Introduction to metamodels & Polynomial chaos expansions

May 2020

Outline

- Metamodels
  - Definition, construction and validation
  - Use for sensitivity analysis
- Polynomial chaos expansion
- Applications in non-destructive testing
- Conclusions

Outline

Metamodels

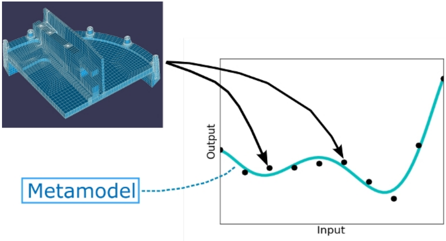
- Definition, construction and validation
- Use for sensitivity analysis

Polynomial chaos expansion

Applications in non-destructive testing

Conclusions

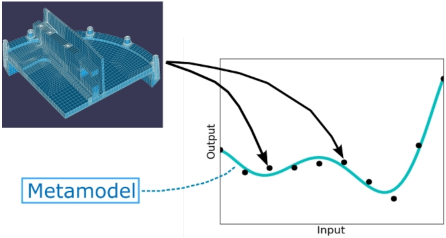
Metamodel - Definition



Meta : a prefix added to the name of something that consciously references or comments upon its own subject or features, e.g. metamodel : a model of another model

A metamodel is an approximation model that mimics the behavior of a computationally expensive simulator by training on *observations (data)* of the latter.

Metamodel - Definition



Expensive simulator: $\boldsymbol{Y} = f(\boldsymbol{X})$

- $\boldsymbol{X}, \; \boldsymbol{Y}$ are vectors of input and output,
- $\boldsymbol{X} = (X_1, \ldots, X_M)$

Metamodel: $\widetilde{\boldsymbol{Y}} = \hat{f}(\boldsymbol{X}, \boldsymbol{\theta})$, $\boldsymbol{\theta}$ being its vector of parameters

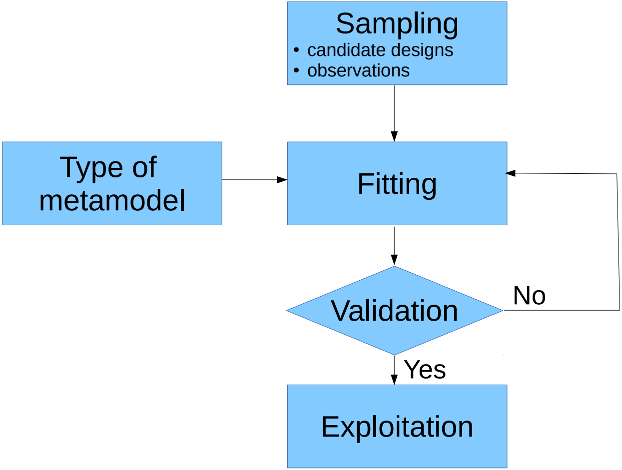
By definition, two requirements of a metamodel $\hat{f}$ are:

- Usefully accurate when predicting away from known observation
- Being significantly cheaper to evaluate than the primary simulator

Metamodel - Objectives

- Show functional relationships between input parameters and the output quantity of interest: impacts of variables
- Augment results from single, expensive simulations: results can be predicted without use of the primary simulator; a continuous predictive function instead of discrete observations
- Optimize the output quantity of interest: determine configurations that maximize the response or achieve specifications or customer requirements
- Replace the primary simulator in uncertainty propagation (surrogate model): sensitivity analysis

Major steps for constructing a metamodel



Major steps for constructing a metamodel

Sampling (define an experimental design):

- A number of possible candidate designs are generated
- The designs are launched with the primary simulator

Constructing the metamodel:

- A type of metamodel is selected (among several available options)
- The metamodel is fitted to the available data
- The metamodel is validated (yes or no)
  - if yes: stop
  - if no, several solutions to be considered
    - change method for fitting: use advanced regression technique instead of least squares errors,
    - change metamodel parameters: increase polynomial degrees,
    - enrich the experimental design where the model is inaccurate or interesting behavior is observed,
    - change the type of metamodel: polynomial chaos instead of second-order response surface

Some types of metamodels

Polynomial models: (response surface)

Second-order model

$$\tilde{Y} = \theta_0 + \sum_{i=0}^{M} \theta_i X_i + \sum_{i=0}^{M} \theta_{ii} X_i^2 + \sum_{i<j} \sum_{j=2}^{M} \theta_{ij} X_i X_j$$

Polynomial chaos expansion:

$$\tilde{\boldsymbol{Y}} = \sum_{0 \leq |\boldsymbol{k}| \leq p} \theta_{\boldsymbol{k}} \psi_{\boldsymbol{k}}(\boldsymbol{X})$$

where $\psi_{\boldsymbol{k}}()$ being polynomial chaos functions

Radial basis function:

$$\tilde{\boldsymbol{Y}} = \sum_{k=1}^{N} \theta_k \psi(\|\boldsymbol{X} - \boldsymbol{X}_k\|)$$

where $\psi()$ being a radial basis function with its centers taken at $\boldsymbol{X}_k$, $k = 1, \ldots N$ in the experimental design

Some types of metamodels

Kriging: (Gaussian process regression)

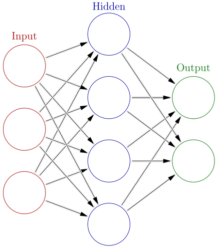
Deterministic trend: linear regression on a fixed basis

$$m(\boldsymbol{X}) = \boldsymbol{r}(\boldsymbol{X})^T \boldsymbol{\theta}$$

Random fluctuation: zero-mean stationary Gaussian process of covariance function

$$k(\boldsymbol{X}, \boldsymbol{X}') = \sigma^2 \rho(\|\boldsymbol{X} - \boldsymbol{X}'\|)$$

Artificial neural network: Radial basis function is a single layer neural network with radial coordinate neurons



Methods for fitting metamodels

Least square regression: minimize sum of squared errors of a linear regression model

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{N} \epsilon_i^2 = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{N} (Y_i - \hat{f}(\boldsymbol{X}_i, \boldsymbol{\theta}))^2$$

Regularized regression methods: minimize sum of squared errors under a constraint

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{N} (Y_i - \hat{f}(\boldsymbol{X}_i, \boldsymbol{\theta}))^2 + \lambda\, R(\boldsymbol{\theta})$$

- $R(\boldsymbol{\theta}) = ||\boldsymbol{\theta}||_2$: Ridge regression,
- $R(\boldsymbol{\theta}) = ||\boldsymbol{\theta}||_1$: LASSO regression,
- λ: non-negative regularization coefficient

Methods for fitting metamodels

Maximum likelihood estimation: e.g. assume that the errors ϵ are independently randomly distributed according to a normal distribution with standard deviation σ

$$\mathcal{L} = \frac{1}{(2\pi\sigma^2)^{N/2}} \prod_{i=1}^{N} exp\left(-\frac{1}{2}\left(\frac{Y_i - \hat{f}(\boldsymbol{X}_i, \boldsymbol{\theta})}{\sigma}\right)^2\right)$$

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \mathcal{L}$$

***K*-fold cross-validation method:**

$\mathcal{K} : \{1, \ldots, N\} \mapsto \{1, \ldots, K\}$ partition of N observations to K roughly equal-sized parts, $K = N$: leave-one-out

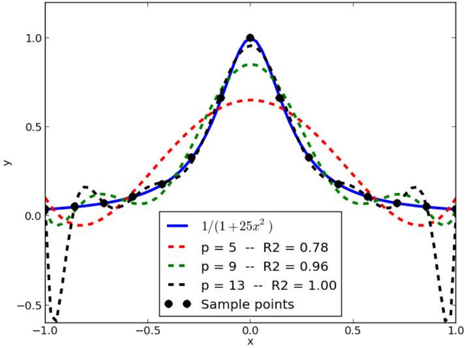
$\hat{f}^{-k}()$: fitted metamodel with k-th part of data set aside

Cross-validation estimate of prediction error:

$$CV(\hat{f}, \boldsymbol{\theta}) = \frac{1}{N} \sum_{i=1}^{N} L(Y^i, \hat{f}^{-\mathcal{K}(i)}(X_i))$$

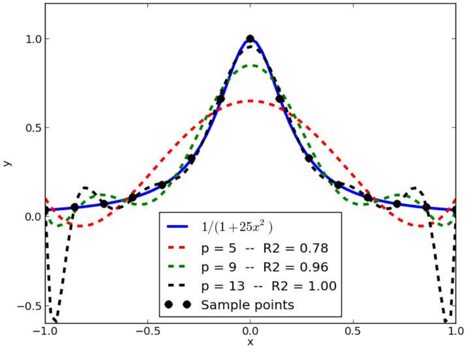
$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} CV(\hat{f}, \boldsymbol{\theta})$$

Overfitting



A metamodel that fits closely or exactly a specific set of data (training set) but fails to *predict* future data reliably

Validation of metamodels

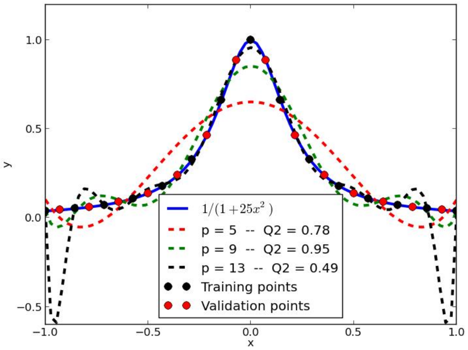


Coefficient of determination R^2

$$R^2 = 1 - Err \quad , \quad Err \propto \sum_{i=1}^{N} \left(f(\xi^{(i)}) - \tilde{f}_{\boldsymbol{a}}(\xi^{(i)}) \right)^2$$

R^2 does not detect over-fitting and overestimates *predictive* performance

Validation of metamodels



Equivalent of R^2 on an independent validation set:

$$Q^2 = 1 - Err \quad , \quad Err \propto \sum_{i=1}^{N_{val}} \left(f(\xi^{(i)}) - \hat{f}_{\mathbf{a}}(\xi^{(i)}) \right)^2$$

Validation on an independent validation set is necessary

Validation of metamodels

Cross-validation consists in dividing the data sample into two sub-samples.

- A metamodel is built with the first sub-sample (training set)
- Its performance is assessed by comparing its predictions with the second sub-sample (test set)

Data are often scarce. Partition in training and validation set is a luxury.

K-fold cross-validation: the data sample is divided into K sub-samples of roughly equal size.

$K = N$: leave-one-out error

Variance-based sensitivity analysis

Consider the model $Y = f(\boldsymbol{X})$ with random inputs $\boldsymbol{X} = \{X_1, \ldots, X_M\}$

- The output dispersion is characterized by its variance $\mathbb{V}\text{ar}\,[Y]$
- Partial variance due to X_i:

$$\mathbb{V}\text{ar}_{X_i}\left[\mathbb{E}_{\boldsymbol{X}\sim X_i}\,[Y|X_i]\right])$$

  $\boldsymbol{X} \sim X_i$: set of all variables except X_i

- Sobol index, interpretable as a variance percentage:

$$S_i = \frac{\mathbb{V}\text{ar}_{X_i}\left[\mathbb{E}_{\boldsymbol{X}\sim X_i}\,[Y|X_i]\right])}{\mathbb{V}\text{ar}\,[Y]}$$

- Sobol indices to interactions can also be defined:

$$S_{ij} = \frac{\mathbb{V}\text{ar}_{X_{ij}}\left[\mathbb{E}_{\boldsymbol{X}\sim X_{ij}}\left[Y|X_{ij}\right]\right])}{\mathbb{V}\text{ar}\,[Y]}$$

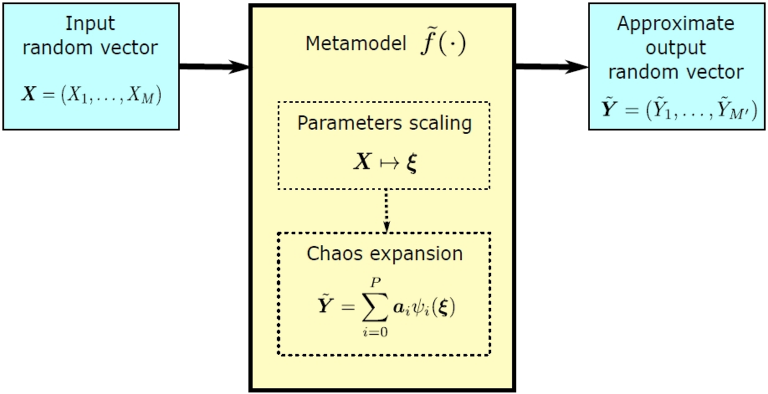
PROBLEM: Estimating the partial variances may require many (costly) model evaluations

Solution: Use an analytic approximation of the model – **Metamodel**

Outline

Polynomial chaos expansion

Let us consider the model : $\boldsymbol{Y} = f(\boldsymbol{X})$, $\boldsymbol{X}, \boldsymbol{Y}$ are random vectors



Decomposition of Y onto an orthonormal polynomial basis

Polynomial chaos basis

Assumption: Independent input random variables $X_1, \ldots, X_M$

Componentwise transform: $\xi_i = \mathcal{T}_i(X_i)$ (often based on CDFs, i.e. $\mathcal{T}_i(\cdot) \equiv \mathcal{F}_{\xi_i}^{-1}(\mathcal{F}_{X_i}(\cdot))$)

— Several possible choices for each $(\xi_i, \mathcal{T}_i)$

— A given ξ_i dictates the choice of a family $(\pi_k^{(i)})_{k \geq 0}$ of orthonormal polynomials

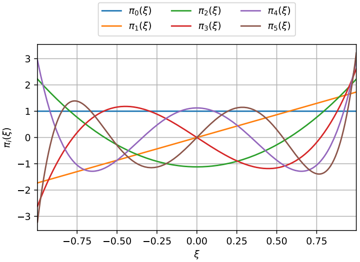
Given a uniform random variable $X \sim \mathcal{U}([0, 10])$, the transform $\xi = X/5 - 1$ leads to $\xi \sim \mathcal{U}([-1, 1])$ for which Legendre polynomials are used:

$$\pi_0(\xi) = 1 \ , \ \pi_1(\xi) = \sqrt{3}\xi \ , \ \pi_2(\xi) = \frac{\sqrt{5}}{2}(3\xi^2 - 1) \ , \ \ldots$$

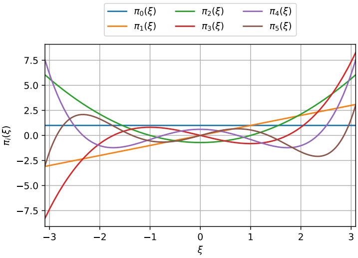
Given a normal random variable $X \sim \mathcal{N}(5, 1)$, the transform $\xi = X - 5$ leads to a standard normal RV $\xi \sim \mathcal{N}(0, 1)$ and Hermite polynomials:

$$\pi_0(\xi) = 1 \ , \ \pi_1(\xi) = \xi \ , \ \pi_2(\xi) = \frac{\sqrt{2}}{2}\left(\xi^2 - 1\right) \ , \ \ldots$$

Polynomial chaos basis



Legendre polynomials



Hermite polynomials

Properties: $\pi_0^{(i)} = 1\,, \quad \mathbb{E}\left[\pi_k^{(i)}(\xi_i)\right] \equiv \int_{\mathcal{D}_\xi} \pi_k^{(i)}(u)\, f_{\xi_i}(u)\, du = 0 \quad \forall k \geq 1$

$$\mathbb{E}\left[\pi_k^{(i)}(\xi_i)\, \pi_l^{(i)}(\xi_i)\right] \equiv \int_{\mathcal{D}_\xi} \pi_k^{(i)}(u)\, \pi_l^{(i)}(u)\, f_{\xi_i}(u)\, du = 1 \quad \text{if} \quad k = l \quad \text{else} \quad 0$$

Relevant for analytical estimation of first-order moments and Sobol' sensitivity indices

Polynomial chaos basis

Multivariate orthonormal polynomials:

$$\psi_{\boldsymbol{k}}(\boldsymbol{\xi}) = \pi^{(1)}_{k_1}(\xi_1) \times \cdots \times \pi^{(M)}_{k_M}(\xi_M)$$

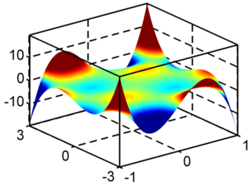
$\boldsymbol{\xi} = (\xi_1, \cdots, \xi_M)$: input vector; $\boldsymbol{k} = (k_1, \cdots, k_M)$: indices vector

Bivariate Legendre-Hermite polynomials:

$$\psi_{0,0}(\xi_1, \xi_2) = \pi^{(1)}_0(\xi_1) \times \pi^{(2)}_0(\xi_2) = 1$$

$$\psi_{1,0}(\xi_1, \xi_2) = \pi^{(1)}_1(\xi_1) \times \pi^{(2)}_0(\xi_2) = \sqrt{3}\xi_1$$

$$\psi_{1,2}(\xi_1, \xi_2) = \pi^{(1)}_1(\xi_1) \times \pi^{(2)}_2(\xi_2) = \frac{\sqrt{6}}{2}\xi_1(\xi_2^2 - 1)$$

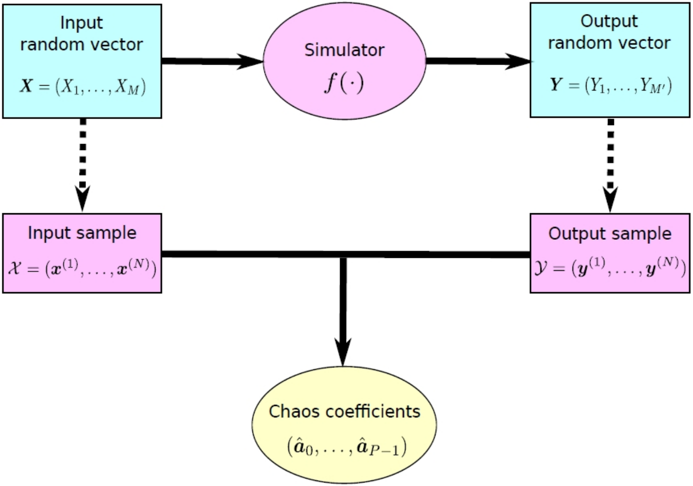


$\psi_{3,3}(\xi_1, \xi_2)$

Polynomial chaos expansion:

$$\tilde{\boldsymbol{Y}} = \sum_{0 \le |\boldsymbol{k}| \le p} a_{\boldsymbol{k}} \psi_{\boldsymbol{k}}(\boldsymbol{\xi}) = \sum_{0 \le |\boldsymbol{k}| \le p} a_{\boldsymbol{k}}\, \pi^{(1)}_{k_1}(\xi_1) \times \cdots \times \pi^{(M)}_{k_M}(\xi_M)$$

Estimation of polynomial chaos coefficients



Caution: the input sample $\mathcal{X}$ must respect the PDF of $\boldsymbol{X}$

Least squares

Model evaluations

$$\mathcal{Y} = \begin{pmatrix} f \circ \mathcal{T}^{-1}(\boldsymbol{\xi}^{(1)}) \\ \vdots \\ f \circ \mathcal{T}^{-1}(\boldsymbol{\xi}^{(N)}) \end{pmatrix}$$

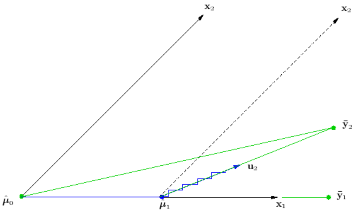
Chaos basis evaluations

$$\boldsymbol{\Psi} = \begin{pmatrix} \psi_{\boldsymbol{\alpha}_0}(\boldsymbol{\xi}^{(1)}) & \cdots & \psi_{\boldsymbol{\alpha}_{P-1}}(\boldsymbol{\xi}^{(1)}) \\ \vdots & \ddots & \vdots \\ \psi_{\boldsymbol{\alpha}_0}(\boldsymbol{\xi}^{(N)}) & \cdots & \psi_{\boldsymbol{\alpha}_{P-1}}(\boldsymbol{\xi}^{(N)}) \end{pmatrix}$$

$$\hat{\boldsymbol{a}} = (\boldsymbol{\Psi}^{\mathsf{T}}\boldsymbol{\Psi})^{-1}\boldsymbol{\Psi}^{\mathsf{T}}\mathcal{Y}$$

Well-posed problem if $N > P$

Least angle regression stepwise



Least angle regression stepwise

First, initialize coefficients to 0 & initial residual to the vector of observations.
At each iteration:

- Find the vector ψ_{α_j} which is the most correlated with the current residual
- Move jointly $(\psi_{\alpha_j}, \psi_{\alpha_k})^T$ coefficients until another vector ψ_{α_l} is much correlated with the current residual

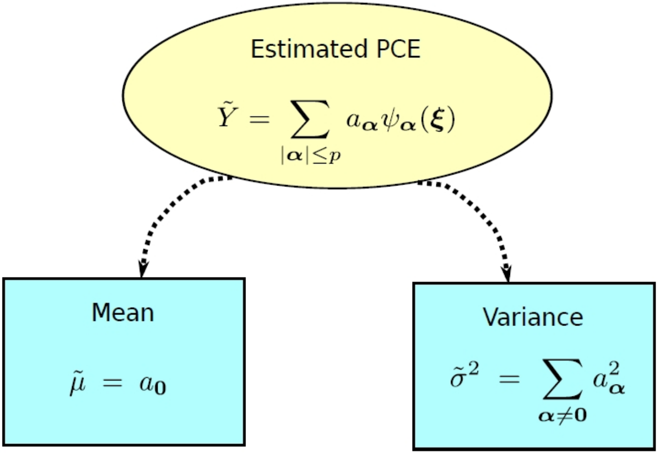
Error indicator

- Q^2 cross-validation on independent test data set
- Due to its linear-regression form, leave-one-out error for polynomial chaos expansions can be obtained without calculating N metamodels:

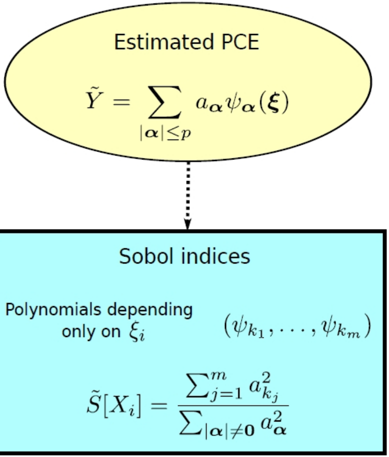
$$Err_{LOO} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{f(\xi^{(i)}) - \hat{f}_{\boldsymbol{a}}(\xi^{(i)})}{1 - h_i}\right)^2$$

where $\hat{f}_{\boldsymbol{a}}$ is the metamodel computed on the full data set, h_i is i-th diagonal term of the hat matrix $\Psi\left(\Psi^T\Psi\right)^{-1}\Psi^T$

Post-processing: closed-form mean and variance



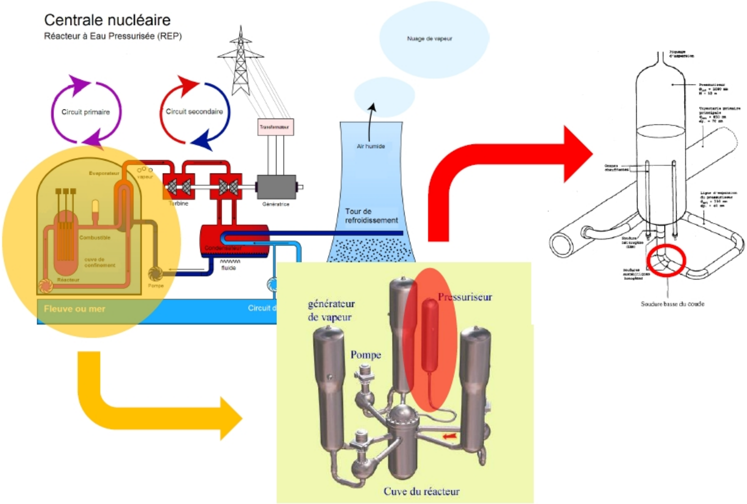
Post-processing: closed-form Sobol indices



Interaction indices can also be derived!

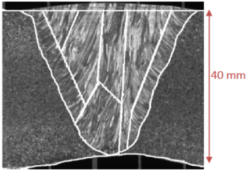
Outline

Primary circuit - Bent tube weld



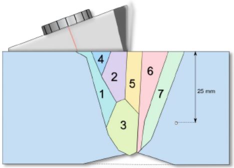
Inspection configuration and modelling

Weld description with
7 homogeneous domains



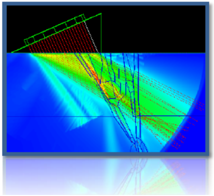
Chassignole et al. , QNDE, 1999 &
Chassignole et al., Ultrasonics, 2009

Inspection configuration



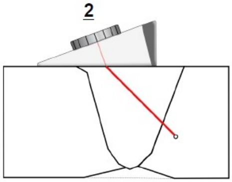
- L 45° waves
- Detection of a sided drilled hole after passing through the weld

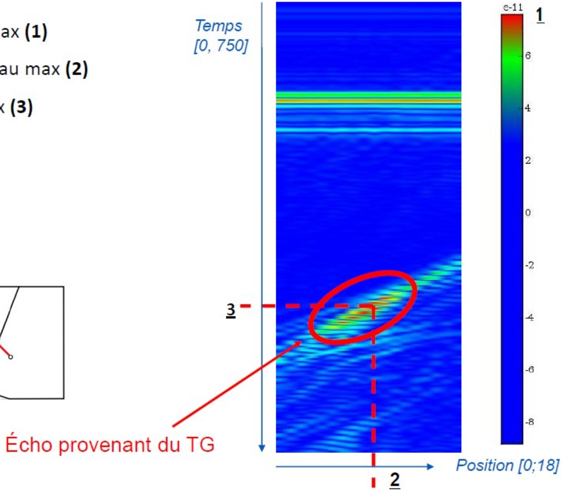
Finite element model
(Athena 2D code)



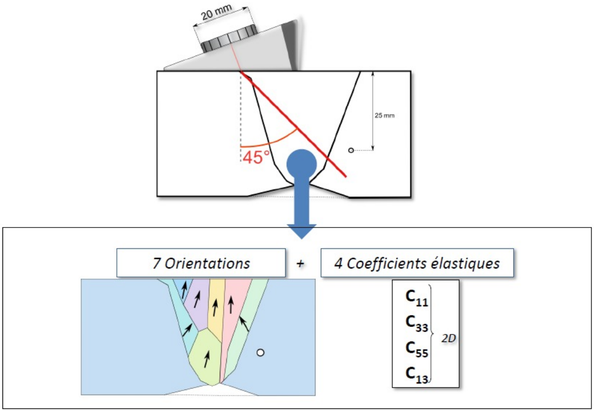
Quantities of interest : inspection results

- Amplitude de l'écho max **(1)**
- La position du capteur au max **(2)**
- Le temps de vol du max **(3)**





Uncertain input data



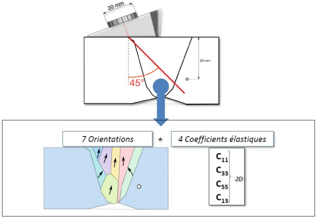
Problem : What is the sensitivity of the NDT output to each input parameter?

Strategy: Evaluate the Sobol sensitivity indices

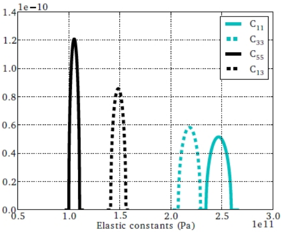
Specification of the input PDFs and chaos basis

Input variables: $X_1, \ldots, X_8$

(independent)



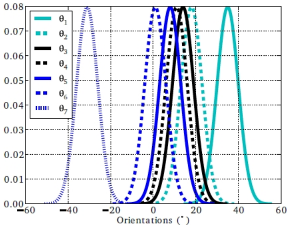
Elastic constants ($X_1 - X_4$)



Beta PDFs

$\xi_i \sim \mathcal{U}(-1,1) \rightarrow$ Legendre poly.

Orientations ($X_5 - X_{11}$)

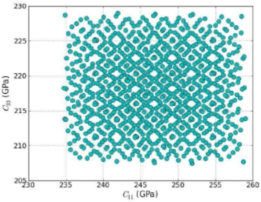


Normal PDFs

$\xi_i \sim \mathcal{N}(0,1) \rightarrow$ Hermite poly.

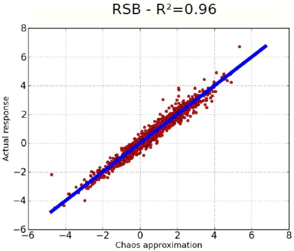
Construction of the polynomial chaos

- Design of experiments: quasi-random sample of size $N = 2\,000$

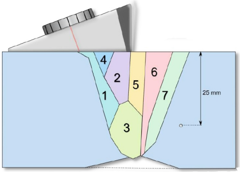


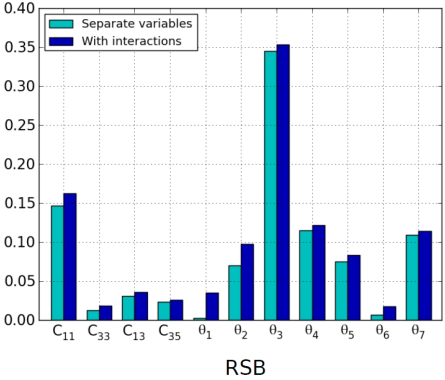
- Distributed calls to the FE model (cluster)

- Fit of chaos proxies with varying degree
  (70% of the sample points)
  $\rightarrow$ optimal degree $p = 3$
- Validation with the 30% remaining points



Sensitivity analysis – Signal-to-noise ratio





Almost no interaction effect (additive structure)

Variability mostly due to the orientations (plus C_{11})

θ_3 plays a major role here $\rightarrow$ Check a finer weld description

Outline

Metamodels

Polynomial chaos expansion

Applications in non-destructive testing

Conclusions

Metamodels & Polynomial chaos

- For a given problem, ideally test several types of metamodels
- Polynomial chaos expansion and Kriging are in OpenTurns
- It is worth assessing carefully the metamodel quality prior to going further

Thank you