Error Estimation of Polynomial Chaos Approximations in Transient Structural Dynamics

T. Dao, Quentin Serra, Sébastien Berger, Éric Florentin

▶ **To cite this version:**

T. Dao, Quentin Serra, Sébastien Berger, Éric Florentin. Error Estimation of Polynomial Chaos Approximations in Transient Structural Dynamics. International Journal of Computational Methods, 2020, 17 (10), ⟨10.1142/S0219876220500036⟩. ⟨hal-02375432⟩

HAL Id: hal-02375432

https://hal.science/hal-02375432v1

Submitted on 9 Dec 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

HAL Authorization

Error estimation of Polynomial Chaos approximations in transient structural dynamics

T.D. Dao[a], Q. Serra[a], S. Berger[a], E. Florentin[a,1,*]

[a]*INSA CVL, Univ. Orléans, Univ. Tours, LaMé (EA 7494)*

Abstract

Usually, within stochastic framework, a testing dataset is used to evaluate the approximation error between a surrogate model (e.g. a Polynomial Chaos expansion) and the exact model. We propose here another method to estimate the quality of an approximated solution of a stochastic process, within the context of structural dynamics. We demonstrate that the approximation error is governed by an equation based on the residue of the approximate solution. This problem can be solved numerically using an approximated solution, here a coarse Monte Carlo simulation. The developed estimate is compared to a reference solution on a simple case. The study of this comparison makes it possible to validate the efficiency of the proposed method. This validation has been observed using different sets of simulations. To illustrate the applicability of the proposed approach to a more challenging problem, we also present a problem with a large number of random parameters. This illustration shows the interest of the method compared to classical estimates.

Keywords: A posteriori error estimate, Polynomial Chaos Expansion, Structural dynamics

1. Introduction

In the last few years, stochastic methods have caught the attention of engineers and researchers in many scientific domains. Indeed, these methods allow the prediction within a confidence level of the possible values of the response of a stochastic system of interest, when the input parameters may vary or when some randomness may occur. Introducing variabilities has several applications such as risk management, optimisation and robust design, the robust optimal design being sometimes different from the optimal deterministic design [1]. It is an interesting tool to estimate the robustness of a numerical method [2]. The biggest drawback of stochastic problems is the huge numerical cost needed to scan the stochastic space. A review of several stochastic methods has been proposed in [3, 4].

Monte Carlo (MC) simulation [5] is the most direct approach and is often used as a reference. It consists in the evaluation of the quantity of interest (QoI) from the stochastic system on a large sample of realizations. The advantage is that it does not require any complex computer code dedicated to stochastic problems, but its drawback is the slow convergence. Thus, it implies a huge computational cost. Variance reduction techniques, for example like Latin Hypercube sampling (LHS) or quasi Monte Carlo (qMC) simulations have been developed to get a better convergence [6].

Several methodologies based on a spectral expansion have been developed to tackle the cost associated with sampling methods. Perturbation methods are based on a Taylor series expansion around the mean value of parameters. They are limited to the case of small variations of the random field around its mean value, and they cannot be applied to non-smooth functions. The Polynomial Chaos (PC) expansion is a spectral method, consisting in searching an approximation of the stochastic response in the space spanned by a finite basis of orthonormal polynomials. It lies on the Homogeneous Chaos theory started by Wiener [7] in the case of Gaussian random variables. It has been extended later

*Corresponding author.
Email address: eric.florentin@insa-cvl.fr (E. Florentin)
[1]Tel.: 02.48.48.40.88

Reference of published article:
https://www.worldscientific.com/doi/abs/10.1142/S0219876220500036

to non-Gaussian random fields [8, 9] and called *generalized Polynomial Chaos* expansion (gPC). Different interesting applications have been presented in [10, 11, 12, 13]. To determine the coefficients of the expansion, two strategies are classically adopted. An intrusive strategy is obtained when the PC expansion is used together with a Galerkin approach, resulting in a very large numerical system where the unknowns are the amplitudes of the coefficient of the expansion. However, it results in a large computational cost and in the need to write a dedicated program. The second kind of methods is non-intrusive, it only uses as inputs the outputs of a deterministic program. Two approaches are classically used to evaluate the expansion coefficients. The first one among them is called Non-Intrusive Spectral Projection [14] and consists in the numerical evaluation of the projection (scalar product in stochastic space) of the QoI on the space spanned by the polynomial basis. The second one relies on linear regression to compute the best (in a least square sense) coefficients of the PC expansion.

Compared to the static case, in transient dynamics the time can simply be seen as an additional parameter. Using the same basis in each time step is called a time-frozen PC expansion and results in large errors in the evaluation of the long-term response [15]. Several strategies to add physical information to the variation of the spectral coefficients have been proposed in [16, 17]. To control the error made by the PC expansion, which is the distance between the response and its projection in the space spanned by the chosen polynomials, error estimates are needed. Several works investigated the a priori convergence behavior of gPC [15, 18]. Recently, a number of authors have developed different strategies to build adaptive gPC, with an error control based on the variance decrease rate [19], the residual decrease rate [20, 21], or scalar values related to the residue of the least square problem solved in regression such as the R^2 coefficient or the Leave One Out LOO error [22, 23, 24]. However, even if relevant, such indicators only provide a qualitative information on the quality of the meta-model, and thus need to fix some tolerance value arbitrarily. Another possibility, often used in statistical learning, consists in measuring the empirical error (Mean Square Error MSE) on a testing dataset. However, the small size of this dataset leads to a non-robust estimate of the MSE.

As far as the authors know, no a posteriori error estimate has been proposed yet to quantify, even approximately, the distance between the response and the PC metamodel in the transient dynamic framework. It is a problem especially because the Polynomial Chaos suffers from the curse of dimensionality, with a computational cost becoming unaffordable for expensive full-scale industrial finite element problems with numerous random parameters. In such a case, only low-order PC approximations are affordable, no a priori tolerance is known to set a value for LOO error, so quantitative a posteriori error estimates are of first importance to evaluate the degree of validity of the metamodel. In this paper, we propose to use a coarse Monte Carlo simulation to solve a stochastic equation associated to the reference problem. The goal is to evaluate at a low numerical cost the error due to the PC expansion. The manuscript is organized as follows. Theories about MC simulation and PC expansion are recalled in sections 2, an example consisting in a two degrees of freedom (dof) problem is presented. The error estimation strategy is presented in section 3. Additional numerical examples are provided in section 4 before concluding in section 5.

2. Usual algorithms for solving linear transient dynamic problems with parametric uncertainties

2.1. Reference problem: mechanical system with parametric uncertainties

In a general way, assuming that the problem has been discretized in a finite number (n) of discrete unknowns, the dynamic equilibrium of a linear mechanical system can be written for each time $t \in [0, t_f]$ as:

$$\begin{cases} \mathbf{M}\underline{\ddot{x}}(t) + \mathbf{C}\underline{\dot{x}}(t) + \mathbf{K}\underline{x}(t) = \underline{F}(t) \\ \underline{x}(t=0) = \underline{x}_0 \\ \underline{\dot{x}}(t=0) = \underline{\dot{x}}_0 \end{cases} \tag{1}$$

where $\mathbf{M}$, $\mathbf{C}$ and $\mathbf{K}$ are respectively the $(n \times n)$ mass matrix, damping matrix and stiffness matrix of the discrete system, $\underline{x}$ is the n-vector of displacements which are unknown, $\underline{x}_0$ and $\underline{\dot{x}}_0$ are respectively the initial displacement and initial velocity, $\underline{F}$ is the vector of loads and t_f the final time. The dots above previous expressions are used in place of time derivatives to alleviate the expression.

In the case of parametric uncertainties, some of the values in the matrices and in the vector of external loads can be random and independent parameters, so that their values observation θ. Consequently, the dynamical response $\underline{x}$ is a random process. In the following, we denote as r the number of random independent parameters. To these parameters are associated the vector of uniform dimensionless parameters $\underline{\xi} \in [-1; 1]^r$. Thus, each the variables appearing in

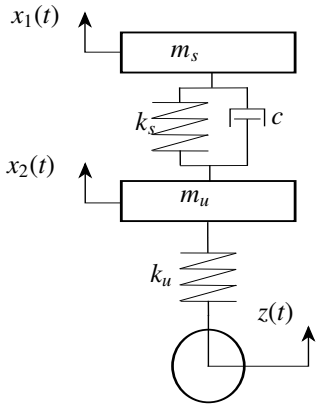


Figure 1: 2-dofs system considered in the study

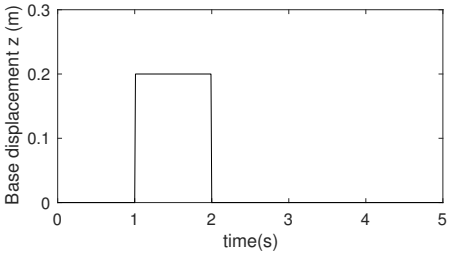


Figure 2: Prescribed base displacement $z(t)$

Eq. (1) can be eventually expressed as a function of the dimensionless parameters $\underline{\xi}$. To alleviate the expressions, the random vector realization $\underline{\xi}(\theta)$ is written $\underline{\xi}$.

As an example, we will consider the following stochastic problem with two degrees of freedom, illustrated in Fig. 1, where the quantity of interest is the random process vector:

$$\underline{x}(t) = [x_1, x_2]^T \tag{2}$$

The mass, damping and stiffness matrices are given by:

$$\mathbf{M} = \begin{bmatrix} m_s & 0 \\ 0 & m_u \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} c & -c \\ -c & c \end{bmatrix}, \mathbf{K} = \begin{bmatrix} k_s(\xi) & -k_s(\xi) \\ -k_s(\xi) & k_s(\xi) + k_u(\xi) \end{bmatrix} \tag{3}$$

The load vector consists in a prescribed base displacement $z(t)$, shown in Fig. 2:

$$F(t) = [0, k_u z(t)]^T \tag{4}$$

The initial conditions are taken equal to zero. Values of the parameters are provided in Tab. 1.

Parameter	Probability law	Interval
k_s	Uniform	[360 . . . 440] N/m
k_u	Uniform	[1800 . . . 2200] N/m
m_s	Determinist	20kg
m_u	Determinist	40kg
c	Determinist	4Ns/m
z_{max}	Determinist	0.2m
$\underline{x}_0$	Determinist	[0,0]
$\underline{\dot{x}}_0$	Determinist	[0,0]

Table 1: Values of input parameters

2.2. Notations

This section aims at defining the notations appearing in the following parts of the paper. The following notations are used for the mean $E[\bullet]$ and the variance $V[\bullet]$ of any random vector:

$$\mu_\bullet = E[\bullet] \quad ; \quad \sigma_\bullet = \sqrt{V[\bullet]} \tag{5}$$

These quantities depend on time. We also need to define time-averaged values of a function defined over $[0, t_f]$. We use as time-average function the operator $\widetilde{\bullet}$ defined for any function of time by:

$$\widetilde{\bullet} = \frac{1}{t_f} \int_0^{t_f} \bullet dt \tag{6}$$

2.3. Polynomial Chaos approximation

In order to reduce the CPU cost when solving stochastic problems, the Polynomial Chaos approximation is often used to solve the stochastic problem associated to Eq. (1). A polynomial chaos approximation of a random process consists in the projection of the exact random process in the space spanned by a family of polynomials. Thus, for any realizations of the vector of input parameters, the QoI of the metamodel is computed through a linear combination of polynomials [25, 26, 27]. Thus the exact realizations of the random variable $\underline{x}$ are approximated for the same parameters by the realizations of the approximate random variable $\underline{x}_{PC}$ such as:

$$\underline{x}(\underline{\xi}, t) \approx \underline{x}_{PC}(\underline{\xi}, t) = \sum_{j=1}^{n_{PC}} \overline{\underline{X}}_j(t)\phi_j(\underline{\xi}) \tag{7}$$

where ϕ_j are the chaos polynomials and $\overline{\underline{X}}_j$ are the n_{PC} chaos coefficients. Although theoretically any infinite family makes it possible to represent exactly the random process, numerically a finite family is needed, so that there is a relationship between the choice of the polynomial family and the convergence of the expansion Eq. (7). An optimal choice of the polynomial family depending on the distribution of the random input parameter $\underline{\xi}$ is recalled in Tab. 2 for continuous distributions [8].

Random variable ξ	Wiener-Askey PC $\phi(\xi)$	Support
Gaussian	Hermite	$(-\infty, +\infty)$
Uniform	Legendre	[a,b]
Gamma	Legendre	$[0, \infty)$
Beta	Jacobi	[a,b]

Table 2: Correspondence between the distribution of the input parameter and the optimal choice of the polynomial basis

Several methods exist to estimate the chaos coefficients $\overline{\underline{X}}_j$. Non-intrusive methods are often preferred because they do not require a dedicated computational code, but only repeated access to a deterministic code. Among non-intrusive versions can be found the Non-Intrusive Spectral Projection and the regression method. We refer the interested reader to [14, 23, 28] for details. Because it is often used in recent literature, we only present the regression approach here. The regression approach consists in searching an optimal set of chaos coefficients $\overline{\underline{X}}_j$ in the least square sense:

$$\left\{\overline{\underline{X}}_0(t), \overline{\underline{X}}_1(t), ..., \overline{\underline{X}}_{n_{PC}}(t)\right\} = \underset{\left\{\underline{X}_0(t), \underline{X}_1(t), ..., \underline{X}_{n_{PC}}(t)\right\}}{\text{Argmin}} \left[\frac{1}{Q}\sum_{q=1}^{Q}\left(\underline{x}(\underline{\xi_q}, t) - \sum_{j=1}^{n_{PC}} \underline{X}_j(t)\phi_j(\underline{\xi_q})\right)^2\right] \tag{8}$$

where $\{\underline{\xi_1}, \underline{\xi_2}, ..., \underline{\xi_Q}\}$ are the Q realizations of the input parameters.

This minimisation is a classical least square minimisation. Introducing the so-called information matrix $\mathbf{\Phi}$ which is the matrix of the polynomials evaluated at the sampling points:

$$\mathbf{\Phi} = (\Phi_{qj})_{q\in\{1..Q\}, j\in\{1..N\}} \text{ with } \Phi_{qj} = \phi_j(\underline{\xi_q}), \tag{9}$$

finding the solution results in computing the pseudo inverse of $\mathbf{\Phi}$ denoted $\mathbf{\Phi}^+$:

$$\mathbf{\Phi}^+ = \left(\mathbf{\Phi}^T\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^T \tag{10}$$

Finally, the number of coefficients is determined by the number of random input parameters r and the maximal order p fixed for each polynomial. Thus, a full tensor product of monovariate polynomials leads to $n_{PC} = (p+1)^r$ polynomials.

Sparse truncation techniques [28] are often used to reduce computational burden. We will only consider the classical truncation scheme here, which conduces to a reduced number of polynomials:

$$n_{PC} = \frac{(p+r)!}{p!r!}. \tag{11}$$

With the regression method, a minimum of $Q = n_{PC}$ simulations are necessary, which may be performed in practice at the $Q = (p+1)^r$ Gauss collocation points or at points chosen with a Latin Hypercube Samples (LHS) method [29]. In the latter case, $Q = kn_{PC}$ simulations (with k a small integer usually equal to 2, 3 or 4) are used.

In this work, we do not aim at reducing the cost of the surrogate model, but we aim at predicting the estimation error within a reasonable cost.

2.3.1. Error due to the PC approximation

Using $\underline{x}_{PC}$ as an approximation of $\underline{x}$ following Eq. (7) leads to an error defined as:

$$\underline{e} = \underline{x}_{PC} - \underline{x} \tag{12}$$

A measure of the error, called "generalization error", can be defined:

$$\varepsilon^2 = E[\underline{e}^T \underline{e}] \tag{13}$$

This global measure of the Mean Square Error corresponds to the sum of the local contribution ε_i^2 of each component of error e_i:

$$\varepsilon^2 = \sum_{i=1}^{n} \varepsilon_i^2 = \sum_{i=1}^{n} \mu_{e_i^2} \tag{14}$$

Several error indicators have been used in the literature to estimate this error with a reasonable cost. Among them, we choose two estimates. The first is the empirical error on a training dataset, because it is intrinsic to the PC approximation and is naturally available directly with the results. Therefore it is largely used in the literature, in particular in work based on adaptive schemes [30]. The second one is the The Leave One Out error which is a particular case of the k-fold cross validation techniques. It is also very practical to implement. Let us recall that we are interested into a cheap error estimates that can be easily implemented into industrial softwares.

The empirical error on training dataset. The Empirical Error (EE) is a measure of the residue of the minimization Eq. (8). The idea is to approximate the empirical error using the different Q sampling points defined in the PC approximation. The empirical error ε_{emp} is then simply computed at a very low cost using:

$$\varepsilon_{emp}^2 = \frac{1}{Q} \sum_{q=1}^{Q} \left(\underline{x}(\underline{\xi}_q) - \underline{x}_{PC}(\underline{\xi}_q) \right)^2 = \frac{1}{Q} \sum_{q=1}^{Q} \underline{e}(\underline{\xi}_q)^T \underline{e}(\underline{\xi}_q) \tag{15}$$

It can be remarked here that the regression approach of the PC naturally minimizes the empirical error. This estimate can also be seen as a coarse Monte Carlo approximation of ε using the sampling points.

Local error estimates $\varepsilon_{emp,i}$ can be used to have some information about the local quality of the i^{th} component of $\underline{x}$. We have:

$$\varepsilon_{emp}^2 = \sum_{i=1}^{n} \varepsilon_{emp,i}^2 \quad \text{with} \quad \varepsilon_{emp,i}^2 = \frac{1}{Q} \sum_{q=1}^{Q} e_i^2(\underline{\xi}_q) \tag{16}$$

The Leave One Out error. The Leave One Out (LOO) error [31, 32] is a measure of the error derived from cross-validation techniques. Considering first a set of Q deterministic samples, it is possible to take $Q-1$ samples arbitrarily - that is a set $\{1, ..., Q\} \backslash q$ - and to derive the associated PC expansion: $\underline{x}_{PC}^{(\backslash q)}$.

The error on the sample $\underline{\xi}_q$ between the new PC approximation and its actual value is thus given by:

$$\underline{\Delta}(\underline{\xi}_q) = \underline{x}_{PC}^{(\backslash q)}(\underline{\xi}_q) - \underline{x}(\underline{\xi}_q) \tag{17}$$

The total leave one out error is then computed by taking the root mean square value of the errors $\underline{\Delta}(\underline{\xi_q})$ according to:

$$\varepsilon_{LOO}^2 = \frac{1}{Q}\sum_{q=1}^{Q} \underline{\Delta}(\underline{\xi_q})^T \underline{\Delta}(\underline{\xi}_q) \tag{18}$$

The LOO error estimate can be obtained by post-processing one single run of Polynomial Chaos as [33]:

$$\varepsilon_{LOO}^2 = \sum_{i=1}^{n} \varepsilon_{LOO,i}^2 \quad \text{with} \quad \varepsilon_{LOO,i}^2 = \frac{1}{Q}\sum_{q=1}^{Q} \frac{1}{h_q^2} e_i^2(\underline{\xi_q}) \tag{19}$$

where h_q are the diagonal terms of the matrix :$\mathbf{H} = \mathbf{Id} - \mathbf{\Phi}\mathbf{\Phi}^+$ and where $\mathbf{Id}$ is the identity matrix. We can remark that the empirical and LOO errors only differ by the h_q ponderation.

The interest of ε_{LOO} or ε_{emp} is that their calculation is based only on the values of the QoI in the training dataset, so the numerical cost is very low. However, they only provide qualitative information on the level of error, as it will be illustrated in the next section. An arbitrary tolerance values has to be fixed to evaluate if the PC expansion approximates the random variable correctly.

The empirical error on a testing dataset. A third measure is very classical in statistical learning. It is another empirical measure, based on a coarse Monte-Carlo method. One can remark that the empirical error based on the training dataset (values used to compute the surrogate) is minimal, because the surrogate coefficients are computed through a least-square minimization of this empirical error. So, one can think that these points are not appropriate to evaluate the error. The idea is to measure an empirical error on arbitrary random points in the stochastic space, different from the training points. The cost is much larger, because the exact response of the QoI is needed on the testing dataset. In typical applications the following recipe is often applied: 70% of the sampling points are affected to training, and 30% are affected to tests. This way of measuring the error is more costly, but leads to error levels much more realistic than with the empirical error on training datasets. However, the inherent question is the choice of the sampling points, and its consequence on the variability of the error estimate.

2.4. Reference solution : Monte Carlo estimation of mean response

Monte-Carlo method is a sampling based technique, resulting from the law of large numbers. If a large number of realizations θ are observed, the statistical moments converge toward the exact moments. Thus, it is often called 'brute force' because it consists simply in the evaluation of the deterministic problem for a large number of realizations. If all the parameters $\underline{\boldsymbol{\xi}}$ are independent, and for a large number of realizations θ, the central limit theorem provides a bound of the error between the first actual moment and the statistical mean computed with n_{MC} realizations $\underline{\boldsymbol{\xi}}_i$, $i \in \{1, 2, \dots n_{MC}\}$.

Using the n_{MC} realizations of x_1, the first component of $\underline{x}$, one can compute its empirical mean $\mu_{x_1}^{emp,n_{MC}}$ and its empirical standard deviation $\sigma_{x_1}^{emp,n_{MC}}$ as follows:

$$\mu_{x_1}^{emp,n_{MC}} = \frac{1}{n_{MC}}\sum_{i=1}^{n_{MC}} x_1(\underline{\xi}_i) \tag{20}$$

and:

$$(\sigma_{x_1}^{emp,n_{MC}})^2 = \frac{1}{n_{MC}-1}\sum_{i=1}^{n_{MC}} (x_1(\underline{\xi}_i) - \mu_{x_1}^{emp,n_{MC}})^2 \tag{21}$$

The difference $e_1 = |\mu_{x_1} - \mu_{x_1}^{emp,n_{MC}}|$ represents the error on the mean of x_1 due to the Monte Carlo estimation and satisfies:

$$|\mu_{x_1} - \mu_{x_1}^{emp,n_{MC}}| \leq C_{MC}\frac{\sigma_{x_1}}{\sqrt{n_{MC}}} \tag{22}$$

where C_{MC} is a constant depending on the level of confidence desired, and following Student's law - typically the value $C_{MC} = 1.96$ corresponding to a 95% confidence interval is chosen associated to a normal distribution when

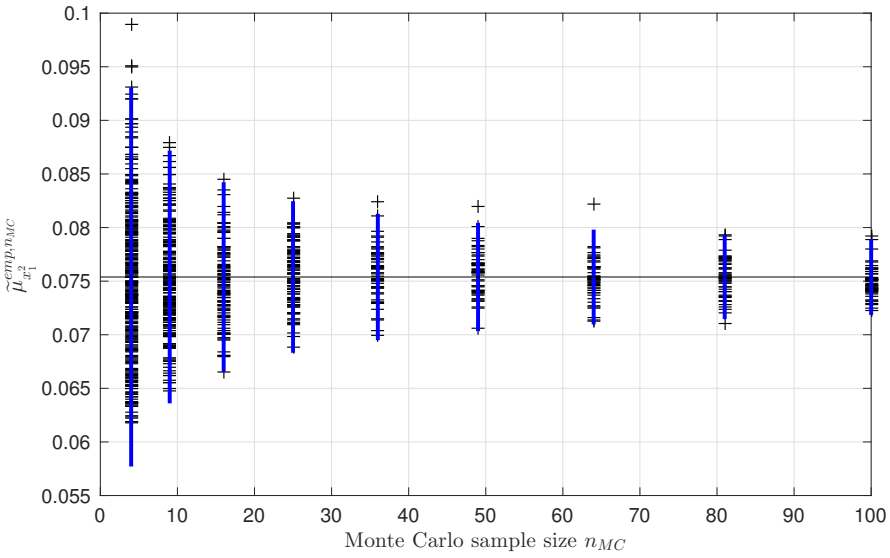


Figure 3: Variability of a Monte Carlo estimation of $\widetilde{\mu}_{x_1^2}^{emp,n_{MC}}$. Reference value $\widetilde{\mu}_{x_1^2}$: black line.Confidence interval $\widetilde{\mu}_{x_1^2} \pm \epsilon_{95\%}$: blue vertical lines. Coarse Monte Carlo simulations : black +

a large number of samples are used. An approximate bound is derived by replacing in Eq. (22) the actual standard deviation σ_{x_1} by the empirical standard deviation $\sigma_{x_1}^{emp,n_{MC}}$, providing the Monte Carlo error estimate $\epsilon_{95\%}$:

$$\epsilon_{x_1,95\%} = 1.96 \frac{\sigma_{x_1}^{emp,n_{MC}}}{\sqrt{n_{MC}}} \tag{23}$$

The convergence rate (here in $\sqrt{n_{MC}}$) obtained by sampling randomly the space of random parameters $\underline{\xi}$ can be improved by using quasi-random sequences such as a Sobol's sequence, a Halton's sequence, or by using a Latin Hypercube discretization of the random space [34, 35].

In the following, because the actual moments μ_{x_1},μ_{x_2} are not known, they are computed approximately with a Monte Carlo method using $n_{MC} = 10^6$. Thus, in the following, we consider the following equality as true for each quantity $\bullet$:

$$\mu_\bullet = \mu_\bullet^{emp,10^6} \tag{24}$$

On Fig. 3 we compare the time-averaged mean value of the square of the first degree of freedom x_1, that is $\widetilde{\mu}_{x_1^2}^{emp,n_{MC}}$ to its converged value $\widetilde{\mu}_{x_1^2}$ for different (small) values of the number of samples n_{MC}.

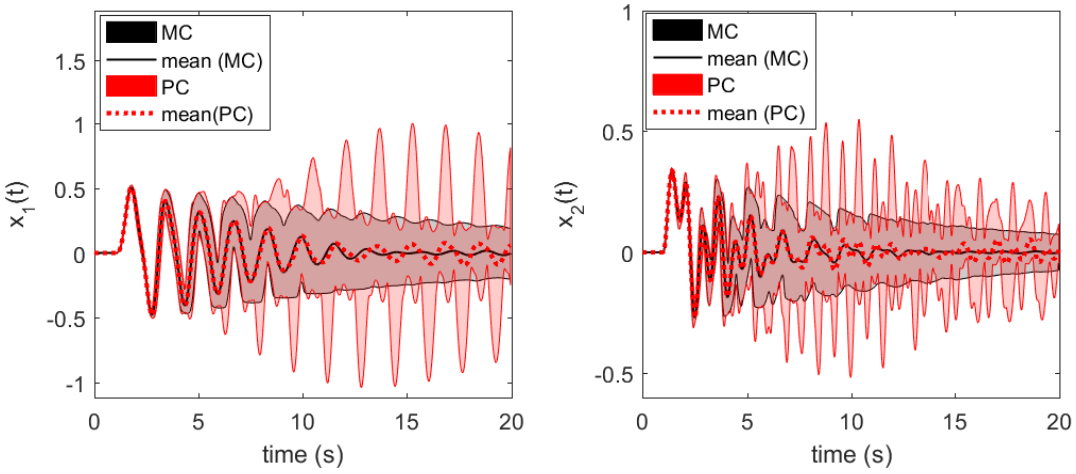
Different values of $\widetilde{\mu}_{x_1^2}^{emp,n_{MC}}$ are obtained when different samples are used. Consequently, even when the number of samples is fixed the empirical mean may vary. In Fig. 3, for a given number of realization n_{MC}, 100 different sets of n_{MC} realizations are computed and lead to different approximations $\widetilde{\mu}_{x_1^2}^{emp,n_{MC}}$ of the mean $\widetilde{\mu}_{x_1^2}$. Each value is presented by a + on Fig. 3. The value of $\widetilde{\mu}_{x_1^2}$ is 0.07505.

In blue, a bar corresponding to the approximation $\widetilde{\mu}_{x_1^2} \pm \epsilon_{x_1^2,95\%}$ is plotted for each value of n_{MC}.

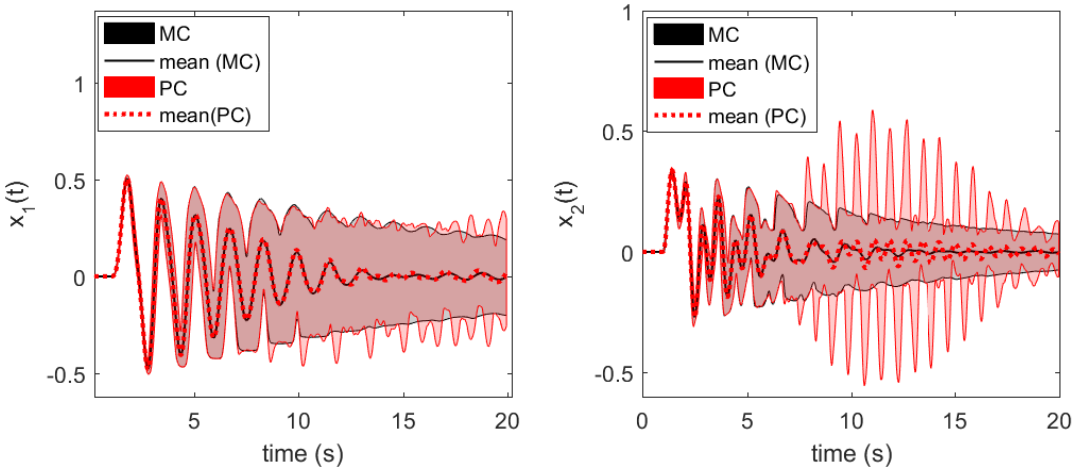
On Fig. 3 it can be observed that even for a small number n_{MC}, the error estimate $\epsilon_{x_1^2,95\%}$ allows to predict accurately the variability in the Monte Carlo estimate.

2.5. Example of the Monte Carlo estimation of mean response for PC expansion

To illustrate the previous section, results of simulation are provided in time domain. We compare here very fine Monte Carlo MC result ($n_{MC} = 10^6$) to Polynomial Chaos (PC) for several orders of polynomials p. The QoI is the displacement of the car $x_1(\xi, t)$.



(a) p=2



(b) p=4

Figure 4: Comparison between MC and PC of mean of QoI and 95% confidence interval of the two degrees of freedom.

Fig. 4 presents the mean of the QoI (thick line) as well as the 95% confidence interval of the QoI as a function of the time.The mean of the PC surrogate is exactly obtained with the first term of the expansion. It can be observed that both orders lead to a good superposition of both the mean of the QoI and the confidence interval. Furthermore we observe the well known result stating that in time simulations, time-frozen PCE lose accuracy as time increases [15] . Indeed, superposition is observed until 4s with p=2 while superposition occurs until 6s with p=4.

In Tab. 3 the amount of deterministic simulation used for both MC and PC are given. For PC, it corresponds to the roots of the chosen Legendre polynomials.

	MC	p=1	p=2	p=3	p=4
Number of computations	10^6	4	9	16	25

Table 3: Number of deterministic samples.

Convergence of the PC approach can be observed in Fig. 5, which shows the value of the time-averaged generalization error $\widetilde{\mu}_{e^2}$ (reference computed by Monte-Carlo method with 10^6 samples) for different polynomial orders.

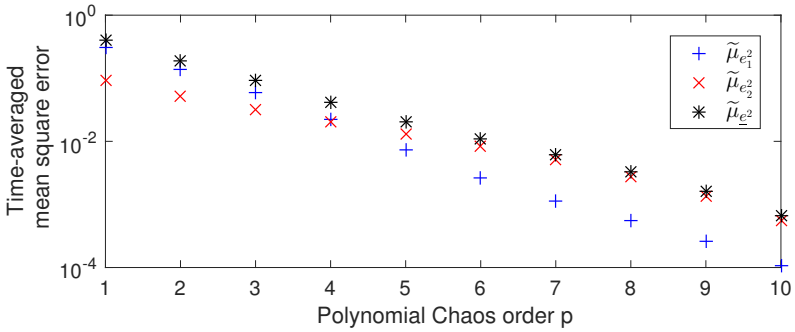
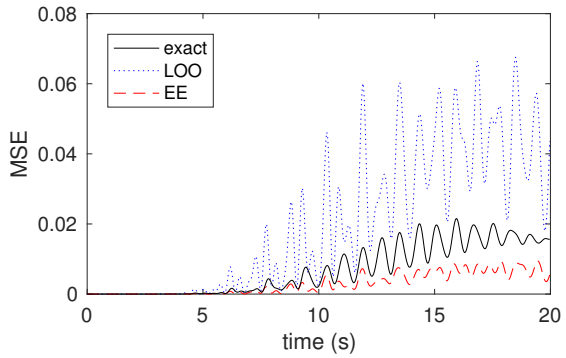


Figure 5: Convergence of Polynomial Chaos.

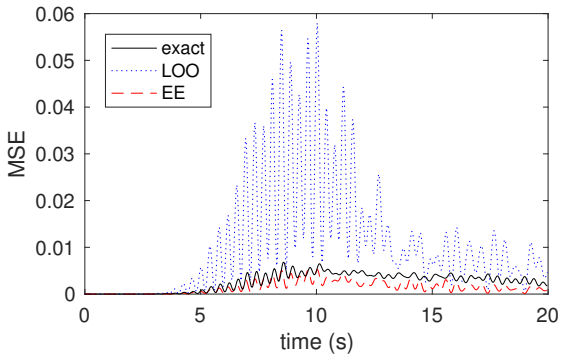
Thus, PC makes it possible to predict accurately the response of the stochastic random variables with a computational cost several order lower than classical Monte Carlo techniques.

Finally, classical error indicators are compared to exact error ε^2 in Fig. 6 for p=2.

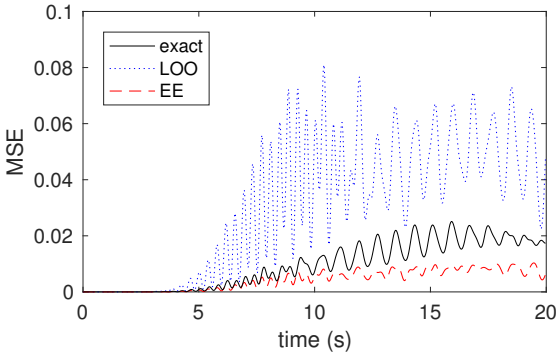
It can be observed, that neither the qualitative or quantitative agree with the generalization error. In the following, the objective is to show if another method can predict more accurately the error, especially for low polynomial orders, at a cost reasonably higher than LOO.



(a) $p = 2$, dof x_1



(b) $p = 2$, dof x_2



(c) $p = 2$, all dofs

Figure 6: Evolutions over time of different error estimates.

3. Estimation of the a posteriori error through the resolution of a coarse Monte Carlo problem

We have seen in the previous section that classical error measures are not perfectly adequate because they only provide a qualitative value of the generalization error. In this part, we aim at developing an algorithm allowing to predict the numerical value of the generalization error, with a low numerical cost.

3.1. Strategy

Introducing error $\underline{e}$ defined in Eq. (12) into equation Eq. (1) the problem can be written:

$$\mathbf{M}(\underline{\xi})\ddot{\underline{e}}(\underline{\xi},t) + \mathbf{C}(\underline{\xi})\dot{\underline{e}}(\underline{\xi},t) + \mathbf{K}(\underline{\xi})\underline{e}(\underline{\xi},t) = \underline{R}(\underline{\xi},t) \tag{25}$$

where $\underline{R}$ is the residue associated to the PC approximated solution of the dynamic equilibrium:

$$\underline{R}(\underline{\xi},t) = \mathbf{M}(\underline{\xi})\ddot{\underline{x}}_{PC}(\underline{\xi},t) + \mathbf{C}(\underline{\xi})\dot{\underline{x}}_{PC}(\underline{\xi},t) + \mathbf{K}(\underline{\xi})\underline{x}_{PC}(\underline{\xi},t) - \underline{F}(\underline{\xi},t) \tag{26}$$

Thus, the generalization error $\underline{e}$ is solution of a linear differential equation in time of second order. The idea presented in this paper consists in using a coarse Monte Carlo scheme to solve equation Eq. (25). Validation of the approach is carried out by comparison with a reference value obtained with a reference solution.

3.2. Coarse Monte Carlo

We recall that the reference is obtained by setting $n_{MC} = 10^6$, however the associated cost is prohibitive in most practical applications. We propose here to evaluate the mean square error using a coarse Monte Carlo simulation i.e. $n_{MC} << 10^6$. Then, an estimate of the mean square error is defined as:

$$\varepsilon^2_{MC} = \mu_{e^2_{MC},n_{MC}} \tag{27}$$

where e_{MC} is a solution of Eq. (25) on the coarse MC sampling. Its time-averaged value is denoted $\widetilde{\varepsilon}^2_{MC}$. Then, the time-averaged efficiency of the estimate ε^2_{MC} is quantified by the scalar η^2 defined by

$$\eta^2 = \frac{\widetilde{\varepsilon}^2_{MC}}{\widetilde{\varepsilon}^2} \tag{28}$$

We will study local error associated to dof x_1 (resp. x_2), we denote η_1 (resp. η_2) its efficiency. Similarly η_{12} corresponds to the efficiency of the global error estimate.

This quantity can be defined either for local error (i.e. e_i) as well as for global error (i.e. $\underline{e}$) and depends on the number of samples considered and on the choice of the samples.

From Eq. (23), it can be seen that error of Monte Carlo decreases when n_{MC} is large or when the variance of the random variable is low. Fig. 7 presents the time-averaged variance of the response and the time-averaged variance of the error, for each of the dofs and for the global response, computed with a 10^6 random Monte Carlo scheme. It is clear that if the Polynomial Chaos order is high enough, the error is low, and then the variance of the error is much lower than the variance of the response. In such a case, a coarse Monte Carlo approach may be efficient on the error problem Eq. (25), while it is not converged if applied on the initial problem Eq.(1). This is the key point of the present paper.

Finally, an additional parameter subjected to investigation is the choice of points, that is the design of experiment leading to ε^2_{MC}. Three possibilities are tested here: a random choice of sampling (Rand), a Latin Hypercube with random smoothing (LHS), or a deterministic choice based on a uniform sampling (US) in every stochastic direction.We choose these three DoE because of their simplicity and very common use in the field of mechanical engineering. Lots of refined samplings could lead to finer results. We decided here to choose the simplest ones, with the idea to illustrate the methodology with generic sampling schemes. Different variants can be derived from this choice, for example the Sobol, and Halton sampling [36] or the sequential sampling schemes [37]

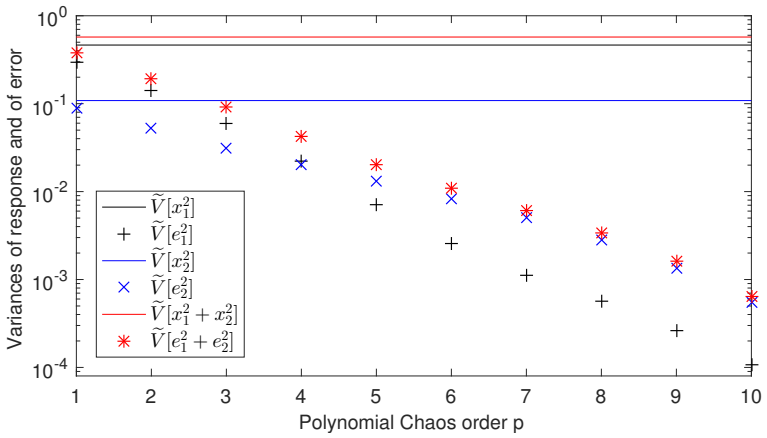


Figure 7: Variances of the response x^2 and of the error e^2.

3.3. Results

The Polynomial Chaos surrogate is generated by using regression approach. We choose as deterministic samples the roots of the polynomials of order $p+1$. Error estimates are presented for several polynomial orders in the following.

Fig. 8 and Fig. 9 present the efficiency η^2 for three choices of sampling when the number of realizations increases.

The meta-model approximation has been carried out with a polynomial order $p = 2$ and $p = 4$ respectively. US being a deterministic sampling, it leads to only one result. On the contrary, the estimates based on LHS or random sampling (Rand) depend on the sampling, so different tries with the same size do not lead to the same value of the predicted estimate.

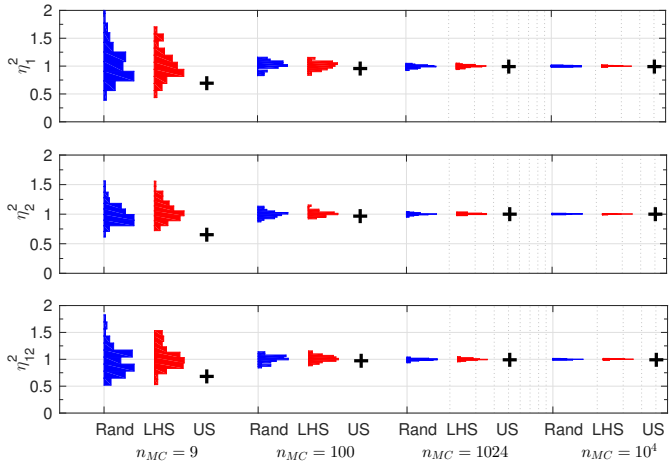


Figure 8: Local and global error estimates for several sampling strategies, with a 2nd order surrogate.

Consequently, histograms of the values of the efficiency are presented in Fig. 8 and Fig. 9. Such bars are obtained by making a large number of replications for different sampling choices (1000 replications for $n_{MC} = 9$ and $n_{MC} = 100$, 500 replications for $n_{MC} = 1024$, and 100 replications for $n_{MC} = 10^4$).

It can be seen first that the three sampling strategies converge to the exact value $\eta^2 = 1$ for the two cases $p = 2$ and $p = 4$. Even if consists in a reassuring result, we recall that the aim is to develop a cheap error estimate, using thus a small number of sampling points.

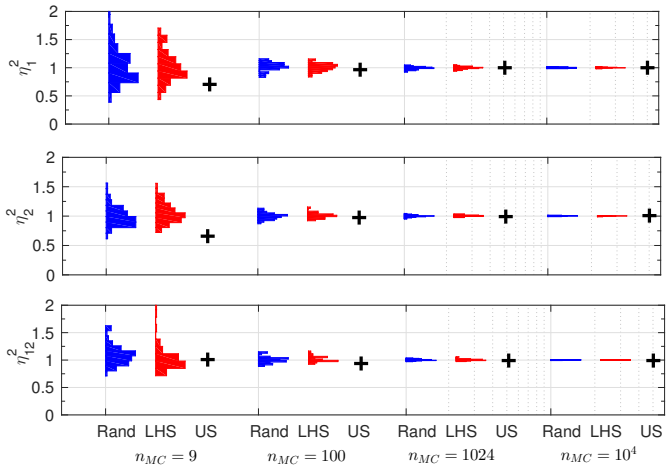


Figure 9: Local and global error estimates for several sampling strategies, with a 4th order surrogate.

For small values of n_{MC}, due to the small numbers of realizations of the resolution of the error equation Eq. (25), the error estimate ε^2_{MC} is subject to large variations. It can be observed that a LHS sampling decreases this variability and should be tried first instead of a pure random sampling. Moreover, Uniform Sampling outperforms the two other sampling strategies in our example: because the sampling strategy is deterministic, only one computation leads to a result, which belongs to the range of possible values obtained with a random sampling strategy. Even if this choice may not be the best sampling which could be achieved with a random sampling, the accuracy of the estimate is quite correct. However, this sampling suffers from the curse of dimensionality, so that this approach may not be applicable when a large number of stochastic parameters is considered. In this case, a LHS sampling is preferable.

Next, in order to investigate the behaviour of the method when a small number of samplings is used, we complete Fig. 8 and Fig. 9 by plotting in Fig. 10 to Fig. 13 the efficiency of the method using an uniform sampling (US) of 3 to 10 points for each stochastic parameter.

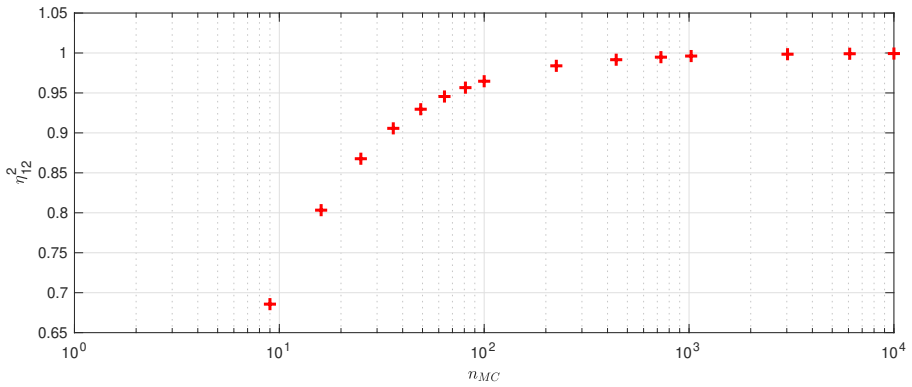


Figure 10: η^2_{12} as a function of n_{MC}, with US method and $p = 2$.

The convergence of η^2 exhibits a smooth behaviour. Classically, in the literature, an error estimated is considered correct if its efficiency is between 0.5 and 2. Here, even the smallest sampling satisfies this condition.

Finally, Fig. 14 shows that the local and global mean square errors, obtained with 16 and 81 samples (uniforming sampling of 4 or 9 points for each stochastic variable), are well predicted. A trading between precision and cost has to be found, as a larger number of points leads to a better superposition, with a higher numerical cost. Here, the use of 81 points is presented for illustration, but this choice is not necessary since the predicted error is already correct

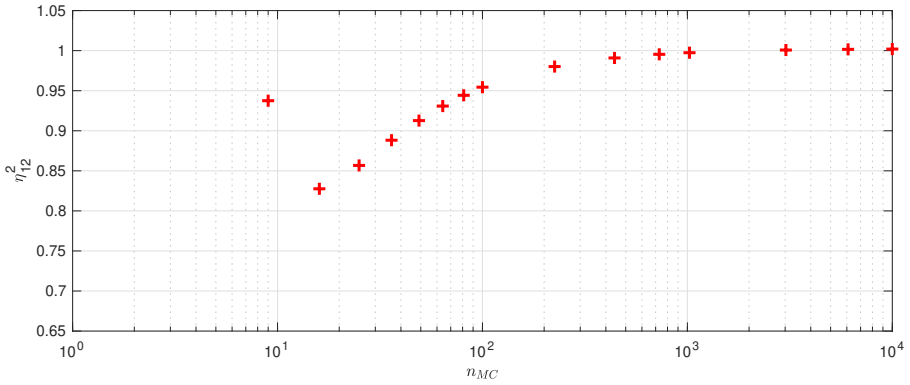


Figure 11: η^2_{12} as a function of n_{MC}, with US method and $p = 3$.

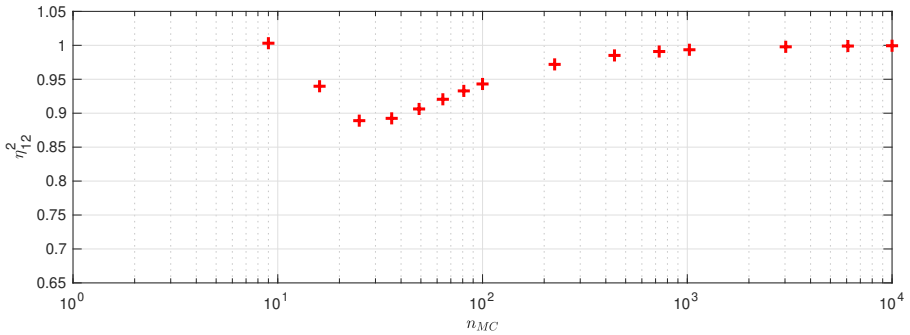


Figure 12: η^2_{12} as a function of n_{MC}, with US method and $p = 4$.

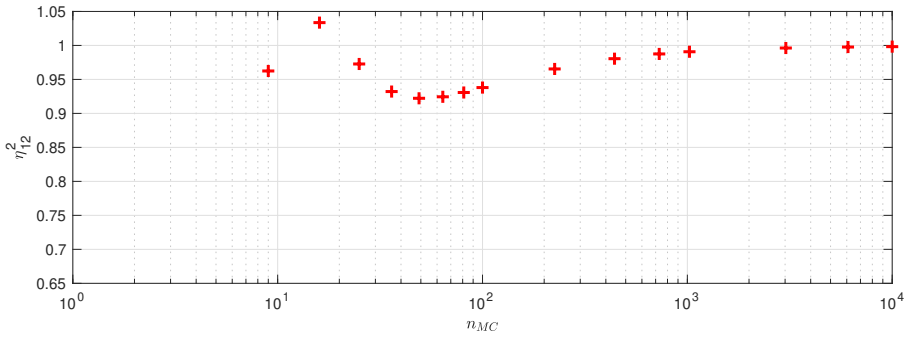
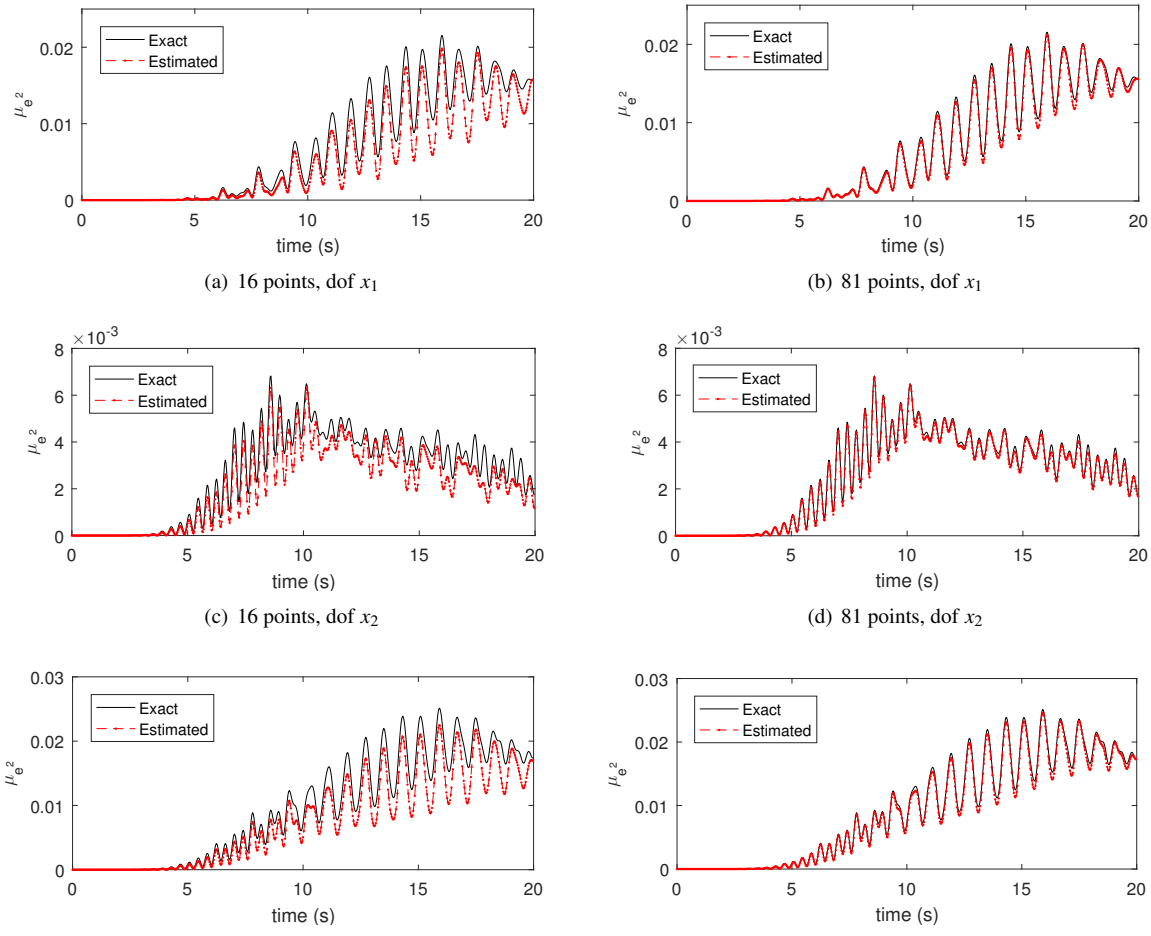


Figure 13: η^2_{12} as a function of n_{MC}, with US method and $p = 5$.

with only 16 samples. Furthermore, in comparison with classical error estimates such as the empirical error and LOO error , it can be observed on Fig. 6 that the method presented here, with only 16 additional computations, enables to predict much more accurately the estimation error. This result is true for local and global errors as well.



(a) 16 points, dof x_1

(b) 81 points, dof x_1

(c) 16 points, dof x_2

(d) 81 points, dof x_2

(e) 16 points, all dofs

(f) 81 points, all dofs

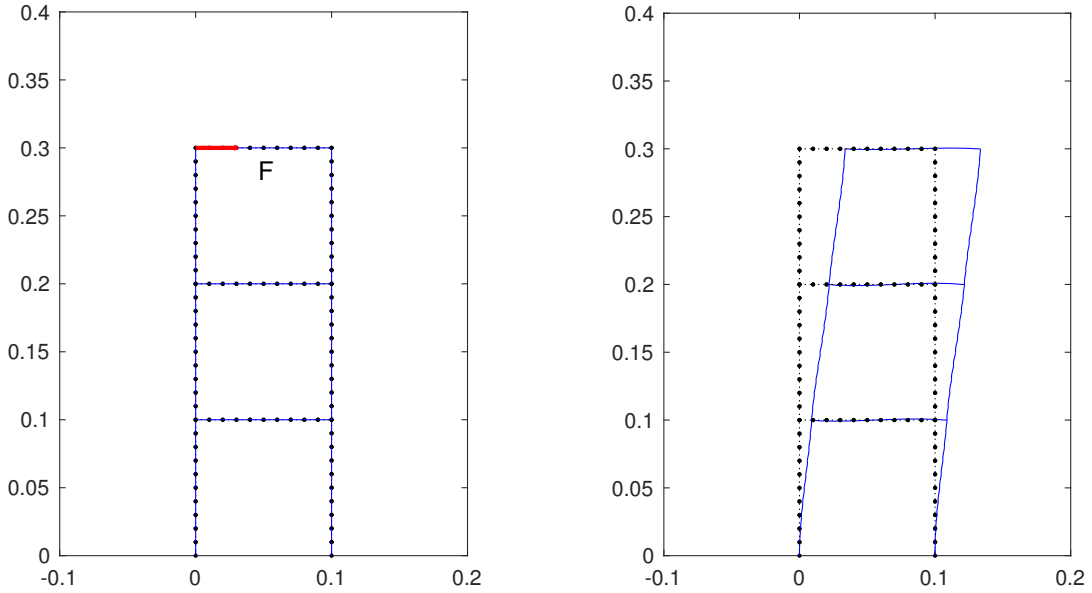
Figure 14: Exact and Estimated mean square errors for US sampling with a polynomial order of the surrogate $p = 2$.

4. Application to a larger benchmark problem

The next section proposes an illustration of the method to a large scale problem in this section. A truss structure is constituted of 9 different beams. Each beam has its own material properties, considered as constant on the length of the beam. Among their properties, two are considered as stochastic: the Young's modulus and the density. Filnally, it leads to 18 stochastic parameters. The truss is discretized using Euler-Bernoulli beams with 3 degrees of freedom per node (two translations and one rotation). The structure is subjected in time domain to a half-sine load on one node, starting at 10ms. Time integration is carried out using a classical Newmark scheme. We consider here only the error between the stochastic surrogate and the stochastic problem, so we do not take into account the error due to the time discretization and to the finite element discretization. An illustration of the truss structure is given in Fig. 15.

Parameter	Probability law	Interval
Young's modulus E	Uniform	$[180\ ;\ 220]\ GPa$
Density ρ	Uniform	$[7200\ ;\ 8400]\ kg/m^3$
Section S	Determinist	$6.10^{-6} m^2$
Section quadratic moment I	Determinist	$4.510^{-12} m^4$
Beam length L	Determinist	$0.1m$
max load F_0	Determinist	$1N$
Duration of load Δt	Determinist	$40ms$

Table 4: Values of input parameters of the truss structure.



(a) Mesh of the structure

(b) Scaled deformed structure

Figure 15: Truss structure

The material parameters are provided in Tab. 4. The finite element model results in 267 dofs. The stochastic space is a 18-dimension space. The quantity of interest is the horizontal displacement of the node loaded by the force. The metamodel is built with a second order PC expansion. It results in 190 stochastic coefficients to solve, and thus to 380 deterministic computations. The testing dataset consists in 20 samples computed with LHS.

To evaluate the error, considering that the exact stochastic response is not reachable with the Monte Carlo method due to the computational cost, we use the developed strategy in this paper. The exact error is not computable at an affordable cost so there is no way to know a priori about the minimal size of the DoE (Design of experiment). Consequently, the error is computed with the present approach for a growing number of LHS-samples, until convergence of the time-averaged predicted mean square error on the horizontal displacement. We chose to stop iterations when the relative tolerance between two steps is lower than 10%. The DoE is computed using a LHS design, the uniform sampling in a 18-dimension space being too expensive.

Fig. 16 presents the evolution of EE, LOO and present approach approximated errors for the horizontal displacement with time. It can be observed that the EE and LOO estimates do not predict the peak of error at proximity of the load, predicted with the method developed here.

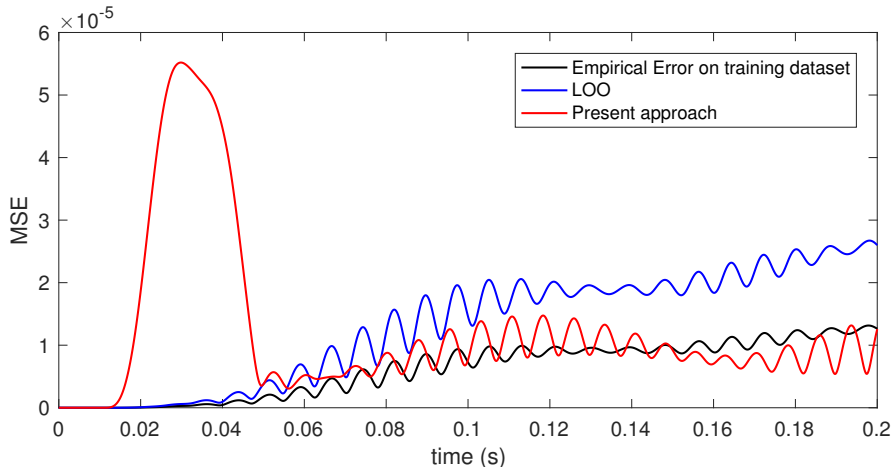


Figure 16: Error estimates applied on the truss structure, for a second order polynomial chaos surrogate.

5. Conclusion

The Polynomial Chaos Expansion is a very powerful tool to study the response of models subjected to parametric uncertainties. Several error values can be found in the literature, but it has been shown that even if their values provide a qualitative answer about the good fit between the exact random unknown and its PC approximation, they do not provide accurate quantitative values. When applied to structural dynamics, it has been demonstrated that the error is solution of an equation similar to the equation of the dynamics of the system, but where the source term is equal to the residue obtained by using the surrogate instead of the exact random variable in the equation of the dynamics. It has to be noted here that the presented approach should be also applicable to other kinds of metamodels. It has been illustrated numerically that solving the error equation with a Monte Carlo approach, even coarsely, provides a good information on the error level. We recommend the use of quasi Monte Carlo samplings to reduce the variance of the predictions. Active learning strategy based on sequential sampling could be a promising solution to be studied. In a forthcoming paper, a variant of this method will be studied on non linear problems. The idea is to solve the error problem using a dedicated Polynomial Chaos approximation.

References

[1] M. Tootkaboni, A. Asadpoure, J. K. Guest, Topology optimization of continuum structures under uncertainty – A Polynomial Chaos approach, Computer Methods in Applied Mechanics and Engineering 201-204 (2012) 263–275.

[2] C. Touzeau, B. Magnain, Q. Serra, E. Florentin, Accuracy and robustness analysis of a geometrical finite element model updating approach for material parameters identification in transient dynamic., International Journal of Computational Methods 16 (2019).

[3] D. Xiu, Fast numerical methods for stochastic computations: a review, Communications in computational physics 5 (2009) 242–272.

[4] G. Stefanou, The stochastic finite element method: Past, present and future, Computer Methods in Applied Mechanics and Engineering 198 (2009) 1031–1051.

[5] G. Fishman, Monte Carlo: concepts, algorithms, and applications, Springer Science & Business Media, 2013.

[6] R. E. Caflisch, Monte Carlo and quasi-Monte Carlo methods, Acta Numerica (1998) 1–49.

[7] N. Wiener, The Homogeneous Chaos, American Journal of Mathematics 60 (1938) 897.

[8] R. Askey, J. Wilson, Some basic hypergeometric orthogonal polynomials that generalize Jacobi polynomials, volume 319, American Mathematical Society, 1985.

[9] D. Xiu, G. E. Karniadakis, The Wiener Askey Polynomial Chaos for Stochastic Differential Equations, SIAM Journal on Scientific Computing 24 (2002) 619–644.

[10] M. Ghienne, C. Blanzé, L. Laurent, Stochastic model reduction for robust dynamical characterization of structures with random parameters, Comptes Rendus Mécanique 345 (2017) 844–867.

[11] D. Weiss, Z. Yosibash, Uncertainty quantification for a 1d thermo-hyperelastic coupled problem using polynomial chaos projection and p-fems, Computers & Mathematics with Applications 70 (2015) 1701 – 1720. High-Order Finite Element and Isogeometric Methods.

[12] S. Hosseini, Z. Asgari, Solution of stochastic nonlinear time fractional pdes using polynomial chaos expansion combined with an exponential integrator, Computers & Mathematics with Applications 73 (2017) 997 – 1007. Advances in Fractional Differential Equations (IV): Time-fractional PDEs.

[13] Z. H. Wang, C. Jiang, X. X. Ruan, Y. Q. Zhang, Z. L. Huang, C. S. Wang, T. Fang, Uncertainty propagation analysis of t/r modules, International Journal of Computational Methods 16 (2019) 1850105.
[14] T. Crestaux, O. Le Maitre, J.-M. Martinez, Polynomial chaos expansion for sensitivity analysis, Reliability Engineering & System Safety 94 (2009) 1161–1172.
[15] X. Wan, G. E. Karniadakis, Long-term behavior of polynomial chaos in stochastic flow simulations, Computer methods in applied mechanics and engineering 195 (2006) 5582–5596.
[16] M. Gerritsma, J.-B. van der Steen, P. Vos, G. Karniadakis, Time-dependent generalized polynomial chaos, Journal of Computational Physics 229 (2010) 8333–8363.
[17] H. C. Ozen, G. Bal, A dynamical polynomial chaos approach for long-time evolution of SPDEs, arXiv preprint arXiv:1605.04604 (2016).
[18] R. A. Todor, C. Schwab, Convergence rates for sparse chaos approximations of elliptic problems with stochastic coefficients, IMA Journal of Numerical Analysis 27 (2007) 232–261.
[19] X. Wan, G. E. Karniadakis, An adaptive multi-element generalized polynomial chaos method for stochastic differential equations, Journal of Computational Physics 209 (2005) 617–642.
[20] B. Chouvion, E. Sarrouy, Development of error criteria for adaptive multi-element polynomial chaos approaches, Mechanical Systems and Signal Processing 66-67 (2016) 201–222.
[21] D. H. Mac, Z. Tang, S. Clenet, E. Creusé, Residual-based a posteriori error estimation for stochastic magnetostatic problems, Journal of Computational and Applied Mathematics 289 (2015) 51–67.
[22] G. Blatman, B. Sudret, Adaptive sparse polynomial chaos expansion based on least angle regression, Journal of Computational Physics 230 (2011) 2345–2367.
[23] B. Sudret, Polynomial chaos expansions and stochastic finite element methods, Risk and reliability in geotechnical engineering (2014) 265–300.
[24] C. V. Mai, B. Sudret, Hierarchical adaptive polynomial chaos expansions, arXiv preprint arXiv:1506.00461 (2015).
[25] G. Kewlani, J. Crawford, K. Iagnemma, A polynomial chaos approach to the analysis of vehicle dynamics under uncertainty, Vehicle System Dynamics 50 (2012) 749–774.
[26] L. Nechak, S. Berger, E. Aubry, A polynomial chaos approach to the robust analysis of the dynamic behaviour of friction systems, European Journal of Mechanics-A/Solids 30 (2011) 594–607.
[27] G. Blatman, B. Sudret, An adaptive algorithm to build up sparse polynomial chaos expansions for stochastic finite element analysis, Probabilistic Engineering Mechanics 25 (2010) 183–197.
[28] G. Blatman, B. Sudret, Sparse polynomial chaos expansions and adaptive stochastic finite elements using a regression approach, Comptes Rendus Mécanique 336 (2008) 518–523.
[29] M. Berveiller, B. Sudret, M. Lemaire, Stochastic finite element: a non intrusive approach by regression, European Journal of Computational Mechanics/Revue Européenne de Mécanique Numérique 15 (2006) 81–92.
[30] K. Cheng, Z. Lu, Adaptive sparse polynomial chaos expansions for global sensitivity analysis based on support vector regression, Computers & Structures 194 (2018) 86 – 96.
[31] D. M. Allen, The prediction sum of squares as a criterion for selecting predictor variables, University of Kentucky, 1971.
[32] A. M. Molinaro, R. Simon, R. M. Pfeiffer, Prediction error estimation: a comparison of resampling methods, Bioinformatics 21 (2005) 3301–3307.
[33] G. Blatman, Adaptive sparse polynomial chaos expansions for uncertainty propagation and sensitivity analysis, Ph.D. thesis, Université Blaise Pascal-Clermont-Ferrand II, 2009.
[34] G. Blatman, B. Sudret, M. Berveiller, Quasi random numbers in stochastic finite element analysis, Mécanique & Industries 8 (2007) 289–297.
[35] M. D. McKay, R. J. Beckman, W. J. Conover, A comparison of three methods for selecting values of input variables in the analysis of output from a computer code, Technometrics 42 (2000) 55–61.
[36] K. Szepietowska, B. Magnain, I. Lubowiecka, E. Florentin, Sensitivity analysis based on non-intrusive regression-based polynomial chaos expansion for surgical mesh modelling, Structural and Multidisciplinary Optimization 57 (2018) 1391–1409.
[37] R. E. Caflisch, Monte carlo and quasi-monte carlo methods, Acta numerica 7 (1998) 1–49.