URSI GASS 2021, Rome, Italy, 28 August - 4 September 2021

Polynomial chaos within the frame of non-destructive testing

Charles Boulitrop*(1), Marc Lambert(1), and Sándor Bilicz(2)
(1) Université Paris-Saclay, CentraleSupélec, CNRS, Laboratoire de Génie Electrique et Electronique de Paris, 91192, Gif-sur-Yvette, France
Sorbonne Université, CNRS, Laboratoire de Génie Electrique et Electronique de Paris, 75252, Paris, France
(2) Department of Broadband Infocommunications and Electromagnetic Theory
Budapest University of Technology and Economics
3 Muegyetem rkp, H-1111 Budapest, Hungary

Abstract

Inverse scattering problems can be seen as an optimization problem in which a cost function has to be minimized. The cost function measures the discrepancy between a physical model written as a function of the sought parameters and the measured data. Due to the high computational cost of physical models, metamodels (also known as surrogate models) have gradually appeared as alternatives. Polynomial chaos (PC) expansions are a type of metamodel based on an explicit spectral decomposition of the physical model, allowing for an exact computation of the gradient of the metamodel. The gradient of the metamodel is then used to improve the performance of the global algorithm Particle Swarm Optimization (PSO).

1 Introduction

Inverse scattering problems have many applications, ranging from medical imaging and groundwater flow modeling to nondestructive testing. The objective is characterizing a region of interest (most often finding its geometrical and physical parameters) from measured data (variation of impedance, electromagnetic or acoustic scattered field, ...) [1], [2]. Inverse scattering problems can be recast as an optimization problem in which a cost function has to be minimized. The cost function measures the discrepancy between a physical model written as a function of the sought parameters and the measured data. Such a minimization can be done using iterative methods which have been proved to be efficient and which aim at progressively refine the estimation of the parameters thanks to a large number of physical model evaluations. However, physical models are often computationally expensive to evaluate, thus making inversion a tedious process. To avoid the computational cost of an exact model, surrogate models - or metamodels - can be used. A metamodel is an approximation of the explicit model, which aims to provide a much faster model evaluation with a small loss in precision due to some approximations [3]–[5].

In this work, the potential complementarity between stochastic global optimization and deterministic local optimization is investigated, provided the gradient of the metamodel is computable. An application of polynomial chaos as metamodel [6] in the scope of eddy-current testing (ECT) application will be presented. First results combining PC expansion as metamodel and particle swarm optimization as optimization technique will be presented. The work is organized as follows: the theoretical framework and the mathematical tools of the PCE are presented in Section 2, inversion algorithms are shortly described in Section 3 and configurations and inversion results are presented in Section 4.

2 Polynomial chaos expansion as metamodel

Polynomial chaos (PC) is a widely used technique which, most often, finds its utility in uncertainty quantification (UQ) problems [7], but can also be used as metamodel [6]. PC expansions are often used to model the random output of a physical model whose inputs are random variables. But they can also be used as an explicit interpolation technique. PC expansions rely on a spectral decomposition on a basis of orthogonal polynomial functions. Let $\mathbf{X} \in \mathbb{R}^M$ be a real-valued vector, input of the physical model $\mathcal{F}$, whose output is noted $\mathbf{Y} \in \mathbb{C}^N$. Formally this gives

$$\mathbf{Y} = \mathcal{F}(\mathbf{X}) \tag{1}$$

The PC expansion of the model $\mathcal{F}$ is then

$$\mathcal{F}(\mathbf{X}) = \sum_{\alpha \in \mathbb{N}^M} c_\alpha \Psi_\alpha(\mathbf{X}) \tag{2}$$

where Ψ_α are the pre-determined multivariate polynomials and c_α the complex-valued coefficients, the latter being computed using the UQLab toolbox [8]. Four methods are available in UQLab: projection, least-squares regression, least-angle regression (LAR) and orthogonal matching pursuit (OMP) among which the chosen method for this work is the least-squares regression.

However the expansion at (2) is infinite and needs to be truncated to be numerically tangible. Three truncation strategies are available in UQLab: a basic truncation on the degree of the polynomials, a maximum interaction scheme and a hyperbolic truncation, the latter being the chosen strategy for this work. The PC expansion then becomes

$$\mathcal{F}(\mathbf{X}) \simeq \hat{\mathcal{F}}(\mathbf{X}) = \sum_{\alpha \in \mathcal{A}^{(n,p)}} c_\alpha \Psi_\alpha(\mathbf{X}) \tag{3}$$

$$\text{where } \mathcal{A}^{(n,p)} = \left\{ \alpha = (\alpha_1 \dots \alpha_M) \in \mathbb{N}^M, \left(\sum_{i=1}^{M} \alpha_i^p \right)^{\frac{1}{p}} \leqslant n \right\} \tag{4}$$

The two parameters governing the truncation (degree n and norm p) are to be set judiciously since they impact both the precision of the metamodel and its computational cost.

The polynomial basis (the Ψ_α polynomials) does not need to be computed. The multivariate polynomials Ψ_α evaluated at $\mathbf{X} = (x_1, \dots, x_M)$ are defined as the product of univariate ones:

$$\Psi_\alpha(\mathbf{X}) = \prod_{i=1}^{M} \psi_{\alpha_i}(x_i) \tag{5}$$

The univariate polynomials are solely determined by the orthogonality of the basis with regard to the distribution of the input vector, according to equation (6).

$$\langle \psi_i, \psi_j \rangle = \int \psi_i(x)\psi_j(x)\rho(x)\mathrm{d}x = \delta_{ij} \tag{6}$$

where ρ is the distribution of input x and δ_{ij} the Kronecker symbol.

Even though the PC expansion formulation may not seem intuitive, it has a significant advantage since it is analytically explicit. This means that computing the gradient of a PC expansion is straightforward [9]. The following proof is given with the Legendre polynomial basis but can be extended to other bases thanks to the recurrence relations presented in [10, Chapter 22]:

$$(n+1)P_{n+1}(x) = (2n+1)xP_n(x) - nP_{n-1}(x) \tag{7}$$

$$(1-x^2)\frac{\mathrm{d}}{\mathrm{d}x}P_n(x) = -nxP_n(x) + nP_{n-1}(x) \tag{8}$$

By combining equations (7) and (8), one can obtain the following recurrence relation

$$\frac{\mathrm{d}}{\mathrm{d}x}P_{n+1}(x) = (n+1)P_n(x) + x\frac{\mathrm{d}}{\mathrm{d}x}P_n(x) \tag{9}$$

Equation (9) is then used to compute the derivative of the univariate polynomials. Then a tensor product is applied to compute the first-order derivative of the PC polynomials w.r.t. each input parameter.

$$\frac{\partial}{\partial x_i}\Psi_\alpha(\mathbf{X}) = \frac{\mathrm{d}}{\mathrm{d}x_i}\psi_{\alpha_i}(x_i) \prod_{\substack{k=1 \\ k \neq i}}^{M} \psi_{\alpha_k}(x_k) \tag{10}$$

The first-order derivative of the PC expansion linearly follows:

$$\frac{\partial}{\partial x_i}\hat{\mathcal{F}}(\mathbf{X}) = \sum_{\alpha \in \mathcal{A}^{(n,p)}} c_\alpha \frac{\partial}{\partial x_i}\Psi_\alpha(\mathbf{X}) \tag{11}$$

where α is defined in equation (4). The gradient of the PC expansion is the vector containing the first-order derivatives of the PC expansion.

$$\nabla\hat{\mathcal{F}}(\mathbf{X}) = \left(\frac{\partial}{\partial x_1}\hat{\mathcal{F}}(\mathbf{X}), \dots, \frac{\partial}{\partial x_M}\hat{\mathcal{F}}(\mathbf{X}) \right) \tag{12}$$

3 Optimization: PSO & n-gPSO

First introduced in [11], then thoroughly discussed in [12], Particle Swarm Optimization (PSO) is an iterative optimization algorithm, inspired by the behavior of swarms of bees, among other things. The idea is to let a swarm of particles move in the search space in directions depending on their best personal position encountered $\mathbf{P}$, and the best position a portion of the swarm has encountered $\mathbf{S}$. The displacement of a particle $\mathbf{X}$ at iteration k is then:

$$\mathbf{v}_{k+1} = c_0\mathbf{v}_k + c_s\mathrm{rand}(0,1)(\mathbf{P}_k - \mathbf{X}_k) + c_s\mathrm{rand}(0,1)(\mathbf{S}_k - \mathbf{X}_k) \tag{13}$$

where c_0 is the particle confidence in its own displacement, c_s is the particle confidence in the swarm.

Two optimization strategies are proposed in this work: the first one is solely based on the PSO algorithm, the second one uses the aforementioned gradient information. Both strategies use the same cost function:

$$\mathcal{E}(\mathbf{X}) = \frac{\left\| \hat{\mathcal{F}}(\mathbf{X}) - Y_{\mathrm{data}} \right\|^2}{\left\| Y_{\mathrm{data}} \right\|^2} \tag{14}$$

where Y_{data} is the output data to be inversed and $\|.\|$ the L2-norm over $\mathbb{C}^N$.

The second strategy, referred to as gPSO, also uses the gradient $\nabla\mathcal{E}(\mathbf{X})$ of the cost function to provide a deterministic way to displace the swarm inside the search space. The first-order derivative of the cost function is simply computable thanks to equations (11) and (14) and to the scalar product $\langle .,. \rangle$ being a hermitian form for complex-valued vectors. Once the gradient of the cost function is obtained, a steepest descent gradient displacement is imposed to the particles for a determined number of iterations. The algorithm then moves the swarm of particles following either the PSO-based displacement or the gradient-based displacement. An appropriate stopping criterion has to be set, usually related to a threshold on the cost function, a maximum number of iterations or a number of iterations passing without improvement of the value of the cost function.

4 Numerical results

Configuration The algorithm has been applied to a NdT configuration. A plate structure affected by a single rectangular crack is inspected by a coil. The problem has 3 real-valued parameters: crack depth, crack length, crack width, which form the input domain. The output domain is a 2D-mapping - referred to as a C-scan - of the impedance variation in the coil over the plate, which is a complex-valued rectangular image. The configuration is shown in Figure 1. The database parameters shown in Table 1e are sampled 10 times each uniformly between the minimal and maximal values, which forms 1×10^3 uniform samples of the input domain. Since the sampling distribution is uniform, the polynomial basis associated is the Legendre polynomial basis. The degree n and norm p of the truncation, introduced in equation (4), have been set respectively to 5 and 0.8 after a parametric study on their influence.

The impedance variation is measured on a grid of 57 points along x and 41 points along y leading to a dimension of 2337 of the output domain. The latter is relatively high and thus can be reduced to improve computation times. To achieve this, a principal component analysis [13] is used. The purpose of PCA is to reduce the dimension of a data set, while keeping as much information as possible. The original set is transformed into a new set of principal components, which are ordered by the amount of information that they represent. The method used to build the principal component basis can be found in [14]. The main difference here is the way the principal components are truncated. An *a posteriori* criterion is used:the normalized root mean square error between the original data set and the data set reconstructed from the truncation is computed for increasing numbers of principal components until this error is less than a given threshold (1×10^{-2} in our case). For this configuration, the output domain dimension is reduced by a factor of $1/40$.

Inversion results on noisy data To assess the robustness of the algorithm to noise, the testing procedure has been carried out on noisy data. The noise has been added to the modulus and to the argument of each C-scan. The modulus noise is a fraction of the maximal modulus found on the C-scan while the argument noise is in degrees.

The inversions have been carried out on 100 different assumed cracks, with the input parameters being sampled by Latin Hypercube Sampling in the input domain. The C-scan to be inverted is generated by means of a full metamodel, *i.e.*, whose dimension was not reduced by PCA. Then the noise is added to the generated C-scan, before being transformed into the principal component domain, to be fed to the optimization loop. The algorithm stops when the cost function value reaches 1×10^{-3}, when 150 iterations have passed or when 15 iterations have passed with no improvement of the value of the cost function.

Figures 2a and 2b show the results of the algorithm when applied on noisy data for different noise levels, respectively 5% on modulus / 5° on argument and 10% on modulus / 10° on argument. The algorithm performs relatively well on noisy data. It does converge but towards a value that is not the sought one.

5 Conclusion and future work

An original inversion algorithm based on polynomial chaos and PSO has been introduced and tested on an eddy-current testing configuration. The combination of local optimization from the gradient descent algorithm and global optimization from PSO improves the convergence speed in terms of iterations but adds some computation time. The algorithm converges fairly well even when confronted to noisy data, performing on average 25% better in terms of absolute error on the reconstructed parameters, especially on the crack width. Reconstructions from gPSO also show 40% less variability than from PSO. The convergence speed can be furtherly improved by optimizing the gradient step size computation. To achieve this, several alternative step size selection strategies are considered. A study on the influence of the number of gradient displacements per algorithm iteration is to be done in that regard. The algorithm is also expected to be tested on other configurations, with an increasing number of parameters as, for instance, in [15].

References

[1] X. Chen, *Computational Methods for Electromagnetic Inverse Scattering*. John Wiley & Sons, Ltd, 2018.

[2] M. Pastorino and A. Randazzo, *Microwave Imaging Methods and Applications*. Artech House, 2018.

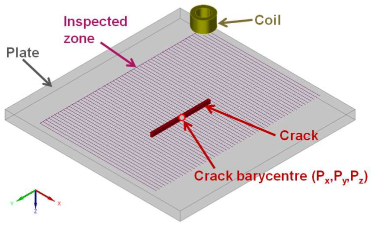
[3] J. P. C. Kleijnen and W. C. M. V. Beers, "Application-driven sequential designs for simulation experiments: Kriging metamodeling", *The Journal of the Operational Research Society*, vol. 15, no. 8, pp. 303–325, 2004.

[4] S. Bilicz, "Sparse grid surrogate models for electromagnetic problems with many parameters", *IEEE Transactions on Magnetics*, vol. 52, no. 3, pp. 1–4, 2016.

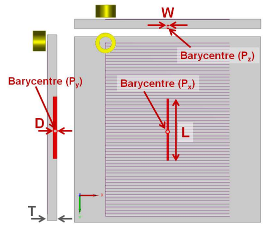
[5] C. Cai, R. Miorelli, M. Lambert, T. Rodet, D. Lesselier, and P.-E. Lhuillier, "Metamodel-based markov-chain-monte-carlo parameter inversion applied in eddy current flaw characterization", *NDT& E International*, vol. 99, pp. 13–22, 2018.

[6] P. Kersaudy, B. Sudret, N. Varsier, O. Picon, and J. Wiart, "A new surrogate modeling technique combining kriging and polynomial chaos expansions – application to uncertainty analysis in computational dosimetry", *Journal of Computational Physics*, vol. 286, pp. 103–117, 2015.

[7] T. Crestaux, O. L. Maître, and J.-M. Martinez, "Polynomial chaos expansion for sensitivity analysis", *Reliability Engineering & System Safety*, vol. 94, no. 7, pp. 1161–1172, 2009, Special Issue on Sensitivity Analysis.



(a)



(b)

Inner radius r_1	1.00 mm
Outer radius r_2	1.75 mm
Height l	2.0 mm
Number of turn N	328
Lift-off l_0	0.303 mm
Frequency f	100 kHz

(c) Coil parameters

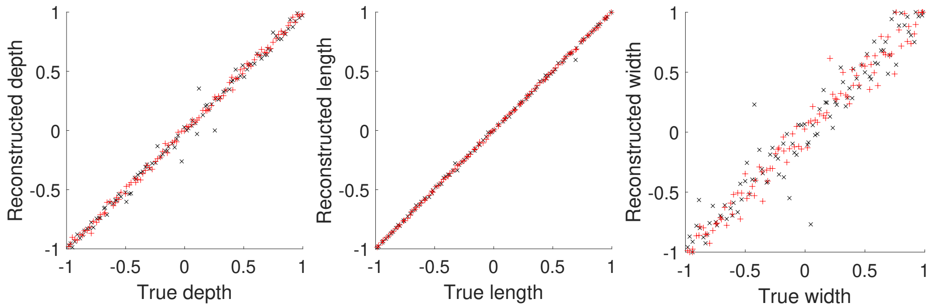
Thickness T	1.55 mm
Conductivity σ	1.02 MS m^{-1}
Height l	2.0 mm

(d) Plate parameters

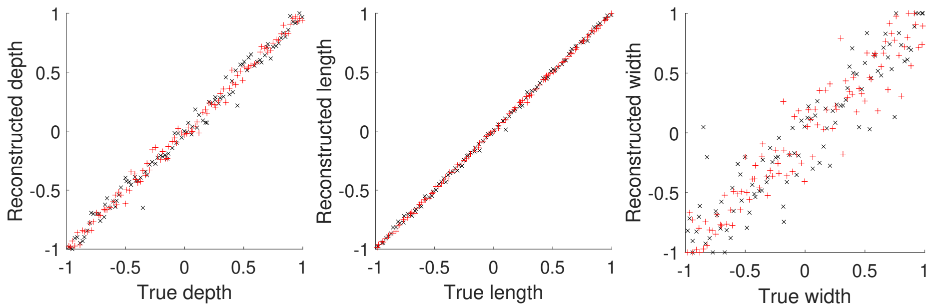
Parameter	Min (mm)	Max (mm)	Samples
Depth D	0.31	1.24	10
Length L	5	15	10
Width W	1×10^{-3}	0.45	10

(e) Database parameters

Figure 1. (a) 3D view of the configuration and (b) projected 2D views



(a) Noise level of 5% and 5°



(b) Noise level of 10% and 10°

Figure 2. Input parameter reconstruction by PSO (in black) and gPSO (in red) on 100 points sampled by LHS. for a

[8] S. Marelli and B. Sudret, "UQLab: A framework for uncertainty quantification in Matlab", *Vulnerability, Uncertainty, and Risk*, pp. 2554–2563.

[9] B. Sudret and C. Mai, "Computing derivative-based global sensitivity measures using polynomial chaos expansions", *Reliability Engineering & System Safety*, vol. 134, pp. 241–250, 2015.

[10] M. Abramowitz and I. Stegun, *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*, ser. Applied Mathematics Series. U.S. Government Printing Office, 1948.

[11] J. Kennedy and R. Eberhart, "Particle swarm optimization", *Proceedings of ICNN'95 - International Conference on Neural Networks*, vol. 4, 1995, 1942–1948 vol.4.

[12] M. Clerc, *Particle Swarm Optimization*. John Wiley & Sons, 2010, vol. 93.

[13] I. T. Jolliffe, *Principal Component Analysis*, 2nd ed. Springer, 2002.

[14] G. Blatman and B. Sudret, "Principal component analysis and least angle regression in spectral stochastic finite element analysis", *Proc. 11th Int. Conf. on Applications of Stat. and Prob. in Civil Engineering (ICASP11), Zurich, Switzerland*, 2011.

[15] S. Bilicz, "Sensitivity analysis of inverse problems in em non-destructive testing", *IET Science, Measurement & Technology*, vol. 14, 543–551(8), 5 2020.