Adaptive surrogate models for reliability analysis and reliability-based design optimization

Vincent Dubourg

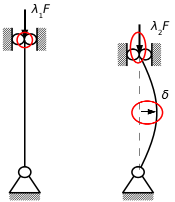
Phimeca Engineering SA
Laboratoire de Mécanique & Ingénieries

Ph. D. Defense – IFMA INSTITUT FRANÇAIS DE MÉCANIQUE AVANCÉE, Clermont-Ferrand – December 5, 2011

Context: The design of imperfect shells against buckling

Buckling is a structural instability phenomenon

- triggered by some excessive load (*to be determined*);
- whose magnitude depends on *uncertain* initial conditions (*e.g.* geometry, material properties and boundary conditions);
- affecting *slender* structures.



Axially compressed beam

Railway track

Silo

"*Slenderness* is the trademark of *optimally designed* structures."

(Ramm & Wall, 2004)

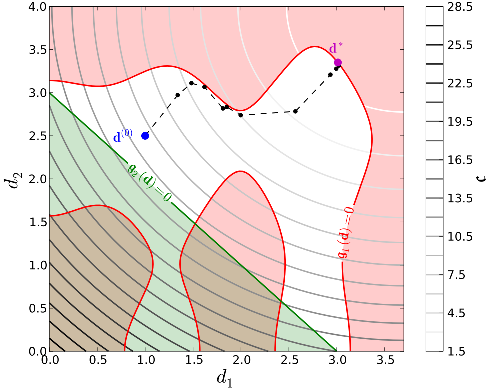
Context: The design of imperfect shells against buckling

Buckling is *the major failure scenario* for submarines pressure hulls.

Design problem formulation

Deterministic design optimization

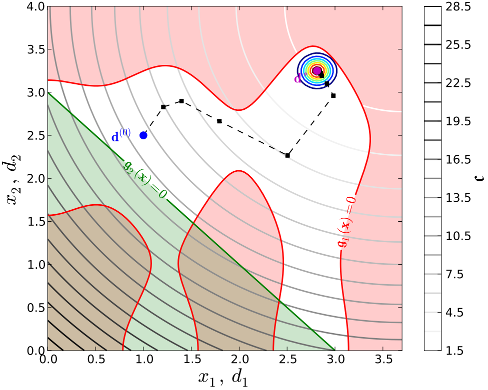
$$\boldsymbol{d}^* = \arg\min_{\boldsymbol{d}\in\mathbb{D}} \mathfrak{c}(\boldsymbol{d}) \ : \begin{cases} \mathfrak{f}_i(\boldsymbol{d}) \leq 0, i = 1, \ldots, n_c \\ \mathfrak{g}_l(\boldsymbol{x}, \boldsymbol{d}) \geq 0, l = 1, \ldots, n_p \end{cases}$$



Design problem formulation

Reliability-based design optimization

$$\boldsymbol{d}^* = \arg\min_{\boldsymbol{d}\in\mathbb{D}} \mathfrak{c}(\boldsymbol{d}) \;:\; \begin{cases} f_i(\boldsymbol{d}) \leq 0, i = 1, \ldots, n_c \\ \mathbb{P}\left[\mathfrak{g}_l(\boldsymbol{X}) \leq 0 \mid \boldsymbol{d}\right] \leq p^0_{f\,l}, l = 1, \ldots, n_p \end{cases}$$



Premise & objectives

- Structural stability models are *computationally expensive* (mostly finite-element based).

⇒ Replace the original expensive model with a *cheaper meta-model*.

- *Reliability approximation techniques* (such as FORM) cannot guarantee the safety level of their designs.

⇒ Develop a strategy that is able to *guarantee the design's safety*.

- Stakeholders target *highly reliable designs*.

⇒ The overall strategy should be scalable to *low failure probabilities*.

A particular interest has been given to *quantifying*, *reducing* and *eliminating* the error induced by the use of a surrogate.

Premise & objectives

- Structural stability models are *computationally expensive* (mostly finite-element based).

⇒ Replace the original expensive model with a *cheaper meta-model*.

- *Reliability approximation techniques* (such as FORM) cannot guarantee the safety level of their designs.

⇒ Develop a strategy that is able to *guarantee the design's safety*.

- Stakeholders target *highly reliable designs*.

⇒ The overall strategy should be scalable to *low failure probabilities*.

A particular interest has been given to *quantifying*, *reducing* and *eliminating* the error induced by the use of a surrogate.

Outline

1. Gaussian process meta-modelling
2. Adaptive designs of experiments
3. Reliability analysis
4. Reliability-based design optimization

Outline

Meta-modelling

Meta-modelling techniques

- aim at constructing a *predictor* $\widetilde{\mathcal{M}}$
- that *mimics* the behaviour of an existing model $\mathcal{M}$

$$\boldsymbol{x} \in \mathbb{X} \subseteq \mathbb{R}^n \longrightarrow \mathcal{M} \longrightarrow y \in \mathbb{Y} \subseteq \mathbb{R}$$

- from a collection of observations gathered in a *dataset*:

$$\mathcal{D} = \left\{\left(\boldsymbol{x}^{(i)}, y_i\right), i = 1, \ldots, m\right\}, \quad y_i = \mathcal{M}\left(\boldsymbol{x}^{(i)}\right), i = 1, \ldots, m$$

- and *statistical considerations*.

Interest for reliability-based design

Such predictors are *much faster to evaluate* than the original model $\mathcal{M}$, and come with a sort of *confidence measure*.

Meta-modelling

Meta-modelling techniques

- aim at constructing a *predictor* $\widetilde{\mathcal{M}}$
- that *mimics* the behaviour of an existing model $\mathcal{M}$

$$\boldsymbol{x} \in \mathbb{X} \subseteq \mathbb{R}^n \longrightarrow \boxed{\mathcal{M}} \longrightarrow y \in \mathbb{Y} \subseteq \mathbb{R}$$

- from a collection of observations gathered in a *dataset*:

$$\mathcal{D} = \left\{\left(\boldsymbol{x}^{(i)}, y_i\right), i = 1, \ldots, m\right\}, \quad y_i = \mathcal{M}\left(\boldsymbol{x}^{(i)}\right), i = 1, \ldots, m$$

- and *statistical considerations*.

Interest for reliability-based design

Such predictors are *much faster to evaluate* than the original model $\mathcal{M}$, and come with a sort of *confidence measure*.

Gaussian process meta-modelling

The Gaussian process prior model

(*Santner* et al., *2003*)

The function $\mathcal{M}$ is a sample path of a Gaussian process (GP) Y:

$$Y(\boldsymbol{x}) = \boldsymbol{f}(\boldsymbol{x})^{\mathsf{T}}\,\boldsymbol{\beta} + Z(\boldsymbol{x}), \quad \boldsymbol{x} \in \mathbb{X}$$

where:

- $\boldsymbol{f}(\boldsymbol{x})^{\mathsf{T}}\,\boldsymbol{\beta}$ is a linear regression model;
- $Z(\boldsymbol{x})$ is a zero-mean, stationary GP with covariance:

$$\mathrm{Cov}\left[Y(\boldsymbol{x}),\, Y(\boldsymbol{x}')\right] = \sigma^2\, R\left(\boldsymbol{x} - \boldsymbol{x}',\, \boldsymbol{\theta}\right), \quad (\boldsymbol{x}, \boldsymbol{x}') \in \mathbb{X} \times \mathbb{X}$$

Hence, given a vector of *observations* $\boldsymbol{Y} = \left(Y_i = Y\left(\boldsymbol{x}^{(i)}\right),\, i = 1, \ldots, m\right)$ and an unobserved value $Y(\boldsymbol{x})$, we have:

$$\begin{Bmatrix} Y(\boldsymbol{x}) \\ \boldsymbol{Y} \end{Bmatrix} \sim \mathcal{N}_{1+m}\left(\begin{Bmatrix} \boldsymbol{f}(\boldsymbol{x})^{\mathsf{T}}\,\boldsymbol{\beta} \\ \mathbf{F}\,\boldsymbol{\beta} \end{Bmatrix}, \sigma^2 \begin{bmatrix} 1 & \boldsymbol{r}(\boldsymbol{x})^{\mathsf{T}} \\ \boldsymbol{r}(\boldsymbol{x}) & \mathbf{R} \end{bmatrix}\right)$$

whose parameters $\mathbf{F}$, $\boldsymbol{r}(\boldsymbol{x})$, $\mathbf{R}$ are inherited from the GP's statistics ($\boldsymbol{f}$ and R).

Gaussian process meta-modelling
The Gaussian process prior model

(Santner et al.*, 2003)*

The function $\mathcal{M}$ is a sample path of a Gaussian process (GP) Y:

$$Y(\boldsymbol{x}) = \boldsymbol{f}(\boldsymbol{x})^{\mathsf{T}} \boldsymbol{\beta} + Z(\boldsymbol{x}), \quad \boldsymbol{x} \in \mathbb{X}$$

where:

- $\boldsymbol{f}(\boldsymbol{x})^{\mathsf{T}} \boldsymbol{\beta}$ is a linear regression model;
- $Z(\boldsymbol{x})$ is a zero-mean, stationary GP with covariance:

$$\mathrm{Cov}\left[Y(\boldsymbol{x}),\, Y(\boldsymbol{x}')\right] = \sigma^2 R\left(\boldsymbol{x} - \boldsymbol{x}',\, \boldsymbol{\theta}\right), \quad (\boldsymbol{x},\, \boldsymbol{x}') \in \mathbb{X} \times \mathbb{X}$$

Hence, given a vector of *observations* $\boldsymbol{Y} = \left(Y_i = Y\left(\boldsymbol{x}^{(i)}\right),\, i = 1, \ldots, m\right)$ and an unobserved value $Y(\boldsymbol{x})$, we have:

$$\left\{ \begin{array}{c} Y(\boldsymbol{x}) \\ \boldsymbol{Y} \end{array} \right\} \sim \mathcal{N}_{1+m} \left(\left\{ \begin{array}{c} \boldsymbol{f}(\boldsymbol{x})^{\mathsf{T}} \boldsymbol{\beta} \\ \mathbf{F}\, \boldsymbol{\beta} \end{array} \right\},\, \sigma^2 \begin{bmatrix} 1 & \boldsymbol{r}(\boldsymbol{x})^{\mathsf{T}} \\ \boldsymbol{r}(\boldsymbol{x}) & \mathbf{R} \end{bmatrix} \right)$$

whose parameters $\mathbf{F}$, $\boldsymbol{r}(\boldsymbol{x})$, $\mathbf{R}$ are inherited from the GP's statistics ($\boldsymbol{f}$ and R).

Gaussian process meta-modelling

Posterior ≡ The best linear unbiased predictor (BLUP) *(Santner et al., 2003)*

Here, we are interested in the *posterior* distribution of the unobserved value given $\boldsymbol{y} = \left(y_i = \mathcal{M}\left(\boldsymbol{x}^{(i)}\right), i = 1, \ldots, m\right)$:

$$\widehat{Y}(\boldsymbol{x}) = [Y(\boldsymbol{x}) \,|\, \boldsymbol{Y} = \boldsymbol{y}]$$

The fundamental theorem of prediction *(Santner et al., 2003)*

$\widehat{Y}(\boldsymbol{x})$ is the *best linear unbiased predictor* w.r.t. the mean squared error:

$$\widehat{Y}(\boldsymbol{x}) = \boldsymbol{a}^*(\boldsymbol{x})^\top \boldsymbol{Y}$$

with:

$$\boldsymbol{a}^*(\boldsymbol{x}) = \arg \min_{\boldsymbol{a}(\boldsymbol{x}) \in \mathbb{R}^m} \mathbb{E}\left[\left(\widehat{Y}(\boldsymbol{x}) - Y(\boldsymbol{x})\right)^2\right] : \mathbb{E}\left[\left(\widehat{Y}(\boldsymbol{x}) - Y(\boldsymbol{x})\right)\right] = 0$$

Gaussian process meta-modelling

Posterior ≡ The best linear unbiased predictor (BLUP) *(Santner et al., 2003)*

Here, we are interested in the *posterior* distribution of the unobserved value given $\boldsymbol{y} = \left(y_i = \mathcal{M}\left(\boldsymbol{x}^{(i)}\right),\, i = 1, \ldots, m\right)$:

$$\widehat{Y}(\boldsymbol{x}) = \left[Y(\boldsymbol{x}) \,|\, \boldsymbol{Y} = \boldsymbol{y}\right]$$

The fundamental theorem of prediction ***(Santner et al., 2003)***

$\widehat{Y}(\boldsymbol{x})$ is the *best linear unbiased predictor* w.r.t. the mean squared error:

$$\widehat{Y}(\boldsymbol{x}) = \boldsymbol{a}^*(\boldsymbol{x})^\mathsf{T}\, \boldsymbol{Y}$$

with:

$$\boldsymbol{a}^*(\boldsymbol{x}) = \arg \min_{\boldsymbol{a}(\boldsymbol{x}) \in \mathbb{R}^m} \mathbb{E}\left[\left(\widehat{Y}(\boldsymbol{x}) - Y(\boldsymbol{x})\right)^2\right] : \mathbb{E}\left[\left(\widehat{Y}(\boldsymbol{x}) - Y(\boldsymbol{x})\right)\right] = 0$$

Gaussian process meta-modelling

The universal Kriging predictor: *an* empirical BLUP

(Santner et al.*, 2003)*

The *universal Kriging* predictor is also Gaussian:

$$\widehat{Y}(\boldsymbol{x}) = \left[Y(\boldsymbol{x}) \,\middle|\, \boldsymbol{Y} = \boldsymbol{y}, \sigma^2, \boldsymbol{\theta} \right] \sim \mathcal{N}_1\left(\mu_{\widehat{Y}}(\boldsymbol{x}), \sigma^2_{\widehat{Y}}(\boldsymbol{x}) \right)$$

where the *mean prediction* $\mu_{\widehat{Y}}(\boldsymbol{x})$ and the *prediction variance* $\sigma^2_{\widehat{Y}}(\boldsymbol{x})$ read:

$$\mu_{\widehat{Y}}(\boldsymbol{x}) = \boldsymbol{f}(\boldsymbol{x})^{\mathsf{T}} \widehat{\boldsymbol{\beta}} + \boldsymbol{r}(\boldsymbol{x})^{\mathsf{T}} \mathbf{R}^{-1} \left(\boldsymbol{y} - \mathbf{F} \widehat{\boldsymbol{\beta}} \right)$$

$$\sigma^2_{\widehat{Y}}(\boldsymbol{x}) = \sigma^2 \left(1 - \boldsymbol{r}(\boldsymbol{x})^{\mathsf{T}} \mathbf{R}^{-1} \boldsymbol{r}(\boldsymbol{x}) + \boldsymbol{u}(\boldsymbol{x})^{\mathsf{T}} \left(\mathbf{F}^{\mathsf{T}} \mathbf{R}^{-1} \mathbf{F} \right)^{-1} \boldsymbol{u}(\boldsymbol{x}) \right)$$

where:

$$\widehat{\boldsymbol{\beta}} = \left(\mathbf{F}^{\mathsf{T}} \mathbf{R}^{-1} \mathbf{F} \right)^{-1} \mathbf{F}^{\mathsf{T}} \mathbf{R}^{-1} \boldsymbol{y}$$

is the *generalized least squares estimate* of $\boldsymbol{\beta}$, and:

$$\boldsymbol{u}(\boldsymbol{x}) = \mathbf{F}^{\mathsf{T}} \mathbf{R}^{-1} \boldsymbol{r}(\boldsymbol{x}) - \boldsymbol{f}(\boldsymbol{x})$$

Gaussian process meta-modelling

Inference of the empirical BLUP parameters

Estimation techniques for $\left(\sigma^2, \boldsymbol{\theta}\right)$ include:

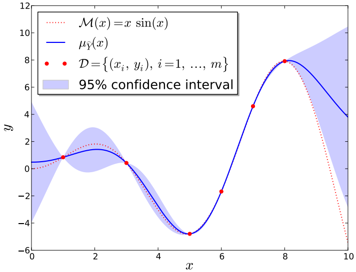
- Variogram estimation *(Cressie, 1993; Chilès and Delfiner, 1999)*
- Cross-validation *(Dubrule, 1983)*
- Bayesian predictors *(Handcock and Stein, 1993; Santner* et al.*, 2003)*
- *Maximum likelihood estimation* *(Welch* et al.*, 1992; Marrel* et al.*, 2008)*

The most common practice in computer experiments is the *maximum likelihood estimation* technique:

$$\left(\sigma^{2*}, \boldsymbol{\theta}^*\right) = \arg \max_{(\sigma^2, \boldsymbol{\theta})} \mathsf{L}\left(\boldsymbol{y} \mid \sigma^2, \boldsymbol{\theta}\right),$$

L being the *likelihood* of the observations w.r.t. Y (Gaussian).

Illustration on a one-dimensional regression exercise



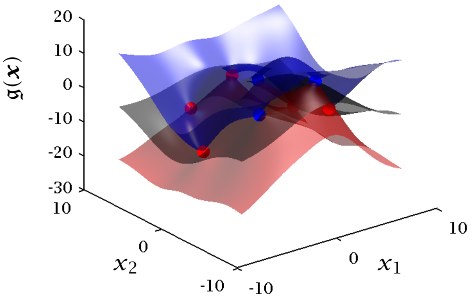
$$\hat{Y}(\boldsymbol{x}) \sim \mathcal{N}_1 \left(\mu_{\hat{Y}}(\boldsymbol{x}),\, \sigma^2_{\hat{Y}}(\boldsymbol{x})\right)$$

Interesting properties

- *interpolating*;
- *asymptotically consistent* (provided the correlation R is "compatible" with the data $\mathcal{Y}$ and the model $\mathcal{M}$);
  (Vazquez, 2005)
- *Gaussian* (consequence of the prior).

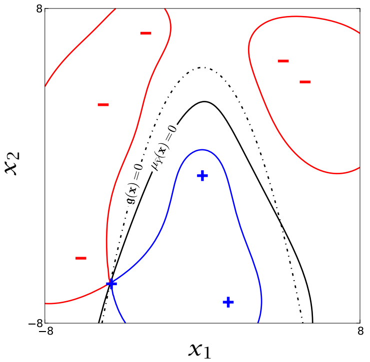
From regression to probabilistic classification

Ex: Let $\mathfrak{g}$ denote a quadratic limit-state function.

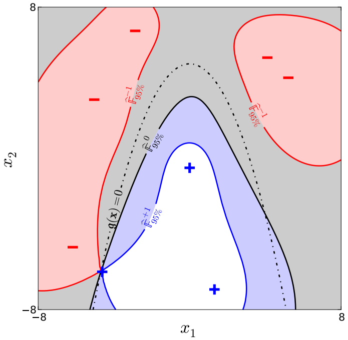


Regression

Classification ($\mathfrak{g} \leq 0$ *vs.* $\mathfrak{g} > 0$)



From regression to probabilistic classification



- Let $\widehat{\mathbb{F}}^{-1}_{95\%}$, $\widehat{\mathbb{F}}^{0}_{95\%}$, $\widehat{\mathbb{F}}^{+1}_{95\%}$ denote the three following *approximate failure subsets*:

$$\widehat{\mathbb{F}}^{i}_{95\%} = \left\{ \boldsymbol{x} \in \mathbb{X} : \mu_{\widehat{Y}}(\boldsymbol{x}) \leq i\, 1.96\, \sigma_{\widehat{Y}}(\boldsymbol{x}) \right\},$$

$$i = -1,\ 0,\ +1.$$

- In turns, this enables the definition of the *margin of uncertainty*:

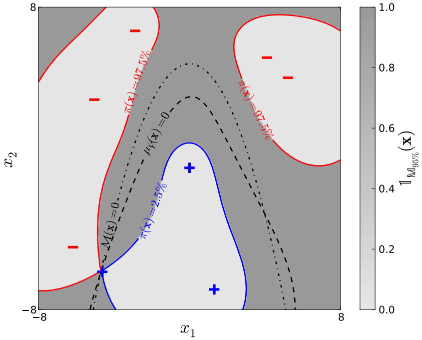
$$\mathbb{M}_{95\%} = \widehat{\mathbb{F}}^{+1}_{95\%} \setminus \widehat{\mathbb{F}}^{-1}_{95\%}$$

- Let π denote the *probabilistic classification function*:

$$\pi(\boldsymbol{x}) = \mathcal{P}\left[\widehat{Y}(\boldsymbol{x}) \leq 0\right] = \Phi\left(\frac{0 - \mu_{\widehat{Y}}(\boldsymbol{x})}{\sigma_{\widehat{Y}}(\boldsymbol{x})}\right)$$

$\mathcal{P}\ (\neq \mathbb{P})$ *denotes the probability measure w.r.t. the Kriging* epistemic *uncertainty.*

From regression to probabilistic classification



- Let $\hat{\mathbb{F}}^{-1}_{95\%}$, $\hat{\mathbb{F}}^{0}_{95\%}$, $\hat{\mathbb{F}}^{+1}_{95\%}$ denote the three following *approximate failure subsets*:

$$\hat{\mathbb{F}}^{i}_{95\%} = \left\{ \boldsymbol{x} \in \mathbb{X} : \mu_{\hat{Y}}(\boldsymbol{x}) \leq i\, 1.96\, \sigma_{\hat{Y}}(\boldsymbol{x}) \right\},$$

$$i = -1,\ 0,\ +1.$$

- In turns, this enables the definition of the *margin of uncertainty*:

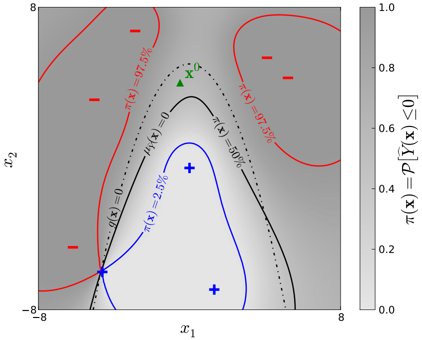
$$\mathbb{M}_{95\%} = \hat{\mathbb{F}}^{+1}_{95\%} \setminus \hat{\mathbb{F}}^{-1}_{95\%}$$

- Let π denote the *probabilistic classification function*:

$$\pi(\boldsymbol{x}) = \mathcal{P}\left[\hat{Y}(\boldsymbol{x}) \leq 0\right] = \Phi\left(\frac{0 - \mu_{\hat{Y}}(\boldsymbol{x})}{\sigma_{\hat{Y}}(\boldsymbol{x})}\right)$$

$\mathcal{P}\,(\neq \mathbb{P})$ *denotes the probability measure w.r.t. the Kriging* epistemic *uncertainty.*

From regression to probabilistic classification



- Let $\widehat{\mathbb{F}}^{-1}_{95\%}$, $\widehat{\mathbb{F}}^{0}_{95\%}$, $\widehat{\mathbb{F}}^{+1}_{95\%}$ denote the three following *approximate failure subsets*:

$$\widehat{\mathbb{F}}^{i}_{95\%} = \left\{\boldsymbol{x} \in \mathbb{X} : \mu_{\hat{Y}}(\boldsymbol{x}) \leq i\, 1.96\, \sigma_{\hat{Y}}(\boldsymbol{x})\right\},$$

$i = -1,\ 0,\ +1.$

- In turns, this enables the definition of the *margin of uncertainty*:

$$\mathbb{M}_{95\%} = \widehat{\mathbb{F}}^{+1}_{95\%} \setminus \widehat{\mathbb{F}}^{-1}_{95\%}$$

- Let π denote the *probabilistic classification function*:

$$\pi(\boldsymbol{x}) = \mathcal{P}\left[\hat{Y}(\boldsymbol{x}) \leq 0\right] = \Phi\left(\frac{0 - \mu_{\hat{Y}}(\boldsymbol{x})}{\sigma_{\hat{Y}}(\boldsymbol{x})}\right)$$

$\mathcal{P}\,(\neq \mathbb{P})$ *denotes the probability measure w.r.t. the Kriging* epistemic *uncertainty.*

Outline

Designs of experiments

Designs of experiments

- A DOE is the input part of a dataset:

$$\mathcal{X} = \left\{\boldsymbol{x}^{(i)},\, i = 1, \ldots, m\right\}$$

- Its *size m* must be minimized for the sake of *efficiency*.
- Experiments must be *selected carefully* for the sake of *accuracy* *(space-filling DOEs, Franco, 2008)*.

Adaptive designs of experiments

- are built in an *iterative* manner;
- on purpose to *refine the predictor locally* (*e.g.* in the vicinity of a contour);

Sequential adaptive DOEs for GP predictors rely on the *maximization* of a so-called *refinement criterion*.

Sequential adaptive DOEs

Refinement criteria for contour approximation

- Simple criteria mostly apply the *margin shrinking concept* for support vector machines

(Hurtado, 2004b; Deheeger, 2008)

- Here, we propose the "*margin probability*":

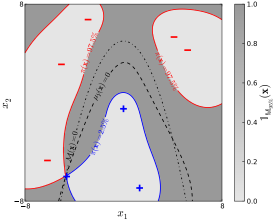
$$\mathcal{P}\left[\widehat{Y}(\boldsymbol{x}) \in \mathbb{M}_{95\%}\right] = \Phi\left(\frac{1.96\,\sigma_{\widehat{Y}}(\boldsymbol{x}) - \mu_{\widehat{Y}}(\boldsymbol{x})}{\sigma_{\widehat{Y}}(\boldsymbol{x})}\right) - \Phi\left(\frac{-1.96\,\sigma_{\widehat{Y}}(\boldsymbol{x}) - \mu_{\widehat{Y}}(\boldsymbol{x})}{\sigma_{\widehat{Y}}(\boldsymbol{x})}\right)$$

(Dubourg et al., 2010a)

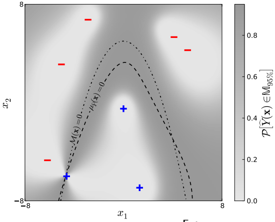
Limitation of sequential strategies

- The multiple modes of these criteria make their *maximization difficult*;
- There does not exist *a single best point*;
- Availability of *distributed computing* platforms for $\mathcal{M}$.

(Ginsbourger et al., 2010)



The margin of uncertainty $\mathbb{M}_{95\%}$



The margin probability $\mathcal{P}\left[\widehat{Y}(\boldsymbol{x}) \in \mathbb{M}_{95\%}\right]$

Sequential adaptive DOEs

Refinement criteria for contour approximation

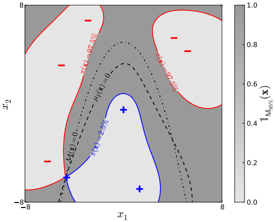
- Simple criteria mostly apply the *margin shrinking concept* for support vector machines

  (Hurtado, 2004b; Deheeger, 2008)

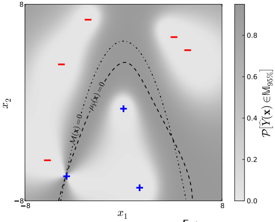
- Here, we propose the “*margin probability*”:

$$\mathcal{P}\left[\widehat{Y}(\boldsymbol{x}) \in \mathbb{M}_{95\%}\right] = \Phi\left(\frac{1.96\,\sigma_{\widehat{Y}}(\boldsymbol{x}) - \mu_{\widehat{Y}}(\boldsymbol{x})}{\sigma_{\widehat{Y}}(\boldsymbol{x})}\right) - \Phi\left(\frac{-1.96\,\sigma_{\widehat{Y}}(\boldsymbol{x}) - \mu_{\widehat{Y}}(\boldsymbol{x})}{\sigma_{\widehat{Y}}(\boldsymbol{x})}\right)$$

(Dubourg et al., 2010a)



The margin of uncertainty $\mathbb{M}_{95\%}$



The margin probability $\mathcal{P}\left[\widehat{Y}(\boldsymbol{x}) \in \mathbb{M}_{95\%}\right]$

Limitation of sequential strategies

- The multiple modes of these criteria make their *maximization difficult*;
- There does not exist *a single best point*;
- Availability of *distributed computing* platforms for $\mathcal{M}$.

  (Ginsbourger et al., 2010)

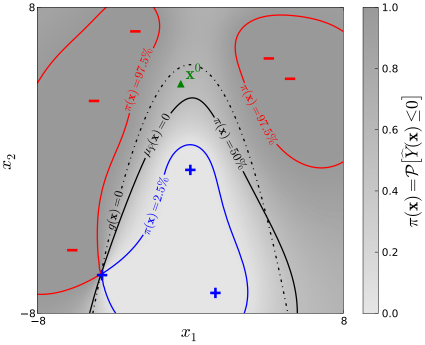
Sampling-based adaptive DOEs

Given an initial dataset $\mathcal{D}$
and a pseudo-PDF w:

1. Fit a Kriging predictor $\widehat{Y}(\boldsymbol{x})$
2. Define a *weighted refinement criterion*

$$C(\boldsymbol{x}) = \mathcal{P}\left[\widehat{Y}(\boldsymbol{x}) \in \mathbb{M}_{95\%}\right] w(\boldsymbol{x})$$

3. Sample N candidates from C
   (MCMC slice sampler, Neal, 2003)
4. Reduce the N candidates to K points
   (K-means clustering, Lloyd, 1982)
5. Enrich the dataset $\mathcal{D}$ with $\left\{\left(\boldsymbol{x}^{(m+k)}, \mathcal{M}\left(\boldsymbol{x}^{(m+k)}\right)\right), k = 1, \ldots, K\right\}$
6. Loop back to step 1



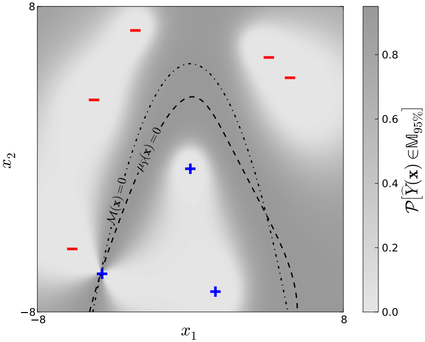
Sampling-based adaptive DOEs

Given an initial dataset $\mathcal{D}$
and a pseudo-PDF w:

1. Fit a Kriging predictor $\widehat{Y}(\boldsymbol{x})$
2. Define a *weighted refinement criterion*

$$C(\boldsymbol{x}) = \mathcal{P}\left[\widehat{Y}(\boldsymbol{x}) \in \mathbb{M}_{95\%}\right] w(\boldsymbol{x})$$

3. Sample N candidates from C
   (MCMC slice sampler, Neal, 2003)
4. Reduce the N candidates to K points
   (K-means clustering, Lloyd, 1982)
5. Enrich the dataset $\mathcal{D}$ with $\left\{\left(\boldsymbol{x}^{(m+k)}, \mathcal{M}\left(\boldsymbol{x}^{(m+k)}\right)\right), k = 1, \ldots, K\right\}$
6. Loop back to step 1



Sampling-based adaptive DOEs

Given an initial dataset $\mathcal{D}$
and a pseudo-PDF w:

1. Fit a Kriging predictor $\widehat{Y}(\boldsymbol{x})$
2. Define a *weighted refinement criterion*

$$C(\boldsymbol{x}) = \mathcal{P}\left[\widehat{Y}(\boldsymbol{x}) \in \mathbb{M}_{95\%}\right] w(\boldsymbol{x})$$

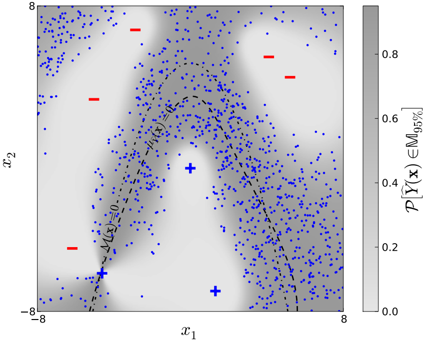
3. Sample N candidates from C

   (MCMC slice sampler, Neal, 2003)

4. Reduce the N candidates to K points

   (K-means clustering, Lloyd, 1982)

5. Enrich the dataset $\mathcal{D}$ with $\left\{\left(\boldsymbol{x}^{(m+k)}, \mathcal{M}\left(\boldsymbol{x}^{(m+k)}\right)\right), k = 1, \ldots, K\right\}$
6. Loop back to step 1



N is given (say $10,000$)

Sampling-based adaptive DOEs

Given an initial dataset $\mathcal{D}$
and a pseudo-PDF w:

1. Fit a Kriging predictor $\widehat{Y}(\boldsymbol{x})$
2. Define a *weighted refinement criterion*

$$C(\boldsymbol{x}) = \mathcal{P}\left[\widehat{Y}(\boldsymbol{x}) \in \mathbb{M}_{95\%}\right] w(\boldsymbol{x})$$

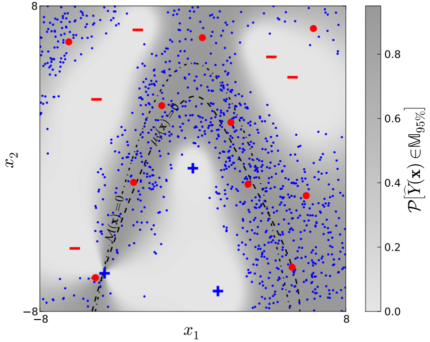
3. Sample N candidates from C

   (MCMC slice sampler, Neal, 2003)

4. Reduce the N candidates to K points

   (K-means clustering, Lloyd, 1982)

5. Enrich the dataset $\mathcal{D}$ with $\left\{\left(\boldsymbol{x}^{(m+k)}, \mathcal{M}\left(\boldsymbol{x}^{(m+k)}\right)\right), k = 1, \ldots, K\right\}$
6. Loop back to step 1



K is given *(say the number of CPUs)*

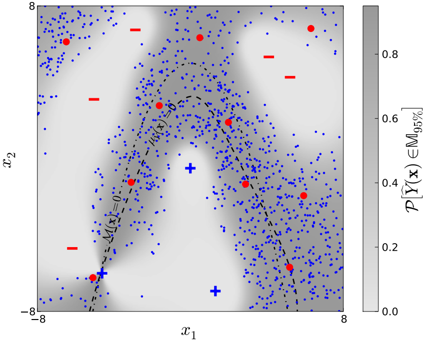
Sampling-based adaptive DOEs

Given an initial dataset $\mathcal{D}$
and a pseudo-PDF w:

1. Fit a Kriging predictor $\widehat{Y}(\boldsymbol{x})$
2. Define a *weighted refinement criterion*

$$C(\boldsymbol{x}) = \mathcal{P}\left[\widehat{Y}(\boldsymbol{x}) \in \mathbb{M}_{95\%}\right] w(\boldsymbol{x})$$

3. Sample N candidates from C
   (MCMC slice sampler, Neal, 2003)
4. Reduce the N candidates to K points
   (K-means clustering, Lloyd, 1982)
5. Enrich the dataset $\mathcal{D}$ with
   $\left\{\left(\boldsymbol{x}^{(m+k)}, \mathcal{M}\left(\boldsymbol{x}^{(m+k)}\right)\right), k = 1, \ldots, K\right\}$
6. Loop back to step 1



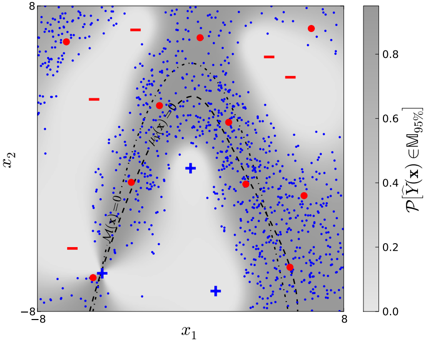
Sampling-based adaptive DOEs

Given an initial dataset $\mathcal{D}$
and a pseudo-PDF w:

1. Fit a Kriging predictor $\widehat{Y}(\boldsymbol{x})$
2. Define a *weighted refinement criterion*

$$C(\boldsymbol{x}) = \mathcal{P}\left[\widehat{Y}(\boldsymbol{x}) \in \mathbb{M}_{95\%}\right] w(\boldsymbol{x})$$

3. Sample N candidates from C
   (MCMC slice sampler, Neal, 2003)
4. Reduce the N candidates to K points
   (K-means clustering, Lloyd, 1982)
5. Enrich the dataset $\mathcal{D}$ with $\left\{\left(\boldsymbol{x}^{(m+k)}, \mathcal{M}\left(\boldsymbol{x}^{(m+k)}\right)\right), k = 1, \ldots, K\right\}$
6. Loop back to step 1

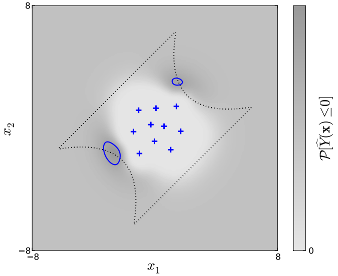


Illustration

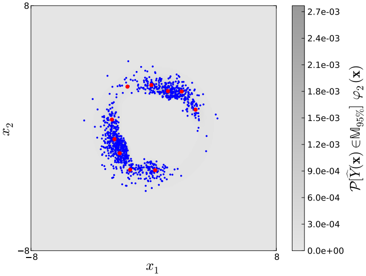
A four-branch series system

(Waarts, 2000)

$$C(\boldsymbol{x}) = \mathcal{P}\left[\widehat{Y}(\boldsymbol{x}) \in \mathbb{M}_{95\%}\right] \varphi_2(\boldsymbol{x}) \qquad K = 10$$



Target contour, design of experiments & prediction



Sampled & clustered refinement criterion

Iteration #1

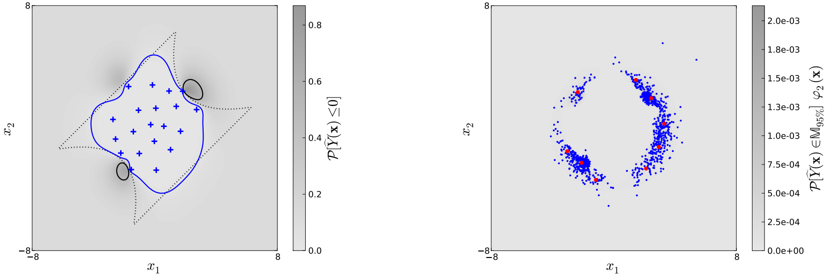
Convergence criteria depend on the *application*...

Illustration

A four-branch series system

(Waarts, 2000)

$$C(\boldsymbol{x}) = \mathcal{P}\left[\widehat{Y}(\boldsymbol{x}) \in \mathbb{M}_{95\%}\right] \varphi_2(\boldsymbol{x}) \qquad K = 10$$



Target contour, design of experiments & prediction

Sampled & clustered refinement criterion

Iteration #2

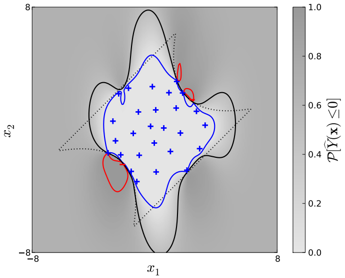
Convergence criteria depend on the *application*...

Illustration

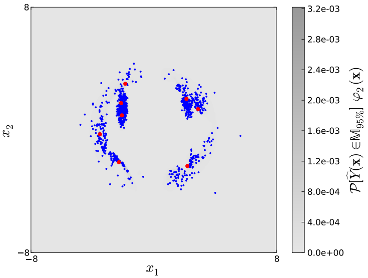
A four-branch series system

(Waarts, 2000)

$$C(\boldsymbol{x}) = \mathcal{P}\left[\hat{Y}(\boldsymbol{x}) \in \mathbb{M}_{95\%}\right] \varphi_2(\boldsymbol{x}) \qquad K = 10$$



Target contour, design of experiments & prediction



Sampled & clustered refinement criterion

Iteration #3

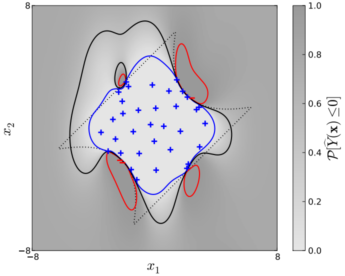
Convergence criteria depend on the *application*...

Illustration

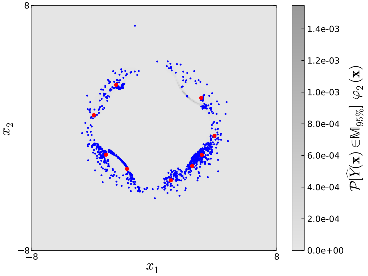
A four-branch series system

(Waarts, 2000)

$$C(\boldsymbol{x}) = \mathcal{P}\left[\widehat{Y}(\boldsymbol{x}) \in \mathbb{M}_{95\%}\right] \varphi_2(\boldsymbol{x}) \qquad K = 10$$



Target contour, design of experiments & prediction



Sampled & clustered refinement criterion

Iteration #4

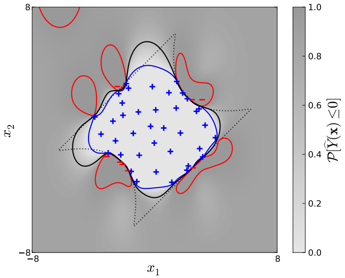
Convergence criteria depend on the *application*...

Illustration

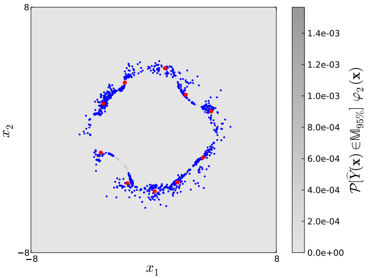
A four-branch series system

(Waarts, 2000)

$$C(\mathbf{x}) = \mathcal{P}\left[\widehat{Y}(\mathbf{x}) \in \mathbb{M}_{95\%}\right] \varphi_2(\mathbf{x}) \qquad K = 10$$



Target contour, design of experiments & prediction



Sampled & clustered refinement criterion

Iteration #5

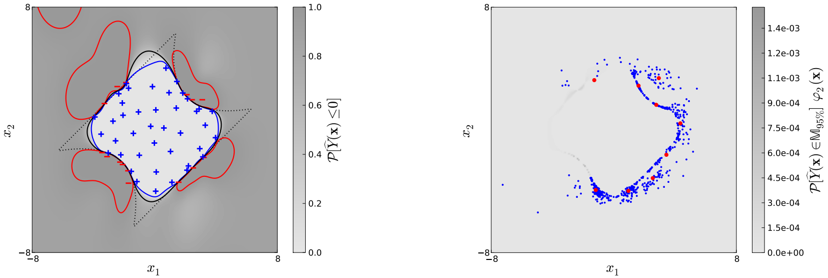
Convergence criteria depend on the *application*...

Illustration

A four-branch series system

(Waarts, 2000)

$$C(\boldsymbol{x}) = \mathcal{P}\left[\widehat{Y}(\boldsymbol{x}) \in \mathbb{M}_{95\%}\right]\,\varphi_2(\boldsymbol{x}) \qquad K = 10$$



Target contour, design of experiments & prediction

Sampled & clustered refinement criterion

Iteration #6

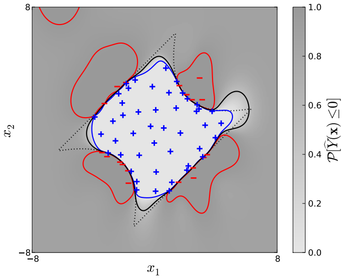
Convergence criteria depend on the *application*...

Illustration

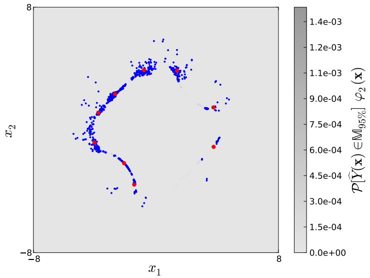
A four-branch series system

(Waarts, 2000)

$$C(\boldsymbol{x}) = \mathcal{P}\left[\hat{Y}(\boldsymbol{x}) \in \mathbb{M}_{95\%}\right] \varphi_2(\boldsymbol{x}) \qquad K = 10$$



Target contour, design of experiments & prediction



Sampled & clustered refinement criterion

Iteration #7

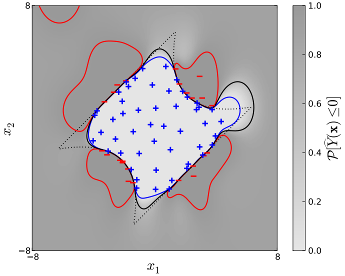
Convergence criteria depend on the *application*...

Illustration

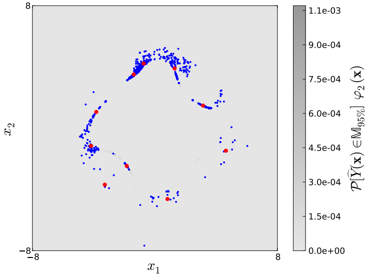
A four-branch series system

(Waarts, 2000)

$$C(\boldsymbol{x}) = \mathcal{P}\left[\widehat{Y}(\boldsymbol{x}) \in \mathbb{M}_{95\%}\right] \varphi_2(\boldsymbol{x}) \qquad K = 10$$



Target contour, design of experiments & prediction



Sampled & clustered refinement criterion

Iteration #8

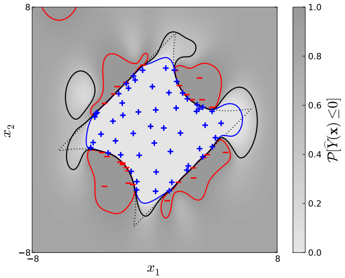
Convergence criteria depend on the *application*...

Illustration

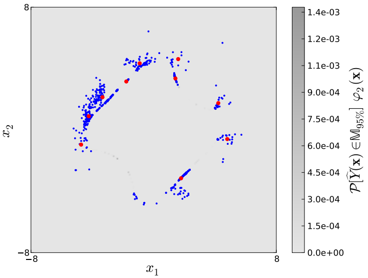
A four-branch series system

(Waarts, 2000)

$$C(\boldsymbol{x}) = \mathcal{P}\left[\widehat{Y}(\boldsymbol{x}) \in \mathbb{M}_{95\%}\right] \varphi_2(\boldsymbol{x}) \qquad K = 10$$



Target contour, design of experiments & prediction



Sampled & clustered refinement criterion

Iteration #9

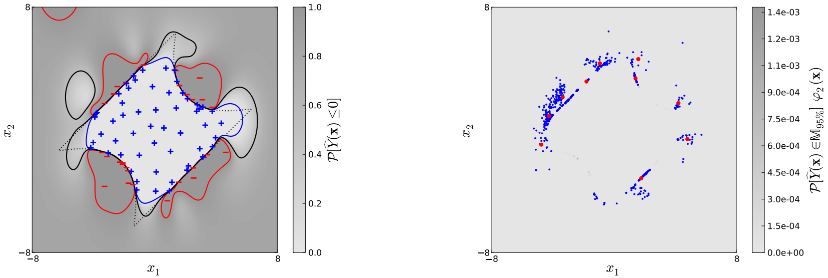
Convergence criteria depend on the *application*...

Illustration

A four-branch series system

(Waarts, 2000)

$$C(\boldsymbol{x}) = \mathcal{P}\left[\widehat{Y}(\boldsymbol{x}) \in \mathbb{M}_{95\%}\right] \varphi_2(\boldsymbol{x}) \qquad K = 10$$



Target contour, design of experiments & prediction

Sampled & clustered refinement criterion

Iteration #9

Convergence criteria depend on the *application*...

Outline

Reliability analysis
Introduction

(Ditlevsen & Madsen, 1996; Lemaire, 2009)

Problem formulation

- Given a *failure domain*:

$$\mathbb{F} = \{\boldsymbol{x} \in \mathbb{X} : \mathfrak{g}(\boldsymbol{x}) \leq 0\}$$

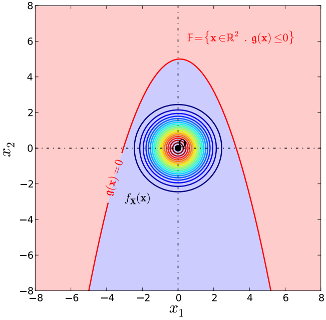
- and a *random vector* $\boldsymbol{X}$ with known distribution:

$$F_{\boldsymbol{X}}(\boldsymbol{x}) = C\left(F_{X_i}(x_i),\, i = 1, \ldots, n\right)$$

(Lebrun & Dutfoy, 2009a,b,c)

- the purpose is to quantify the reliability of a design in the form of a *failure probability*:

$$p_f = \mathbb{P}\left[\boldsymbol{X} \in \mathbb{F}\right] = \int_{\mathbb{F}} f_{\boldsymbol{X}}(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$$



$\mathbb{P}\,(\neq \mathcal{P})$ denotes the probability measure w.r.t. the random vector $\boldsymbol{X}$.

Reliability analysis

Monte Carlo sampling as a motivation for the *structural reliability methods*

Monte Carlo sampling

- The failure probability rewrites:

$$p_f = \int_{\mathbb{X}} \mathbb{1}_{\mathbb{F}}(\boldsymbol{x})\, f_{\boldsymbol{X}}(\boldsymbol{x})\, \mathrm{d}\boldsymbol{x} = \mathbb{E}\left[\mathbb{1}_{\mathbb{F}}(\boldsymbol{X})\right]$$

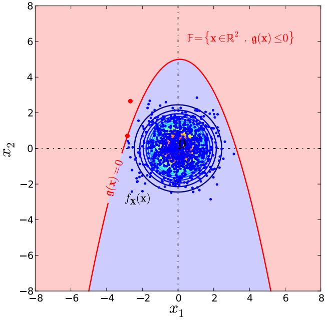
- Hence, the central limit theorem ensures that:

$$\widehat{p}_f = \frac{1}{N} \sum_{i=1}^{N} \mathbb{1}_{\mathbb{F}}\left(\boldsymbol{X}^{(i)}\right) \hookrightarrow \mathcal{N}_1\left(p_f,\, \frac{p_f\,(1-p_f)}{N}\right)$$

- *provided N is sufficiently large!*
- In order to involve $\widehat{p}_f$ in an optimization loop:

$$p_f \approx 10^{-k} \Rightarrow N \geq 10^{k+2}$$

- *Structural reliability methods* aim at reducing N



Surrogate-based reliability analysis

Principle

- A *surrogate-based* estimator:

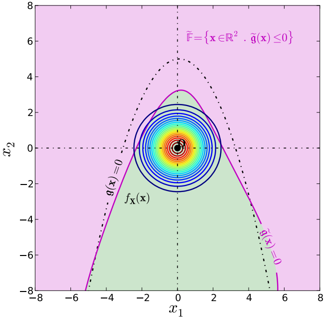
$$\tilde{p}_f = \int_{\tilde{\mathbb{F}}} f_{\boldsymbol{X}}(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$$

- where:

$$\tilde{\mathbb{F}} = \{\boldsymbol{x} \in \mathbb{X} : \tilde{\mathfrak{g}}(\boldsymbol{x}) \leq 0\} \approx \mathbb{F}$$

  and $\tilde{\mathfrak{g}}$ is a *meta-model* of $\mathfrak{g}$.
- $\tilde{\mathfrak{g}}$ is built from $m \ll N$ runs of $\mathfrak{g}$.

For the sake of *efficiency*
low probabilities can be handled by
subset sampling (Au & Beck, 2001)



Error (bias) quantification?

- Provided $\tilde{\mathfrak{g}}$ is a *Kriging predictor*:

$$\hat{\mathbb{F}}^{-1}_{95\%} \subseteq \hat{\mathbb{F}}^{0}_{95\%} \subseteq \hat{\mathbb{F}}^{+1}_{95\%} \Rightarrow p^{-1}_{f\,95\%} \leq p^{0}_{f\,95\%} \leq p^{+1}_{f\,95\%}$$

- Hence the following *empirical error*:

$$\Delta p_{f\,95\%} = \log_{10}\left(\frac{p^{+1}_{f\,95\%}}{p^{-1}_{f\,95\%}}\right)$$

$$\Delta p_{f\,95\%} \leq \Delta_0$$

Surrogate-based reliability analysis

Principle

- A *surrogate-based* estimator:

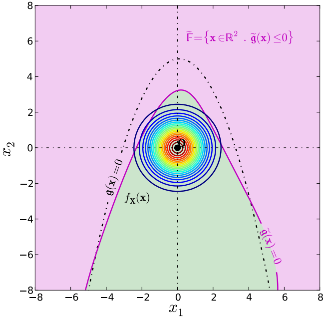
$$\tilde{p}_f = \int_{\tilde{\mathbb{F}}} f_{\boldsymbol{X}}(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$$

- where:

$$\tilde{\mathbb{F}} = \{\boldsymbol{x} \in \mathbb{X} : \tilde{\mathfrak{g}}(\boldsymbol{x}) \leq 0\} \approx \mathbb{F}$$

  and $\tilde{\mathfrak{g}}$ is a *meta-model* of $\mathfrak{g}$.
- $\tilde{\mathfrak{g}}$ is built from $m \ll N$ runs of $\mathfrak{g}$.

For the sake of *efficiency*
low probabilities can be handled by
subset sampling (Au & Beck, 2001)



Error (bias) quantification?

- Provided $\tilde{\mathfrak{g}}$ is a *Kriging predictor*:

$$\hat{\mathbb{F}}^{-1}_{95\%} \subseteq \hat{\mathbb{F}}^{0}_{95\%} \subseteq \hat{\mathbb{F}}^{+1}_{95\%} \Rightarrow p^{-1}_{f\,95\%} \leq p^{0}_{f\,95\%} \leq p^{+1}_{f\,95\%}$$

- Hence the following *empirical error*:

$$\Delta p_{f\,95\%} = \log_{10}\left(\frac{p^{+1}_{f\,95\%}}{p^{-1}_{f\,95\%}}\right)$$

$$\Delta p_{f\,95\%} \leq \Delta_0$$

Surrogate-based reliability analysis

Principle

- A *surrogate-based* estimator:

$$\widetilde{p}_f = \int_{\widetilde{\mathbb{F}}} f_{\boldsymbol{X}}(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$$

- where:

$$\widetilde{\mathbb{F}} = \{\boldsymbol{x} \in \mathbb{X} : \widetilde{\mathfrak{g}}(\boldsymbol{x}) \leq 0\} \approx \mathbb{F}$$

  and $\widetilde{\mathfrak{g}}$ is a *meta-model* of $\mathfrak{g}$.
- $\widetilde{\mathfrak{g}}$ is built from $m \ll N$ runs of $\mathfrak{g}$.

Error (bias) quantification?

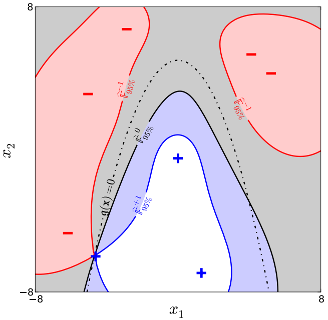
- Provided $\widetilde{\mathfrak{g}}$ is a *Kriging predictor*:

$$\widehat{\mathbb{F}}^{-1}_{95\%} \subseteq \widehat{\mathbb{F}}^{0}_{95\%} \subseteq \widehat{\mathbb{F}}^{+1}_{95\%} \Rightarrow p^{-1}_{f\,95\%} \leq p^{0}_{f\,95\%} \leq p^{+1}_{f\,95\%}$$

- Hence the following *empirical error*:

$$\Delta p_{f\,95\%} = \log_{10}\left(\frac{p^{+1}_{f\,95\%}}{p^{-1}_{f\,95\%}}\right)$$

For the sake of *efficiency*
low probabilities can be handled by
subset sampling *(Au & Beck, 2001)*



$\Delta p_{f\,95\%} \leq \Delta_0$

Surrogate-based reliability analysis

Principle

- A *surrogate-based* estimator:

$$\widetilde{p}_f = \int_{\widetilde{\mathbb{F}}} f_{\boldsymbol{X}}(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$$

- where:

$$\widetilde{\mathbb{F}} = \{\boldsymbol{x} \in \mathbb{X} : \widetilde{\mathfrak{g}}(\boldsymbol{x}) \leq 0\} \approx \mathbb{F}$$

  and $\widetilde{\mathfrak{g}}$ is a *meta-model* of $\mathfrak{g}$.
- $\widetilde{\mathfrak{g}}$ is built from $m \ll N$ runs of $\mathfrak{g}$.

Error (bias) quantification?

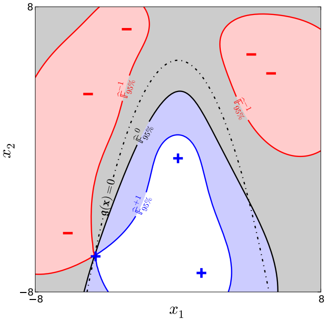
- Provided $\widetilde{\mathfrak{g}}$ is a *Kriging predictor*:

$$\widehat{\mathbb{F}}^{-1}_{95\%} \subseteq \widehat{\mathbb{F}}^{0}_{95\%} \subseteq \widehat{\mathbb{F}}^{+1}_{95\%} \Rightarrow p^{-1}_{f\,95\%} \leq p^{0}_{f\,95\%} \leq p^{+1}_{f\,95\%}$$

- Hence the following *empirical error*:

$$\Delta p_{f\,95\%} = \log_{10}\left(\frac{p^{+1}_{f\,95\%}}{p^{-1}_{f\,95\%}}\right)$$

For the sake of *efficiency*
low probabilities can be handled by
subset sampling *(Au & Beck, 2001)*



$$\Delta p_{f\,95\%} \leq \Delta_0$$

Meta-model-based importance sampling

Importance sampling

(Rubinstein & Kroese, 1981, 2008)

Principle

- **Premise:** $f_{\boldsymbol{X}}(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$ is *inappropriate*!
- Given an admissible *instrumental PDF* h, the failure probability rewrites:

$$p_f = \int_{\{\boldsymbol{x}\in\mathbb{X}:\, h(\boldsymbol{x})>0\}} \mathbb{1}_{\mathbb{F}}(\boldsymbol{x})\, \frac{f_{\boldsymbol{X}}(\boldsymbol{x})}{h(\boldsymbol{x})}\, h(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$$

$$= \mathbb{E}_{\boldsymbol{Z}}\left[\mathbb{1}_{\mathbb{F}}(\boldsymbol{Z})\,\frac{f_{\boldsymbol{X}}(\boldsymbol{Z})}{h(\boldsymbol{Z})}\right]$$

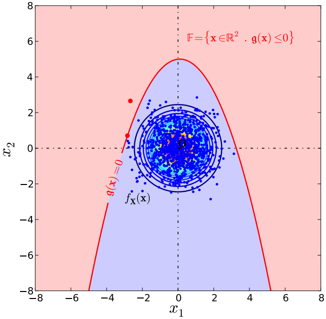
where $\boldsymbol{Z} \sim h$.

Optimal importance sampling

$$h^*(\boldsymbol{x}) = \frac{\mathbb{1}_{\mathbb{F}}(\boldsymbol{x})\, f_{\boldsymbol{X}}(\boldsymbol{x})}{}$$

reduces the variance of estimation to zero!

h^* is impracticable!



Find another h close to h^*

Meta-model-based importance sampling

Importance sampling

(Rubinstein & Kroese, 1981, 2008)

Principle

- **Premise:** $f_{\mathbf{X}}(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$ is *inappropriate*!
- Given an admissible *instrumental PDF* h, the failure probability rewrites:

$$p_f = \int_{\{\boldsymbol{x}\in\mathbb{X}:\, h(\boldsymbol{x})>0\}} \mathbb{1}_{\mathbb{F}}(\boldsymbol{x})\,\frac{f_{\mathbf{X}}(\boldsymbol{x})}{h(\boldsymbol{x})}\,h(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$$
$$= \mathbb{E}_{\boldsymbol{Z}}\left[\mathbb{1}_{\mathbb{F}}(\boldsymbol{Z})\,\frac{f_{\mathbf{X}}(\boldsymbol{Z})}{h(\boldsymbol{Z})}\right]$$

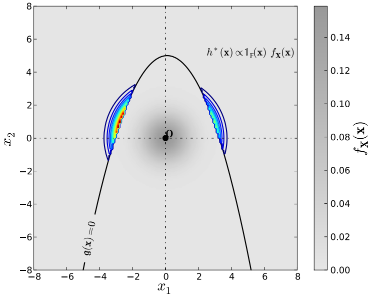
where $\boldsymbol{Z} \sim h$.

Optimal importance sampling

$$h^*(\boldsymbol{x}) = \frac{\mathbb{1}_{\mathbb{F}}(\boldsymbol{x})\,f_{\mathbf{X}}(\boldsymbol{x})}{p_f}$$

reduces the variance of estimation to zero!

h^* is impracticable!



Find another h close to h^*

Meta-model-based importance sampling

Importance sampling

(Rubinstein & Kroese, 1981, 2008)

Principle

- **Premise:** $f_{\mathbf{X}}(\mathbf{x})\,\mathrm{d}\mathbf{x}$ is *inappropriate*!
- Given an admissible *instrumental PDF* h, the failure probability rewrites:

$$p_f = \int_{\{\mathbf{x}\in\mathbb{X}:\, h(\mathbf{x})>0\}} \mathbb{1}_{\mathbb{F}}(\mathbf{x})\,\frac{f_{\mathbf{X}}(\mathbf{x})}{h(\mathbf{x})}\,h(\mathbf{x})\,\mathrm{d}\mathbf{x} = \mathbb{E}_{\mathbf{Z}}\left[\mathbb{1}_{\mathbb{F}}(\mathbf{Z})\,\frac{f_{\mathbf{X}}(\mathbf{Z})}{h(\mathbf{Z})}\right]$$

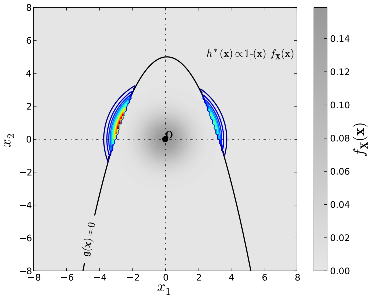
where $\mathbf{Z} \sim h$.

Optimal importance sampling

$$h^*(\mathbf{x}) = \frac{\mathbb{1}_{\mathbb{F}}(\mathbf{x})\,f_{\mathbf{X}}(\mathbf{x})}{p_f}$$

reduces the variance of estimation to zero!

h^* is impracticable!



Find another h close to h^*

Meta-model-based importance sampling
Approximation of the optimal instrumental PDF

- Given the *probabilistic classification function* of a Kriging predictor:

$$\pi(\boldsymbol{x}) = \mathcal{P}\left[\widehat{Y}(\boldsymbol{x}) \leq 0\right] = \Phi\left(\frac{0 - \mu_{\widehat{Y}}(\boldsymbol{x})}{\sigma_{\widehat{Y}}(\boldsymbol{x})}\right)$$

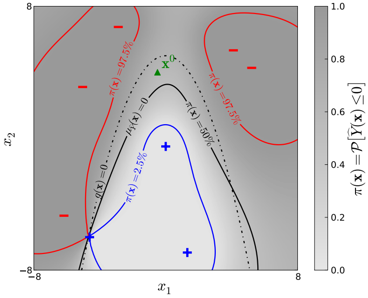
- We propose the following Kriging-based approximation:

$$\widehat{h}^*(\boldsymbol{x}) = \frac{\pi(\boldsymbol{x})\, f_{\boldsymbol{X}}(\boldsymbol{x})}{p_{f\,\varepsilon}}$$

where:

$$p_{f\,\varepsilon} = \int_{\mathbb{X}} \pi(\boldsymbol{x})\, f_{\boldsymbol{X}}(\boldsymbol{x})\, \mathrm{d}\boldsymbol{x} = \mathbb{E}\left[\pi(\boldsymbol{X})\right]$$

is the *augmented failure probability* (“$\mathcal{P} + \mathbb{P}$”).



Meta-model-based importance sampling

Approximation of the optimal instrumental PDF

- Given the *probabilistic classification function* of a Kriging predictor:

$$\pi(\boldsymbol{x}) = \mathcal{P}\left[\widehat{Y}(\boldsymbol{x}) \leq 0\right] = \Phi\left(\frac{0 - \mu_{\widehat{Y}}(\boldsymbol{x})}{\sigma_{\widehat{Y}}(\boldsymbol{x})}\right)$$

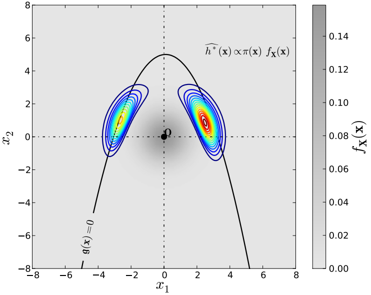
- We propose the following Kriging-based approximation:

$$\widehat{h}^*(\boldsymbol{x}) = \frac{\pi(\boldsymbol{x})\, f_{\boldsymbol{X}}(\boldsymbol{x})}{p_{f\,\varepsilon}}$$

where:

$$p_{f\,\varepsilon} = \int_{\mathbb{X}} \pi(\boldsymbol{x})\, f_{\boldsymbol{X}}(\boldsymbol{x})\, \mathrm{d}\boldsymbol{x} = \mathbb{E}\left[\pi(\boldsymbol{X})\right]$$

is the *augmented failure probability* (“$\mathcal{P} + \mathbb{P}$”).



Meta-model-based importance sampling

Proposed estimator

- Substituting $\widehat{h^*}$ for h, it turns out that:

$$p_f = p_{f\,\varepsilon}\,\alpha_{\text{corr}}$$

where:

$$\alpha_{\text{corr}} = \mathbb{E}_Z\left[\frac{\mathbb{1}_{\mathbb{F}}(Z)}{\pi(Z)}\right]$$

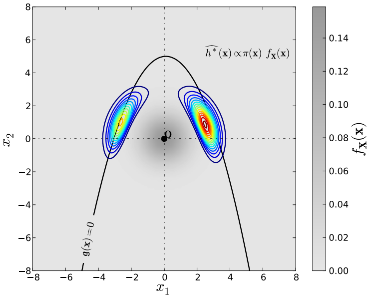
is the *correction factor*, with $Z \sim \widehat{h^*}$.

- Optimal importance sampling can be reached as the "*control variate*":

$$\pi(X) \hookrightarrow \mathbb{1}_{\mathbb{F}}(X)$$

meaning that:

$$\begin{cases} p_{f\,\varepsilon} \longrightarrow p_f \\ \alpha_{\text{corr}} \longrightarrow 1 \end{cases}$$



Meta-model-based importance sampling
Proposed estimator

- Substituting $\widehat{h^*}$ for h, it turns out that:

$$p_f = p_{f\,\varepsilon}\,\alpha_{\text{corr}}$$

where:

$$\alpha_{\text{corr}} = \mathbb{E}_{Z}\left[\frac{\mathbb{1}_{\mathbb{F}}(Z)}{\pi(Z)}\right]$$

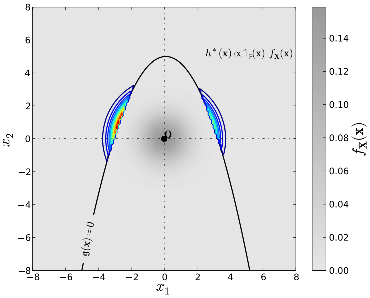
is the *correction factor*, with $Z \sim \widehat{h^*}$.

- Optimal importance sampling can be reached as the "*control variate*":

$$\pi(X) \hookrightarrow \mathbb{1}_{\mathbb{F}}(X)$$

meaning that:

$$\begin{cases} p_{f\,\varepsilon} \longrightarrow p_f \\ \alpha_{\text{corr}} \longrightarrow 1 \end{cases}$$



Meta-model-based importance sampling
Proposed estimator (cont')

- Eventually, the failure probability can be computed as:

$$\widehat{p}_{f\,\text{metaIS}} = \widehat{p}_{f\,\varepsilon}\,\widehat{\alpha}_{\text{corr}}$$

where:

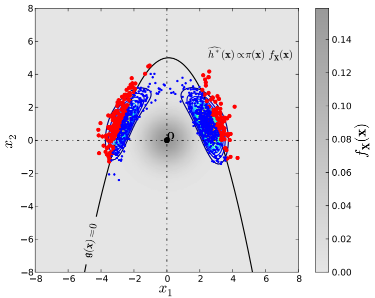
$$\widehat{p}_{f\,\varepsilon} = \frac{1}{N_\varepsilon}\sum_{i=1}^{N_\varepsilon} \pi\left(\boldsymbol{X}^{(i)}\right)$$

$$\widehat{\alpha}_{\text{corr}} = \frac{1}{N_{\text{corr}}}\sum_{i=1}^{N_{\text{corr}}} \frac{\mathbb{1}_{\mathbb{F}}\left(\boldsymbol{Z}^{(i)}\right)}{\pi\left(\boldsymbol{Z}^{(i)}\right)}$$

- The final coefficient of variation is:

$$\delta_{\text{metaIS}} = \sqrt{{\delta_\varepsilon}^2 + {\delta_{\text{corr}}}^2 + {\delta_\varepsilon}^2\,{\delta_{\text{corr}}}^2} \underset{\delta_\varepsilon,\,\delta_{\text{corr}} \ll 1}{\approx} \sqrt{{\delta_\varepsilon}^2 + {\delta_{\text{corr}}}^2}$$

Sampling from $\widehat{h^*}$ resorts to MCMC
(Robert & Casella, 2004)



Meta-model-based importance sampling

Trade-off between m and N_{corr}

- Optimality is reached if:

$$\alpha_{\text{corr}} = 1$$

- But α_{corr} is *expensive to evaluate* so that it should be estimated only *once*!
- Hence, we propose the following *leave-one-out estimate*:

$$\widehat{\alpha}_{\text{corr LOO}} = \frac{1}{m} \sum_{i=1}^{m} \frac{\mathbb{1}_{\mathbb{F}}\left(\boldsymbol{x}^{(i)}\right)}{\mathcal{P}\left[Y\left(\boldsymbol{x}^{(i)}\right) \leq 0 \mid \boldsymbol{Y}_{-i} = \boldsymbol{y}_{-i}\right]}$$

where $\boldsymbol{y}_{-i}$ denotes all the observations in the dataset $\mathcal{D}$ *but the i-th one*.

- The following condition is used to *stop* the sampling-based adaptive enrichment of the dataset $\mathcal{D}$ at the k-th iteration:

$$\left|\widehat{\alpha}_{\text{corr LOO}}^{(k)} - 1\right| \leq \epsilon_\alpha^1 \quad \textbf{and} \quad \frac{\left|\widehat{\alpha}_{\text{corr LOO}}^{(k)} - \widehat{\alpha}_{\text{corr LOO}}^{(k-1)}\right|}{\widehat{\alpha}_{\text{corr LOO}}^{(k-1)}} \leq \epsilon_\alpha^2 \quad \textbf{or} \quad m > m_{\max}$$

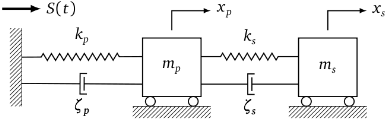
- Then, the true value of α_{corr} is estimated by $\widehat{\alpha}_{\text{corr}}$.

Illustration #1

Influence of the failure probability *(Bourinet et al., 2011)*

A two d.o.f. damped oscillator

- Let us consider the following *seismic control device*:



where S is a stationary Gaussian white noise (*ground motion*).

- The limit-state function for the *secondary spring* is:

$$\mathfrak{g}(\boldsymbol{x}) = F_s - k_s \max_{t \in [0;T]} |x_s(t)|$$

(Igusa & Der Kiureghian, 1985)

Probabilistic model (8 RVs)

Variable	Distribution	Mean	C.o.V.
m_p	Lognormal	1.5	10%
m_s	Lognormal	0.01	10%
k_p	Lognormal	1	20%
k_s	Lognormal	0.01	20%
ζ_p	Lognormal	0.05	40%
ζ_s	Lognormal	0.02	50%
F_S	Lognormal	$\{15, 21.5, 27.5\}$	10%
S_0	Lognormal	100	10%

(Der Kiureghian & De Stefano, 1990)

Illustration #1

Influence of the failure probability (cont') *(Bourinet* et al.*, 2011)*

Results

μ_{F_S}	values	FORM[a]	Subset Sampling	Meta-IS	SVM + Subset[a]
15	N	1,179	300,000	**464 + 200**	**1,719**
	p_f	2.19×10^{-2}	4.63×10^{-3}	4.80×10^{-3}	4.78×10^{-3}
	C.o.V.	–	<3%	<5%	<4%
21.5	N	2,520	500,000	**336 + 400**	**2,865**
	p_f	3.50×10^{-4}	4.75×10^{-5}	4.46×10^{-5}	4.42×10^{-5}
	C.o.V.	–	<4%	<5%	<7%
27.5	N	2,727	700,000	**480 + 200**	**4,011**
	p_f	3.91×10^{-6}	3.47×10^{-7}	3.76×10^{-7}	3.66×10^{-7}
	C.o.V.	–	<5%	<5%	<10%

[a] As computed by *Bourinet* et al.*(2011)*.

Chosen cuts of the failure domain

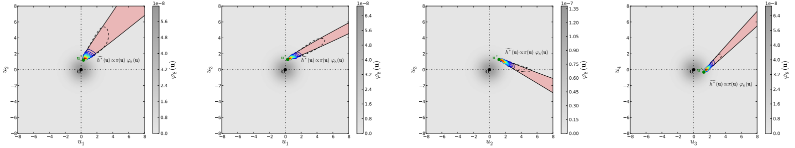


Illustration #2

Influence of the dimension n *(Rackwitz, 2001)*

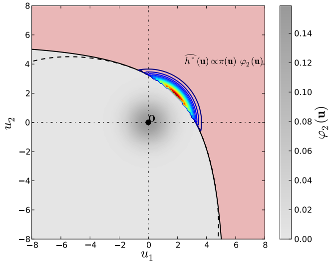
Problem formulation

- The limit-state function reads:

$$\mathfrak{g}(\boldsymbol{x}) = (n + 0.6\sqrt{n}) - \sum_{i=1}^{n} x_i$$

- The probabilistic model is:

$$\boldsymbol{X} \sim \mathcal{LN}\left(\mathbf{1},\, 0.2\,\mathbf{Id}_n\right)$$



Results

n	2	50	100
Crude Monte Carlo sampling (ref.)			
$\hat{p}_{f\,\text{MC}}$	4.78×10^{-3}	1.91×10^{-3}	1.73×10^{-3}
δ_{MC}	$\leq 2\%$	$\leq 2\%$	$\leq 2\%$
N	522,000	1,100,000	1,450,000
Metamodel-based importance sampling			
m	6×2	6×50	6×100
$\hat{p}_{f\,\varepsilon}$	4.85×10^{-3}	1.95×10^{-3}	1.83×10^{-3}
δ_{ε}	$\leq 1.41\%$	$\leq 1.41\%$	$\leq 1.41\%$
N_{corr}	**100**	**1,500**	**2,100**
$\hat{\alpha}_{\text{corr}}$	**1.00**	**0.99**	**0.93**
δ_{corr}	0%	$\leq 1.41\%$	$\leq 1.41\%$
$m + N_{\text{corr}}$	**112**	**1,800**	**2,700**
$\hat{p}_{f\,\text{metaIS}}$	4.85×10^{-3}	1.93×10^{-3}	1.70×10^{-3}
δ_{metaIS}	$\leq 1.41\%$	$\leq 2\%$	$\leq 2\%$

Outline

Reliability-based design optimization

Introduction

(*Tsompanakis* et al., *2008*)

Problem formulation

$$\boldsymbol{d}^* = \arg\min_{\boldsymbol{d}\in\mathbb{D}} \quad \mathfrak{c}(\boldsymbol{d}) : \quad \begin{cases} \mathfrak{f}_i(\boldsymbol{d}) \leq 0, & i = 1, \ldots, n_c \\ p_{f\,l}(\boldsymbol{d}) \leq p^0_{f\,l}, & l = 1, \ldots, n_p \end{cases}$$

where $\boldsymbol{d}$ is exclusively involved in the definition of the random vector $\boldsymbol{X}$ (*e.g. mean values*).

Bottlenecks

- The *repeated* reliability estimations are *computationally expensive*;
- Most NLP constrained optimization algorithms require *the gradients of the failure probabilities*.

Solutions

- Nested approaches (*Enevoldsen & Sørensen, 1994*)
- Sequential approaches (*Du & Chen, 2004*)
- Surrogate-based approaches (*Eldred* et al., *2002*)

Surrogate-based RBDO
The augmented reliability space

(Taflanidis & Beck, 2008, 2009a,b)

Motivation

Building the Kriging surrogates *from scratch* for each nested reliability analysis would be particularly inefficient.

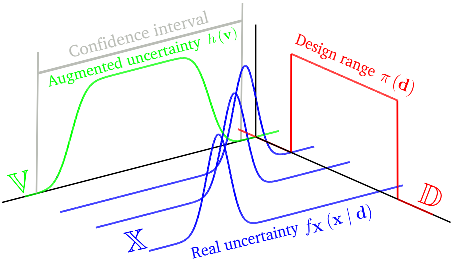
Definition

- The admissible range $\mathbb{D}$ simply *augments the spread* of f_X:

$$h(\boldsymbol{x}) = \int_{\mathbb{D}} f_X(\boldsymbol{x} \mid \boldsymbol{d}) \, \pi(\boldsymbol{d}) \, \mathrm{d}\boldsymbol{d}$$

  where π is the uniform distribution over $\mathbb{D}$.

- The idea is to work on a *sufficiently large confidence region* of h.



Surrogate-based RBDO
Reliability sensitivity analysis

Motivations

- NLP optimization algorithms require *the gradient of the failure probabilities*;
- How to compute these derivatives with *Monte Carlo* techniques?

The score function approach *(Rubinstein, 1976, 1986)*

Given a random vector $\boldsymbol{X}$ with parameter d, *provided its support $\mathbb{X}$ does not depend on d*:

$$\frac{\partial p_f(d)}{\partial d} = \mathbb{E}_{\boldsymbol{X}}\left[\mathbb{1}_{\mathbb{F}}(\boldsymbol{X})\,\frac{\partial \log f_{\boldsymbol{X}}(\boldsymbol{X} \mid d)}{\partial d}\right]$$

Interesting properties

- *A simple post-processing* of a reliability analysis!
- The **score function** comes *analytically* when the copula formalism is used.
  (Lee et al., 2011a,b)
- The approach extends to *reduction variance techniques* such as:
  - subset sampling *(Song et al., 2009)*
  - (meta-model-based) importance sampling.

Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

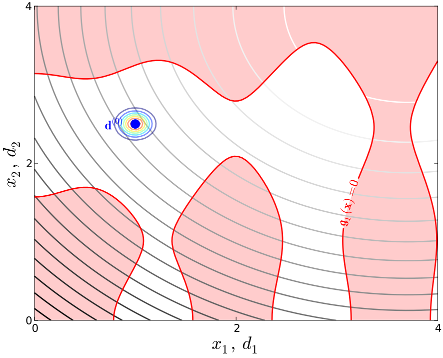
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

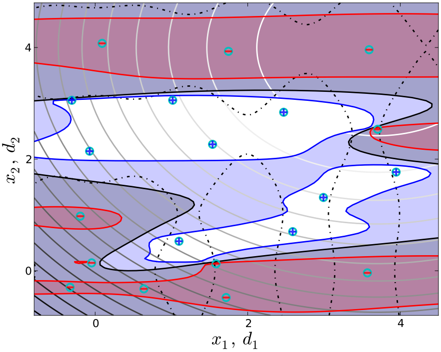
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

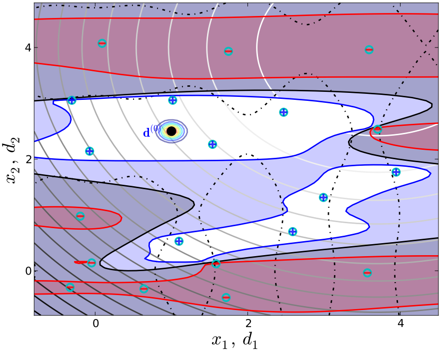
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

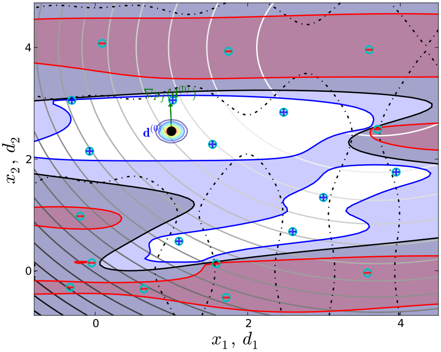
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

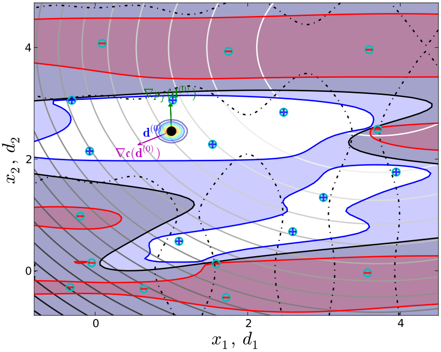
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

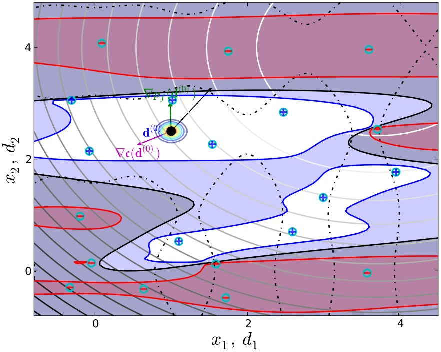
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

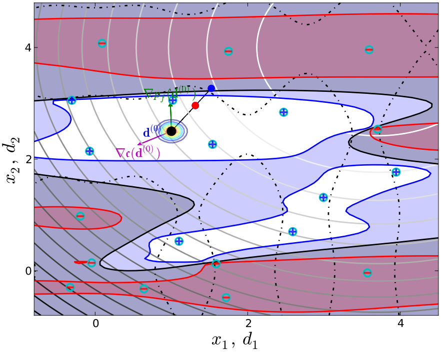
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

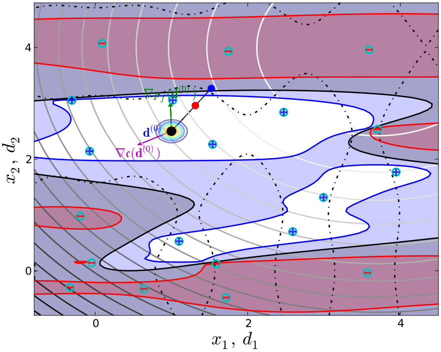
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

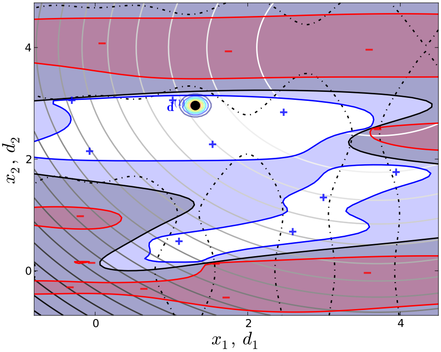
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

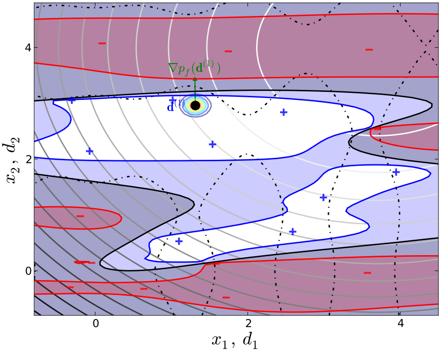
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

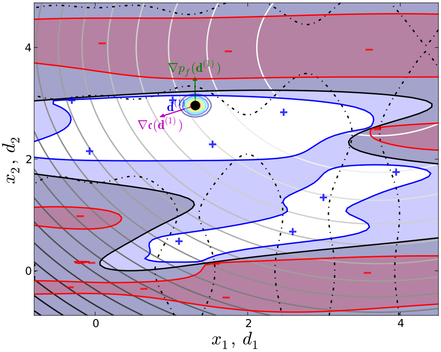
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

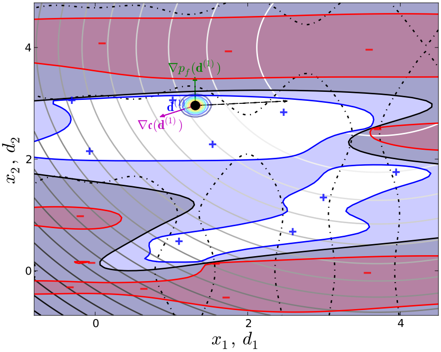
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

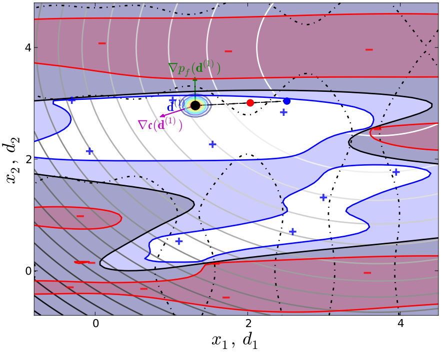
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

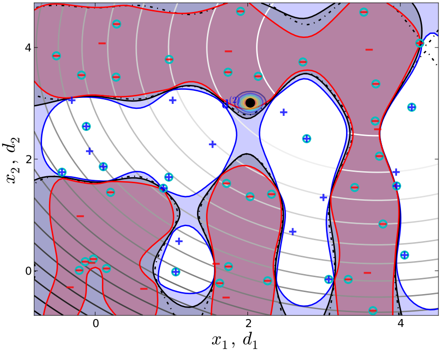
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\widehat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

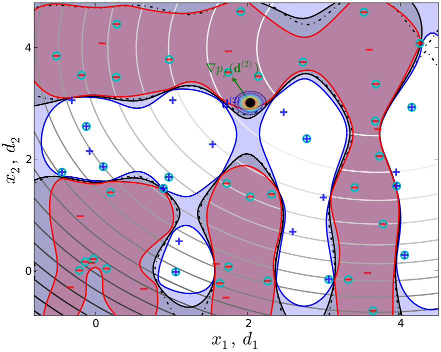
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

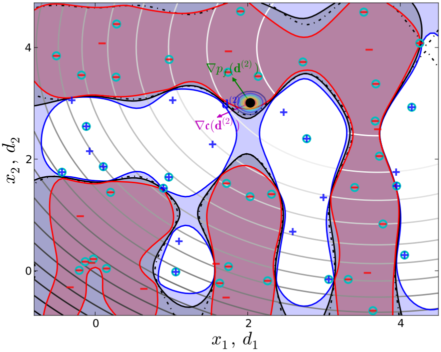
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO
Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

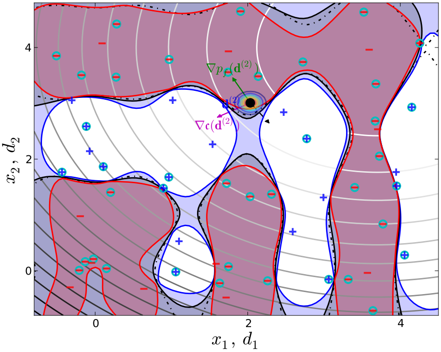
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\widehat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

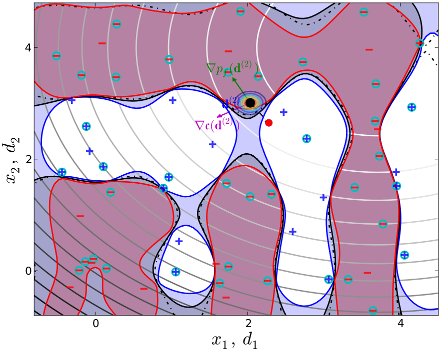
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

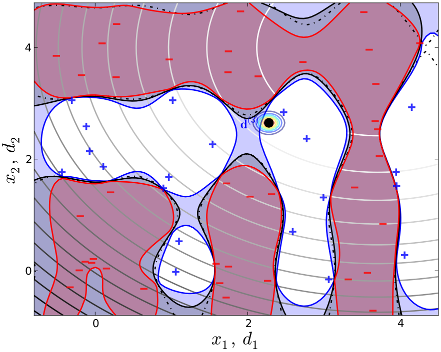
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

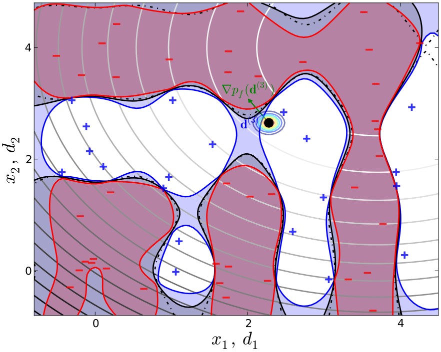
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

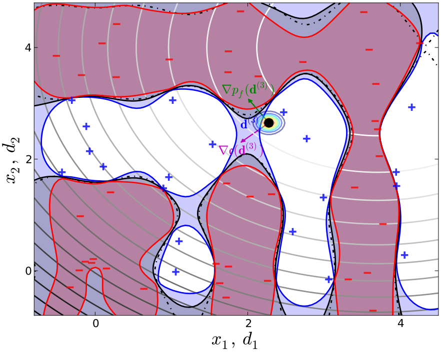
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

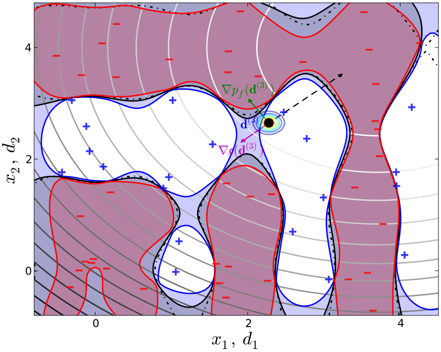
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

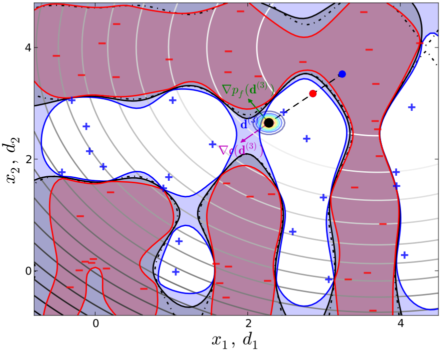
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

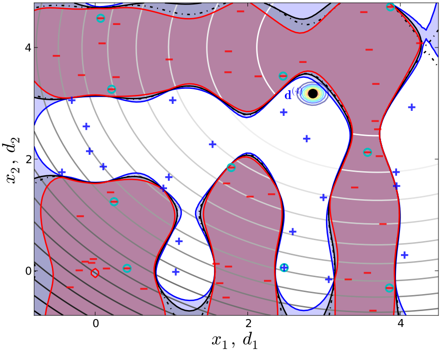
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\widehat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

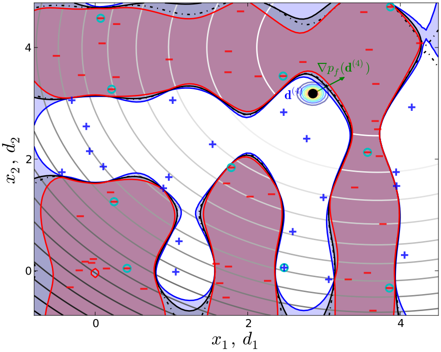
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

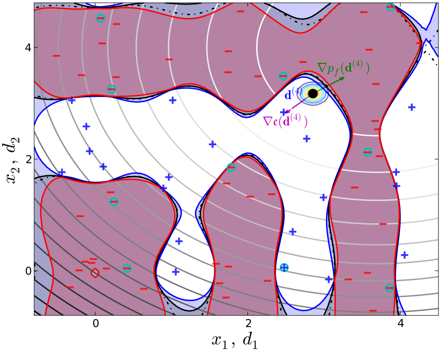
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

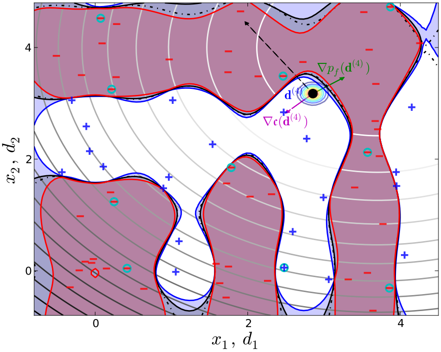
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Surrogate-based RBDO
Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

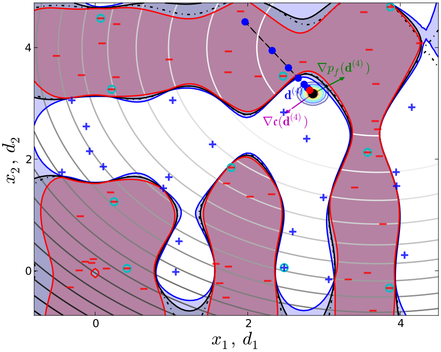
$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.

Optimizer converged!



Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

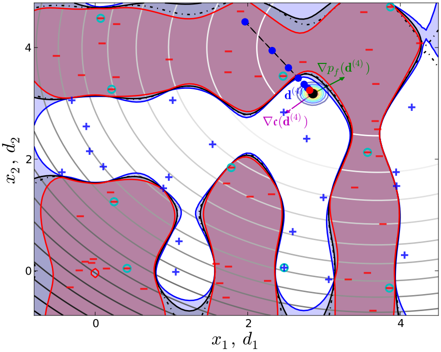
$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.

Optimizer converged!



Shrink $\mathbb{D}$ and decrease Δ_0

Surrogate-based RBDO

Overview of the proposed algorithm

Given an initial design $\boldsymbol{d}^{(0)} \in \mathbb{D}$ (*bounded*):

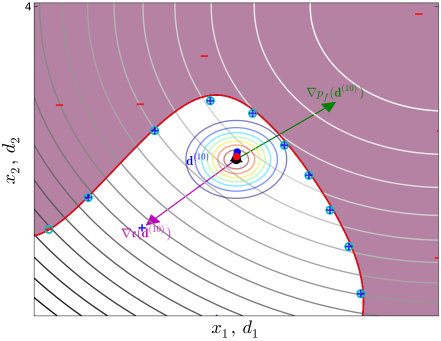
1. Determine the *augmented reliability space*;
2. Fit an *adaptive Kriging surrogate* with target local accuracy:

$$\Delta p_{f\,95\%}\left(\boldsymbol{d}^{(i)}\right) \leq \Delta_0$$

3. Compute $\hat{p}^{\,0}_{f\,95\%}(\boldsymbol{d}^{(i)})$ and *its gradient*;
4. Compute *search direction* $\boldsymbol{h}^{(i)}$ using min-max formulation;
5. Perform *approximate line-search* using Goldstein-Armijo step size rule;

$$\boldsymbol{d}^{(i+1)} = \boldsymbol{d}^{(i)} + s^{(i)}\,\boldsymbol{h}^{(i)}$$

6. Loop back to step 2 until the *optimizer converges*.



Converged!

Validation

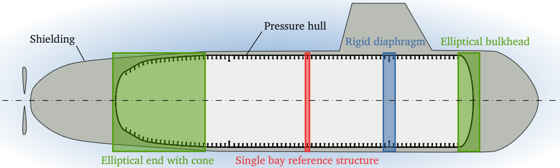
The approach has been *validated* over a chosen set of *6 academic examples*.

Example	n	n_d	n_p	N	Features
Euler buckling of a straight column	3	2	1	20	Reference analytical solution
A highly nonlinear limit-state *(Lee & Jung, 2008)*	2	2	2	80/10	Strong nonlinearity
Three nonlinear limit-states *(Shan & Wang, 2008)*	2	2	3	20/10/10	Multiple limit-states
A short column under oblique bending *(Royset et al., 2001)*	2	2	1	70	Benchmark $\mathfrak{c}(\boldsymbol{d}) = \mathfrak{c}_0(\boldsymbol{d}) + \mathfrak{c}_f(\boldsymbol{d})\, p_f(\boldsymbol{d})$
A bracket structure *(Chateauneuf & Aoues, 2008)*	8	2	2	160/90	Influence of the dimension Benchmark
A 23-member plane truss bridge *(Blatman & Sudret, 2008b)*	10	2	1	350	Influence of the dimension A first simplistic FE-based example

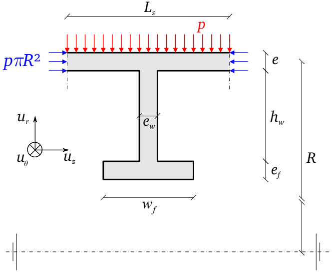
$n = \dim(\boldsymbol{X})$, $n_d = \dim(\boldsymbol{d})$, n_p *is the number of probabilistic constraints,*
N *is the number of evaluations for each limit-state function.*

Application to the design of an imperfect submarine hull

Problem formulation



Single bay reference structure



Design objectives

- The design should minimize the following *weight ratio*:

$$\mathfrak{c}(\boldsymbol{d}) = \frac{\rho_{\text{sea water}}\, \mathcal{V}_{\text{sea water}}(\boldsymbol{d})}{\rho_{\text{steel}}\, \mathcal{V}_{\text{steel}}(\boldsymbol{d})}$$

- while ensuring structural integrity for an *accidental depth charge*:

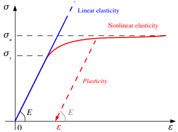
$$\mathfrak{g}(\boldsymbol{x}) = p_{\text{collapse}}(\boldsymbol{x}) - p_{\text{acc}}$$

Application to the design of an imperfect submarine hull

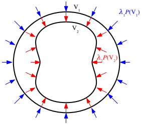
Mechanical and probabilistic modelling

Mechanical modelling *(Noirfalise, 2009)*

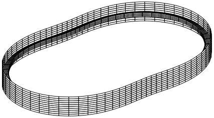
- Nonlinear finite element model (geometry, material and load);
- Shape imperfections distributed according to *two critical buckling patterns*



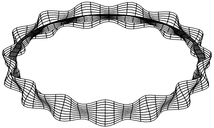
Nonlinear elasticity



Follower forces



Mode 2 (A_2)



Mode 14 (A_{14})

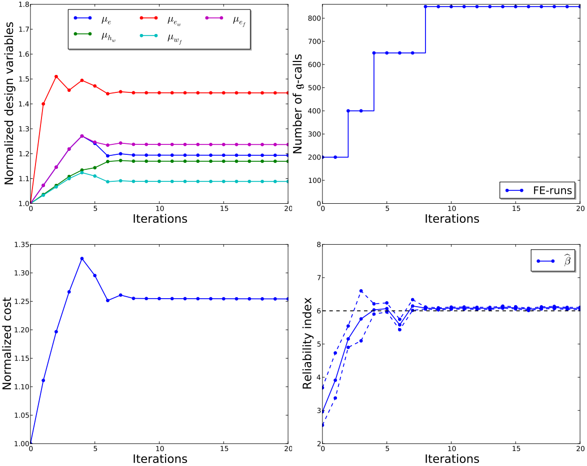
Probabilistic model

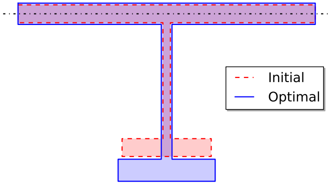
Variable		Distribution	Mean	C.o.V.
E	(MPa)	Lognormal	$200,000$	5%
σ_y	(MPa)	Lognormal	390	5%
σ_u	(MPa)	Lognormal	570	3%
e	(mm)	Lognormal	μ_e	3%
h_w	(mm)	Lognormal	μ_{h_w}	3%
e_w	(mm)	Lognormal	μ_{e_w}	3%
w_f	(mm)	Lognormal	μ_{w_f}	3%
e_f	(mm)	Lognormal	μ_{e_f}	3%
A_2	(mm)	Lognormal	$\frac{1}{3}\,\frac{5\,R}{1{,}000}$	50%
A_{14}	(mm)	Lognormal	$\frac{1}{3}\,\frac{L_s}{100}$	50%

Probabilistic constraint

$$\mathbb{P}\left[p_{\text{collapse}}(\boldsymbol{X}) \leq p_{\text{acc}} \mid \boldsymbol{d}\right] \leq 10^{-9}$$

Application to the design of an imperfect submarine hull
Results





- +25% on the weight ratio
- but the failure probability was drastically reduced ($10^{-3} \rightarrow 10^{-9}$):

$$\widehat{p}_{f\,\text{metaIS}}(\boldsymbol{d}^*) = 10^{-9}$$
with a C.o.V. $\delta_{\text{metaIS}} < 5\%$

- The whole procedure required about *1,000 FE runs*.

Conclusions

Adaptive Kriging surrogates

- Universal Kriging enable an objective *quantification of the substitution error*;
- Sampling-based adaptive DOEs enabled a *reduction of this error*, while making the *use of distributed computing platforms* possible.

Meta-model-based importance sampling

- extends the use of Kriging predictors to more complex problems (*dimension* and *degree of non-linearity*);
- enables an *elimination of the error* induced by the use of a surrogate.

Surrogate-based RBDO

- The *augmented reliability space* ensures the coupling “optimization–reliability–surrogates” is efficient;
- The score function approach revealed efficient for *reliability sensitivity analysis*;

> The overall strategy answered the original problem:
> *the design of imperfect shells against buckling*

Future work

RBDO using meta-model-based importance sampling

- The failure probability gradient is *already available*;
- How to *recycle* the DOE *efficiently*?

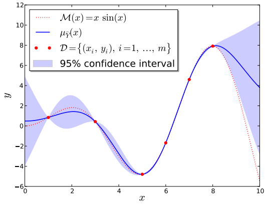
Sampling-based adaptive DOEs for global (constrained) optimization

- This could use *state-of-the-art refinement criteria* *(Jones* et al.*, 1998)*
- This would possibly allow to formulate *prior belief about the location of the optimizer* through the weighting PDF w...

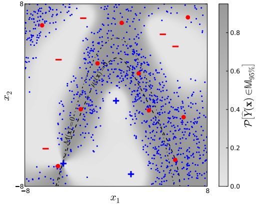
Bayesian model updating featuring expensive-to-evaluate likelihood functions

- When should the surrogate be *refined*?
- Which *error measure*?

Thank you for your attention!

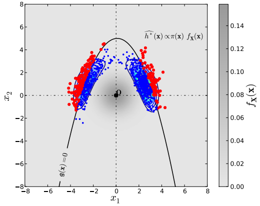


Gaussian process predictors

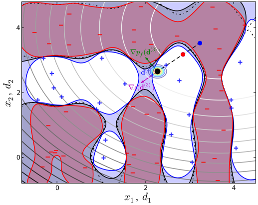


Sampling-based adaptive DOEs

?



Meta-model-based importance sampling



Surrogate-based RBDO