Adaptive numerical designs for the calibration of computer codes

Guillaume Damblin

French Alternative Energies and Atomic Energy Commission,
Division of nuclear energy (DEN)

(Joint work with Pierre Barbillon, Merlin Keller, Alberto Pasanisi and Eric Parent)

SAMO - November 30th to December 3rd 2016 - University of Reunion Island

Presentation Outline

Notations

Let $r(\mathbf{x}) \in \mathbb{R}$ be a physical quantity of interest (q.o.i.) where:

- $\mathbf{x} \in \mathcal{X}$ is a vector of control variables,
- only noisy values $z(\mathbf{x}) = r(\mathbf{x}) + \epsilon(\mathbf{x})$ can be collected.

Let $y(\mathbf{x}, \mathbf{t})$ be a deterministic computer code aiming at predicting $r(\mathbf{x})$:

- treated as a **black-box**,
- $\mathbf{t} \in \mathcal{T}$ is a vector of uncertain parameters (may have no physical interpretation).

Calibration consists in determining the values of $\mathbf{t} = \boldsymbol{\theta}$ making the best agreement between $r(\mathbf{x})$ and $y(\mathbf{x}, \boldsymbol{\theta})$.

Statistical model for calibration

Basic assumptions:

- for any $\boldsymbol{x} \in \mathcal{X}$, the physical q.o.i. $r(\boldsymbol{x}) \in \mathbb{R}$ is deterministic,
- $\exists \boldsymbol{\theta} \in \mathcal{T}$; $\forall x \in \mathcal{X}$ $y(\boldsymbol{x}, \boldsymbol{\theta}) = r(\boldsymbol{x})$.

Statistical model:

- based on n physical measurements $\{(\boldsymbol{x}_1^f, z_1^f), \ldots, (\boldsymbol{x}_n^f, z_n^f)\}$,
- $\boldsymbol{x}_i^f \in \mathbb{R}^d \rightarrow \boxed{\text{true physical quantity } r(\boldsymbol{x}_i^f)} \rightarrow z_i^f \in \mathbb{R}$, then:

$$\begin{aligned} z_i^f &= r(\boldsymbol{x}_i^f) + \epsilon_i \\ &= y(\boldsymbol{x}_i^f, \boldsymbol{\theta}) + \epsilon_i \end{aligned} \tag{1}$$

where $\epsilon_i \underset{i.i.d.}{\sim} \mathcal{N}(0, \sigma_m^2)$ with σ_m^2 is assumed known.

Calibration goal: estimation of $\boldsymbol{\theta}$ from Equation (1),

Objective Bayesian calibration

$\mathbf{X}^f = [\boldsymbol{x}_1^f, \cdots, \boldsymbol{x}_n^f]^T$, $\boldsymbol{z}^f = (z_1^f, \cdots, z_n^f)^T$.

Posterior distribution:

$$\pi(\boldsymbol{\theta}|\boldsymbol{z}^f) \propto \frac{1}{(\sqrt{2\pi}\sigma_m)^n} \exp\left[-\frac{1}{2\sigma_m^2} SS(\boldsymbol{\theta})\right] \pi(\boldsymbol{\theta}).$$

where $SS(\boldsymbol{\theta}) = ||\boldsymbol{z}^f - y(\mathbf{X}^f, \boldsymbol{\theta})||^2$ and $\pi(\boldsymbol{\theta})$ is non-informative.

1) $y(\boldsymbol{x}, \boldsymbol{\theta})$ can be non-linear with regard to $\mathbf{t} \Longrightarrow$ no closed-form expression for $\pi(\boldsymbol{\theta}|\boldsymbol{z}^f)$,
2) simulations are often time-consuming $\Longrightarrow$ MCMC sampling is infeasible.

Gaussian process(GP)-based posterior distribution:

- set a GP emulator on either $SS(\boldsymbol{\theta})$ or the likelihood [5]
- set a GP emulator on $y(.,.)$: a more flexible method !

GP-based posterior distribution

Gaussian process emulator on $\mathcal{X} \times \mathcal{T}$ [6]:

- Numerical design of experiments: $\mathbf{D}_M = [(\mathbf{x}_1, \mathbf{t}_1), \cdots, (\mathbf{x}_M, \mathbf{t}_M)]^T$,
- GP emulator: $y(\mathbf{x}, \mathbf{t}) | y(\mathbf{D}_M) \sim \mathcal{GP}\left(\mu^M_{\boldsymbol{\beta}, \boldsymbol{\Psi}}(\mathbf{x}, \mathbf{t}),\, V^M_{\boldsymbol{\Psi}, \sigma^2}((\mathbf{x}, \mathbf{t}), (\mathbf{x}', \mathbf{t}'))\right)$

GP-based posterior distribution [3, 1]:

1) $(\hat{\boldsymbol{\beta}}, \hat{\sigma}^2, \hat{\boldsymbol{\Psi}}) = \underset{\boldsymbol{\beta}, \sigma^2, \boldsymbol{\Psi}}{\operatorname{argmax}}\, \mathcal{L}(\boldsymbol{\beta}, \sigma^2, \boldsymbol{\Psi} | y(\mathbf{D}_M))$ and $\mathbf{D}_{\boldsymbol{\theta}} = [(\boldsymbol{x}^f_1, \boldsymbol{\theta}), \cdots, (\boldsymbol{x}^f_n, \boldsymbol{\theta})]^T$,

2) $\pi^C(\boldsymbol{\theta} | \mathbf{z}^f, y(\mathbf{D}_M)) \propto \mathcal{L}^C(\mathbf{z}^f | \boldsymbol{\theta}, y(\mathbf{D}_M)) \pi(\boldsymbol{\theta})$ where

$$\mathcal{L}^C(\mathbf{z}^f | \boldsymbol{\theta}, y(\mathbf{D}_M)) \propto |V^M_{\hat{\boldsymbol{\Psi}}, \hat{\sigma}^2}(\mathbf{D}_{\boldsymbol{\theta}}) + \sigma^2_m \mathbf{I}_n|^{-1/2} \exp\left\{ -\frac{1}{2}\left[(\boldsymbol{z}^f - \mu^M_{\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\Psi}}}(\mathbf{D}_{\boldsymbol{\theta}}))^{\mathrm{T}}) \right.\right.$$
$$\left.\left. (V^M_{\hat{\boldsymbol{\Psi}}, \hat{\sigma}^2}(\mathbf{D}_{\boldsymbol{\theta}}) + \sigma^2_m \mathbf{I}_n)^{-1} (\boldsymbol{z}^f - \mu^M_{\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\Psi}}}(\mathbf{D}_{\boldsymbol{\theta}})) \right] \right\}$$

The Kullback-Leibler divergence

The KL divergence measures how far a probability distribution is from a reference [2]:

$$\mathrm{KL}\left(\pi(\boldsymbol{\theta}|\boldsymbol{z}^f)||\pi^C(\boldsymbol{\theta}|\boldsymbol{z}^f, y(\mathbf{D}_M))\right) = \int \pi(\boldsymbol{\theta}|\boldsymbol{z}^f)\Big(\log\left(\pi(\boldsymbol{\theta}|\boldsymbol{z}^f)\right) - \log\left(\pi^C(\boldsymbol{\theta}|\boldsymbol{z}^f, f(\mathbf{D}_M))\right)\Big)\mathrm{d}\boldsymbol{\theta}.$$

Main goal of the work: minimizing this KL-divergence!

- if M is large enough, we proved that,

$$\lim_{M\to\infty} \mathrm{KL}\left(\pi(\boldsymbol{\theta}|\boldsymbol{z}^f)||\pi^C(\boldsymbol{\theta}|\boldsymbol{z}^f, y(\mathbf{D}_M))\right) \tag{2}$$

- if M is small, we need to construct a numerical design $\mathbf{D}_M$ with care!

Main idea : apply the **Expected Improvement (EI)** criterion for minimizing $SS(\theta)$ [4].

The EI criterion designed for $SS(\boldsymbol{\theta})$

Assuming $\mathbf{D}_k$ has been constructed:

- $y_k(.) := y(.)|y(\mathbf{D}_k)$ is the current GP distribution,
- EI criterion applied to $SS(\boldsymbol{\theta})$

$$EI_k(\boldsymbol{\theta}) = \mathbb{E}\left[(m_k - SS_k(\boldsymbol{\theta}))\mathbf{1}_{SS_k(\boldsymbol{\theta})\leq m_k}\right]$$

where $m_k := \min\{SS(\boldsymbol{\theta}_1), \cdots, SS(\boldsymbol{\theta}_{k-1}), SS(\boldsymbol{\theta}_k)\}$ and $SS_k(.)$ has a generalized chi-squared distribution.

Maximization: $\boldsymbol{\theta}_{k+1} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, EI_k(\boldsymbol{\theta})$

- $EI_k(.)$ has no closed-form expression,
- maximization performed in a greedy fashion.

EGO algorithm:

1) **Initial step**: $m_0 = SS(\boldsymbol{\theta}_0)$ is calculated based on an initial design $\mathbf{D}_0$;
2) **For 1 $\leq$ k $\leq$ M**: $\mathbf{D}_{k+1} = \mathbf{D}_k \cup \{(\boldsymbol{x}_1^f, \boldsymbol{\theta}_{k+1}), \cdots, (\boldsymbol{x}_n^f, \boldsymbol{\theta}_{k+1})\}_{1\leq i\leq n}$ (n code runs at each iteration).
   - Update the current minimum $m_{k+1} := \min\{SS_k(\boldsymbol{\theta}_1), \cdots, SS_k(\boldsymbol{\theta}_k), SS_k(\boldsymbol{\theta}_{k+1})\}$,
   - Update the GP distribution $y_{k+1}(.) := y(.)|y(\mathbf{D}_{k+1})$.

One at a time strategy

The previous algorithm does not work anymore when $n \approx M$

One at a time strategy : $\mathbf{D}_{k+1} = \mathbf{D}_k \cup (\boldsymbol{x}^\star, \boldsymbol{\theta}_{k+1})$ where $\boldsymbol{x}^\star \in \{\boldsymbol{x}_1^f, \cdots, \boldsymbol{x}_n^f\}$

Two criteria to pick up $\boldsymbol{x}^\star \in \{\boldsymbol{x}_1^f, \cdots, \boldsymbol{x}_n^f\}$:

1) Based on the uncertainty of the GPE

$$\boldsymbol{x}^\star = \text{argmax } = \mathbb{V}[y_k(\mathbf{x}_i^f, \boldsymbol{\theta}_{k+1})]$$

2) Trade-off between the uncertainty of the GPE and the sensitivity of $\boldsymbol{x}_i^f$ in the calibration process

$$\boldsymbol{x}^\star = \text{argmax } \frac{\mathbb{V}\big(y_k(\mathbf{x}_i^f, \boldsymbol{\theta}_{k+1})\big)}{\max\limits_{i=1,\cdots,n} \mathbb{V}\big(y_k(\mathbf{x}_i^f, \boldsymbol{\theta}_{k+1})\big)} \times \frac{\mathbb{V}_{\boldsymbol{\theta}}[\mu^k_{\boldsymbol{\beta},\boldsymbol{\Psi}}(\mathbf{x}_i^f, \boldsymbol{\theta})]}{\max\limits_{i=1,\cdots,n} \mathbb{V}_{\boldsymbol{\theta}}[\mu^k_{\boldsymbol{\beta},\boldsymbol{\Psi}}(\mathbf{x}_i^f, \boldsymbol{\theta})]}\,.$$

Updating of the current minimum: $m_{k+1} := \min\{\mathbb{E}[SS_k(\boldsymbol{\theta}_1)], \cdots, \mathbb{E}[SS_k(\boldsymbol{\theta}_k)], \mathbb{E}[SS_k(\boldsymbol{\theta}_{k+1})]\}$

Illustration on an academic example

$\mathcal{X} = [0, 1]$, $\mathcal{T} = [5, 15]$:
$y(x, \tau) = (6x - 2)^2 \times \sin(tx - 4)$

Simulated physical data:

- $\mathbf{X}^f = [x_1^f = 0.1, x_2^f = 0.3, x_3^f = 0.8]$,
- $\epsilon_i \sim \mathcal{N}(0, 0.3^2)$,
- $z_i^f = y(x_i^f, \theta) + \epsilon_i$ where $\boldsymbol{\theta = 12}$,

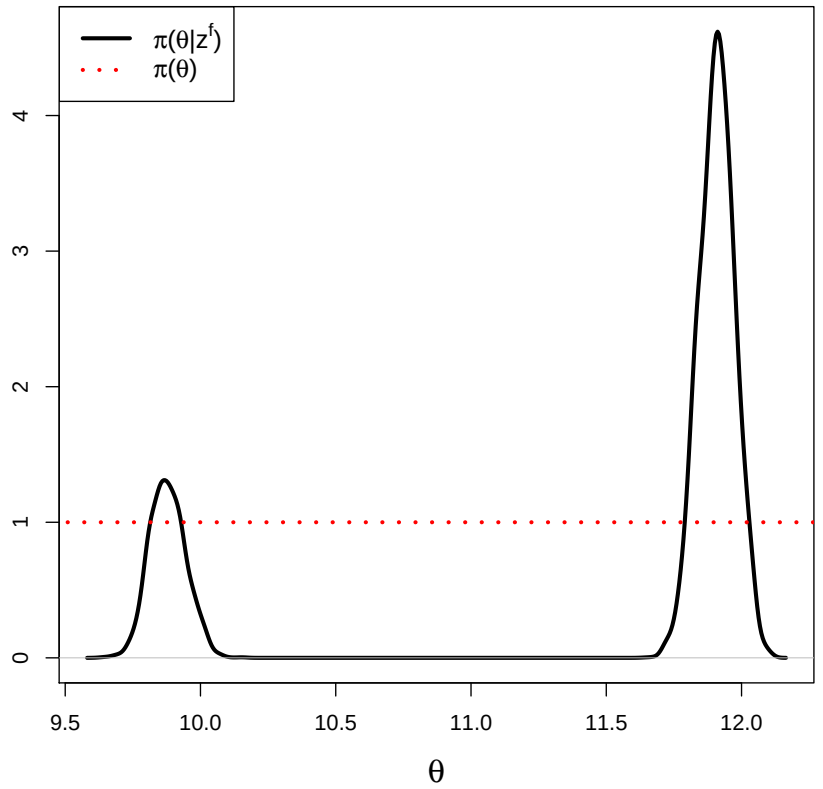
Prior distribution:

- $\pi(\theta) \propto \mathbf{1}_{[5,15]}(\theta)$,

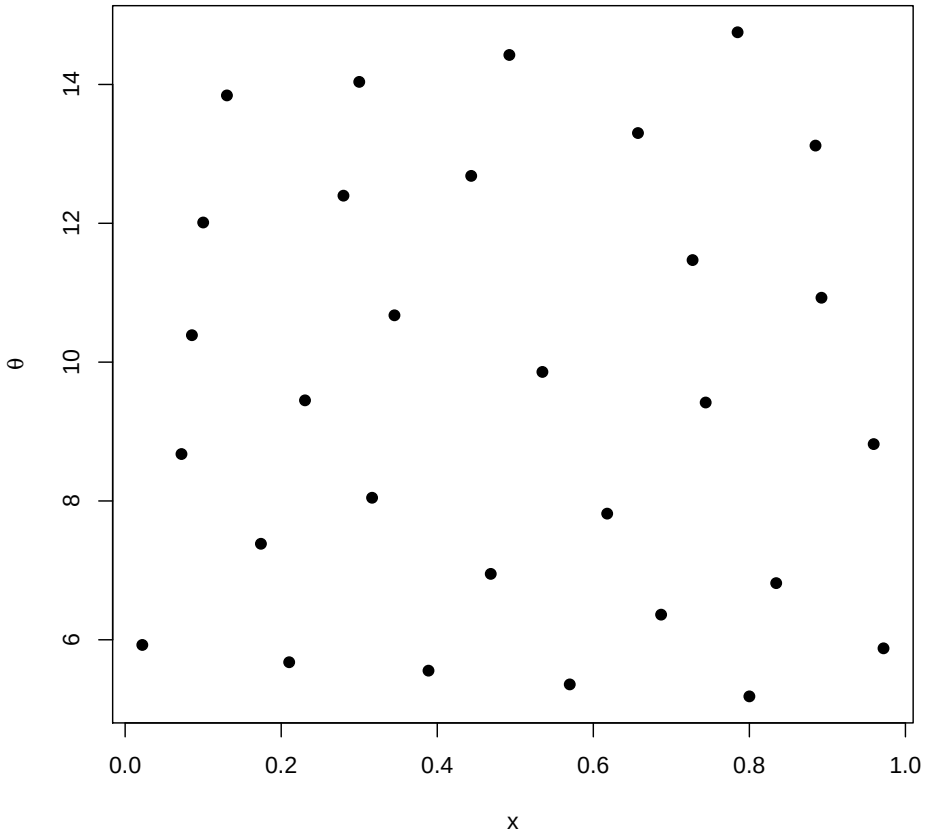
GP-based Bayesian calibration:

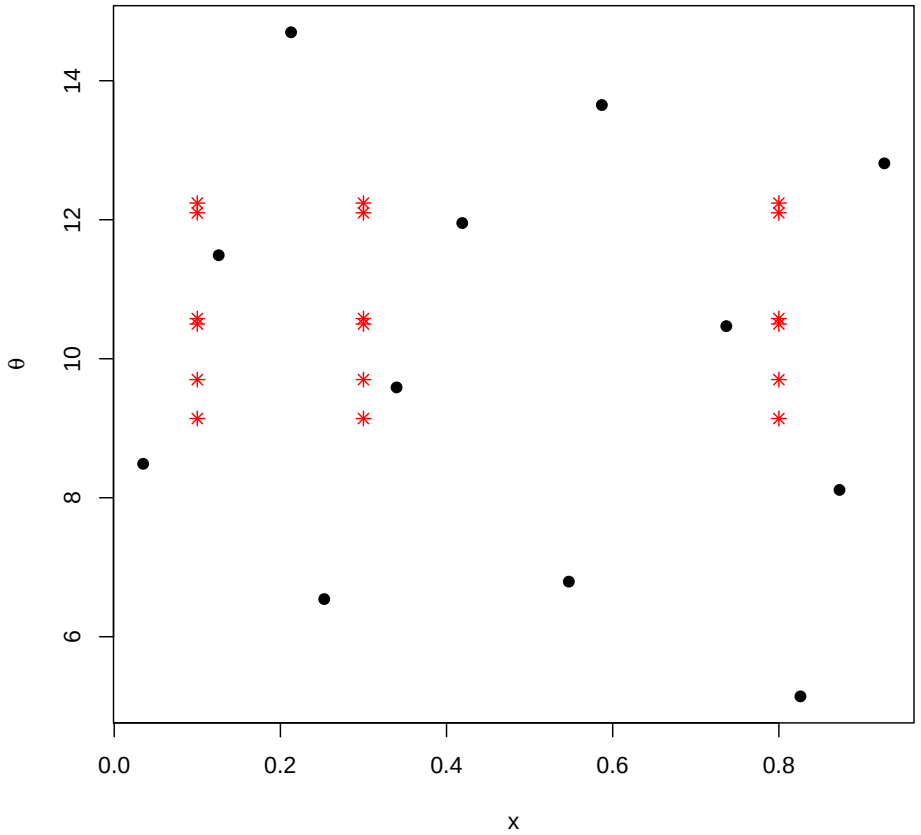
- The GP is built with a constant mean and a Matern 5/2 correlation function.

Actual posterior distribution $\pi(\theta|\mathbf{z}^f)$

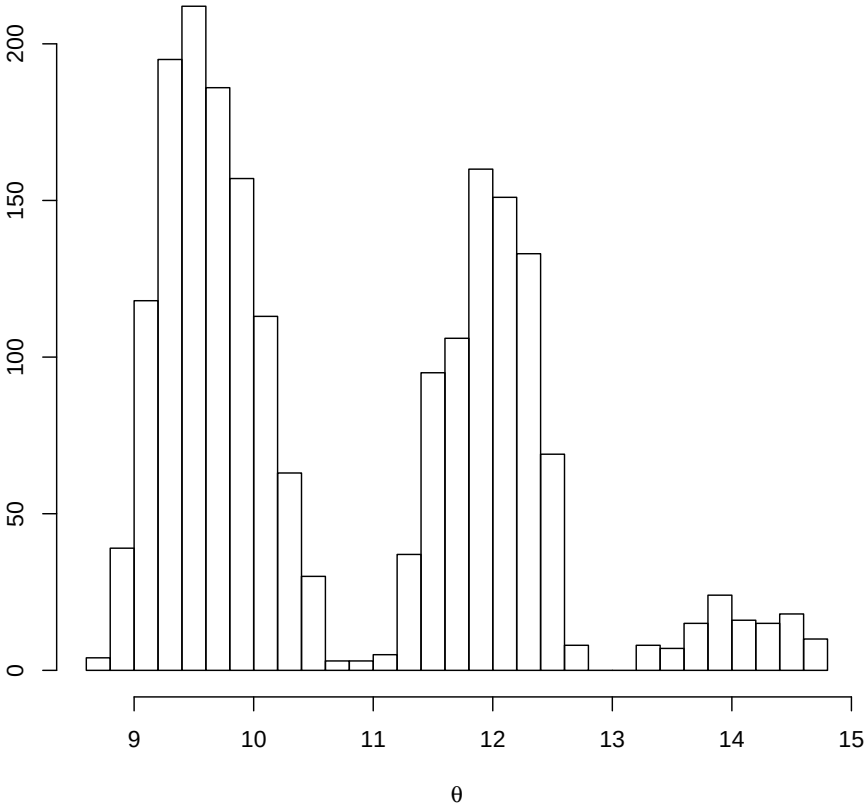


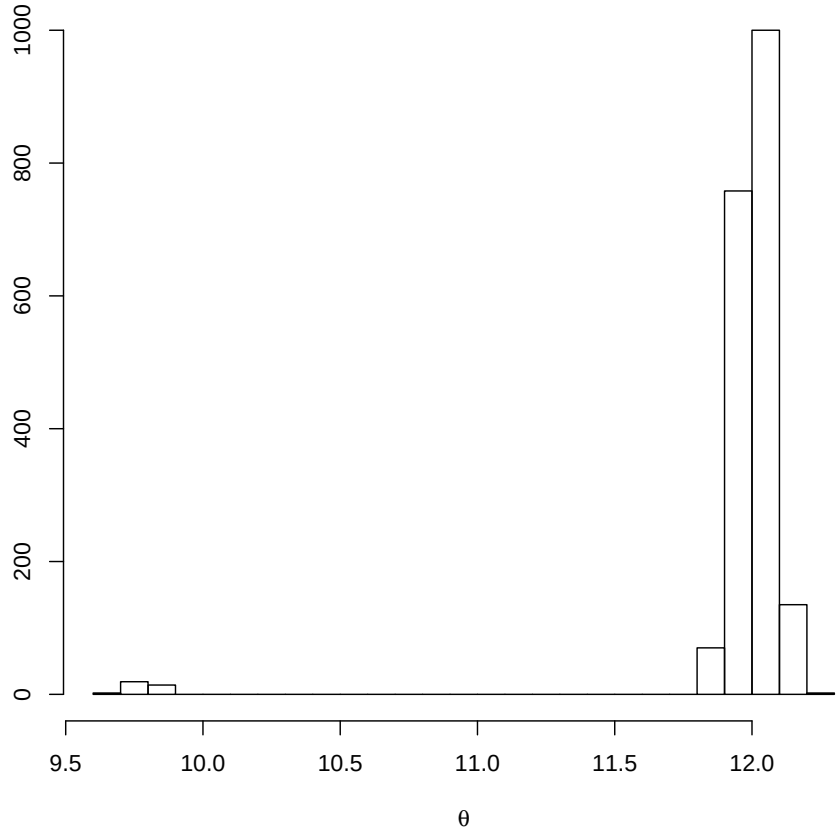
One shot maximin design vs EGO-based design: $M = 30$





Corresponding GP-based posterior distributions



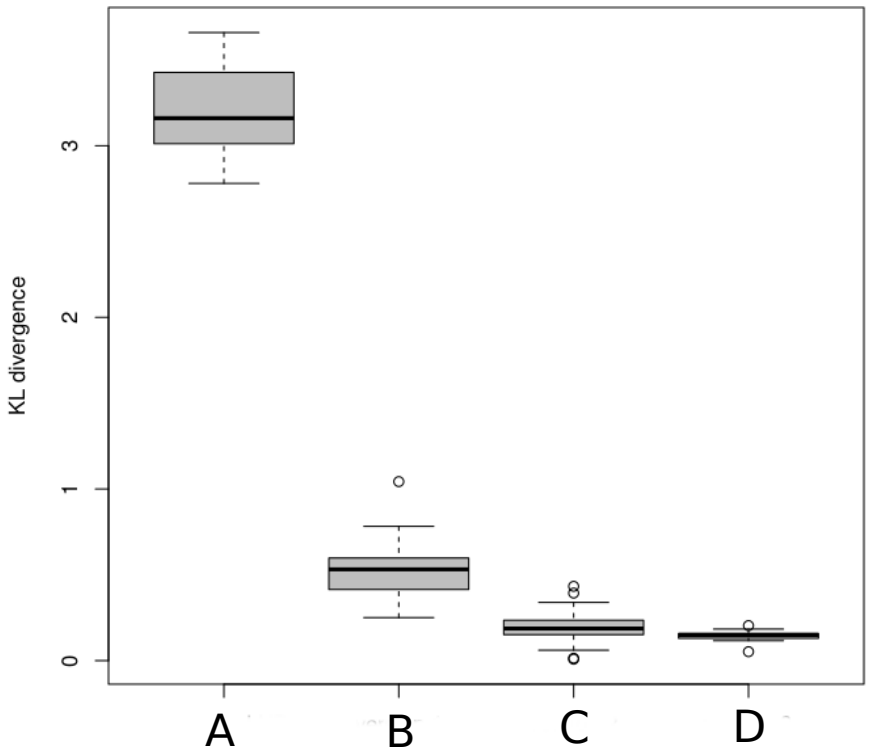


Results in terms of the KL divergence

Impact of $\mathbf{D}_M$ in terms of the KL divergence ?

- A: maximin Latin Hypercube Design (LHD),
- B: adaptive designs using EGO,
- C and D: adaptive design using one at a time EGO.

Boxplots of the KL divergence computed between $\pi^C(\boldsymbol{\theta}|\boldsymbol{z}^f, y(\mathbf{D}_M))$ *and* $\pi(\boldsymbol{\theta}|\boldsymbol{z}^f)$



Conclusions

The GP-based calibration should consist of:

1. learning sequentially the GP emulator using the EI criterion applied to $SS(.)$,
2. then, sampling the GP-based posterior distribution.

Results are much improved compared to one shot designs in terms of the KL divergence

To go further:

- Take into account the prior distribution in the writing of the EI criterion (when it is informative),
- Extend the method when an unknown discrepancy function is inserted between the code output and the physical system,
- Build $\mathbf{D}_M$ from the KL divergence instead of $SS(.)$?

DAMBLIN, G. ; BARBILLON, P. ; KELLER, M. ; PASANISI, A. ; PARENT, E., *Adaptive numerical designs for the calibration of computer codes*, http://arxiv.org/abs/1502.07252 (2015).

M.J Bayarri, J.O Berger, R. Paulo, J Sacks, J.A Cafeo, J. Cavendish, C.H Lin, and J. Tu.

A framework for validation of computer models.
Technometrics, 49(2):138–154, 2007.

T.M. Cover and J.A. Thomas.

Elements of Information Theory.
Wiley-Interscience, 1991.

D.D. Cox, J.S. Park, and E.S. Clifford.

A statistical method for tuning a computer code to a data base.
Computational Statistics and Data Analysis, 37(1):77–92, 2001.

D.R. Jones, M. Schonlau, and W.J. Welch.

Efficient global optimization of expensive black-box functions.
Journal of Global Optimization, 13(4):455–492, 1998.

M.T. Pratola, S.R. Sain, D. Bingham, M. Wiltberger, and E.J. Rigler.

Fast sequential computer model calibration of large nonstationary spatial-temporal processes.
Technometrics, 55(2):232–242, 2013.

T.J. Santner, B.J. Williams, and W.I Notz.

The Design and Analysis of Computer Experiments.
Springer-Verlag, 2003.