Bayesian Annealed Sequential Importance Sampling (BASIS): an unbiased version of Transitional Markov Chain Monte Carlo

Stephen Wu[*†]

Postdoctoral
CSELab
ETH-Zurich
CH -8092
Switzerland
Email: stewu@ism.ac.jp

Panagiotis Angelikopoulos

Postdoctoral
CSELab
ETH-Zurich
CH -8092
Switzerland

Costas Papadimitriou

Professor
Department of Mechanical Engineering
University of Thessaly
38334 Volos
Greece
Email: costasp@mie.uth.gr

Petros Koumoutsakos

Professor
CSELab
ETH-Zurich
CH -8092
Switzerland
Radcliffe Institute of Advanced Study
Harvard University
Cambridge, MA 02138
Department of Mechanical Engineering
Massachusetts Institute of Technology
Cambridge, MA 02138
Email: petros@ethz.ch

ABSTRACT

The Transitional Markov Chain Monte Carlo (TMCMC) is one of the efficient algorithms for performing MCMC in the context of Bayesian uncertainty quantification in parallel computing architectures. However, the features that are associated with its efficient sampling are also responsible for its introducing of bias in the sampling. We demonstrate that the Markov chains of each subsample in TMCMC may result in uneven chain lengths that distort the intermediate target distributions and introduce bias accumulation in each stage of the TMCMC algorithm. We remedy this drawback of TMCMC by proposing uniform chain lengths, with or without burn-in, so that the algorithm emphasizes Sequential Importance Sampling over MCMC. The proposed Bayesian Annealed Sequential Importance Sampling (BASIS) removes the bias of the original TMCMC and at the same time increases its parallel efficiency. We demonstrate the advantages and drawbacks of BASIS in modeling of bridge dynamics using finite elements and a disk-wall collision using discrete element methods.

Nomenclature

MCMC Markov Chain Monte Carlo
SMC Sequetial Monte Carlo
SIS Sequential Importance Sampling
TMCMC Traditional Markov Chain Monte Carlo
BASIS Bayesian Annealed Sequential Importance Sampling

1 Introduction

Bayesian inference relies on the development and implementation of effective sampling techniques for the underlying probability distributions. Markov Chain Monte Carlo (MCMC) techniques are widely considered as the most effective way of performing such samplings in high dimensional spaces. Traditional MCMC methods are inefficient in drawing samples from distributions that are peaked or multimodal. In order to remedy such situations, a large variety of Sequential Monte Carlo (SMC) and Sequential Importance Sampling (SIS) methods have been developed since some of the early publications on these subjects [1–4]. Some examples include the Annealed Importance Sampling [5], the Adaptive Metropolis-Hastings method [6], and the Iterated Batch Importance Sampling Algorithm [7], etc. Among them, [8] proposed the Transitional Markov Chain Monte Carlo (TMCMC) method with applications on Bayesian inference of structural dynamics problems, as well as other engineering applications [9]. The fundamental concept of TMCMC is to sequentially update samples drawn from prior distribution (usually easy to sample from) to the posterior distribution in the Bayesian framework. We remark the similarity of this concept with the simulated annealing method for optimization [10]. The likelihood function is annealed by an exponent between 0 and 1 to form the intermediate target distributions that guide the evolution of the samples. Such an approach can also be found in [11].

TMCMC uses the resampling method to update samples from one intermediate distribution to another. However, having multiple resampling steps results in a sample degeneracy, with the number of distinct samples decreasing as the number of steps increases. In order to resolve the degeneracy problem, an MCMC step is conducted after resampling. In this step, the j^{th} sample may be selected N_j times. In TMCMC, the total MCMC length for the j^{th} sample after resampling is proportional to N_j, which is different from the classical SMC approaches. These N_j's may vary significantly across different samples. In this paper, we show that such uneven chain lengths lead to bias in the resulting posterior sample set that is accumulated from each resampling step. Such a problem has not been recognized and explicitly studied in spite of its applications in many engineering problems [12–15]. We propose a simple remedy to this bias by adjusting the uniformity of the length for each MCMC chain, which found its root in some of the previous studies in SMC [11]. We note that the key sampling feature of this algorithm is to apply an adaptive annealing schedule to Sequential Importance Sampling. The MCMC part of the algorithm is only for improving sampling efficiency. As TMCMC has been developed for sampling posterior distributions in Bayesian inference, we name the improved algorithm as Bayesian Annealed Sequential Importance Sampling (BASIS) to better reflect these aspects of the method. The proposed algorithm improves the practicality of the original TMCMC for engineering applications.

The paper is structured as follow: we start with an overview of the originally proposed TMCMC algorithm by [8]. Then, we demonstrate the cause of the bias in the current TMCMC implementation, which is an important theoretical issue. After that, we test our proposed generalized algorithm, BASIS, on two engineering applications: parameter estimation of a disk-wall collision model and model updating of a bridge. Finally, we give some concluding remarks regarding the use of BASIS.

---

*Currently at The Institute of Statistical Mathematics, Tokyo, Japan.
†Address all correspondence for this paper to this author.

2 The Original TMCMC

In the Bayesian framework, the likelihood $p(D|\theta)$ represents the probability of observing data D given a set of model parameters θ. The posterior distribution of θ, $p(\theta|D)$, is proportional to the product of the likelihood function and a prior distribution $p(\theta)$:

$$p(\theta|D) = \frac{p(D|\theta)p(\theta)}{p(D)} \tag{1}$$

where $p(D)$ is the evidence that is used for model selection [16].

The original TMCMC method [8] provides an efficient algorithm to sample from challenging probability density functions (PDFs), such as, peaked or multimodal posterior PDFs. Furthermore, the evidence is estimated as a by-product of the method, while in classical MCMC methods this requires extra computations. The original formulation of TMCMC constructs multiple intermediate target distributions as follows:

$$\begin{gathered} f_j(\theta) \propto p(D|\theta)^{\zeta_j} p(\theta) \\ \text{for } j = 1,\ldots,m \text{ and } 0 = \zeta_1 < \zeta_2 < \cdots < \zeta_m = 1 \end{gathered} \tag{2}$$

Starting from samples drawn from the prior at stage $j = 1$, the posterior samples are obtained by updating the prior samples through $m-2$ intermediate target distributions. We denote $\Theta_j = \{\theta_{j,k} | k = 1,\ldots,N_j\}$ as a set of N_j samples of θ distributed as $f_j(\theta)$, so that Θ_{j+1} is obtained by:

1. Obtain $\tilde{\Theta}_{j+1}$ by resampling Θ_j for N_{j+1} times with probabilities proportional to $W_j = \{w_{j,k} | k = 1,\ldots,N_j\}$, where

$$w_{j,k} = \frac{f_{j+1}(\theta_{j,k})}{f_j(\theta_{j,k})} = p(D|\theta_{j,k})^{\zeta_{j+1}-\zeta_j} \tag{3}$$

   i.e., randomly pick a sample from Θ_j with probabilities proportional to W_j and add it into the set $\tilde{\Theta}_{j+1}$ (repeat N_{j+1} times).
2. For each unique sample in $\tilde{\Theta}_{j+1}$, conduct MCMC with length equal to the number of times this sample being repeated. The proposal PDF for the MCMC step is a Gaussian distribution centered at the sample with covariance equal to $\beta^2 \mathrm{COV}(\Theta_j)$, where $\mathrm{COV}(\Theta_j)$ denotes the sample covariance matrix of Θ_j and β is a user-specified scaling factor. Θ_{j+1} is the collective samples from this MCMC step.

When $\zeta_j = 1$, this procedure stops and the last set of samples Θ_m is approximately distributed according to the target posterior PDF, $p(\theta|D)$. The evidence is readily obtained from:

$$p(D) \approx \prod_{j=1}^{m-1} \left(\frac{1}{N_j} \sum_{k=1}^{N_j} w_{j,k} \right) \tag{4}$$

[8] argue that the choice of intermediate steps ζ_j should be based on the uniformity of the weights $w_{j,k}$. This is consistent with a similar argument used in the Annealed Importance Sampling method [5]. At each stage j, ζ_{j+1} is chosen such that the coefficient of variation (CV) of $w_{j,k}$ is smaller than some preset threshold γ_{CV}. [8] suggested $\gamma_{CV} = 1$ and $\beta = 0.2$. Algorithm 1 and Fig. 1 summarize the TMCMC algorithm.

TMCMC can be parallelized very efficiently, which makes it an attractive tool for uncertainty quantifications of large science and engineering problems (see for example, [17]). A parallelized version of TMCMC is available in the open-source software Π4U [18]. A MATLAB version of the parallelized TMCMC with improvements suggested in this paper is also available at http://www.cse-lab.ethz.ch/software/BASIS .

3 Bias in the Original TMCMC

In the original TMCMC algorithm, the MCMC step after resampling is intended to avoid degeneracy, i.e., the reduction of unique samples after multiple resampling steps. Also, a longer chain length is assigned to more important samples in order to better explore the important region of the parameter domain. However, the use of non-uniform chain lengths may introduce bias to the samples. This is a natural result deduced from the well known mixing problem for MCMC sampling [19].

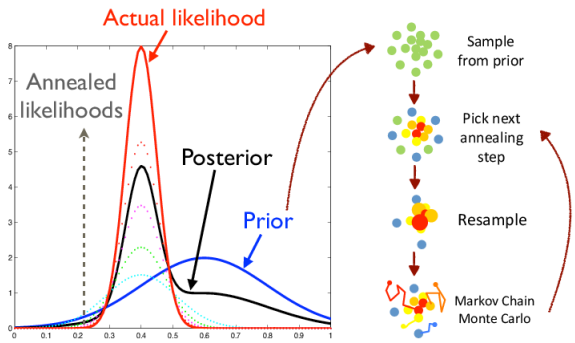

Fig. 1. Summary of the TMCMC algorithm.

Although, in theory, the original resampling scheme in TMCMC converges to the target distribution asymptotically, it fails in many practical applications due to the error coming from the multiple chains mixing phenomenon of MCMC. In this section, we demonstrate this theoretical issue though Gaussian posterior distributions.

We consider the likelihood function to be an n-dimensional standard Gaussian distribution, with zero mean and identity covariance matrix. With a uniform prior, the posterior PDF equals a truncated Gaussian distribution. If we choose a very broad uniform prior that covers almost all the probability content in the standard Gaussian likelihood, the posterior PDF is approximately equivalent to the Gaussian likelihood. The evidence is equal to the product of the Gaussian cumulative density function evaluated inside the uniform distribution domain and the constant value of the uniform distribution. To investigate how the non-uniform chain length affects the bias, we introduce one more parameter, called the maximum chain length l_{max}, to the original TMCMC resampling step. This parameter is included in our BASIS algorithm to generalize TMCMC for different uses (see Section 4). After assigning chain length to each sample following resampling, if a chain has a length longer than l_{max}, the chain is split into multiple chains uniformly such that each chain is shorter than l_{max}. For example, if we have a chain with length 28 and $l_{max} = 10$, the chain is split into three chains with length 9, 9 and 10. We represent with $l_{max} = \infty$ the original TMCMC algorithm. We note that $l_{max} = 1$ gives the extreme case of guaranteed uniform chain length. Note that l_{max} also controls the efficiency of parallelization of TMCMC. Since the longest chain length is the limiting factor of the run time of parallelized TMCMC [18], a smaller l_{max} implies a more uniform chain length, thus resulting in more efficient parallelization. In other words, the uniform chain length using small l_{max} provides a natural work balance in

ALGORITHM 1: Original TMCMC

Input: Define likelihood function $p(D|\theta)$ and prior distribution $p(\theta)$; set parameters β and γ_{CV}, number of samples N_j for each stage j (usually same for all stages) and maximum number of stages N_{max}.
Output: Θ_{final} — a set of posterior samples approximating $p(\theta|D)$, S — an estimation for the evidence $p(D)$.
Draw initial sample set $\Theta_1 = \{\theta_{1,k} | k = 1, \ldots, N_1\}$ from $p(\theta)$;
Initialize $\zeta_1 \leftarrow 0$, $S \leftarrow 1$ and $j \leftarrow 1$;
repeat
  Choose ζ_{j+1} such that CV of $w_{j,k} < \gamma_{CV}$ and $\zeta_{j+1} \leq 1$;
  Calculate $w_{j,k}$ with the chosen ζ_{j+1};
  $S \leftarrow S \cdot \frac{1}{N_j} \sum_{k=1}^{N_j} w_{j,k}$;
  for $k = 1, \ldots, N_{j+1}$ **do**
    $\tilde{\theta}_{j+1,k} \leftarrow$ draw a sample from Θ_j with probabilities $\propto w_{j,k}$;
  end
  for *each unique sample in* $\{\tilde{\theta}_{j+1,k} | k = 1, \ldots, N_{j+1}\}$ **do**
    Perform MCMC with length equal to the number of copies of the unique sample;
    [*BASIS adds an upper bound l_{max} to the chain length]
    Add resulting samples into Θ_{j+1};
  end
  $j \leftarrow j + 1$;
until $\zeta_j = 1$;
$\Theta_{final} \leftarrow \Theta_j$;

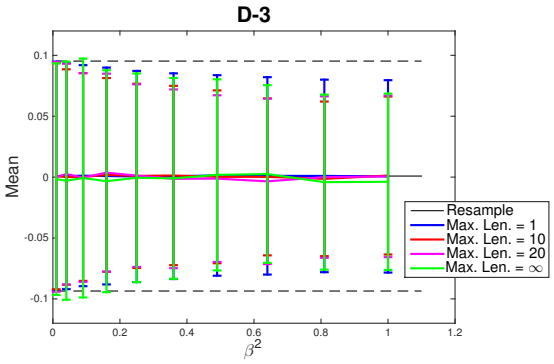

Fig. 2. Mean estimates of running a single stage TMCMC on a 3D standard Gaussian likelihood with varying maximum chain length from 1 to ∞. The black lines represent results directly after resampling and before MCMC (solid line – mean value; dashed lines – one standard deviation bound). Only the 3rd dimension (D-3) is shown. The other two dimensions have the exact same behavior.

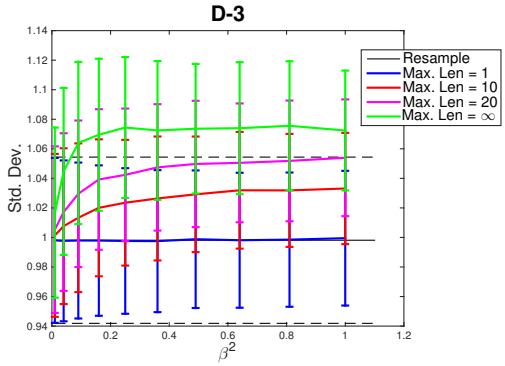

Fig. 3. Standard deviation estimates of running a single stage TMCMC on a 3D standard Gaussian likelihood with varying maximum chain length from 1 to ∞. The black lines represent results directly after resampling and before MCMC (solid line – mean value; dashed lines – one standard deviation bound). Only the 3rd dimension (D-3) is shown. The other two dimensions have the exact same behavior.

parallelization.

3.1 3D standard Gaussian posterior

In order to demonstrate the statements made above, we consider a 3D standard Gaussian likelihood with uniform prior between -5 to 5 in each dimension. We enforce only one annealing step, i.e., $\zeta_2 = 1$, in order to focus on a single resampling step of the TMCMC algorithm. We vary l_{max} to be 1, 10, 20 and ∞, and the proposal size parameter for the MCMC step β to be from 0.1 to 1 with interval 0.1. We investigate the accuracy of the resulting estimates for the mean, standard deviation and correlation among all dimensions by running each setup for 1000 times. Figure 2 to 5 show the results with error bars.

The mean and correlation estimates for different values of l_{max} are consistently accurate with no bias. Also, there is no significant change on the acceptance rate for the MCMC step as l_{max} varies. However, there is clear bias induced in the standard deviation by the increase of l_{max}. The bias increases as β^2 increases, i.e., the proposal PDF is broader. In the original TMCMC algorithm, this bias is accumulated in each stage. Since the evidence is estimated using samples from all intermediate stages in TMCMC, the final estimation will have a significant error accumulated from each stage. To visualize the error, we run the same problem setup using the TMCMC algorithm with adaptively chosen ζ_j (varying l_{max} and β^2). We use 1000 samples for each stage and $\gamma_{CV} = 1$. We run each parameter setup for 200 times to collect statistics. We define the ln(Evidence) residuals as the difference between the log-evidence computed from TMCMC and the actual log-evidence. Figure 6 shows that an increase of l_{max} leads to a clear bias for the evidence estimate. Note that the acceptance rate and number of stages in TMCMC are consistent for all values of l_{max}.

The original TMCMC algorithm often has relatively short Markov chains as it is limited by the number of samples required by the user. It is possible that the sample set obtained after resampling is not distributed close to the target intermediate distribution $f_j(\theta)$. More importantly, it is unlikely that the samples from different chains are well mixed in this situation. Hence, the short Markov chains are not at the stationary state. [20] modified the TMCMC for application on finite fault earthquake source inversion by adding a long burn-in period to the Markov chains. As a result, an improvement is observed along with some other modifications proposed in the paper. This approach is also suggested in [21] although the bias

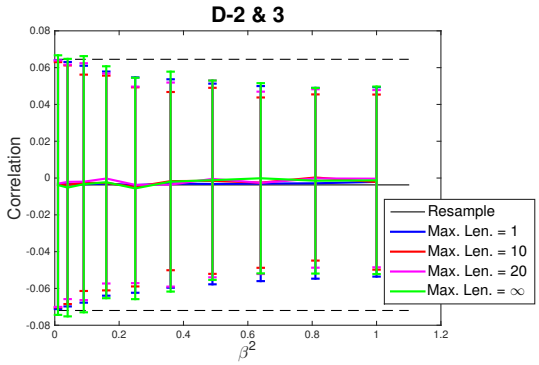

Fig. 4. Correlation estimates of running a single stage TMCMC on a 3D standard Gaussian likelihood with varying maximum chain length from 1 to ∞. The black lines represent results directly after resampling and before MCMC (solid line – mean value; dashed lines – one standard deviation bound). Only the correlation between the 2nd and the 3rd dimension (D-2 & 3) is shown. The other two pairs have the exact same behavior.

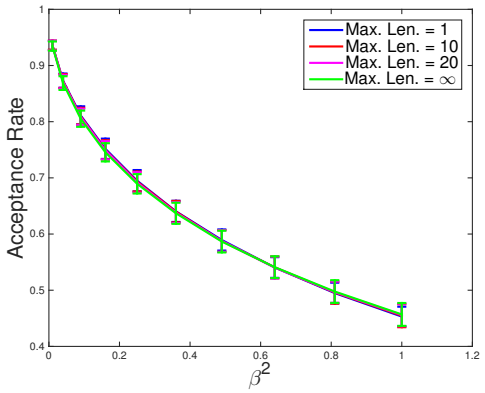

Fig. 5. Acceptance rate of all MCMC runs when running a single stage TMCMC on a 3D standard Gaussian likelihood with varying maximum chain length from 1 to ∞.

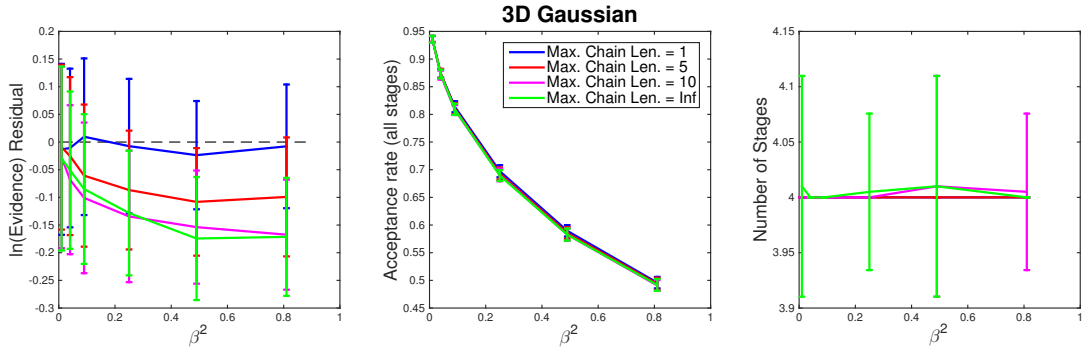

Fig. 6. Results of running TMCMC with varying l_{max} and β^2.

problem is not discussed there. Here, we run the same problem setup as shown above using TMCMC with various burn-in period ranging from 0 to 50. We use 1000 samples for each stage and $\gamma_{CV} = 1$. We run each parameter setup for 100 times to collect statistics of the results. Figure 7 compares results between $l_{max} = 1$ and $l_{max} = \infty$ (original TMCMC). The results show that a longer burn-in period reduces the bias for $l_{max} = \infty$ as each chain converges to the stationary distribution (see [19] and [22] for discussions on multiple Markov chains convergence). On the other hand, the burn-in period does not affect the already good convergence for $l_{max} = 1$. Note that for $l_{max} = 1$, the parallelism efficiency is improved. In contrast, the burn-in period introduces extra computational efforts and there is no rigorous way to choose the length of a burn-in period. Hence, it is more beneficial to remove the bias using more uniform chain length, e.g., $l_{max} = 1$.

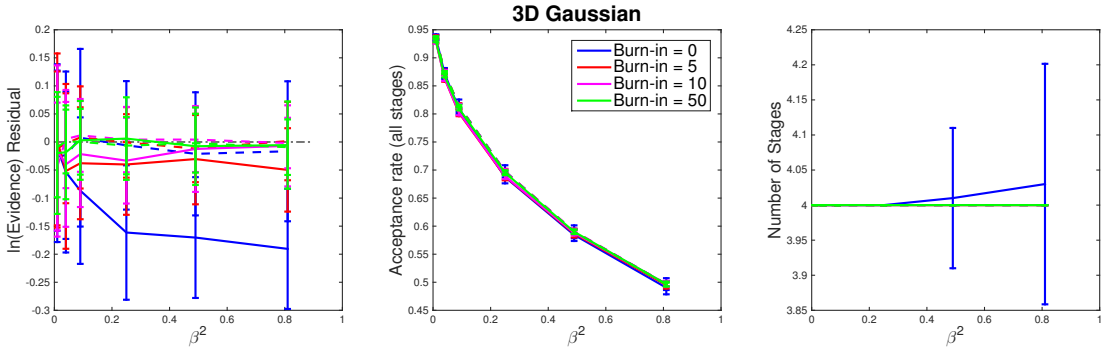

Fig. 7. Results of running different TMCMC with burn-in period for every MCMC step. Dashed lines represent $l_{max} = 1$, and solid lines represent $l_{max} = \infty$, i.e., the original TMCMC algorithm.

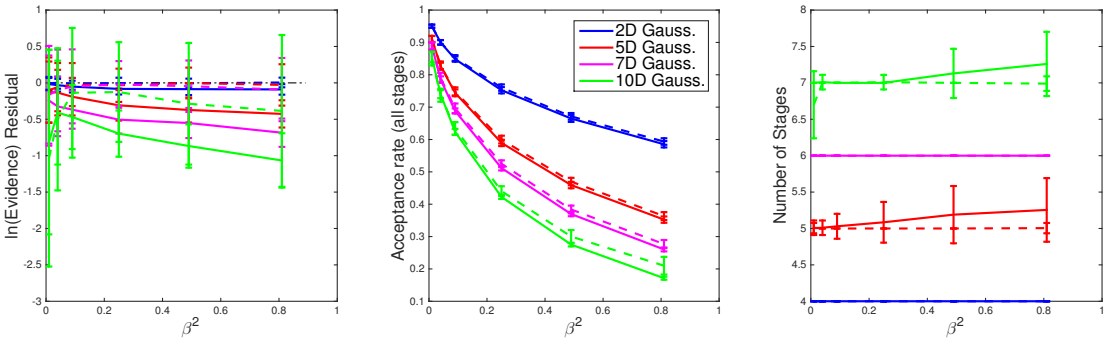

Fig. 8. Results of running different TMCMC with various dimension Gaussian likelihoods. Dashed lines represent $l_{max} = 1$, and solid lines represent $l_{max} = \infty$, i.e., the original TMCMC algorithm.

3.2 Higher dimensional, bi-modal and unidentifiable Gaussian posterior

To further verify the claims in the previous section, we compare results from the $l_{max} = 1$ and $l_{max} = \infty$ using various types of more challenging posterior distributions. First, we consider n dimensional standard Gaussian likelihoods, where $n = 2, 5, 7, 10$. We choose n dimensional uniform distributions as the prior. The range for each dimension is the same (between -5 and 5). We use $\gamma_{CV} = 1$, 1000 samples for $n = 2$ and 5 and 2000 samples for $n = 7$ and 10. No burn-in period is considered in this case. We run each parameter setup for 100 times to collect statistics of the results. Figure 8 shows that the reduction of the bias from using $l_{max} = 1$ is consistently observed in different dimensional spaces. Note that it is expected to observe a decrease of acceptance rate and an increase of the number of stages when dimension increases. This is because the prior space is relatively larger with respect to the importance region of the likelihood in a higher dimensional space.

Next, we consider a bi-modal likelihood composed by two 2D Gaussian distributions:

Peak 1: mean = $(-3.5, 0)$ and covariance matrix = 1.0 $\times$ 2D identity matrix
Peak 2: mean = $(3.5, 0)$ and covariance matrix = 0.5 $\times$ 2D identity matrix

The final bi-modal likelihood is constructed as a weighted sum of the two Gaussian distributions, with weights 0.7 and 0.3 for peak 1 and 2, respectively. A uniform prior between -7 and 7 is chosen for all dimensions. We use $\gamma_{CV} = 1$ and vary the number of samples per stage to be 200, 500 and 1000. No burn-in period is considered in this case. We run each parameter setup for 100 times to collect statistics of the results. Figure 9 to 12 show that there is bias induced in the standard deviation by using $l_{max} = \infty$, thus, the evidence estimation is inaccurate as demonstrated in Fig. 9. The results are insensitive to the number of samples per stage. Note that because the two peaks are designed to be well separated from each other, the posterior samples from TMCMC is well clustered into two peaks. Hence, we can compute the corresponding mean, standard deviation and correlation for each peak.

Finally, we approximate an unidentifiable posterior using a 6D zero-mean Gaussian distribution with no correlation between all dimensions. The variance of dimension one (equals 2.0) is 10 times of the other dimensions (equals 0.2). Dimension one is considered as unidentifiable comparing to the other dimensions because of its significantly larger variance. A uniform prior between -10 and 10 is chosen for all dimensions. We use $\gamma_{CV} = 1$ and vary the number of sample to be 100, 500 and 1000. No burn-in period is considered. Figure 13 to 15 show that the evidence estimates are sensitive to the β^2 value. This is because when β^2 is too small, the samples cannot explore the unidentifiable region well enough in each

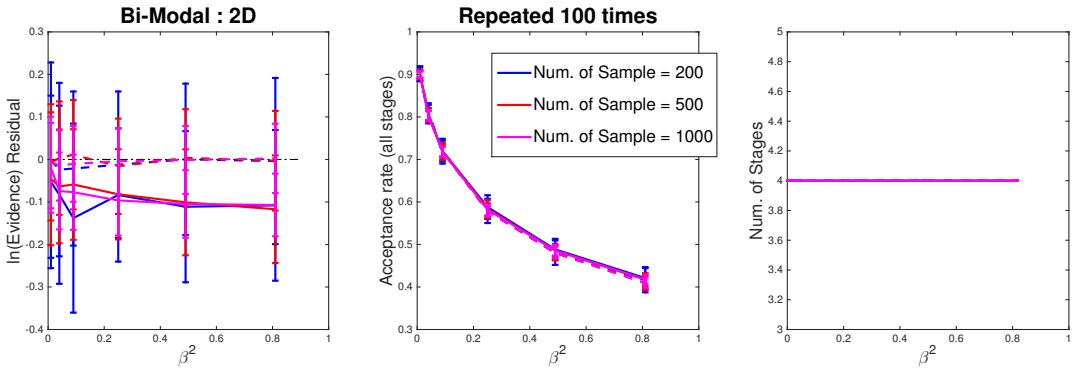

Fig. 9. Results of running different TMCMC with bi-modal Gaussian likelihoods. Dashed lines represent $l_{max} = 1$ and solid lines represent $l_{max} = \infty$, i.e., the original TMCMC algorithm.

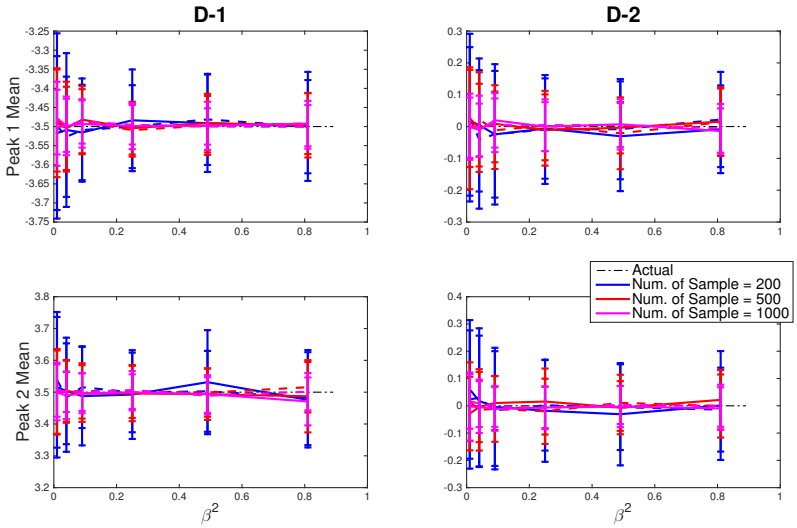

Fig. 10. Mean estimates of running different TMCMC with bi-modal Gaussian likelihoods (two dimensions: D-1 and D-2). Dashed lines represent $l_{max} = 1$ and solid lines represent $l_{max} = \infty$, i.e., the original TMCMC algorithm.

TMCMC stage. Hence, small β value shall be avoided when unidentifiable region is expected in the posterior distribution. Also, the standard deviation estimates have a slight bias for the $l_{max} = \infty$ case. The bias is not very significant because the likelihood is relatively peaked in all dimension except in dimension one. This issue will be revisited in the bridge example in Section 4.2.

3.3 Stage-wise convergence of TMCMC

In the previous examples, the number of prior samples is large enough to cover all important regions of the likelihood function. Hence, $l_{max} = 1$ is enough to explore the domain to obtain an accurate estimate of the evidence. In practice, the likelihood function may be very narrow with respect to the chosen prior. We hypothesize that the resampling step is introducing significant bias in each TMCMC stage for this case, especially in the first few stages. The burn-in period becomes very important to correct the bias. Here, we choose to study a single Gaussian likelihood function and a uniform prior as this gives an analytical form for the intermediate evidence functions $F_j = \int p(D|\theta)^{\zeta_j} p(\theta)\, d\theta$. This allows us to study the stage-wise convergence of TMCMC.

Given an 1D Gaussian likelihood with mean μ and variance σ^2 and a uniform prior between $-a$ and a, one can derive:

$$F_j = \frac{1}{2a\sqrt{\zeta_j}} (2\pi\sigma^2)^{\frac{1-\zeta_j}{2}} \operatorname{erf}\left(\frac{a-\mu}{\sigma}\sqrt{\frac{\zeta_j}{2}}\right) \tag{5}$$

where $\operatorname{erf}(\cdot)$ denotes the error function. This can be generalized to high dimension Gaussian likelihood with diagonal

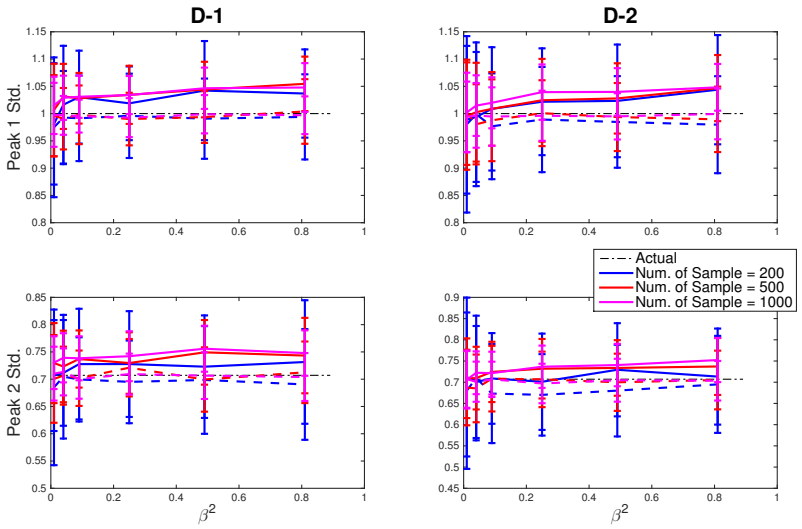

Fig. 11. Standard deviation estimates of running different TMCMC with bi-modal Gaussian likelihoods (two dimensions: D-1 and D-2). Dashed lines represent $l_{max} = 1$ and solid lines represent $l_{max} = \infty$, i.e., the original TMCMC algorithm.

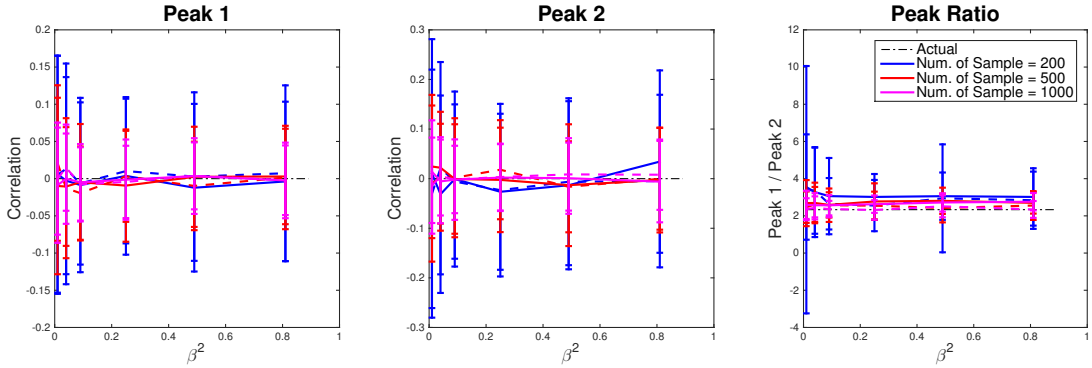

Fig. 12. Estimates for correlation and the weights' ratio between the two Gaussian distributions when running different TMCMC with bi-modal Gaussian likelihoods. Dashed lines represent $l_{max} = 1$ and solid lines represent $l_{max} = \infty$, i.e., the original TMCMC algorithm.

covariance matrix and uncorrelated prior in each dimension. In this study, we choose a 3D Gaussian likelihood with mean 1 and variance 0.04 for each dimension independently and a uniform prior between -5 and 5 for each dimension. We run TMCMC with different burn-in settings, each for 100 times to collect statistics of the evidence estimates. In all cases, more than 80% out of the 100 runs are completed with 8 stages. We focus our analysis on those cases in order to obtain a consistent conclusion on the stage-wise convergence. Figures 16, 17, 18 and 19 show results for no burn-in, 20 steps burn-in at the first 2 stages, 50 steps burn-in at the first 2 stages and 20 steps burn-in at all stages, respectively.

Based on the results we wish to make the following remarks:

1. A larger evidence estimation error in the early stages
2. The error is independent of the annealing step size $\zeta_{j+1} - \zeta_j$, which may be an evidence of the success of the automatic step size choice in TMCMC
3. Adding burn-in in early stages reduces error in later stages

Note that the results between using 20 steps burn-in and 50 steps burn-in for the first 2 stages are similar. This indicates that 20 steps burn-in is long enough in this case. It is not obvious from the figures that adding burn-in only in the early stages can have significant improvement toward the evidence estimation. Table 1 shows the accuracy of the final evidence estimate for different burn-in setting. One can observe a significant reduction in the mean and standard deviation of estimation error after adding burn-in to the first 2 stages only. As a result, for problems that involve computationally costly likelihood evaluation, adding burn-in only in the early stages is an efficient alternative to improve the performance of the TMCMC as compared to adding burn-in for all stages. Furthermore, note that the bias for the $l_{max} = \infty$ case is not significant as compared to our studies in the previous sections. This is because when the posterior distribution is very peaked at the optimal values, the bias

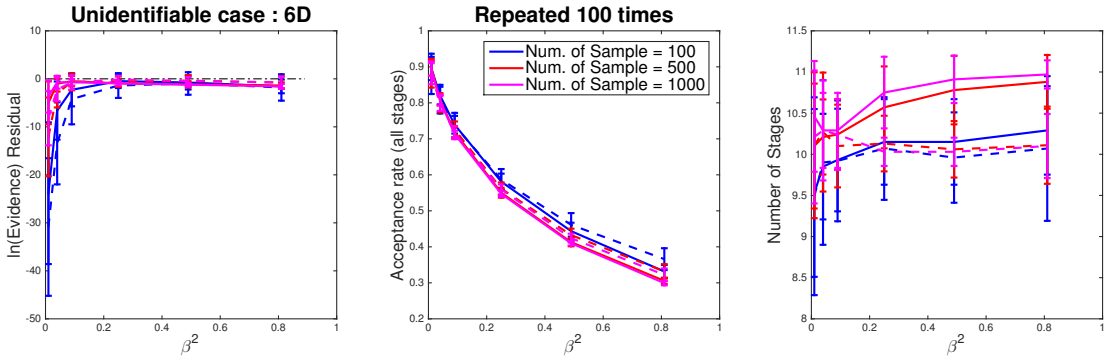

Fig. 13. Results of running different TMCMC with approximately unidentifiable Gaussian likelihoods. Dashed lines represent $l_{max} = 1$ and solid lines represent $l_{max} = \infty$, i.e., the original TMCMC algorithm.

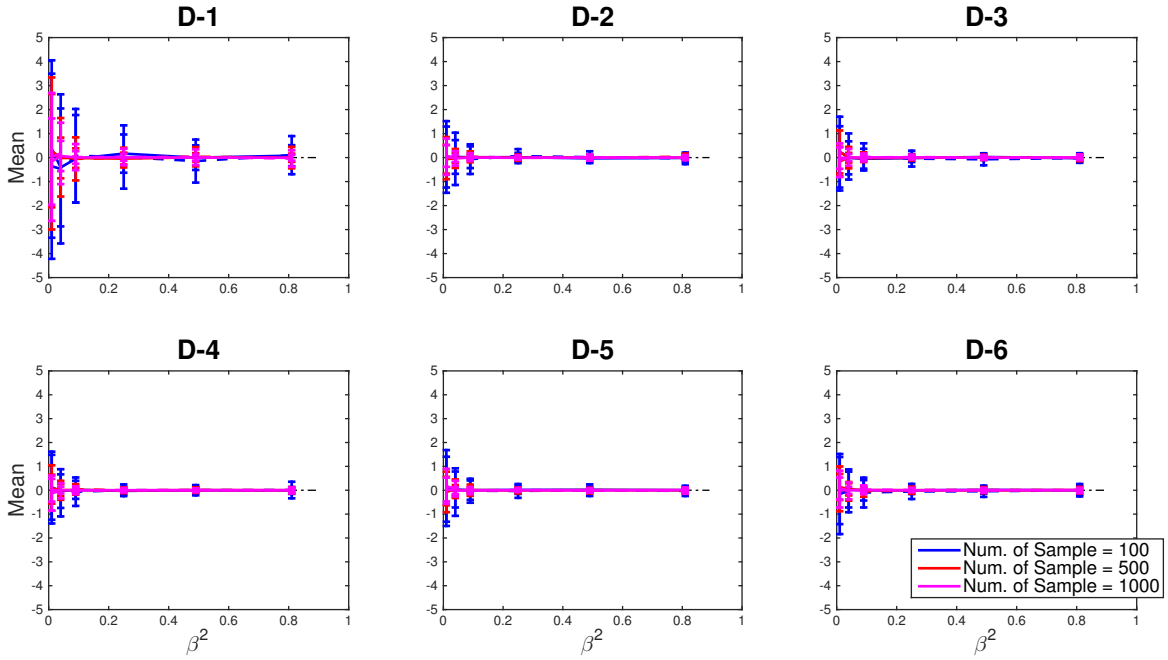

Fig. 14. Mean estimates of running different TMCMC with approximately unidentifiable Gaussian likelihoods (six dimensions: D-1, . . ., D-6). Dashed lines represent $l_{max} = 1$ and solid lines represent $l_{max} = \infty$, i.e., the original TMCMC algorithm.

coming from the standard deviation estimation becomes negligible (further discussion in the next section).

4 Applications

In this section, we investigate the bias problem of the original TMCMC algorithm ($l_{max} = \infty$) and compare its results with those obtained using $l_{max} = 1$ in two applications: (1) calibrating the force-displacement model for a Discrete Element Method (DEM) on simulations of disk-wall collision [23], and (2) parameter estimation of a high fidelity model of a bridge [24]. The setup of both problems follows the protocol outlined in [18].

4.1 Parameter estimation of a disk-wall collision model

We simulate the collision of a steel disk with a steel wall using DEM [23]. Figure 20 shows an example of the simulation scenario. We calibrate four parameters in a force-displacement model specified by a normal and tangential force based on a non-linear model [25, 26]: k^t and μ are the tangential spring stiffness and the friction coefficient of the material, respectively; α^n and γ^n are the exponential term and the dissipative constant, respectively, that describe the nonlinear nature of the viscoelastic force-displacement law. Two extra parameters σ_1^2 and σ_2^2 are used to model the additive noise for both normal and tangential velocity data, respectively. The data set and the likelihood formulation are described in [23]. Table 2

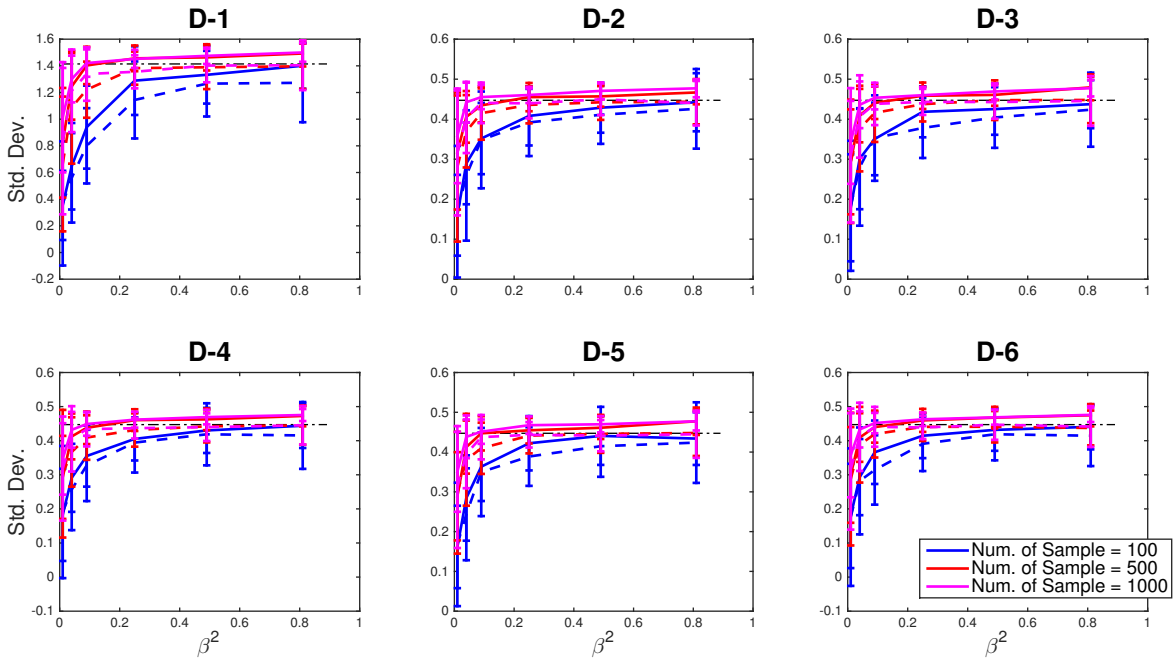

Fig. 15. Standard deviation estimates of running different TMCMC with approximately unidentifiable Gaussian likelihoods (six dimensions: D-1, . . ., D-6). Dashed lines represent $l_{max} = 1$ and solid lines represent $l_{max} = \infty$, i.e., the original TMCMC algorithm.

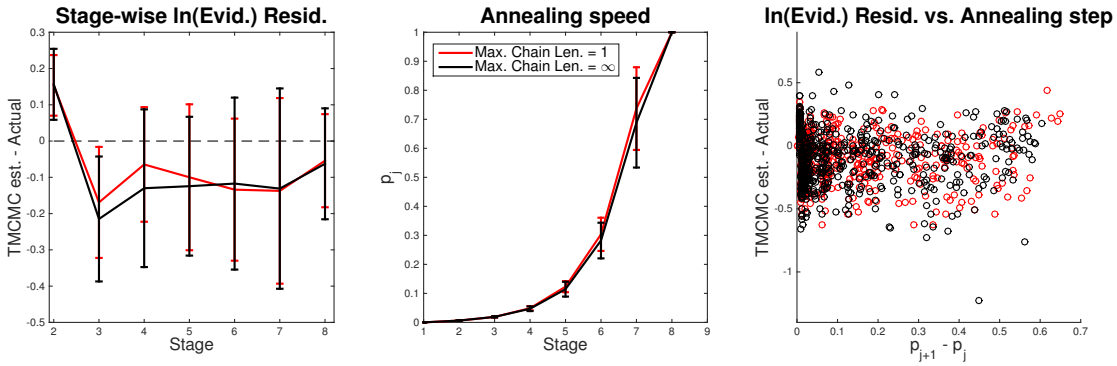

Fig. 16. Testing stage-wise evidence convergence of TMCMC using 3D Gaussian likelihood with uniform prior. No burn-in in all stages.

lists the ranges of the independent uniform priors for all parameters in this model.

We perform Bayesian uncertainty quantification for model calibration using $l_{max} = 1$ and $l_{max} = \infty$ with different number of samples per stage and burn-in period settings. Each setting is repeated 100 times in order to obtain representative statistics on the resulting estimations. The algorithm parameters β and γ_{CV} are set to the suggested values 0.2 and 1, respectively. Table 3 and 4 show the mean (E) and coefficient of variation (CV) of the six model parameters and the log-evidence value from runs with $l_{max} = \infty$ and $l_{max} = 1$, respectively. According to our study with the Gaussian likelihoods, the case of 20-step burn-in for all stages with 4000 samples with $l_{max} = 1$ can be taken as a reference for converged results, and the results are consistent for both $l_{max} = 1$ and $l_{max} = \infty$. For the case of $l_{max} = \infty$, the log-evidence value converges to around 16.66 as the number of samples per stage increases from 2000 to 8000 with no burn-in period. This is clear bias comparing to the reference value of 17.28 with a low CV value of 0.68%. Note that when a burn-in period is added, the bias is significantly reduced. On the other hand, for $l_{max} = 1$, the log-evidence estimations are converging to the reference value as the number of samples per stage increases without any burn-in period. Note that by adding burn-in period to only the early stages for the 2000 samples case, there is already a significant improvement in the log-evidence estimation (relative to the low CV value). Figure 21 shows an example of the posterior samples taken from one run of the reference case with $l_{max} = 1$. One can observe two unidentifiable parameters and a bi-modal parameter in the posterior distribution. The other three parameters are not very peaked at the optimal values. In this case, the bias from the original TMCMC algorithm is more obvious. In the next example, we show that the bias becomes irrelevant when the posterior distribution is very peaked around some optimal

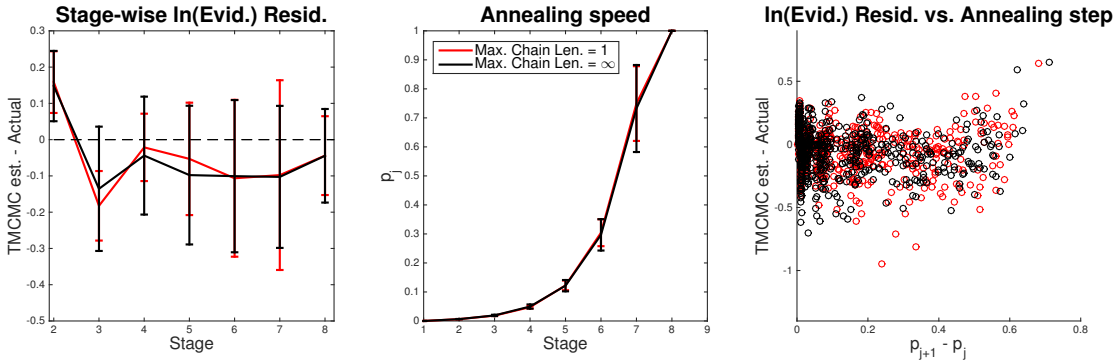

Fig. 17. Testing stage-wise evidence convergence of TMCMC using 3D Gaussian likelihood with uniform prior. 20 steps of burn-in for each Markov chain in the first two stages only.

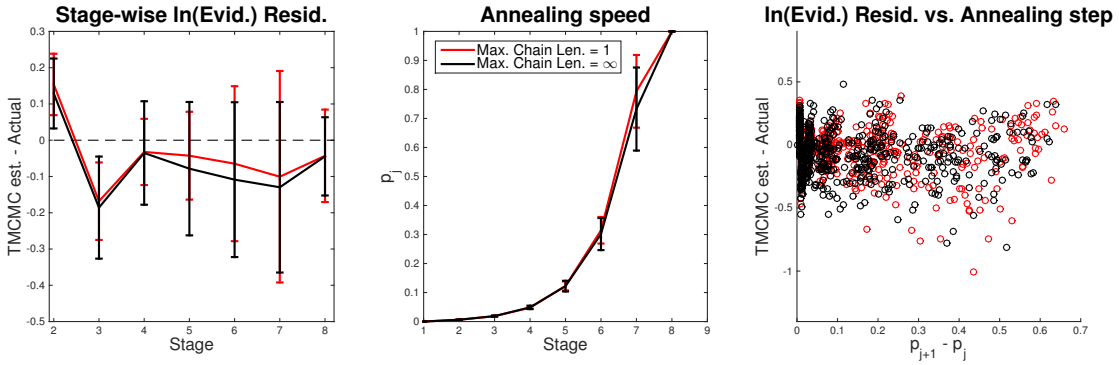

Fig. 18. Testing stage-wise evidence convergence of TMCMC using 3D Gaussian likelihood with uniform prior. 50 steps of burn-in for each Markov chain in the first two stages only.

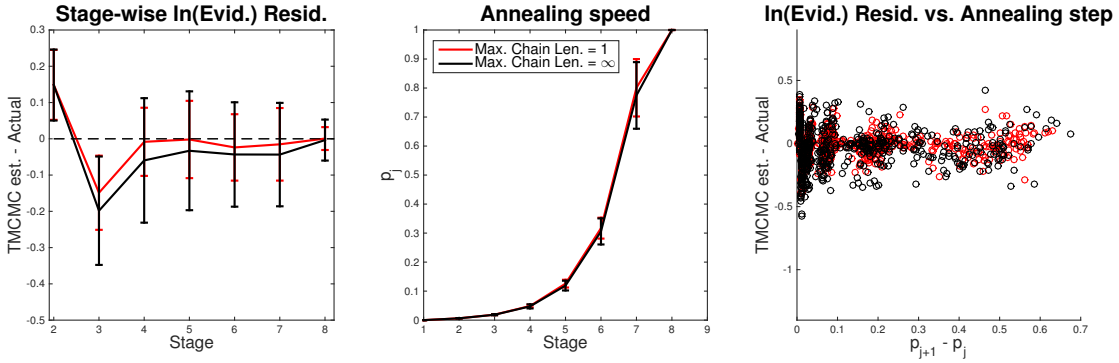

Fig. 19. Testing stage-wise evidence convergence of TMCMC using 3D Gaussian likelihood with uniform prior. 20 steps of burn-in for each Markov chain in all stages.

values.

4.2 Model updating of a bridge

This example involves parameter estimation of a high fidelity finite element model of the Metsovo bridge based on modal data [24]. We calibrate five model parameters, which represent the modulus of elasticity (normalized by a set of predetermined nominal values) of five subdivided structural components (Fig. 22). θ_1 and θ_2 denote the two pier components of the bridge, while θ_3 to θ_5 denote the structural components of the deck. An extra parameter σ is used in the model, which denotes the standard deviation of an additive Gaussian noise to the finite element model simulation. A uniform prior is used for all parameters, with range 0.2 to 2 for θ_1 to θ_5 and 0.00001 to 1 for σ. The algorithm parameters β and γ_{CV} are set to the suggested values 0.2 and 1, respectively. 1000 samples are used in each annealing stage.

Table 5 shows the mean (E) and coefficient of variation (CV) of the six model parameters and the log-evidence value

Table 1. Total evidence error of TMCMC using 3D Gaussian likelihood with uniform prior (statistics from 100 runs).

Cases	TMCMC estimate – Actual evidence			
	$l_{max} = 1$		$l_{max} = \infty$	
	Mean	Std. Dev.	Mean	Std. Dev.
Burn-in = 0 [All stages]	−0.495	1.163	−0.583	0.874
Burn-in = 20 [First 2 stages]	−0.190	0.668	−0.180	0.679
Burn-in = 50 [First 2 stages]	−0.169	0.761	−0.270	0.710
Burn-in = 20 [All stages]	−0.010	0.281	−0.118	0.421

Table 2. Boundaries for the independent uniform priors of all parameters in the force-displacement model.

Parameter	Min.	Max.
μ	0.05	0.2
α^n	0.15	1.35
γ^n	0.1	15.0
k^t	$0.1 \cdot 10^4$	$15.0 \cdot 10^4$
σ_1^2	0.0	$1 \cdot 10^{-3}$
σ_2^2	0.0	$1 \cdot 10^{-3}$

Table 3. Estimates of ln(Evidence), mean (E) and coefficient of variation (CV) of each parameter for the disk-wall collision example with $l_{max} = \infty$. Numbers in parenthesis are coefficient of variation for each estimate calculated from 100 trials. All TMCMC runs take 7 – 9 stages to finish.

($l_{max} = \infty$)	No burn-in			20-step burn-in [All stages]	
	2000 samples,	4000 samples,	8000 samples,	2000 samples,	4000 samples
E_μ	0.106 (0.97%)	0.106 (0.62%)	0.106 (0.41%)	0.106 (0.45%)	0.106 (0.35%)
CV_μ	6.03% (7.75%)	6.06% (6.09%)	6.09% (3.57%)	5.97% (4.53%)	6.02% (3.16%)
E_{α^n}	0.471 (5.67%)	0.471 (4.81%)	0.468 (3.32%)	0.464 (3.64%)	0.466 (2.24%)
CV_{α^n}	15.37% (24.92%)	16.43% (23.49%)	17.13% (16.30%)	18.37% (18.90%)	18.17% (13.68%)
E_{γ^n}	7.649 (19.01%)	7.727 (14.57%)	7.625 (10.49%)	7.547 (9.39%)	7.558 (5.81%)
CV_{γ^n}	50.01% (20.26%)	51.21% (17.79%)	53.32% (12.10%)	55.35% (12.57%)	54.89% (7.96%)
E_{k^t}	1.01e+4 (6.85%)	1.00e+4 (5.64%)	1.00e+4 (3.39%)	9.89e+3 (4.92%)	9.91e+3 (3.37%)
CV_{k^t}	27.34% (30.37%)	29.23% (22.32%)	29.57% (13.26%)	30.88% (17.46%)	30.76% (12.13%)
E_{σ_1}	5.22e-4 (18.94%)	5.29e-4 (12.27%)	5.18e-4 (8.67%)	5.13e-4 (7.31%)	5.15e-4 (4.94%)
CV_{σ_1}	48.82% (25.44%)	49.95% (15.46%)	52.44% (10.89%)	53.68% (8.74%)	53.72% (6.30%)
E_{σ_2}	3.69e-4 (23.01%)	3.60e-4 (15.54%)	3.46e-4 (12.28%)	3.25e-4 (9.82%)	3.17e-4 (7.32%)
CV_{σ_2}	69.16% (19.85%)	73.38% (12.00%)	74.94% (9.36%)	79.54% (5.77%)	81.65% (4.30%)
ln(Evid.)	16.539 (3.12%)	16.651 (1.96%)	16.660 (1.42%)	17.140 (1.39%)	17.216 (0.99%)

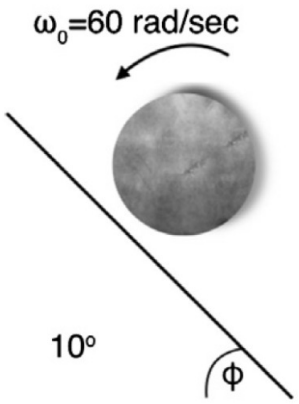

Fig. 20. Simulation of a disk impacting at an inclined surface [18].

Table 4. Estimates of ln(Evidence), mean (E) and coefficient of variation (CV) of each parameter for the disk-wall collision example with $l_{max} = 1$. Numbers in parenthesis are coefficient of variation for each estimate calculated from 100 trials. All TMCMC runs take 7 – 9 stages to finish. Bolded values are the reference to compare the other results with (including Tab. 3).

($l_{max} = 1$)	No burn-in			20-step burn-in	
				[First 3 stages]	[All stages]
	2000 samples,	4000 samples,	8000 samples,	2000 samples,	4000 samples
E_{μ}	0.106 (2.42%)	0.106 (1.82%)	0.106 (1.60%)	0.106 (0.80%)	**0.106 (0.17%)**
CV_{μ}	5.91% (20.97%)	5.87% (18.16%)	5.90% (15.44%)	5.94% (8.59%)	**5.96% (1.84%)**
E_{α^n}	0.470 (9.60%)	0.463 (10.20%)	0.467 (7.01%)	0.463 (5.12%)	**0.463 (1.56%)**
CV_{α^n}	14.77% (46.97%)	16.11% (43.29%)	16.18% (29.80%)	18.52% (25.03%)	**19.09% (8.02%)**
E_{γ^n}	7.744 (28.58%)	7.461 (27.18%)	7.543 (20.86%)	7.534 (14.29%)	**7.508 (4.17%)**
CV_{γ^n}	46.92% (43.02%)	52.95% (40.80%)	52.63% (29.68%)	55.51% (18.11%)	**56.25% (5.12%)**
E_{k^t}	9.94e+3 (12.82%)	1.01e+4 (8.69%)	9.87e+3 (9.83%)	9.88e+3 (5.70%)	**9.93e+3 (1.77%)**
CV_{k^t}	28.07% (45.02%)	27.20% (37.62%)	30.07% (34.68%)	30.81% (20.72%)	**30.69% (6.59%)**
E_{σ_1}	5.39e-4 (25.08%)	5.29e-4 (23.74%)	5.15e-4 (20.26%)	5.09e-4 (14.69%)	**5.08e-4 (3.33%)**
CV_{σ_1}	45.56% (42.67%)	48.83% (36.29%)	51.02% (27.35%)	54.01% (21.84%)	**54.99% (3.75%)**
E_{σ_2}	3.97e-4 (36.14%)	3.65e-4 (29.15%)	3.43e-4 (26.43%)	3.30e-4 (18.65%)	**3.15e-4 (4.85%)**
CV_{σ_2}	62.30% (35.51%)	70.93% (32.78%)	78.84% (35.88%)	80.02% (15.42%)	**82.63% (2.93%)**
ln(Evid.)	16.832 (4.31%)	17.030 (4.16%)	17.176 (2.75%)	17.210 (1.56%)	**17.28 (0.68%)**

from runs with $l_{max} = 1$ and $l_{max} = \infty$ and with different burn-in period settings. The estimations are obtained from the mean results after 20 repeated runs with the coefficient of variation of the estimate in the parentheses. In this study, no analytical solution is available for comparison. According to our study with the Gaussian likelihoods, the cases of 20-step burn-in for all stages with $l_{max} = 1$ can be taken as a reference for converged results, and the results are consistent for both $l_{max} = 1$ and $l_{max} = \infty$ with relatively small CV values. These results are also consistent with the findings in [18].

This example has a very peaked posterior distribution, where almost all of the parameters have CV around or less than 1%. Therefore, the bias from the $l_{max} = \infty$ case does not have a significant influence in this case. Because $l_{max} = \infty$ is more efficient for searching the high probability region (longer chain for largely resampled samples), it outperforms the $l_{max} = 1$ case in this particular example. Note that by adding burn-in to early stages only, the result is significantly improved in terms of the variation among the 20 repeated runs.

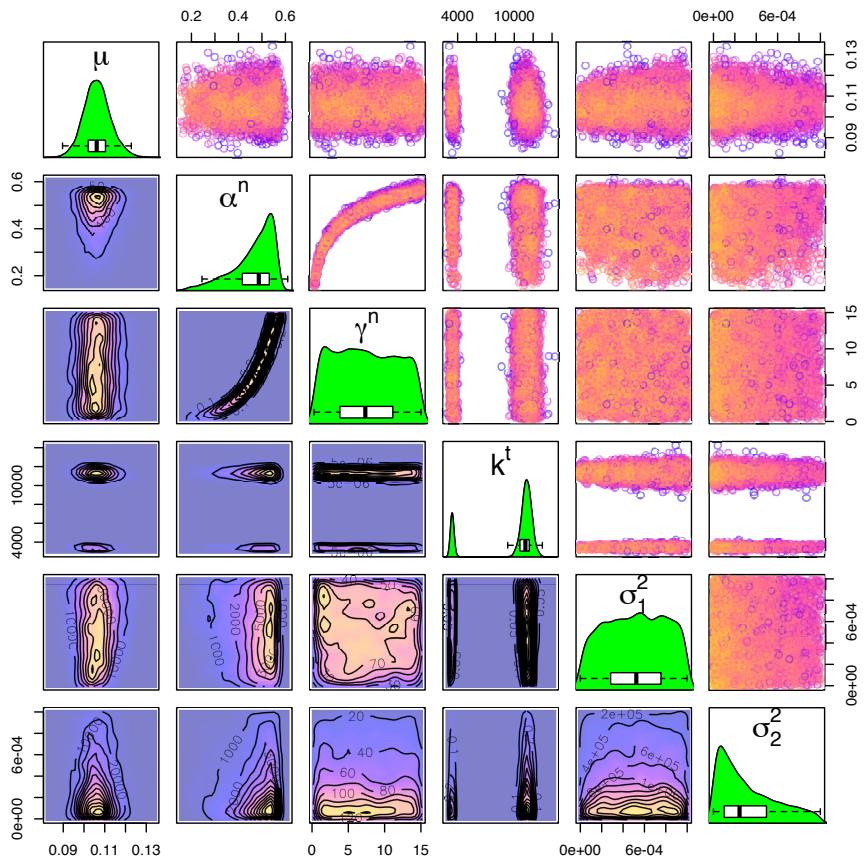

Fig. 21. Posterior samples for one case of the disk-wall collision example. Upper diagonal: projection of the posterior samples in all pairs of 2D parameter space (colors indicate log-likelihood values of the samples). Diagonal: marginal distributions of the model parameters estimated using kernel histograms. Box-plots denotes the mean and 95% percentiles. Lower diagonal: projected densities in 2D parameter space constructed via a kernel estimate (coloring according to log-posterior values).

5 Conclusions

We demonstrate that the original TMCMC algorithm proposed by [8] leads to bias in the estimates of the evidence and the posterior distribution, and show a practical solution depending on the properties of the actual applications. We show that this error emanates from the resampling step, which involves an MCMC step that is atypical to the classical SMC approaches. In the original resampling step, MCMC steps are performed with chain length equal to the number of times that a sample is resampled at a given intermediate stage. The variability in the chain lengths leads to bias depending on the proposal PDF. The problem is resolved by using a uniform chain length for all Markov chains or adding burn-in to all chains for better mixing behavior. However, this solution reduces the sampling efficiency for peaked posterior distributions. In terms of computational concern, using a uniform chain length is a better solution because it does not add extra computations, and also it improves the efficiency of parallelism due to an inherently better work balance. Hence, we improve the algorithm by adding two parameters: maximum chain length l_{max} and burn-in per chain n_{burn}. These extra parameters allow users to tune between TMCMC and the classical SMC approaches. We name this improved algorithm as Bayesian Annealed Sequential Importance Sampling (BASIS) to better reflect the characteristics of the method. The example of disk-wall collision simulation verifies our solution to the bias.

In general, we suggest using $l_{max} = 1$ and $n_{burn} = 0$ (the same resampling scheme used in [7, 11]). When the number of prior samples is not large enough to represent the important space well, e.g., high dimensional problems, $n_{burn} > 0$ is suggested. One drawback of setting $l_{max} = 1$ is that it reduces the explorative capability of the sampling algorithm. As a result, a large amount of samples is needed when the posterior distribution is very peaked. In this case, our results from the bridge example suggest using $l_{max} = \infty$ and $n_{burn} > 0$. In practice, this is often the case when the posterior distribution is expected to be peaked with respect to the prior domain. We note that one should not sacrifice accuracy of the distribution in each stage for exploration power. When computational power is allowed, the most accurate implementation is to use $l_{max} = 1$ with $n_{burn} > 0$. This method leads to extra computational effort and the choice of the burn-in period is a difficult problem

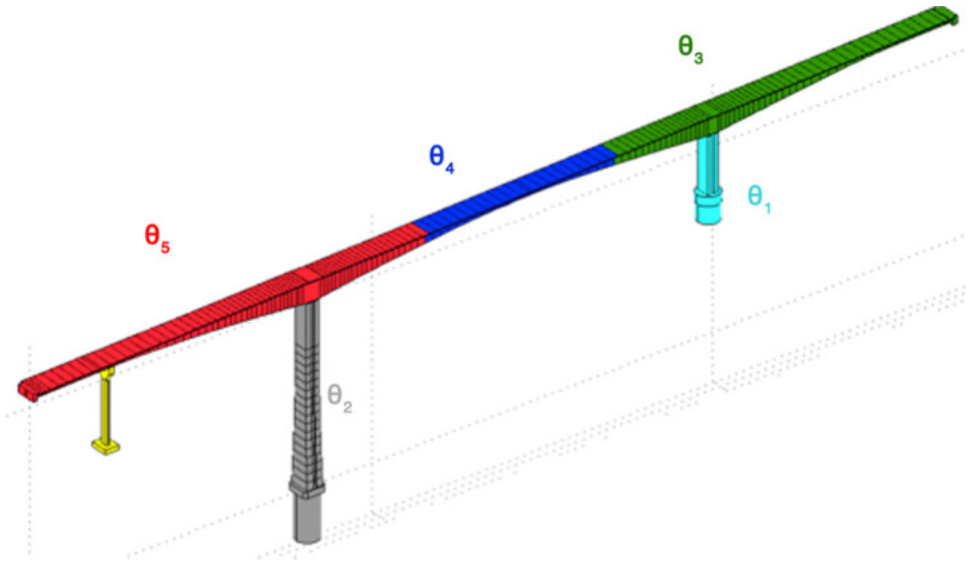

Fig. 22. Subdivided structural components of the Metsovo bridge [18].

Table 5. Estimates of ln(Evidence), mean (E) and coefficient of variation (CV) of each parameter for the bridge example with $l_{max} = 1$ and $l_{max} = \infty$. Numbers in parenthesis are coefficient of variation for each estimate calculated from 20 trials. All runs take 12 – 17 stages to finish. Bolded values are the reference to compare the other results with.

	$l_{max} = 1$			$l_{max} = \infty$		
	No burn-in	20-step burn-in [First 4 stages]	20-step burn-in [All stages]	No burn-in	20-step burn-in [First 4 stages]	20-step burn-in [All stages]
E_{θ_1}	1.006 (10.46%)	1.015 (0.85%)	**1.014 (0.06%)**	1.015 (0.38%)	1.015 (0.42%)	1.014 (0.09%)
CV_{θ_1}	1.67% (56.20%)	0.91% (19.04%)	**1.07% (2.27%)**	1.04% (11.48%)	1.09% (11.69%)	1.06% (3.53%)
E_{θ_2}	1.124 (11.83%)	1.021 (0.78%)	**1.019 (0.03%)**	1.019 (0.42%)	1.019 (0.28%)	1.019 (0.06%)
CV_{θ_2}	1.45% (46.87%)	0.92% (22.67%)	**0.97% (2.34%)**	0.95% (11.80%)	1.01% (10.13%)	1.01% (3.10%)
E_{θ_3}	1.020 (4.02%)	1.011 (0.42%)	**1.010 (0.02%)**	1.010 (0.20%)	1.011 (0.15%)	1.010 (0.05%)
CV_{θ_3}	0.84% (58.93%)	0.51% (19.41%)	**0.53% (2.57%)**	0.52% (11.93%)	0.54% (10.51%)	0.53% (3.51%)
E_{θ_4}	1.043 (8.73%)	1.002 (0.69%)	**1.001 (0.03%)**	1.001 (0.29%)	1.002 (0.32%)	1.001 (0.07%)
CV_{θ_4}	1.35% (43.98%)	0.78% (19.68%)	**0.89% (2.55%)**	0.85% (13.81%)	0.89% (13.11%)	0.88% (3.74%)
E_{θ_5}	1.000 (4.95%)	1.010 (0.36%)	**1.009 (0.02%)**	1.009 (0.15%)	1.009 (0.13%)	1.009 (0.03%)
CV_{θ_5}	0.84% (54.26%)	0.43% (25.99%)	**0.44% (2.89%)**	0.45% (10.16%)	0.47% (10.13%)	0.44% (4.46%)
E_{σ}	0.024 (34.46%)	0.010 (3.99%)	**0.010 (0.20%)**	0.010 (1.74%)	0.010 (1.20%)	0.010 (0.38%)
CV_{σ}	3.39% (39.08%)	4.83% (17.27%)	**5.07% (3.07%)**	5.17% (8.74%)	5.16% (10.73%)	5.10% (3.61%)
ln(Evid.)	656.726 (9.45%)	796.451 (0.48%)	**799.219 (0.09%)**	792.387 (0.74%)	796.099 (0.31%)	798.182 (0.13%)

in itself. To improve computational efficiency, we show in our examples that adding burn-in period to only the early stages already results in significant improvement for the accuracy of the estimations. Our final implementation of the algorithm also includes parameters to control the number of stages to add burn-in. In conclusion, BASIS fuses the resampling schemes in the original TMCMC and the classical SMC approaches using the l_{max} parameter, and we demonstrate a rule of thumb to choose l_{max} depending on the property of the practical applications.

Ongoing work includes studying the effect of using a proposal PDF, instead of the prior, for the initial samples, as well as using better MCMC proposals, such as including gradient and Hessian information. A Matlab version of the BASIS with parallelization support can be downloaded from http://www.cse-lab.ethz.ch/software/BASIS .

Acknowledgements

This work utilized computational resources granted by the Swiss National Super- computing Centre (CSCS) under project ID s436. We gratefully acknowledge support from the European Research Council (ERC) Advanced Investigator Award (No. 2-73985-14), ETH-Zurich and computational time at the HPC Cluster Brutus. We would also like to thank Dr. Diego Rossinelli, Dr. Panagiotis Hadjidoukas and Lina Kulakova in the CSELab at ETH-Zurich for providing information and related codes to the case study.

References

[1] Gordon, N., Salmond, D., and Smith, A., 1993. "Novel approach to nonlinear/non-gaussian bayesian state estimation". *IEE Proceedings F on Radar and Signal Processing,* **140**(2), pp. 107–113.

[2] Kitagawa, G., 1996. "Monte carlo filter and smoother for non-gaussian nonlinear state space models". *Journal of Computational and Graphical Statistics,* **5**(1), pp. 1–25.

[3] Del Moral, P., 1996. "Non linear filtering: Interacting particle solution". *Markov Processes and Related Fields,* **2**(4), pp. 555–580.

[4] Liu, J., and Chen, R., 1998. "Sequential monte carlo methods for dynamic systems". *Journal of the American Statistical Association,* **93**(443), pp. 1032–1044.

[5] Neal, R., 1998. Annealed importance sampling. Tech. Rep. 9805, Department of Statistics, University of Toronto, Toronto, Ontario, Canada.

[6] Beck, J., and Au, S., 2002. "Bayesian updating of structural models and reliability using markov chain monte carlo simulation". *Journal of Engineering Mechanics-ASCE,* **128**(4), April, pp. 380–391.

[7] Chopin, N., 2002. "A sequential particle filter method for static models". *Biometrika,* **89**(3), pp. 539–551.

[8] Ching, J., and Chen, Y., 2007. "Transitional markov chain monte carlo method for bayesian model updating, model class selection, and model averaging". *Journal of Engineering Mechanics-ASCE,* **133**(7), July, pp. 816–832.

[9] Ching, J., and Wang, J., 2016. "Application of the transitional markov chain monte carlo algorithm to probabilistic site characterization". *Engineering Geology,* **203**(SI), pp. 151–167.

[10] Kirkpatrick, S., Gelatt, C., and Vecchi, M., 1983. "Optimization by simulated annealing". *Science,* **220**, pp. 671–680.

[11] Del Moral, P., Doucet, A., and Jasra, A., 2006. "Sequential monte carlo samplers". *J. R. Statist. Soc. B,* **68**(3), pp. 411–436.

[12] Angelikopoulos, P., Papadimitriou, C., and Koumoutsakos, P., 2012. "Bayesian uncertainty quantification and propagation in molecular dynamics simulations: A high performance computing framework". *Journal of Chemical Physics,* **137**, p. 144103.

[13] Zhang, Y., and Yang, W., 2014. "A comparative study of the stochastic simulation methods applied in structural health monitoring". *Engineering Computations,* **31**(7), pp. 1484–1513.

[14] Ortiz, G., Alvarez, D., and Bedoya-Ruiz, D., 2015. "Identification of bouc-wen type models using the transitional markov chain monte carlo method". *Computers and Structures,* **146**, pp. 252–269.

[15] He, S., and Ng, C., 2017. "Guided wave-based identification of multiple cracks in beams using a bayesian approach". *Mechanical Systems And Signal Processing,* **84**(A), pp. 324–345.

[16] Beck, J., 2010. "Bayesian system identification based on probability logic". *Structural Control and Health Monitoring,* **17**(7), pp. 825–847.

[17] Angelikopoulos, P., Papadimitriou, C., and Koumoutsakos, P., 2012. "Bayesian uncertainty quantification and propagation in molecular dynamics simulations: A high performance computing framework". *J. Chem. Phys.,* **137**(14), October.

[18] Hadjidoukas, P., Angelikopoulos, P., Papadimitriou, C., and Koumoutsakos, P., 2015. "Π4U: A high performance computing framework for bayesian uncertainty quantification of complex models". *Journal of Computational Physics,* **284**, pp. 1–21.

[19] Glynn, P. W., and Heidelberger, P., 1991. "Analysis of initial transient deletion for replicated steady-state simulations". *Oper. Res. Lett.,* **10**(8), November, pp. 437–443.

[20] Minson, S., Simons, M., and Beck, J., 2013. "Bayesian inversion for finite fault earthquake source models i-theory and algorithm". *Geophysical Journal International,* **194**(3), pp. 1701–1726.

[21] Betz, W., Papaioannou, I., and Straub, D., 2016. "Transitional markov chain monte carlo: Observations and improvements". *Journal of Engineering Mechanics-ASCE,* **04016016**.

[22] Rosenthal, J., 2000. "Parallel computing and monte carlo algorithms". *Far East Journal of Theoretical Statistics,* **4**(207-236).

[23] Hadjidoukas, P. E., Angelikopoulos, P., Rossinelli, D., Alexeev, D., Papadimitriou, C., and Koumoutsakos, P., 2014. "Bayesian uncertainty quantification and propagation for discrete element simulations of granular materials". *Comput. Meth. Appl. Mech. Eng.,* **282**, December, pp. 218–238.

[24] Papadimitriou, C., and Papadioti, D.-C., 2013. "Component mode synthesis techniques for finite element model updating". *Comput. Struct.,* **126**, September, pp. 15–28.

[25] Kruggel-Emden, H., Wirtz, S., and Scherer, V., 2008. "A study on tangential force laws applicable to the discrete element method (dem) for materials with viscoelastic or plastic behavior". *Chem. Eng. Sci.,* **63**(6), March, pp. 1523–1541.

[26] Tsuji, Y., Tanaka, T., and Ishida, T., 1992. "Lagrangian numerical-simulation of plug flow if cohesionless particles in a horizontal pipe". *Powder Technol.,* **71**(3), September, pp. 239–250.