



HAL
open science

EFISUR A sampling criterion for constrained bayesian optimization with uncertainties

Céline Helbert, Reda El Amri, Rodolphe Le Riche, Christophette Blanchet-Scalliet, Sébastien da Veiga

► **To cite this version:**

Céline Helbert, Reda El Amri, Rodolphe Le Riche, Christophette Blanchet-Scalliet, Sébastien da Veiga. EFISUR A sampling criterion for constrained bayesian optimization with uncertainties. SIAM UQ, Feb 2024, Trieste, Italy. hal-04806339

HAL Id: hal-04806339

<https://hal.science/hal-04806339v1>

Submitted on 27 Nov 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

EFISUR

A sampling criterion for constrained bayesian optimization with uncertainties

Céline Helbert (ECL - ICJ)

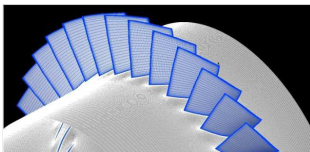
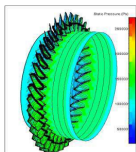
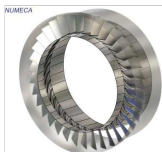
February 29, 2024

Joint work with Mohamed Reda El Amri (IFPEN), Rodolphe Le Riche (EMSE - LIMOS), Christophette Blanchet (ECL - ICJ) and Sébastien Da Veiga (SAFRAN).

Test case : the NASA rotor 37, a representative transonic axial-flow compressor

Industrial Problem

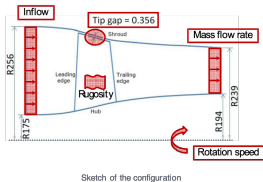
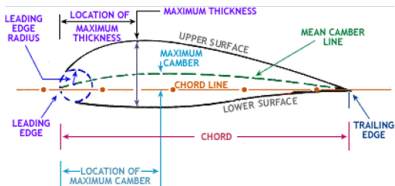
- How to design the blades to have **optimum** performances ?
- Under some **uncertainties** (manufacturing + external conditions)
- And **constraints** (manufacturing and design practices)



Test case : the NASA rotor 37, a representative transonic axial-flow compressor

Inputs

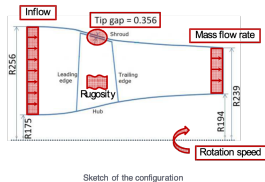
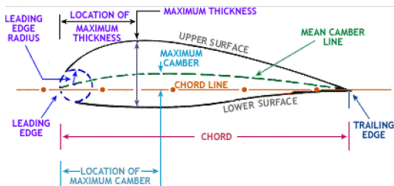
- 20 design variables, i.e. 5 shape parameters at 4 control points : the chord, the maximum thickness, the location of this maximum, the pitch angle, the sweep (sketched as α)
- 7 uncertain variables : 2 manufacturing uncertainties (tip gap, rugosity on the blade), 3 Inflow uncertainties (pressure, temperature, azimuthal momentum), 2 operational uncertainties (flow rate, rotation speed)



Test case : the NASA rotor 37, a representative transonic axial-flow compressor

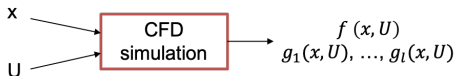
Inputs

- 20 design variables, i.e. 5 shape parameters at 4 control points : the chord, the maximum thickness, the location of this maximum, the pitch angle, the sweep (sketched as α)
- 7 uncertain variables : 2 manufacturing uncertainties (tip gap, rugosity on the blade), 3 Inflow uncertainties (pressure, temperature, azimuthal momentum), 2 operational uncertainties (flow rate, rotation speed)



outputs

- Objective function : polytropic efficiency (to be maximized)
- Constraints : inlet and outlet flow angles, deceleration, etc.



Problem formulation taking into account uncertainty

$$\min_{\mathbf{x} \in \mathcal{S}_{\mathcal{X}} \subset \mathbb{R}^d} \mathbb{E}[f(\mathbf{x}, \mathbf{U})] \quad \text{s.t.} \quad \mathbb{P}(g_1(\mathbf{x}, \mathbf{U}) \leq 0, \dots, g_l(\mathbf{x}, \mathbf{U}) \leq 0) \geq 1 - \alpha$$

with $\alpha = 5\%$ and $\mathbf{U} \sim \rho_{\mathbf{U}}$ with support $\mathcal{S}_{\mathbf{U}} \subset \mathbb{R}^m$



Problem formulation taking into account uncertainty

$$\min_{\mathbf{x} \in \mathcal{S}_{\mathcal{X}} \subset \mathbb{R}^d} \mathbb{E}[f(\mathbf{x}, \mathbf{U})] \quad \text{s.t.} \quad \mathbb{P}(g_1(\mathbf{x}, \mathbf{U}) \leq 0, \dots, g_l(\mathbf{x}, \mathbf{U}) \leq 0) \geq 1 - \alpha$$

with $\alpha = 5\%$ and $\mathbf{U} \sim \rho_{\mathbf{U}}$ with support $\mathcal{S}_{\mathbf{U}} \subset \mathbb{R}^m$

Equivalent formulation

$$\min_{\mathbf{x} \in \mathcal{S}_{\mathcal{X}}} z(\mathbf{x}) \quad \text{s.t.} \quad c(\mathbf{x}) \leq 0$$

where $z(\mathbf{x}) = \mathbb{E}[f(\mathbf{x}, \mathbf{U})]$ and $c(\mathbf{x}) = 1 - \alpha - \mathbb{E}[\mathbb{1}_{\{g_i(\mathbf{x}, \mathbf{U}) \leq 0, i=1, \dots, l\}}]$.



Problem formulation taking into account uncertainty

$$\min_{\mathbf{x} \in \mathcal{S}_{\mathcal{X}} \subset \mathbb{R}^d} \mathbb{E}[f(\mathbf{x}, \mathbf{U})] \quad \text{s.t.} \quad \mathbb{P}(g_1(\mathbf{x}, \mathbf{U}) \leq 0, \dots, g_l(\mathbf{x}, \mathbf{U}) \leq 0) \geq 1 - \alpha$$

with $\alpha = 5\%$ and $\mathbf{U} \sim \rho_{\mathbf{U}}$ with support $\mathcal{S}_{\mathbf{U}} \subset \mathbb{R}^m$

Equivalent formulation

$$\min_{\mathbf{x} \in \mathcal{S}_{\mathcal{X}}} z(\mathbf{x}) \quad \text{s.t.} \quad c(\mathbf{x}) \leq 0$$

where $z(\mathbf{x}) = \mathbb{E}[f(\mathbf{x}, \mathbf{U})]$ and $c(\mathbf{x}) = 1 - \alpha - \mathbb{E}[\mathbb{1}_{\{g_i(\mathbf{x}, \mathbf{U}) \leq 0, i=1, \dots, l\}}]$.

Objective

- Solve this problem through Bayesian Optimization
- Choose a convenient Gaussian process modeling and a relevant acquisition function

Outline

1. Introduction
2. Gaussian Process regression framework
3. Sequential sampling scheme
4. Numerical Experiments
5. Safran test case

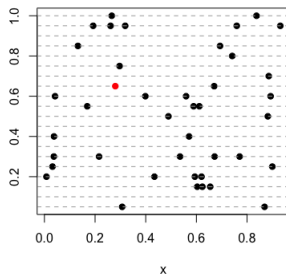
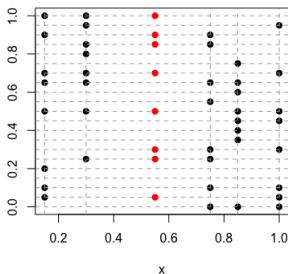
Recall of the problem formulation

$$\min_{\mathbf{x} \in \mathcal{S}_{\mathcal{X}}} z(\mathbf{x}) \quad \text{s.t.} \quad c(\mathbf{x}) \leq 0$$

where $z(\mathbf{x}) = \mathbb{E}[f(\mathbf{x}, \mathbf{U})]$ and $c(\mathbf{x}) = 1 - \alpha - \mathbb{E}[\mathbb{1}_{\{g_i(\mathbf{x}, \mathbf{U}) \leq 0, i=1, \dots, l\}}]$.

Remark

- Building surrogate models for z (expectation over \mathbf{U}) and c (probability over \mathbf{U}) would need too many computer code evaluations (Monte Carlo).
- We build surrogate models for f and g_1, \dots, g_l directly in the joint space $\mathcal{S}_{\mathcal{X}} \times \mathcal{S}_{\mathcal{U}}$. Surrogate models for z and c are then deduced.



Recall of the problem formulation

$$\min_{\mathbf{x} \in \mathcal{S}_X} z(\mathbf{x}) \quad \text{s.t.} \quad c(\mathbf{x}) \leq 0$$

where $z(\mathbf{x}) = \mathbb{E}[f(\mathbf{x}, \mathbf{U})]$ and $c(\mathbf{x}) = 1 - \alpha - \mathbb{E}[\mathbb{1}_{\{g_i(\mathbf{x}, \mathbf{u}) \leq 0, i=1, \dots, l\}}]$.

Remark

- Building surrogate models for z (expectation over \mathbf{U}) and c (probability over \mathbf{U}) would need too many computer code evaluations (Monte Carlo).
- We build surrogate models for f and g_1, \dots, g_l directly in the joint space $\mathcal{S}_X \times \mathcal{S}_U$. Surrogate models for z and c are then deduced.

Assume: f and g_1, \dots, g_l are realization of independent Gaussian processes F and G_i such that

$$F(\mathbf{x}, \mathbf{u}) \sim \mathcal{GP}(m_F(\mathbf{x}, \mathbf{u}), k_F(\mathbf{x}, \mathbf{u}, \mathbf{x}', \mathbf{u}')),$$
$$\forall i = \{1, \dots, l\}, G_i(\mathbf{x}, \mathbf{u}) \sim \mathcal{GP}(m_{G_i}(\mathbf{x}, \mathbf{u}), k_{G_i}(\mathbf{x}, \mathbf{u}, \mathbf{x}', \mathbf{u}')),$$

Let $F^{(t)}$ and $G_i^{(t)}$ denote the Gaussian processes conditioned on the t observations obtained at points $D^{(t)} = \{(\mathbf{x}_k, \mathbf{u}_k), k = 1, \dots, t\}$.

Surrogate model for z :

$$Z^{(t)}(\mathbf{x}) = \int_{\mathbb{R}^m} F^{(t)}(\mathbf{x}, \mathbf{u}) \rho_{\mathbf{U}}(\mathbf{u}) d\mathbf{u}$$

remark : Z is still a Gaussian process with known mean and known covariance function given by :

$$m_Z^{(t)}(\mathbf{x}) = \int_{\mathbb{R}^m} m_F^{(t)}(\mathbf{x}, \mathbf{u}) \rho_{\mathbf{U}}(\mathbf{u}) d\mathbf{u},$$
$$k_Z^{(t)}(\mathbf{x}, \mathbf{x}') = \int \int_{\mathbb{R}^{2m}} k_F^{(t)}(\mathbf{x}, \mathbf{u}, \mathbf{x}', \mathbf{u}') \rho_{\mathbf{U}}(\mathbf{u}, \mathbf{u}') d\mathbf{u} d\mathbf{u}'.$$

Surrogate model for z :

$$Z^{(t)}(\mathbf{x}) = \int_{\mathbb{R}^m} F^{(t)}(\mathbf{x}, \mathbf{u}) \rho_{\mathbf{U}}(\mathbf{u}) d\mathbf{u}$$

remark : Z is still a Gaussian process with known mean and known covariance function given by :

$$m_Z^{(t)}(\mathbf{x}) = \int_{\mathbb{R}^m} m_F^{(t)}(\mathbf{x}, \mathbf{u}) \rho_{\mathbf{U}}(\mathbf{u}) d\mathbf{u},$$
$$k_Z^{(t)}(\mathbf{x}, \mathbf{x}') = \int \int_{\mathbb{R}^{2m}} k_F^{(t)}(\mathbf{x}, \mathbf{u}, \mathbf{x}', \mathbf{u}') \rho_{\mathbf{U}}(\mathbf{u}, \mathbf{u}') d\mathbf{u} d\mathbf{u}'.$$

Surrogate model for c :

$$C^{(t)}(\mathbf{x}) = 1 - \alpha - \int_{\mathbb{R}^m} \mathbb{1}_{\cap_{i=1}^I \{G_i^{(t)}(\mathbf{x}, \mathbf{u}) \leq 0\}} \rho_{\mathbf{U}}(\mathbf{u}) d\mathbf{u}$$

remark : $C^{(t)}$ is not Gaussian anymore !

Algorithm Data-driven optimization in the joint space

- 1: Create an initial Design of Experiments (DoE) of size t in the joint space and calculate simulator responses:
 - 2: $D^{(t)} = \{(\mathbf{x}_i, \mathbf{u}_i), i = 1, \dots, t\}$, and associated $f^{(t)}$ and $g_i^{(t)}$
 - 3: **while** $t \leq$ maximum budget **do**
 - 4: Create the GPs of the objective and the constraints in the joint space:
 $F^{(t)}$ and $(G_i^{(t)})_{i=1}^l$
 - 5: Calculate the processes $Z^{(t)}$ and $C^{(t)}$ in the search space $\mathcal{S}_{\mathcal{X}}$
 - 6: Select $(\mathbf{x}_{t+1}, \mathbf{u}_{t+1})$ based on $Z^{(t)}$ and $C^{(t)}$
 - 7: Calculate simulator responses at the next point $(\mathbf{x}_{t+1}, \mathbf{u}_{t+1})$
 - 8: Update the DoE:
 - 9: $D^{(t+1)} = D^{(t)} \cup (\mathbf{x}_{t+1}, \mathbf{u}_{t+1})$, $f^{(t+1)} = f^{(t)} \cup f(\mathbf{x}_{t+1}, \mathbf{u}_{t+1})$,
 - 10: $g_i^{(t+1)} = g_i^{(t)} \cup g_i(\mathbf{x}_{t+1}, \mathbf{u}_{t+1})$, $i = 1, \dots, l$, $t \leftarrow t + 1$
 - 11: **end while**
 - 12: **end**
-

Algorithm Data-driven optimization in the joint space

- 1: Create an initial Design of Experiments (DoE) of size t in the joint space and calculate simulator responses:
 - 2: $D^{(t)} = \{(\mathbf{x}_i, \mathbf{u}_i), i = 1, \dots, t\}$, and associated $f^{(t)}$ and $g_i^{(t)}$
 - 3: **while** $t \leq$ maximum budget **do**
 - 4: Create the GPs of the objective and the constraints in the joint space:
 $F^{(t)}$ and $(G_i^{(t)})_{i=1}^l$
 - 5: Calculate the processes $Z^{(t)}$ and $C^{(t)}$ in the search space $\mathcal{S}_{\mathcal{X}}$
 - 6: **Select** $(\mathbf{x}_{t+1}, \mathbf{u}_{t+1})$ based on $Z^{(t)}$ and $C^{(t)}$
 - 7: Calculate simulator responses at the next point $(\mathbf{x}_{t+1}, \mathbf{u}_{t+1})$
 - 8: Update the DoE:
 - 9: $D^{(t+1)} = D^{(t)} \cup (\mathbf{x}_{t+1}, \mathbf{u}_{t+1}), f^{(t+1)} = f^{(t)} \cup f(\mathbf{x}_{t+1}, \mathbf{u}_{t+1}),$
 - 10: $g_i^{(t+1)} = g_i^{(t)} \cup g_i(\mathbf{x}_{t+1}, \mathbf{u}_{t+1}), i = 1, \dots, l, t \leftarrow t + 1$
 - 11: Set $n = n + 1$
 - 12: **end while**
 - 13: **end**
-

Outline

1. Introduction
2. Gaussian Process regression framework
3. Sequential sampling scheme
4. Numerical Experiments
5. Safran test case

General principle for robust global optimization algorithms

Define a progress measure $P(x)$ in relation with the problem formulation and calculated from the GPs trajectories

- $\mathbf{x}_{\text{targ}} = \arg \max_{\mathbf{x} \in \mathcal{S}_{\mathcal{X}}} \mathbb{E} (P^{(t)}(\mathbf{x}))$
- $(\mathbf{x}_{t+1}, \mathbf{u}_{t+1}) = \arg \min_{(\tilde{\mathbf{x}}, \tilde{\mathbf{u}}) \in \mathcal{S}_{\mathcal{X}} \times \mathcal{S}_{\mathcal{U}}} \text{VAR} (P^{(t+1)}(\mathbf{x}_{\text{targ}}))$ where $P^{(t+1)}$ is evaluated with GPs updated according to $D^{(t+1)} = D^{(t)} \cup \{(\tilde{\mathbf{x}}, \tilde{\mathbf{u}})\}$.

General principle for robust global optimization algorithms

Define a progress measure $P(x)$ in relation with the problem formulation and calculated from the GPs trajectories

- $\mathbf{x}_{\text{targ}} = \arg \max_{\mathbf{x} \in \mathcal{S}_{\mathcal{X}}} \mathbb{E} (P^{(t)}(\mathbf{x}))$
- $(\mathbf{x}_{t+1}, \mathbf{u}_{t+1}) = \arg \min_{(\tilde{\mathbf{x}}, \tilde{\mathbf{u}}) \in \mathcal{S}_{\mathcal{X}} \times \mathcal{S}_{\mathcal{U}}} \text{VAR} (P^{(t+1)}(\mathbf{x}_{\text{targ}}))$ where $P^{(t+1)}$ is evaluated with GPs updated according to $D^{(t+1)} = D^{(t)} \cup \{(\tilde{\mathbf{x}}, \tilde{\mathbf{u}})\}$.

What is a natural choice for $P(x)$?

General principle for robust global optimization algorithms

Define a progress measure $P(x)$ in relation with the problem formulation and calculated from the GPs trajectories

- $\mathbf{x}_{\text{targ}} = \arg \max_{\mathbf{x} \in \mathcal{S}_{\mathcal{X}}} \mathbb{E} (P^{(t)}(\mathbf{x}))$
- $(\mathbf{x}_{t+1}, \mathbf{u}_{t+1}) = \arg \min_{(\tilde{\mathbf{x}}, \tilde{\mathbf{u}}) \in \mathcal{S}_{\mathcal{X}} \times \mathcal{S}_{\mathcal{U}}} \text{VAR} (P^{(t+1)}(\mathbf{x}_{\text{targ}}))$ where $P^{(t+1)}$ is evaluated with GPs updated according to $D^{(t+1)} = D^{(t)} \cup \{(\tilde{\mathbf{x}}, \tilde{\mathbf{u}})\}$.

What is a natural choice for $P(x)$? **Feasible Improvement**

$$FI^{(t)}(\mathbf{x}) = I^{(t)}(\mathbf{x}) \mathbb{1}_{\{C^{(t)}(\mathbf{x}) \leq 0\}} ,$$

where $I^{(t)}(\mathbf{x}) = (z_{\min}^{\text{feas}} - Z^{(t)}(\mathbf{x}))^+$ denotes the improvement over the current feasible minimum value defined as $z_{\min}^{\text{feas}} = \min_{\mathbf{x} \in \mathcal{X}_t} m_Z^{(t)}(\mathbf{x})$ s.t. $\mathbb{E}[C^{(t)}(\mathbf{x})] \leq 0$.

Main steps of the EFISUR algorithm - first step

$$\mathbf{x}_{\text{targ}} = \arg \max_{\mathbf{x} \in \mathcal{S}_{\mathcal{X}}} \mathbb{E} (FI^{(t)}(\mathbf{x})) . \quad (1)$$

Because of the independence of Z and C ,

$$\mathbb{E} (FI^{(t)}(\mathbf{x})) = EI^{(t)}(\mathbf{x}) \mathbb{P}(C^{(t)}(\mathbf{x}) \leq 0)$$

- $EI^{(t)}(\mathbf{x}) = (z_{\min}^{\text{feas}} - m_Z^{(t)}(\mathbf{x})) \Phi \left(\frac{z_{\min}^{\text{feas}} - m_Z^{(t)}(\mathbf{x})}{\sigma_Z^{(t)}(\mathbf{x})} \right) + \sigma_Z^{(t)}(\mathbf{x}) \phi \left(\frac{z_{\min}^{\text{feas}} - m_Z^{(t)}(\mathbf{x})}{\sigma_Z^{(t)}(\mathbf{x})} \right)$,
- $\mathbb{P}(C^{(t)}(\mathbf{x}) \leq 0)$ can be approximated with available numerical methods.

Main steps of the EFISUR algorithm - second step

Minimize the variance of the one-step-ahead feasible improvement

$$(\mathbf{x}_{t+1}, \mathbf{u}_{t+1}) = \arg \min_{(\tilde{\mathbf{x}}, \tilde{\mathbf{u}}) \in \mathcal{S}_{\mathcal{X}} \times \mathcal{S}_{\mathcal{U}}} \text{VAR} \left(I^{(t+1)}(\mathbf{x}_{\text{targ}}) \mathbb{1}_{\{C^{(t+1)}(\mathbf{x}_{\text{targ}}) \leq 0\}} \right),$$

As it is too difficult, this problem is replaced by the following

$$\mathbf{u}_{t+1} = \arg \min_{\tilde{\mathbf{u}} \in \mathcal{S}_{\mathcal{U}}} \text{VAR} \left((z_{\min}^{\text{feas}} - Z^{(t+1)}(\mathbf{x}_{\text{targ}}))^+ \int_{\mathbb{R}^m} \text{VAR} \left(\mathbb{1}_{\cap_{i=1}^I \{G_i^{(t+1)}(\mathbf{x}_{\text{targ}}, \mathbf{u}) \leq 0\}} \right) \rho_{\mathbf{U}}(\mathbf{u}) d\mathbf{u} \right)$$

with GPs updated according to $D^{(t+1)} = D^{(t)} \cup \{(\mathbf{x}_{\text{targ}}, \tilde{\mathbf{u}})\}$.

Our algorithm : EFISUR

Algorithm Data-driven optimization in the joint space

- 1: Create an initial Design of Experiments (DoE) of size t in the joint space and calculate simulator responses:
- 2: $D^{(t)} = \{(\mathbf{x}_i, \mathbf{u}_i), i = 1, \dots, t\}$, and associated $f^{(t)}$ and $g_i^{(t)}$
- 3: **while** $t \leq$ maximum budget **do**
- 4: Create the GPs of f and g_1, \dots, g_l in the joint space: $F^{(t)}$ and $(G_i^{(t)})'_{i=1}$
- 5: Calculate the processes $Z^{(t)}$ and $C^{(t)}$ in the search space $\mathcal{S}_{\mathcal{X}}$
- 6: **Optimize** $\mathbb{E}(F^{(t)})$ and set $\mathbf{x}_{t+1} = \mathbf{x}_{\text{targ}}$ such that

$$\mathbf{x}_{\text{targ}} = \arg \max_{\mathbf{x} \in \mathcal{S}_{\mathcal{X}}} \text{EFl}^{(t)}(\mathbf{x})$$

- 7: **Sample** the next uncertain point by solving

8:

$$\mathbf{u}_{t+1} = \arg \min_{\tilde{\mathbf{u}} \in \mathcal{S}_{\mathcal{U}}} S(\mathbf{x}_{\text{targ}}, \tilde{\mathbf{u}})$$

- 9: Calculate simulator responses at the next point $(\mathbf{x}_{t+1}, \mathbf{u}_{t+1})$
- 10: Update the DoE:
- 11: $D^{(t+1)} = D^{(t)} \cup (\mathbf{x}_{t+1}, \mathbf{u}_{t+1})$, $f^{(t+1)} = f^{(t)} \cup f(\mathbf{x}_{t+1}, \mathbf{u}_{t+1})$,
- 12: $g_i^{(t+1)} = g_i^{(t)} \cup g_i(\mathbf{x}_{t+1}, \mathbf{u}_{t+1})$, $i = 1, \dots, l$, $t \leftarrow t + 1$
- 13: **end while**

Outline

1. Introduction
2. Gaussian Process regression framework
3. Sequential sampling scheme
4. Numerical Experiments
5. Safran test case

A first competitive algorithm : EFlrand

Algorithm Data-driven optimization in the joint space

- 1: Create an initial Design of Experiments (DoE) of size t in the joint space and calculate simulator responses:
- 2: $D^{(t)} = \{(\mathbf{x}_i, \mathbf{u}_i), i = 1, \dots, t\}$, and associated $f^{(t)}$ and $g_i^{(t)}$
- 3: **while** $t \leq$ maximum budget **do**
- 4: Create the GPs of f and g_1, \dots, g_l in the joint space: $F^{(t)}$ and $(G_i^{(t)})_{i=1}^l$
- 5: Calculate the processes $Z^{(t)}$ and $C^{(t)}$ in the search space $\mathcal{S}_{\mathcal{X}}$
- 6: **Optimize** $\mathbb{E}(F^{(t)})$ and set $\mathbf{x}_{t+1} = \mathbf{x}_{\text{targ}}$ such that

$$\mathbf{x}_{\text{targ}} = \arg \max_{\mathbf{x} \in \mathcal{S}_{\mathcal{X}}} \mathbb{E}F^{(t)}(\mathbf{x})$$

- 7: **Sample the next uncertain point randomly**, $\mathbf{u}_{t+1} \sim \rho \mathbf{u}$
 - 8: Calculate simulator responses at the next point $(\mathbf{x}_{t+1}, \mathbf{u}_{t+1})$
 - 9: Update the DoE:
 - 10: $D^{(t+1)} = D^{(t)} \cup (\mathbf{x}_{t+1}, \mathbf{u}_{t+1})$, $f^{(t+1)} = f^{(t)} \cup f(\mathbf{x}_{t+1}, \mathbf{u}_{t+1})$,
 - 11: $g_i^{(t+1)} = g_i^{(t)} \cup g_i(\mathbf{x}_{t+1}, \mathbf{u}_{t+1})$, $i = 1, \dots, l$, $t \leftarrow t + 1$
 - 12: **end while**
 - 13: **end**
-

A second competitive algorithm : cEldevNum

Algorithm Data-driven optimization in the joint space

- 1: Create an initial Design of Experiments (DoE) of size t in the joint space and calculate simulator responses:
- 2: $D^{(t)} = \{(\mathbf{x}_i, \mathbf{u}_i), i = 1, \dots, t\}$, and associated $f^{(t)}$ and $g_i^{(t)}$
- 3: **while** $t \leq$ maximum budget **do**
- 4: Create the GPs of f and g_1, \dots, g_l in the joint space: $F^{(t)}$ and $(G_i^{(t)})'_{i=1}$
- 5: Calculate the GP of the mean objective, $Z^{(t)}$, in the search space $\mathcal{S}_{\mathcal{X}}$
- 6: **Optimize** the EI under quantile constraints :

$$\mathbf{x}_{t+1} = \arg \max_{\mathbf{x} \in \mathcal{S}_{\mathcal{X}}} \text{EI}^{(t)}(\mathbf{x}) \text{ s. t. } \forall i \in \{1, \dots, l\}, q_{1-\alpha/l}(m_{G_i}^{(t)}(\mathbf{x}, \mathbf{U})) \leq 0$$

- 7: **Sample** the next uncertainty by minimizing the deviation number,

$$\mathbf{u}_{t+1} = \arg \min_{\mathbf{u}} \min_{i=1, \dots, l} \frac{|m_{G_i}^{(t)}(\mathbf{x}_{t+1}, \mathbf{u})|}{\sigma_{G_i}^{(t)}(\mathbf{x}_{t+1}, \mathbf{u})}$$

- 8: Calculate simulator responses at the next point $(\mathbf{x}_{t+1}, \mathbf{u}_{t+1})$
 - 9: Update the DoE:
 - 10: **end while**
 - 11: **end**
-

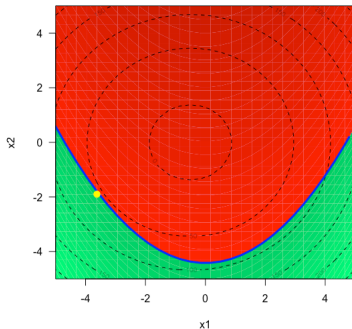
The problem is the following:

$$f(\mathbf{x}, \mathbf{u}) = 5(x_1^2 + x_2^2) - (u_1^2 + u_2^2)$$

$$+ x_1(u_2 - u_1 + 5)$$

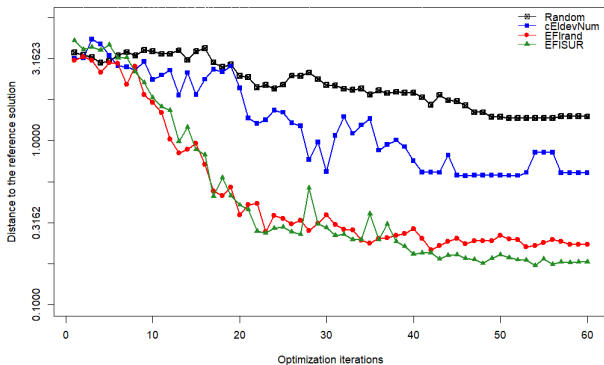
$$+ x_2(u_1 - u_2 + 3)$$

$$g(\mathbf{x}, \mathbf{u}) = -x_1^2 + 5x_2 - u_1 + u_2^2 - 1$$

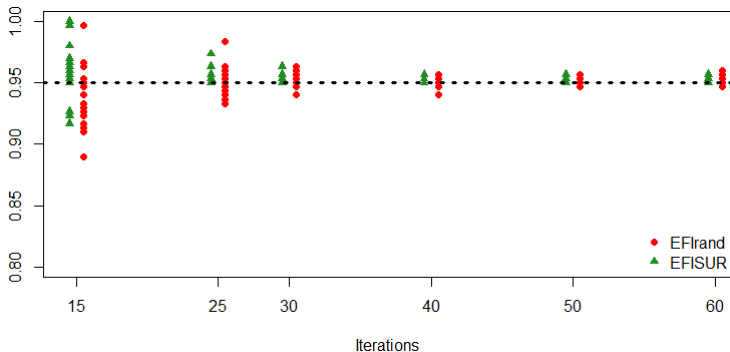


minimize $\mathbb{E}[f(\mathbf{x}, \mathbf{U})]$ such that $\mathbb{P}(g(\mathbf{x}, \mathbf{U}) \leq 0) \geq 1 - \alpha$

with $\mathbf{x} \in [-5, 5]^2$, $\mathbf{U} \sim \mathcal{U}([-5, 5]^2)$

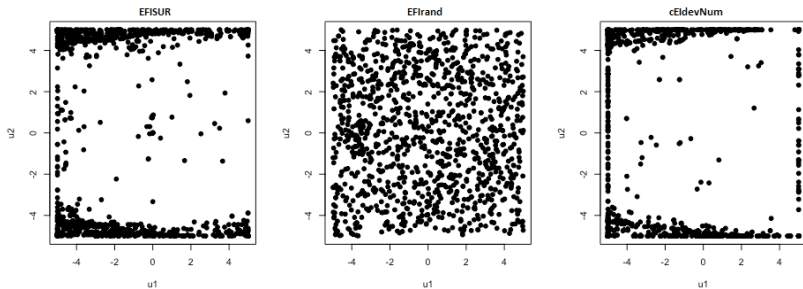


Distance to the reference solution. The mean is calculated from 30 replications of the runs. Initial DOE are composed of 8 points.



A posteriori probability of satisfying the constraint at the current “feasible” point z_{\min}^{feas} for different iterations of the EFISUR and EFIRand strategies.

$$z_{\min}^{\text{feas}} = \min_{\mathbf{x} \in \mathcal{X}_t} m_Z^{(t)}(\mathbf{x}) \text{ s.t. } \mathbb{E}[C^{(t)}(\mathbf{x})] \leq 0.$$

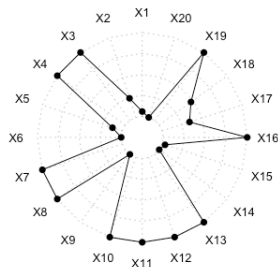
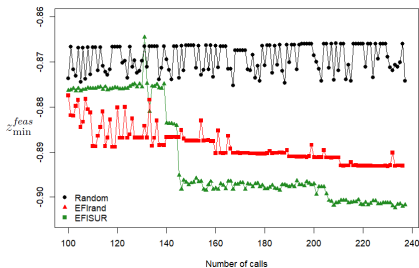


Enrichment in the uncertain space, \mathcal{S}_U , for the three methods.

Outline

1. Introduction
2. Gaussian Process regression framework
3. Sequential sampling scheme
4. Numerical Experiments
5. Safran test case

SAFRAN test case:



- Convergence history of the average objective function of the current feasible minimum, z_{\min}^{feas} . Initial DOE are composed of 100 points in dimension 27.
- Relative coordinates of the optimal design with respect to their respective lower and upper bounds.

Thank you for your attention