Efficient techniques for Bayesian inverse modeling of large-order computational models

C. Papadimitriou
University of Thessaly, Volos, Greece

P. Angelikopoulos, P. Koumoutsakos
ETH-Zurich, Switzerland

D.-C. Papadioti
University of Thessaly, Volos, Greece

ABSTRACT: Bayesian tools for inverse modeling are based on asymptotic approximations and stochastic simulation algorithms (SSA). Such tools require a number of moderate to large number of system re-analyses. For large-order numerical models of engineering systems, the computational requirements in Bayesian tools can be excessive. Using the Transitional MCMC algorithm, this study proposes efficient techniques for reducing the computational demands to manageable levels. Adaptive surrogate models are used to reduce the number of full system runs by an order of magnitude and parallel computing algorithms are employed to efficiently distribute the Transitional MCMC computations in multi-core CPUs. Applications in structural dynamics are emphasized in this work. Recently developed fast and accurate component mode synthesis techniques, consistent with the finite element parameterization, are implemented to achieve drastic reductions in the system order, resulting in additional substantial computational savings. An example of a bridge model with hundred of thousand of degrees of freedom demonstrates the capabilities of the proposed framework and the remarkable computational savings that can be achieved.

1 INTRODUCTION

Bayesian inference is used for quantifying uncertainty and calibrating models of engineering systems based on measurements. It is also used for propagating the modeling uncertainties in simulations of the system behavior for updated robust predictions of system performance, reliability and safety (Papadimitriou et al. 2001). The Bayesian tools consist of Laplace methods of asymptotic approximation and more accurate stochastic simulation algorithms such as Markov Chain Monte Carlo (MCMC) (Metropolis et al. 1953), Transitional MCMC (Ching and Chen 2007) and Delayed Rejection Adaptive Metropolis (DRAM) (Haario et al. 2006).

For large number of measurements, the Bayesian central limit theorem is used to approximate the posterior distribution of the model parameters by a Gaussian distribution centered at the most probable value of the model parameters with covariance equal to the inverse of the Hessian of minus the logarithm of the posterior distribution evaluated at the most probable value. This approximation involves solving an optimization problem as well as computing the Hessian of a function that depends on the quantities of interest evaluated through model simulations. Gradient-based optimization algorithms can be used with first-order adjoint methods to efficiently compute the most probable value. Second-order adjoint methods can then be used to compute the Hessian involved in the Gaussian distribution. An example of the application of first and second-order adjoint in structural dynamics can be found in the work by Ntotsios and Papadimitriou (2008). However, problems may arise when the gradient-based algorithms converge to local optima. Also, for some models of physical systems it is not possible to formulate an adjoint problem. In such cases stochastic methods, such as CMA (Hansen et al. 2003) can be used to estimate the global optima in the expense of more computational effort due to significantly more system re-analyses required. Once the optimum has been determined, the Hessian can be obtained either numerically or by using efficient methods (e.g. Lyness and Moler 1969).

The asymptotic estimate in Bayesian inverse modeling is approximate. Moreover, even for large number of data, it may fail to give a good representation of the posterior probability distribution in the case of multimodal distributions. In addition, the asymptotic approximation fails to provide acceptable estimates for un-identifiable cases manifested for relatively large number of model parameters in relation to the information contained in the data.

For more accurate estimates, one should use stochastic simulation algorithms (SSA) to generate samples that populate the posterior distribution and then evaluate robust prediction integrals using sam-

ple estimates. Among the stochastic simulation algorithms available, the transitional MCMC algorithm (Ching and Chen 2007) is one of the most promising algorithms for selecting the most probable model as well as finding and populating with samples the importance region of interest of the posterior probability distribution, even in the unidentifiable cases and multi-modal posterior distributions. In addition, the TMCMC method yields an estimate of the evidence of the model class based on the samples generated by the algorithm.

SSA tools involve generating samples for tracing and then populating the important uncertainty region in the parameter space, as well as evaluating integrals over high-dimensional spaces of the uncertain model parameters. They usually require a very large number of system re-analyses to be performed over the space of uncertain parameters. Consequently, the computational demands depend highly on the number of system analyses and the time required for performing a system analysis. For models involving hundreds of thousands or even million degrees of freedom and localized nonlinear actions activated during system operation, the computational demands may be excessive. The present work proposes methods for drastically reducing the computational demands at the system, algorithm and hardware levels involved in the implementation of Bayesian tools.

Surrogate models and high performance computing techniques are integrated (Angelikopoulos et al. 2012) in Bayesian inverse techniques to efficiently handle the excessive computational cost associated with large number of re-analyses of large-order, industrial size, computational models of hundreds of thousands or millions degrees of freedom encountered in practical applications. Parallel computing algorithms can be used to efficiently distribute the computations in available multi-core CPUs in clusters with heterogeneous architectures.

Application of the proposed computational framework to model calibration and robust response updating in structural dynamics is emphasized. Efficient fast and accurate component mode synthesis (CMS) techniques, consistent with the finite element (FE) model parameterization, are also implemented to achieve drastic reductions in the system order, resulting in additional substantial computational savings (Papadimitriou and Papadioti 2013). Large-order linear computational models taken from civil engineering applications demonstrate that remarkable reductions in computational effort can be achieved, allowing the excessive computations in Bayesian inverse techniques for large-order computational models to be reduced to manageable levels.

2 BAYESIAN INVERSE MODELING

2.1 *Parameter estimation*

Consider a parameterized model class M_m of an actual engineering system used to predict various output quantities of interest $g(\underline{\theta}_m \,|\, \mathsf{M}_m)$ of the system, where $\underline{\theta}_m \in R^{N_m}$ is a set of parameters in this model class that need to be estimated using experimental data $D \equiv \{\hat{y}_r, r = 1, \cdots, N_d\}$.

Following a Bayesian formulation (Beck and Katafygiotis 1998, Beck 2010, Yuen 2010) and assuming that the observation data and the model predictions satisfy the prediction error equation

$$\underline{\hat{y}} = \underline{g}(\underline{\theta}_m \,|\, \mathsf{M}_m) + \underline{e} \tag{1}$$

where the error term $\underline{e} \sim N(\underline{0}, \Sigma)$ is a zero-mean Gaussian vector with covariance $\Sigma = \Sigma(\underline{\theta}_e)$ depending on the parameters $\underline{\theta}_e$ of the prediction error model class M_e, the updated distribution $p(\underline{\theta} \,|\, D, \mathsf{M})$ of the augmented parameter set $\underline{\theta} = (\underline{\theta}_m, \underline{\theta}_e)$, given the data D and the combined model class $\mathsf{M} = \{\mathsf{M}_m, \mathsf{M}_e\}$, results from the application of the Bayes theorem as follows

$$p(\underline{\theta} \,|\, D, \mathsf{M}) = p(D \,|\, \underline{\theta}, \mathsf{M})\ \pi(\underline{\theta} \,|\, \mathsf{M}) \,/\, p(D \,|\, \mathsf{M}) \tag{2}$$

where

$$p(D \,|\, \underline{\theta}, \mathsf{M}) = \frac{|\Sigma(\underline{\theta}_e)|^{-1/2}}{(2\pi)^{m/2}} \exp\left[-\frac{1}{2} J(\underline{\theta}; \mathsf{M})\right] \tag{3}$$

is the likelihood of observing the data from the model class,

$$J(\underline{\theta}; \mathsf{M}) = [\underline{\hat{y}} - \underline{g}(\underline{\theta}_m \,|\, \mathsf{M})]^T \Sigma^{-1}(\underline{\theta}_e)[\underline{\hat{y}} - \underline{g}(\underline{\theta}_m \,|\, \mathsf{M})] \tag{4}$$

is the measure of fit between the experimental and model predicted properties, $\pi(\underline{\theta} \,|\, \mathsf{M})$ is the prior probability distribution of the model parameters based on previous knowledge and/or user experience, and $p(D \,|\, \mathsf{M})$ is the evidence of the model class.

2.2 *Robust predictions*

Let q be an output quantity of interest for the system. Posterior robust predictions of q are obtained by taking into account the updated uncertainties in the model parameters given the measurements D. Let $p(q \,|\, \underline{\theta}, \mathsf{M})$ be the conditional probability distribution of q given the values of the parameters. Using the total probability theorem, the posterior robust probability distribution $p(q \,|\, D, \mathsf{M})$ of q, taking into account the model M and the data D, is given by (Papadimitriou et al. 2001)

$$p(q \,|\, D, \mathsf{M}) = \int p(q \,|\, \underline{\theta}, \mathsf{M})\ p(\underline{\theta} \,|\, D, \mathsf{M})\ d\underline{\theta} \tag{5}$$

as an average of the conditional probability distribution $p(q \,|\, \underline{\theta}, \mathsf{M})$ weighting by the posterior probabil-

ity distribution $p(\underline{\theta} \mid D, \mathsf{M})$ of the model parameters. Let $G(q)$ be a function of the output quantity of interest q. A posterior robust performance measure of the system given the data D is

$$E[G(q) \mid D, \mathsf{M})] = \int G(q)\, p(\underline{\theta} \mid D, \mathsf{M})\, d\underline{\theta} \qquad (6)$$

Stochastic simulation methods can be conveniently used to estimate the integral from the samples $\underline{\theta}^{(i)}$, $i = 1, \cdots, N$, generated from the posterior probability distribution $p(\underline{\theta} \mid D, \mathsf{M})$ in (2). In this case, the integrals (5) and (6) can be approximated by

$$p(q \mid D, \mathsf{M}) \approx \frac{1}{N} \sum_{i=1}^{N} p(q \mid \theta^{(i)}, \mathsf{M}) \qquad (7)$$

and

$$E[G(q) \mid D, \mathsf{M})] \approx \frac{1}{N} \sum_{i=1}^{N} G(q^{(i)}) \qquad (8)$$

respectively.

3 STOCHASTIC SIMULATION ALGORITHMS

Markov Chain Monte Carlo (MCMC) algorithms are used to efficiently draw samples from the posterior distribution. MCMC variants such as Differential Evolution MC (Braak et al. 2006) or Differential Evolution Random Subsampling MC (DREAM) (Braak et al. 2008) were introduced to improve parallel efficiency. These methods consist of a population of chains that interact by exchanging information but at the same time preserve the MCMC convergence characteristics at the individual chain level. Another MCMC method which can be categorized in the framework of Evolutionary Strategy MCMC methods (Drugan and Thierens 2010) is the TMCMC (Ching and Chen 2007). This method is a generalization of the method proposed by Au and Beck (2002) extended by notions inherent to simulated annealing algorithms.

The TMCMC method has been proposed to address the problem of choosing the right adaptive proposal PDF in MCMC methods for accelerating convergence to the posterior PDF. This can be a serious problem when the support of the posterior PDF in the parameter space has complex geometry, and/or when the posterior PDF is very peaked and isolated in a small region in the parameter space. Due to a large number of independent parallel chains involved, TMCMC is more efficient in terms of parallel efficiency compared to the DRAM and DREAM algorithms. DRAM is essentially serial and DREAM based on Differential Evolution suggests the use of a few independent parallel chains (Braak et al. 2008) proportional to the parameter space dimensionality. TMCMC also applies to multimodal posterior PDFs as it handles efficiently very peaked or very flat PDFs along certain directions in the parameter space, as well as it estimates the evidence $p(D \mid \mathsf{M})$ which can further be used for model selection.

4 SURROGATE MODELS BASED ON KRIGING

The most time consuming part of the TMCMC algorithm is the repeated evaluation of the likelihood function, requiring a large number of full model reanalyses. Surrogate models are used to reduce the computational time at the level of the TMCMC algorithm. The objective is to avoid the expensive full model simulation runs at a new sampling point in the parameters space by exploiting the function evaluations from previous full model runs that are available at the neighbour (design) points in order to generate an approximate estimate.

Surrogate models are especially well-suited for use with the TMCMC algorithm since at each intermediate stage in the TMCMC algorithm, a large number of samples that sufficiently cover the supports of the intermediate posterior PDFs from the current and previous stages are available to be used as design points for approximating the likelihood estimate at a new sample based on a surrogate technique. An adaptive surrogate technique can thus be used to exploit the information from available neighbour samples for providing an approximate estimate at a new sample.

4.1 *Kriging interpolation*

The kriging technique Lophaven et al. (2002) is used to approximate the function evaluation at a sampling point using the function evaluations at neighbour points in the parameter space. Consider m design/sample points $[\underline{\theta}_1, \ldots, \underline{\theta}_m]$ in the parameter space and let $\underline{Y} = [J(\underline{\theta}_1), \ldots, J(\underline{\theta}_m)]^T$ be the available values of a response function $J(\underline{\theta})$ at these points. Using the kriging method, a function $J(\underline{\theta})$ is approximated at a point θ in the parameter space in the form

$$J(\underline{\theta}) = \underline{f}^T(\underline{\theta})\underline{\beta} + \underline{\varepsilon}(\underline{\theta}) \qquad (9)$$

where $\underline{f}^T(\underline{\theta})\underline{\beta}$ is the mean response, $\underline{f}^T(\underline{\theta}) = [f(\underline{\theta}_1), \ldots, f(\underline{\theta}_m)]$ are user selected basis functions, usually in polynomial form, $\underline{\beta} = [\beta_1, \ldots, \beta_m]^T$ are regression coefficients to be estimated from the generalized least square method, and $\underline{\varepsilon}(\underline{\theta})$ is a zero mean stochastic process with covariance $E[\underline{\varepsilon}(\underline{\theta}_i)\underline{\varepsilon}(\underline{\theta}_j)] = \sigma^2 R(\underline{\varphi}; \underline{\theta}_i, \underline{\theta}_j)$ which depends on the variance σ^2 and a set of parameters $\underline{\varphi}$ appearing in the structure of the correlation function $R(\underline{\varphi}; \underline{\theta}_i, \underline{\theta}_j)$ of the stochastic process $\underline{\varepsilon}(\underline{\theta})$. A common choice of the correlation function is the exponential form:

$$R(\underline{\varphi};\underline{\theta}_i,\underline{\theta}_j)=\exp\left[-\sum_{k=1}^{m}\varphi_k\,|\underline{\theta}_i-\underline{\theta}_j\,|^{\varphi_{m+1}}\right] \quad (10)$$

with $\varphi_j \geq 0$, $j=1,\ldots,m$, and $0\leq\varphi_{m+1}\leq 2$. The optimal choice of the parameters $\underline{\beta}$, σ^2 and $\underline{\varphi}$ are the ones that maximize the likelihood function given the values at the design points. This likelihood function is given in the form

$$L=\frac{1}{\sqrt{2\pi}\sigma^m\,|R|^{1/2}}\exp\left[-\frac{1}{2\sigma^2}(\underline{Y}-F\underline{\beta})^T R^{-1}(\underline{Y}-F\underline{\beta})\right] \quad (11)$$

The optimal values of $\underline{\beta}$ and σ^2 are given by

$$\hat{\underline{\beta}}=(F^T R^{-1}F)^{-1}F^T R^{-1}\underline{Y} \quad (12)$$

$$\hat{\sigma}^2=\frac{1}{m}(\underline{Y}-F\underline{\beta})^T R^{-1}(\underline{Y}-F\underline{\beta}) \quad (13)$$

while the optimal values $\hat{\underline{\varphi}}$ of $\underline{\varphi}$ are obtained by minimizing the minus the log-likelihood function

$$\begin{aligned}G(\underline{\varphi})&=-\log L(\underline{\varphi},\hat{\underline{\beta}}(\underline{\varphi}),\hat{\sigma}^2(\underline{\varphi}))\\&=\frac{1}{2}\ln|R(\underline{\varphi})|+\frac{m}{2}\ln[(\underline{Y}-F\underline{\beta})^T R^{-1}(\underline{\varphi})(\underline{Y}-F\underline{\beta})]\end{aligned} \quad (14)$$

The prediction at the point $\underline{\theta}$ in the model parameter space and its mean square error are given respectively by [28, 29]

$$J(\underline{\theta})=\underline{f}^T(\underline{\theta})\hat{\underline{\beta}}+\underline{r}^T\hat{R}^{-1}(\underline{Y}-F\hat{\underline{\beta}}) \quad (15)$$

and

$$s^2(\underline{\theta})=\hat{\sigma}^2[1-\underline{r}^T\hat{R}^{-1}\underline{r}+\underline{u}^T(F^T\hat{R}^{-1}F)^{-1}\underline{u}] \quad (16)$$

where $\hat{R}=R(\hat{\underline{\varphi}})$, $\underline{u}=F^T R^{-1}\underline{r}-\underline{f}(\underline{\theta})$, F is a matrix with components $F_{ij}=f_j(\underline{\theta}_i)$, R is the matrix with components $R_{ij}=R(\underline{\varphi};\underline{\theta}_i,\underline{\theta}_j)$.

The estimation of the kriging parameters $\hat{\underline{\varphi}}$ involve solving a potentially high dimensionality minimization problem, occurring due to the choice of anisotropic correlation lengths assignment. Following the analysis of Lophaven et al. (2002), the kriging prediction can quickly deteriorate if the minimization is not accurate. A bounded pattern-search method is used in this work. Statistically, the root mean squared error (RMSE) or the standard deviation $s(\underline{\theta})$ represents the predicted deviation of the kriging metamodel from the actual response. This standard deviation $s(\underline{\theta})$ will be used in the paper to accept or reject a surrogate prediction.

4.2 *Adaptive kriging and TMCMC*

In general, the accuracy of the surrogate estimates based on a number of support points depend on the smoothness of the function to be evaluated on the region in the parameter space that is covered by the support points. The smaller the size of this region, the higher the expected accuracy from the surrogate estimates. It is thus expected that higher accuracy of the surrogate estimate at a point in the parameter space will be achieved by using support points at the neighbor of the surrogate point instead of global support points that cover the whole region or significant parts of the region/domain in the parameter space.

A feature of the TMCMC algorithm is that the MC samples generated from the multiple chains cover the whole support of the posterior distribution of the TMCMC stage j. As a result, a new MC sample point in the parameter space generated from the TMCMC algorithm is close to neighbor points that could be used as support points to generate a surrogate estimate of the function instead of an expensive real estimate. Each new MC point is then associated with a different set of neighbor points, depending of its location in the parameter space. This surrogate estimate based on different support points located at the neighbor of the current surrogate point is an adaptive surrogate procedure.

To maintain the accuracy of the TMCMC algorithm, the error $s(\underline{\theta})$ of the function evaluation due to surrogate estimate has to be kept relatively small. If this is not possible, it is then more suitable to use a full function evaluation at the surrogate point.

In order to ensure a high quality approximation, a surrogate estimate at stage j is performed and accepted if it simultaneously obeys the following heuristic rules:

1. The design points used for interpolation correspond to real full system simulations and not other surrogate estimates.
2. The surrogate point belongs to the convex hull of the design points so that an interpolation is performed, while extrapolations are prohibited.
3. The surrogate estimate is based on a user-defined minimum number of design points, which are in the neighbor of the surrogate point. The minimum number of design points depends on the dimension of the uncertain parameter space and the order of the kriging interpolation. To avoid overfitting and discontinuities that arise from the fact that the surrogate estimates at two neighbour points might be based on two different sets of design points, the number of design points is selected to be larger than the minimum number of points required to perform the surrogate estimate.
4. The neighbour design points are selected as the ones closest to the surrogate estimate and also within the hyper-ellipse of the TMCMC proposal covariance matrix scaled to include the minimum number of design points.

5. The kriging approximation maintains its locality, by choosing the design points as such belonging to a hyper-surface defined by the scaled proposal covariance matrix. The scaling factor scales the neighborhood around the surrogate point up to a predefined scaling number. A surrogate estimate is not allowed if the scaling factor exceeds a prespecified number so that only local estimates are accepted.
6. The surrogate estimate is checked whether its predicted value is within a 95% quintile of all the design point likelihood values accounted so far. The purpose of the threshold is to discourage overshooting surrogate estimates to get the highest possible plausibility weights, as this will lead during the next stage following the re-sampling step in the generation of a long chain with artificially induced large rejection rates, quickly decaying sampling quality and potentially completely destroying the sampling procedure.
7. The surrogate estimate is accepted if the prediction error $s(\underline{\theta})$, given in (16) by the kriging method, is smaller than a user specified value. The effect of this value on the accuracy will be demonstrated in the numerical examples.

At this point it is very important to note the different possibilities one has in order to do the interpolation. One can interpolate at either the higher measure of fit level or at the lower model simulation level. At the measure of fit level, the interpolations are performed for the measure of fit function $J(\underline{\theta};\mathsf{M})$ in (4) or even the likelihood function level in (3). At the model simulation level, the interpolations are performed for the functions $\underline{g}(\underline{\theta}_m \,|\, \mathsf{M}_m)$, which may include various output quantities of interest.

5 TMCMC PARALLEL IMPLEMENTATION

The TMCMC algorithm is very-well suited for parallel implementation in a computer cluster. Details of the parallel implementation are given in Angelikopoulos et al. (2012). Specifically, a parallel implementation algorithm is activated at every stage of the TMCMC algorithm exploiting the large number of short, variable length, chains that need to be generated starting from the leader samples determined from the TMCMC algorithm at the particular stage. Static and dynamic scheduling schemes can be conveniently used to optimally distribute these chains in a multi-host configuration of complete heterogeneous computer workers. The static scheduling scheme distributes the chains in the workers using a weighted round-robin algorithm so that the number of likelihood evaluations is arranged to be the same for each computer worker.

The static scheduling scheme is computational efficient when the computational time for a likelihood evaluation is the same independently of the location of sample in the parameter space as well as when surrogate estimates are not activated. The dynamic scheduling scheme is more general, ensuring a more efficient balancing of the loads per computer worker in the case of variable run time of likelihood function evaluations and unknown number of surrogates activated during estimation. Specifically, each worker is periodically interrogated at regular time intervals by the master computer about its availability and samples from TMCMC chains are submitted to the workers on a first come first serve basis to perform the likelihood function evaluations so that the idle time of the multiple workers is minimized.

6 APPLICATION IN STRUCTURAL DYNAMICS

The computational efficiency and accuracy of the proposed method is demonstrated by applying it to a structural dynamics problem. Specifically, the method is used to update the uncertainties in the parameters of the FE model of the Metsovo bridge based on simulated modal data.

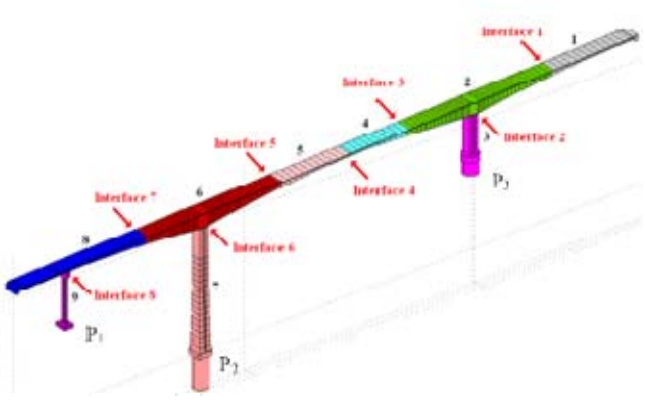

Figure 1. Components of FE model of bridge.

6.1 *Description of structure and model*

A description of the bridge can be found in Papadimitriou and Papadioti (2013). A detailed FE model of the bridge is created using 3-dimensional tetrahedron quadratic Lagrange FEs and is shown in Figure 1. An extra coarse mesh is chosen to predict the lowest 20 modal frequencies and mode shapes of the bridge. The size of the elements in the extra coarse mesh is the maximum possible one that can be considered, with typical element length of the order of the thickness of the deck cross-section. This model has 97,636 FEs and 562,101 DOFs.

6.2 *FE model reduction using CMS*

The CMS method for model updating presented in the work by Papadimitriou and Papadioti (2013) is used to significantly reduce the size of the FE model such that the predictions of the required lowest 20 modes are accurate. CMS techniques (Craig and Bampton 1965) divide the structure into components with mass and stiffness matrices that are reduced, using fixed-interface and constrained modes, to alleviate part of the computational effort. However, direct application of CMS technique at each TMCMC sampling point requires the re-computation of the eigen-problem and the interface constrained modes for each component. This is a very time consuming operation and computationally more expensive that solving directly the original matrices for the eigenvalues and the eigenvectors. For certain parameterization schemes for which the mass and stiffness matrices of a component depend linearly on only one of the free model parameters to be updated, the full reanalyses of the component eigen-problems are avoided. The eigenproperties and the interface constrained modes can be computed as a function of the model parameters directly from the eigenproperties and the interface constrained modes that correspond to a nominal value of the model parameters. Details of the formulation are presented in (Papadimitriou and Papadioti 2013). The end result is that the reduced mass and stiffness matrices of the structure for each re-analyses are obtained from the reduced component mass and stiffness matrices obtained for a nominal structure. This is an important result which saves substantial computational effort since it avoids (a) re-computing the fixed-interface and constrained modes for each component, and (b) assembling the reduced matrices from these components. The formulation guarantees that the reduced system is based on the exact component modes for all values of the model parameters.

6.3 *Application and numerical results*

Let ω_c be the highest modal frequency that is of interest in FE model updating. In this study the cut-off frequency is selected to be the 20^{th} modal frequency ($\omega_c = 4.55$ Hz) of the nominal FE model. Following CMS technique, the bridge is divided into nine components as shown in Figure 1. This division introduces 8 interfaces as shown in the same figure. The modes selected to be kept for each component have frequency $\omega_{\max} = \rho\omega_c$, where the ρ values affect the computational efficiency and accuracy of the CMS technique. The ρ value is chosen to be $\rho = 8$, resulting in a substantial reduction of model DOF by more than two orders of magnitude. Specifically, the total number of DOF is 3,586 (286 internal modes and 3,300 interface DOFs) for all components. The highest error in the estimation of the lowest twenty modal frequencies is less than 0.01%. A further reduction in the number of generalized coordinates is achieved by retaining only a fraction of the constrained interface modes with frequency less than $\omega_{\max} = \nu\omega_c$. Using $\nu = 200$, an order of magnitude reduction in the interface DOFs is obtained with the kept modes to be 592 and the errors in the estimates for the lowest 20 modal frequencies to be less than 0.02%. It this thus evident that using CMS a drastic reduction in the number of generalized coordinates, without sacrificing in accuracy, is obtained which can exceed three orders of magnitude in this case.

The computational time needed to estimate the lowest 20 modal properties using CMS with $\rho = 8$ and $\nu = 200$ is two orders of magnitude less than the time required to solve the complete FE model. It is thus obvious that CMS is expected to drastically reduce the computational effort in Bayesian inverse modeling without sacrificing in accuracy.

The FE model is parameterized using five parameters associated with the modulus of elasticity of one or more structural components shown in Figure 1. Specifically, the first two parameters θ_1 and θ_2 account respectively for the modulus of elasticity of the pier components 3 and 7 of the bridge. The parameter θ_3 accounts for the modulus of elasticity of the components 1 and 2 of the deck, the parameter θ_4 accounts for the components 4 and 5, while the parameter θ_5 accounts for the components 6 and 8. The component 9 is not parameterized. The model parameters are introduced to scale the nominal values of the properties that they model so that the value of the parameters equal to one corresponds to the nominal value of the FE model.

The estimation of the parameter values and their uncertainties of the FE model is based on modal frequencies and mode shapes. Simulated, noise contaminated, measured modal frequencies and mode shapes are generated by adding a 1% and 3% Gaussian noise to the modal frequencies and modeshape components, predicted by the nominal non-reduced FE models. 38 sensors are placed on the bridge to monitor vertical and transverse accelerations. The measured data contain the values of the ten lowest modal frequencies and modeshapes. Details of the likelihood function used and the form of the objective function (4) are given in Christodoulou and Papadimitriou (2007).

The model updating is performed using the stochastic simulation algorithm TMCMC with 1000 samples per TMCMC stage (Ching and Chen 2007). The number of FE model runs depends on the number of TMCMC stages which was estimated to be 19. The resulting number of FE model re-analyses are 19,000. The parallelization features of TMCMC (Angelikopoulos et al. 2012) were also exploited, taking advantage of the available four-core multi-

threaded computer unit to simultaneously run eight TMCMC samples in parallel. For comparison purposes, the computational effort for solving the eigenvalue problem of the original unreduced FE model is approximately 139 seconds. Multiplying this by the number of 19,000 TMCMC samples and considering parallel implementation in a four-core multi-threaded computer unit, the total computational effort for the model class is expected to be of the order 7 days. In contrast, for the reduced-order model for $\rho = 8$ and $\nu = 200$, the computational demands for running the model class are reduced to approximately 14 minutes. It is thus evident that a drastic reduction in computational effort for performing the structural identification based on a set of monitoring data is achieved from approximately 7 days for the unreduced model class to 14 minutes for the reduced model class, without compromising the predictive capabilities of the proposed parameter estimation methodology. This results in a factor of over 700 reduction in computational effort.

The effectiveness of the surrogate estimates for use with TMCMC algorithm is next demonstrated. The surrogate estimate is performed at the measure of fit level. The surrogate estimates are based on a second-order kriging approximation. In order to quantify the accuracy of the surrogate estimates, we introduce the following measure in the parameter space. Let q_i be a vector with components $q_{j,i}$ that measure the moment of the marginal distributions of the j-th model parameter in the set θ, where the subscript i refers to the i-th surrogate-based TMCMC run (s-TMCMC). Herein, the terms q represents the mean μ or the standard deviation σ of the marginal distributions of a model parameter θ. Due to the stochastic nature of the TMCMC algorithm, a sufficiently large number of s-TMCMC runs need to be performed to get a statistical meaningful estimate for the moment q computed from the s-TMCMC algorithm.

The confidence intervals of the estimated means and standard deviations of the marginal distributions of the model parameters using the s-TMCMC for various values of the errors $s(\underline{\theta})$ allowed for the kriging estimates are shown in Figure 2 and 3, respectively, and compared to the confidence intervals for the same quantities estimated using the TMCMC without surrogates. The results in this figure are computed for 50 TMCMC or 50 s-TMCMC independent runs. It is seen that the results provided by the surrogate-based s-TMCMC estimates match very well those given by the regular TMCMC estimates for even high values of the errors $s(\underline{\theta})$ allowed for the kriging approximation.

For large fractional errors of 10% and 50% corresponding to values in the figures of $s(\underline{\theta})$=0.1 and 0.5, respectively, the computational effort is deduced by 80 and 95%. This is approximately one order of magnitude reduction that can achieved by the integration of the surrogate estimates in TMCMC algorithm.

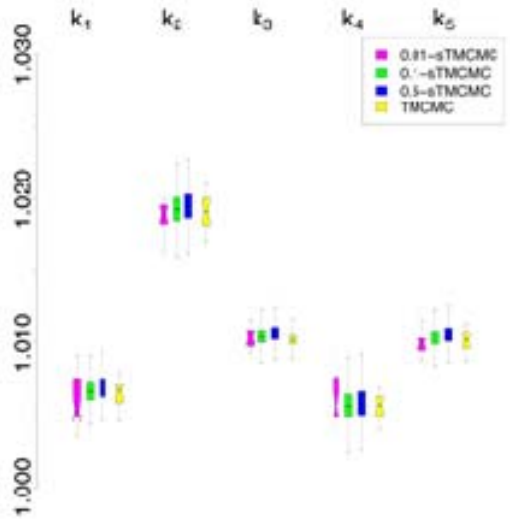

Figure 2 Comparison of the confidence intervals for the mean values as estimated from the TMCMC and s-TMCMC algorithms for errors 0.01, 0.1 and 0.5.

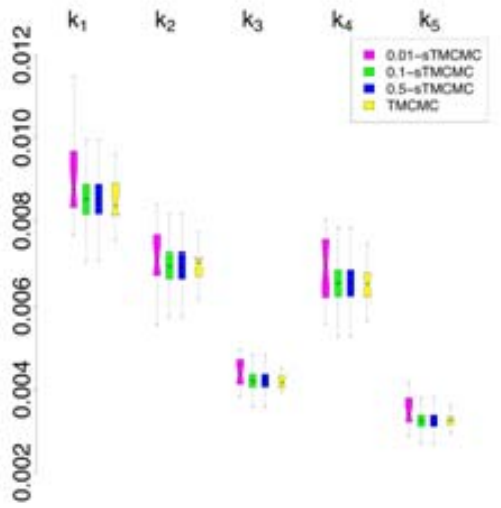

Figure 3. Comparison of the confidence intervals for the standard deviation values as estimated from the TMCMC and s-TMCMC algorithms for errors 0.01, 0.1 and 0.5.

Overall, using the model reduction, the surrogate approximations and parallel implementation in a 4-core multi-threaded computer, the 562,101 DOF finite element model, requiring 19,000 model runs, can be performed, depending on the allowed error in the surrogate estimate, in 1 to 3 minutes instead of 7 days, which constitutes a remarkable reduction of three to four orders of magnitude in computational effort.

7 CONCLUSIONS

Stochastic simulation algorithms, such as the TMCMC algorithm, used in Bayesian inverse modeling require a large number of FE model simulation runs. For large order computational models with hundred of thousands or even million DOFs, the computational demands involved in the TMCMC sampling algorithm may be excessive. Drastic reductions can be achieved using surrogate models and parallel implementation of the TMCMC algorithm. Surrogate models are well adapted to the TMCMC algorithm for significantly reducing the number of full model runs required. An adaptive kriging technique is effectively integrated within the parallel multiple chain TMCMC algorithm, resulting in substantial reduction of the number of full system reanalyses, essentially speeding-up computations by more than an order of magnitude. The proposed kriging technique exploits the availability of large number of multi-chain MCMC samples in the local neighbor of a surrogate estimate. Parallel computing algorithms are also very well suited to be used with TMCMC algorithm to efficiently distribute the computations in available multi-core CPUs.

Application of the framework to Bayesian inverse modeling in structural dynamics using vibration measurements was emphasized in this work. Recent developments in CMS techniques for parameter estimation, exploiting certain parameterization schemes often encountered in FE model updating, were shown to be effective in drastically reducing the order of the structural models and thus the computational effort required at the system level. Application of the framework on the model updating of a bridge demonstrated a remarkable reduction in computational time as high as four orders of magnitude.

ACKNOWLEDGEMENTS

This research has been co-financed by the European Union (European Social Fund - ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) - Research Funding Program: Aristeia.

REFERENCES

Angelikopoulos, P., Papadimitriou, C. & Koumoutsakos, P. 2012. Bayesian uncertainty quantification and propagation in molecular dynamics simulations: A high performance computing framework. *The Journal of Chemical Physics* 137(14) DOI: 10.1063/1.4757266.

Beck, J. L. & Au, S. K. 2002. Bayesian updating of structural models and reliability using Markov chain Monte Carlo simulation. *ASCE Journal of Engineering Mechanics* 128 (4): 380–391.

Beck, J.L. 2010. Bayesian system identification based on probability logic. *Structural Control and Health Monitoring* 17(7): 825–847.

Beck, J.L. & Katafygiotis, L.S. 1998. Updating models and their uncertainties. I: Bayesian statistical framework, *ASCE Journal of Engineering Mechanics* 124(4): 455-461.

Braak, C. 2006. A Markov Chain Monte Carlo version of the genetic algorithm Differential Evolution: easy Bayesian computing for real parameter spaces. *Stat. Comp.* 16: 239-249.

Braak, C.J. & Vrugt, J.A. 2008. Differential Evolution Markov Chain with snooker updater and fewer chains. *Stat. Comp.* 18: 435-446.

Ching J. & Chen Y.C. 2007. Transitional Markov Chain Monte Carlo method for Bayesian updating, model class selection, and model averaging. *ASCE Journal of Engineering Mechanics* 133: 816–832.

Craig Jr., R.R. & Bampton, M.C.C. 1965. Coupling of substructures for dynamic analysis. *AIAA J*. 6(7): 678–685.

Christodoulou, K. & Papadimitriou, C. 2007. Structural identification based on optimally weighted modal residuals. *Mechanical Systems and Signal Processing* 21: 4-23.

Drugan, M. & Thierens, D. 2010. Recombination operators and selection strategies for evolutionary Markov Chain Monte Carlo algorithms. *Evol. Intel.* 3: 79-101.

Haario, H. Laine, M., Mira, A. & Saksman, E. 2006. DRAM: Efficient adaptive MCMC. *Statistics and Computing* 16: 339-354.

Hansen, N., Mueller, S. D. & Koumoutsakos, P. 2003. Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES). *Evol. Comput.* 11(1): 1-18.

Lophaven, S.N., Nielsen, H.B. & Sondergaard, J. 2002. Dace, a matlab kriging toolbox. *Technical Report IMM-TR-2002-12*, DTU, DK-2800 Kgs Lyngby – Denmark.

Lyness, J.M. & Moler, C.B. 1969. Generalized Romberg methods for integrals of derivatives. *Numerische Mathematik*, 14(1): 1-13.

Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H. & Teller, E. 1953. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics* 21130(62): 1087-1092.

Ntotsios, E. & Papadimitriou, C. 2008. Multi-objective optimization algorithms for finite element model updating. *ISMA2008 International Conference on Noise and Vibration Engineering*, Leuven, 1895-1909.

Papadimitriou, C., Beck, J.L. & Katafygiotis, L.S. 2001. Updating robust reliability using structural test data. *Probabilistic Engineering Mechanics* 16(2): 103-113.

Papadimitriou, C. & Papadioti, D.C. 2013. Component Mode Synthesis Techniques for Finite Element Model Updating, *Computers and Structures*, DOI: 10.1016/j.compstruc.2012.10.018.

Yuen, K.V. 2010. *Bayesian Methods for Structural Dynamics and Civil Engineering*. John Wiley & Sons.