\theta})])] = \rho(\tau)$, for all k . Thus, $\text{Var}(\tilde{E}[g(\boldsymbol{\theta})])$ becomes

$$\begin{aligned} \text{Var}(\tilde{E}[g(\boldsymbol{\theta})]) &= \frac{1}{N^2} \sum_{k=1}^N \rho(0) + \frac{2}{N^2} \sum_{\tau=1}^{N-1} \sum_{k=1}^{N-\tau} \rho(\tau) \\ &= \frac{\rho(0)}{N} + \frac{2}{N^2} \sum_{\tau=1}^{N-1} (N-\tau) \rho(\tau) \\ &= \frac{\rho(0)}{N} \left[1 + 2 \sum_{\tau=1}^{N-1} \left(1 - \frac{\tau}{N} \right) \frac{\rho(\tau)}{\rho(0)} \right] \\ &= \frac{\rho(0)}{N} (1 + \lambda) \end{aligned}$$

where $\lambda = 2 \sum_{\tau=1}^{N-1} (1 - \tau/N) \rho(\tau) / \rho(0)$

References

- Beck, J. L., and Au, S. K. (2000). "Updating robust reliability using Markov chain simulation." *Proc., Int. Conf. on Monte Carlo Simulation*, Monte Carlo, Monaco.
- Beck, J. L., and Au, S. K. (2002). "Bayesian updating of structural models and reliability using Markov chain Monte Carlo simulation." *J. Eng. Mech.*, 128(2), 380–391.
- Beck, J. L., and Katafygiotis, L. S. (1991). "Updating of a model and its uncertainties utilizing dynamic test data." *Proc., 1st Int. Conf. on Computational Stochastic Mechanics*, Computational Mechanics Publications, Boston, 125–136.
- Beck, J. L., and Katafygiotis, L. S. (1998). "Updating models and their uncertainties: Bayesian statistical framework." *J. Eng. Mech.*, 124(4), 455–461.
- Beck, J. L., and Yuen, K. V. (2004). "Model selection using response measurements: A Bayesian probabilistic approach." *J. Eng. Mech.*, 130(2), 192–203.
- Cheung, S. H., and Beck, J. L. (2007a). "New stochastic simulation

- method for updating robust reliability of dynamic systems." *Proc., 18th Engineering Mechanics Conf.*, ASCE, Reston, Va.
- Cheung, S. H., and Beck, J. L. (2007b). "Bayesian model updating of higher-dimensional dynamic systems." *Proc., 10th Int. Conf. on Applications of Statistics and Probability in Civil Engineering (ICASP10)*, Univ. of Tokyo, Tokyo.
- Ching, J., and Chen, Y. J. (2007). "Transitional Markov chain Monte Carlo method for Bayesian model updating, model class selection, and model averaging." *J. Eng. Mech.*, 133(7), 816–832.
- Ching, J., Muto, M., and Beck, J. L. (2006). "Structural model updating and health monitoring with incomplete modal data using Gibbs Sampler." *Comput. Aided Civ. Infrastruct. Eng.*, 21(4), 242–257.
- Cox, R. T. (1961). *The algebra of probable inference*, Johns Hopkins University Press, Baltimore.
- Duane, S., Kennedy, A. D., Pendleton, B. J., and Roweth, D. (1987). "Hybrid Monte Carlo." *Phys. Lett. B*, 195(2), 216–222.
- Forest, E., and Ruth, R. D. (1990). "Fourth-order symplectic integration." *Physica D*, 43(1), 105–117.
- Geman, S., and Geman, D. (1984). "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images." *IEEE Trans. Pattern Anal. Mach. Intell.*, 6(6), 194–207.
- Griewank, A. (1989). *On automatic differentiation. Mathematical programming: Recent developments and applications*, M. Iri and K. Tanabe, eds., 83–108.
- Hastings, W. K. (1970). "Monte Carlo sampling methods using Markov chains and their applications." *Biometrika*, 57(1), 97–109.
- Jaynes, E. T. (2003). *Probability theory: The logic of science*, Cambridge University Press, London.
- Kagiwada, H., Kalaba, R., Rosakho, N., and Spingarn, K. (1986). "Numerical derivatives and nonlinear analysis." *Mathematical Concepts and methods in science and engineering 31*, Plenum Press, New York.
- Katafygiotis, L. S., and Lam, H. F. (2002). "Tangential-projection algorithm for manifold representation in unidentifiable model updating problems." *Earthquake Eng. Struct. Dyn.*, 31(4), 791–812.
- Kleinman, N., Spall, J. C., and Naiman, D. Q. (1999). "Simulation-based optimization using stochastic approximation using common random numbers." *Manage. Sci.*, 45(11), 1570–1578.
- Mackenzie, P. B. (1989). "An improved hybrid Monte Carlo method." *Phys. Lett. B*, 226(3), 369–371.
- Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). "Equations of state calculations by fast computing machines." *J. Chem. Phys.*, 21(6), 1087–1092.
- Muto, M., and Beck, J. L. (2008). "Bayesian updating of hysteretic structural models using stochastic simulation." *J. Vib. Control*, 14(1–2), 7–34.
- Neal, R. M. (1994). "An improved acceptance procedure for the hybrid Monte Carlo algorithm." *J. Comput. Phys.*, 111(1), 194–203.
- Oh, C. K., Beck, J. L., and Yamada, M. (2008). "Bayesian learning using automatic relevance determination prior with an application to earthquake early warning." *J. Eng. Mech.*, 134(12), 1013–1020.
- Papadimitriou, C., Beck, J. L., and Katafygiotis, L. S. (2001). "Updating robust reliability using structural test data." *Probab. Eng. Mech.*, 16(2), 103–113.
- Rall, L. B. (1981). "Automatic differentiation—Techniques and applications." *Lecture notes in computer science*, Vol. 120, Springer, Berlin.
- Robert, C. P., and Casella, G. (1999). *Monte Carlo statistical methods*, Springer, New York.
- Sadegh, P., and Spall, J. C. (1998). "Optimal random perturbations for multivariate stochastic approximation using a simultaneous perturbation gradient approximation." *IEEE Trans. Autom. Control*, 43(10), 1480–1484.
- Spall, J. C. (1997). "Accelerated second-order stochastic optimization using only function measurements." *Proc., 36th IEEE Conf. on Decision and Control*, 1417–1424.
- Spall, J. C. (1998a). "An overview of the simultaneous perturbation method for efficient optimization." *Johns Hopkins APL Tech. Dig.*, 19(4), 482–492.
- Spall, J. C. (1998b). "Implementation of the simultaneous perturbation algorithm for stochastic optimization." *IEEE Trans. Aerosp. Electron. Syst.*, 34(3), 817–823.
- Tierney, L. (1994). "Markov chain for exploring posterior distributions." *Ann. Stat.*, 22(4), 1701–1762.
- Wolfe, P. (1982). "Checking the calculation of gradients." *ACM TOMS*, 6(4), 337–343.