

Reliability based optimization in aeroelastic stability problems using polynomial chaos based metamodels

Anup Suryawanshi¹ · Debraj Ghosh¹

Received: 20 October 2014 / Revised: 13 July 2015 / Accepted: 14 August 2015 / Published online: 22 December 2015
© Springer-Verlag Berlin Heidelberg 2015

Abstract In this work, reliability based design optimization (RBDO) of two aeroelastic stability problems is addressed: (i) divergence, which arises in static aeroelasticity, and (ii) flutter, which arises in dynamic aeroelasticity. A set of design variables is considered as random variables, and the mean mass is minimized for a given set of constraints — including the probability of failure by divergence or flutter. The optimization process requires repeated evaluation of reliability, which is a major contributor to the total computational cost. To reduce this cost, a polynomial chaos expansion (PCE)-based metamodel is created over a grid in the parameter space. These precomputed PCEs are then interpolated for reliability calculation at intermediate points in the parameter space, as demanded by the optimization algorithm. Two new modifications are made to this method in this work. First, the Gauss quadrature rule is used — instead of statistical simulation — to estimate the chaos coefficients for higher computational speed. Second, to increase this computational gain further, a non-uniform grid is chosen instead of a uniform one, based on relative importance of the design parameters. This relative importance is found from a global sensitivity analysis. This new modified method is applied on a rectangular unswept cantilever wing model. For both optimization problems, it is

observed that the proposed method yields accurate results with a considerable computational cost reduction, when compared to simulation based methods. The effect of grid spacing is also explored to achieve the best computational efficiency.

Keywords Optimization under uncertainty · Monte Carlo simulations · Polynomial chaos expansion · Fluid-structure interaction · Nonlinear analysis · Metamodels

1 Introduction

Aeroelastic stability is an important concern in many engineering applications such as cable-supported bridge structures, tall chimneys, and aircrafts, to name a few (Dowell et al. 2004). In a fluid-structure interaction (FSI) problem, when the dynamic component of the structure is involved, flutter may occur at a certain velocity — termed as flutter speed. Whereas, when the dynamic component of the structure is absent, divergence may occur. Both of these instabilities may lead to a very large deformation, and thus to a structural failure. Thus, at the design stage, appropriate steps are taken to avoid these instabilities. However, predicting the critical flutter and divergence speeds becomes difficult due to uncertainties arising from manufacturing variability, inadequate knowledge about complex physics such as joints, variability in the flow parameters, and variability in flutter derivatives (Petit 2004; Beran et al. 2006; Kareem 2008; Cheng et al. 2003; Seo and Caracoglia 2011). Thus, these uncertainties must be considered during the design phase of such structures (Choi et al. 2007). The probability of failure calculation

✉ Anup Suryawanshi
suryawanshianup@gmail.com
Debraj Ghosh
dghosh@civil.iisc.ernet.in

¹ Department of Civil Engineering, Indian Institute of Science, Bangalore, 560012, India

considering these uncertainties is treated in the broad area of reliability analysis (Ditlevsen and Madsen 1996; Matthies et al. 1997; Okazawa et al. 2002; Schenk and Schuëller 2005; Rackwitz 2001; Hu and Youn 2011b; Chowdhury et al. 2009). Therefore, a search for an optimal design leads to a reliability-based design optimization (RBDO) — also referred to as reliability based optimization (Schuëller and Jensen 2008; Stanford and Beran 2012; Allen and Maute 2005; Valdebenito and Schuëller 2010; Wang et al. 2013; Paiva et al. 2014), as opposed to the deterministic optimization (Librescu and Maalawi 2009).

The solution process of an RBDO method has two major components: the optimization algorithm, and the reliability computation. The reliability computation may appear either in the objective function or in inequality constraints. Selection of an optimization algorithm depends on the convexity of the objective function; for convex objective functions a gradient-based method can be used, whereas non-convex functions require global optimization methods such as genetic algorithm. On the other hand, selection of a reliability method depends upon the desired accuracy level and availability of computational power. Recent surveys on RBDO can be found in Refs. Valdebenito and Schuëller (2010) and Aoues and Chateaufneuf (2010). The solution strategies of RBDO problems are often classified into three types of approaches, as (i) the usual double loop approach, where the outer loop is for the optimization iterations and the inner loop is for the reliability computation, (ii) single loop approach (Kuschel and Rackwitz 1997; Liang et al. 2007), and (iii) decoupling approach. While the double loop approach is the most general approach, the other two approaches were later developed to reduce the total computational cost under certain mitigating assumptions on the problem features — see Ref. Valdebenito and Schuëller (2010) and the references therein for more details. In the current work, the double-loop approach is used. A major computational cost in solving an RBDO problem is incurred from the requirement of repeated calculation of the reliability. Even outside the purview of RBDO, a number of methods have been proposed thus far on reducing the cost of reliability calculations. Reliability calculation methods can be broadly classified as simulation based — such as Monte Carlo, importance sampling, subset simulation (Au and Beck 2001), to name a few, and methods that do not rely on statistical simulation — such as first order reliability method (FORM), second order reliability method (SORM), to name a few. Reviews of reliability computation methods can be found in Rackwitz (2001) and Manohar and Gupta (2005). The prime goal of these reliability calculation methods is to maximize the accuracy of the reliability estimate with a minimal computational cost. The current work is aimed at reducing the cost of repeated evaluation of reliability, and thereby reducing the total cost of computation in

RBDO. To this end, a metamodel is used. Metamodels, also termed as response surfaces or surrogate models, are computationally inexpensive alternative models that mimic or approximate the actual physical response — which requires an expensive computation for evaluation. In general, a metamodel is created to approximate the response over a parameter range of interest. As the size of this parameter range increases, improvement of the metamodel is required. Thus, there is a trade-off between the cost of the metamodel and the desired accuracy. The cost of the metamodel includes both construction and execution costs. These metamodels then substitute the actual expensive model in the reliability computation phase, and thus help in reducing the total cost at this phase. One widely used approach for constructing metamodels is using polynomial expansions. Often the spectral stochastic finite element (Ghanem and Spanos 2003) method (SSFEM) is used to construct metamodel. Accordingly, a random variable, vector, or process is approximated using a set of orthogonal random polynomials — termed as polynomial chaos expansion (PCE) or generalized polynomial chaos expansion (gPCE) (Xiu and Karniadakis 2003). This polynomial expansion is later used instead of cumbersome numerical solvers such as finite element. Examples of usage of PCE and gPCE in uncertainty quantification (UQ) in aeroelastic systems are found in Refs. Witteveen (2008), Xiu et al. (2012) and Oladyschkin and Nowak (2012). A few examples of their usage in RBDO are found in Refs. Kim et al. (2006), Wei et al. (2008); Maute et al. (2009), Eldred and Burkardt (2009), Xiong et al. (2011), Blatman and Sudret (2010), Hu and Youn (2011a), Coelho et al. (2011), Wang et al. (2013), Ng and Eldred (2012) and Zhang (2013).

On the other hand, computational fluid dynamics (CFD) simulations are progressively becoming the norm in the flow computation community, including the area of aerospace engineering. This development is also encouraging the computational aeroelasticity, or the fluid-structure interaction community in general, to develop coupled CFD-CSD (computational structural dynamics) solvers (Geuzaine et al. 2003). However, all these solvers are computationally expensive, and the cost of using such a solver in an RBDO problem may become prohibitive. This high cost automatically encourages development of metamodel based techniques. Motivated by the recent progress of usage of SSFEM in the area of aeroelasticity and associated optimization — as discussed above — PCE-based metamodels have been developed (Coelho et al. 2011; Ng and Eldred 2012). While in the existing literature — including the papers cited here — the PCE is constructed or re-computed in each optimization iteration, in Coelho et al. (2011) a different approach is taken. There the PCE is constructed only once at a set of uniformly-spaced grid points in the design parameter space, and later interpolated during the

optimization. The chaos coefficients are estimated using a statistical simulation method such as Monte Carlo (MC) and Latin hypercube sampling (LHS) (Choi et al. 2004). It is observed that despite the additional cost of pre-computation of PCEs, the cost saving made by this interpolation leads to a faster RBDO.

The main contribution of the current work lies in improving the computational speed of this PCE metamodel by introducing following two modifications. (i) The Gauss quadrature is used to estimate the chaos coefficients, instead of a statistical simulation, to reduce the number of executions of the true model. Here, the true model corresponds to the full-scale aeroelasticity solver. (ii) Based on a global sensitivity analysis in the design parameter space using Sobol’ indices, a non-uniform grid is used for constructing the metamodel. This step led to even fewer executions of the true model. A gradient based optimization method is used (Arora 2006; Rao 2008).

This paper is organized as follows. The RBDO problem and the proposed metamodel are presented in Section 2. They are put in the context of aeroelastic stability problems in Section 3. Numerical studies are presented in Section 4. Finally, concluding remarks are made in Section 5.

2 Reliability based design optimization: the proposed method

Consider the probability space (Ω, \mathcal{F}, P) with Ω denoting the sample space with elements θ , \mathcal{F} denoting a corresponding σ -algebra, and P denoting the probability measure. In this space, let a set of random variables $\{\eta_i(\theta)\}_{i=1}^{i=p}$ — expressed here as a p -dimensional vector $\boldsymbol{\eta}$ — characterize the underlying uncertainty in the problem. That is, all the random parameters in the design optimization problem are functions of this vector $\boldsymbol{\eta}$. Examples of such random parameters are geometric dimensions of the structure, stiffness and mass properties, wind field parameters such as velocity, and the angle of attack. Among these random parameters, let the (uncertain) design parameters be expressed as a d dimensional vector $\mathbf{x}(\boldsymbol{\eta})$ whose components $x_i(\boldsymbol{\eta})$ are independent random variables. Let another d dimensional vector $\boldsymbol{\mu}_x$ denote the mean of $\mathbf{x}(\boldsymbol{\eta})$. The RBDO problem considered here is of the form

$$\begin{aligned} & \underset{\boldsymbol{\mu}_x}{\text{Minimize}} f(\mathbf{x}) \\ & \text{Subjected to} \\ & g_i(\mathbf{x}) \leq 0, \quad i = 1, 2, 3, \dots, k_1 \\ & h_j(\mathbf{x}) = 0, \quad j = 1, 2, 3, \dots, k_2 \end{aligned} \tag{1}$$

where $f(\mathbf{x}) \in \mathbb{R}$ denotes the objective function, and $g_i(\mathbf{x})$ -s and $h_j(\mathbf{x})$ -s denote the inequality and equality

constraints, respectively. For brevity, the argument $\boldsymbol{\eta}$ is suppressed from $\mathbf{x}(\boldsymbol{\eta})$. Often, a bound on the parameters such as $\mu_{min} \leq \mu_x \leq \mu_{max}$ is also used. Although the design parameters are random, the optimization parameters $\boldsymbol{\mu}_x$, the objective function, and the constraints are finally expressed as deterministic quantities. In the current case $f(\mathbf{x})$ is the mean mass. One inequality constraint is the probability of failure — denoted here as P_f — should not exceed a tolerance value P_{tol} , that is, $P_f \leq P_{tol}$. Details of these expressions will be given later in the context of numerical studies.

The optimization problem stated in Eq. 1 is solved here in a double-loop approach. The outer loop corresponds to the conventional deterministic optimization loop. The inner loop corresponds to the probability of failure computation. In this work, a gradient based method is used for the optimization loop, without any loss of generality. Since the objective function is the mean mass, a statistical simulation is not required for its computation. However, one inequality constraint requires computation of the probability of failure P_f , which is an expensive calculation. Note that this probability calculation needs to be performed multiple times in an optimization problem. Therefore a significant amount of computational resource is spent in this step.

In this work, a PCE-based metamodel (Coelho et al. 2011) is used to reduce this cost. However, two modifications are proposed to this metamodel to accelerate the computation. This metamodel and the modifications are described next.

2.1 Polynomial chaos expansion

A square-integrable random variable, random vector, or random process can be expressed in a mean-square convergent series using random orthogonal polynomial bases — known as PCE for Hermite bases, and gPCE for other bases. For Hermite bases, the basic random variables should be Gaussian. Therefore, if the random vector $\boldsymbol{\eta}$ is not Gaussian, it needs to be transformed to another q -dimensional random vector $\boldsymbol{\xi}$ with elements $\{\xi_i\}_{i=1}^{i=q}$ as independent standard normal variables. Denote the joint probability density function (PDF) of $\boldsymbol{\xi}$ as $p(\boldsymbol{\xi})$, and define the expectation operator $\mathbb{E}\{\cdot\}$ as

$$\mathbb{E}\{\cdot\} = \int_{\mathbb{R}^q} \cdot p(\boldsymbol{\xi}) d\boldsymbol{\xi} \tag{2}$$

Then, a square integrable random variable $u(\boldsymbol{\xi})$ is expressed in PCE as Ghanem and Spanos (2003)

$$u(\boldsymbol{\xi}) = \sum_{i=0}^{\infty} u_i \psi_i(\boldsymbol{\xi}), \quad u_i \in \mathbb{R}, \tag{3}$$

here u_i are referred to as chaos coefficients, ψ_i are the Hermite polynomials in the set ξ , holding the properties

$$\psi_0 \equiv 1, \quad \mathbb{E}\{\psi_i\} = 0 \text{ for } i > 0, \quad (4)$$

$$\mathbb{E}\{\psi_i \psi_j\} = \delta_{ij} \mathbb{E}\{\psi_i^2\}, \quad (5)$$

with δ_{ij} denoting the Kronecker delta function. The PCE requires a truncation for computational purpose, as

$$u(\xi) \approx \hat{u}(\xi) = \sum_{i=0}^{r-1} u_i \psi_i(\xi), \quad u_i \in \mathbb{R}, \quad (6)$$

where the value of r depends upon q , and the degree or order of polynomial expansion, denoted by d . The number of terms in the truncated expansion is

$$r = \binom{q+d}{d} = \frac{(q+d)!}{q!d!}. \quad (7)$$

2.2 The PCE metamodel and proposed modifications

For the optimization problem, there is a constraint $g_i(\mathbf{x})$ in terms of divergence or flutter velocity. These velocities are nonlinear functions of the design parameter vector \mathbf{x} . This nonlinearity of the mapping leads to two assertions: (I) The PCE needs to be at least of order two. (II) The probability distribution of the divergence or flutter velocity, and thus the corresponding chaos coefficients, will vary with a variation in the mean value $\mu_{\mathbf{x}}$ of the design parameter. Following assertion (II), when the PCE is used for approximating the flutter or divergence velocity, the chaos coefficients need to be recomputed for any fresh value of $\mu_{\mathbf{x}}$ sought by the optimization during the iterations. To this end, in Coelho et al. (2011), first a uniform grid on the optimization parameter space $\mu_{\mathbf{x}}$ was chosen. Then, the chaos coefficients on these grid points were estimated using statistical simulation, by evaluating

$$u_i = \frac{\mathbb{E}\{u(\xi) \psi_i(\xi)\}}{\mathbb{E}\{\psi_i^2(\xi)\}} = \frac{\sum_{j=1}^{j=N} u(\xi_j) \psi_i(\xi_j)}{\sum_{j=1}^{j=N} \psi_i^2(\xi_j)} \quad (8)$$

for $i = 0, 1, \dots, r-1$, where ξ_j denotes the j^{th} realization of the random vector ξ and N denotes the sample size. ξ_j -s are generated by a readily available random number generator (Manohar and Gupta 2005; Valdebenito and Schuëller 2010).

These $u^{(i)}$ -s are stored. Subsequently, the chaos coefficients at any intermediate point $\mu_{\mathbf{x}}$ are estimated by interpolating the stored $u^{(i)}$ -s. The interpolation can be done by various methods, such as using spline functions. The P_f at this point is estimated directly from the resulting PC expansion.

Finally, these coefficients were interpolated on-the-fly during the optimization loop. Thus the computational saving is achieved by avoiding multiple runs of the expensive FSI solver at these intermediate points. This process is graphically depicted in Fig. 1. This figure shows a set of 16 grid points on a two-dimensional parameter space $\mu_{\mathbf{x}} \in \mathbb{R}^2$, here $u^{(i)}$ denotes the set of chaos coefficients evaluated at the i^{th} grid-point.

We propose two modifications to this method. The first modification is on the evaluation of the chaos coefficients. While the statistical simulation has the advantage of being dimension independent, for low-dimensional space (i.e. low q) the Gauss quadrature is economical. Accordingly, again (8) is used. However, now the numerator is evaluated using Gauss quadrature and the denominator is obtained from a table lookup (Ghanem and Spanos 2003). Quadrature points for integration can be found in Abramowitz and Stegun (1984). For higher dimensions, sparse grids or Smolyak cubature can be used (Ng and Eldred 2012), instead of the tensor product structure. Note that both simulation and Gauss quadrature methods are “non-intrusive”, that is, the (expensive) flow-structure interaction solver (or the numerical code to compute the critical velocity) is called repeatedly. The second change is to use a non-uniform grid. To achieve it, first the Sobol’ indices (Sobol’ 2001) are estimated for evaluating sensitivities. These indices naturally gave a rank ordering of the design variables. Then, while constructing the grid, most important variables are divided into finer divisions while the remaining variables are divided coarsely. This selection of a grid with different resolutions along different random variables would lead to reduce the total number of grid points, and thereby, would lead to fewer executions of the true model.

The proposed modified method can be summarized as follows. First, find the Sobol’ indices, and rank-order the design variables accordingly. Then, following the relative importance, discretize the space of design variables into a non-homogeneous grid. Then, estimate all chaos coefficients at each grid point using Gauss quadrature, and store them. Then, start the optimization algorithm, and in each iteration, interpolate the chaos coefficients to estimate the probability of failure P_f . The final outcome of the entire optimization process is the set of design points $\mu_{\mathbf{x}}$ and the corresponding objective function such as mass. The entire optimization method is presented in a flowchart in Fig. 2.

The proposed modified method, although presented here for aeroelastic stability problems, is applicable to some other mechanics problems as well, since the characteristics of the aeroelastic problems are not used anywhere. The corresponding mechanics solver can directly be used to estimate the chaos coefficients at the grid points.

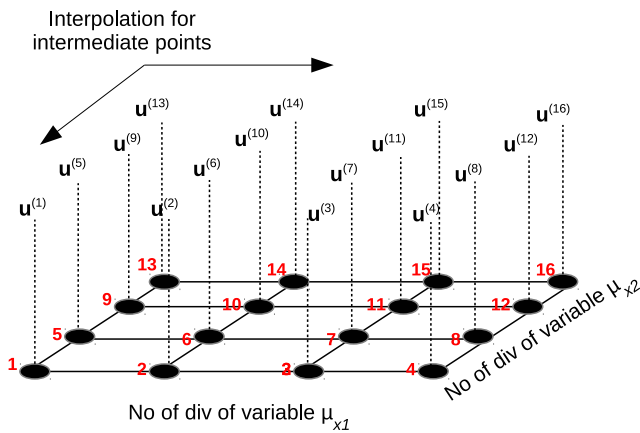


Fig. 1 Interpolation over the grid on the optimization parameter space

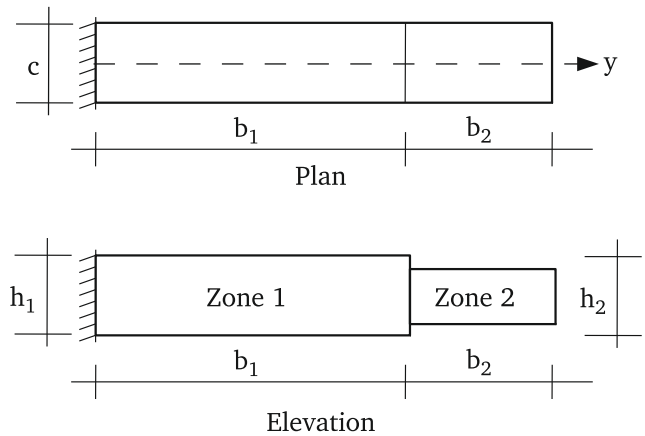


Fig. 3 The rectangular wing model

3 Optimization problems in aeroelastic stability

For demonstrating and testing the proposed method, a rectangular wing with uniform chord width but varying thickness is considered, as shown in Fig. 3. In this figure, the total semi-span of the wing is composed of two zones with lengths b_1 and b_2 , and thicknesses h_1 and h_2 , respectively. The uniform chord of the wing is denoted by c . This model

and other such simplified models are often used to demonstrate or test computational tools (Librescu and Maalawi 2009; Stanford and Beran 2012).

3.1 Divergence

The solution strategy adopted to obtain the divergence velocity of the rectangular wing model is outlined here, further details can be found in Ref. Librescu and Maalawi (2009). Divergence is mainly associated with the torsional instability of the wing cross section. To study the divergence phenomenon, the flow is assumed to be steady and incompressible, and the fluid load is modeled using the aerodynamic strip theory. The governing differential equation of torsion is expressed as Bisplinghoff et al. (2013)

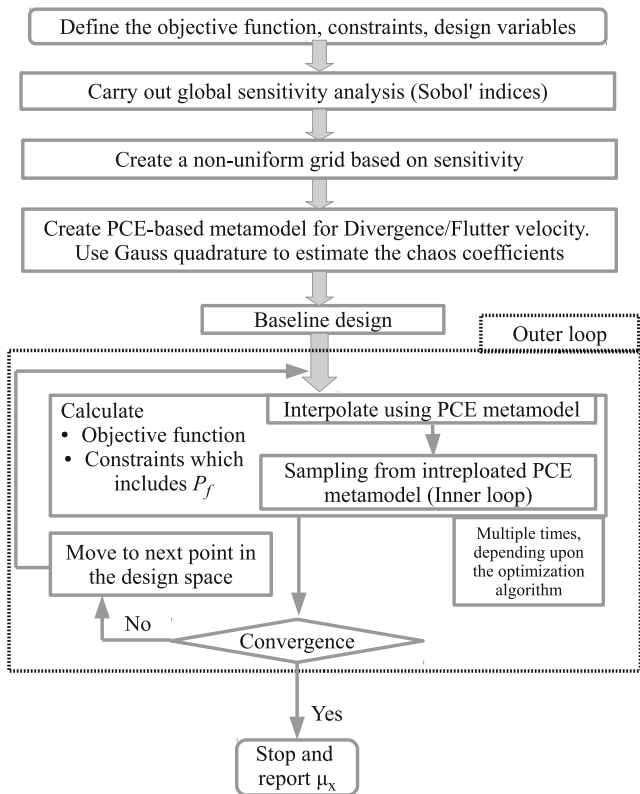


Fig. 2 Flowchart of the RBDO using the proposed metamodel

$$\frac{d}{dy} \left(GJ \frac{d\alpha_e}{dy} \right) + T(y) = 0 \tag{9}$$

where GJ denotes the torsional stiffness, α_e denotes the elastic twist, and $T(y)$ denotes the torque due to the wind load about the axis y . Using the zero displacement and zero slope boundary conditions, and other usual simplifications, the following nonlinear equation is obtained

$$\tan \left(\hat{V} \frac{\hat{b}_1}{\sqrt{\hat{h}_1}} \right) \tan \left(\hat{V} \frac{\hat{b}_2}{\sqrt{\hat{h}_2}} \right) = \sqrt{\frac{\hat{J}_1}{\hat{J}_2}}, \tag{10}$$

where, \hat{V} is the unknown, \hat{h}_1 and \hat{h}_2 are normalized thicknesses, \hat{b}_1 and \hat{b}_2 are normalized lengths of each zone, and \hat{J}_1 and \hat{J}_2 are normalized torsional constants of each zone, respectively. The normalization is done with respect to a set of baseline parameters. The lowest possible root of (10) gives the divergence velocity of the wing model.

3.2 Flutter

In the flutter calculation, the dynamic effect is included. The equation of motion of a multi-degree of freedom vibrating aeroelastic system can be written as

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{C}\dot{\mathbf{q}} + \mathbf{K}\mathbf{q} = \mathbf{F}(\dot{\mathbf{q}}, \mathbf{q}, t), \quad (11)$$

where \mathbf{M} denotes the inertia matrix, \mathbf{C} denotes the damping matrix, \mathbf{K} denotes the stiffness matrix, \mathbf{q} denotes the displacement vector, and $\mathbf{F}(\dot{\mathbf{q}}, \mathbf{q}, t)$ denotes the excitation. The dots over \mathbf{q} represent derivatives with respect to time. A widely-used two-degrees-of freedom (2DOF) model of an airfoil cross section with heaving (h) and pitching (α) degrees of freedom is shown in Fig. 4, and is used in this study.

The 2DOF model is subjected to motion dependent forces that are expressed using the flutter derivatives H_1^* , H_2^* , H_3^* , A_1^* , A_2^* , and A_3^* . Flutter derivatives are functions of the reduced frequency k , which itself is dependent on the natural frequency and the characteristic length of the cross section. The motion dependent forces on the 2DOF model are given as Scanlan and Tomko (1971)

$$L = \frac{\rho_{air} V_\infty^2}{2} (c) \left(k H_1^* \frac{\dot{h}}{V_\infty} + k H_2^* \frac{c \dot{\alpha}}{2 V_\infty} + k^2 H_3^* \alpha \right), \quad (12)$$

$$M = \frac{\rho_{air} V_\infty^2}{4} (c^2) \left(k A_1^* \frac{\dot{h}}{V_\infty} + k A_2^* \frac{c \dot{\alpha}}{2 V_\infty} + k^2 A_3^* \alpha \right), \quad (13)$$

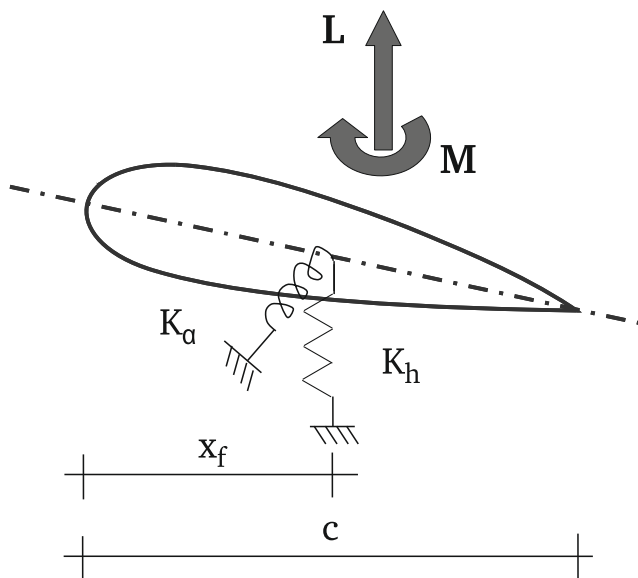


Fig. 4 Spring mounted airfoil cross section with lift force and torsional moment

where ρ_{air} denotes the density of air, V_∞ denotes the freestream velocity, and c denotes the chord of the airfoil. All motion dependent forces on the right hand side of (11) are transferred to the left hand side to combine them with the corresponding terms. This step yields the equation

$$\mathbf{M}\ddot{\mathbf{q}} + [\rho_{air} V_\infty \mathbf{D} + \mathbf{C}]\dot{\mathbf{q}} + [\rho_{air} V_\infty^2 \mathbf{E} + \mathbf{K}]\mathbf{q} = \mathbf{0}, \quad (14)$$

where the matrices \mathbf{D} and \mathbf{E} are formed by rearranging the terms from (12) and (13). \mathbf{D} is referred to as the aerodynamic damping and \mathbf{E} is referred to as the aerodynamic stiffness. These matrices are functions of the freestream velocity V_∞ , the reduced frequency k , and the flutter derivatives H_i^* , A_i^* , $i = 1, 2, 3$. Typical forms of the above mentioned matrices are given next. The inertia matrix \mathbf{M} is expressed as

$$\mathbf{M} = \begin{bmatrix} m_h & 0 \\ 0 & I_\alpha \end{bmatrix},$$

the structural stiffness matrix \mathbf{K} is expressed as

$$\mathbf{K} = \begin{bmatrix} K_h & 0 \\ 0 & K_\alpha \end{bmatrix},$$

the aerodynamic damping matrix \mathbf{D} is expressed as

$$\mathbf{D} = \begin{bmatrix} k H_1^* c / 2 & k H_2^* c^2 / 4 \\ k A_1^* c^2 / 4 & k A_2^* c^3 / 8 \end{bmatrix},$$

and the aerodynamic stiffness matrix \mathbf{E} is expressed as

$$\mathbf{E} = \begin{bmatrix} 0 & k^2 H_3^* c / 2 \\ 0 & k^2 A_3^* c^2 / 4 \end{bmatrix}.$$

Here m_h , I_α , K_h and K_α represent mass, mass moment of inertia, flexural and torsional stiffness, respectively. Upon moving to a state-space form, (14) leads to a complex eigenvalue problem. The coefficient of the imaginary term of the solution of this complex eigenvalue problem represents the damping ratio. The airflow velocity at which the damping ratio becomes negative is the flutter velocity of the aeroelastic system. A typical damping trend for flutter speed calculation is shown in Fig. 5.

3.3 The optimization problem

The RBDO problem for safety against divergence and flutter is formulated next. In both aeroelastic stability problems, the semi-span of wing, b , and the thickness of the cross section h are considered as random design parameters. For the wing model with two zones, it implies that b_1 , h_1 , b_2 , and h_2 are random. However, constraints on the total length of the wing and the thickness ratio are imposed. The objective function is the mean mass, and the inequality constraints

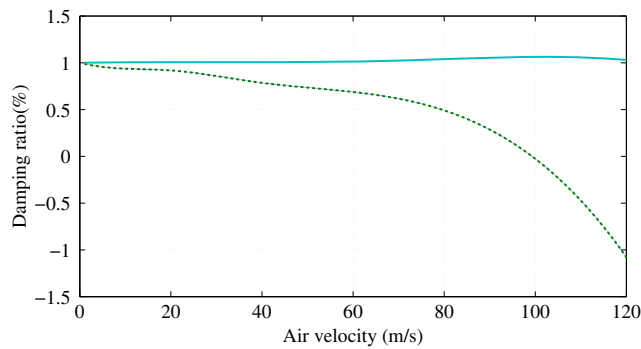


Fig. 5 Typical damping trend for flutter speed calculation

are in terms of the exceedance probabilities of the divergence and flutter velocities, respectively. The optimization problem is stated as

$$\begin{aligned}
 & \underset{(\mu_{h_1}, \mu_{b_1}, \mu_{h_2}, \mu_{b_2})}{\text{Minimize}} && (\mu_{h_1}, \mu_{b_1}, \mu_{h_2}, \mu_{b_2}) \\
 & \text{Subjected to} && \\
 & && P_f \leq P_{tol}, \\
 & \mu_{h_{lower}} \leq \mu_{h_i} \leq \mu_{h_{upper}}, && i = 1, 2 \\
 & \mu_{b_{lower}} \leq \mu_{b_i} \leq \mu_{b_{upper}}, && i = 1, 2 \\
 & && \mu_{b_1} + \mu_{b_2} = b_{tot}, \\
 & && \mu_{h_1} / \mu_{h_2} = h_{rat}.
 \end{aligned} \tag{15}$$

Here, b_{tot} denotes the total span of the wing, h_{rat} denotes the fixed ratio of thickness, $\mu_{h_{lower}}$ and $\mu_{h_{upper}}$ are lower and upper bounds of the mean wing thickness, and $\mu_{b_{lower}}$ and $\mu_{b_{upper}}$ are the lower and upper bounds of the mean length of each wing zone, respectively. All of these constants, along with P_{tol} , are fixed by the designer. P_f is defined as $P(V_{divergence/flutter} \geq V_\infty)$, where the freestream velocity V_∞ is also random. Note that the equality constraints reduce the dimensionality of the problem by two, that is, only μ_{h_1} and μ_{b_1} can be considered as optimization parameters.

4 Numerical studies

The proposed modified method is implemented for both divergence and flutter instabilities, and the results are reported in this section. Accuracy of these results is verified against two common methods of solving RBDO problems — Monte Carlo, and LHS. The computational speed gain, when compared to these two methods, is also reported. While estimating the chaos coefficients, both LHS and Gauss quadrature are used, and the comparison is reported. Upon evaluation of the chaos coefficients and subsequent interpolation using bicubic splines, the sampling from the metamodel — which is in terms of a PCE with known

coefficients — is performed using LHS. The geometric dimensions h_1 and b_1 are modeled as independent random variables as

$$\begin{aligned}
 h_1(\xi_1) &= \mu_{h_1} + \frac{\sigma_{h_1}}{\sqrt{2}}(\xi_1^2 - 1), \\
 b_1(\xi_2) &= \mu_{b_1} + \frac{\sigma_{b_1}}{\sqrt{2}}(\xi_2^2 - 1),
 \end{aligned} \tag{16}$$

where σ_{h_1} and σ_{b_1} denote standard deviations, ξ_1 and ξ_2 are two independent standard normal variables. Positivity of these two dimensions is imposed by choosing $\frac{\sigma_{h_1}}{\sqrt{2}} < \mu_{h_1}$ and $\frac{\sigma_{b_1}}{\sqrt{2}} < \mu_{b_1}$. The freestream velocity is modeled as an independent random variable with Rayleigh distribution. All computations are performed using Matlab (MathWorks 2013).

Two proposed modifications are now implemented in sequence. First, the Gauss quadrature is implemented, and the accuracy and computational speed gain are reported in Section 4.1. Then, the non-uniform grid is implemented, in addition to the Gauss quadrature, and the results are reported in Section 4.2.

4.1 Use of the Gauss quadrature for RBDO

In this section, the first proposed modification, that is, the Gauss quadrature, is implemented in the RBDO process. In all examples, the chaos coefficients are obtained using the 4 point Gauss quadrature.

4.1.1 Example 1: divergence

First, it should be tested if the PCE performs as a good metamodel for the divergence velocity. To this end, at an arbitrarily chosen design point μ_x , the chaos coefficients u_i defined in (3) are generated using (i) a statistical sampling, and (ii) the Gauss quadrature. Then, the probability density functions (PDFs) estimated using the second order

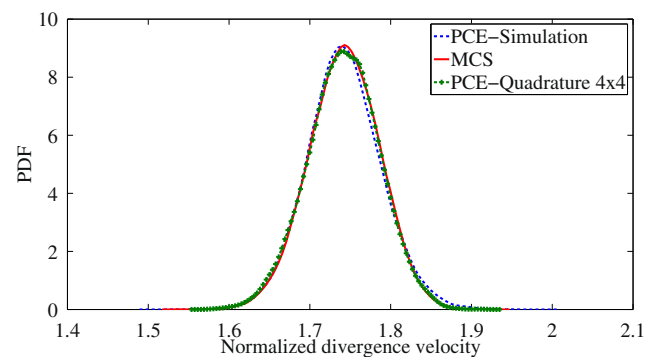


Fig. 6 Comparison of PDFs for divergence velocity using MC, PCE with simulation, and PCE with 4 point Gauss quadrature. Second order chaos expansion is used

Table 1 Results for the optimization problem considering divergence instability

| Column 1 | | Column 2 | | Column 3 | |
|----------------------|--------------------------|------------------|--------------|------------------|----------------|
| | | Simulation based | | PCE metamodel | |
| | | MC | LHS | Simulation | Quadrature |
| Number of executions | True model | 71001 | 70001 | 25×10^3 | 25×16 |
| | Metamodel | - | - | 73000 | 71000 |
| Optimization results | (μ_{h_1}, μ_{b_1}) | (0.175, 3.9) | (0.175, 3.9) | (0.17, 3.9) | (0.17, 3.9) |
| | (μ_{h_2}, μ_{b_2}) | (0.14, 3.6) | (0.14, 3.6) | (0.14, 3.6) | (0.14, 3.6) |

$P_{tol} = 10^{-2}$, and 10^3 realizations are used in all samplings. The divergence velocity is in m/s and design variables are in m . The grid size for the metamodel is 5×5

PCE are plotted in Fig. 6, and are compared with the true PDF estimated using Monte Carlo sampling. The comparison shows a good match, which implies that the PCE can serve as a good metamodel for the chosen problem. The maximum difference in the chaos coefficients estimated by quadrature and by simulation is found to be 1.8 % for the mean and linear terms, and 11.1 % for the quadratic terms.

The optimization problem is next solved, where the standard deviations of the random variables are chosen as 5 % of the mean, and P_{tol} is chosen as 10^{-2} . The number of realizations in all statistical simulations is chosen as 10^3 . A second order PCE is used to create the metamodel, as this order was numerically found to be sufficient. The metamodel was created using a 5×5 grid on the parameter space. That means, a total of 25 six-dimensional vectors of chaos coefficients are evaluated and stored in the computer memory. Initial values of the design variables (μ_{h_1}, μ_{b_1}) are (0.12, 5.625). The results are reported in Table 1. In this table, Column 2 reports the results from a pure simulation-based RBDO — used here as the benchmark for comparison, and Column 3 reports the results using the proposed method. While the initial design point cor-

responds to $P_f = 0.99$, the optimal design corresponds to $P_f = 0.01$. Similar trend is followed in all other examples.

The key observations from this table are as follows. First, the proposed PCE metamodel based method yields good accuracy in solving the optimization problem compared to pure simulation-based methods. Next, the number of executions of the true model is reduced significantly, especially when the quadrature is used to evaluate the chaos coefficients. This reduction is the main gain in the proposed method. As the mechanics model — termed as the true model here — becomes more elaborate, its relative execution cost becomes higher compared to the metamodel, since the metamodel involves only a few scalar-level operations.

4.1.2 Example 2: flutter

The RBDO considering flutter instability is considered next. The flutter derivatives reported in Ref. Le Maître et al. (2003) are used in (14). In the literature, the flutter derivatives are obtained for a few values of the reduced frequency k . Hence, to evaluate the flutter derivatives at intermediate values of k , a spline interpolation is used. Similar to the divergence case, applicability of the PCE as a metamodel is again verified here. Thus, the PDFs of the flutter velocity at an arbitrarily chosen design point are compared in Fig. 7. Here also it is noticed that the second-order PCE yields a good approximation.

For the optimization problem two P_{tol} -s are chosen: 10^{-1} and 10^{-3} . First $P_{tol} = 10^{-1}$ is considered, and the results are reported in Table 2. The initial iterate is chosen as $(\mu_{h_1}, \mu_{b_1}) = (0.18, 1.5)$ and a 10×10 grid is used for creating the metamodel. Here V_{fl} denotes the flutter velocity of the wing with optimal dimensions. 10^2 realizations are used in all statistical samplings. Column 2 of Table 2 shows the computational cost for pure simulation based optimization, whereas column 3 shows the computational cost for PCE

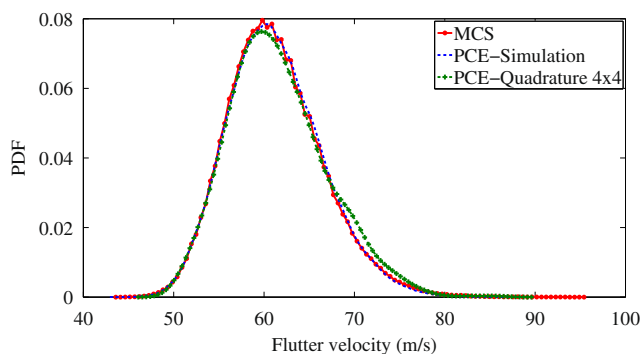


Fig. 7 Comparison of PDFs for flutter velocity using MC, PCE with simulation, and PCE with 4 point Gauss quadrature. Second order chaos expansion is used. The grid size is 10×10

Table 2 Results for the optimization problem considering flutter instability

| Column 1 | | Column 2 | | Column 3 | |
|----------------------|--------------------------|------------------|-------------|-------------------|-------------|
| | | Simulation based | | PCE metamodel | |
| | | MC | LHS | Simulation | Quadrature |
| Computation time | | ~162 min | ~151 min | ~149 min | ~29 min |
| Number of executions | True model | 11501 | 10401 | 100×10^2 | 1600 |
| | Metamodel | - | - | 10200 | 10700 |
| Optimization results | (μ_{h_1}, μ_{b_1}) | (0.16, 2.98) | (0.16, 3.1) | (0.16,3.05) | (0.16,3.05) |
| | (μ_{h_2}, μ_{b_2}) | (0.13, 4.52) | (0.12, 4.4) | (0.12,4.45) | (0.12,4.45) |

$P_{tol} = 10^{-1}$, and 10^2 realizations are used in all samplings. The flutter velocity is in m/s and design variables are in m . The grid size for the metamodel is 10×10

metamodel based optimization. Similar to the divergence problem, it is observed that (i) the metamodel-based method yields good accuracy in the solution, and (ii) the quadrature-based PCE yields considerable computational acceleration — both in terms of computational time and number of executions of the true model.

The convergence history of the objective function is shown in Fig. 8. It is observed from this figure that the optimization converges in 12 iterations. The rate of convergence of the RBDO process is driven by the default tolerance level defined within the solver. Since a lower tolerance level will require more iterations, the computational saving using the metamodel will be more pronounced in that case.

For problems of practical interest, often a much lower value of P_f is expected. Hence, results for the flutter RBDO problem for $P_{tol}=10^{-3}$ are obtained next. The grid size is 10×10 , and 10^4 realizations are used in all statistical simulations. The results are reported in Table 3. It is observed that, for this low value of P_f , the computational time required by the pure simulation-based method was prohibitive, and thus is not reported. Here the computational

gain is even more pronounced — both in terms of computational time and number of executions of the true model. The cost difference between the sampling-based and quadrature-based metamodels is very high — the first one takes about seven days whereas the second one takes only two-and-half hours.

The effect of the grid size in the PCE metamodel on accuracy and computational time is studied next. In this case, the LHS with 10^4 realizations is used for $P_{tol} = 10^{-3}$. Figure 9 shows the cost breakup, as well as the total computational cost of the RBDO. It is observed that the total computational costs for very coarse grids of sizes 3×3 and 6×6 are the least. However, the optimization results are found to be relatively inaccurate for these coarse grid sizes — and thus are not reported here. It is observed that accuracy is achieved from grid size of 7×7 onwards. The total computational cost keeps fluctuating till grid size of 13×13 , after which it begins to converge. It should be noted that the total computational cost of estimating the PC coefficients increases with the grid size. The fluctuating component of the total computational cost is the time required for optimization.

To further explore the effect of grid size, the number of reliability calculations for each grid size for $P_f=10^{-3}$

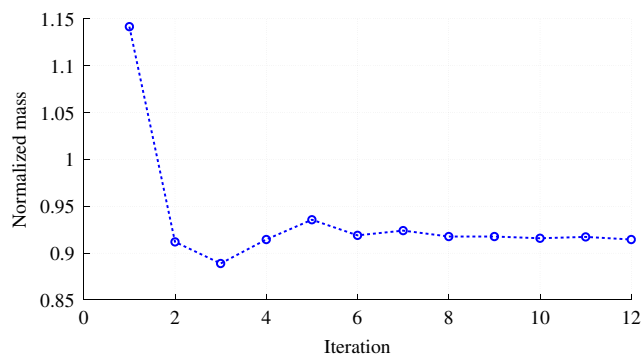


Fig. 8 Typical convergence history of the objective function for grid size 10×10 and $P_f=10^{-1}$. The chaos coefficients were obtained using the quadrature rule

Table 3 Results for the optimization problem considering flutter instability

| Column 1 | | Column 2 | Column 3 |
|----------------------|--------------------------|-------------------|-----------------|
| | | PCE Metamodel | |
| | | Simulation | Quadrature |
| Computation time | | ~7 days | ~149 min |
| Number of executions | True model | 100×10^4 | 16×100 |
| | Metamodel | 640000 | 590000 |
| Optimization results | (μ_{h_1}, μ_{b_1}) | (0.16,1.9) | (0.16,1.95) |
| | (μ_{h_2}, μ_{b_2}) | (0.12,5.6) | (0.12,5.55) |

$P_{tol} = 10^{-3}$, and 10^4 realizations are used in all samplings. The flutter velocity is in m/s and design variables are in m . The grid size for the metamodel is 10×10

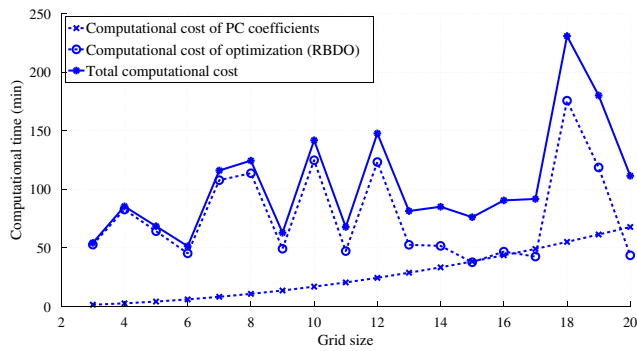


Fig. 9 Computational time for each grid size for the flutter problem with $P_{tol}=10^{-3}$ and the number of realizations = 10^4 . PC coefficients are estimated using 4 point Gauss quadrature

with 10^4 realizations is presented in Fig. 10. The default tolerance level of the solver is used in this case. It should be noted that a lower tolerance level would require more reliability calculations than shown in this figure. In this figure, note that the number of reliability calculations does not vary significantly beyond a grid size — at and above 8×8 in this case. This observation asserts the existence of an optimal grid size for the metamodel in terms of computational cost. However, the level of accuracy cannot be judged by this figure. As mentioned earlier, the time required to calculate the flutter velocity for each realization of random variables is reduced to a great extent by replacing the true model with the PCE based metamodel. This helps to perform reliability estimation in a computationally inexpensive manner in RBDO.

Thus, in both the examples it is noticed that the usage of Gauss quadrature to estimate the chaos coefficients has led to a significant computational saving with comparable accuracy.

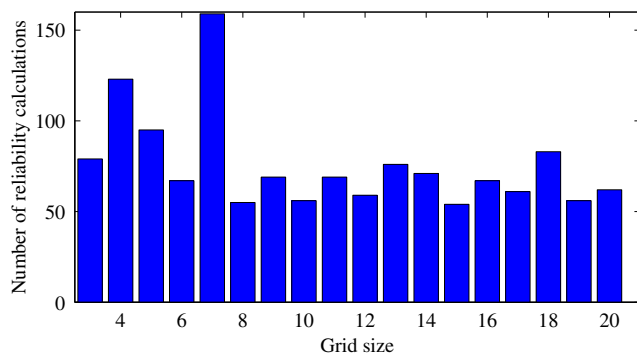


Fig. 10 Reliability calculations for each grid size for the flutter problem with $P_f=10^{-3}$. 10^4 realizations are used, and the default tolerance level of the optimization solver is used

Table 4 Results for the optimization problem considering divergence instability upon using Sobol’ indices and therefore a non-uniform grid

| | | PCE Metamodel Quadrature |
|----------------------|--|-----------------------------|
| Number of executions | True model Metamodel | 20×16 73000 |
| Optimization results | (μ_{h_1}, μ_{b_1}) (μ_{h_2}, μ_{b_2}) | (0.17,3.9) (0.14,3.6) |

$P_{tol}=10^{-2}$, and 10^3 realizations are used. The grid size for the metamodel is 5×4

4.2 Use of a non-uniform grid

The second proposed modification, usage of a non-uniform grid based on global sensitivity, will be numerically tested now.

4.2.1 Example 1: divergence

For the divergence example, Sobol’ indices for length and thickness are found to be 0.63 and 0.37, respectively. Thus, to obtain an appropriate mesh size, discretization should be finer for the length parameter b , and coarser for the thickness h . Accordingly, a mesh size of 5×4 is chosen, instead of the previous size 5×5 . The RBDO is carried out for the metamodel obtained using this non-uniform mesh. The cost of computation of Sobol’ indices is lower compared to the RBDO. The optimized values of the design parameters are given in Table 4. The results in this table should be compared with quadrature results of Column 3 of Table 1. Here, a reduction in the computational cost is observed due to fewer executions of the true model.

4.2.2 Example 2: flutter

Sobol’ indices for flutter velocity in the present problem are found to be as follows: length $b : 0.9$, and thickness $h : 0.1$. Using this rank ordering, the uniform mesh size 10×10 is

Table 5 Results for the optimization problem considering flutter instability upon using Sobol’ indices and therefore a non-uniform grid

| | | PCE Metamodel Quadrature |
|----------------------|--|-----------------------------|
| Computation time | | ~ 102 min |
| Number of executions | True model Metamodel | 50×16 560000 |
| Optimization results | (μ_{h_1}, μ_{b_1}) (μ_{h_2}, μ_{b_2}) | (0.16,1.98) (0.12,5.52) |

$P_{tol}=10^{-3}$, and 10^4 realizations are used. The grid size for the metamodel is 10×5

now changed to a non-uniform mesh of size 10×5 . Again, the non-uniform mesh is kept finer in the ‘*b*-direction’ and coarser in the ‘*h*-direction’. However, the number of divisions along *h* is half the number of divisions along *b*. The reason is, compared to the divergence example, here the relative sensitivity for *h* is much lower. The RBDO is carried out for the metamodel obtained using this non-uniform mesh. The optimized values of the design parameters are given in Table 5. This table should be compared with Column 3 of Table 3. By this comparison, a reduction in the computational cost is observed — from 149 minutes to 102 minutes. The reduction in the computational time is mainly due to lesser number of executions of the true model. The Sobol’ indices are obtained by evaluating the PC coefficients using the Gauss quadrature. The cost of computation involved in evaluating these indices is small compared to the RBDO.

Thus, in both the examples, it is observed that the selection of a non-uniform grid has led to a considerable cost saving. The benefit is more pronounced in the flutter example.

5 Concluding remarks

Both the proposed modifications to the PCE-based metamodel have led to a significant cost saving. As the cost of running the true model increases, the computational gain in using the metamodel increases. The modified method has potential to be used in problems outside the area of aeroelasticity as well. Note that the Gauss quadrature in a standard tensor product grid is best when the stochastic dimension of the problem — characterized by the number of independent random variables in the design space — is low. As this dimensionality grows, the current method can be extended in two ways: First, by using a dimensional reduction via the same sensitivity analysis. Second, by using a sparse grid instead of the tensor product Gauss quadrature. These issues remain as a topic of further exploration.

References

- Abramowitz M, Stegun I (1984) Handbook of mathematical functions with formulas, graphs, and mathematical tables. John Wiley & Sons Inc.
- Allen M, Maute K (2005) Reliability-based shape optimization of structures undergoing fluid-structure interaction phenomena. *Comput Methods Appl Mech Eng* 194(30):3472–3495
- Aoues Y, Chateauneuf A (2010) Benchmark study of numerical methods for reliability-based design optimization. *Struct Multidiscip Optim* 41(2):277–294
- Arora JS (2006) Introduction to Optimum Design. Academic Press. An imprint of Elsevier
- Au SK, Beck JL (2001) Estimation of small failure probabilities in high dimensions by subset simulation. *Probabilistic Engineering Mechanics* 16(4):263–277
- Beran PS, Pettit CL, Millman DR (2006) Uncertainty quantification of limit-cycle oscillations. *J Comput Phys* 217(1):217–247
- Bisplinghoff RL, Ashley H, Halfman RL (2013) Aeroelasticity. Courier Dover Publications
- Blatman G, Sudret B (2010) An adaptive algorithm to build up sparse polynomial chaos expansions for stochastic finite element analysis. *Probabilistic Engineering Mechanics* 25(2):183–197
- Cheng J, Jiang JJ, Xiao RC (2003) Aerostatic stability analysis of suspension bridges under parametric uncertainty. *Eng Struct* 25(13):1675–1684
- Choi SK, Grandhi RV, Canfield RA, Pettit CL (2004) Polynomial chaos expansion with latin hypercube sampling for estimating response variability. *AIAA J* 42(6):1191–1198
- Choi SK, Grandhi RV, Canfield RA (2007) Reliability-based structural design. Springer
- Chowdhury R, Rao BN, Prasad MA (2009) High-dimensional model representation for structural reliability analysis. *Commun Numer Methods Eng* 25(4):301–337
- Coelho RF, Lebon J, Bouillard P (2011) Hierarchical stochastic metamodels based on moving least squares and polynomial chaos expansion. *Struct Multidiscip Optim* 43(5):707–729
- Ditlevsen O, Madsen OH (1996) Structural reliability methods. John Wiley and Sons
- Dowell EH, Peters DA, Clark R, Scanlan R, Cox D, Simiu E, Curtiss HJ, Sisto F, Edwards JW, Strganac TW, Hall KC (2004) A Modern course in aeroelasticity. Kluwer Academic Publishers, Dordrecht
- Eldred MS, Burkardt J (2009) Comparison of non-intrusive polynomial chaos and stochastic collocation methods for uncertainty quantification. *AIAA Paper* 976(2009):1–20
- Geuzaine P, Brown G, Harris C, Farhat C (2003) Aeroelastic dynamic analysis of a full F-16 configuration for various flight conditions. *AIAA J* 41(3):363–371
- Ghanem R, Spanos P (2003) Stochastic finite elements: A spectral approach. Dover Publications, Revised edn.
- Hu C, Youn BD (2011a) Adaptive-sparse polynomial chaos expansion for reliability analysis and design of complex engineering systems. *Struct Multidiscip Optim* 43(3):419–442
- Hu C, Youn BD (2011b) An asymmetric dimension-adaptive tensor-product method for reliability analysis. *Struct Saf* 33(3):218–231
- Kareem A. (2008) Numerical simulation of wind effects: a probabilistic perspective. *J Wind Eng Ind Aerodyn* 96(10):1472–1497
- Kim NH, Wang H, Queipo NV (2006) Efficient shape optimization under uncertainty using polynomial chaos expansions and local sensitivities. *AIAA J* 44(5):1112–1115
- Kuschel N, Rackwitz R (1997) Two basic problems in reliability-based structural optimization. *Mathematical Methods of Operations Research* 46(3):309–333
- Le Maî OP, Scanlan RH, Knio OM (2003) Estimation of the flutter derivatives of an NACA airfoil by means of Navier–Stokes simulation. *Journal of Fluids and Structures* 17(1):1–28
- Liang J, Mourelatos ZP, Nikolaidis E (2007) A single-loop approach for system reliability-based design optimization. *J Mech Des* 129(12):12151224
- Librescu L, Maalawi KY (2009) Aeroelastic design optimization of thin-walled subsonic wings against divergence. *Thin-Walled Struct* 47(1):89–97
- Manohar CS, Gupta S (2005) Modeling and evaluation of structural reliability: current status and future directions. In: Jagadish KS, Iyengar RN (eds) Recent Advances in Structural Engineering. University Press, Hyderabad, pp 90–187

- MathWorks (2013). Matlab. <http://www.mathworks.com/products/matlab/>
- Matthies HG, Brenner CE, Bucher CG, Soares CG (1997) Uncertainties in probabilistic numerical analysis of structures and solids - stochastic finite elements. *Struct Saf* 19(3):283–336
- Maute K, Weickum G, Eldred M (2009) A reduced-order stochastic finite element approach for design optimization under uncertainty. *Struct Saf* 31(6):450–459
- Ng LWT, Eldred MS (2012) Multifidelity uncertainty quantification using non-intrusive polynomial chaos and stochastic collocation. In: 53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, 23 - 26 April, Honolulu, Hawaii
- Okazawa S, Oideb K, Ikeda K, Terada K (2002) Imperfection sensitivity and probabilistic variation of tensile strength of steel members. *Int J Solids Struct* 39(2):1651–1671
- Oladyshkin S, Nowak W (2012) Data-driven uncertainty quantification using the arbitrary polynomial chaos expansion. *Reliab Eng Syst Saf* 106:179–190
- Paiva RM, Crawford C, Suleman A (2014) Robust and reliability-based design optimization framework for wing design. *AIAA J* 52(4):711–724
- Pettit CL (2004) Uncertainty quantification in aeroelasticity : Recent results and research challenges. *J Aircr* 41(5):1217–1229
- Rackwitz R (2001) Reliability analysis - a review and some perspectives. *Struct Saf* 23(4):365–395
- Rao SS (2008) *Engineering Optimization: Theory and Practice*, New Age International (P) Limited Publishers
- Scanlan RH, Tomko JJ (1971) Airfoil and bridge deck flutter derivatives. *J Eng Mech Div* 97(6):1717–1737
- Schenk CA, Schuëller GI (2005) *Uncertainty assessment of large finite element systems*. Springer, Berlin/Heidelberg/New York
- Schuëller GI, Jensen HA (2008) Computational methods in optimization considering uncertainties - an overview. *Comput Methods Appl Mech Eng* 198(1):2–13
- Seo DW, Caracoglia L (2011) Estimation of torsional-flutter probability in flexible bridges considering randomness in flutter derivatives. *Eng Struct* 33(8):2284–2296
- Sobol' IM (2001) Global sensitivity indices for nonlinear mathematical models and their monte carlo estimates. *Math Comput Simul* 55(13):271–280
- Stanford B, Beran P (2012) Computational strategies for reliability-based structural optimization of aeroelastic limit cycle oscillations. *Struct Multidiscip Optim* 45(1):83–99
- Valdebenito MA, Schuëller GI (2010) A survey on approaches for reliability-based optimization. *Struct Multidiscip Optim* 42(5):645–663
- Wang X, Hirsch C, Liu Z, Kang S, Lacor C (2013) Uncertainty-based robust aerodynamic optimization of rotor blades. *Int J Numer Methods Eng* 2:111–127
- Wei DL, Cui ZS, Chen J (2008) Uncertainty quantification using polynomial chaos expansion with points of monomial cubature rules. *Comput Struct* 86(23):2102–2108
- Witteveen JA, Loeven A, Sarkar S, Bijl H (2008) Probabilistic collocation for period-1 limit cycle oscillations. *J Sound Vib* 311(1):421–439
- Xiong F, Xue B, Yan Z, Yang S (2011) Polynomial chaos expansion based robust design optimization. *Quality, Reliability, Risk, Maintenance, and Safety Engineering (ICQR2MSE)*, International Conference on IEEE, pages 868–873
- Xiu D, Karniadakis G (2003) Modeling uncertainty in flow simulations via generalized polynomial chaos. *J Comput Phys* 187(1):137–167
- Xiu D, Lucor D, Su CH, Karniadakis GE (2002) Stochastic modeling of flow-structure interactions using generalized polynomial chaos. *Journal of Fluid Engineering* 124(1):51–59
- Zhang Y (2013) *Efficient uncertainty quantification in aerospace analysis and design*. PhD Thesis. Missouri University of Science and Technology