



EIGENFUNCTION EXPANSION BASED GALERKIN APPROACHES FOR STOCHASTIC FINITE ELEMENT ANALYSIS

S. E. Pryse* and S. Adhikari*

College of Engineering
Swansea University

Bay Campus, Fabian Way, Swansea, Wales, SA1 8EN
e-mail: s.adhikari@swansea.ac.uk

Key words: Stochastic differential equations, Error minimisation, Stochastic finite element method, eigendecomposition, Galerkin, Reduced methods, Eigenfunctions

Abstract. *A novel approach is suggested to compute the response of discretized stochastic elliptic partial differential equations by utilising the stochastic finite element analysis method. The mathematical form of the approach is established by projecting random scalars onto a random basis. The method is based upon an eigendecomposition of a system's stiffness matrix. Computational reduction is achieved by approximating the random eigensolutions, and by only including dominant terms. Two novel error minimisation techniques have been proposed in order to lower the error introduced by to the approximations and the truncation: i) A weak Galerkin approach, ii) A strong Galerkin approach. These have been applied through introducing unknown multiplicative scalars into the expression of the response. It can be shown that the unknown scalars for both approaches can be expressed in a closed-form expression. The proposed methods are applied to analyse the bending of a static stochastic cantilever beam. The results obtained through the proposed approaches are compared with those obtained by using direct Monte Carlo Simulations and by using polynomial chaos.*

1 INTRODUCTION

1.1 Background

Uncertainties can substantially affect the analysis of physical structures. These uncertainties can occur in the properties of the material, in the geometry or boundary conditions of the structure or in the applied loads [1]. In order to represent the uncertainties that occur in physical systems, a stochastic finite element method [SFEM] can be applied. In this work, a stochastic elliptic partial differential equation is considered

$$-\nabla [a(\mathbf{x}, \omega) \nabla \mathbf{u}(\mathbf{x}, \omega)] = p(\mathbf{x}) \quad \mathbf{x} \text{ in } \mathcal{D} \quad (1)$$

with the associated Dirichlet condition

$$\mathbf{u}(\mathbf{x}, \omega) = 0 \quad \mathbf{x} \text{ in } \mathcal{D} \quad (2)$$

The spatial dimension under consideration is a bounded domain $\mathcal{D} \in \mathbb{R}^d$ with piecewise

Lipschitz boundary $\partial\mathcal{D}$, where d is less than four. (Ω, \mathcal{F}, B) is a probability space where $\omega \in \Omega$ is a sample point from the sampling space Ω , \mathcal{F} is the complete σ -algebra over the subsets of Ω and B is the probability measure. In Equation (1) $a : \mathbb{R}^d \times \Omega \rightarrow \mathbb{R}$ is a random field [2], which can be viewed as a set of random variables indexed by $\mathbf{x} \in \mathbb{R}^d$. We assume the random field $a(\mathbf{x}, \omega)$ to be stationary and square integrable. $a(\mathbf{x}, \omega)$ is also able to model different physical quantities. Following the discretization of Equation (1) through the SFEM [3], this work aims to produce a new solution approach in order to solve the discretised equivalent of Equation (1).

1.2 Discretisation of the stochastic partial differential equation

The random process $a(\mathbf{x}, \omega)$ seen in Equation (1) can be expanded by a generalised Fourier expansion known as the Karhunen-Loève expansion

$$a(\mathbf{x}, \omega) = a_0(\mathbf{x}) + \sum_{i=1}^{\infty} \sqrt{\tilde{\lambda}_i} \tilde{\xi}_i(\omega) \tilde{\boldsymbol{\phi}}_i(\mathbf{x}) \quad (3)$$

Here a_0 is the mean function, and $\tilde{\lambda}_i$ and $\tilde{\boldsymbol{\phi}}_i$ the eigenvalues and eigenvectors satisfying the integral equation

$$\int_D C_a(\mathbf{x}_1, \mathbf{x}_2) \tilde{\boldsymbol{\phi}}_j(\mathbf{x}_1) d\mathbf{x}_1 = \tilde{\lambda}_j \tilde{\boldsymbol{\phi}}_j(\mathbf{x}_2) \quad \forall j = 1, 2, \dots \quad (4)$$

We will consider $\tilde{\xi}_i(\omega)$ to be uncorrelated standard Gaussian random variables. After truncating the series seen in Equation (3) to the M th term, the resulting equation can be substituted into the original stochastic elliptical partial differential equation. By applying appropriate boundary conditions, the discretized equation takes the form

$$\left[\mathbf{A}_0 + \sum_{i=1}^M \xi_i(\omega) \mathbf{A}_i \right] \mathbf{u}(\omega) = \mathbf{f} \quad (5)$$

where $\mathbf{A}_0 \in \mathbb{R}^{n \times n}$ represents a deterministic, positive definite, symmetric matrix. $\mathbf{A}_i \in \mathbb{R}^{n \times n}$ are random symmetric matrices for $i = 1, 2, \dots, M$, $\mathbf{u}(\omega) \in \mathbb{R}^n$ the response vector and $\mathbf{f} \in \mathbb{R}^n$ the deterministic input force vector. The details of obtaining the discretized equivalent of Equation (1) has been omitted, but can be located in numerous textbooks including [3].

2 MOVIVATION

For simplicity, we can express Equation (5) as

$$\mathbf{A}(\omega) \mathbf{u}(\omega) = \mathbf{f} \quad (6)$$

where the random matrix $\mathbf{A}(\omega) = \mathbf{A}_0 + \sum_{i=1}^M \xi_i(\omega) \mathbf{A}_i$. The exact solution to the set of stochastic linear equations given above can be obtained through a direct Monte Carlo Simulations [MCS] approach by solving the following expression for each sample.

$$\mathbf{u}(\omega) = \mathbf{A}(\omega)^{-1} \mathbf{f} \quad (7)$$

Convergence is guaranteed if the number of realizations is sufficiently large. However, direct MCS can be seen as a computationally expensive method [4], especially if there is a large number of stochastic linear equations to be solved. In order to avoid the use of direct MCS,

alternative methods have been explored. The response of Equation (6) can be represented through summing products of random scalars and deterministic vectors

$$\mathbf{u}(\omega) = \sum_{j=1}^{M_1} a_j(\omega) \mathbf{g}_j \quad (8)$$

Equation (8) can be considered as the polynomial chaos method

$$\mathbf{u}(\omega) = \sum_{k=1}^P H_k(\xi(\omega)) \mathbf{u}_k \quad (9)$$

where $H_k(\xi(\omega))$ represents the polynomial chaoses (corresponding to the random scalars), and \mathbf{u}_k represents unknown deterministic vectors that need to be determined. The value of P is determined by a basic random variable M and by the order of the Polynomial Chaos expansion p . In this instance, M corresponds to the order of the Karhunen-Loève expansion

$$P = \sum_{j=0}^p \frac{(M+j-1)!}{j!(M-1)!} \quad (10)$$

It is evident that P increases rapidly when either the order of the Karhunen-Loève expansion or the order of the Polynomial Chaos expansion is increased. The unknown vector \mathbf{u}_k can be obtained by using a Galerkin error minimising approach [3]. This approach leads to a system of linear equations of size $nP \times nP$. A possible drawback to this approach is the high computational cost if either n or P is sufficiently large.

Due to the vector in Equations (8) being deterministic, we aim to acquire a solution where both the scalars and vectors are random.

$$\mathbf{u}(\omega) = \sum_{j=1}^{M_2} a_j(\omega) \mathbf{g}_j(\omega) \quad (11)$$

We aim to see if Equation (11) can incorporate the full stochastic nature of Equation (6). Therefore the aim of this paper is to obtain an expression for the response of Equation (6) that is of the same form as Equation (11).

3 INTRODUCING THE METHOD

In order to implement our aim, the random eigenvalue problem is initially considered

$$\mathbf{A}(\omega) \boldsymbol{\phi}_k(\omega) = \lambda_k(\omega) \boldsymbol{\phi}_k(\omega); \quad k = 1, 2, \dots, n \quad (12)$$

For convenience, the matrices of the random eigenvalues and eigenvectors of $\mathbf{A}(\omega)$ are defined as follows

$$\boldsymbol{\Lambda}(\omega) = \text{diag} [\lambda_1(\omega), \lambda_2(\omega), \dots, \lambda_n(\omega)] \in \mathbb{R}^{n \times n} \quad (13)$$

$$\boldsymbol{\Phi}(\omega) = [\boldsymbol{\phi}_1(\omega), \boldsymbol{\phi}_2(\omega), \dots, \boldsymbol{\phi}_n(\omega)] \in \mathbb{R}^{n \times n} \quad (14)$$

The random eigenvalues are arranged in ascending order so $\lambda_1(\omega) < \lambda_2(\omega) < \dots < \lambda_n(\omega)$. The corresponding eigenvectors are arranged in the same order. Due to the orthogonality of $\boldsymbol{\Phi}(\omega)$, we can deduce that $\boldsymbol{\Phi}(\omega)^{-1} = \boldsymbol{\Phi}(\omega)^T$. Thus the following identities

can be defined (ω has been omitted for notational convenience)

$$\Phi^T A \Phi = \Lambda; \quad A = \Phi^{-T} \Lambda \Phi^{-1} \quad \text{and} \quad A^{-1} = \Phi \Lambda^{-1} \Phi^T \quad (15)$$

Using these identities, the response of Equation (6) can be expressed as

$$\mathbf{u}(\omega) = \Phi \Lambda^{-1} \Phi^T \mathbf{f} = \sum_{j=1}^n \frac{\phi_j^T(\omega) \mathbf{f}}{\lambda_j(\omega)} \phi_j(\omega) \quad (16)$$

Equation (16) is of the same form as Equation (11) where $\frac{\phi_j^T(\omega) \mathbf{f}}{\lambda_j(\omega)}$ corresponds to the scalar term $a_j(\omega)$, $\phi_j(\omega)$ corresponds to the vector term $\mathbf{g}_j(\omega)$ and n corresponds to M_2 . In this particular method, $\phi_j(\omega)$ forms a complete orthogonal basis. Therefore, it can be concluded that the response of Equation (6) can be expressed in the same form as Equation (11) where we have random scalars projected onto a stochastic basis.

4 COMPUTATIONAL REDUCTION

4.1 Approximating the random eigenvalues and eigenvectors

Approximating the random eigenvalues and eigenvectors may improve the calculation cost and there are numerous methods of doing so. Direct MCS can be used in collaboration with the random eigenvalue problem in order to calculate the exact values of the random eigenvalues and eigenvectors; however this method is computationally expensive. Numerous papers have proposed improvements to the direct MCS method. [5] uses a subspace iteration scheme with carefully selected start-vectors, whilst [6] compares the subspace iteration method with an approach that uses component mode synthesis. [7] proposes a method of obtaining the random eigenvalues and random eigenvectors through expanding the random eigenvalues and eigenvectors by a polynomial chaos approach. However this method is computationally expensive. Due to its low computational cost, a first order perturbation approach for obtaining the random eigenvalues and eigenvectors has been explored.

Solutions of different perturbation methods are obtained by truncating the Taylor series expansion. Due to its efficiency and ease, the first order perturbation method has been used. The j th random eigenvalue and its corresponding random eigenvector is given by

$$\lambda_j \approx \lambda_{j_0} + \sum_{k=1}^M \left(\frac{\partial \lambda_j}{\partial \xi_k} \right) \xi_k(\omega) \quad (17)$$

$$\phi_j \approx \phi_{j_0} + \sum_{k=1}^M \left(\frac{\partial \phi_j}{\partial \xi_k} \right) \xi_k(\omega) \quad (18)$$

where $\xi_k(\omega)$ is a set of Gaussian random variables with mean zero and unit variance. By differentiating the eigenvalue equation with respect to ξ_k , pre-multiplying with ϕ_j^T and utilising that $\phi_j^T \phi_j = 1$, $\frac{\partial \lambda_j}{\partial \xi_k}$ can be expressed as

$$\frac{\partial \lambda_j}{\partial \xi_k} = \phi_{j_0}^T \frac{\partial A}{\partial \xi_k} \phi_{j_0} \quad (19)$$

In the instance of Equation (19), $\frac{\partial A}{\partial \xi_k} = \mathbf{A}_k$.

The derivative of $\boldsymbol{\phi}_j$ with respect to ξ_k can be calculated by expanding $\frac{\partial \boldsymbol{\phi}_j}{\partial \xi_k}$ as a linear combination of the deterministic eigenvalues and eigenvectors [8] [9]

$$\frac{\partial \boldsymbol{\phi}_j}{\partial \xi_k} = \sum_{i=1, i \neq j}^N \alpha_{jki} \boldsymbol{\phi}_k \quad \text{where} \quad \alpha_{jki} = \frac{\boldsymbol{\phi}_{i_0}^T \frac{\partial \mathbf{A}}{\partial \xi_k} \boldsymbol{\phi}_{j_0}}{\lambda_{j_0} - \lambda_{i_0}} \quad (20)$$

In this instance, $\frac{\partial A}{\partial \xi_k} = \mathbf{A}_k$. This method requires all the deterministic eigenvalues and eigenvectors to be known. However care is needed with this approach as all the eigenvalues need to be unique and the coefficient of variation needs to be of a moderate value [10]. A simplified approach is proposed in [11] where only a limited number of eigenvalues and eigenvectors is needed. For the case of repeated eigenvalues, the general method of approximating the response of Equation (6) continues to be valid. However a different approach would be needed whilst approximating the eigensolutions to that given in this section. This case is beyond the scope of this paper.

Other approaches based on the perturbation method have been proposed to approximate random eigensolutions. An approach based on the perturbation method and the Rayleigh quotient is presented in [12]; this method results in an improvement in the accuracy of the eigensolution. However [13] reports that the accuracy of the approximations obtained by the first order perturbation method and the method presented in [12] deteriorates if the uncertainty in the system is sufficiently large. [13] proposes methods to suffice this problem with one being the Padé approximation which is seen in [14].

4.2 Truncation

The series given in Equation (16) can be truncated after a certain amount of terms. The high terms of the summation have a relatively low value due to the eigenvalues being ordered ascendingly; this allows the low valued terms to be discarded whilst retaining the dominant terms in the series. The number of terms retained in Equation (16) can either be predefined or determined by

$$\frac{\lambda_{1_0}}{\lambda_{t_0}} < \epsilon_{trunc} \quad (21)$$

where λ_{1_0} corresponds to the first, and therefore the smallest deterministic eigenvalue and λ_{t_0} to the t th deterministic eigenvalue. The value of t would then correspond to the number of terms to be retained in the truncation. The value ϵ_{trunc} is to be selected appropriately. Hence Equation (16) can be truncated as follows

$$\mathbf{u}(\omega) \approx \sum_{j=1}^t \frac{\boldsymbol{\phi}_j^T(\omega) \mathbf{f}}{\lambda_j(\omega)} \boldsymbol{\phi}_j(\omega) \quad (22)$$

where $\lambda_j(\omega)$ and $\boldsymbol{\phi}_j(\omega)$ represent the random eigenvalues and the random eigenvectors. A low-cost MCS is performed in order to obtain the full response of $\mathbf{u}(\omega)$.

5 ERROR MINIMISATION

In Section 3, an approximation for the response vector has been proposed where random scalars are projected onto a stochastic basis. We have also shown that it's possible to approximate the random eigensolutions in Section 4.1. However these approximations in addition to the truncation introduced in Section 4.2 introduces error into the calculation. This has motivated an error minimisation approach, and as a consequence, two Galerkin approaches have been considered:

- A weak Galerkin approach [SPWG]
- A strong Galerkin approach [SPSG]

The SPWG approach is initially considered.

5.1 A weak Galerkin approach [SPWG]

For this approach, the solution vector is modified to take the following form

$$\tilde{\mathbf{u}}(\omega) \approx \sum_{j=1}^t c_j \left(\frac{\boldsymbol{\phi}_j^T(\omega) \mathbf{f}}{\lambda_j(\omega)} \right) \boldsymbol{\phi}_j(\omega) \quad (23)$$

where $\lambda_j(\omega) \in \mathbb{R}$ and $\boldsymbol{\phi}_j(\omega) \in \mathbb{R}^{N \times N}$ represent the random eigenvalues and eigenvectors, $\mathbf{f} \in \mathbb{R}^{N \times N}$ the deterministic applied force and $c_j \in \mathbb{R}^t$ are deterministic constants which need to be determined. The residual vector for this the new approach is defined as

$$\mathbf{r}_1(\omega) = \mathbf{A}(\omega)\tilde{\mathbf{u}}(\omega) - \mathbf{f} \quad (24)$$

By making the residual orthogonal to a basis function, the deterministic scalars c_j can be computed. As Equation (23) can be viewed as a projection onto a subset of the random eigenvectors, the residual can be made orthogonal to the same subset of random eigenvectors.

$$\mathbf{r}_1(\omega) \perp \boldsymbol{\phi}_k(\omega) \quad \forall k = 1, 2, \dots, t \quad (25)$$

Here $\langle v(\omega), w(\omega) \rangle = \int_{\omega} v(\omega)w(\omega)P(d\omega)$ defines the inner product norm. By using this condition and the expression for the residual, one has

$$\mathbb{E} \left\{ \boldsymbol{\phi}_k^T(\omega) \left[\mathbf{A}(\omega) \left(\sum_{j=1}^t c_j \left(\frac{\boldsymbol{\phi}_j^T(\omega) \mathbf{f}}{\lambda_j(\omega)} \right) \boldsymbol{\phi}_j(\omega) \right) - \mathbf{f} \right] \right\} = 0 \quad (26)$$

where $\mathbb{E}\{\blacksquare\}$ donates the expected value. For notational convenience, we can define $\beta_j(\omega) = \left(\frac{\boldsymbol{\phi}_j^T(\omega) \mathbf{f}}{\lambda_j(\omega)} \right)$, thus it can be shown that Equation (26) can take the following form

$$\mathbb{E} \left\{ \sum_{j=1}^t \boldsymbol{\phi}_k^T(\omega) \mathbf{A}(\omega) \boldsymbol{\phi}_j(\omega) \beta_j(\omega) c_j \right\} = \mathbb{E}\{\boldsymbol{\phi}_k(\omega)\} \mathbf{f} \quad (27)$$

By defining the vector $\mathbf{c} = [c_1, c_2, \dots, c_t]^T$, Equation (27) can be simplified to

$$\mathbb{E}\{\mathbf{Z}(\omega)\} \mathbf{c} = \mathbb{E}\{\mathbf{y}(\omega)\} \quad j, k = 1, 2, \dots, t \quad (28)$$

where $\mathbf{Z}_{kj} = \boldsymbol{\phi}_k^T(\omega) \mathbf{A}(\omega) \boldsymbol{\phi}_j(\omega) \beta_j(\omega); \quad \forall j, k = 1, 2, \dots, t,$ and $\mathbf{y}(\omega) = \boldsymbol{\phi}_k(\omega) \mathbf{f}; \quad \forall k = 1, 2, \dots, t.$ Therefore by solving the set of linear equations given by Equation (28), an explicit closed form for the unknown coefficients can be obtained. The arising expected values can be computed by using low-order fast Monte Carlo Simulations.

5.2 A strong Galerkin approach [SPSG]

In a similar manner to the previous approach, the solution vector can be modified to take the following form

$$\dot{\mathbf{u}}(\omega) \approx \sum_{j=1}^t d_j(\omega) \left(\frac{\boldsymbol{\phi}_j^T(\omega) \mathbf{f}}{\lambda_j(\omega)} \right) \boldsymbol{\phi}_j(\omega) \quad (29)$$

where $\lambda_j(\omega) \in \mathbb{R}$ and $\boldsymbol{\phi}_j(\omega) \in \mathbb{R}^{N \times N}$ represent the random eigenvalues and eigenvectors, $\mathbf{f} \in \mathbb{R}^{N \times N}$ the deterministic applied force. Contrary to the previous approach, $d_j(\omega) \in \mathbb{R}$ are unknown constants that need to be computed for each realisation. The residual vector for this approach is defined as

$$\mathbf{r}_2(\omega) = \mathbf{A}(\omega) \dot{\mathbf{u}}(\omega) - \mathbf{f} \quad (30)$$

By making the residual orthogonal to a basis function, $d_j(\omega)$ can be computed. By using the same analogy as seen in the previous approach, the residual can be made orthogonal to a subset of random eigenvectors.

$$\mathbf{r}_2(\omega) \perp \boldsymbol{\phi}_k(\omega) \quad \forall k = 1, 2, \dots, t \quad (31)$$

Thus resulting in the following expression

$$\boldsymbol{\phi}_k^T(\omega) \left[\mathbf{A}(\omega) \left(\sum_{j=1}^t d_j(\omega) \left(\frac{\boldsymbol{\phi}_j^T(\omega) \mathbf{f}}{\lambda_j(\omega)} \right) \boldsymbol{\phi}_j(\omega) \right) - \mathbf{f} \right] = 0 \quad (32)$$

For notational convenience, we can define $\beta_j(\omega) = \left(\frac{\boldsymbol{\phi}_j^T(\omega) \mathbf{f}}{\lambda_j(\omega)} \right).$ Therefore, the above expression can be manipulated to give

$$\left(\sum_{j=1}^t \boldsymbol{\phi}_k^T(\omega) \mathbf{A}(\omega) \boldsymbol{\phi}_j(\omega) \beta_j(\omega) d_j(\omega) \right) = \boldsymbol{\phi}_k(\omega) \mathbf{f} \quad (33)$$

where $d_j(\omega)$ would be computed for each realisation. By defining the vector $\mathbf{d}(\omega) = [d_1(\omega), d_2(\omega), \dots, d_t(\omega)]^T,$ Equation (23) can be simplified to

$$\mathbf{Z}(\omega) \mathbf{d}(\omega) = \mathbf{y}(\omega) \quad j, k = 1, 2, \dots, t \quad (34)$$

where $\mathbf{Z}_{kj} = \boldsymbol{\phi}_k^T(\omega) \mathbf{A}(\omega) \boldsymbol{\phi}_j(\omega) \beta_j(\omega); \quad \forall j, k = 1, 2, \dots, t,$ and $\mathbf{y}(\omega) = \boldsymbol{\phi}_k(\omega) \mathbf{f}; \quad \forall k = 1, 2, \dots, t.$ The number of equations that need to be solved in order to calculate the unknown vector $\mathbf{d}(\omega)$ corresponds to the value of $t.$ Therefore, similarly to the weak Galerkin approach, the lower the dimension of the reduced system, the fewer the number of equations that need to be solved.

5.3 A summary of the different methods

Thus far we have discussed four different methods for computing and approximating the response of Equation (6):

- Directly solving in order to compute the exact solution (Equation (7)) [DMCS]
- Approximating the solution by projecting random scalars onto a stochastic basis (Equation (16)) [SP]
- Approximating the solution by projecting a random scalar onto a stochastic basis (including a weak Galerkin error minimisation approach) (Equation (23)) [SPWG]
- Approximating the solution by projecting a random scalar onto a stochastic basis (including a strong Galerkin error minimisation approach) (Equation (29)) [SPSG]
- Approximating the solution by polynomial chaos (Equation (9)) [PC]

In the subsequent section, these five methods are applied to a static Euler-Bernoulli cantilever beam. The effectiveness and efficiency of the approximation methods are scrutinised.

6 APPLICATION

The computational method has been applied to an Euler-Bernoulli cantilever beam clamped at one end i.e. the displacement at the clamped end is zero. A deterministic vertical point load is applied at the free end of the beam, where $P = 1.00$ N. The length of the beam under consideration is 1.00 m, and its cross-section is a rectangle of length 0.04 m and height 0.005 m. Figure 1 illustrates the system.

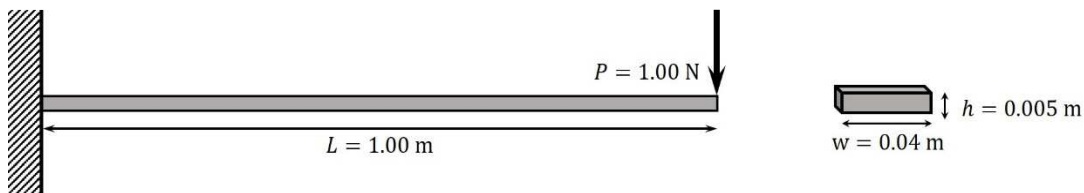


Figure 1: The configuration of the cantilever beam

The system has been discretized into a 100 elements by using SFEM. Details of the discretization can be found in numerous books including [16]. Consequently, the dimension of the corresponding determinant matrix is 200×200 . For the deterministic case, the Young's modulus is $E = 200 \times 10^9$ Nm⁻² thus corresponding to a steel beam. The deterministic second moment of area (moment of inertia) of the beam is 4.1667×10^{-11} m⁴.

$$I = \frac{0.04 \times 0.005^3}{12} = 4.1667 \times 10^{-11} \text{ m}^4 \quad (35)$$

The bending rigidity of the beam, EI , can be assumed to be a stationary Gaussian random field of the form

$$EI(x, \omega) = \overline{EI} (1 + a(x, \omega)) \quad (36)$$

where $\overline{EI} = 83.33 \text{ Nm}^2$. The function $a(x, \omega)$ represents a stationary Gaussian field with zero mean, with x being the coordinate direction of the length of the beam. The standard deviation is given by $0.18 \times \overline{EI}$, and the covariance function by

$$C_a(x_1, x_2) = \sigma_a^2 e^{(|x_1 - x_2|) / \mu_a} \quad (37)$$

where μ_a is the correlation length and σ_a the standard deviation. In this instance, the correlation length is set as 0.50m. The KL expansion of the stiffness matrix, given by Equation (3), has been truncated and two terms have been kept.

All the methods listed at the conclusion of Section 5 have been simulated 10,000 times and the performances of the approximation methods compared with that of the DMCS approach. For the SP, SPWG and SPSG methods, Equations (16), (23) and (29) have been truncated to only include the first 4 terms.

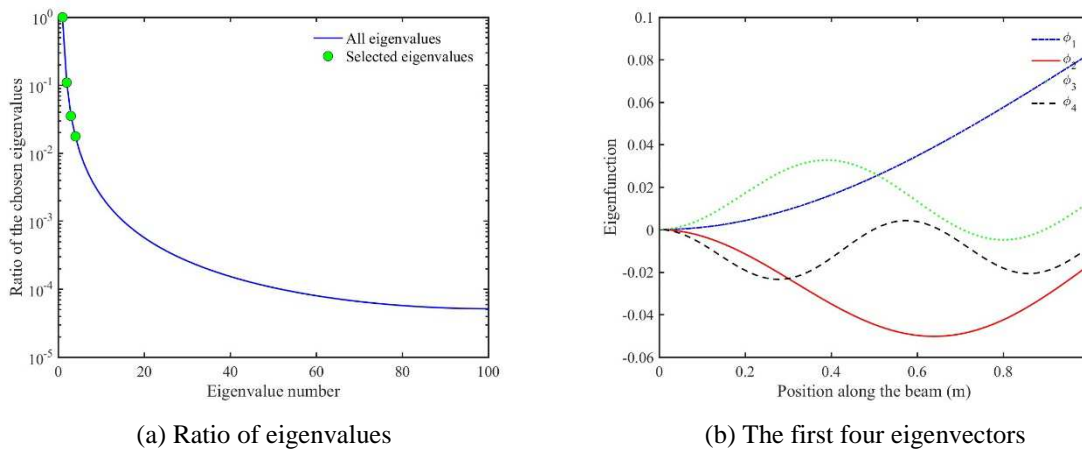


Figure 2: Deterministic eigensolutions of the cantilever beam

Figure 2a illustrates the ratio between the first and the j th eigenvalue of the deterministic system and the first four eigenvalues have been highlighted. Figure 2b illustrates the first four eigenvectors of the deterministic system.

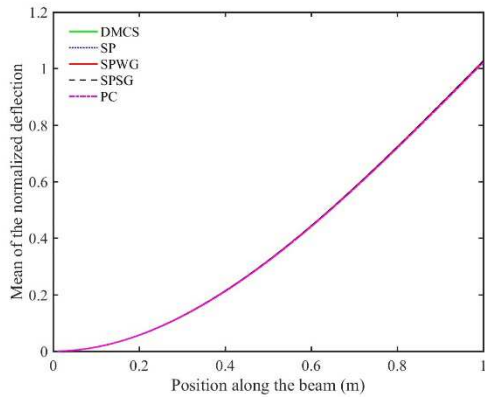
The displacement of the beam can be normalised by

$$h = \frac{f L^3}{3 E I} \quad (38)$$

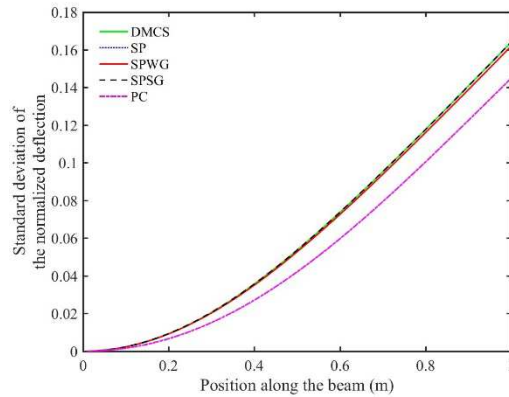
A detailed reference to this value is give in [16]. This normalisation value ensures that the deterministic vertical displacement has a value of 1 at the tip. Figures 3a and 3b illustrate the mean and standard deviation of the normalised vertical displacement at all nodes of the beam. It is apparent that all the approximation methods captures the mean of the DMCS method pretty well. The SPSG method best captures the standard deviation of the DMCS method, whilst both the SP and SPWG methods considerably out performs the PC approach. The percentage error of the vertical displacement arising when using the SP, SPWG, SPSG and the PC methods in place of the DMCS method is illustrated in Figures 3c and 3d. The percentage error is given by

$$\epsilon_{\%} = 100 \times \frac{|DMCS - CM|}{DMCS} \quad (39)$$

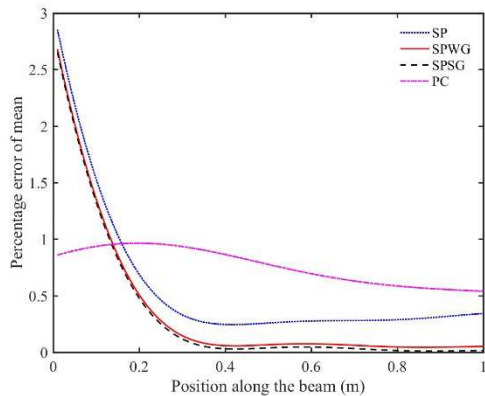
where *DMCS* indicates the solution of the direct approach, and *CM* the solution of the comparable methods. Barring the initial 0.20m of the bar, the percentage error of the mean and standard deviation of the SP, SPWG and SPSG methods are considerably lower than that of the PC method. When comparing the SPWG and SPSG methods it is apparent that the percentage error of the mean is always slightly lower when the SPSG method is used. However when comparing the percentage error of the standard deviation, barring the initial 0.30m of the



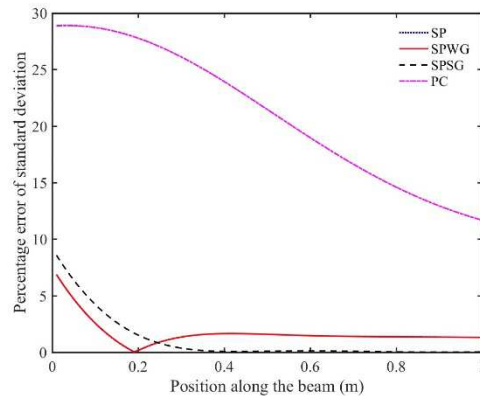
(a) Mean of the vertical displacement



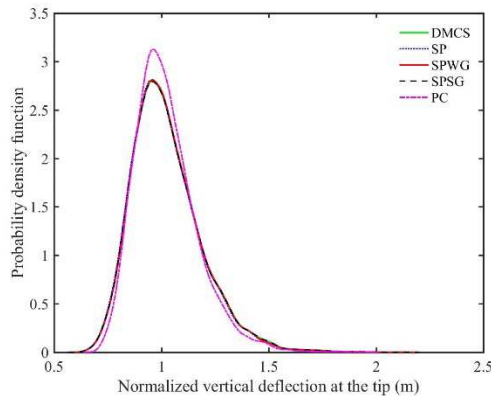
(b) Standard deviation of the vertical displacement



(c) Percentage error of the mean



(d) Percentage error of the standard deviation



(e) Probability density function of the vertical displacement at the tip of the cantilever beam

the SPSG method considerably outperforms the SPWG method. The pdf of the vertical displacement at the tip for the DMCS, SP, SPWG, SPSG and PC methods is illustrated in Figure 3e.

The L^2 relative error for the mean and standard deviation of the response of the cantilever beam is considered. The L^2 relative error of the mean is defined as

$$\epsilon_{L^2} = \frac{\|\boldsymbol{\mu}_{DMCS} - \boldsymbol{\mu}_{CM}\|_2}{\|\boldsymbol{\mu}_{DMCS}\|_2} \quad (40)$$

where $\boldsymbol{\mu}_{DMCS}$ denotes the mean of the response vector obtained by using the DMCS method and $\boldsymbol{\mu}_{CM}$ the mean of the response vector of the comparable methods. Table 1 displays the L^2 relative error for the mean. The expression for the L^2 relative error of the standard deviation takes a similar form to that of Equation (4). The L^2 relative error for the standard deviation for different truncation values of Equations (16), (23) and (29) and the PC approach is displayed in Table 2.

Number of terms	SP	SPWG	SPSG	PC
1	0.0226	0.0226	0.0226	-
2	0.0063	0.0058	0.0058	-
3	0.0031	0.0019	0.0019	-
4	0.0036	9.59×10^{-4}	9.07×10^{-4}	-
5	0.0025	6.37×10^{-4}	5.54×10^{-4}	-
6	0.0025	4.56×10^{-4}	3.30×10^{-4}	-
10	0.0025	3.28×10^{-5}	9.54×10^{-5}	-
-	-	-	-	0.0017

Table 1: L^2 error arising in the mean of the response for each of the reduced methods

Number of terms	SP	SPWG	SPSG	PC
1	0.0426	0.0426	0.0426	-
2	0.0092	0.0092	0.0029	-
3	0.0099	0.0099	0.0016	-
4	0.0101	0.0101	0.0010	-
5	0.0102	0.0101	5.14×10^{-4}	-
6	0.0102	0.0102	3.56×10^{-4}	-
10	0.0102	0.0102	9.88×10^{-5}	-
-	-	-	-	0.0635

Table 2: L^2 error arising in the standard deviation of the response for each of the reduced methods

Except for when only one term is used in the summations, in comparison to the SP method the SPWG and SPSG methods lower the L^2 relative error for the mean. When considering the L^2 relative error for the standard deviation, the SPWG method does not seem to significantly lower the error. However barring when only one term is kept in each of the summations, a substantial drop in the L^2 relative error is seen when the SPSG method is used. As the L^2 relative errors tend to decrease as the number of terms in the summations increase, it is

apparent that the truncation value can be chosen in line with an acceptable level of error.

Table 3 contains the CPU times for the five methods attempted. The computational cost was calculated on a standard 24 GB RAM computer with a 3.60 GHz processor.

Method	DMCS	SP	SPWG	SPSG	PC
CPU (sec)	10.21	0.76	3.25	3.24	0.84

Table 3: Computational time

Regarding CPU, the SP method outperforms the other approximation methods. Although that using the SPWG and SPSG methods is faster than the DMCS method, the PC method is faster. However when the order of the PC method or the number of degrees of freedom is sufficiently increased, the CPU of the PC method will substantially increase and in turn, will be higher than the SPWG and SPSG methods. Generally it can be concluded that the SPSG method outperforms the SPWG method as it produces less error in a similar CPU time. However the SPWG method does not need as much storage capacity as the SPSG method. For the SPWG method only t scalars need to be stored, whilst the SPSG method requires $(t \times \text{number of samples})$ scalars to be stored.

7 CONCLUSION

An approach has been suggested to calculate the response of discretized stochastic elliptic partial differential equations. Through utilising the stochastic finite element method and the random eigenvalue problem, it has been proven that the solution can be represented through projecting random scalar onto a random basis. Due to the high computational cost associated with calculating the exact solution, a reduced approach is proposed where random eigenvalues and eigenvectors are approximated and low valued terms are discarded. Two novel multiplicative Galerkin error minimisation approaches have been presented. One being a weak Galerkin approach and the other being a strong Galerkin approach. Both approaches have been presented through projecting residuals onto random eigenvectors. The proposed methods have been used to analyse the bending of a static Euler-Bernoulli cantilever beam. The solutions obtained through the proposed methods have been compared with those obtained through direct MCS and through using a polynomial chaos method. Although both error minimisation approaches lowers the error of the mean, it is only the strong Galerkin error minimisation approach that substantially lowers the error arising in the standard deviation. However the computational time is significantly lower when the error minimisation methods are not used. Further work will be carried out would focus on efficient ways of computing random eigenfunctions and performing model-order reduction.

8 ACKNOWLEDGEMENTS

The authors acknowledge the financial support received from Engineering Research Network Wales (one of three Sêr Cymru National Research Networks).

BIBLIOGRAPHY

- [1] G. Maymon, *Structural Dynamics and Probabilistic Analysis for Engineers*, Butterworth-Heinemann, (2008).
- [2] E. VanMarcke, *Random Fields: Analysis and Synthesis*, MIT Press, Cambridge MA, (1983).
- [3] R. G. Ghanem and P. D. Spanon, *Stochastic Finite Elements: A Spectral Approach (revised edition)*, Dover Publications Inc., (2012).
- [4] K. Chang, *e-Design: Computer-Aided Engineering Design*, Academic Press, (2015).
- [5] G. S. Szekely and G. I. Schueller, "Computational procedure for a fast calculation of eigenvectors and eigenvalues of structures with random properties", *Computer Methods in Applied Mechanics and Engineering*, **191**, 799-816 (2001).
- [6] H. J. Pradlwarter, G. I. Schueller and G. S. Szekely, "Random eigenvalue problems for large systems", *Computers & Structures*, **80**, 2415-2424 (2002).
- [7] R. Ghanem and D. Ghosh, "Efficient characterization of the random eigenvalue problem in a polynomial chaos decomposition", *International Journal for Numerical Methods in Engineering*, **72**, 486-504 (2007).
- [8] S. Adhikari, *Structural dynamic analysis with generalized damping models: identification*, Mechanical Engineering and Solid Mechanics Series, Wiley - ISTE, London, (2014).
- [9] R. L. Fox and M. P. Kapoor, "Rates of change of eigenvalues and eigenvectors", *AIAA Journal*, **6**, 2426-2429 (1968).
- [10] J. E. Hurtado and A. H. Barbat, "Monte carlo techniques in computational stochastic mechanics", *Archives of Computational Methods in Engineering*, **5**, 3-29 (1998).
- [11] R. B. Nelson, "Simplified calculation of eigenvector derivatives", *AIAA Journal*, **14**, 1201-1205 (1976).
- [12] S. H. Chen, D. T. Song and A. J. Ma, "Eigensolution reanalysis of modified structures using perturbations and rayleigh quotients", *Communications in Numerical Methods in Engineering*, **10**, 111-119 (1994).
- [13] S. H. Chen, X. W. Yang and H. D. Lian, "Comparison of several eigenvalue reanalysis methods for modified structures", *Structural and Multidisciplinary Optimization*, **20**, 253-259 (2000).
- [14] B. Cochelin, N. Damil and M. Potierferry, "Asymptotic numerical-methods and pade approximants for nonlinear elastic structures", *International Journal for Numerical Methods in Engineering*, **37**, 1187-1213 (1994).
- [15] D. J. Dawe, *Matrix and Finite Element Displacement Analysis of Structures*, Oxford University Press, (1984).
- [16] A. Appleton, *Thermodynamic & Mechanical Properties of Matter*, Alan Appleton, (1996).