ELSEVIER

Available online at www.sciencedirect.com

ScienceDirect

Computers and Structures 85 (2007) 277–290

Computers & Structures

www.elsevier.com/locate/compstruc

On using deterministic FEA software to solve problems in stochastic structural mechanics

Sachin K. Sachdeva *, Prasanth B. Nair, Andy J. Keane

Computational Engineering and Design Group, School of Engineering Sciences, University of Southampton, Highfield, Southampton SO17 1BJ, UK

Received 21 October 2005; accepted 31 October 2006
Available online 21 December 2006

Abstract

Over the last three decades there has been an outstanding growth in the development of deterministic finite element codes with extensive analysis capabilities. Extension of such deterministic codes to solve problems in stochastic mechanics is of much interest to the academic research community and industry. In this paper we discuss some of the issues involved in integrating fully grown third-party deterministic finite element codes with stochastic projection schemes. The objective of this study is to lay the foundation for development of an easy-to-use general-purpose stochastic finite element software for carrying out probabilistic analysis of large-scale engineering systems. We present a brief introduction to stochastic reduced basis projection schemes and the steps involved in coupling them with a typical deterministic finite element software. We demonstrate with the help of a number of case studies how a coupled framework can be used for solving problems in probabilistic mechanics.

Keywords: Stochastic finite element analysis; Projection schemes; Deterministic FE software; Probabilistic mechanics; Reduced basis methods

1. Introduction

In recent years, there has been much interest in developing general-purpose probabilistic methods for reliability analysis and design of engineering systems in the presence of uncertainty. Stochastic finite element analysis (FEA) is becoming an increasingly important research field due to the increasing push towards high-fidelity analysis capability for uncertainty quantification and their application to robust design optimization. A key step in these approaches involves estimating the structural reliability which is usually defined in terms of the probability of failure (P_f) given by the following multi-fold integral:

$$P_{\mathrm{f}} = \int_{g(\boldsymbol{\theta})\leqslant 0} f(\boldsymbol{\theta})\,\mathrm{d}\boldsymbol{\theta}, \tag{1}$$

where $f(\boldsymbol{\theta})$ is the joint probability density function of the vector of random variables $\boldsymbol{\theta}$ and $g(\boldsymbol{\theta})$ defines the limit state function. To approximate P_{f} via Monte Carlo simulations or a response surface method, the multi-dimensional limit state function (which is not usually given in explicit form) needs to be evaluated repeatedly. Hence, a great deal of focus of the contemporary research in stochastic mechanics has been on harnessing deterministic FE codes to compute the limit state function at a single point.

Many such software systems are available, among which the most noted are NESSUS [1], COSSAN [2] and CalREL/FERUM/OpenSees [3–6]. ANSYS Inc. has incorporated probabilistic design capabilities in its recent releases, namely the ANSYS Probabilistic Design System and the ANSYS DesignExplorer [7]. For a detailed discussion on general-purpose software for structural reliability; see, for example, [8]. NESSUS is especially attractive since it already contains interfaces to many commercial FE codes and is easy to use because of a user friendly graphical user interface. The SSFEM module for FERUM developed by Sudret and Kiureghian [6] is a standalone code for

[*] Corresponding author. Address: 15 Wakami Crescent, Chellaston, Derby DE73 6XN, UK.
E-mail addresses: sks2006@alumni.soton.ac.uk, sachin.sachdeva@rolls-royce.com (S.K. Sachdeva).
URL: http://www.soton.ac.uk/~sachin (S.K. Sachdeva).

0045-7949/$ - see front matter © 2006 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compstruc.2006.10.008

structural response and reliability analysis based on polynomial chaos expansions. However, because of not being able to interact with more powerful third-party codes its capabilities are limited. In most of the software systems the underlying algorithm for computing the probability of failure hinges on approximate methods such as FORM/SORM, polynomial response surfaces or simulation methods. These algorithms are particularly attractive since they require minimal interaction with the third-party (FE) codes which in turn reduces implementation complexity. Another important contribution was made by Keese [9] and Yongke [10] who successfully coupled a stochastic finite element library (StoFEL) with ANSYS for approximating the structural response statistics and carrying out graphical post-processing.

The objective of this paper is to show how a deterministic FE software can be used to facilitate the stochastic analysis of engineering systems. The key idea is to leverage deterministic FE codes to carry out spatial discretization to arrive at a system of random algebraic equations. We use reduced basis projection schemes for extracting the solution process by solving this system of equations. In the semi-discretized form of the partial differential equations (PDEs) the solution process is approximated by a linear combination of unknown coefficients and stochastic basis vectors chosen from a preconditioned stochastic Krylov subspace of appropriate dimension. Subsequently, Galerkin projection schemes are used to extract the solution process. For a detailed discussion on the theoretical and implementation aspects of stochastic projection schemes; see, Refs., [11–15]. Post-processing and sampling the solution can provide a measure of the probability of failure or any other response statistic of interest. On the downside, coupling stochastic projection approaches to deterministic solvers is more involved since it requires communication with the FE code at various levels. A more detailed discussion on this issue is presented later in this paper. We demonstrate and establish the advantages of a coupled framework where a deterministic FE software system and a stochastic solver can interact with each other.

In the remainder of this paper we present further details on stochastic projection schemes and how these schemes can be made to interact with deterministic FE software. The next section outlines stochastic finite element analysis including random field discretization and semi-discretization of stochastic PDEs. Section 3 presents a brief introduction to subspace projection schemes. In Section 4 we discuss the issues and the steps involved in interfacing stochastic projection schemes with third-party FE codes when the source code is not available. Section 5 demonstrates how such a coupling can be achieved. We present a case study using *FEAP$_{pv}$* as a third-party FE code. A number of illustrative problems in stochastic mechanics are solved using the coupled framework to demonstrate the capability and the scope of the strategies discussed. In the last section we conclude the work by summarizing the key points and outlining some directions for future research.

2. Stochastic finite elements

To illustrate the basic steps involved in stochastic FEA, consider a two-dimensional isotropic solid with random elasticity matrix. The elasticity matrix can be represented as a function of spatial coordinates and a random dimension, i.e.,

$$\mathbf{D}(\mathbf{x};\omega) = h(\mathbf{x};\omega)\mathbf{D}_0, \tag{2}$$

where $h(\mathbf{x};\omega) : \Omega \times \mathbb{R}^2 \to \mathbb{R}$ represents a random field and $\mathbf{D}_0$ is the deterministic part of the elasticity matrix. To facilitate the computational treatment of uncertainties we need to discretize random fields into a finite number of random variables. In the next subsection we briefly discuss random field discretization techniques.

2.1. Random field discretization

Various discretization techniques are available in the literature [16,17] for approximating random fields, including the mid-point method, shape function methods, optimal linear estimation (OLE), weighted integral methods, orthogonal series expansion and the Karhunen–Loève (KL) expansion scheme; see, for example, Refs. [6,18,19]. Amongst the various techniques for random field discretization, KL expansion is particularly useful since it can capture the variability using a fewer number of random variables. However, for KL expansions, closed form solutions are only available for some special correlation functions and simple geometries. Therefore, a numerical solution is essential to deal with complicated domains and more general correlation functions. In this section we present a Galerkin based numerical procedure for solving the KL eigenvalue problem.

2.2. Numerical solution of KL eigenvalue problem

The Karhunen–Loève expansion is one the most commonly used techniques for representing random fields in terms of a finite set of random variables in a Fourier-type series as

$$h(\mathbf{x};\omega) = \langle h(\mathbf{x};\omega)\rangle + \sum_{i=1}^{M} \theta_i(\omega)\sqrt{\lambda_i}\kappa_i(\mathbf{x}), \tag{3}$$

where $\theta_i(\omega)$ is a set of random variables, λ_i and $\kappa_i(\mathbf{x})$ are the eigenvalues and eigenfunctions of the following integral eigenvalue problem:

$$\int_{\mathscr{D}} R_h(\mathbf{x},\mathbf{x}')\kappa_i(\mathbf{x})\,\mathrm{d}\mathbf{x} = \lambda_i\kappa_i(\mathbf{x}'), \tag{4}$$

where $R_h(\mathbf{x},\mathbf{x}')$ is the correlation function of the random field $h(\mathbf{x};\omega)$. The solution of the above equation can be easily obtained using the numerical quadrature method [20]. However, expansion methods where each eigenfunction is approximated by a linear combination of chosen basis functions and undetermined coefficients, are more accurate and computationally efficient for analytically defined kernels [19]. In an expansion method we approximate each eigenfunction as follows:

$$\kappa_i(\mathbf{x}) = \sum_{j=1}^{N} c_{ij}\pi_j(\mathbf{x}), \tag{5}$$

where $\pi_1(\mathbf{x}), \pi_2(\mathbf{x}), \ldots, \pi_N(\mathbf{x})$ are the basis functions and c_{ij} are the undetermined coefficients. Substituting the above equation into Eq. (4) and orthogonalizing the error with respect to each basis function $(\pi_k(\mathbf{x}))$ gives

$$\sum_{j=1}^{N} c_{ij}\left[\int_\phi \int_\phi R_h(\mathbf{x},\mathbf{x}')\pi_j(\mathbf{x})\pi_k(\mathbf{x}')\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{x}'\right] - \lambda_i \sum_{j=1}^{N} c_{ij}\left[\int_\phi \pi_j(\mathbf{x}')\pi_k(\mathbf{x}')\,\mathrm{d}\mathbf{x}'\right] = 0, \tag{6}$$

which can be compactly written as

$$\mathbf{A}\mathbf{X} = \mathbf{\Lambda}\mathbf{M}\mathbf{X}, \tag{7}$$

where the matrices are defined as follows:

$$\mathbf{A}_{jk} = \int_\phi \int_\phi R_h(\mathbf{x},\mathbf{x}')\pi_j(\mathbf{x})\pi_k(\mathbf{x}')\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{x}', \tag{8}$$

$$\mathbf{X}_{ij} = c_{ij}, \tag{9}$$

$$\mathbf{\Lambda}_{ik} = \delta_{ik}\lambda_k, \tag{10}$$

$$\mathbf{M}_{jk} = \int_\phi \pi_j(\mathbf{x})\pi_k(\mathbf{x})\,\mathrm{d}\mathbf{x}. \tag{11}$$

It is interesting to note that the above matrices can be easily generated by using a standard deterministic FE code if the basis functions $\pi_i(\mathbf{x})$'s are chosen to be the finite element shape functions over the region ϕ [9]. For example, the matrix $\mathbf{M}$ can be generated as a special case of the mass matrix by setting the density value to unity. The matrix $\mathbf{A}$ can be approximated as follows:

$$\mathbf{A} \approx \int_\phi \int_\phi \mathbf{N}(\mathbf{x})^{\mathrm{T}} R_h(\mathbf{x},\mathbf{x}')\mathbf{N}(\mathbf{x}')\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{x}', \tag{12}$$

where $\mathbf{N}(\mathbf{x})$ is a vector of finite element shape functions. The covariance function $R_h(\mathbf{x},\mathbf{x}')$ can be approximated as $\mathbf{N}(\mathbf{x})\mathbf{C}\mathbf{N}(\mathbf{x}')^{\mathrm{T}}$, where $\mathbf{C}_{ij} = R_h(\mathbf{x}_i, \mathbf{x}_j)$. Substituting this approximation for the covariance function into Eq. (12) gives

$$\mathbf{A} = \mathbf{M}\mathbf{C}\mathbf{M}. \tag{13}$$

Once matrices $\mathbf{A}$ and $\mathbf{M}$ are computed Eq. (7) can be readily solved for eigenvalues $\mathbf{\Lambda}$ and the eigenvectors $\mathbf{X}$. The value of the eigenfunctions at any point within the domain can be approximated using Eq. (5).

The discretized version of a general non-Gaussian random field can also be written in a similar form; for example, using a polynomial chaos expansion, the elasticity matrix in Eq. (2) can be written as

$$\widehat{\mathbf{D}}(\mathbf{x};\omega) = \sum_{i=0}^{N} \mathbf{D}_i(\mathbf{x})\Gamma_i(\boldsymbol{\theta}), \tag{14}$$

where $\Gamma_i(\boldsymbol{\theta})$ are multi-dimensional Hermite polynomials in $\theta_1, \theta_2, \ldots, \theta_M$ and $\mathbf{D}_i$ are the coefficients of expansion which can be computed as

$$\mathbf{D}_i(\mathbf{x}) = \frac{\langle \mathbf{D}(\mathbf{x};\omega)\Gamma_i(\boldsymbol{\theta})\rangle}{\langle \Gamma_i^2(\boldsymbol{\theta})\rangle}, \tag{15}$$

where $\langle\cdot\rangle$ denotes the expectation operator.

2.3. Spatial discretization

As the problem is set in a probabilistic framework, spatial discretization has to be carried out using the discretized random fields. Using standard mesh generation procedures, the spatial domain can be readily discretized into a number of elements [21]. However, since the constitutive matrix is random, the stochastic element stiffness matrix is given by the following integral evaluated over the domain $\mathscr{D}_e$ of each element

$$k^e = \int_{\mathscr{D}_e} \mathbf{B}^{\mathrm{T}}\mathbf{D}(\mathbf{x};\omega)\mathbf{B}\,\mathrm{d}\mathbf{x}, \tag{16}$$

where $\mathbf{B}$ is the strain–displacement matrix. Substituting Eq. (14) into the above equation gives

$$k^e = \int_{\mathscr{D}_e} \mathbf{B}^{\mathrm{T}}\left[\sum_{i=0}^{N} \mathbf{D}_i(\mathbf{x})\Gamma_i(\boldsymbol{\theta})\right]\mathbf{B}\,\mathrm{d}\mathbf{x} = \sum_{i=0}^{N} k_i^e\Gamma_i(\boldsymbol{\theta}), \tag{17}$$

where $k_i^e = \int_{\mathscr{D}_e} \mathbf{B}^{\mathrm{T}}\mathbf{D}_i(\mathbf{x};\omega)\mathbf{B}\,\mathrm{d}\mathbf{x}$. Computation of the above integral is straightforward if the deterministic FE solver allows for a user defined elasticity matrix. However, alternative strategies for approximating Eq. (17) must be developed if such a capability does not exist. In summary, to compute each element matrix the deterministic solver must receive random field data from the stochastic solver, i.e., the expansion coefficient $\mathbf{D}_i(\mathbf{x})$.

Subsequent assembly of the element stiffness matrices by the deterministic solver and application of the specified boundary conditions result in the global stochastic stiffness matrix $\mathbf{K}(\boldsymbol{\theta}) \in \mathbb{R}^{n\times n}$. For a solid subjected to static loads we arrive at the following system of linear random algebraic equations:

$$\mathbf{K}(\boldsymbol{\theta})\mathbf{u}(\boldsymbol{\theta}) = \mathbf{f}, \tag{18}$$

where $\mathbf{u}(\boldsymbol{\theta}) \in \mathbb{R}^n$ is the random displacement vector. $\mathbf{f} \in \mathbb{R}^n$ denotes the excitation vector which we assume to be deterministic for simplicity of presentation. It follows from Eq. (17) that the global stiffness matrix $\mathbf{K}(\boldsymbol{\theta})$ can be written in the following form:

$$\mathbf{K}(\boldsymbol{\theta}) = \sum_{i=0}^{N} \mathbf{K}_i\Gamma_i(\boldsymbol{\theta}), \tag{19}$$

where $\Gamma_i(\boldsymbol{\theta})$ are multi-dimensional Hermite polynomials in $\theta_1, \theta_2, \ldots, \theta_M$, $\mathbf{K}_0 \in \mathbb{R}^{n\times n}$ is the mean global stiffness matrix and $\mathbf{K}_i \in \mathbb{R}^{n\times n}$ are weighted stiffness matrices.

3. Stochastic subspace projection schemes

A number of numerical methods for solving Eq. (18) can be found in the literature such as perturbation methods, Neumann series expansions, response surface methods, etc. However, our focus is on the application of stochastic projection schemes since recent studies have shown that they are computationally efficient and provably convergent [11–15]. The fundamental result underpinning stochastic reduced basis projection schemes is that the solution of Eq. (18) lies in the stochastic Krylov subspace. The stochastic Krylov subspace of order m is defined as

$$\mathscr{K}_m(\mathbf{K}(\boldsymbol{\theta}),\mathbf{f}) = \mathrm{span}\{\mathbf{f},\mathbf{K}(\boldsymbol{\theta})\mathbf{f},\mathbf{K}(\boldsymbol{\theta})^2\mathbf{f},\ldots,\mathbf{K}(\boldsymbol{\theta})^{m-1}\mathbf{f}\}. \quad (20)$$

A stochastic reduced basis approximation of the solution process can hence be written as

$$\hat{\mathbf{u}}(\boldsymbol{\theta}) = \xi_0\boldsymbol{\psi}_0(\boldsymbol{\theta}) + \xi_1\boldsymbol{\psi}_1(\boldsymbol{\theta}) + \cdots + \xi_n\boldsymbol{\psi}_m(\boldsymbol{\theta}) = \boldsymbol{\Psi}(\boldsymbol{\theta})\boldsymbol{\xi}, \quad (21)$$

where $\boldsymbol{\Psi}(\boldsymbol{\theta}) = \{\boldsymbol{\psi}_0(\boldsymbol{\theta}),\boldsymbol{\psi}_1(\boldsymbol{\theta}),\ldots,\boldsymbol{\psi}_m(\boldsymbol{\theta})\} \in \mathbb{R}^{n\times(m+1)}$ is a set of basis vectors spanning the stochastic Krylov subspace $\mathscr{K}_m(\mathbf{K}(\boldsymbol{\theta}),\mathbf{f})$ and $\boldsymbol{\xi} = \{\xi_0,\xi_1,\ldots,\xi_m\}^{\mathrm{T}} \in \mathbb{R}^{m+1}$ is a vector of undetermined coefficients. An arbitrary degree of accuracy can be achieved by using more and more basis vectors from $\mathscr{K}_m(\mathbf{K}(\boldsymbol{\theta}),\mathbf{f})$. In practice, however, basis vectors spanning the preconditioned stochastic Krylov subspace $\mathscr{K}_m(\langle\mathbf{K}(\boldsymbol{\theta})\rangle^{-1}\mathbf{K}(\boldsymbol{\theta}),\mathbf{f})$ are used to ensure that highly accurate results can be obtained using a few basis vectors. A good choice for preconditioner is the deterministic matrix $\mathbf{K}_0^{-1}$ [11].

It is to be noted here that the basis vectors chosen from the preconditioned stochastic Krylov subspace $\mathscr{K}_m(\langle\mathbf{K}_0^{-1}\mathbf{K}(\boldsymbol{\theta}),\mathbf{f})$ can be readily applied to solve problems with Gaussian uncertainties. However, to facilitate the treatment of non-Gaussian uncertainties basis vectors shall be further expanded in terms of multi-dimensional Hermite polynomials. This can be easily achieved by exploiting the recursive nature of the basis vectors spanning $\mathscr{K}_m(\langle\mathbf{K}_0^{-1}\mathbf{K}(\boldsymbol{\theta}),\mathbf{f})$. This step also enables the application of higher order reduced basis methods which is difficult to achieve otherwise. For a detailed discussion and motivation; see, for example, [13–15]. The general expression for the stochastic basis vector $\hat{\boldsymbol{\psi}}_m(\boldsymbol{\theta})$ after rewriting in terms of Hermite polynomials is given by

$$\hat{\boldsymbol{\psi}}_m(\boldsymbol{\theta}) = \sum_{i=1}^{P_1}\psi_i^m\Gamma_i(\boldsymbol{\theta}), \quad (22)$$

where $\psi_k^m = \left(\mathbf{K}_0^{-1}\sum_{i=0}^{P_1}\sum_{j=0}^{P_1}\mathbf{K}_i\psi_j^m\mathbf{D}_{ijk}\right)/\langle\Gamma_k^2\rangle$ (for $m > 1$) and the tensor $\mathbf{D}_{ijk} = \langle\Gamma_i\Gamma_j\Gamma_k\rangle$. Substituting the preceding equation into Eq. (21) and rearranging gives

$$\hat{\mathbf{u}}(\boldsymbol{\theta}) = \left[\sum_{i=0}^{P_1}\boldsymbol{\Pi}_i\Gamma_i\right]\hat{\xi}, \quad (23)$$

where $\boldsymbol{\Pi}_i = [\boldsymbol{\psi}_i^0,\boldsymbol{\psi}_i^1,\ldots,\boldsymbol{\psi}_i^m] \in \mathbb{R}^{n\times(m+1)}$.

To compute the approximate solution process we employ the Galerkin projection scheme which ensures that the residual error vector is orthogonal to the approximating subspace. Substituting Eqs. (19) and (23) into Eq. (18) gives

$$\boldsymbol{\epsilon}(\boldsymbol{\theta}) = \left(\sum_{i=0}^{N}\mathbf{K}_i\Gamma_i(\boldsymbol{\theta})\right)\left(\sum_{i=0}^{P_1}\boldsymbol{\Pi}_i\Gamma_i(\boldsymbol{\theta})\right)\hat{\xi} - \mathbf{f}. \quad (24)$$

The vector of undetermined coefficients $\hat{\xi}$ can now be computed by enforcing the Galerkin orthogonality condition $\boldsymbol{\epsilon}(\boldsymbol{\theta}) \perp \boldsymbol{\Psi}(\boldsymbol{\theta})$ which gives the following system of equations:

$$\left\langle\left(\sum_{i=0}^{P_1}\boldsymbol{\Pi}_i^{\mathrm{T}}\Gamma_i\right)\left(\sum_{j=0}^{N}\mathbf{K}_j\Gamma_j\right)\left(\sum_{k=0}^{P_1}\boldsymbol{\Pi}_k\Gamma_k\right)\right\rangle\hat{\xi} = \left\langle\left(\sum_{i=0}^{P_1}\boldsymbol{\Pi}_i^{\mathrm{T}}\Gamma_i\right)\mathbf{f}\right\rangle. \quad (25)$$

Since by definition $\langle\Gamma_i\rangle = 0$ for $i > 0$ and $\langle\Gamma_0\rangle = 1$, the preceding equation simplifies to

$$\left(\sum_{i=0}^{P_1}\sum_{j=0}^{N}\sum_{k=0}^{P_1}\boldsymbol{\Pi}_i^{\mathrm{T}}\mathbf{K}_j\boldsymbol{\Pi}_k\mathbf{D}_{ijk}\right)\hat{\xi} = \boldsymbol{\Pi}_0^{\mathrm{T}}\mathbf{f}, \quad (26)$$

where $\mathbf{D}_{ijk} = \langle\Gamma_i\Gamma_j\Gamma_k\rangle$. It can be noted that Eq. (26) is a reduced order deterministic set of linear algebraic equations of size $(m+1)\times(m+1)$ which can be readily solved for the coefficient vector $\hat{\xi}$. Alternative projection schemes based on the stochastic Petrov–Galerkin approach and strong orthogonality conditions can also be devised to estimate the undetermined coefficients; see Nair [13] for an overview of the theoretical aspects of stochastic reduced basis projection schemes.

It is straightforward to post-process the final expression of the solution process obtained using stochastic reduced basis projection schemes. Applying the expectation operator to Eq. (23) gives the following expression for the mean:

$$\bar{\mathbf{u}} = \left\langle\left[\sum_{i=0}^{P_1}\Pi_i\Gamma_i\right]\hat{\xi}\right\rangle, \quad (27)$$

which reduces to $\bar{\mathbf{u}} = \Pi_0\hat{\xi}$ after using the properties of the PC basis functions. Closed form expressions for the covariance matrix and the L_2-norm of the residual error can also be readily derived; see, Ref. [15].

4. A typical coupled framework

In this section we present a brief outline of how a general stochastic toolkit capable of performing random field discretization and solving a system of random algebraic equations can be coupled with existing deterministic FE codes with minimal or no intrusive modifications. Fig. 1 outlines an overview of the steps involved in a typical coupled software framework. It shows various levels of interactions between different computational modules. The model information is accessed via an intervening interface by the stochastic solver to facilitate the random field discretization. The stochastic solver then sends the discretized random field data to the deterministic code which it uses to create the assembled mean and weighted stiffness matrices.

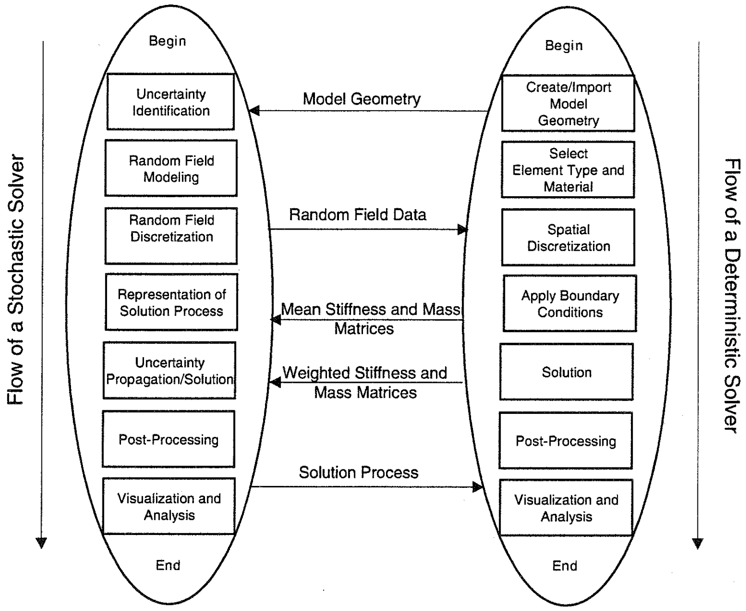
Flow of a Stochastic Solver
Begin
Uncertainty Identification
Random Field Modeling
Random Field Discretization
Representation of Solution Process
Uncertainty Propagation/Solution
Post-Processing
Visualization and Analysis
End
Model Geometry
Random Field Data
Mean Stiffness and Mass Matrices
Weighted Stiffness and Mass Matrices
Solution Process
Begin
Create/Import Model Geometry
Select Element Type and Material
Spatial Discretization
Apply Boundary Conditions
Solution
Post-Processing
Visualization and Analysis
End
Flow of a Deterministic Solver

Fig. 1. Levels of communications between a stochastic solver and a deterministic solver.

Subsequently the stochastic solver approximates the solution process using stochastic projection schemes. Post-processing can be done using either or both the solvers.

4.1. Choice of interface

Before delving into the details of the various steps listed in Fig. 1, we first discuss some issues involved in the design and development of an appropriate intervening interface. Such an interface is essential to facilitate the efficient transfer of data and information between various codes. Essentially, we need to create various input files for the deterministic FE software to generate the mean and weighted stiffness matrices. Automation of the process, exploitation of parallelization opportunities and a graphical user interface (GUI) are some of the desired features. Any of the computer languages, *viz.*, C, C++, Java, Fortran, etc. can be used to create such an intervening link. However, we propose the use of the Matlab [22] interpretive technical computing environment to build the interface. Matlab provides an intuitive computing environment with advanced tools for algorithm and application development. Besides, it can be used as a powerful scripting environment to control the execution and inputs/outputs from a number of participating codes. As Matlab is routinely and widely used in academia and industry to build and test algorithms, it seems to be a preferred environment to be able to attract a wide variety of users as well as developers.

4.2. General approach

Fig. 2 lists the set of instructions a user must carry out either manually or via the Matlab scripting environment. Step 1 involves creating a geometry model of the structure under consideration. For simple geometries deterministic FE software or the open source mesh generator DISTMESH [23] can be used. For more complex geometries any of the readily available CAD packages such as CATIA or Pro/Engineer can be helpful. Subsequently, a mesh is created using the same software and written to a file. Most of the CAD packages provide access to nodal and element connectivity data which can be easily read by Matlab.

In the next step the analyst must identify uncertain parameters and construct probabilistic models. For scalar uncertainties probability distribution functions of the random parameters are constructed. This could be done by applying kernel density estimation techniques to available field data [24] or by soliciting expert opinion. While, for spatially varying random fields a suitable correlation function is assumed or constructed from field data [16]. Next, the random field model, i.e, mean, standard deviation and the correlation function is passed to the stochastic toolkit to carry out the random field discretization. The stochastic solver needs the mesh description for discretizing the random field model.[1] The random field can be discretized on the structural mesh using any of the discretization techniques such as the Karhunen–Loéve expansion scheme (as discussed earlier). The data arising from random field discretization is subsequently passed to the deterministic solver to facilitate the spatial discretization.

Spatial discretization involves the computation of the integral in Eq. (17). As mentioned earlier if the deterministic FE code allows for user defined constitutive properties,

[1] It is possible in theory to carry out random field discretization independently of a CAD package – we present a problem later where we have chosen to do so to demonstrate the generality of implementation.

```
Algorithm for Coupling
1. Create geometry and mesh with a CAD software
2. Identify uncertain parameters
3. Create random field models
4. Discretize random fields using the stochastic toolkit
5. Select element types, material types and boundary conditions
6. Retrieve data from steps 1, 4 and 5
7. Create input files for deterministic FE software
For i = 1 to N+1 do
8. Run deterministic FE software, determine $\mathbf{K}_i$
End
9. Retrieve and send all $\mathbf{K}_i$ and $\mathbf{f}_i$ to stochastic toolkit
10. Run stochastic solver to generate the solution process
11. Postprocessing and visualization of response statistics
```

Fig. 2. Computational steps involved for N random variables.

this can be done efficiently. Here, the expansion coefficients in Eq. (14) are treated as pseudo-elasticity matrices for calculating PC expansions of the element stiffness matrix. An alternative approach is to use the following approximation for the stochastic element stiffness matrix [10]:

$$k^e = \int_{\mathscr{D}_e} \mathbf{B}^{\mathrm{T}}\mathbf{D}_0\mathbf{B}\,\mathrm{d}\mathbf{x} + \left[\int_{\mathscr{D}_e} \mathbf{B}^{\mathrm{T}}\left(\sum_{i=1}^{N}\mathbf{D}_i(\mathbf{x}_{\mathrm{cent}})\right)\mathbf{B}\,\mathrm{d}\mathbf{x}\right]\Gamma_i(\boldsymbol{\theta}). \tag{28}$$

In other words, it is assumed that the expansion coefficients do not change significantly over the element domain. Each element appears to have a unique constitutive matrix given by $\mathbf{D}_i(\mathbf{x}_{\mathbf{cent}})$, where $\mathbf{x}_{\mathbf{cent}}$ is the centroid of the element under consideration.

Having obtained the mesh data, random field data, elements and material constants, loading, and boundary conditions we can generate $N+1$ different input files using Matlab. These input files contain instructions to be carried out by the deterministic FE software. Input files can be fed in parallel into the deterministic FE software to compute the mean and weighted stiffness matrices and load vectors. Given the mean and weighted stiffness matrices a PC expansion of the global stiffness matrix can be constructed as in Eq. (19). Given the PC expansion of the stochastic global stiffness matrix, we can run the stochastic solver and apply stochastic projection schemes to extract the solution process by solving a system of linear random algebraic equations. The current methodology can also be used for direct Monte Carlo simulation and perturbation analysis. It is worth noting here that PC projection schemes [25] can also be used in lieu of the stochastic reduced basis projection scheme in this step. Once the solution process has been approximated in a reduced basis, post-processing can be carried out to extract all the response statistics of interest. The summary statistics can be fed back into the deterministic FE code for graphical post-processing, if required.

5. Demonstrative applications

In this section we demonstrate how the steps discussed in the previous sections can be implemented to solve a range of problems in probabilistic mechanics. We have chosen $FEAP_{pv}$ as the third-party FE code for the case studies. However, the approach presented is quite general and can be used in conjunction with any third-party FE codes such as ABAQUS and ANSYS [26].

5.1. Implementation details

It is evident that leveraging the deterministic FE capabilities compliments the stochastic FE analysis. Fig. 3 shows the essential elements of a typical stochastic finite element library. Model building, mesh generation and computation of stiffness matrices can be done in a conventional deterministic manner whereas random field discretization and solution of the system of random algebraic equations is carried out using stochastic solvers.

We now illustrate how the steps discussed in earlier sections can be implemented to solve a typical problem in probabilistic mechanics. Consider a thin square plate of unit length, clamped at one edge while being subjected to

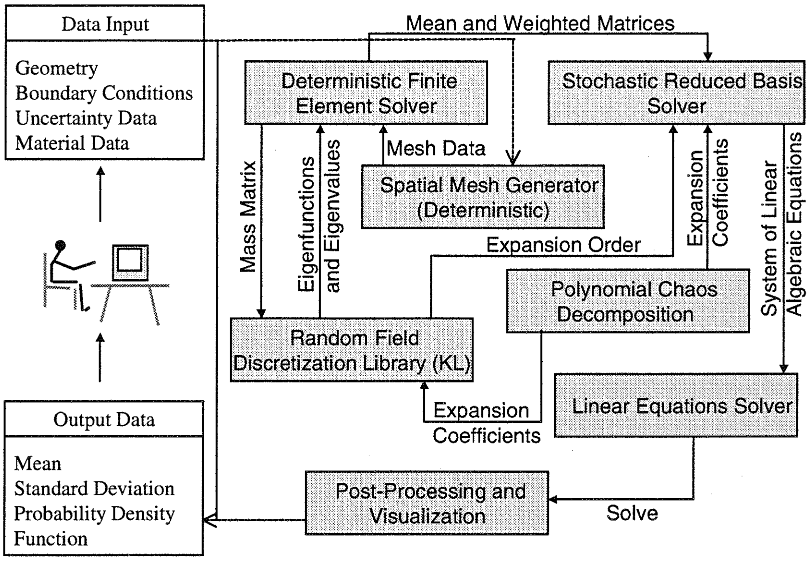
Data Input
Geometry
Boundary Conditions
Uncertainty Data
Material Data
Mean and Weighted Matrices
Deterministic Finite Element Solver
Stochastic Reduced Basis Solver
Mesh Data
Spatial Mesh Generator (Deterministic)
Mass Matrix
Eigenfunctions and Eigenvalues
Expansion Order
Expansion Coefficients
System of Linear Algebraic Equations
Polynomial Chaos Decomposition
Random Field Discretization Library (KL)
Expansion Coefficients
Linear Equations Solver
Output Data
Mean
Standard Deviation
Probability Density Function
Post-Processing and Visualization
Solve

Fig. 3. A typical stochastic library.

a uniform in plane tension at the opposite edge; see, Fig. 4. The external loads are assumed to be deterministic and of unit magnitude.

The domain of the plate is discretized into 16 square elements which leads to a total of 50 degrees of freedom. A mesh file is generated using Matlab containing nodal coordinates and connectivity data for the elements. The modulus of elasticity of the plate is modeled as a two-dimensional Gaussian random field with mean $\mu = 1$ and standard deviation $\sigma = 0.2$. The random field is assumed to follow the exponential correlation model given below:

$$R(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{|x_1 - x_1'|}{b_1} - \frac{|x_2 - x_2'|}{b_2}\right), \tag{29}$$

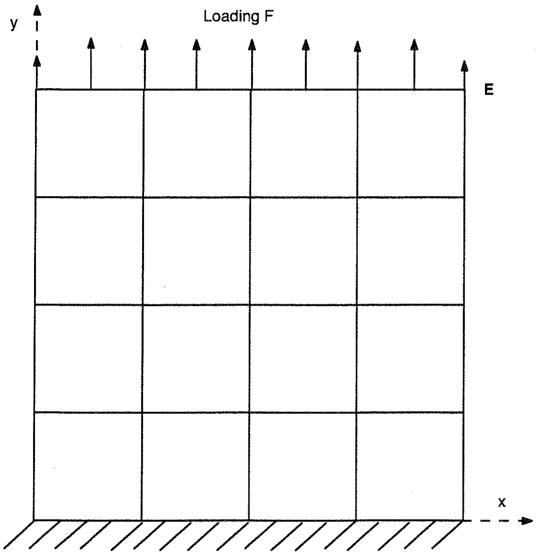

Fig. 4. Schematic of the plate problem.

where $b_1 = b_2 = 1$. The random field is discretized using the KL scheme with four random variables. We select plane stress elements and material type as elastic isotropic with $v = 0$. With this information we generate five input files for $FEAP_{pv}$, one for the mean stiffness matrix and the other four for weighted stiffness matrices. These input files are fed into $FEAP_{pv}$ to compute the matrices $\mathbf{K}_i$, $i = 0, 1, \ldots, 4$ and the load vector which are eventually used by the stochastic solver to compute the solution process. We use three stochastic basis vectors for reduced basis approximation of the solution process.

Three strategies for calculating the element stiffness matrices are compared. In the first strategy, the integral in Eq. (16) is directly evaluated using Gauss quadrature. In the second strategy, the element stiffness matrices are computed via the centroid approximation given in Eq. (28). In both the strategies we use the analytical solution for the KL eigenvalue problem over the square domain. In the third strategy we use the centroid approximation technique in Eq. (28) and the expansion coefficients $\mathbf{D}_i(\mathbf{x})$ are computed using Galerkin based numerical solution of the KL eigenvalue problem as presented in Section 2.

Table 1 compares the results obtained using the three strategies. We compare the mean and standard deviation of the displacement at point E in Fig. 4. It can be noted that the accuracy of Eq. (28) is reasonable even for the coarse mesh used. The Galerkin based numerical solution

Table 1
Comparison of discretization schemes

Deflection at point E	Mean deflection in y-direction	Standard deviation in y-direction
KL Exact	1.03249	0.19859
KL Approx.	1.03381	0.20148
KL Numerical	1.03414	0.19919

gives results very similar to the analytical solution for the test case. The approximation errors incurred tend to reduce rapidly when the mesh is refined further. It is anticipated that for problems where the correlation length of the random field is small, a highly refined mesh may be required to ensure good approximations.

In the next subsection we present some illustrative examples to demonstrate the capabilities of the coupling strategy outlined in the present paper.

5.2. Elasticity problem – a plate with holes

The problem considered here is a square plate with five circular holes as shown in Fig. 5. The plate geometry is defined by the parameters $l = 1$, $R = 0.2$ and $r = 0.1$. The plate is clamped at the bottom edge CD and subjected to a uniform in-plane tension of unit magnitude at the opposite edge AB. The Young's modulus of the plate is modeled by a two-dimensional homogeneous Gaussian random process with exponential correlation function given by Eq. (29), where $\mathbf{b} = [0.1 \quad 0.1]$. The mean and the standard deviation of the Young's modulus are assumed to be unity and 0.2, respectively and Poisson's ratio is set to zero. Fig. 6 shows the spatially discretized plate containing 945 triangular elements and 566 nodes. The model contains a total of 1132 degrees of freedom with two degrees of freedom per node. Mesh generation was carried out using the open source code DISTMESH [23].

The random field is discretized using the numerical KL scheme. As discussed earlier in Section 2.1 a numerical solution of the KL integral eigenvalue problem is crucial to discretize random fields defined over complicated domains such as the one shown in Fig. 5. A finite element based numerical approach is used here to decompose the exponential correlation kernel which provides the eigenfunctions and the eigenvalues.

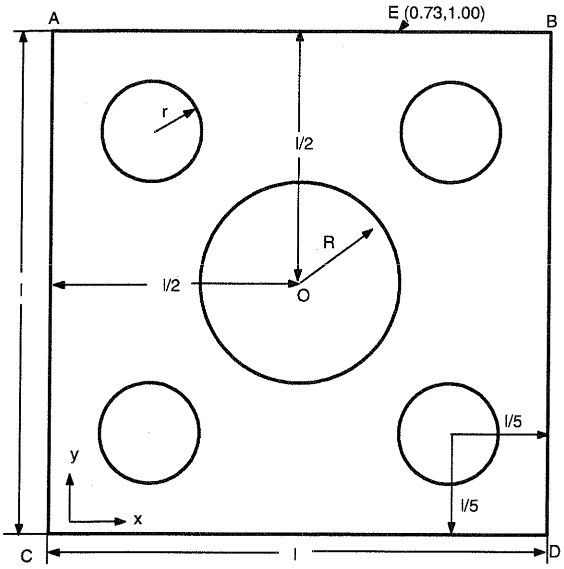

Fig. 5. Schematic for problem.

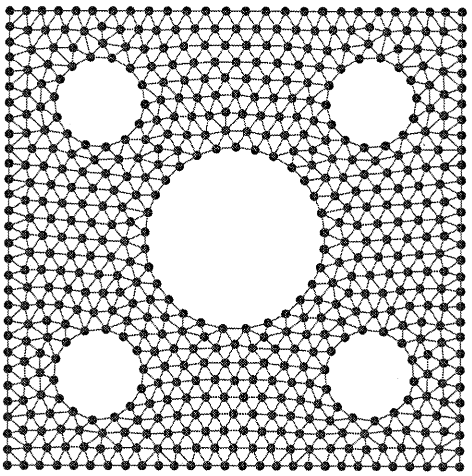

Fig. 6. Spatial triangulated mesh.

We use $FEAP_{pv}$ to compute the mean and weighted stiffness matrices for carrying out the probabilistic analysis using the stochastic solver. The mean and standard deviation of the nodal response values are computed for up to six terms in the KL expansion and up to seven stochastic basis vectors in reduced basis approximation of the solution process. Figs. 7 and 8 exhibit the results when six modes are retained in the KL expansion and fifth-order stochastic reduced basis scheme is used to estimate the solution. Fig. 7a shows the undeformed mesh and imposed deterministic deformation and Fig. 7b shows undeformed mesh and imposed probabilistic mean deformation. The two figures look quite similar, however, the standard deviation of displacement as reported in Fig. 8 is significant in both directions.

Table 2 contains the estimated mean and standard deviation of displacement at the point E in Fig. 5. The coordinates of the point E are $x = 0.73$ and $y = 1.00$ and is the point of maximum deflection in the y-direction. Mean and standard deviation are computed in both x- and y-directions using reduced basis approximations with up to six stochastic basis vectors and with six independent random variable in the KL expansion.

It can be observed from the results presented in Table 2 that the values of the mean and the standard deviation of displacement at the point E tend to converge as the order of approximation is increased. In principle, a good estimate of the solution process can only be obtained by reducing the error in representing the random field as well as in solving the system of random algebraic equations. Further, the mesh used for spatial discretization must be sufficiently fine to capture the local solution characteristics.

5.3. Aircraft wing – bending problem

The next problem considered is a simplified aircraft wing model in bending. The wing geometry is shown in Fig. 9, where the values in brackets show the coordinates of points

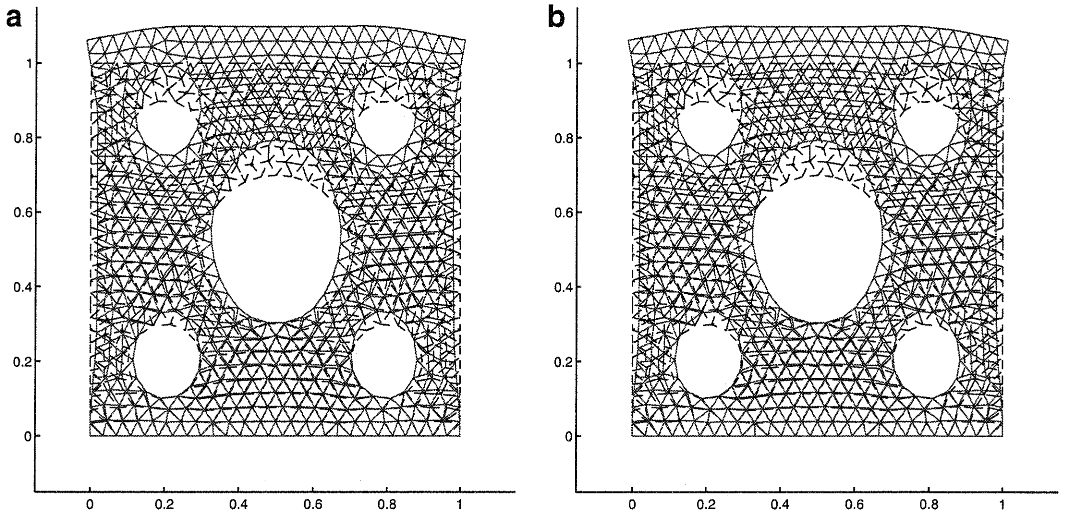

Fig. 7. Undeformed mesh, deterministic solution and the mean solution using six stochastic basis vectors and six terms in the the KL expansion: (a) undeformed mesh and deterministic deflection and (b) undeformed mesh and mean deflection.

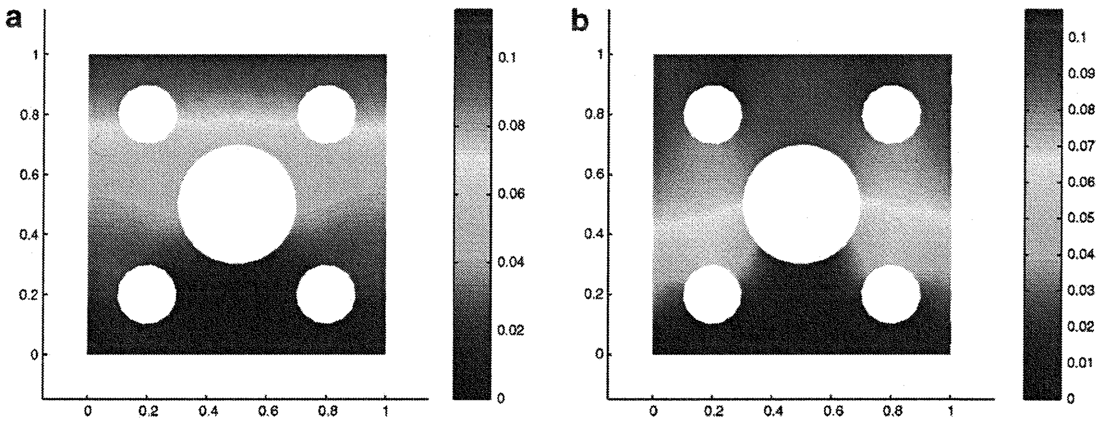

Fig. 8. Estimation of standard deviation using six stochastic basis vectors and six terms in the KL expansion: (a) standard deviation in x-direction and (b) standard deviation in y-direction.

Table 2
Estimated values of the mean and the standard deviation using SRBMs of order up to five and six terms in the KL expansion

For $M = 6$	Mean		Standard deviation	
	Direction x	Direction y	Direction x	Direction y
SRBMI	8.311095e−02	2.541100e+00	1.088438e−01	1.042519e−01
SRBMII	8.376228e−02	2.541139e+00	1.108716e−01	1.061627e−01
SRBMIII	8.377725e−02	2.541139e+00	1.107900e−01	1.061283e−01
SRBMIV	8.376410e−02	2.541138e+00	1.107829e−01	1.061225e−01
SRBMV	8.376349e−02	2.541138e+00	1.107865e−01	1.061225e−01

A, B, C, D and E, respectively. All degrees of freedom are restrained along the edge AE and a transverse force F of magnitude 2 units is applied in z-direction along the edge CD. The Young's modulus of the wing is assumed to be a two-dimensional homogeneous Gaussian random process with mean = 0.26 MPa and standard deviation 10% of the mean value and Poisson's ration $v = 0.3$. The exponential correlation function is used to model the uncertainty in the Young's modulus.

As mentioned earlier in Section 4.2, random field discretization can be carried out independently of the FE solver by using any of the CAD packages in conjunction with a random field dicretization library. This is a desirable feature of a stochastic FEA framework for a number of

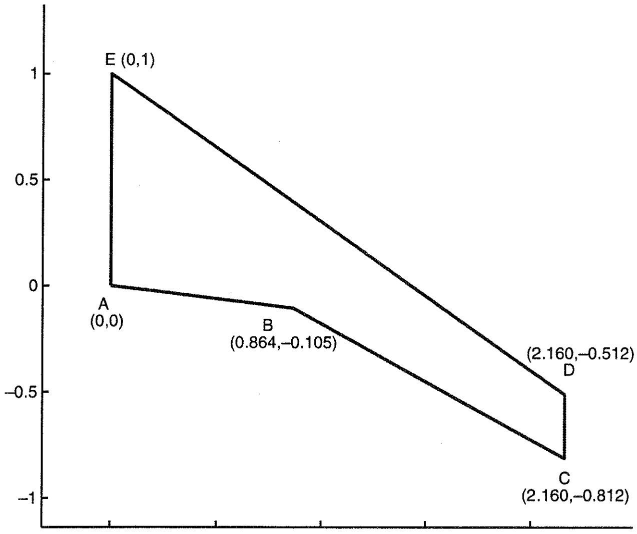

Fig. 9. Schematic of an airplane wing.

reasons [27]. Firstly, since the random field discretization procedure does not change with the choice of element formulations and boundary conditions, discretized random field data can be computed and stored and made available whenever needed. Secondly, for loosely correlated random fields one can have the freedom of choosing a relatively coarser mesh compared to the spatial mesh. Thirdly, the spatial mesh is regulated by the stress gradients of the response whereas random field mesh size depends on the correlation length of the covariance function [28]. Hence, a separate mesh is often desirable in practice.

Here we demonstrate the generality of our software framework which allows the use of two independent meshes for spatial and random field discretization, respectively. Fig. 10a shows the spatial mesh for the airplane wing problem. The discretization is carried out using the open source code DISTMESH [23]. The mesh contains a total of 133 triangulated elements with 97 nodes. For illustration purpose, we use a separate mesh for random field discretization as shown in Fig. 10b. This mesh is generated using ANSYS and contains 157 four-noded planar elements with 188 nodes. Once again we use a spatially varying exponential correlation function as in Eq. (29) to characterize the random field.

The number of KL eigenmodes to be retained while discretizing a random field depends on the magnitudes of the corresponding eigenvalues. Only those eigenvalues with greatest values are retained. The eigenvalues represent the amount of energy (variance) captured along each of the corresponding eigenmode. The percentage of the total energy captured by first M eigenmodes can be computed by the formula

$$E_{\text{captured}} = \frac{\sum_{i=1}^{M}\lambda_i}{\sum_{i=1}^{\infty}\lambda_i} \times 100. \tag{30}$$

The fact that a larger number of terms in the KL expansion is necessary to represent the random field accurately for smaller correlation lengths is supported by the curves shown in Fig. 11. Fig. 11 represents the decaying trend of the eigenvalues obtained by solving the KL integral eigenvalue problem for the wing problem. Note that for larger correlation lengths the decay is rapid whereas the decay is slower for smaller correlation lengths; e.g., using only 10 terms 92% variance can be captured for correlation length $[2.0 \quad 1.0]$ whereas 89 terms are needed to capture the same amount of variance when correlation length is reduced to $[0.2 \quad 0.1]$.

We analyze the response at the point C on the wing structure (see Fig. 9), which is the point of maximum transverse displacement. The mean transverse displacement and the standard deviation are computed for the correlation length, $\mathbf{b} = [0.2 \quad 0.1]$. Fig. 12 shows the estimated mean

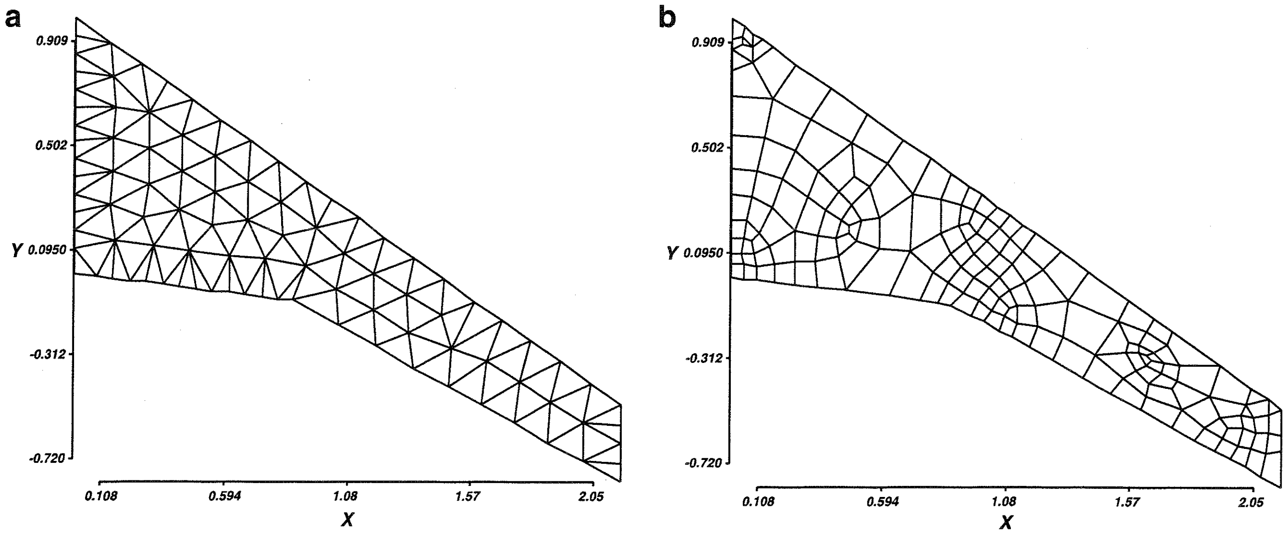

Fig. 10. Spatial and random field meshes for the airplane wing problem: (a) spatial mesh and (b) random field mesh.

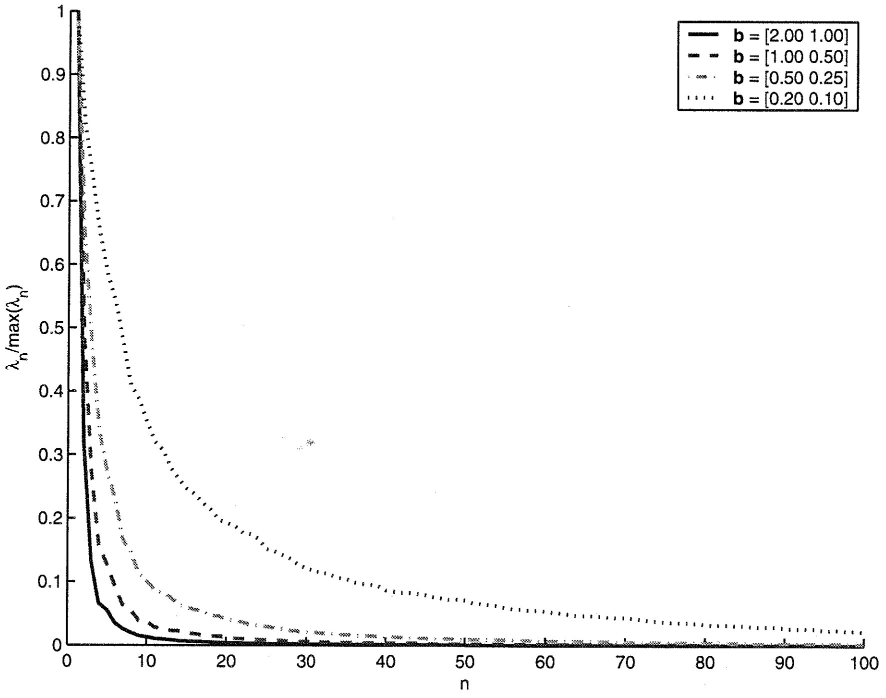

Fig. 11. Normalized eigenvalues for different correlation lengths.

and standard deviation by using first- and second-order reduced basis methods for up to 50 terms in the KL expansion. The graph for estimated percentage variance captured for increasing number of eigenmodes is also shown which illustrates the contribution of each eigenvalue.

Once appropriate meshes for spatial and random field discretization have been chosen, a typical solution methodology to a probabilistic problem involves two levels. One is establishing the minimum number of terms to be retained in the KL expansion for a given correlation length to represent the spatial variability accurately. For the wing structure the first ten eigenmodes capture more than 92% variance when the correlation length is [2.0 1.0] which ensures an acceptable level of accuracy. At the second level an appropriate number of basis vectors for representing the solution process is required to minimize the errors in solving the resulting system of random algebraic equations.

We now approximate the solution using up to four stochastic basis vectors for the case when correlation length **b** is [2.0 1.0]. Convergence is rapidly achieved for the mean and the standard deviation. Estimated L_2 norm of residual error values for SRBMI, SRBMII and SRBMIII are 4.747e−3, 1.362e−3 and 3.716e−4, respectively. Fig. 13 shows the mean values of the displacement of the wing estimated using third-order stochastic reduced basis method. It was noted that the maximum value of the transverse displacement given by deterministic analysis is 0.470, whereas the mean and standard deviation of deflection at the same point given by the probabilistic approach is 0.474 and 0.036, respectively. Fig. 14 shows the standard deviation of the transverse deflection along the span of the wing.

5.4. Timing studies

Solving real life engineering problems in a probabilistic setting becomes computationally challenging especially when the number of random variables and/or degrees of freedom are increased. It has been a daunting task to solve a problem in stochastic mechanics with more than a few tens of random variables and a few thousands of degrees of freedom. In this section we demonstrate how such complex problems can be solved using a coupled framework which combines reduced basis projections schemes with third-party FE software. We present some timing studies conducted on a linear static plate problem with random Young's modulus.

Consider again the simple plate problem described in Section 5.1. We discretized the plate into a number of quadrilateral elements and compute the solution process using second-order reduced basis projection schemes. We retained 20 random variables in the KL expansion for random field discretization. The scatter plot (on log scale) in Fig. 15 reports the wall time taken to solve the problem for increasing number of degrees of freedom. The runs were conducted on a Pentium IV 2 GHz processor running on Linux. The wall time shown in the graph does not account for the time taken for random field discretization. The total wall time shown includes the time taken for creating the input files for the deterministic FE software, time taken by deterministic FE software to compute the mean and weighted matrices and the time taken by stochastic solver to compute the solution process and its statistics. We computed the solution process for up to 20,402 degrees

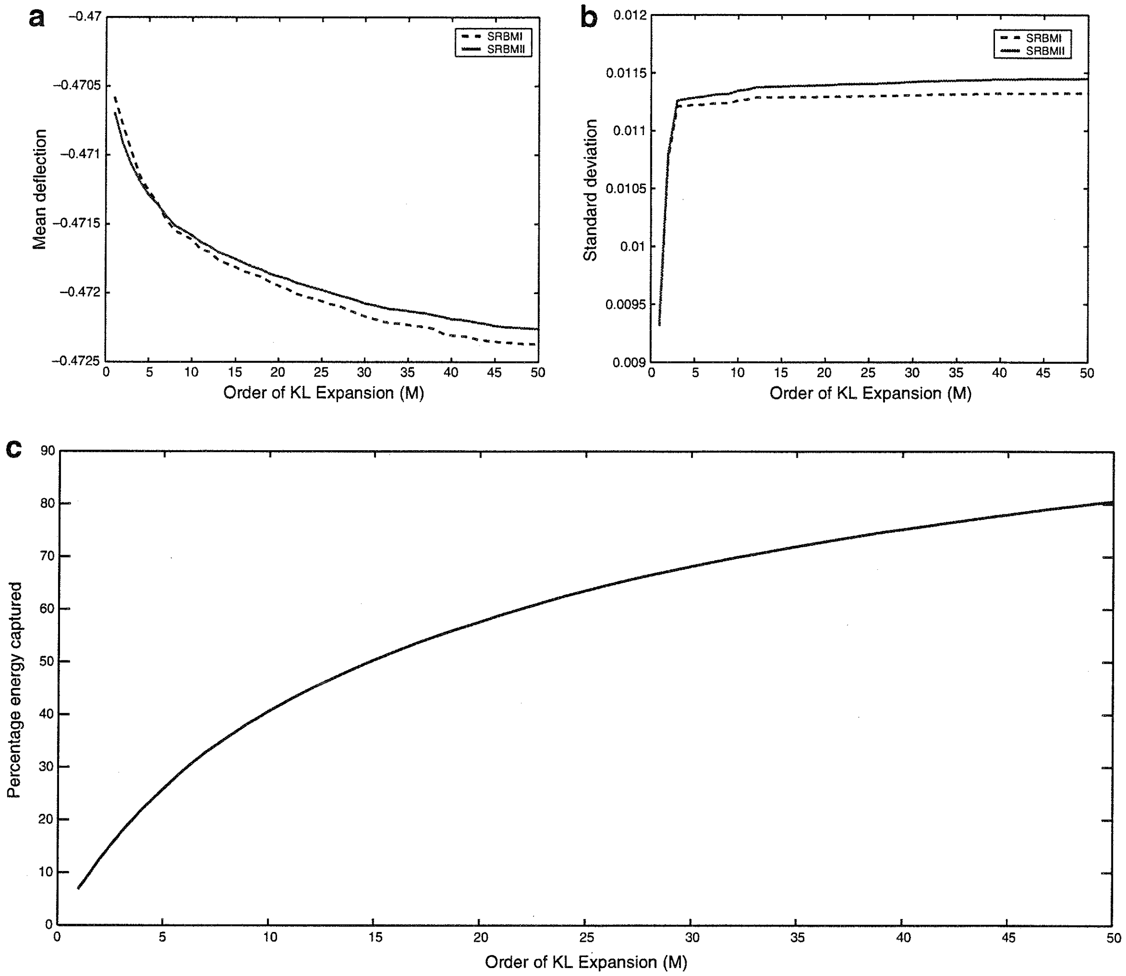

Fig. 12. Mean displacement, standard deviation and estimated variance for correlation length [0.2 0.1] versus number of terms in the KL expansion: (a) mean displacement computed using SRBMI and SRBMII, (b) standard deviation computed using SRBMI and SRBMII and (c) percentage energy captured.

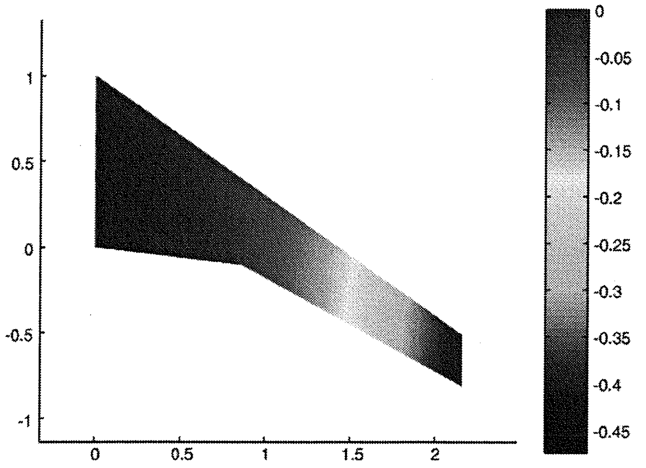

Fig. 13. Mean transverse deflection.

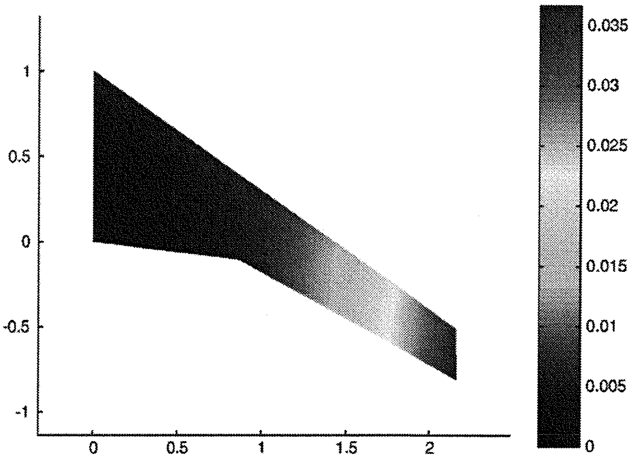

Fig. 14. Standard deviation of transverse deflection.

of freedom. The results obtained are quite encouraging and suggest that problems with thousands of degrees of freedom and many random variables can be solved in under an hour of computer time. Note that alternative schemes such as MCS and PC projection schemes would be significantly more expensive for the same problem [14]. It is to be

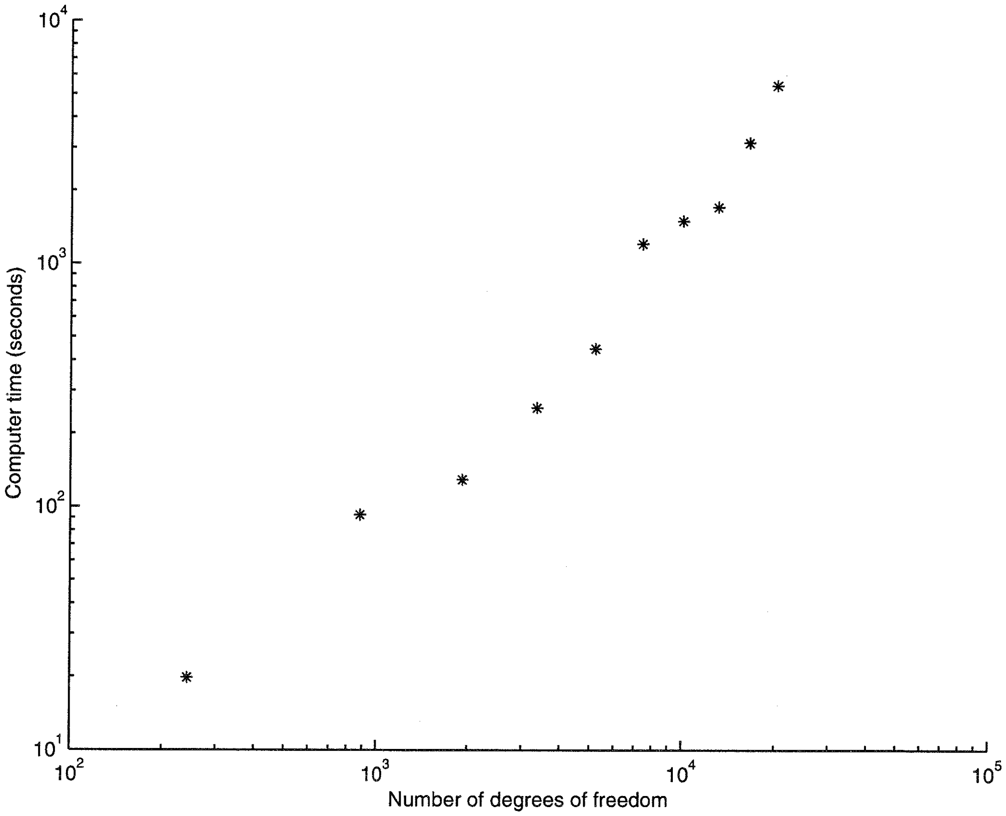

Fig. 15. Scatter plot for up to 20,402 degrees of freedom for 20 random variables in KL expansion (log scale).

noted here that parallelization of computation of mean and weighted stiffness matrices can make the whole process many times faster. Further, optimizing the file read/write process is also expected to significantly improve speed.

6. Concluding remarks

Over the past three decades, a significant amount of effort has gone into developing and testing deterministic FE software. However, general-purpose software tools for stochastic analysis are still at its very beginning. This paper presented a stepping stone towards developing a general-purpose software for computational stochastic mechanics. The approach presented exploits existing FEA tools for carrying out spatial discretization of stochastic PDEs and has a number of attractive features: (1) users familiar with FEA and a basic understanding of probability theory will be able to use stochastic projection schemes, (2) direct access to the extensive library of existing finite element formulations for plate and shell structures, three-dimensional elasticity, heat transfer, etc., becomes possible, and (3) future extensions of the framework, for example to non-linear problems and non-probabilistic uncertainty models, can be readily implemented.

In this paper we presented strategies for interfacing stochastic reduced basis projection schemes with third-party deterministic finite element codes. We outlined a brief introduction to stochastic projections schemes and addressed the issues involved while using them in conjunction with deterministic FE software. We demonstrated the usefulness and capabilities of such a coupling with the help of a number of problems from stochastic structural mechanics. However, the coupling presented works only for linear static problems with uncertainties in material properties. Further work is required to extend the present framework to tackle geometric uncertainties.

One of the major challenges in developing such a software is the choice of solution method for semi-discretized system of equations. Our emphasis is on using stochastic reduced basis projection schemes for solving real life engineering problems since they have a rigorous mathematical underpinning and are computationally efficient at the same time. We presented detailed timing studies to demonstrate the usefulness of a typical coupled framework. The results obtained suggest that linear stochastic systems with many thousands of degrees of freedom and tens of random variables can be solved using a tractable amount of computational resources.

Acknowledgement

This research was supported by a grant from the School of Engineering Sciences at the University of Southampton.

References

[1] Thacker BH, Riha DS, Fitch SHK, Huyse LJ, Pleming JB. Probabilistic engineering analysis using the nessus software. Struct Safety 2006;28:83–107.

[2] Schueller GI, Pradlwarter HJ. COSSAN (computational stochastic structural analysis) – stand-alone toolbox, user's manual. Institute of Engineering Mechanics, Leopold-Franzens University, Innsbruck, Austria, 1996–2005.

[3] Liu P-L, Kiureghian AD. Optimization algorithms for structural reliability analysis. Tech. Rep. UCB/BSEM-86/09, Division of Structural Engineering and Structural Mechanics, Department of Civil Engineering, University of California, Berkeley (CA), 1989.

[4] Sudret T, Kiureghian AD. Finite element reliability and sensitivity methods for performance-based earthquake engineering. Tech. Rep. 2003/14, Pacific Earthquake Engineering Center, University of California, Berkeley (CA), 2004.

[5] Kiureghian AD, Haukaas T, Fajimura K. Structural reliability software at University of California, Berkley. Struct Safety 2006;2006:44–67.

[6] Sudret B, Kiureghian AD. Stochastic finite elements and reliability: a state-of-the-art report. Tech. Rep. UCB/SEMM-2000/08, University of California, Berkeley, 2000.

[7] Reh S, Beley J-D, Mukherjee S, Khor EH. Probabilistic finite element analysis using ANSYS. Struct Safety 2006;28:17–43.

[8] Pellissetti MF, Schueller GI. On general purpose software in structural reliability – an overview. Struct Safety 2006;28:3–16.

[9] Keese A. Numerical solution of systems with stochastic uncertainties – a general purpose framework for stochastic finite elements. PhD thesis, TU Braunschweig, Germany, Fachbereich Mathematik and Informatik, 2004.

[10] Yongke Y. Coupling a stochastic finite element solver with ANSYS and visualization of the results. Master's thesis, Institute of scientific computing, Technische Universitat, Braunschweig, Germany, June 2003.

[11] Nair PB. On the theoretical foundations of stochastic reduced basis methods. AIAA Paper 2001-1677.

[12] Nair PB, Keane AJ. Stochastic reduced basis methods. AIAA J 2002;40:1653–64.

[13] Nair PB. Projection schemes in stochastic finite element analysis. In: Nikolaidis E, Ghiocel DM, editors. CRC engineering design reliability handbook. Boca Raton, FL: CRC Press; 2004 [chapter 21].

[14] Sachdeva SK, Nair PB, Keane AJ. Comparative study of projection schemes for stochastic finite element analysis. Comput Methods Appl Mech Eng 2006;195:2371–92.

[15] Sachdeva SK, Nair PB, Keane AJ. Hybridization of stochastic reduced basis methods with polynomial chaos expansions. Probab Eng Mech 2006;121:182–92.

[16] Vanmarcke E. Random fields: analysis and synthesis. London, England: MIT Press; 1988.

[17] Huyse L, Maes MA. Random field modeling of elastic properties using homogenization. J Eng Mech 2001;127:27–36.

[18] Li CC, Kiureghian AD. Optimal discretization of random fields. J Eng Mech, ASCE 1993;119:1136–54.

[19] Huang SP, Quek ST, Phoon KK. Convergence study of the truncated Karhunen–Loeve expansion for simulation of stochastic processes. Int J Numer Methods Eng 2001;52:1029–43.

[20] Press WH, Flannery BP, Teukolsky SA, Vetterling WT. Numerical recipes in C: the art of scientific computing. Cambridge University Press; 1992.

[21] Zienkiewicz OC, Taylor RL. Finite element method, vol. 1. London: Butterworth Heinemann; 2000.

[22] MATLAB The language of technical computing, version 7. Natick, MA: The MathWorks, Inc.; 1984–2002.

[23] Persson P-O, Strang G. A simple mesh generator in Matlab. SIAM Rev 2004;46(2):329–45.

[24] Izenman AJ. Recent developments in nonparametric density estimation. J Am Statist Assoc 1991;86:205–24.

[25] Ghanem R, Spanos P. Stochastic finite elements: a spectral approach. Springer-Verlag; 1991.

[26] Sachdeva SK. Subspace projection schemes for stochastic finite element analysis. PhD thesis, Faculty of Engineering, Science and Mathematics, University of Southampton, UK, 2006.

[27] Papadrakakis M, Charmpis DC. Improving the computational efficiency in finite element analysis of shells with uncertain properties. Comput Methods Appl Mech Eng 2005;194:1447–78.

[28] Kiureghian AD, Ke J-B. The stochastic finite element method in structural reliability. Probab Eng Mech 1988;3(2):83–91.