# The kriging approach to optimization

Rodolphe Le Riche
www.emse.fr/~leriche
CNRS and Ecole des Mines de St-Etienne

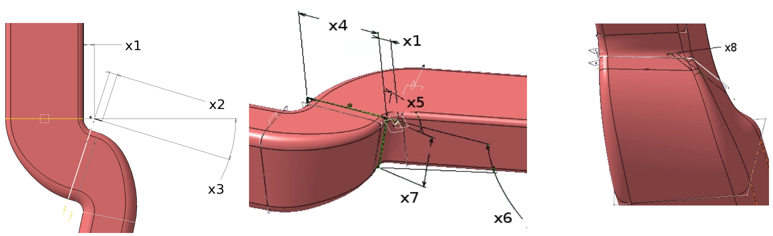
Openturns Users Day, June 2014

# optimization using engineering simulations as a dialog between a physicist / engineer and a statistician

# Optimizing from engineering simulations

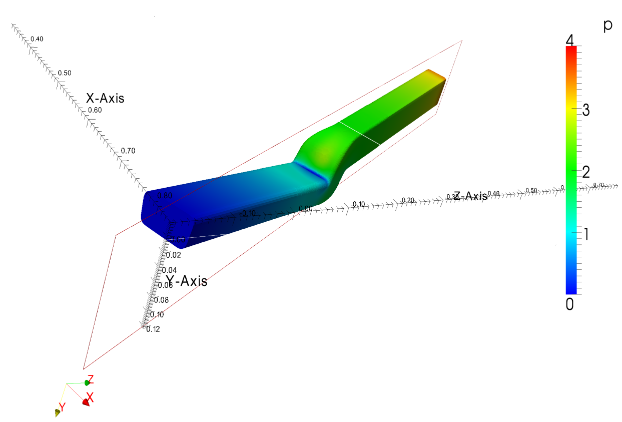
Knowledge about a physical model stored in a simulator with inputs and outputs

$x$



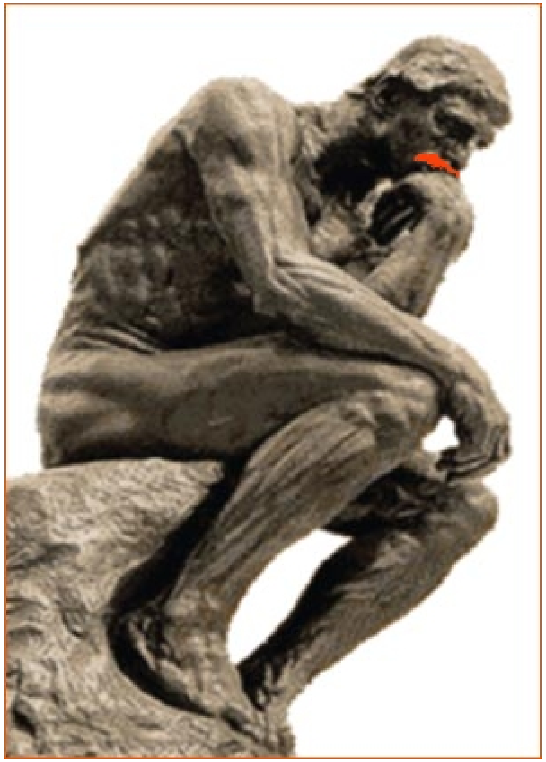

Expl : dimensions of a pipe

**Simulation**
software
(FE, FVol, BE, ... )
[sec to hours]




Expl : pressure field

# The virtual prototyping idea



*The simulation seems fairly realistic.*
*Let's use it to **decide** what is an optimal configuration.*

The physicist / engineer

# Mathematical formulation of the optimization

A decision (e.g., a design decision) is **formulated** as an optimization problem :

$$\text{Mathematical goal :} \quad \min_{x \in S \subset \mathbb{R}^n} f(x)$$

*f(.)*, the cost function (pressure drop, masse, constraint violation, distance to goal, cost, risk, ...).

Constraints, $g(x) \leq 0$ , are not explicitely discussed in this talk. As a patch, you may assume that

$$\begin{array}{l} \min_{x \in S \subset \mathbb{R}^n} f(x) \\ g(x) \leq 0 \end{array} \rightarrow \quad \min_{x \in S \subset \mathbb{R}^n} f(x) + p \times max^2(0, g(x))$$

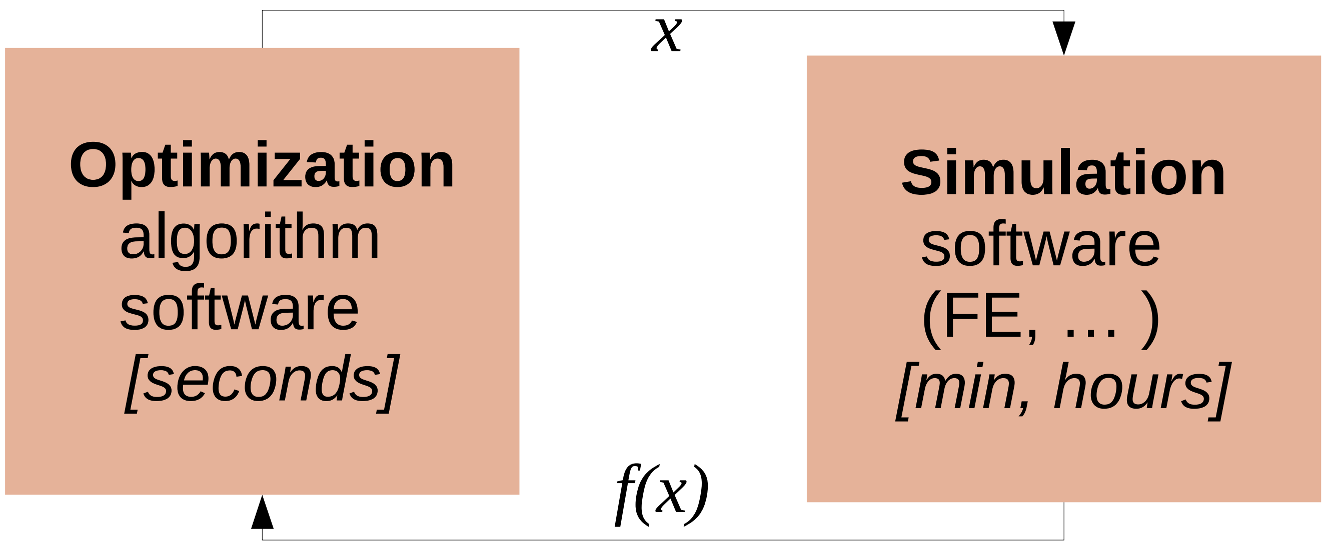
$p$ , a vector of penalty positive scalars.

Constraints satisfaction problem : A. Chaudhuri, R. Le Riche and M. Meunier, *Estimating feasibility using multiple surrogates and ROC curves*, 54th AIAA SDM Conference, Boston, USA, 8-11 April 2013.

# Automatic use of the simulator

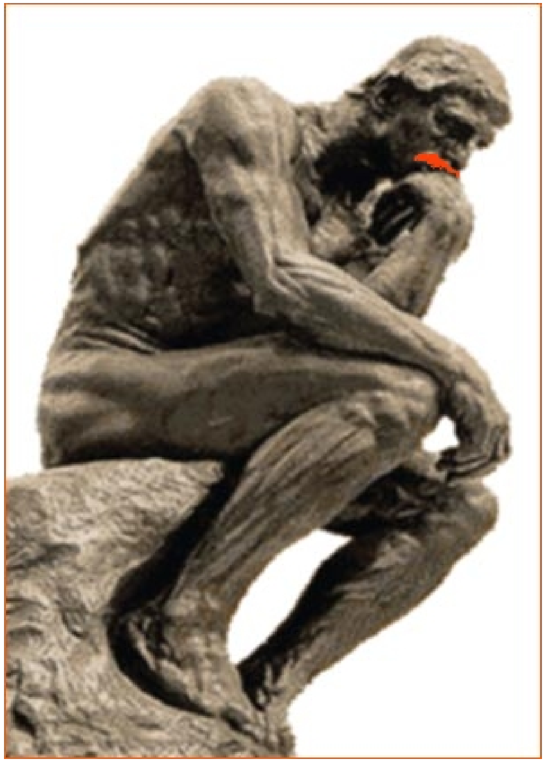
The physicist / engineer : $f(x)$ is not known analytically. Let's try $M$ points and keep the best one, $\underset{i=1,M}{arg\ min}\ f\left(x^i\right)$

An optimization program will automatically call the simulator.




Communication between programs by file, pipe, messages.

# By the way, what strategy for the optimization ?



*1 call to f takes 1 min.*
*I have 8 variables, $x_i$.*
*I will discretize each variable into 10 possible values and make a grid. That is*
*$10 \times 10 \times \ldots \times 10 = 10^8$ simulations, i.e.,*
*… 190 years of calculation !*

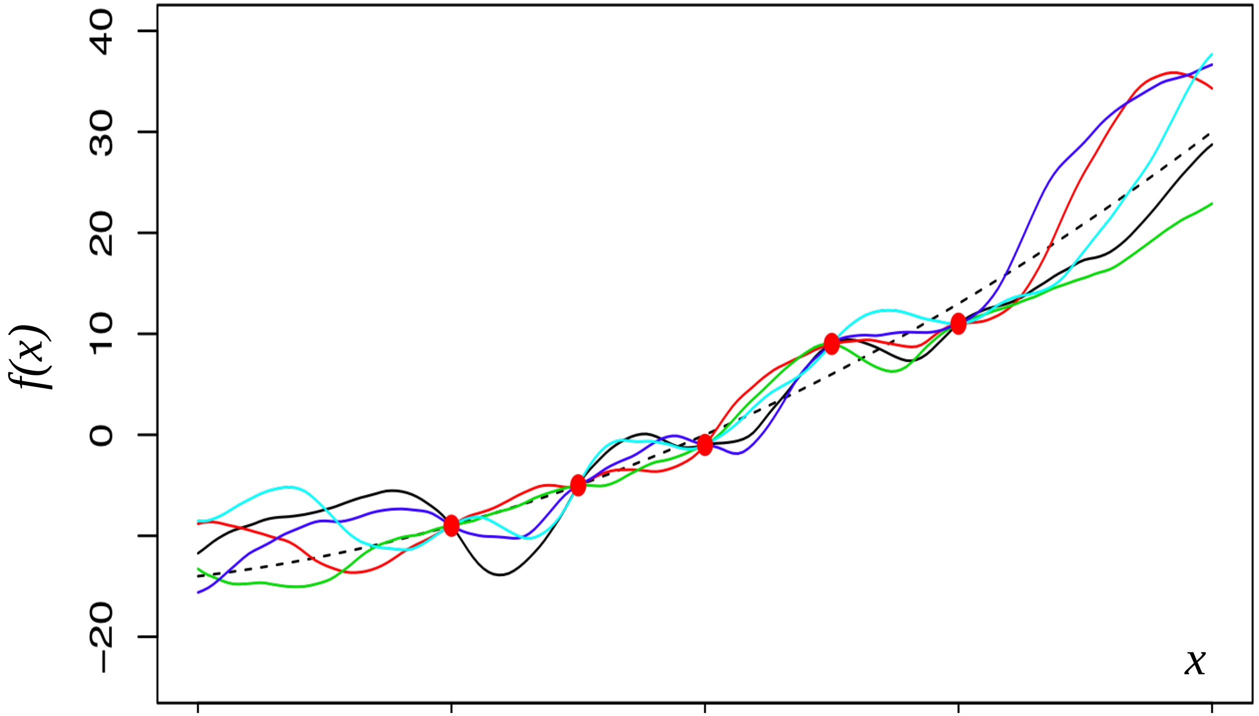
*Grids are too expensive, but I will try random points. In 95 % of the cases, I can wait 10h (600 calls to f), I will know the optimum with an accuracy on each variable better than [1] … 50 % of the total range of each variable !*

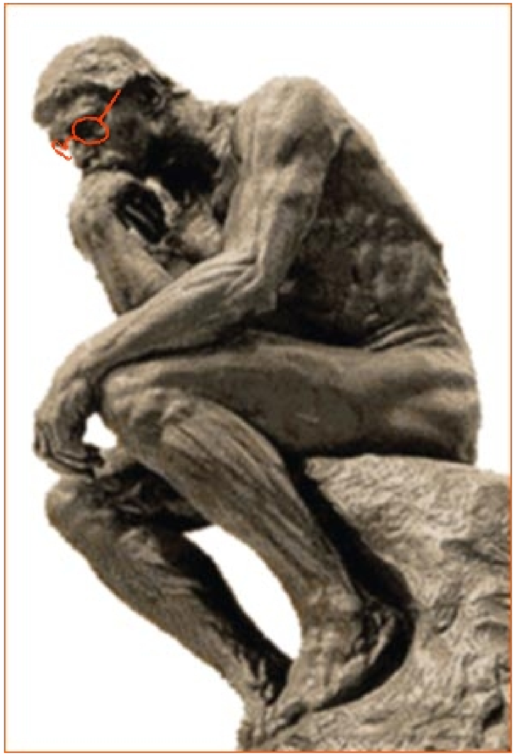
*I need a statistician.*

The simulation time is the bottleneck. Even 1 min.

[1] $\Delta$, the accuracy, and normalized variables between 0 and 1, then $1-\left(1-\Delta^n\right)^M = \text{Confidence}$

*This looks easy ! There are $M$ observations $x^i$, $f(x^i)$ . They are spatially correlated. We can use a Gaussian process indexed by $x$ and conditioned by the observations to guess values of $f$ at unexplored points $x$*



The statistician

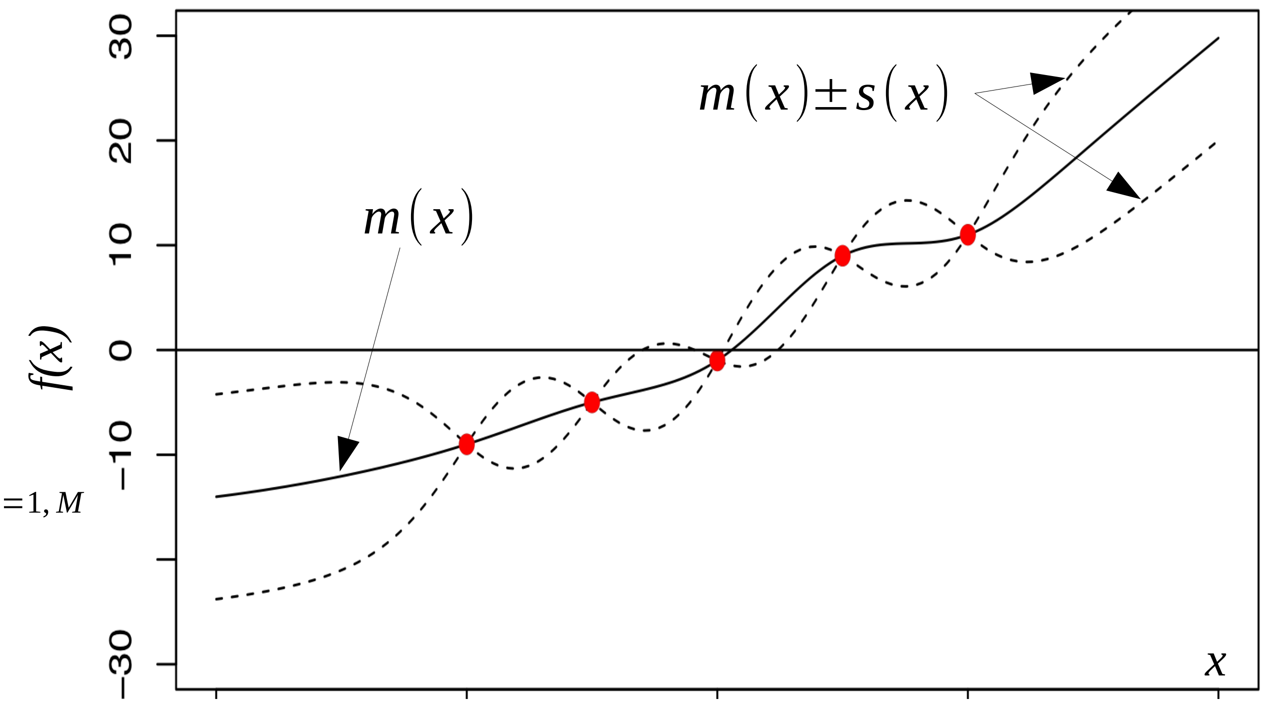
!!! only a 1D representation (complexity of dimension is lost in the drawing)
Red bullets = observations, dashed line = true function = $f(x)$, coloured lines = possible functions based on the observations.

Statistical model of $f(x)$ :

$$F(x) \sim N\left(m(x), s^2(x)\right)$$

and $F$ is correlated in space,

$$\boldsymbol{c}(x) = \left[Cov\left(F(x), F(x^i)\right)\right]_{i=1,M}$$
$$\boldsymbol{C} = \left[Cov\left(F(x^i), F(x^j)\right)\right]_{i,j}$$




Kriging average : $m(x) = \mu + \boldsymbol{c}^{\boldsymbol{T}}(x)\boldsymbol{C}^{-1}(\boldsymbol{f} - \mu\boldsymbol{1})$

Kriging variance : $s^2(x) = \sigma^2 - \boldsymbol{c}^T(x)\boldsymbol{C}^{-1}\boldsymbol{c}(x)$

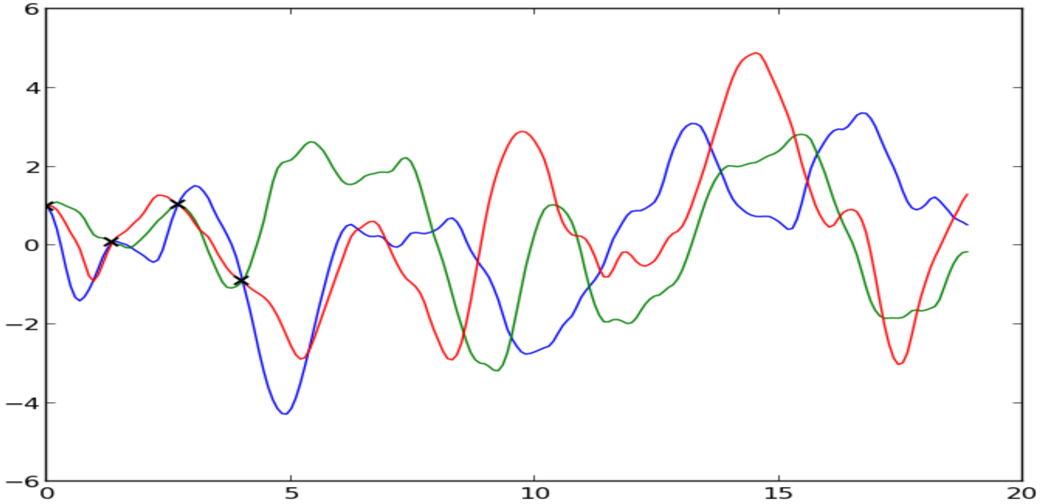
Important : choice of the kernel (stationary)

$Cov(F(x), F(x')) =$ a function of $|x - x'|$ and parameters $\theta$ (length scale)

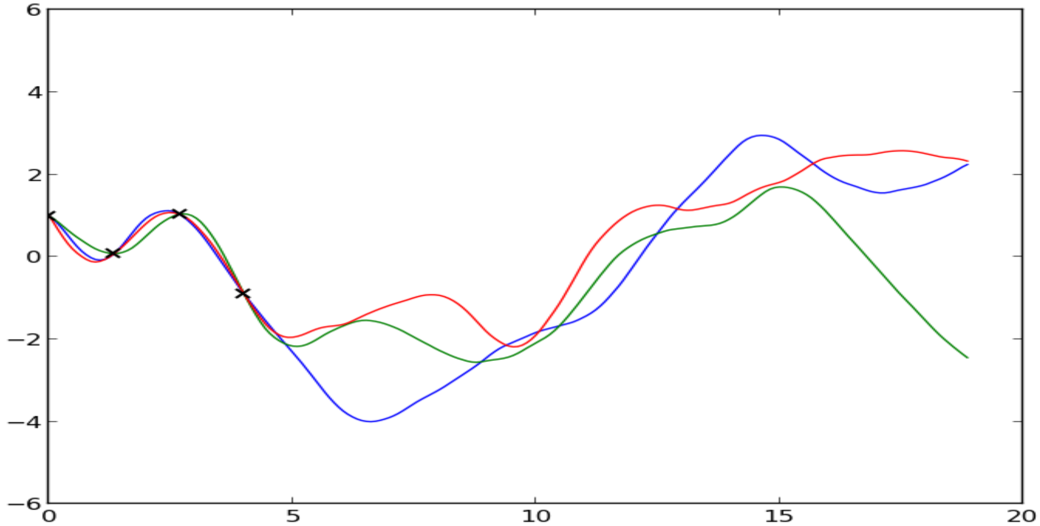
Not all functions are kernel functions.

[see Rasmussen & Williams, *GPML*, 2006 for general explanations, see Mohammadi, Le Riche, Touboul and Bay, *On regularization techniques in statistical learning by GP*, NICST'2013]

# From simulator to kernel design



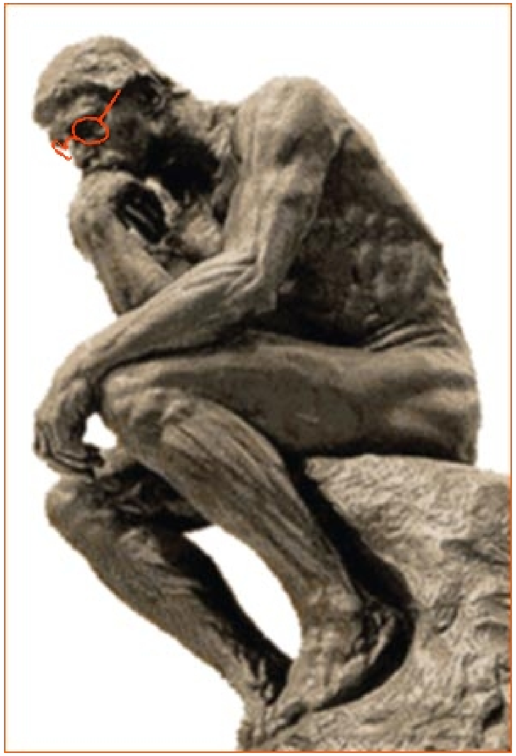

Matern 5/2 kernels, $\sigma^2$=4, **length scale = 1**



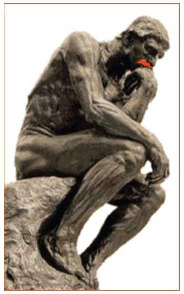

Matern 5/2 kernels, $\sigma^2$=4, **length scale = 3**

*My approach is general, yet its prediction properties are sensitive to the kernel choice... and there are so many possible kernels.*

*I need a physicist / engineer.*



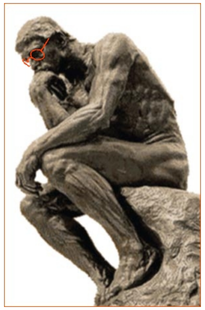
# From simulator to kernel design (cont.)

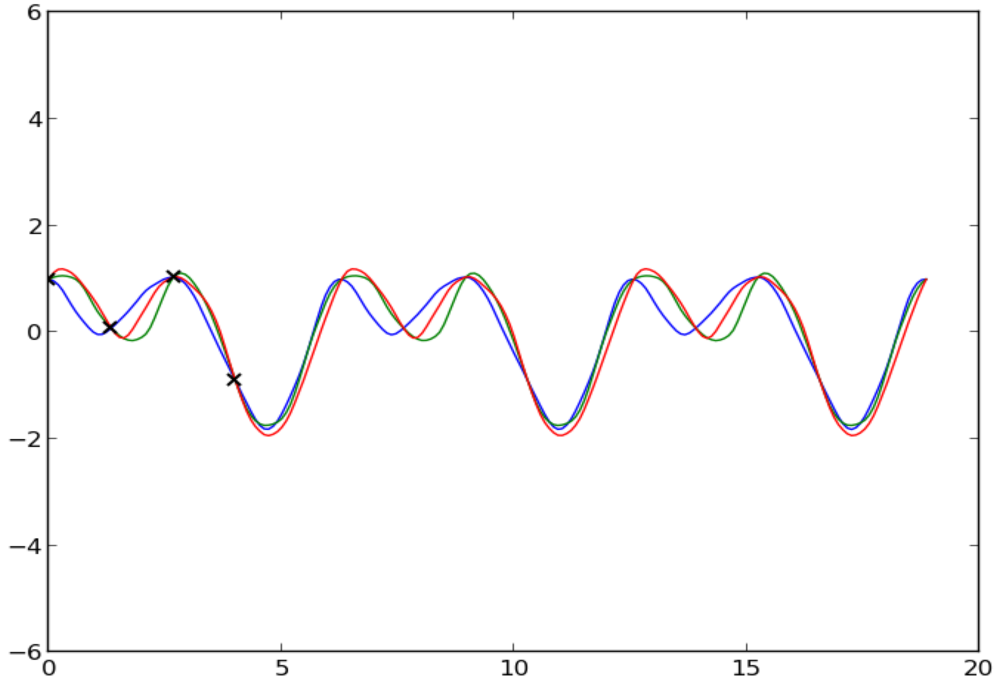


*f(x) is periodic* (example)

*then the kernel could be of the form* [2]

$$Cov(F(x), F(x')) = \sigma^2 \exp\left(\frac{-1+\cos(x-x')}{\theta^2}\right)$$






**The periodicity knowledge allows to considerably reduce statistical uncertainties.**

**Other typical expert knowledge : derivatives, symmetries, rotations, PDE's, correlated multi-fidelity simulators, previous designs, ... .**

**Kernel design is an active research domain.**

[2] N. Durrande, R. Le Riche and S. Avril, *MRI sequence denoising using Gaussian processes*, Euromech 534 colloquium on Advanced experimental approaches and inverse problems in tissue biomechanics, May 2012.
N. Durrande, J. Hensman, M. Rattray, N. D. Lawrence, *Gaussian process models for periodicity detection*, submitted to JRSSb in 2013.

# Kriging and optimization

**Search the volume** **in an efficient manner.**

Balance **EXPLORATION** with **INTENSIFICATION**

- **We will deterministically fill the design space in an efficient order.**

- **Other global search principles**

  - **Stochastic searches** : (pseudo)-randomly sample the design space $S$, use probabilities to intensify search in known high performance regions and sometimes explore unknown regions.
  - (pseudo-)**Randomly restart** local searches.
  - (and mix the above principles)

# A state-of-the-art global optimization algorithm using metamodels : EGO

(D.R. Jones et al., JOGO, 1998)

EGO = Efficient Global Optimization = use a « kriging » metamodel to define the Expected Improvement (EI) criterion. Maximize EI to creates new $x$'s to simulate.

EGO deterministically creates a series of design points that ultimately would fill $S$.

Some opensource implementations :

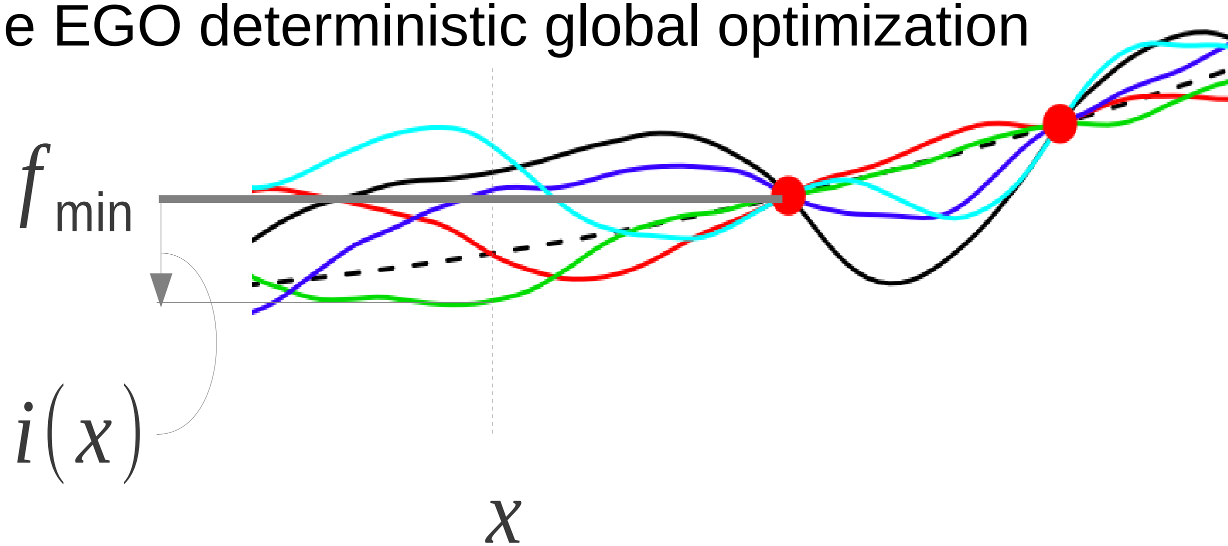
- DiceOptim in R (EMSE & Bern Univ.)
- Krisp in Scilab (Riga Techn. Univ & EMSE)
- STK: a Small (Matlab/GNU Octave) Toolbox for Kriging, (Supelec)

# (one point-) Expected improvement

A natural measure of progress : the improvement,

$$I(x) = \left[f_{\text{min}} - F(x)\right]^{+} \mid F(\boldsymbol{x}) = f(\boldsymbol{x}) \quad , \quad \text{where } [.]^{+} \equiv max(0,.)$$

- The expected improvement is known analytically.
- It is a parameter free measure of the exploration-intensification compromise.
- Its maximization defines the EGO deterministic global optimization algorithm.

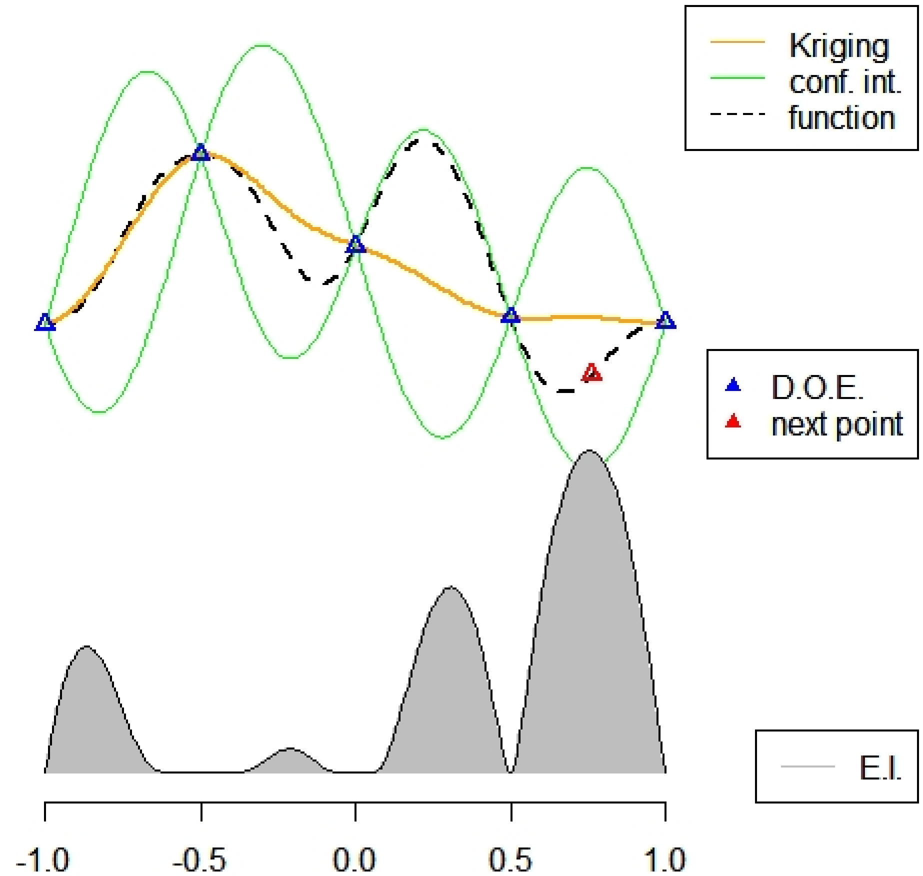



$$EI(x) = s(x) \times \left( u(x)\Phi(u(x)) + \varphi(u(x)) \right) \quad , \quad \text{where } u(x) = \frac{f_{\text{min}} - m_k(x)}{s(x)}$$

# One EGO iteration

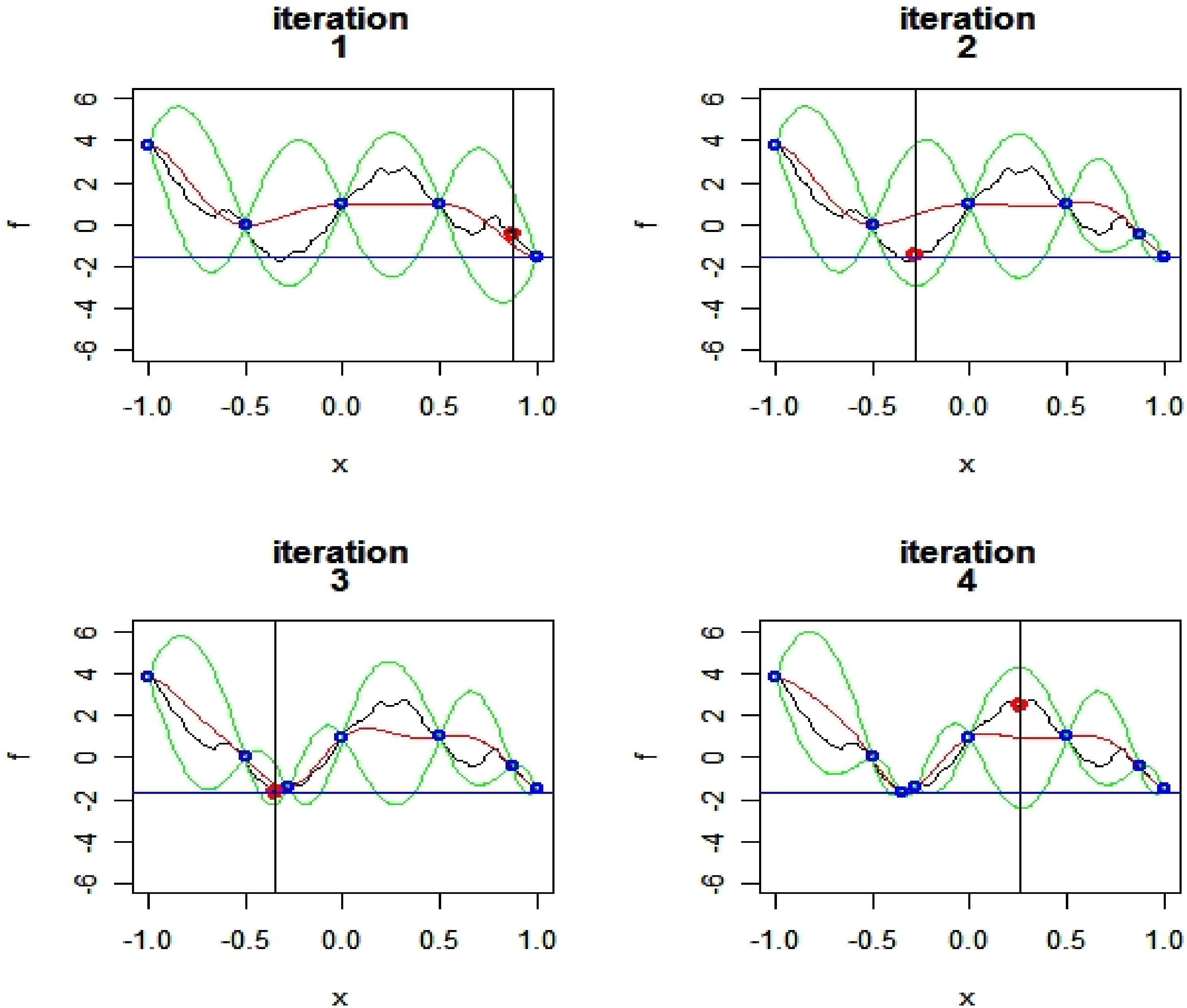
At each iteration, EGO adds to the t known points the one that maximizes EI,

$$x^{t+1} = arg\,max_x\, EI(x)$$



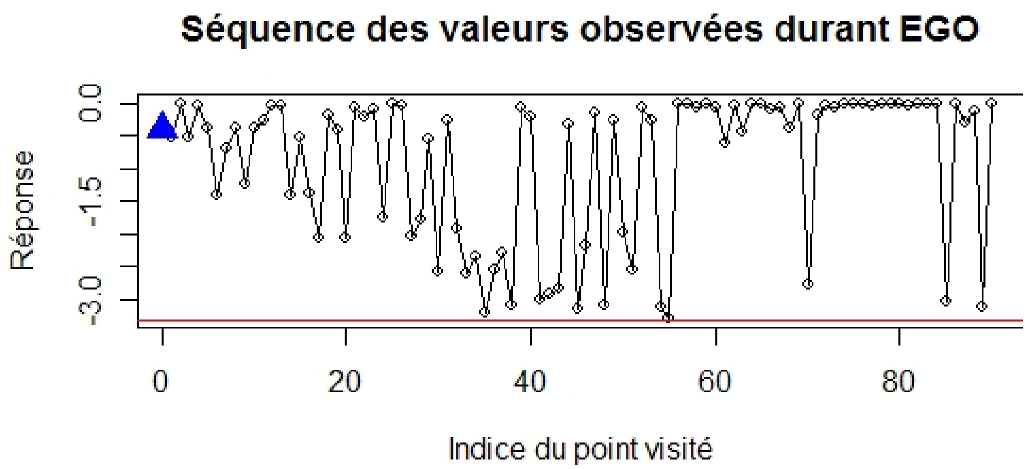

then, the kriging model is updated ...

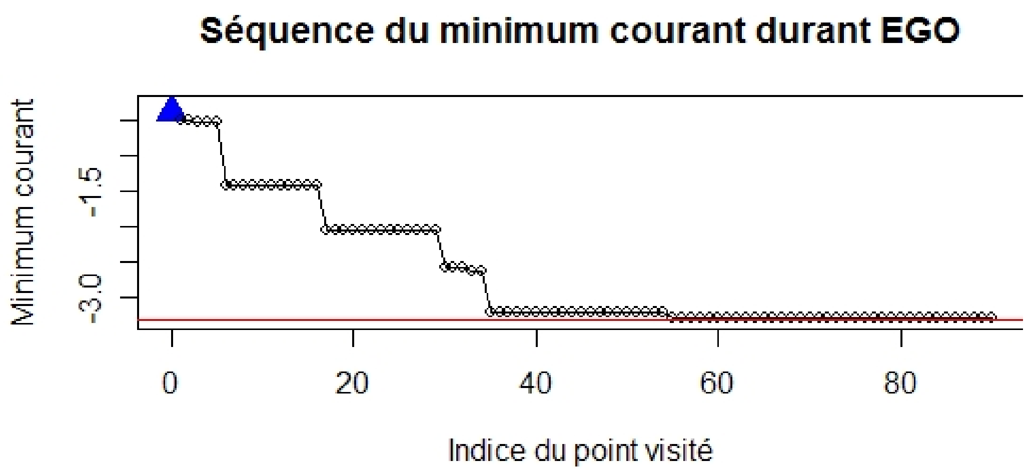
# EGO : example

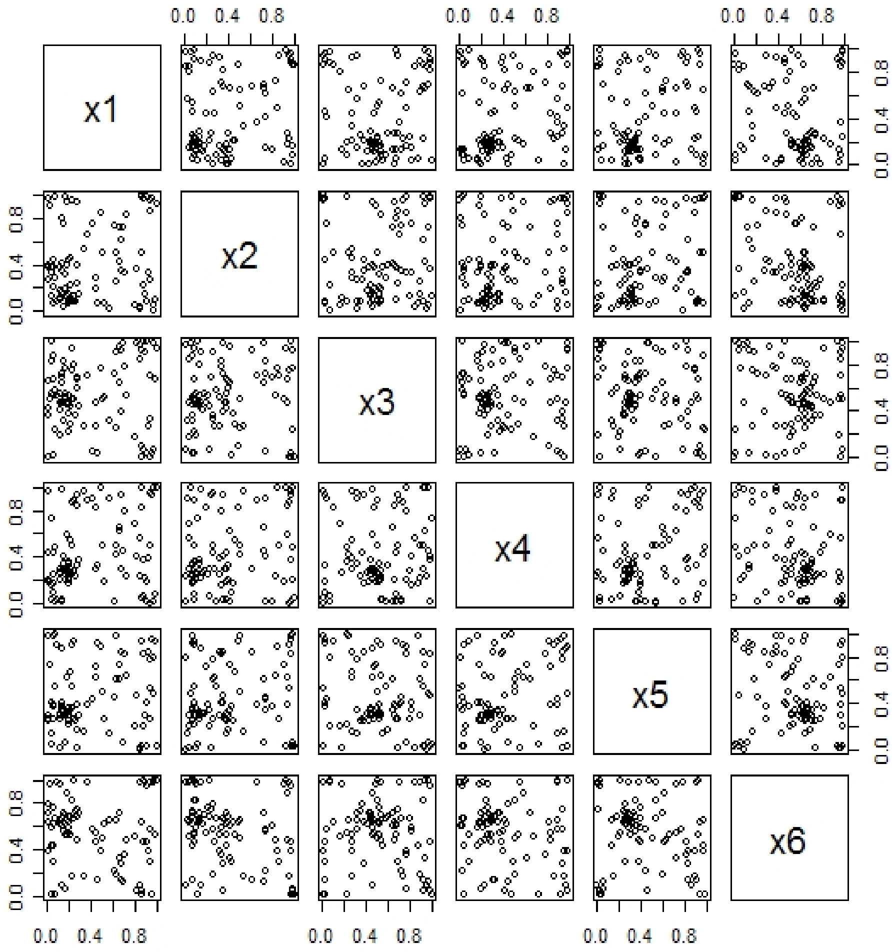

iteration 1
iteration 2
iteration 3
iteration 4
f
x
6
4
2
0
-2
-4
-6
-1.0
-0.5
0.0
0.5
1.0

# EGO : exemple en 6D

Fonction de Hartman, *f(x*)=-3.32* , 10 points dans le plan d'expérience initial.



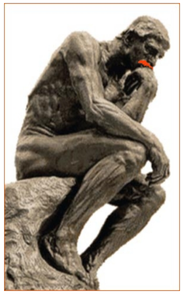



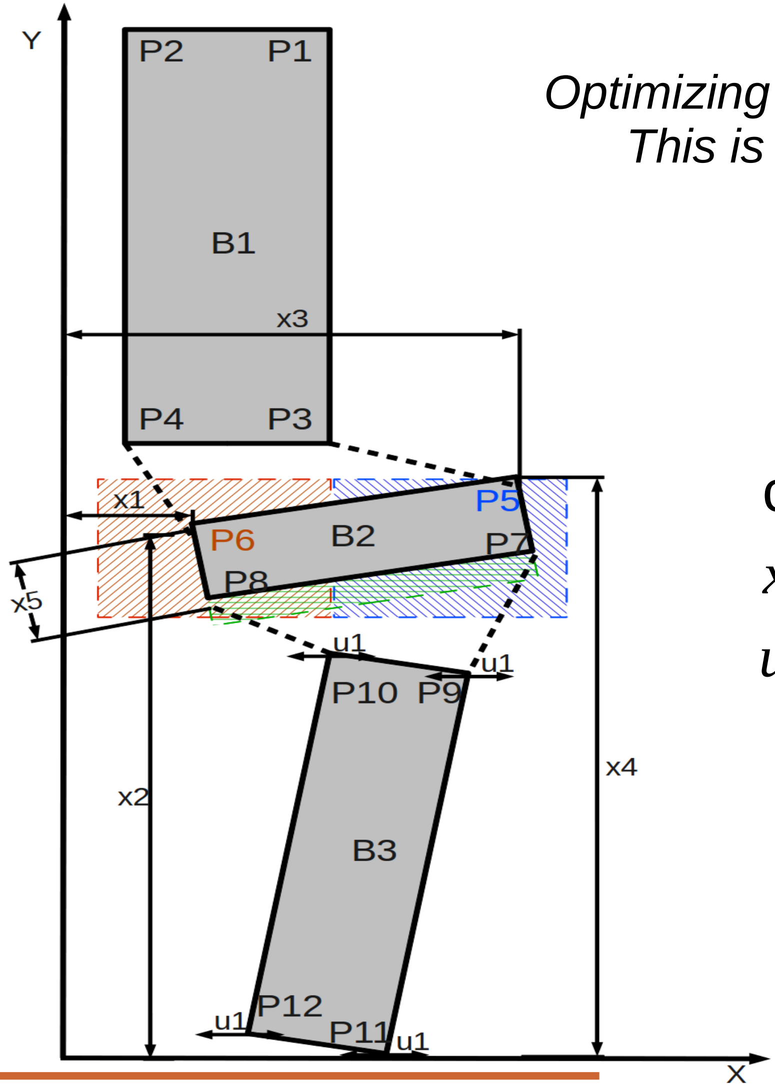

(DiceOptim, D. Ginsbourger, 2009)

# Accounting for uncertainties in the optimization

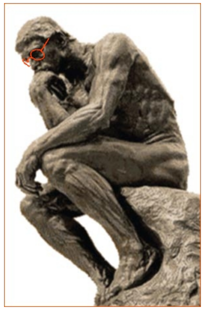
# Duct design with uncertain boundary conditions



*There is this tricky situation I keep running into.*
*I am designing a structure, and the boundary conditions are not well controlled …*




*Optimizing with uncertainties.*
*This is a difficult problem.*
*Thanks for asking.*

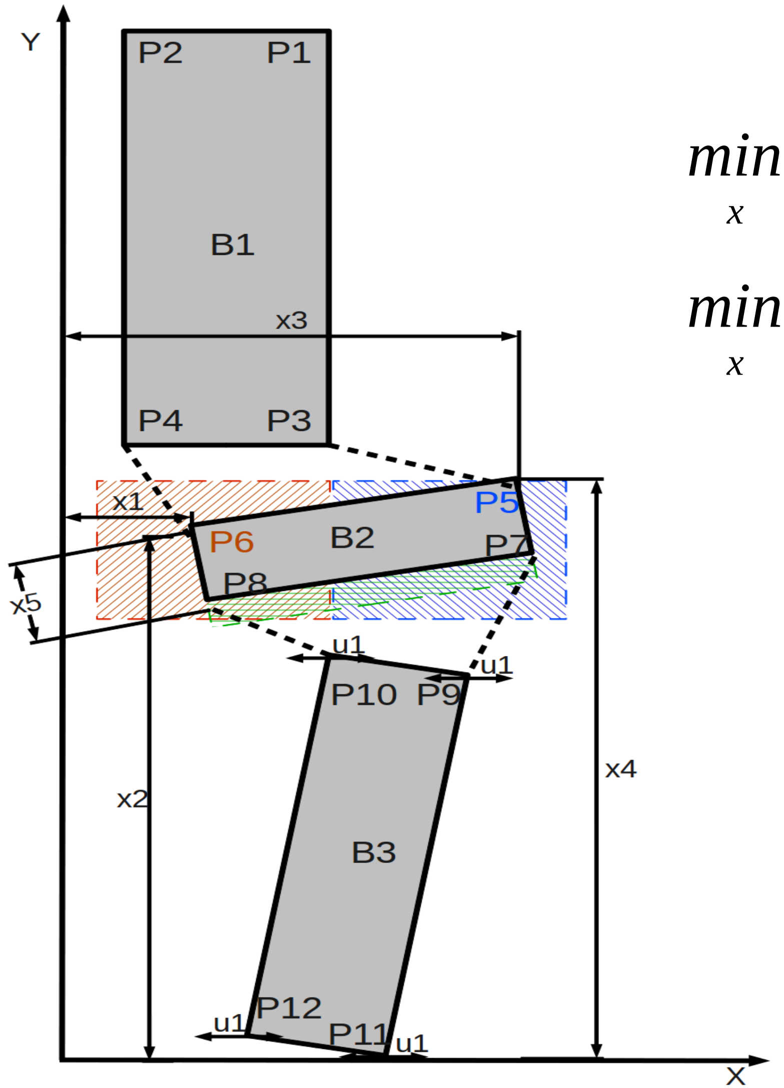


conditioner duct design

$x_1$ ,…, $x_5$ : designs variables

$u_1$ : random noise (Gaussian)
(manufacturing tolerance)

## Duct design with uncertain boundary conditions
## Problem formulation




$$\min_x E_U(\text{normal flow Std Dev @ P9P10})$$

$$\min_x E_U(\text{pressure loss P1P2-P11P12})$$

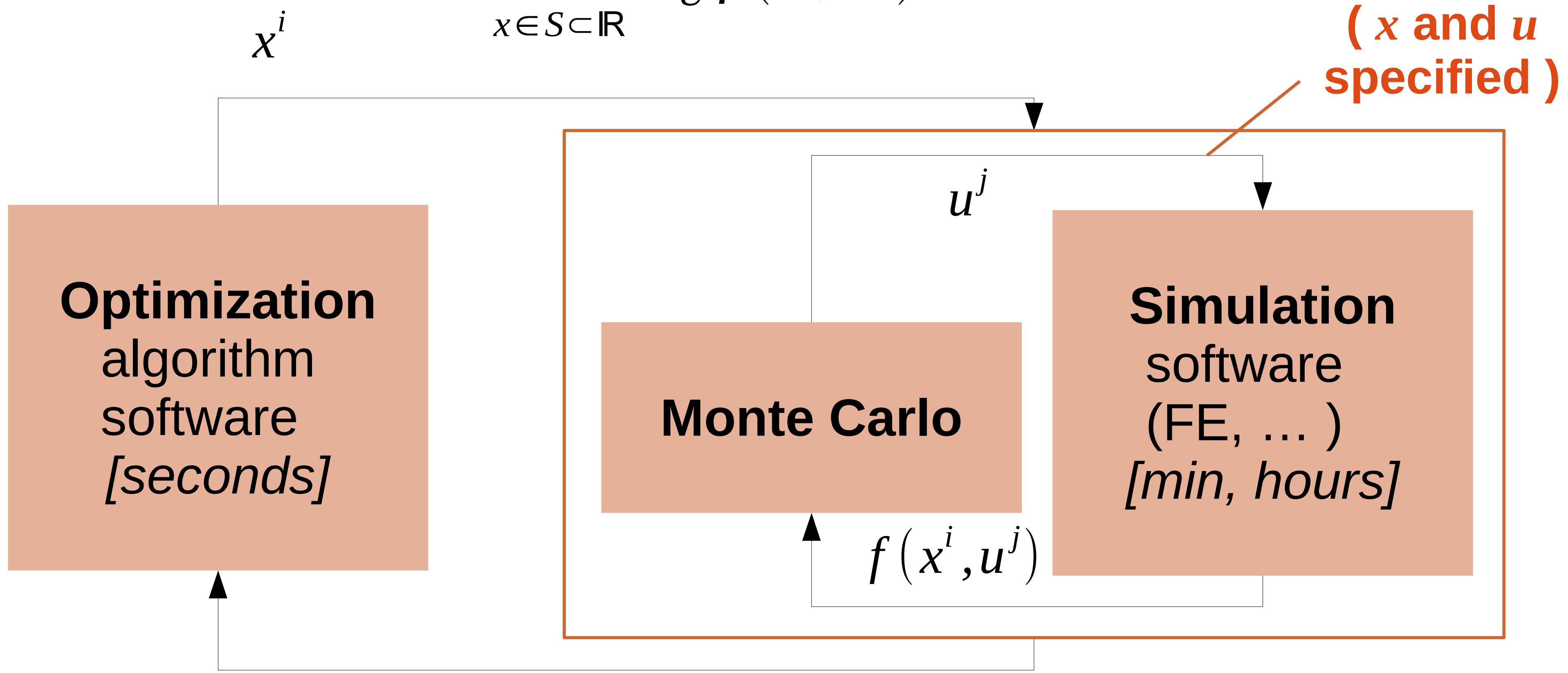
$$\min_x E_U(f(x,U))$$

2 difficult tasks put together : optimization * uncertainty propagation (e.g., Monte Carlo). Inherently 2 imbricated loops.

Cf. J. Janusevskis and R. Le Riche, *Robust optimization of a 2D air conditioning duct using kriging*, technical report hal-00566285, feb. 2011.

# Example of naive optimization with uncertainties

$$\min_{x \in S \subset \mathbb{R}} E_U f(x, U)$$

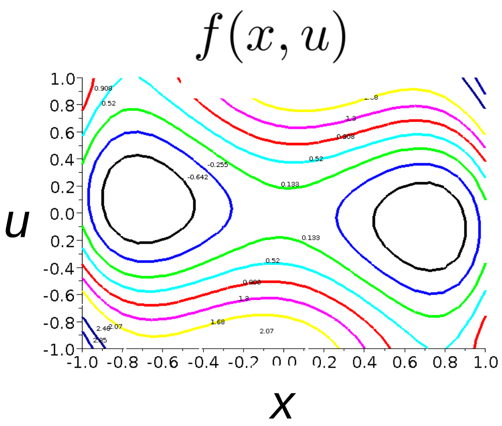



$$E_U f(x^i, U) \approx \frac{1}{N_{MC}} \sum_{j=1}^{N_{MC}} f(x^i, u^j)$$

Drawbacks : the cost of a simulation is multiplied by $N_{MC}$ and the estimation is noisy.

# Kriging based optimization with uncertainties
# Integrated kriging

**Objective :** $$\min_x \mathbb{E}_U[f(x,U)]$$




Principle : work in the joint (x,u) space.

Cf. J. Janusevskis and R. Le Riche, *Simultaneous kriging-based estimation and optimization of mean response*, Journal of Global Optimization, Springer, 2012

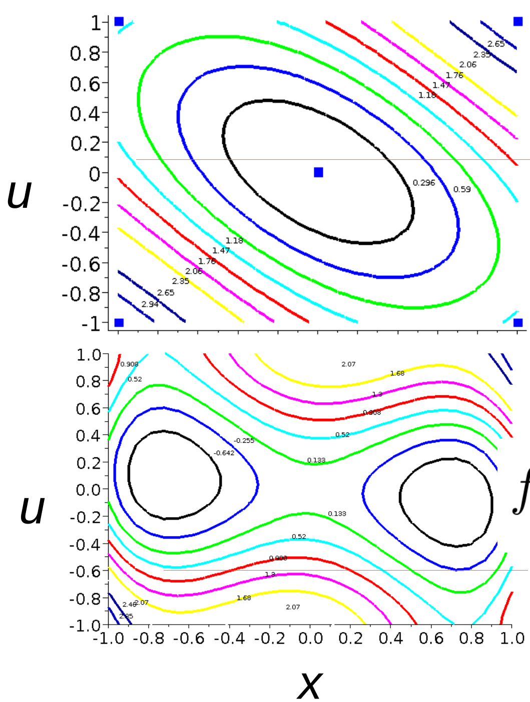
# Kriging based optimization with uncertainties
## Integrated kriging

$$\min_x \mathbb{E}_U[f(x,U)]$$ **: objective**

$Y^t_{(x,u)}(\omega)$ **: kriging approximation to deterministic** $f(x,u)$

$Z^t_{(x)}(\omega) = \mathbb{E}_U[Y^t_{(x,U)}(\omega)]$ **: integrated process approximation to** $\mathbb{E}_U[f(x,U)]$

**integrate**

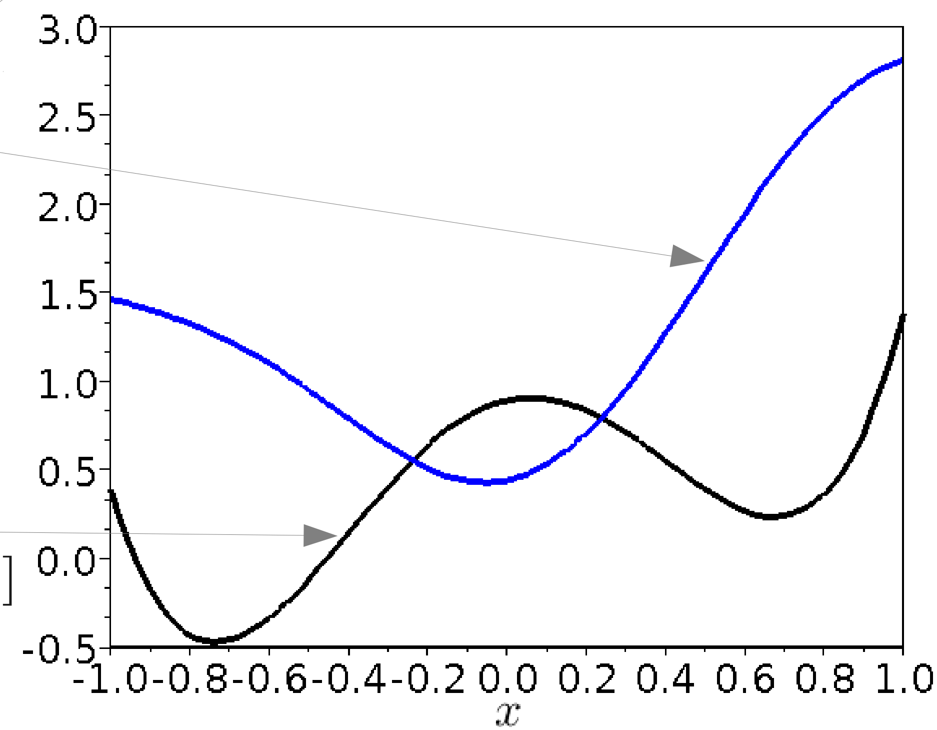



$\mathbb{E}\left(Y^t_{(x,u)}(\omega)\right)$

$f(x,u)$

**approximation** $E_\Omega[Z_{(x)}(\omega)]$

**objective** $E_U[f(x,U)]$



*Z* is a process approximating the objective function $\mathbb{E}_U[f(x,U)]$

Optimize with an Expected Improvement criterion,

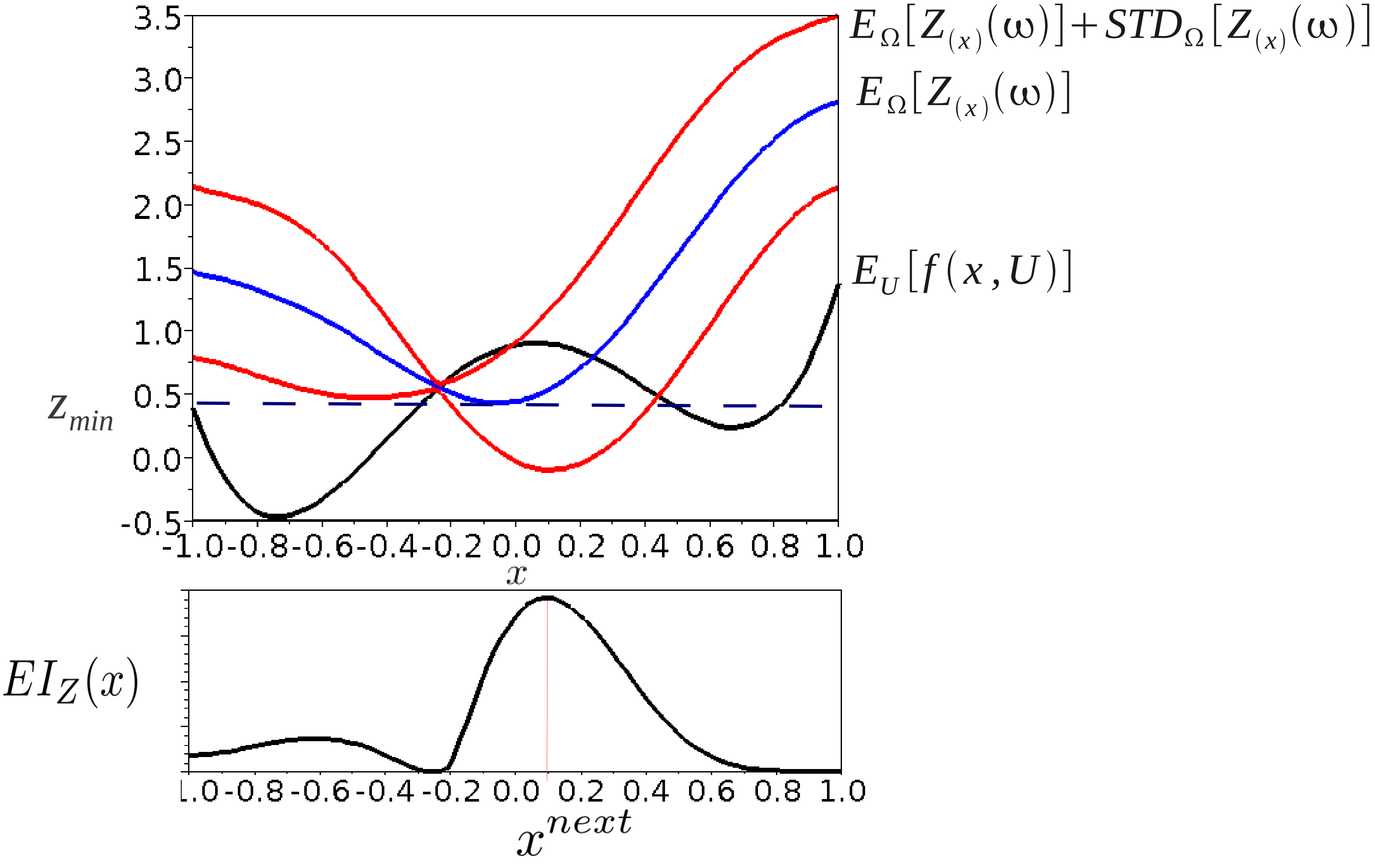
$$\boxed{x^{next} = \arg\max_x EI_Z(x)}$$

Optimize with an Expected Improvement criterion,

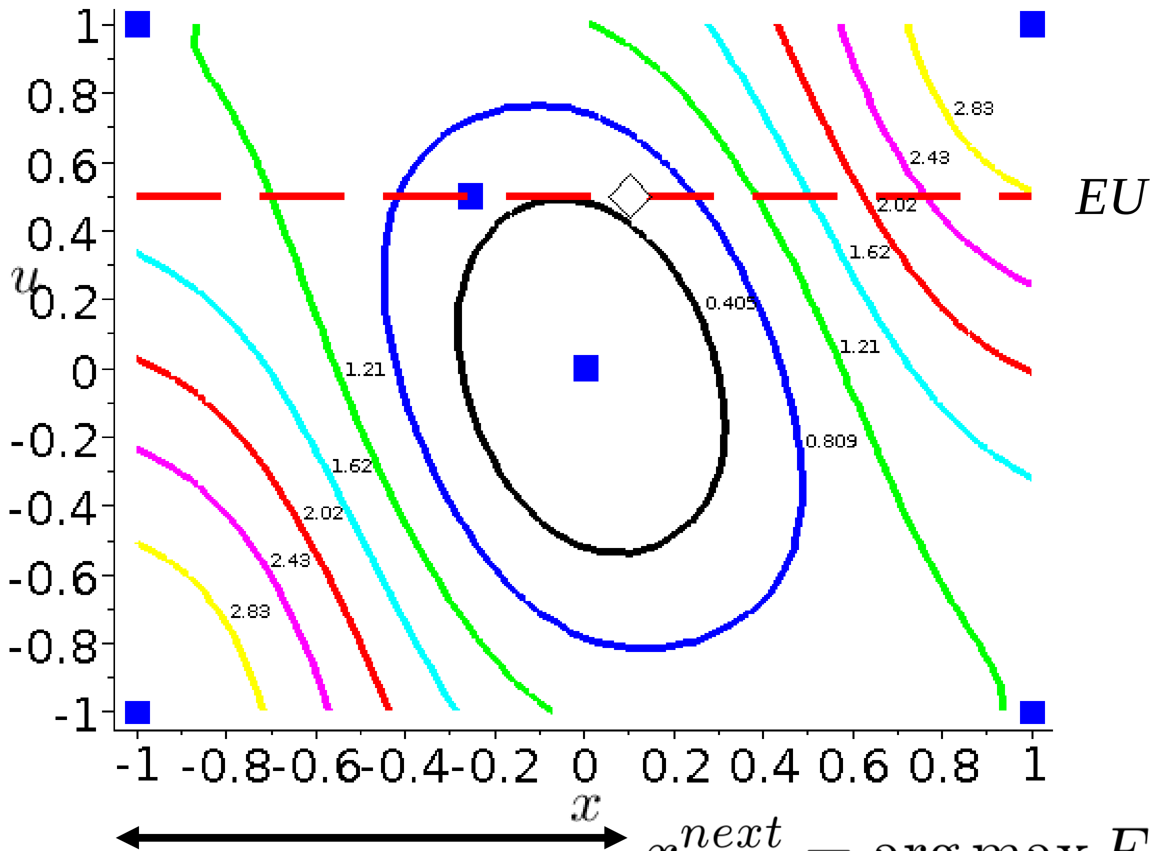
$I_Z(x)=max(z_{min}-Z(x),0)$ , but $z_{min}$ not observed (in integrated space).

$\Rightarrow$ Define $z_{min} = \min_{x^1,\ldots,x^t} E(Z(x))$

## Kriging based optimization with uncertainties
## Integrated kriging



# Kriging based optimization with uncertainties
# Integrated kriging




$$x^{next} = \arg\max_x EI_Z^t(x)$$

*x* ok. What about *u* ? (which we need to call the simulator)

# Kriging based optimization with uncertainties
# Integrated kriging

$x^{next}$ gives a region of interest from an optimization of the expected $f$ point of view.

One simulation will be run to improve our knowledge of this region of interest → one choice of $(x,u)$.

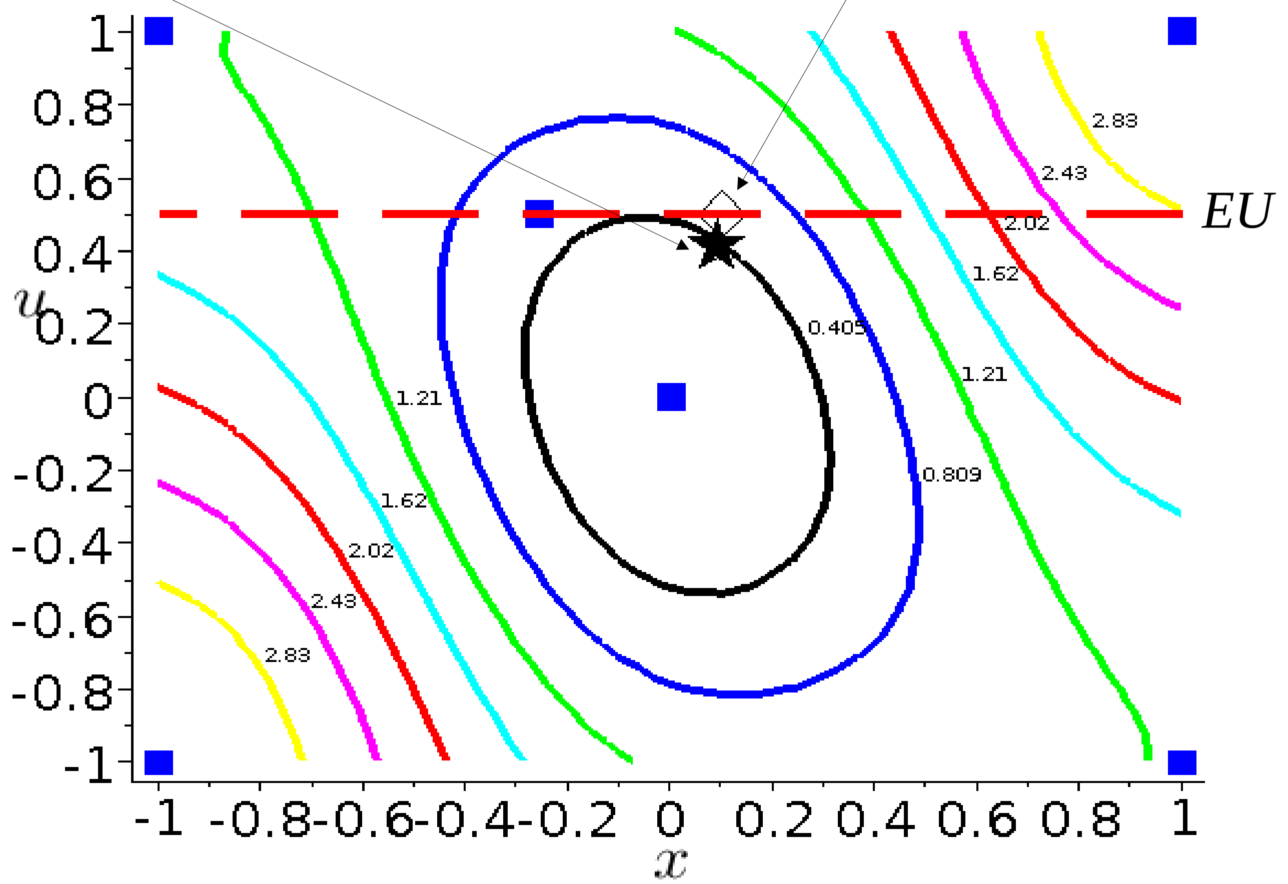
Choose $(x^{t+1},u^{t+1})$ that provides the most information, i.e., which minimizes the variance of the integrated process at $x^{next}$
(possible because the variance does not depend on $f$ evaluations, only on the points positions)

$$\boxed{(x^{t+1}, u^{t+1}) = \arg\min_{x,u} \mathbb{VAR}\ Z^{t+1}_{(x^{next})}(\omega)}$$

## Kriging based optimization with uncertainties
## Integrated kriging

$$x^{next} = \arg\max_x EI_Z^t(x)$$

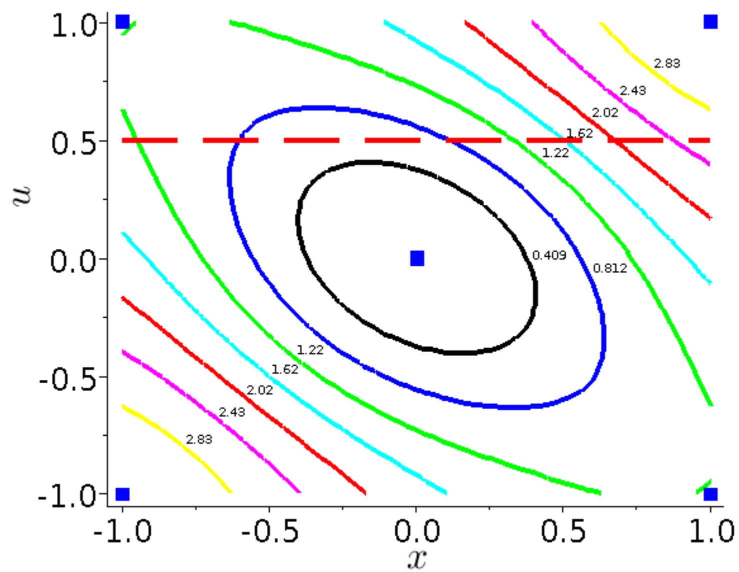
$$(x^{t+1}, u^{t+1}) = \arg\min_{x,u} \mathbb{VAR}\ Z^{t+1}_{(x^{next})}(\omega)$$

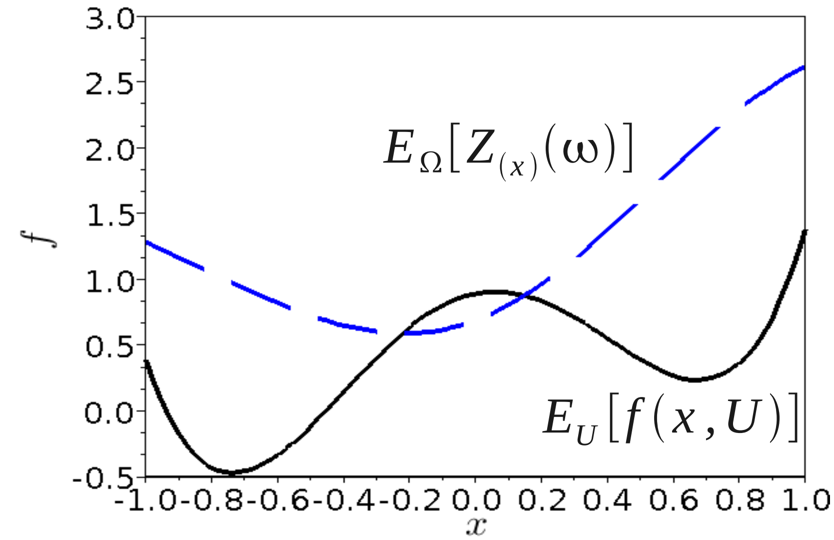


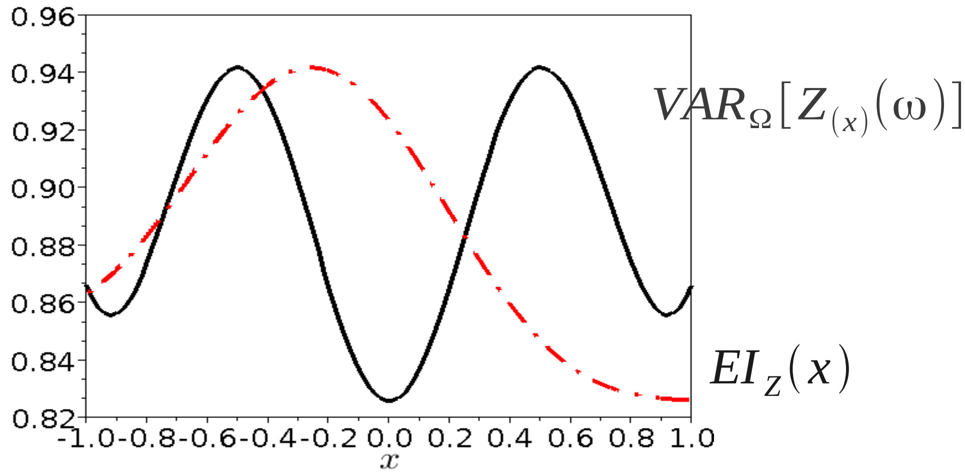
## Kriging based optimization with uncertainties, U controlled

# 2D Expl, simultaneous optimization and sampling

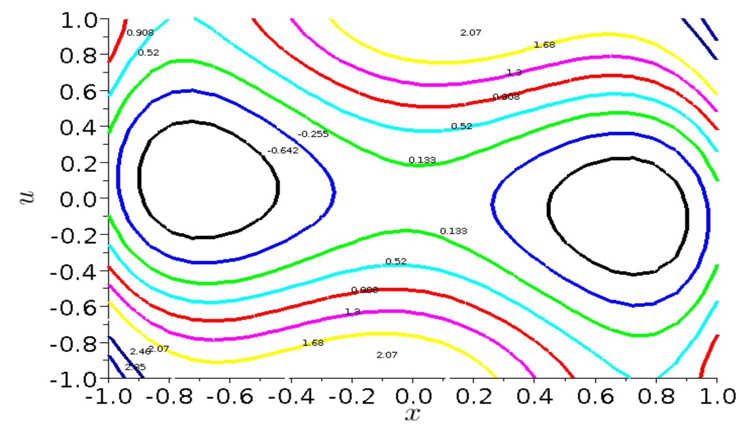
DOE and $E[Y(x,u)]$



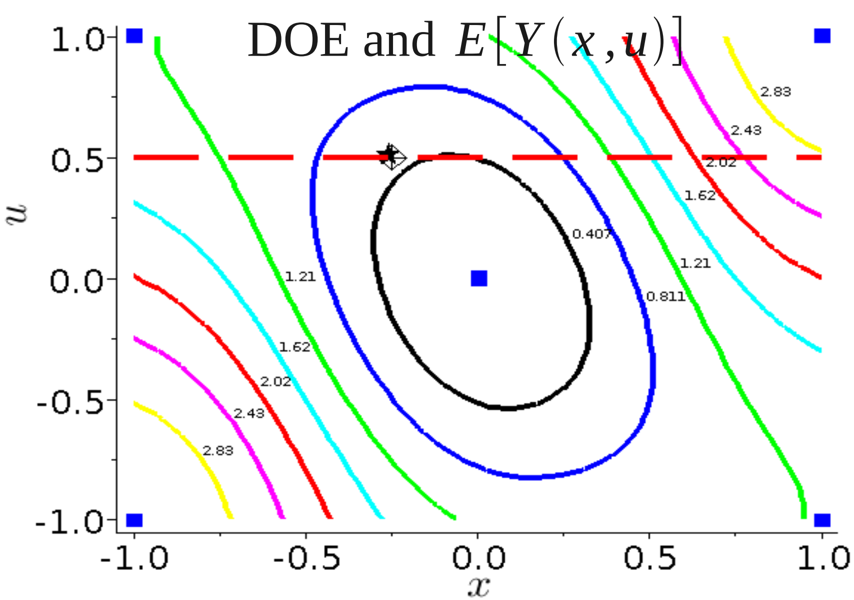



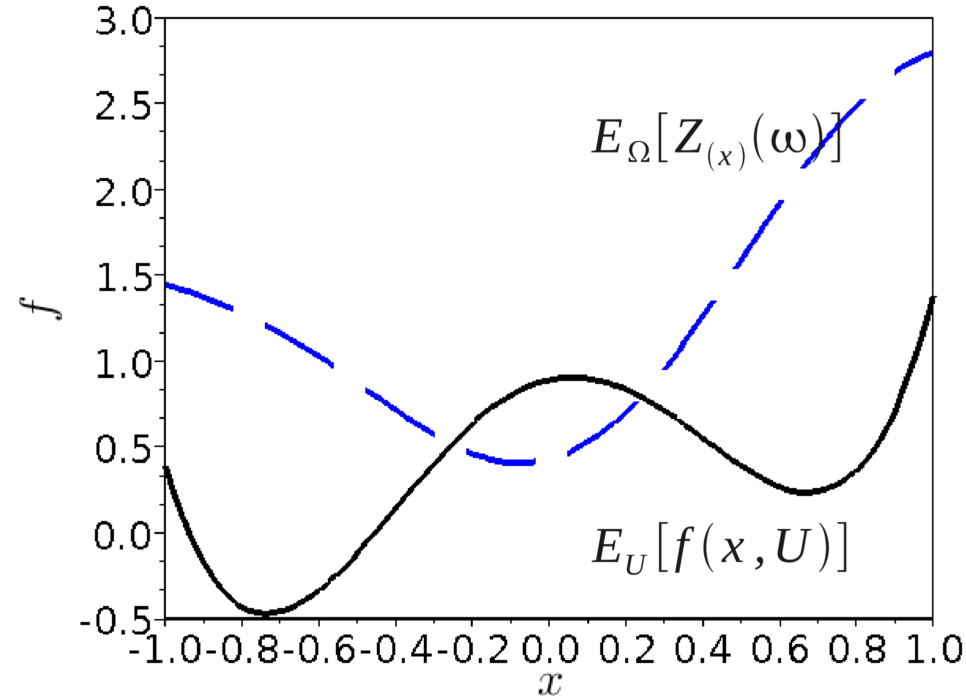

test function

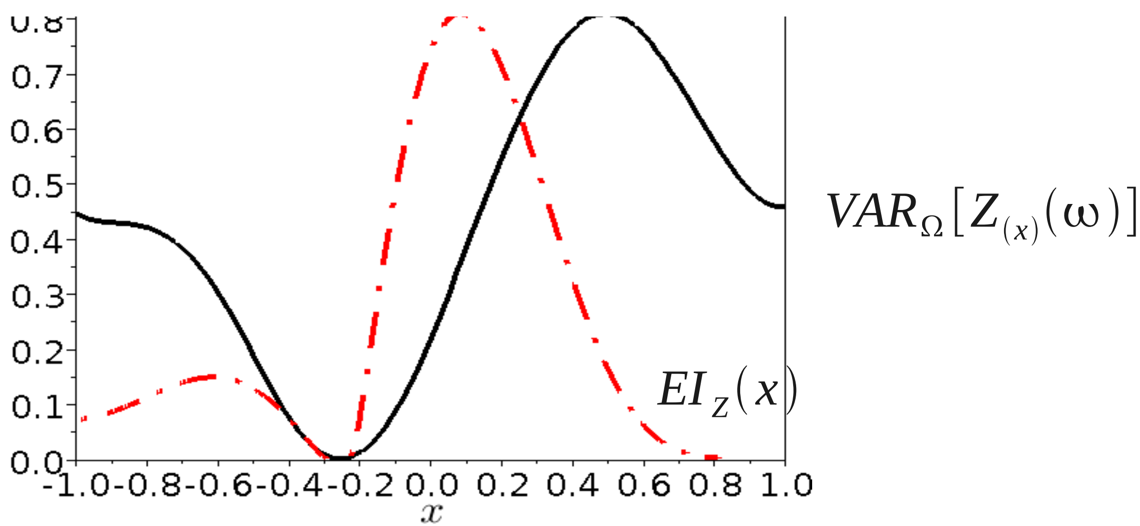


# Kriging based optimization with uncertainties, U controlled

## 1st iteration



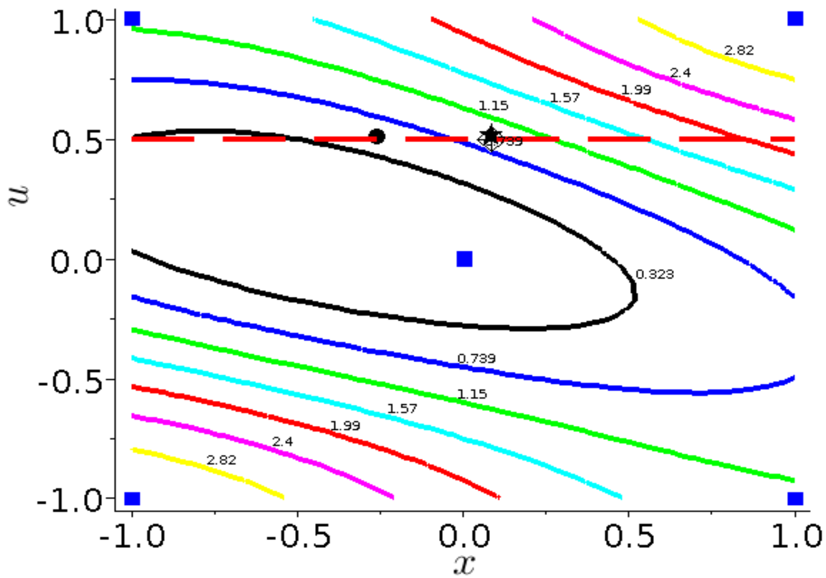



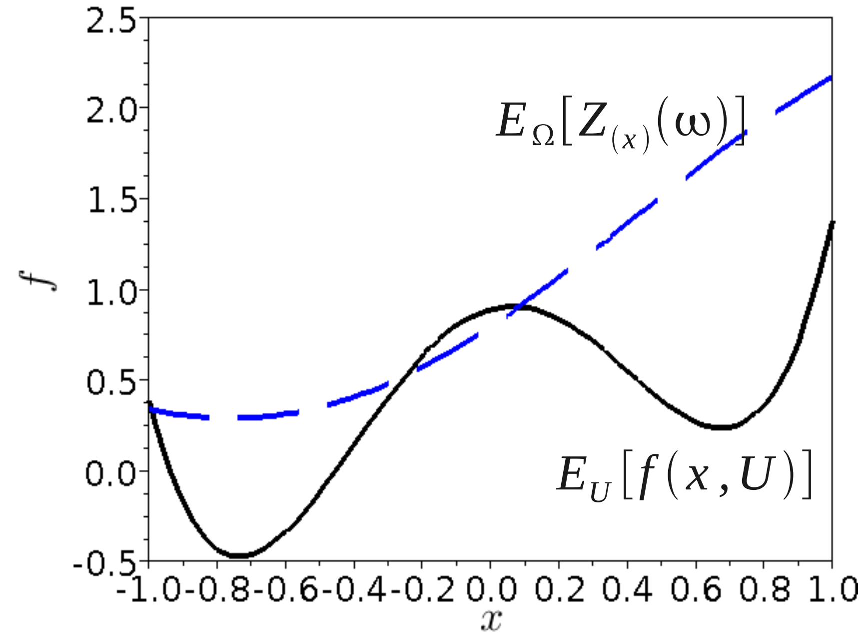

◇ $-\ (x^{next}, \mu)$

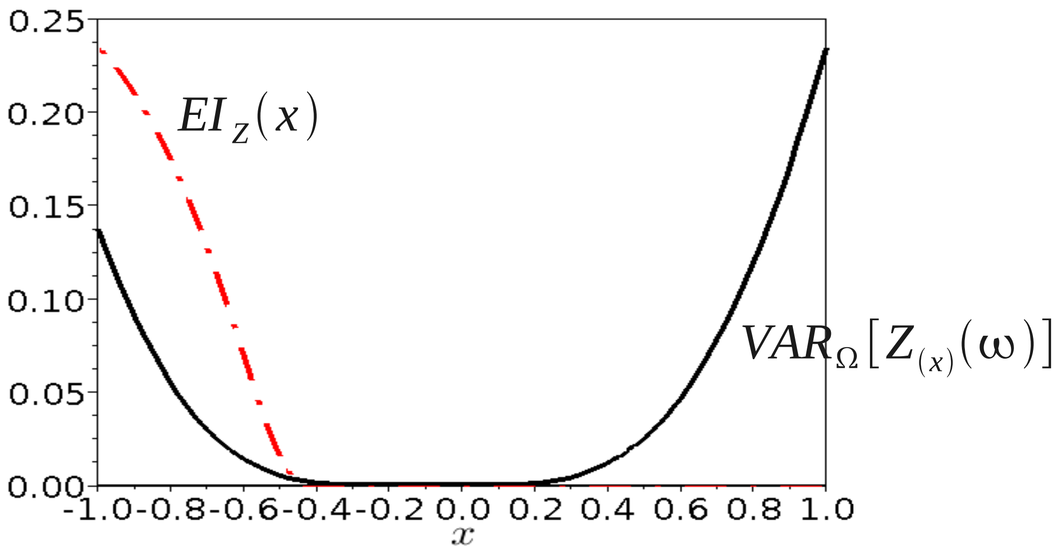
★ $-\ (x^{t+1}, u^{t+1})$

# Kriging based optimization with uncertainties, U controlled
## 2nd iteration

DOE and $E[Y(x,u)]$



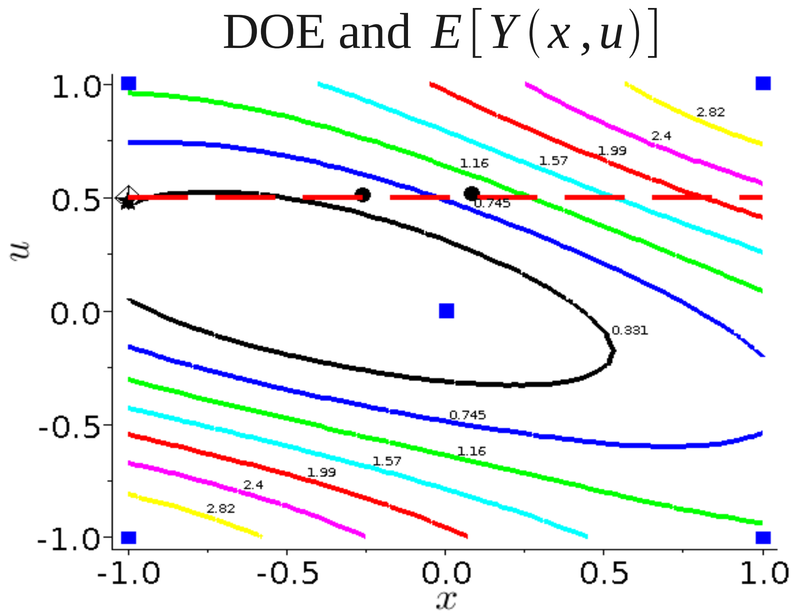



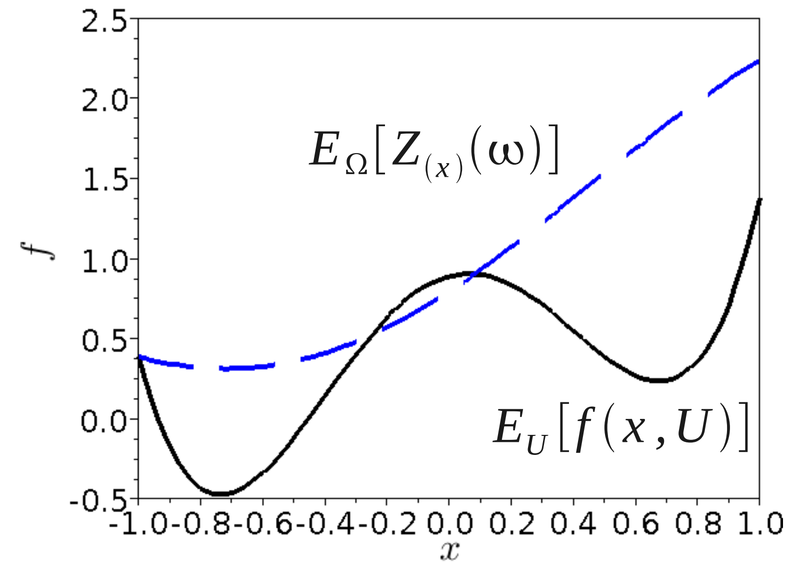

◇ – $(x^{next}, \mu)$

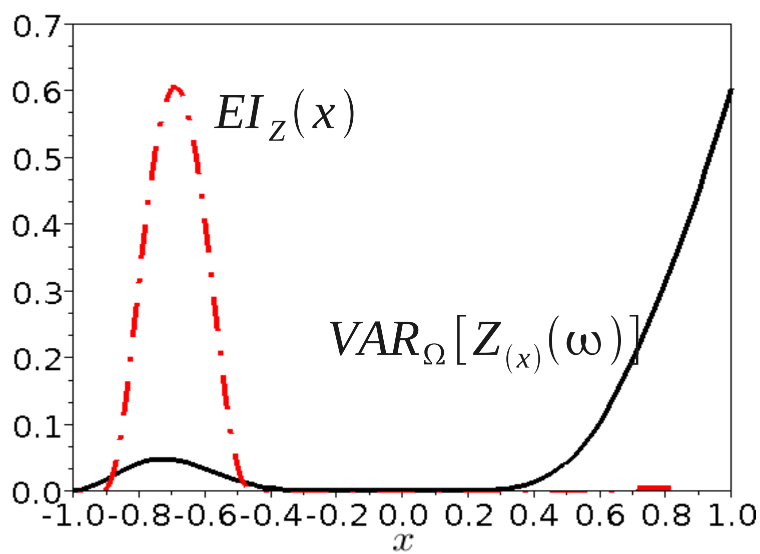
★ – $(x^{t+1}, u^{t+1})$

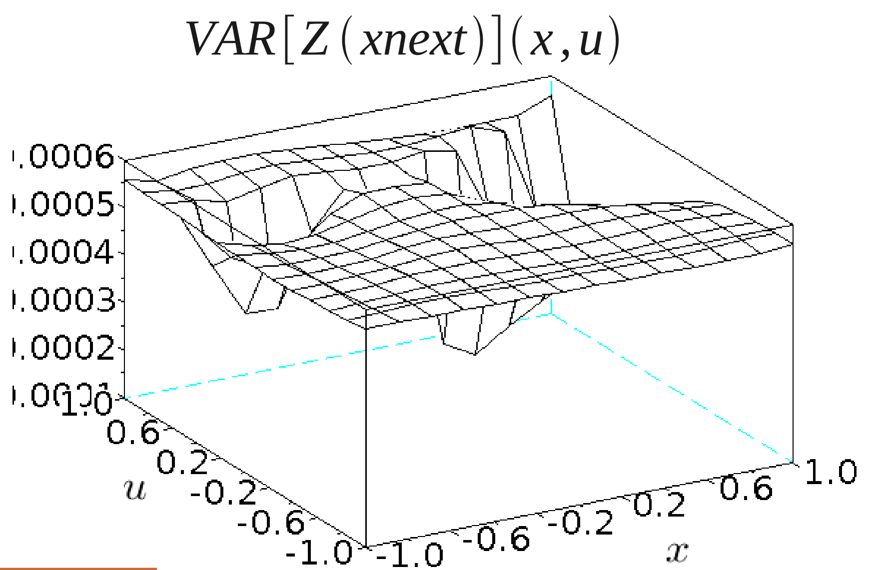
# Kriging based optimization with uncertainties, U controlled

## 3rd iteration



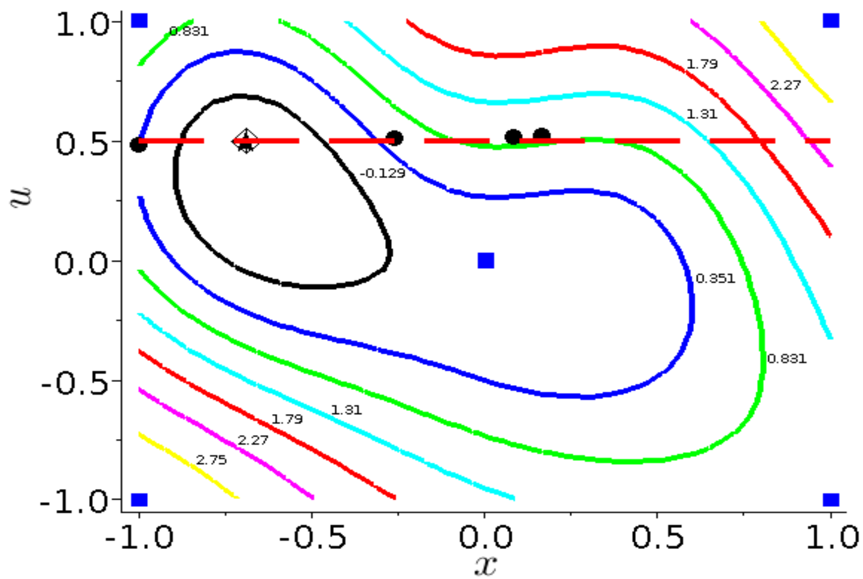



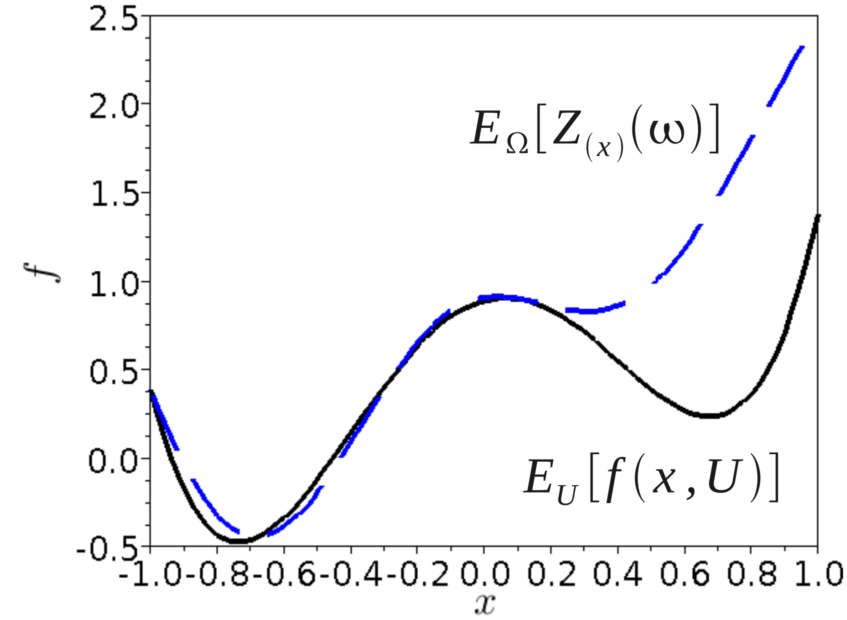



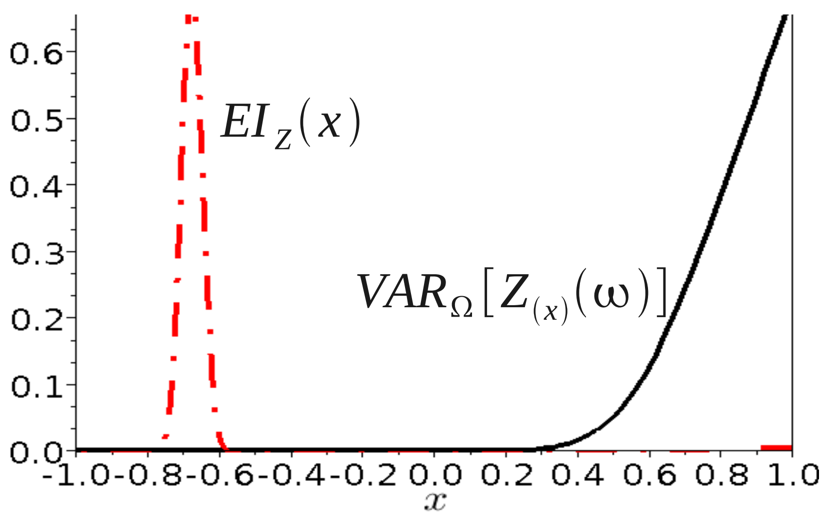



## 5th iteration

DOE and $E[Y(x,u)]$



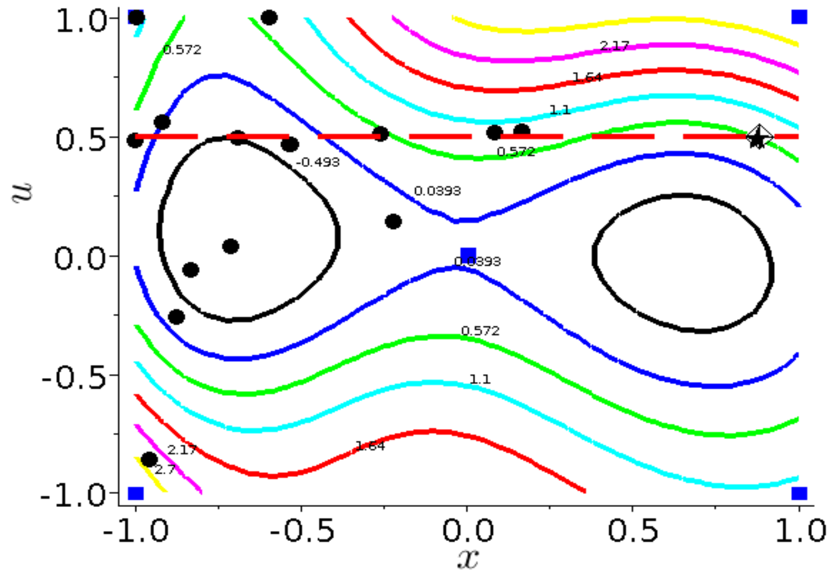



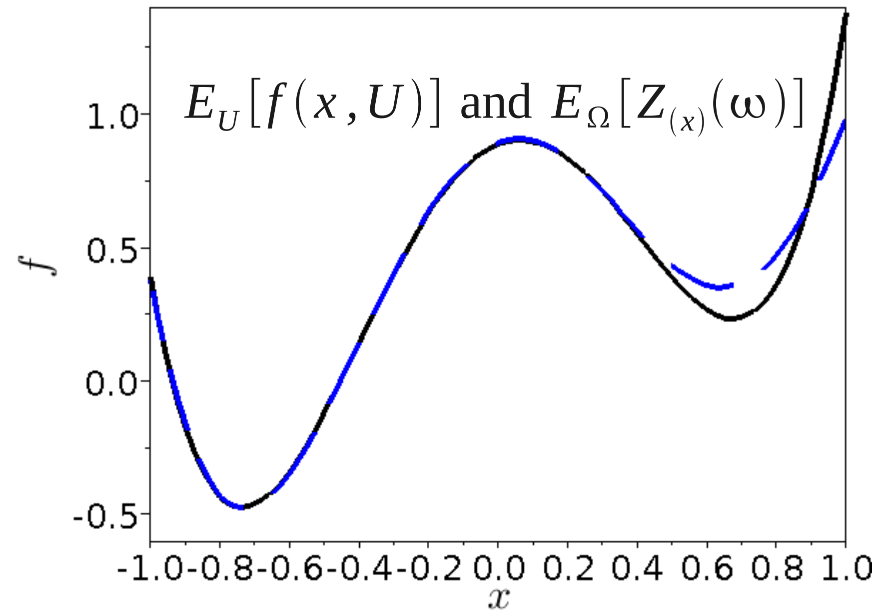

– $(x^{next},\mu)$

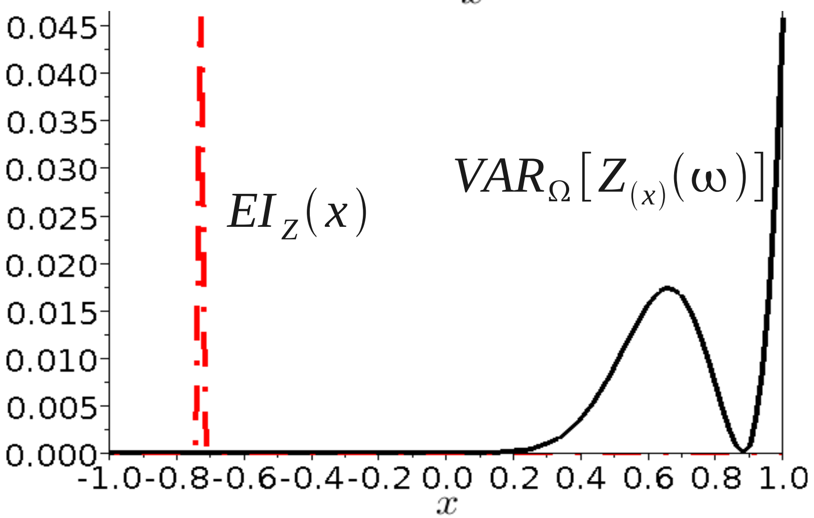
– $(x^{t+1},u^{t+1})$

# Kriging based optimization with uncertainties, U controlled
## 17th iteration

DOE and $E[Y(x,u)]$

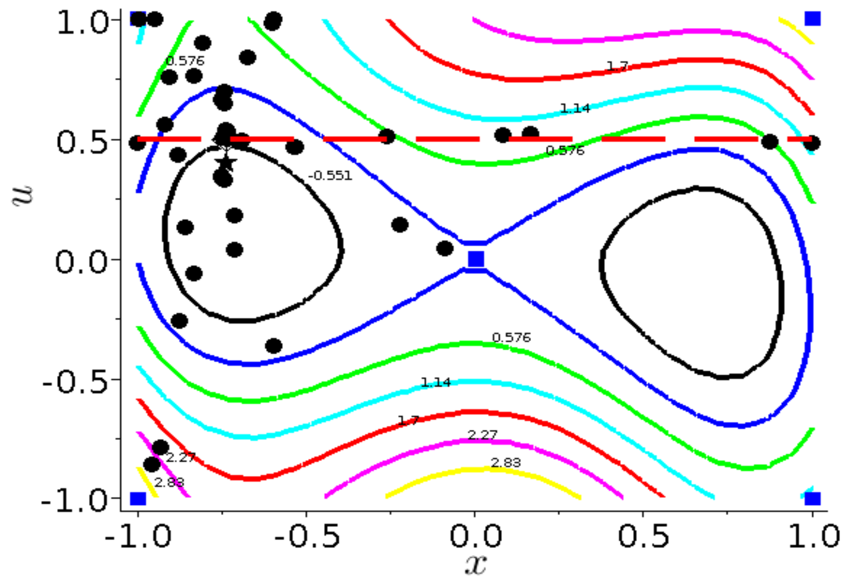


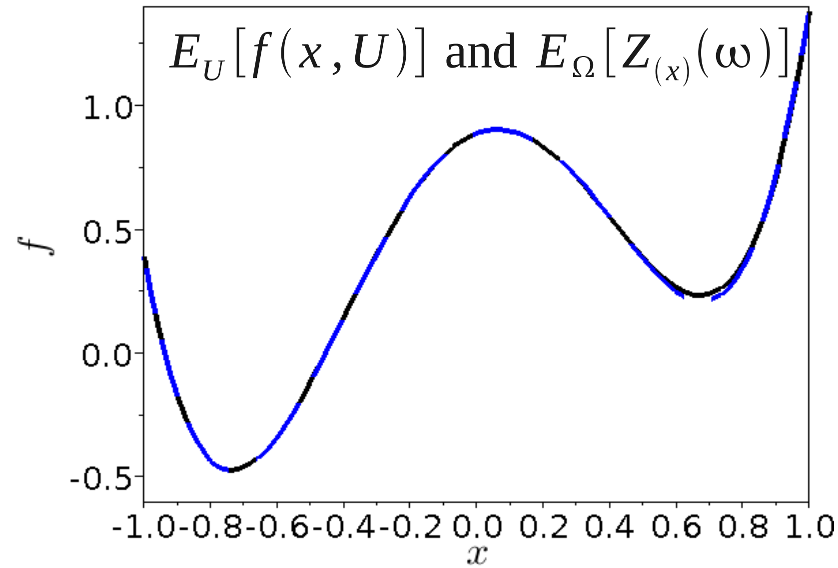



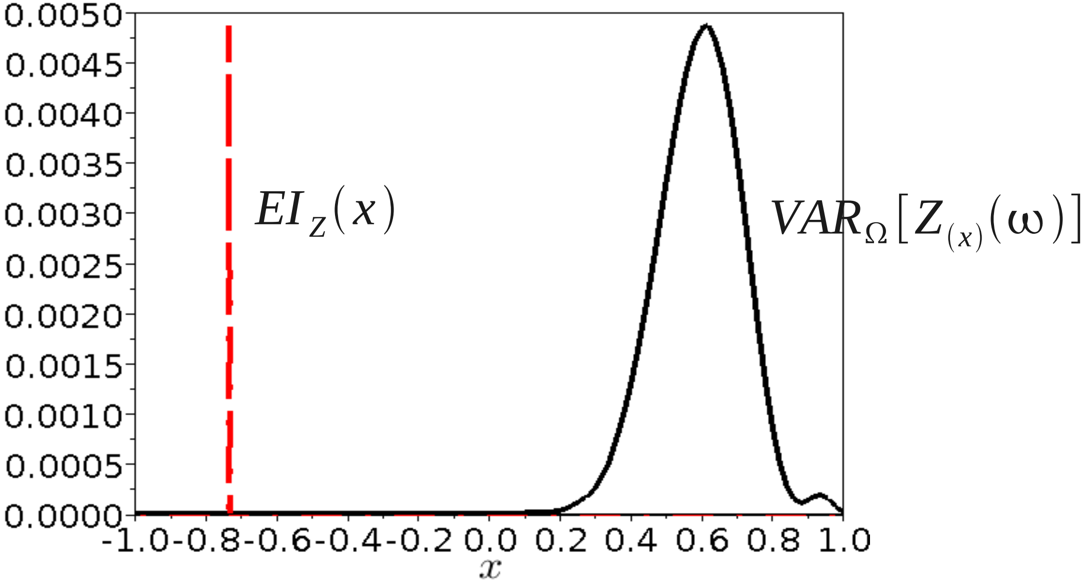



## Kriging based optimization with uncertainties, U controlled
# 50th iteration

DOE and $E[Y(x,u)]$



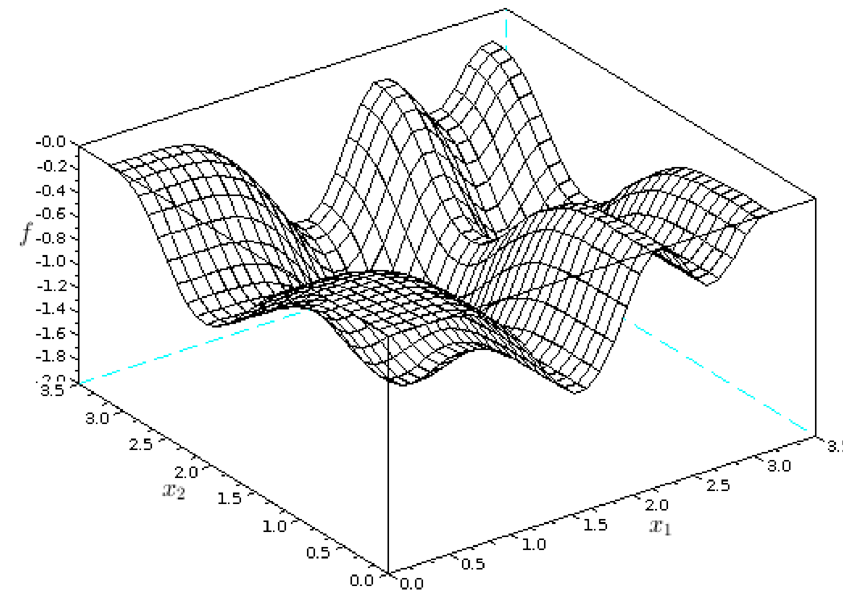



# Test functions

Test cases based on Michalewicz function

$$f(x) = -\sum_{i=1}^{n} \sin(x_i)\left[\sin(i x_i^2/\pi)\right]^2$$

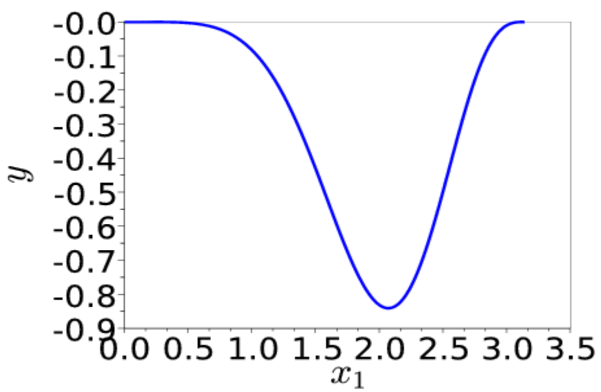
$$f(x,u) = f(x) + f(u)$$

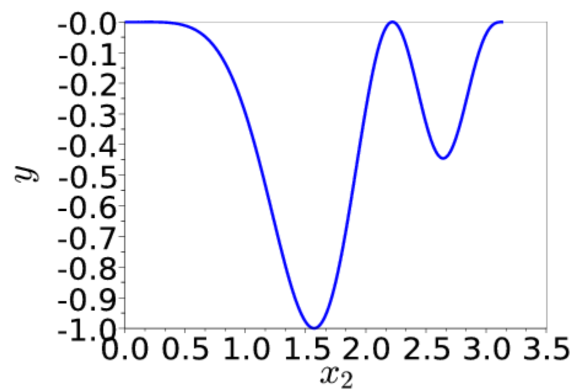


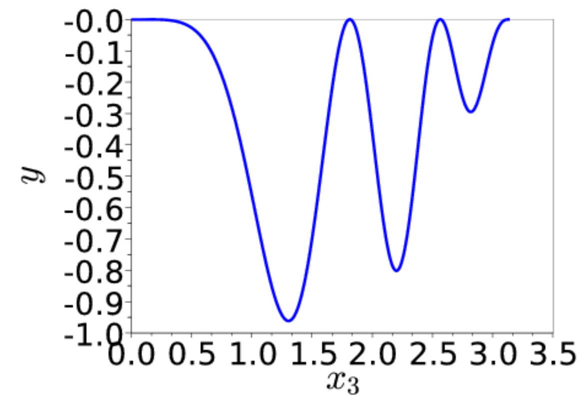

2D: $n_x=1 \quad n_u=1 \quad \mu=1.5 \quad \sigma=0.2$

4D: $n_x=2 \quad n_u=2 \quad \mu=[1.5,\ 2.1] \quad \sigma=[0.2,\ 0.2]$

6D: $n_x=3 \quad n_u=3 \quad \mu=[1.5,\ 2.1,\ 2] \quad \sigma=[0.2,\ 0.2,\ 0.3]$

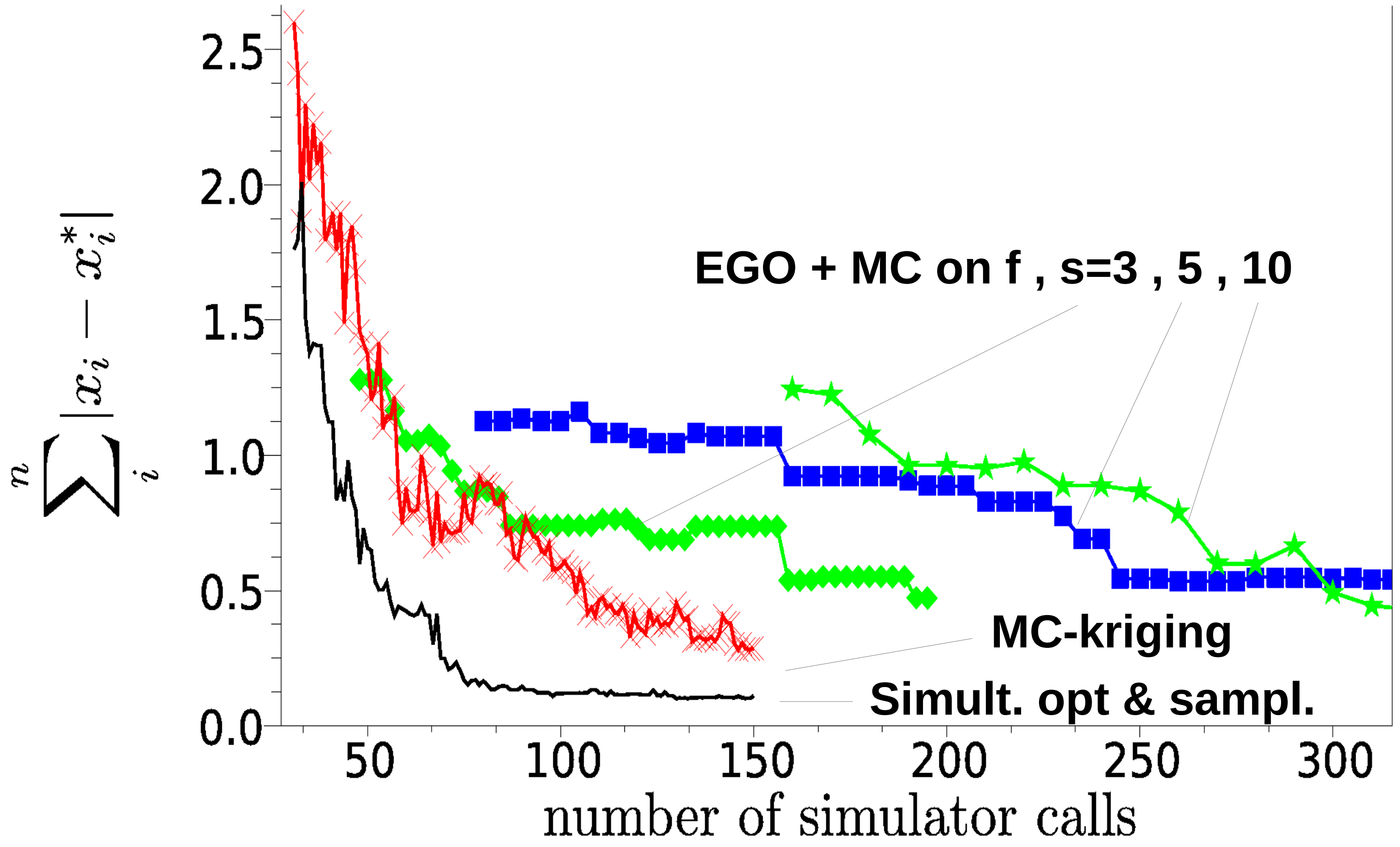


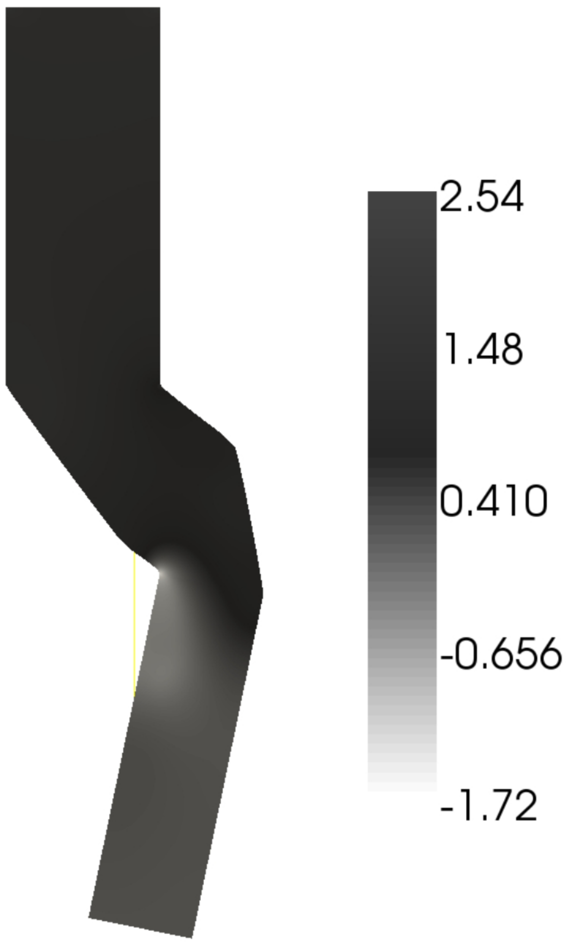
# Kriging based optimization with uncertainties, U controlled
# Test results

6D Michalewicz test case, $n_{x=3}$ =3 , $n_U$ =3 .
Initial DOE: RLHS , m=($n_x$+$n_U$)*5 = (3+3)*5 = 30;
10 runs for every method.



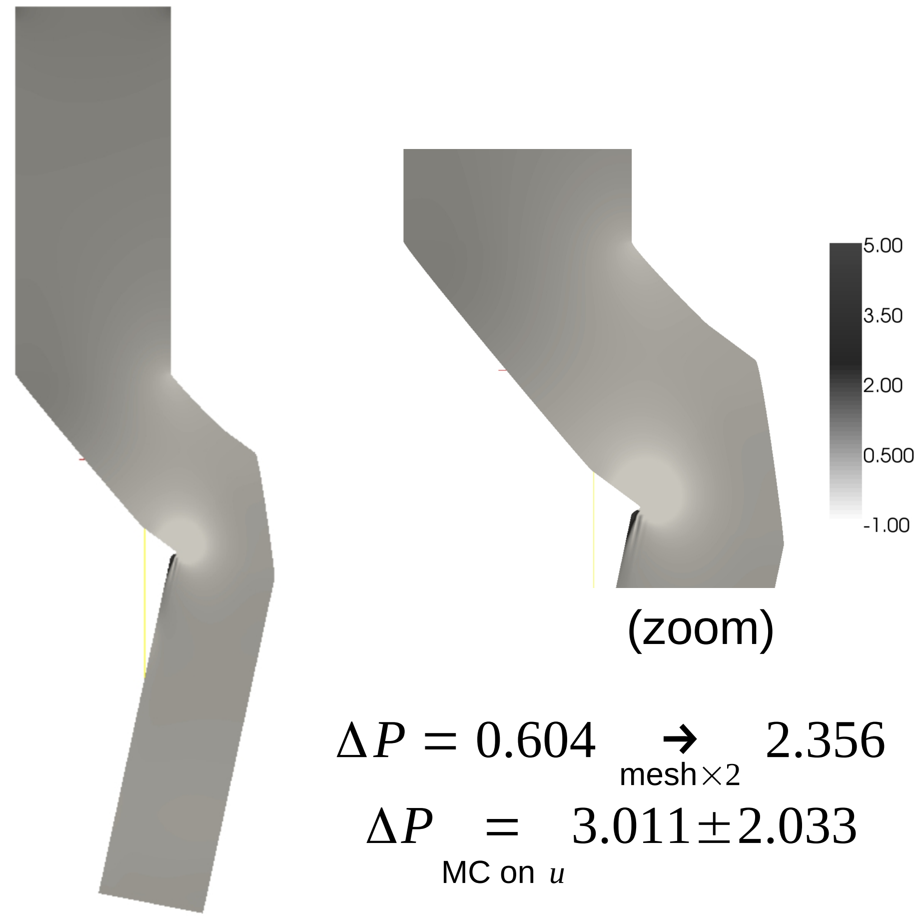
# Duct design with uncertain boundary conditions
# Pressure loss results

robust design




$$\Delta P \underset{\text{MC on } u}{=} 1.198 \pm 0.069$$

deterministic design for u=0



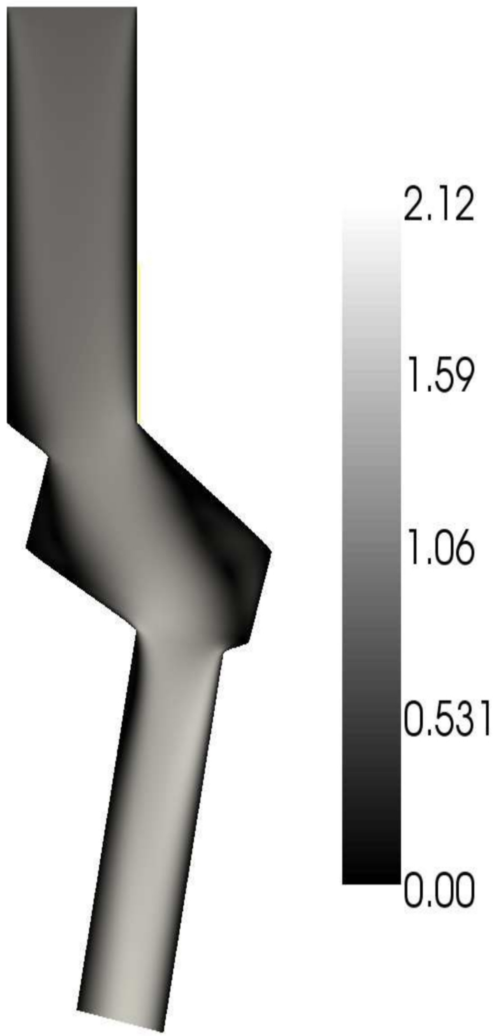

$$\Delta P = 0.604 \underset{\text{mesh}\times 2}{\rightarrow} 2.356$$

$$\Delta P \underset{\text{MC on } u}{=} 3.011 \pm 2.033$$

The result is not stable w.r.t. mesh changes.
The optimization exploits meshing flaws.

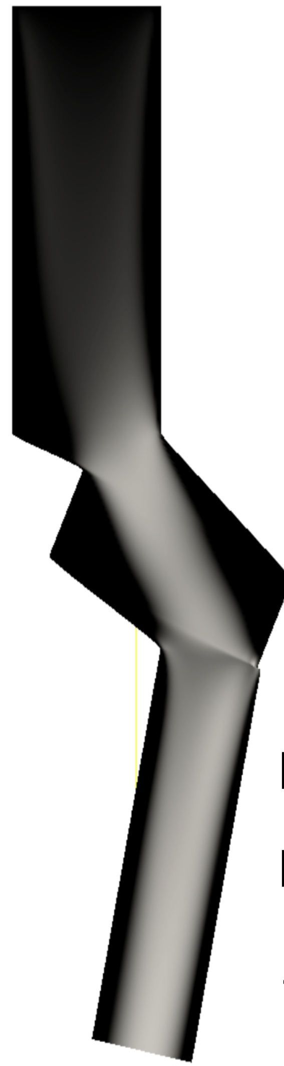
# Duct design with uncertain boundary conditions
# Flow uniformity results

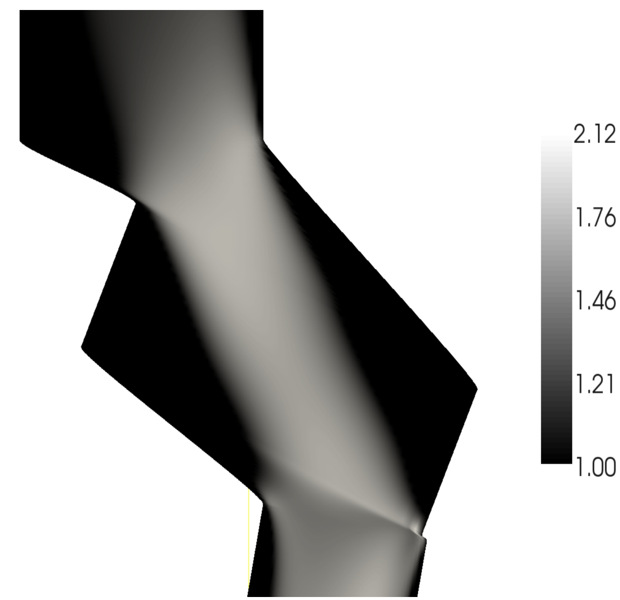
robust design




Flow Std Dev $\underset{\text{MC on } u}{=} 0.155 \pm 0.003$

deterministic design for u=0






(zoom)

Flow Std Dev $\underset{u=0}{=} 0.142 \xrightarrow[\text{mesh}\times 2]{} 0.532$

Flow Std Dev $\underset{\text{MC on } u}{=} 0.243 \pm 0.112$

The result is not stable w.r.t. mesh changes. The optimization exploits meshing flaws.

# partial conclusion

Accounting for uncertainties in design

mechanics → statistics

- is a practical issue (there are always model uncertainties or inherent randomnesses)
- raises difficult challenges that foster research

statistics → mechanics

- the collaboration between physical and statistical models will continue to  bring new ideas : optimizers are stringent tests for simulators, noise on u as a way to reduce mesh sensitivity, …

U controlled : J. Janusevskis and R. Le Riche, *Simultaneous kriging-based estimation and optimization of mean response*, Journal of Global Optimization, Springer, 2012

U not controlled : Le Riche, Picheny, Ginsbourger, Meyer, Kim, *Gears design with shape uncertainties using Monte Carlo simulations and kriging*, SDM, AIAA-2009-2257.

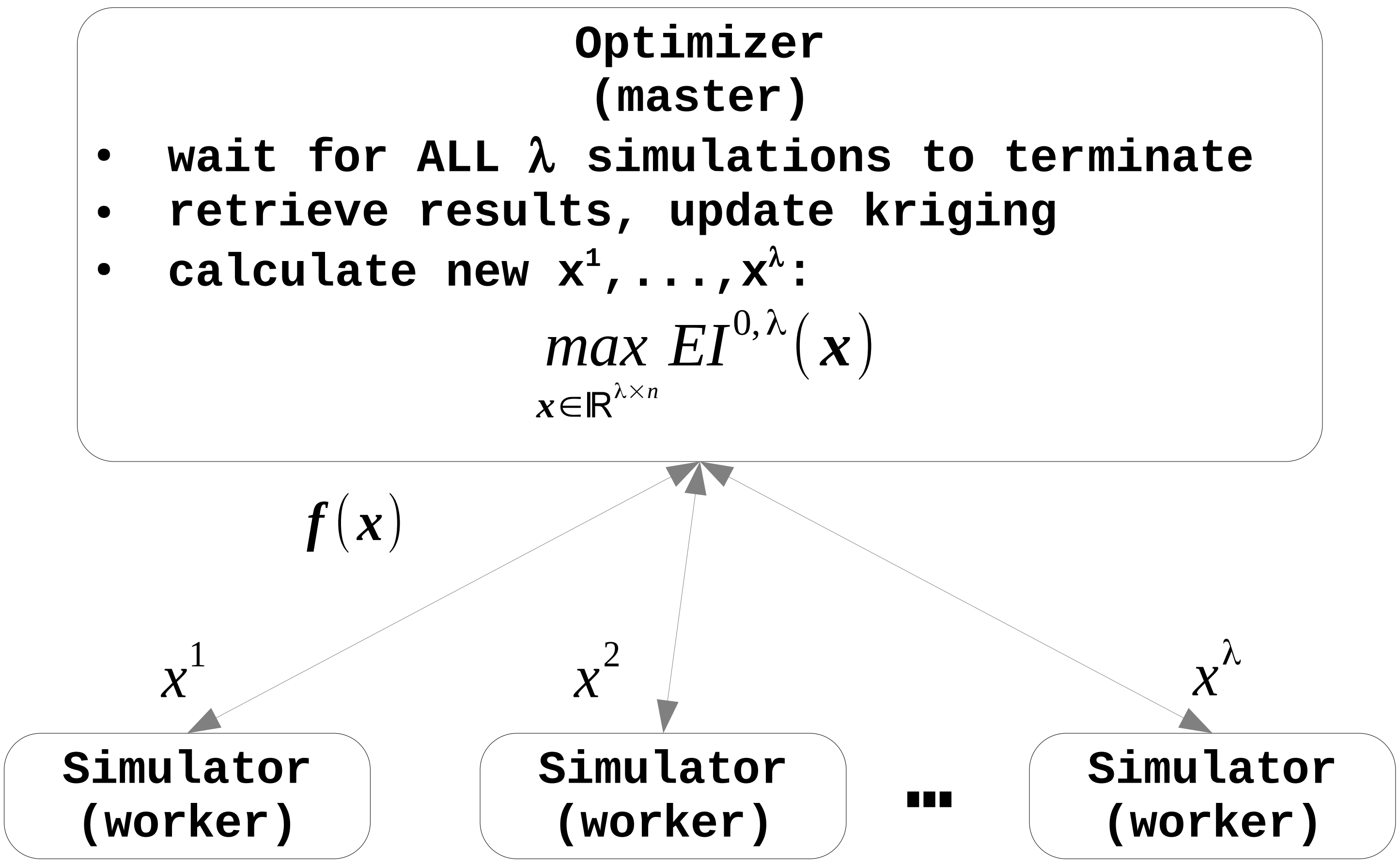
Noisy optimization : D. Salazar, R. Le Riche, G. Pujol and X. Bay, *An empirical study of the use of confidence levels in RBDO with Monte Carlo simulations*, in Multidisciplinary Design Optimization in Computational Mechanics, Wiley/ISTE Pub., 2010.

## Extensions of kriging-based optimization to parallel computing

- since the cost of calculating the objective function is a stumbling block
- Kriging key feature for distribution : joint information brought by a set of points can be measured

# Synchronous parallel EI : flow chart

A master-worker structure between computing nodes :



# Synchronous parallel EI : criterion

$\lambda$ nodes are available for new simulations at $x^1,\ldots,x^\lambda \ (\equiv \boldsymbol{x})$

$\rightarrow$ choose $x^1, \ldots, x^\lambda$ so that they maximize the synchronous $\lambda$ points EI

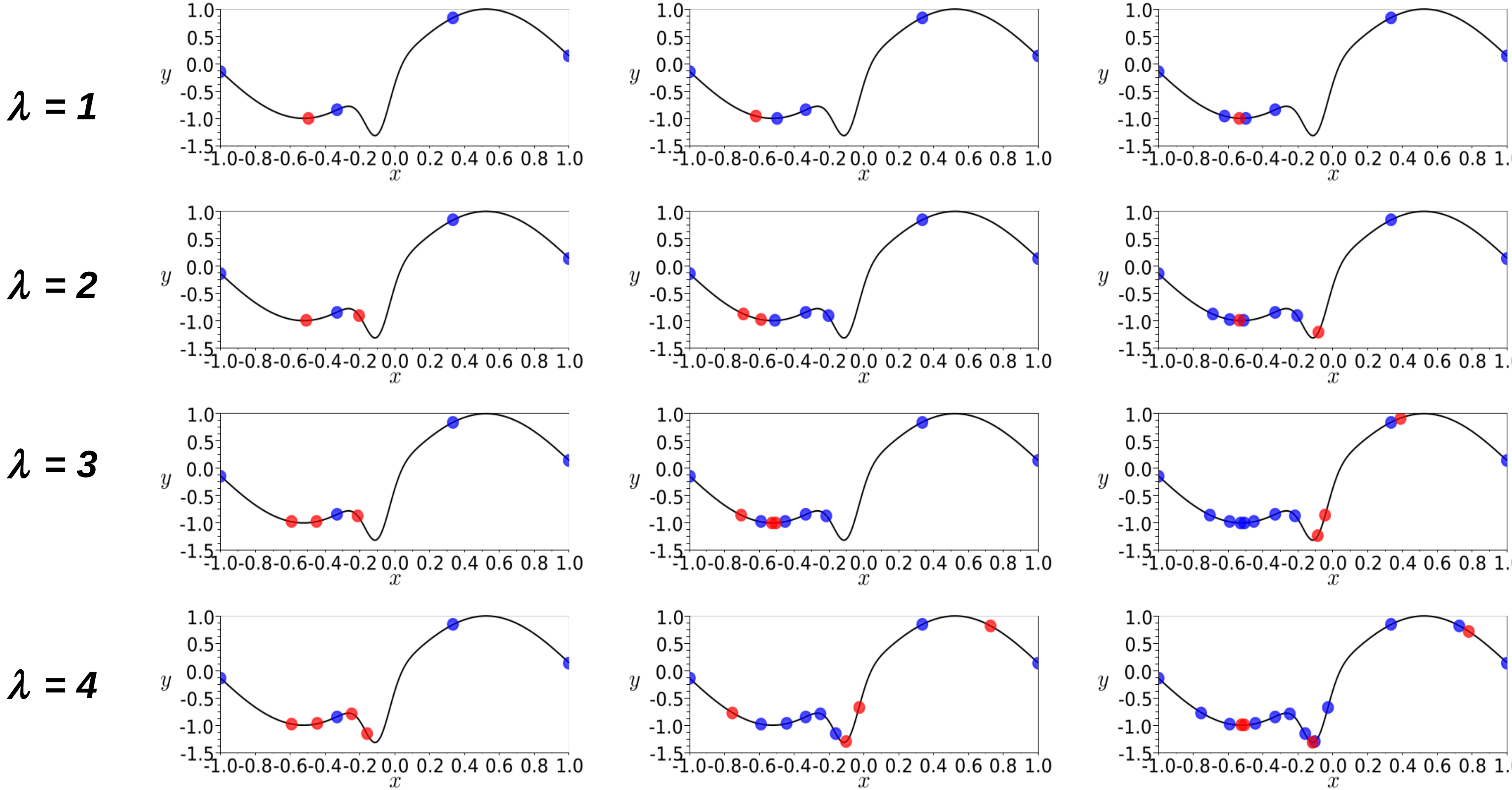
$$EI^{0,\lambda}(\boldsymbol{x}) = E\left[ f_{\text{min}} - min\left(F(x^1),\ldots,F(x^\lambda)\right) \right]^+ \ | \ F(x^{1\ldots M}) = f(x^{1\ldots M})$$

Compare to the sequential 1 point EI, from the EGO algorithm :

$$EI(x) \equiv EI^{0,1}(x) = E\left[f_{\text{min}} - F(x)\right]^+ \ | \ F(x^{1\ldots m}) = f(x^{1\ldots m})$$

[cf. D. Ginsbourger, R. Le Riche and L. Carraro, Kriging is well-suited to parallelize optimization, CIEOP, 2010 ]

# $EI^{0,\lambda}$ is different from repeated $EI^{0,1}$



# Limitations of $EI^{0,\lambda}$

**The number of nodes that can be used is limited by the problem to be solved**

$$\max_{\boldsymbol{x}\in\mathbb{R}^{\lambda\times n}} EI^{0,\lambda}(\boldsymbol{x})$$

**which is in dimension $\lambda \times n$ .**

**The computing nodes have different speeds and the simulations different durations.**

**Time model :**

$\lambda$ nodes

$T$ : time for 1 simulation, random variable , $T \sim U[t_{min}, t_{max}]$

$t_O =$ time for 1 optimization

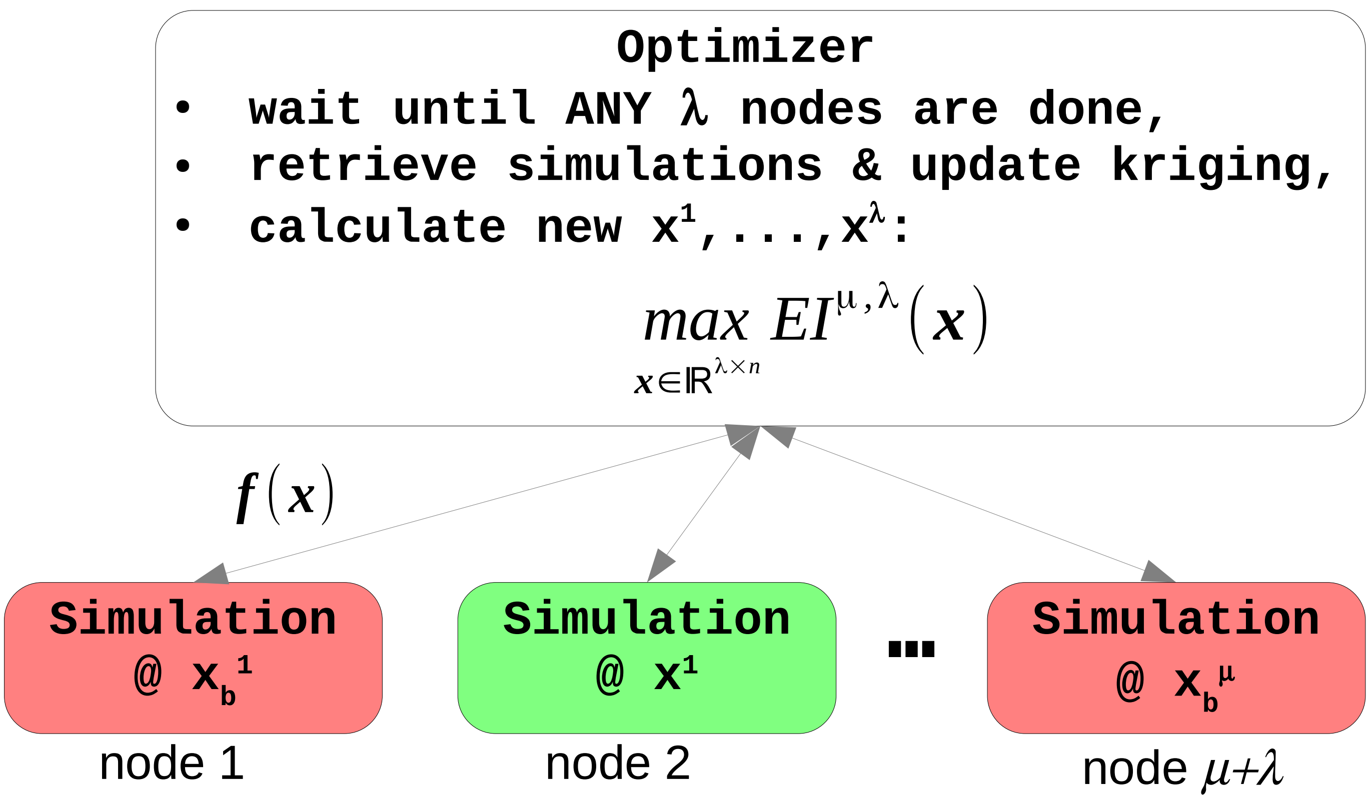
$T_{WC}$ : wall clock time for 1 generation

$$E(T_{WC}) = t_O + E(T_{\lambda:\lambda}) \underset{\lambda \gg 1}{\rightarrow} O(t_O + t_{max})$$

# Asynchronous parallel EI : flow chart

- It allows to use $m > \lambda+\mu$ nodes (actually ok for any optimizer that is not sensitive to the order of return of the points).

- *But* $EI^{\mu,\lambda}$ takes full account of past and on-going simulations and « optimally » (w.r.t. EI criterion) handles $\lambda+\mu$ nodes.



# Asynchronous parallel EI : criterion

$\lambda$ nodes are available for new simulations at $x^1,\ldots,x^\lambda \;(\equiv \boldsymbol{x})$

$\mu$ nodes are busy running simulations at $x_b^1,\ldots,x_b^\mu \;(\equiv \boldsymbol{x_b})$

$$EI^{\mu,\lambda}(\boldsymbol{x}) = E\left[min\left(f_{\text{min}}, F(\boldsymbol{x_b})\right) - min\left(F(\boldsymbol{x})\right)\right]^+ \mid F(x^{1\ldots M}) = f(x^{1\ldots M})$$

Recall the 1 point sequential EI and the synchronous EI :

$$EI(x) \equiv EI^{0,1}(x) = E\left[f_{\text{min}} - F(x)\right]^+ \mid F(x^{1\ldots m}) = f(x^{1\ldots m})$$

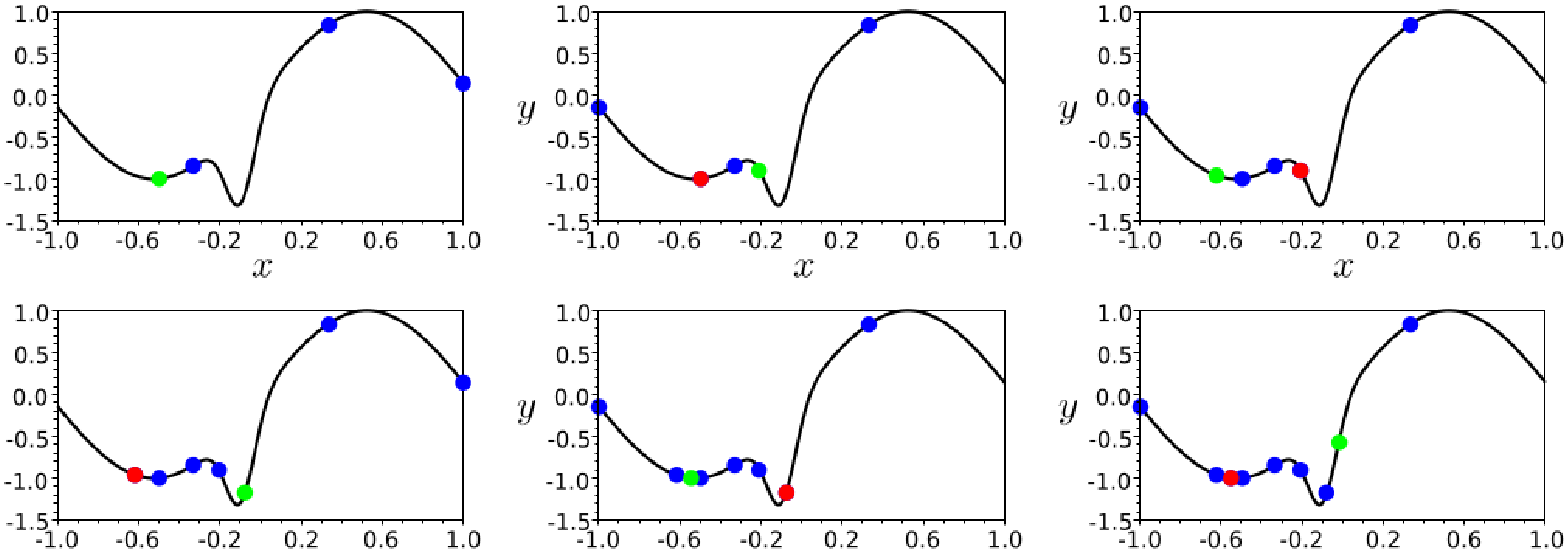
$$EI^{0,\lambda}(\boldsymbol{x}) = E\left[f_{\text{min}} - min\left(F(\boldsymbol{x})\right)\right]^+ \mid F(x^{1\ldots m}) = f(x^{1\ldots m})$$

Property : $EI^{\mu,\lambda}(\boldsymbol{x}) \to 0^+$ as $\boldsymbol{x} \to \boldsymbol{x_b}$

(the search is pushed away from already sampled points which are being evaluated)

# Asynchronous parallel EI : illustration

$$x^{t+1} = \underset{x \in S \subset \mathbb{R}^n}{arg\ max}\ EI^{\mu,\lambda}(\boldsymbol{x}) \quad \text{where} \quad \mu=1 \quad \text{and} \quad \lambda=1$$



# Advantages of $EI^{\mu,\lambda}$ over $EI^{0,\lambda}$

**The number of nodes used (m > $\mu$+$\lambda$) is not limited by**

$$\max_{\boldsymbol{x}\in\mathbb{R}^{\lambda\times n}} EI^{\mu,\lambda}(\boldsymbol{x})$$

**which is in dimension $\lambda \times n$ ( $\lambda$ = 1 as best default strategy )**

**Time model in $O(m^{-1})$ :**

$m > \mu + \lambda$ nodes

$T$ : time for 1 simulation, random variable, $T \sim U[t_{min}, t_{max}]$

$t_O =$ time for 1 optimization

$T_{WC}$ : wall clock time for 1 generation

$$E(T_{WC}) \approx t_O + \frac{E(t_{\lambda:m})}{m}$$

# Asynchronous parallel EI : results

100 independant runs on 3 functions, m = 32 computing nodes

| Label | Cost function | Domain | Minimal value | Modality |
|---|---|---|---|---|
| "michalewicz2d" | $\sum_{i=1}^{2} \sin(x_i)\sin^2(ix_i^2/\pi)$ | $[0,5]^2$ | $-1.841$ | multimodal |
| "rosenbrock6d" | $\sum_{i=1}^{5} 100(x_{i+1}-x_i^2)^2+(1-x_i)^2$ | $[0,5]^6$ | $0$ | unimodal |
| "rank1approx9d" | $\lVert \mathbf{A}_{4\times 5} - \mathbf{x}_{1\ldots 4}\mathbf{x}_{5\ldots 9}^T \rVert_2,\ a_{ij} \sim U(0,1)^1$ | $[-1,1]^9$ | $0.712$ | bimodal |

$S_G$ ; $S_T$ = generation speed up ; time speed up w.r.t. $EI^{0,1}$ sync (EGO)

| | micha2D $S_G$ ; $S_T$ | rosen6D $S_G$ ; $S_T$ | rank1 $S_G$ ; $S_T$ |
|---|---|---|---|
| $EI^{0,1}$ sync | 1 ; 1 | 1 ; 1 | 1 ; 1 |
| $EI^{0,4}$ sync | 3.8 ; 3.0 | 2.9 ; 2.3 | 1.3 ; 1.0 |
| $EI^{31,1}$ async | 0.8 ; 8.3 | 0.4 ; 4.4 | 0.4 ; 4.1 |
| $EI^{28,4}$ async | 2.58 ; 20.4 | 1.2 ; 9.2 | 0.8 ; 6.4 |

- **$EI^{\mu,\lambda}$ is better generation wise than $EI^{\mu,1}$**
- **asynchronous algos are slower generation wise than synchronous algos**
- **asynchronous algos are faster in wall-clock time than synchronous algos**

# Asynchronous parallel EI algorithm
# Selected bibliography

***EI*** $^{\mu,\lambda}$


analytical bounds

- J. Janusevskis, R. Le Riche and D. Ginsbourger, *Parallel expected improvements for global optimization: summary, bounds and speed-up*, HAL technical report no. hal-00613971, Aug. 2011.

Bayes approach, analytical bounds

- Janusevskis, J., Le Riche, R., Ginsbourger, D. and R. Girdziusas, *Expected improvements for the asynchronous parallel global optimization of expensive functions : potentials and challenges*, selected articles from the LION 6 Conference, LNCS 7219, Aug. 2012

MC evaluation

- J. Janusevskis, R. Girdziusas and R. Le Riche, *On integration of multi-point improvements*, NIPS workshop on Bayesian Optimization and Decision Making, Lake Tahoe, USA, dec. 2012.

time model, empirical tests

- R. Le Riche, R. Girdziusas and J. Janusevskis, *A study of asynchronous budgeted optimization* , NIPS workshop on Bayesian Optimization and Decision Making, Lake Tahoe, USA, dec. 2012.

# Conclusions

Thanks to its spatial convariance, kriging is a rich approach for optimizing with real simulators :

- mathematical framework for metamodel uncertainties
- reconciles design of experiments and optimization

Perspectives :

- high dimensions, large number of analyses
- optimization efficiency (e.g., BBOB contests)
- adding expert knowledge to the kernel choice
- multi-fidelity models and kriging based optimization