Multi-fidelity Co-kriging model

Application to Sequential design and Sensitivity Analysis

Loic Le Gratiet[1 2]

[1] Université Denis-Diderot, Paris, France

[2] CEA, DAM, DIF, F-91297 Arpajon, France

May 17th 2013
Atelier GdR Mascot-Num

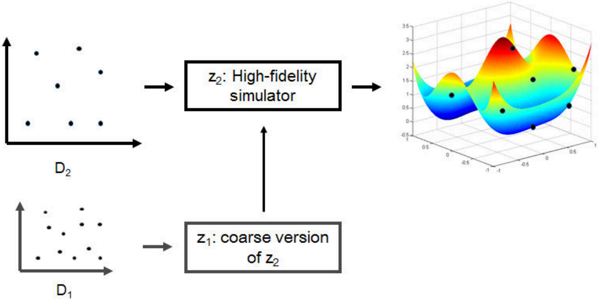
Outline

Outline

Framework

- Mathematical formalism:

$$z_2(x) : x \in Q \subset \mathbb{R}^d \mapsto \mathbb{R}$$



- Objective: we want to build a surrogate model for the output of the high-fidelity simulator $z_2(x)$.
- Framework: a fast and coarse version $z_1(x)$ of $z_2(x)$ is available.

Modelling with recursive formulation

- We consider the model:

$$\begin{cases} Z_2(x) = \rho \tilde{Z}_1(x) + \delta(x) \\ \tilde{Z}_1(x) \perp \delta(x) \end{cases}$$

where

$$\tilde{Z}_1(x) \sim [Z_1(x) | \mathbf{Z}_1 = \mathbf{z}_1, \boldsymbol{\beta}_1, \sigma_1^2, \boldsymbol{\theta}_1]$$

$\mathbf{D}_2 \subset \mathbf{D}_1$ and

$$Z_1(x) \sim \mathrm{GP}\left(\mathbf{f}_1'(x)\boldsymbol{\beta}_1, \sigma_1^2 r_1(x, \tilde{x}; \boldsymbol{\theta}_1)\right)$$

$$\delta(x) \sim \mathrm{GP}\left(\mathbf{f}_\delta'(x)\boldsymbol{\beta}_\delta, \sigma_\delta^2 r_\delta(x, \tilde{x}; \boldsymbol{\theta}_\delta)\right)$$

- The predictive distribution is given by (Simple Cokriging):

$$[Z_2(x) | \mathbf{Z}_2 = \mathbf{z}_2, \mathbf{Z}_1 = \mathbf{z}_1, \rho, \boldsymbol{\beta}, \boldsymbol{\sigma}^2, \boldsymbol{\theta}] = \mathrm{GP}\left(\mu_{Z_2,SK}(x), \sigma_{Z_2,SK}^2(x, \tilde{x})\right)$$

where $\boldsymbol{\beta} = (\boldsymbol{\beta}_1, \boldsymbol{\beta}_\delta)$, $\boldsymbol{\sigma}^2 = (\sigma_1^2, \sigma_\delta^2)$ and $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_\delta)$.

Important properties

- In a Universal Cokriging framework, we denote by:

$$\begin{aligned} \mu_{Z_2,UK}(x) &= \mathbb{E}\left[Z_2(x)|\mathbf{Z}_2=\mathbf{z}_2,\mathbf{Z}_1=\mathbf{z}_1,\boldsymbol{\sigma}^2,\boldsymbol{\theta}\right] \\ \sigma^2_{Z_2,UK}(x) &= \mathrm{var}\left(Z_2(x)|\mathbf{Z}_2=\mathbf{z}_2,\mathbf{Z}_1=\mathbf{z}_1,\boldsymbol{\sigma}^2,\boldsymbol{\theta}]\right) \end{aligned}$$

- Predictive mean and variance decomposition:

$$\begin{aligned} \mu_{Z_2,UK}(x) &= \hat{\rho}\mu_{Z_1,UK}(x)+\mu_{\delta,UK}(x) \\ \sigma^2_{Z_2,UK}(x) &= \hat{\rho}^2\sigma^2_{Z_1,UK}(x)+\sigma^2_{\delta,UK}(x) \end{aligned}$$

- Fast Cross Validation equations can be derived. In a Leave-One-Out case, we denote by:

$$\begin{aligned} \varepsilon_{LOO,2}(x_i^{(2)}) &= \hat{\rho}_{-i}\varepsilon_{LOO,1}(x_i^{(2)})+\varepsilon_{LOO,\delta}(x_i^{(2)}) \\ \sigma^2_{LOO,2}(x_i^{(2)}) &= \hat{\rho}^2_{-i}\sigma^2_{LOO,1}(x_i^{(2)})+\sigma^2_{LOO,\delta}(x_i^{(2)}) \end{aligned}$$

Outline

Sequential design strategy

- Objective: we want to reduce the generalization error:

$$\mathrm{IMSE} = \hat{\rho}^2 \int_Q \sigma^2_{Z_1,UK}(x)dx + \int_Q \sigma^2_{\delta,UK}(x)dx$$

- Sequential design strategy: we choose a new point x_{n+1} such that:

$$x_{n+1} = \arg\max_x \sigma^2_{Z_2,SK}(x)$$

- Question: which level of code is worth being simulated?
  - What is the contribution of each code level on the model's error?
  - What is the expected error reduction?
  - What is the the CPU-time ratio between the simulators?

Code level selection

- What is the contribution of each code level on the model's error?

$$\sigma^2_{Z_2,UK}(x) = \hat{\rho}^2 \sigma^2_{Z_1,UK}(x) + \sigma^2_{\delta,UK}(x)$$

- What is the expected error reduction?
  - Error reduction for the level $z_1(x)$:

$$\hat{\rho}^2 \sigma^2_{Z_1,UK}(x_{n+1}) \prod_{i=1}^{d} \boldsymbol{\theta}_1^i$$

  - Error reduction for the bias $\delta(x)$

$$\sigma^2_{\delta,UK}(x_{n+1}) \prod_{i=1}^{d} \boldsymbol{\theta}_\delta^i$$

- What is the the CPU-time ratio between the simulators?
  - C: CPU-time ratio between $z_2(x)$ and $z_1(x)$.
  - 1 run for $z_2(x) \Leftrightarrow C+1$ runs for $z_1(x)$ (i.e. $\mathbf{D}_2 \subset \mathbf{D}_1$)

Code level selection

- Potential uncertainty reduction for level 2:

$$\hat{\rho}^2 \sigma^2_{Z_1,UK}(x_{n+1}) \prod_{i=1}^{d} \theta_1^i + \sigma^2_{\delta,UK}(x_{n+1}) \prod_{i=1}^{d} \theta_\delta^i$$

- Potential uncertainty reduction for level 1:

$$(C+1)\, \hat{\rho}^2 \sigma^2_{Z_1,UK}(x_{n+1}) \prod_{i=1}^{d} \theta_1^i$$

- It is worth to simulate the code level $z_1(x)$ when:

$$(C+1)\, \hat{\rho}^2 \sigma^2_{Z_1,UK}(x_{n+1}) \prod_{i=1}^{d} \theta_1^i > \hat{\rho}^2 \sigma^2_{Z_1,UK}(x_{n+1}) \prod_{i=1}^{d} \theta_1^i + \sigma^2_{\delta,UK}(x_{n+1}) \prod_{i=1}^{d} \theta_\delta^i$$

i.e.
$$\frac{\hat{\rho}^2 \sigma^2_{Z_1,UK}(x_{n+1}) \prod_{i=1}^{d} \theta_1^i}{\hat{\rho}^2 \sigma^2_{Z_1,UK}(x_{n+1}) \prod_{i=1}^{d} \theta_1^i + \sigma^2_{\delta,UK}(x_{n+1}) \prod_{i=1}^{d} \theta_\delta^i} > \frac{1}{(C+1)}$$

New design strategy

- Problem: the following criterion does not take into account the real model error

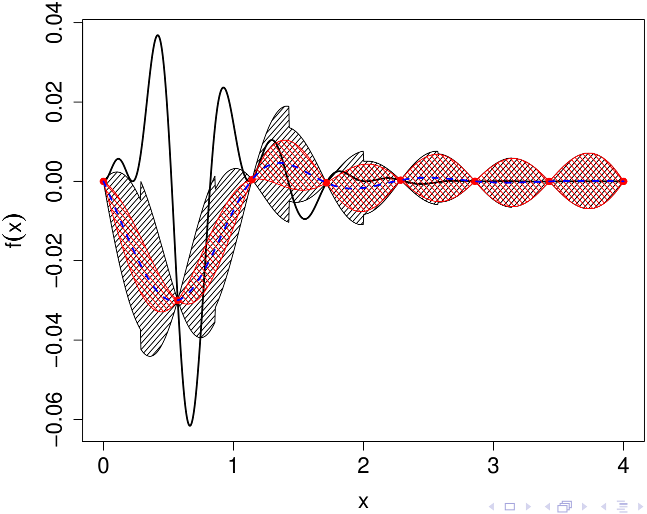
$$x_{n+1} = \arg\max_x \sigma^2_{Z_2,SK}(x)$$

- New strategy: to deal with the real error, we choose the following criterion

$$x_{n+1} = \arg\max_x \quad \hat{\rho}^2 \sigma^2_{Z_1,UK}(x) \left(1 + \sum_{i=1}^{n_1} \frac{\varepsilon^2_{LOO,1}(x_i^{(1)})}{\sigma^2_{LOO,1}(x_i^{(1)})} \mathbf{1}_{x \in V_{i,1}} \right) + \sigma^2_{\delta,UK}(x) \left(1 + \sum_{i=1}^{n_2} \frac{\varepsilon^2_{LOO,\delta}(x_i^{(1)})}{\sigma^2_{LOO,\delta}(x_i^{(1)})} \mathbf{1}_{x \in V_{i,2}} \right)$$

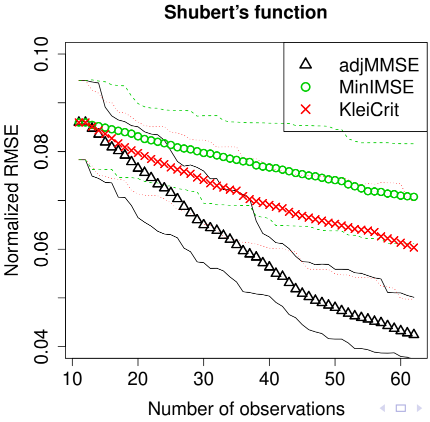
where $V_{i,j}$ is the Voronoi cell associated to the point $x_i^{(j)}$, $j = 1, 2$, $i = 1, \ldots, n_j$.

New criterion: illustration



Academic example: Schubert's function

$$f(x) = \left(\sum_{i=1}^{5} i\cos\left((i+1)x^1 + i\right)\right)\left(\sum_{i=1}^{5} i\cos\left((i+1)x^2 + i\right)\right)$$

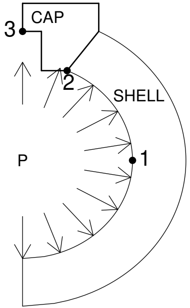


Example: Spherical tank under internal pressure

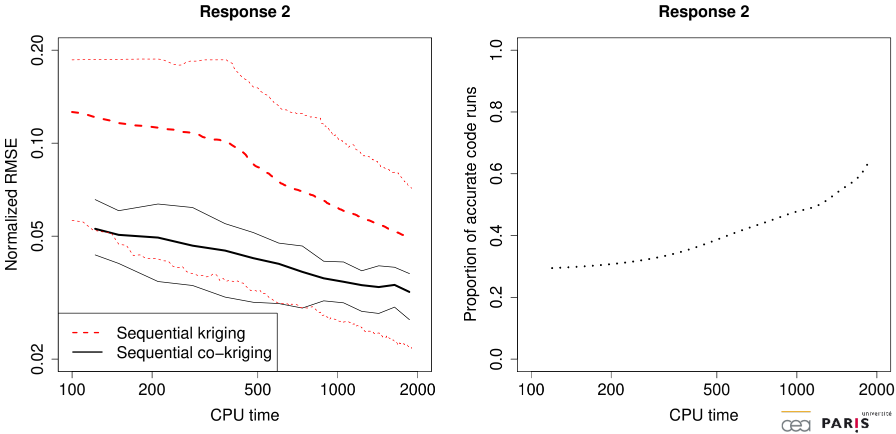
$$z_1(x) = \frac{3}{2}\frac{(R_{int} + T_{shell})^3}{(R_{int} + T_{shell})^3 - R_{int}^3} P$$

- $z_2(x)$: von Mises stress at point 1, 2 or 3.

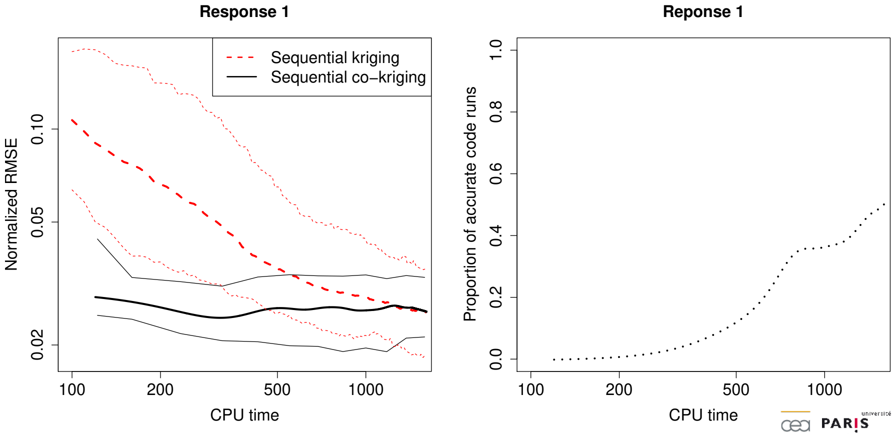
$$x = \begin{pmatrix} P \\ R_{int} \\ T_{shell} \\ T_{cap} \\ E_{shell} \\ E_{cap} \\ \sigma_{y,shell} \\ \sigma_{y,cap} \end{pmatrix}$$



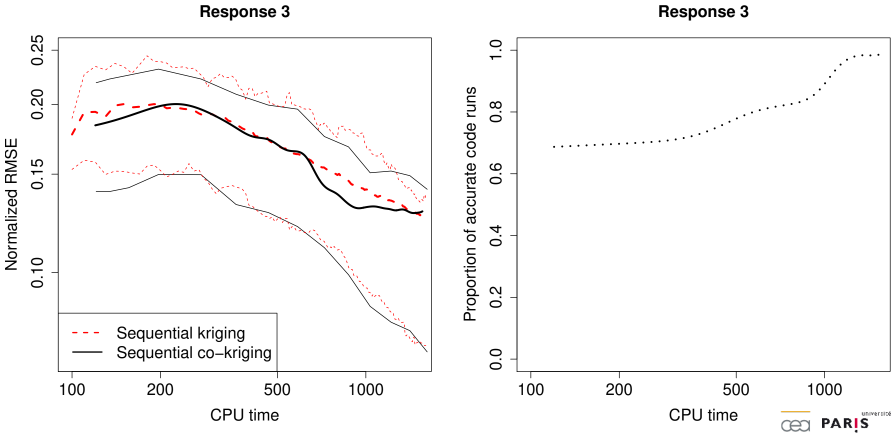
Result at point 2



Result at point 1



Result at point 3



Outline

Sobol indices

- The input is a random vector $X = (X^{d_1}, X^{d_2})$, $d = d_1 + d_2$, with measure

$$\mu = \mu^{d_1} \otimes \mu^{d_2}$$

  on $Q = Q^{d_1} \times Q^{d_2}$.

- Objective: we are interested in evaluating the closed Sobol index:

$$\mathrm{S}^{d_1} = \frac{\mathrm{V}^{d_1}}{\mathrm{V}} = \frac{\mathrm{var}_X\left(\mathbb{E}_X\left[z_2(x)|X^{d_1}\right]\right)}{\mathrm{var}_X\left(z_2(x)\right)}$$

- Let us consider $\tilde{Z}_2(x)$ such that:

$$\tilde{Z}_2(x) \sim [Z_2(x)|\mathbf{Z}_2 = \mathbf{z}_2, \mathbf{Z}_1 = \mathbf{z}_1, \boldsymbol{\sigma}^2, \boldsymbol{\theta}]$$

- We consider the estimator:

$$\tilde{\mathrm{S}}^{d_1} = \frac{\mathrm{var}_X\left(\mathbb{E}_X\left[\tilde{Z}_2(x)|X^{d_1}\right]\right)}{\mathrm{var}_X\left(\tilde{Z}_2(x)\right)}$$

Sobol index estimation

- Remark 1: $\tilde{S}^{d_1}$ is now a random variable.
- Let us consider $(X_i)_{i=1,\dots,m}$ and $(\tilde{X}_i)_{i=1,\dots,m}$ such that

$$X_i = (X_i^{d_1}, X_i^{d_2}) \quad \tilde{X}_i = (X_i^{d_1}, \tilde{X}_i^{d_2}) \quad X_i^{d_2} \perp \tilde{X}_i^{d_2}$$

- $\tilde{S}^{d_1}$ can be estimated with [Janon et al. (2012)]:

$$\tilde{S}_m^{d_1} = \frac{\frac{1}{m}\sum_{i=1}^m \tilde{Z}_2(X_i)\tilde{Z}_2(\tilde{X}_i) - \left(\frac{1}{2m}\sum_{i=1}^m \tilde{Z}_2(X_i) + \tilde{Z}_2(\tilde{X}_i)\right)^2}{\frac{1}{m}\sum_{i=1}^m \tilde{Z}_2(X_i)^2 - \left(\frac{1}{2m}\sum_{i=1}^m \tilde{Z}_2(X_i) + \tilde{Z}_2(\tilde{X}_i)\right)^2}$$

- Remark 2: the estimator $\tilde{S}_m^{d_1}$ has two sources of uncertainty:
  1. The Monte-Carlo error
  2. The meta-model error

Sampling from $\tilde{S}_m^{d_1}$

- Generate a sample $\mathbf{x} = \{(x_i)_{i=1,\dots,m}, (\tilde{x}_i)_{i=1,\dots,m}\}$ of the random vector $(X_i, \tilde{X}_i)_{i=1,\dots,m}$.
- For $k = 1, \dots, N_Z$
  - Sample a realization $\tilde{z}_2(\mathbf{x})$ of $\tilde{Z}_2(\mathbf{x})$ at points in $\mathbf{x}$.
  - Compute $\tilde{S}_{m,k,1}^{d_1}$ from $\tilde{z}_2(\mathbf{x})$.
  - For $l = 2, \dots, B$
    - Sample with replacements two samples $\mathbf{u}$ and $\tilde{\mathbf{u}}$ from $\{(x_i)_{i=1,\dots,m}\}$ and $\{(\tilde{x}_i)_{i=1,\dots,m}\}$.
    - Compute $\tilde{S}_{m,k,l}^{d_1}$ from $\tilde{z}_2(\mathbf{x}^B)$ with $\mathbf{x}^B = \{\mathbf{u}, \tilde{\mathbf{u}}\}$.

- Two issues:
  1. How to sample from $\tilde{Z}_2(x) \sim [Z_2(x)|\mathbf{Z}_2 = \mathbf{z}_2, \mathbf{Z}_1 = \mathbf{z}_1, \boldsymbol{\sigma}^2, \boldsymbol{\theta}]$?
  2. How to deal with large m ?

1. Sampling from $\tilde{Z}_2(x)$

- We consider the process $\tilde{Z}_2^*(x) \sim [Z_2(x) | \mathbf{Z}_2 = \mathbf{z}_2, \mathbf{Z}_1 = \mathbf{z}_1, \rho, \boldsymbol{\beta}_\delta, \boldsymbol{\sigma}^2, \boldsymbol{\theta}]$
- $\tilde{Z}_2^*(x)$ is Gaussian and can be written with the following form:

$$\tilde{Z}_2^*(x) = \rho \tilde{Z}_1^*(x) + \delta^*_{\rho, \boldsymbol{\beta}_\delta}(x)$$

where

$$\begin{pmatrix} \rho \\ \boldsymbol{\beta}_\delta \end{pmatrix} = \mathcal{N}\left(\mu_{\rho, \boldsymbol{\beta}_\delta}, \Sigma_{\rho, \boldsymbol{\beta}_\delta}\right)$$

$$\tilde{Z}_1^*(x) \sim \mathrm{GP}\left(\mu_{\tilde{Z}_1^*(x)}, \Sigma_{\tilde{Z}_1^*(x)}\right)$$

and conditionally to $(\rho, \boldsymbol{\beta}_\delta)$

$$\delta^*_{\rho, \boldsymbol{\beta}_\delta}(x) \sim \mathrm{GP}\left(\mu_{\delta^*(x)}, \Sigma_{\delta^*(x)}\right)$$

1. Sampling from $\tilde{Z}_2(x)$

- We can sample from the distribution $[Z_2(x)|\mathbf{Z}_2 = \mathbf{z}_2, \mathbf{Z}_1 = \mathbf{z}_1, \boldsymbol{\sigma}^2, \boldsymbol{\theta}]$ as follows:
  1. Sample a realization $\tilde{z}_1^*(x)$ of $\tilde{Z}_1^*(x)$.
  2. Sample a realization $(\rho^r, \boldsymbol{\beta}_\delta^r)$ of $(\rho, \boldsymbol{\beta}_\delta)$.
  3. Sample a realization $\tilde{\delta}^*_{\rho^r, \boldsymbol{\beta}_\delta^r}(x)$ of $\delta^*_{\rho^r, \boldsymbol{\beta}_\delta^r}(x)$.
  4. Set $\tilde{z}_2(x) = \rho^r \tilde{z}_1^*(x) + \tilde{\delta}^*_{\rho^r, \boldsymbol{\beta}_\delta^r}(x)$
- Remember that we have $Z_1(x) \sim \mathrm{GP}\left(\mathbf{f}_1'(x)\boldsymbol{\beta}_1, \sigma_1^2 r_1(x, \tilde{x})\right)$ and

$$\tilde{Z}_1^*(x) \sim \mathrm{GP}\left(\mu_{\tilde{Z}_1^*}(x), \Sigma_{\tilde{Z}_1^*}(x, \tilde{x})\right)$$

with

$$\mu_{\tilde{Z}_1^*}(x) = \mathbf{f}_1'(x)\hat{\boldsymbol{\beta}}_1 + \mathbf{r}_1'(x)\mathbf{R}_1^{-1}\left(\mathbf{z}_1 - \mathbf{F}_1\hat{\boldsymbol{\beta}}_1\right)$$

$$\Sigma_{\tilde{Z}_1^*}(x, \tilde{x}) = \sigma_1^2 \left(1 - \begin{pmatrix} \mathbf{f}_1'(x) & \mathbf{r}_1'(x) \end{pmatrix} \begin{pmatrix} 0 & \mathbf{F}_1' \\ \mathbf{F}_1 & \mathbf{R}_1 \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{f}_1(\tilde{x}) \\ \mathbf{r}_1(\tilde{x}) \end{pmatrix}\right)$$

$$\hat{\boldsymbol{\beta}}_1 = \left(\mathbf{F}_1'\mathbf{R}_1^{-1}\mathbf{F}_1\right)^{-1} \mathbf{F}_1'\mathbf{R}_1^{-1}\mathbf{z}_1$$

1. Sampling with large m

- Let us consider $Y_1(x) \sim \mathrm{GP}\left(0, \sigma_1^2 r_1(x, \tilde{x})\right)$ and

$$\tilde{Y}_1^*(x) = \mu_{\tilde{Z}_1^*}(x) - \mu_{\tilde{Y}_1^*}(x) + Y_1(x)$$

  where

$$\mu_{\tilde{Y}_1^*}(x) = \mathbf{f}_1'(x)\tilde{\beta}_1 + \mathbf{r}_1'(x)\mathbf{R}_1^{-1}\left(Y_1(\mathbf{D}_1) - \mathbf{F}_1\tilde{\beta}_1\right)$$

$$\tilde{\beta}_1 = \left(\mathbf{F}_1'\mathbf{R}_1^{-1}\mathbf{F}_1\right)^{-1}\mathbf{F}_1'\mathbf{R}_1^{-1}Y_1(\mathbf{D}_1)$$

  then:

$$\boxed{\tilde{Y}_1^*(x) \quad \stackrel{\mathcal{L}}{=} \quad \tilde{Z}_1^*(x)}$$

- Consequence: we can sample from the conditional distribution of $\tilde{Z}_1^*(x)$ by sampling from the unconditioned distribution of $Y_1(x)$ and then by applying a linear transformation on it.

1. Sampling with large m

- The suggested approach allows for avoiding ill-conditioned covariance matrices (indeed, $\Sigma_{\tilde{Z}_1^*}(x, \tilde{x})$ is close to zero around points in $\mathbf{D}_1$) .
- Further, it allows for using efficient algorithms to compute the realizations, e.g:
  - The stationary case: $r_1(x, \tilde{x}) = r_1(x - \tilde{x})$

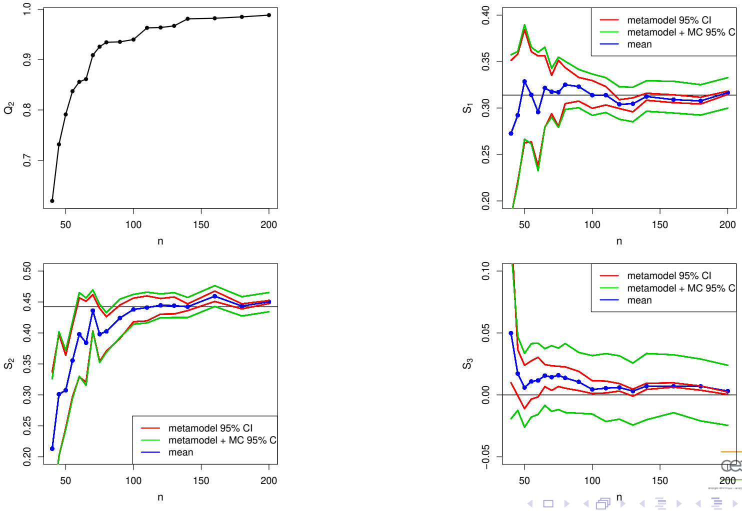
$$Y_1(x) = \int \sqrt{S(\omega)} e^{2\pi i \langle \omega, x \rangle} \hat{\eta}_\omega d\omega, \qquad S(\omega) = \int e^{-2\pi i \langle \omega, h \rangle} r_1(h) dh$$

  - Tensorised covariance kernels: $r_1(x, \tilde{x}) = \prod_{i=1}^d r_1^i(x^i, \tilde{x}^i)$

$$Y_1(x) = \sum_{p_1, \dots, p_d \geq 0} \prod_{i=1}^d \sqrt{\lambda_{p_i}} \phi_{p_i}(x) Z_{p1, \dots, p_d}, \quad Z_{p_1, \dots, p_d} \sim \mathcal{N}(0, 1)$$

where $r_1^i(x^i, \tilde{x}^i) = \sum_{p_i \geq 0} \lambda_{p_i} \phi_{p_i}(x^i) \phi_{p_i}(\tilde{x}^i)$ (Mercer's Theorem with the right assumptions).

Ishigami function



Outline

Bibliography Multi-Fidelity

KENNEDY, M. C. & O'HAGAN, A. (2000) Predicting the output from a complex computer code when fast approximations are available. *Biometrika* **87**, 1-13.

BOYLE, PHILLIP & FREAN, MARCUS (2005) Dependent gaussian processes. *Advances in Neural Information Processing Systems* **17**, 217–224.

QIAN, Z. & JEFF WU, C. F. (2008) Bayesian Hierarchical Modeling for Integrating Low-accuracy and High-accuracy Experiments. *Technometrics* **50**, 192-204.

LE GRATIET, L. (2012), *Bayesian analysis of hierarchical multi-fidelity codes*, submitted to SIAM UQ.

LE GRATIET, L. & GARNIER, J. (2012), *Recursive co-kriging model for Design of Computer experiments with multiple levels of fidelity with an application to hydrodynamic*, submitted to International Journal of Uncertainty Quantification.

R CRAN package MuFiCokriging:

http://cran.r-project.org/web/packages/MuFiCokriging

Bibliography Sequential Design

- BATES, R.A., BUCK, R., RICCOMAGNO, E. & WYNN, H. (1996), *Experimental design of computer experiments for large systems*, Journal of the Royal Statistical Society B, 58 (1):77-94.
- KLEIJNEN, J. & VAN BEERS, W. (2004), *Application-driven sequential design for simulation experiments: Kriging metamodelling.*, Journal of the Operational Research Society, 55:876-883.
- LE GRATIET, L. & CANNAMELA CLAIRE (2012), *Kriging-based sequential design strategies using fast cross-validation techniques with extensions to multi-fidelity computer codes*, submitted to TECHNOMETRICS.

Bibliography Sensitivity Analysis

J.E. OAKLEY & A O'HAGAN (2004), *Probabilistic sensitivity analysis of complex models: A bayesian approach*, *Journal of the Royal Statistical Society B*, **66**(3):751–769, 2004.

A. MARREL, B. IOOSS, B. LAURENT & O. ROUSTANT (2012), *Calculations of Sobol indices for the gaussian process metamodel*, *Reliability Engineering and System Safety*, **94**:742–751, 2009.

A. JANON, T. KLEIN, A. LAGNOUX, M. NODET & C. PRIEUR (2012), *Asymptotic normality and efficiency of two Sobol index estimators*, submitted to ESAIM probability and statistics, 2012.

LE GRATIET, L. & CANNAMELA, C. (2013), *Multi-Fidelity Sensitivity Analysis*, Oral presentation to SAMO 2013, Paper in progress with Iooss B..

Thanks for your attention