

Efficient Stochastic Optimization using Chaos Collocation Method with modeFRONTIER

Valentino Pediroda, Lucia Parussini and Carlo Poloni

University of Trieste

Sumeet Parashar and Nader Fateh

ESTECO North America Inc

Mauro Poian

ESTECO srl

Copyright © 2008 SAE International

ABSTRACT

Robust Design Optimization (RDO) using traditional approaches such as Monte Carlo (MC) sampling requires tremendous computational expense. Performing a RDO for problems involving time consuming CAE analysis may not even be possible within time constraints. In this paper a new stochastic modeling technique based on chaos collocation method is used to measure the mean and standard deviation (σ) for uncertain output parameters. For a given accuracy, chaos collocation method requires far less sample evaluations compared to MC. The efficient evaluation of mean and std. deviation terms using chaos collocation method makes it quite attractive to be used with RDO methods. In this work the RDO of an automotive engine design is performed employing chaos collocation method. The solution strategy is implemented in commercial Process Integration and Design Optimization (PIDO) software tool *modeFRONTIER*. *modeFRONTIER* provides a very effective environment to apply multi-objective optimization algorithms to various CAE or in-house analysis and simulation tools. The engine design simulations were performed using GT-Power through *modeFRONTIER*. The chaos collocation method is coded in MATLAB scripts that are also invoked through *modeFRONTIER*. The rest of the paper covers an introduction describing the motivation and challenges. The chaos collocation method is described followed by a description of its application through *modeFRONTIER*. The engine design optimization problem is explained followed by a discussion of RDO results.

INTRODUCTION

Designing a robust engineering system to give a stable performance under uncertain operating conditions is not always easy if not impossible. Part of the challenge

arises due to the fact that predicting and searching a robust design could become quite a computationally expensive task. Typical approaches for uncertainty quantification include use of Monte Carlo sampling assuming a probabilistic distribution for uncertain parameters. Variation of output parameters could be quantified as mean and standard deviation based on evaluation at sampled points. For a nonlinear problem, quantifying uncertainty with a given accuracy would require bigger sample size and hence more computation.

Robust Design Optimization (RDO) methods try to find designs that are insensitive to uncertain parameters rather than controlling the cause of uncertainty. Optimization algorithms try to achieve this by formulating one of the objectives as minimizing standard deviation of system response under uncertainty. Quantifying the uncertainty in system response at every optimization iteration would require system evaluation at multiple sampled points. Computational expense would increase further as the number of uncertain parameters increase. Hence RDO of design problems involving time consuming CAE analysis may not always be possible without sacrificing accuracy.

UNCERTAINTY QUANTIFICATION WITH CHAOS COLLOCATION METHOD

SETTING THE PROBLEM: STOCHASTIC DIFFERENTIAL EQUATION - In this section the Generalized Polynomial Chaos is explained in order to solve the following stochastic differential equation:

$$L(\mathbf{x}, t, \theta; \phi) = f(\mathbf{x}, t, \theta) \quad (1)$$

where L is a differential operator which contains space and time differentiation and can be non linear and dependent on random parameters θ , $\phi(\mathbf{x}, t, \theta)$ is the solution and function of the space $\mathbf{x} \in \mathbb{R}^d$, time t and random parameters θ ; $f(\mathbf{x}, t, \theta)$ is a space, time and random parameters dependent source term.

THE GENERALIZED POLYNOMIAL CHAOS - Under specific conditions [1], a stochastic process can be expressed as a spectral expansion based on suitable orthogonal polynomial with weights associated with a particular density. The first study in this field is the Wiener process [2][3], which can be written as a spectral expansion in terms of Hermite polynomials with normal distributed input parameters.

The basic idea is to project the variables of the problems on a stochastic space modelled by a set of eigenfunctions Ψ that depend on random variables $\xi(\theta)$, where θ is the random event. For example, the variable ϕ has the following spectral finite dimensional representation:

$$\phi(\mathbf{x}, t, \theta) = \sum_{i=0}^{\infty} \phi_i(\mathbf{x}, t) \Psi_i(\xi(\theta)) \quad (2)$$

In practical terms the Eq.(2) divides the random variables $\phi(\mathbf{x}, t, \theta)$ into a deterministic part, the coefficient $\phi_i(\mathbf{x}, t)$, and a stochastic part, the polynomial chaos $\Psi_i(\xi(\theta))$.

The basis $\{\Psi_i\}$ is a set of orthogonal polynomials with respect to the probability density function of the input parameters. Following the Askey scheme[4], it is possible to introduce the Generalized Polynomial Chaos[5]; thanks to this theory, known also as *Askey-chaos*, for certain input parameter distribution there exist the best representation in terms of convergence rate. For example, for Gaussian random input, we have the Hermite Polynomial Chaos representation, for Gamma distribution the Laguerre representation, for Beta distribution the Jacoby representation, for Uniform distribution the Legendre representation, etc.

In this paper we focus mainly on Gaussian random input, so we represent the variable $\phi(\mathbf{x}, t, \theta)$ in terms of Hermite spectral representation (following the Askey terms):

$$\begin{aligned} \phi(\mathbf{x}, t, \theta) = & \phi_0(\mathbf{x}, t) H_0 + \sum_{i_1=1}^{\infty} \phi_{i_1}(\mathbf{x}, t) H_1(\xi_{i_1}(\theta)) + \\ & \sum_{i_1=1}^{\infty} \sum_{i_2=1}^{i_1} \phi_{i_1 i_2}(\mathbf{x}, t) H_2(\xi_{i_1}(\theta), \xi_{i_2}(\theta)) + \\ & \sum_{i_1=1}^{\infty} \sum_{i_2=1}^{i_1} \sum_{i_3=1}^{i_2} \phi_{i_1 i_2 i_3}(\mathbf{x}, t) H_3(\xi_{i_1}(\theta), \xi_{i_2}(\theta), \xi_{i_3}(\theta)) + \dots \end{aligned} \quad (3)$$

where $H_p(\xi_{i_1}, \dots, \xi_{i_p})$ is the Hermite polynomial of order p in terms of a n -dimensional Gaussian random variable $\xi = (\xi_1, \dots, \xi_n)$ distributed as $N(0, 1)$.

The Hermite polynomial is expressed in general form by:

$$H_p(\xi_{i_1}, \dots, \xi_{i_p}) = e^{\frac{1}{2} \xi^T \xi} (-1)^p \frac{\partial^p}{\partial \xi_{i_1} \dots \partial \xi_{i_p}} e^{-\frac{1}{2} \xi^T \xi} \quad (4)$$

and for one-dimensional case:

$$H_0 = 1, \quad H_1 = \xi, \quad H_2 = \xi^2 - 1, \quad H_3 = \xi^3 - 3\xi, \quad \dots \quad (5)$$

For practical cases, the series has to be truncated to a finite numbers of terms, here denoted with N . So the Eq.(2), using the one-to-one correspondence between the function $H_p(\xi_{i_1}, \dots, \xi_{i_p})$ and $\Psi_p(\xi)$ for Gaussian random input, becomes:

$$\phi(\mathbf{x}, t, \theta) = \sum_{i=0}^N \phi_i(\mathbf{x}, t) H_i(\xi) \quad (6)$$

The number of total terms of the series is determined by:

$$N + 1 = \frac{(n + p)!}{n! p!} \quad (7)$$

where n is the uncertainties dimensionality and p is the expansion order.

As an example, for a second order 2-D Hermite polynomial expression, we get the following form:

$$\begin{aligned} \phi(\mathbf{x}, t, \theta) = & \phi_0(\mathbf{x}, t) + \\ & \phi_1(\mathbf{x}, t) \xi_1(\theta) + \phi_2(\mathbf{x}, t) \xi_2(\theta) + \\ & \phi_3(\mathbf{x}, t) (\xi_1^2(\theta) - 1) + \phi_4(\mathbf{x}, t) (\xi_2^2(\theta) - 1) + \phi_5(\mathbf{x}, t) \xi_1(\theta) \xi_2(\theta) \end{aligned} \quad (8)$$

where $\xi_1(\theta)$ and $\xi_2(\theta)$ are the two random independent variables.



CHAOS POLYNOMIAL AND CHAOS COLLOCATION - Substituting the Polynomial Chaos series, given in Eq.(6) for Gaussian random input, into the stochastic differential Eq.(1) we obtain:

$$L\left(\mathbf{x}, t, \theta; \sum_{i=0}^N \phi_i(\mathbf{x}, t) \Psi_i(\xi(\theta))\right) \cong f(\mathbf{x}, t, \theta) \quad (9)$$

The method of Weighted Residuals is adopted to solve this equation. The coefficients $\phi_i(\mathbf{x}, t)$ are obtained by imposing the inner product of the residual with respect to a weight function equal to zero.

If the weight functions are chosen to be the same as the expansion functions Ψ_i we produce Galerkin method. Performing the Galerkin projection on both sides of the equation, the form becomes:

$$\left\langle L\left(\mathbf{x}, t, \theta; \sum_{i=0}^N \phi_i(\mathbf{x}, t) \Psi_i\right), \Psi_j \right\rangle = \langle f(\mathbf{x}, t, \theta), \Psi_j \rangle \quad j = 0, \dots, N \quad (10)$$

If the operator L is non linear, the deterministic set of $N+1$ equation is coupled and this intrusive form is called *Chaos Polynomial*.

If we employ Dirac delta function as weight function we produce Collocation method. Using a Collocation projection on both sides of Eq.(9), we obtain:

$$L(\mathbf{x}, t, \theta_j; \phi) = f(\mathbf{x}, t, \theta_j) \quad j = 0, \dots, N. \quad (11)$$

This formulation is a linear system equivalent to solving a deterministic problem at each grid point; this non-intrusive form is called *Chaos Collocation*.

To reconstruct the stochastic solution $\phi(\mathbf{x}, t, \theta)$, the Eq.(2) is used:

$$E_{PC}(\phi) = \mu_\phi = \phi_0(\mathbf{x}, t) \quad (12)$$

$$Var_{PC}(\phi) = \sigma_\phi^2 = \sum_{i=1}^N \phi_i^2(\mathbf{x}, t) \langle \Psi_i^2 \rangle \quad (13)$$

Here

$$\langle \Psi_i, \Psi_j \rangle = \langle \Psi_i^2 \rangle \delta_{ij} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} e^{-\xi^2/2} \Psi_i(\xi) \Psi_j(\xi) d\xi = 2^i i! \delta_{ij} \quad (14)$$

where δ_{ij} is the Kronecker operator.

The two approaches are based on the same theory, but gives different numerical representations. In practice intrusive Chaos Polynomial method consists in resolution of a coupled system of deterministic equations, non-intrusive Chaos Collocation method consists in resolution of a decoupled system of deterministic equations. It is evident the difficulty to design an efficient intrusive solver, both because of computational cost and because of the obvious handicap to imply an internal modification of the deterministic solver [6]. The non-intrusive methodology has a simpler computational management. A remarkable advantage of this approach is the deterministic solver represents a black-box and there is no need to modify it. This means the non-intrusive method is more versatile than intrusive method.

A still open problem of Chaos Collocation approach [7] is the difficulty to select collocation points: with multi dimensional uncertainties the choice is not unique[8][9]. This problem does not exist for one stochastic parameter, because collocation points are the roots of polynomial of order $p+1$.

In this paper we focus our attention on the comparison between the Chaos Collocation Method and Monte Carlo Method. After analysis of simple comparison results, a Multi Objective Optimization is performed on the test problem using Multi Objective Genetic Algorithm, for reaching good solutions in term of performance as well as stability.

TEST PROBLEM

The chaos collocation method is used with an optimization method to perform a robust design optimization of an automotive engine.

The engine is an automotive 4-stroke, 4 cylinder, 2.2 litre engine with a bore of 93mm, a stroke of 81mm and a compression ratio of 8.9. There are two valves per cylinder and fuel is injected into the intake port. The engine runs in speed mode, where the engine speed is specified and the engine torque is calculated. Combustion is modeled by a Wiebe function (with spark trimming simulated by the 50% burn point parameter), in-cylinder heat transfer is modeled using the Woschni model and in-cylinder wall temperatures are fixed at specified values for WOT (wide open throttle) and 5000 rpm. **GT-Power**, from Gamma Technologies (www.gtisoft.com), is used to perform the engine simulation. Figure 1 shows the engine simulation model as built in GT-power.

The optimization objectives were: minimize emissions (NOX), and minimize fuel consumption (BSFC). The problem involved four design variables (parameters): intake cam timing, exhaust cam timing, intake runner length and 50% burn point.