Novel algorithm using Active Metamodel Learning and Importance Sampling: application to multiple failure regions of low probability

Nassim Razaaly, Pietro Marco Congedo

Novel algorithm using Active Metamodel Learning and Importance Sampling: application to multiple failure regions of low probability

Nassim Razaaly, Pietro Marco Congedo

RESEARCH REPORT
N° 9079
June 29, 2017
Project-Teams CARDAMOM

ISSN 0249-6399   ISRN INRIA/RR--9079--FR+ENG

Novel algorithm using Active Metamodel Learning and Importance Sampling: application to multiple failure regions of low probability

Nassim Razaaly*, Pietro Marco Congedo †

Project-Teams CARDAMOM

Research Report n° 9079 — June 29, 2017 — 30 pages

* team CARDAMOM
† team CARDAMOM

RESEARCH CENTRE
BORDEAUX – SUD-OUEST

351, Cours de la Libération
Bâtiment A 29
33405 Talence Cedex

Abstract: Calculation of tail probabilities is of fundamental importance in several domains, such as for example risk assessment. One major challenge consists in the computation of low-failure probability and multiple-failure regions, especially when an unbiased estimation of the error is required.

Methods developed in literature rely mostly on the construction of an adaptive surrogate, tackling some problems such as the metamodel building criterion and the global computational cost, at the price of a generally biased estimation of the failure probability. In this paper, we propose a novel algorithm permitting to both building an accurate metamodel and to provide a statistically consistent error. In fact, it relies on a novel metamodel building strategy, which aims to refine the limit-state region in all the branches "equally", even in the case of multiple failure regions, with a robust stopping building criterion. Secondly, two "quasi-optimal" importance sampling techniques are used, which permit, by exploiting the accurate knowledge of the metamodel, to provide an unbiased estimation of the failure probability, even if the metamodel is not fully accurate. As a consequence, the proposed method provides a very accurate unbiased estimation even for low failure probability or multiple failure regions. Several numerical examples are carried out, showing the very good performances of the proposed method with respect to the state-of-the-art in terms of accuracy and computational cost. Additionally, another importance sampling technique is proposed in this paper, permitting to drastically reduce the computational cost when estimating some reference values, or when a very weak failure-probability event should be computed directly from the metamodel.

Key-words: Tail probability , Importance Sampling , Risk Analysis , Multiple Failure Regions , Low Failure Probability , Unbiased estimation, Rare Event, Reliability

Novel algorithm using Active Metamodel Learning and Importance Sampling: application to multiple failure regions of low probability

Résumé : Le calcul de faibles probabilités est fondamental dans plusieurs domaines, comme par exemple l'évaluation des risques. Un défi majeur consiste à calculer de probabilités très faibles et des régions d'échec multiples, surtout lorsqu'une estimation non-biaisée de l'erreur est demandée. Les méthodes développées dans la littérature reposent principalement sur la construction d'un modèle substitut adaptatif, au prix d'une estimation généralement biaisée de la probabilité d'échec. Dans ce papier, nous proposons un nouvel algorithme permettant à la fois de construire un métamodèle précis et de fournir une erreur statistiquement cohérente. En fait, il repose sur une stratégie de construction de métamodèle qui vise à affiner la région limite dans toutes les branches de la même façon, même dans le cas de régions d'échec multiples, avec un critère de construction robuste. Deuxièmement, on utilise deux techniques d'échantillonnage de Importance Sampling "quasi-optimale", qui permettent, en exploitant la connaissance précise du métamodèle, de fournir une estimation non-biaisée de la probabilité d'échec. Plusieurs exemples numériques sont réalisés, ce qui montre les très bonnes performances de la méthode proposée par rapport à l'état de l'art en termes de précision et de coût de calcul. En outre, une autre technique d'échantillonnage de type Importance Sampling est proposée dans cet article, permettant de réduire considérablement le coût de calcul lors de l'estimation de certaines valeurs de référence, ou lorsqu'un événement de probabilité de défaillance très faible doit être calculé directement à partir du métamodèle.

Mots-clés : Probabilité de queue, Importance Sampling, analyse du risque, régions multiples d'échec, estimation non-biaisée

1 Introduction

Let us introduce a probabilistic model, described by its physical n-dimensional random vector $\mathbf{Y}$ with its probability density function (PDF) $f_{\mathbf{Y}}$, and by a performance function J representing the system response. Failure can be defined as the event $\{J(\mathbf{y}) < 0\}$ so that the failure probability is formulated as follows:

$$p_f = \mathbb{P}([J(\mathbf{Y}) < 0]) = \mathbb{E}_{\mathbf{Y}}[\mathbb{1}_{J<0}(\mathbf{Y})] = \int_{\mathbb{R}^n} \mathbb{1}_{J<0}(\mathbf{y}) f_{\mathbf{Y}}(\mathbf{y}) d\mathbf{y} \tag{1}$$

where $\mathbb{1}_{J<0}$ is the failure indicator function, being equal to one if $J(\mathbf{y}) < 0$, and zero otherwise. An isoprobabilistic transformation T (e.g. Rosenblatt transform) is used to define the standard random variables X and the performance function G in the standard space as

$$X = T(Y) \tag{2}$$

$$G(X) = J(T^{-1}(X)). \tag{3}$$

We recall that $X \sim \mathcal{N}(0, I_n)$ is the standard normal random vector of $\mathbb{R}^n$. The failure probability reads then

$$p_f = \mathbb{P}([G(\mathbf{X}) \leq 0]) = \mathbb{E}[\mathbb{1}_{G<0}(\mathbf{X})] = \int_{\mathbb{R}^n} \mathbb{1}_{G<0}(\mathbf{x}) f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x}. \tag{4}$$

Computing failure analysis in the standard space simplifies the analysis, since the random variables are decorrelated and normalized. Moreover, sampling a standard normal law is easy and fast. A typical approach to the estimation of the failure probability (Eq. 4) consists in resorting to a standard, or crude, Monte Carlo (MC) scheme. Its estimator reads

$$\hat{p}_{f,MC} = \mathbb{E}_{\mathbf{X}}[\mathbb{1}_{G<0}(\mathbf{X})] = \frac{1}{N_{MC}} \sum_{k=1}^{N_{MC}} \mathbb{1}_{G<0}(\mathbf{x}^{(k)}), \tag{5}$$

where $\{\mathbf{x}^{(1)}, ..., \mathbf{x}^{(N_{MC})}\}$ is a set of N_{MC} independent samples from the random vector $\mathbf{X}$. This estimator is asymptotically unbiased and convergent.

Its variance estimator reads:

$$\hat{\sigma}^2_{f,MC} = \frac{\hat{p}_{f,MC} - \hat{p}^2_{f,MC}}{N_{MC}}. \tag{6}$$

To assess the uncertainty on the probability of failure, we use the coefficient of variation [1] defined as follows:

$$\hat{\delta}_{f,MC} = \frac{\hat{\sigma}_{f,MC}}{\hat{p}_{f,MC}} = \sqrt{\frac{1 - \hat{p}_{f,MC}}{\hat{p}_{f,MC} N_{MC}}}. \tag{7}$$

Consider also that the $(k - \sigma)$ prediction interval is defined as $[\hat{p}_{f,MC} \pm k\hat{\sigma}_{f,MC}]$. We can observe that the convergence rate is very low, $\propto N^{-\frac{1}{2}}$. If a target error of $\hat{\delta}_{f,MC} < 10^{-\Delta}$ is targeted, a failure probability as low as $p_f = 10^{-\gamma}$ would require a number of samples of $N_{MC} \approx 10^{2\Delta+\gamma}$. For example, with $\hat{\delta}_{f,MC} < 1\%$, a failure probability of $p_f = 10^{-5}$ would require a number of simulations of $N_{MC} \approx 10^9$, and, possibly, a prohibitive computational cost.

Various methods have been proposed in the literature to address this problem. We briefly recall the general ideas behind the two most widely used methods.
A first family of method aims to reduce the variance estimator in order to increase the convergence rate. Importance sampling (IS) techniques [2] [3] are part of this family, and have been developed over the past few decades to shift the underlying distribution towards the failure region so as to learn more efficiently information from rare events. The success of the method relies on a prudent choice of the importance sampling density, which undoubtedly requires the knowledge of the system in the failure region. One great advantage of this method is that it provides a statistically consistent error. The second class of methods relies on the substitution of the original performance function by a surrogate model within a sampling-based scheme; a metamodel is in general orders of magnitude faster to be evaluated. Several metamodels have been proposed in literature, such as for example quadratic response surfaces [4], polynomial chaos expansions [5], neural networks [6], kriging [7], etc. Note that in such metamodel-based approaches, Monte-Carlo sampling (AK-MCS [8]) or IS techniques (AK-IS [9], MetaAK-IS^2 [1], KAIS [10], AK-SS [11]) are used directly on the surrogate.

In particular, in [8], the AK-MCS (Active learning reliability method combining Kriging and Monte Carlo Simulation) is introduced. Results presented in this paper are very good, even for multiple failure regions, estimating very accurately failure probabilities with a relatively low number of calls to the performance function. Nevertheless, if for any stability reason (associated to the construction of the meta-model), one has to stop the refinement process, the failure branches might be not-equally refined. Indeed, AK-MCS usually does not refine each failure branches in a balanced way during the refinement process. Moreover, assessing a relatively low probability with this method remains an issue. Globally, the major drawback of the metamodel-based approaches is that it is impossible to keep the approximation error under control.

In order to cure this issue, [12] proposed to resort to a kriging-based surrogate model to approximate the optimal importance density. The so-called meta-IS [12] algorithm allows to obtain a new estimator of the failure probability as the product of a term given by a standard MC estimation based on the kriging approximation, and a correction factor computed by means of a IS technique applied to the original performance function. Nevertheless, in [13], they identified an apparent unbalanced effort between the estimation of the correction factor and the refinement of the metamodel itself.

In this paper, we propose a novel method called MetaAL-OIS (Metamodel-based combining Active Learning and quasi-Optimal Importance Sampling) method, which permits to explore all the failure regions simultaneously at a very low computational cost. It is based on the learning function used in AK-MCS [8], the k-Means clustering algorithm [14], and a MCMC sampling method. The metamodel proposed in this paper focuses on the limit state performance function. It is designed to fit with multiple failure regions, and very low probability. It provides several advantages compared to other metamodels :

- A parameter is used to avoid points clustering and also permits to implicitly control the refinement cost.

- For multiple failure regions, which could be also non-connected, the failure branches are refined back and forth during the process. So if the initial number of samples of the DOE (Design Of Experiment) required during the metamodel building is too large (impossibility to build a metamodel for instance), the resulting metamodel will be "equally" refined in all the branches. Then, an importance sampling procedure can be carried out in order to obtain an unbiased estimator of the failure probability. Moreover, the metamodel can be used in order to obtain a gross estimation of the failure probability.

- It focuses on the limit-state function: only candidate points close to the limit-state function are considered.

The second contribution of the proposed method is the use of a quasi-optimal Importance Sampling Density (ISD) ([12]) built from the metamodel, in order to obtain an unbiased estimation of p_f (see Section 3.2). Two different estimators are considered in this case. One [12] involves the product of two terms: namely one obtained by sampling the *surrogate* performance function, and the other one being a corrector factor computed from the *original* performance function. The second one combines a Control Variate technique with Importance Sampling, permitting to obtain another unbiased estimator, using the same performance function evaluations. Then, it is possible to select the best estimator aposteriori without further computations.

An additional contribution of this paper is also the formulation of a method permitting to *significantly* reduce the computational time (compared to MCS) in order to compute a failure probability directly from the metamodel, especially when the failure probability is expected to be very low. AK-IS [9], KAIS [10] and AK-SS [11] have proposed similar approaches. The idea here is to mimic the quasi-optimal ISD (Importance Sampling Density) behaviour with a gaussian mixture law. This gaussian mixture density thus serves as the ISD in the IS method, in order to reduce dramatically the number of metamodel calls, when computing the failure probability from the metamodel. The same method can be applied to the original performance function, for example in analytic cases, in order to compute high-fidelity reference values.

This paper is organized as follows. First, some general definitions concerning Gaussian Processes (GP) and IS theory are recalled in Section 2, where the sampling MCMC-based algorithms are described in subsection 2.3. The new algorithm proposed in this paper is illustrated in Section 3. In particular, the metamodel adaptive refinement and the quasi-optimal IS strategy are described in subsection 3.1 and 3.3, respectively. Section 4 illustrates several numerical examples in order to assess the accuracy and efficiency of the MetaAL-OIS method. Finally, Section 5 is devoted to draw final conclusions and some perspectives.

2 General concepts

In this section, some general concepts useful in order to illustrate the MetaAL-OIS are introduced. Indeed, the first step of the proposed algorithm, is to build a metamodel $\tilde{G}$ of the original performance function:

- The metamodel is initialized with Latin Hypercube Sampling (LHS) points, and a Gaussian Process (GP) is used to define the surrogate $\tilde{G}$
- Some candidate points are sampled using a Markov Chain Monte Carlo (MCMC) sampling technique using a surrogate for $\mathbb{1}_{G<0}(\mathbf{x})$, namely the probabilistic classification function $\pi(\mathbf{x})$.

Then, once the metamodel is accurately refined on the surrogate limit-state region $\{\tilde{G}(\mathbf{x}) = 0\}$, an IS method is extensively used on for the computation of different quantities using different Importance Sampling Densities (ISD). Finally, the unbiased estimation computation is carried out by using an IS method with the quasi-optimal ISD defined with $\pi(\mathbf{x})$, points being sampled with a multimodal MCMC sampling technique.

Before illustrating the MetaAL-OIS algorithm in Section 3, we introduce then some concepts and definitions: GP and probabilistic classification function (Subsection 2.1), importance sampling and optimal importance PDF (Subsection 2.2) and MCMC sampling methods (Subsection 2.3).

2.1 Probabilistic classification using Gaussian Processes

A metamodel is built from a *design of experiments* (D.o.E.), a set of computed experiments denoted by $\mathcal{X} = \{\mathbf{x}_1, ..., \mathbf{x}_m\}$, belonging to the support $\mathcal{D}_\mathbf{x}$ of $\mathbf{X}$. The performance function G is assumed to be a sample path of an underlying GP; its *best linear unbiased estimation* (BLUE, see [15]) at a point $\mathbf{x}$ is shown to be a Gaussian random variable $\hat{G}(\mathbf{x})$:

$$\hat{G}(\mathbf{x}) \sim \mathcal{N}(\mu_{\hat{G}(\mathbf{x})}, \sigma_{\hat{G}(\mathbf{x})}),$$

where $\mathcal{N}$ refers to the univariate gaussian law, and $\mu_{\hat{G}(\mathbf{x})}, \sigma_{\hat{G}(\mathbf{x})}$ are computed by means of the Gaussian Process algorithm. Further details can be found in [16]. The surrogate for the original performance function is denoted by:

$$\tilde{G}(\mathbf{x}) = \mu_{\hat{G}(\mathbf{x})}.$$

GP provides also an epistemic prediction uncertainty which is characterized by its variance $\sigma^2_{\hat{G}(\mathbf{x})}$. Here, we use the RBF (Radial Basis Functions) kernel (Library *GPy* written by the *Sheffield Machine Learning* group [17]). Note that the algorithm presented in this paper is obviously also compatible with any metamodel based on gaussian processes, including Kriging. According to the predicted behavior of the function, appropriate kernel or GP-based metamodel could be selected.

We introduce the *probabilistic classification function* [12]:

$$\pi(\mathbf{x}) = \mathcal{P}[\hat{G}(\mathbf{x}) \leq 0] \tag{8}$$

where the probability measure $\mathcal{P}[\cdot]$ refers to the Gaussian nature of the GP predictor $\hat{G}(\mathbf{x})$. The function $\pi(\mathbf{x})$ rewrites as follows:

$$\pi(\mathbf{x}) = \phi\left(\frac{0 - \mu_{\hat{G}(\mathbf{x})}}{\sigma_{\hat{G}(\mathbf{x})}}\right), \text{ if } \mathbf{x} \notin \mathcal{X} \tag{9}$$

where ϕ denotes the *cumulative density function* (CDF) of the one-dimensional standard normal law. Concerning the points of the experimental design for which the prediction variance is equal to zero, the above function reads

$$\pi(\mathbf{x}) = \begin{cases} 1 \text{ if } \mathbf{x} \in \mathcal{X},\ G(\mathbf{x}) \leq 0 \\ 0 \text{ if } \mathbf{x} \in \mathcal{X},\ G(\mathbf{x}) > 0 \end{cases} \tag{10}$$

It may be interpreted as the probability that the predictor $\hat{G}(\mathbf{x})$ is negative with respect to the epistemic uncertainty. We will later use π as a surrogate for $\mathbb{1}_{G<0}$.

2.2 Importance Sampling Theory

This subsection describes the importance sampling method, that is applied once the metamodel $\tilde{G}$ is built in the context of the proposed approach.
We consider the generic computation of

$$p_g = \mathbb{E}[g(\mathbf{X})], \tag{11}$$

where g refers either to $\mathbb{1}_{G<0}$, $\mathbb{1}_{\tilde{G}<0}$, π or $\mathbb{1}_{G<0} - \mathbb{1}_{\tilde{G}<0}$. Let h be a proposal PDF assumed to dominate $gf_{\mathbf{X}}$ meaning that

$$\forall \mathbf{x} \in \mathcal{D}_{\mathbf{X}}, \quad h(\mathbf{x}) = 0 \implies g(\mathbf{x}) f_{\mathbf{X}}(\mathbf{x}) = 0. \tag{12}$$

Then, p_g may be rewritten as follows:

$$p_g = \int_{\mathbb{R}^n} g(\mathbf{x}) \frac{f_{\mathbf{X}}(\mathbf{x})}{h(\mathbf{x})} h(\mathbf{x}) d\mathbf{x} = \mathbb{E}_h [g(\mathbf{X}) \frac{f_{\mathbf{X}}(\mathbf{X})}{h(\mathbf{X})}] \tag{13}$$

It easily leads to the *importance sampling estimator*:

$$\hat{p}_g = \frac{1}{N} \sum_{k=1}^{N} g(\mathbf{x}^{(k)}) \frac{f_{\mathbf{X}}(\mathbf{x}^{(k)})}{h(\mathbf{x}^{(k)})} \tag{14}$$

where $\{\mathbf{x}^{(1)}, ..., \mathbf{x}^{(N)}\}$ is a set of independent samples drawn from the instrumental density h. According to the central limit theorem, this estimation is unbiased and its quality may be measured by means of its variance estimator:

$$\hat{\sigma}_g^2 = \frac{1}{N-1} \left(\frac{1}{N} \sum_{k=1}^{N} g(\mathbf{x}^{(k)}) \frac{f_{\mathbf{X}}(\mathbf{x}^{(k)})^2}{h(\mathbf{x}^{(k)})^2} - \hat{p}_g^2 \right) \tag{15}$$

The corresponding Coefficient of Variation (COV) $\hat{\delta}_g$, quantifying the estimation accuracy, is defined as

$$\hat{\delta}_g = \frac{\hat{\sigma}_g}{\hat{p}_g} \tag{16}$$

It can be shown that the estimator's variance is zero (optimality of the IS estimator) when the instrumental pdf is chosen as the optimal importance pdf defined by:

$$h_g^*(\mathbf{x}) = \frac{g(\mathbf{x}) f_{\mathbf{X}}(\mathbf{x})}{p_g} \tag{17}$$

However, this PDF involves p_g in its denominator, so it is not implementable in practice. A good instrumental sampling pdf h should have the following properties:

- $h(\mathbf{x}) > 0$ whenever $g(\mathbf{x}) f_{\mathbf{X}}(\mathbf{x})$
- $h(\mathbf{x})$ should be close to be proportional to $g(\mathbf{x}) f_{\mathbf{X}}(\mathbf{x})$
- it should be easy to compute $h(\mathbf{x})$

In the specific case of $g = \mathbb{1}_{G<0}$, the optimal ISD reads:

$$h_G^*(\mathbf{x}) = \frac{\mathbb{1}_{G<0}(\mathbf{x}) f_{\mathbf{X}}(\mathbf{x})}{p_f} \tag{18}$$

In this case, we use $\pi(\mathbf{x})$ as a surrogate for $\mathbb{1}_{G<0}(\mathbf{x})$. In order to ensure to normalize the resulting quasi-optimal ISD h_π^* as a PDF, a constant $p_\pi = \int_{\mathbb{R}^n} \pi(\mathbf{x}) f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x} = \mathbb{E}[\pi(\mathbf{X})]$ has to be computed, so h_π^* reads:

$$h_\pi^*(\mathbf{x}) = \frac{\pi(\mathbf{x}) f_{\mathbf{X}}(\mathbf{x})}{p_\pi}. \tag{19}$$

Note that using $\mathbb{1}_{\tilde{G}<0}(\mathbf{x})$ as a surrogate for $\mathbb{1}_{G<0}(\mathbf{x})$ is tempting, leading to the following ISD $h^*_{\tilde{G}}$

$$h^*_{\tilde{G}}(\mathbf{x}) = \frac{\mathbb{1}_{\tilde{G}<0}(\mathbf{x}) f_{\mathbf{X}}(\mathbf{x})}{p_{\tilde{f}}}. \tag{20}$$

However, the condition $\mathbb{1}_{\tilde{G}<0}(\mathbf{x}) f_{\mathbf{X}}(\mathbf{x}) = 0 \implies \mathbb{1}_{G<0}(\mathbf{x}) f_{\mathbf{X}}(\mathbf{x}) = 0$ can not be ensured, so a bias may occur. In this sense, $h^*_{\pi}(\mathbf{x})$ is a robust quasi-optimal ISD and does not induce a bias in the final estimation. We illustrate how p_π is computed in subsection 3.2.

2.3 MCMC Metropolis-Hastings sampling

In order to sample points according to a given target PDF $p(\mathbf{x})$, we resort to a MCMC Metropolis-Hastings algorithm (see algorithm 1).

Algorithm 1: Metropolis-Hastings sampler

```
Input: Seed $\mathbf{x}^{(0)}$ s.t. $p(\mathbf{x}^{(0)}) > 0$, Proposal PDF $q$, $N \in \mathbb{N}$
Output: $\{\mathbf{x}^{(i)}\}_{i \in [\![1,N]\!]} \sim p$
$i = 0$ ;
while $i < N$ do
    Propose a new candidate $\mathbf{x}^* \sim q(\cdot|\mathbf{x}^{(i)})$ ;
    $r^{(i+1)} = \min(1; \frac{p(\mathbf{x}^*)q(\mathbf{x}^{(i)}|\mathbf{x}^*)}{p(\mathbf{x}^{(i)})q(\mathbf{x}^*|\mathbf{x}^{(i)})})$ ;
    Sample $u \sim \mathcal{U}[0,1]$ ;
    if $u < r^{(i+1)}$ then
        Accept $\mathbf{x}^*$. $\mathbf{x}^{(i+1)} \leftarrow \mathbf{x}^*$ ;
    else
        Reject $\mathbf{x}^*$. $\mathbf{x}^{(i+1)} \leftarrow \mathbf{x}^{(i)}$ ;
    $i \leftarrow i + 1$ ;
```

Note that it is only required to compute $p(\mathbf{x})$ within a multiplicative constant. When dealing with an unimodal pdf p, we resort to the modified Metropolis-Hasting sampler proposed in [18]. It requires a new parameter, called in the following as α_{MCMC}. The new candidate computation step is replaced by $\mathbf{x}^* \sim \mathcal{U}[\mathbf{x}^{(i)} - \alpha_{MCMC}, \mathbf{x}^{(i)} + \alpha_{MCMC}]$. This proposal PDF q being symmetrical, the new acceptance probability reads $r^{(i+1)} = \min(1; \frac{p(\mathbf{x}^*)}{p(\mathbf{x}^{(i)})})$. A burning procedure is used: the first N^{min}_{MCMC} points of the chain are rejected. Moreover, a filtering procedure is done, consisting in taking only one point every s_{MCMC}. It is suggested in [18] to take $\alpha_{MCMC} = 1$.

When dealing with a multimodal proposal PDF p, we use a multimodal Proposal PDF q as a mixture of gaussian law, depending of the current position of the chain. The mixture of gaussian law's parameters' computations are detailed in the subsection 3.2 and we consider that p contains at maximum M modes.

Finally, the algorithm 2 is used, which is inspired from [18]. Note that $c \sim \mathcal{D}(\alpha)$ refers to sample a scalar c from the discrete law $\mathcal{D}(\alpha_1, ..., \alpha_M)$. For $k \in [\![1, M]\!]$, $\boldsymbol{e_k}$ is the matrix defined by $\boldsymbol{e_k} = [\delta_{kj}]_{i,j \in [\![1,M]\!]}$, δ denoting the Kronecker function, so it is null except on the k^{th} column where it is 1.
Also, for $z = (c, \boldsymbol{\mu})$ and $\tilde{z} = (\tilde{c}, \tilde{\boldsymbol{\mu}})$, we have:

$$q(z|\tilde{z}) = \alpha_c f_{\mathcal{N}(\tilde{\mu}_c, \Sigma_c)}(\mu_c) \tag{21}$$

where $f_{\mathcal{N}(\mu,\Sigma)}$ refers to the multivariate gaussian PDF with mean μ and covariance Σ.

Algorithm 2: Multimodal Metropolis-Hastings sampler

```
Input: $(\Sigma_1, ..., \Sigma_M)$ a set of $(d \times d)$ covariance matrices
$\alpha = (\alpha_1, ..., \alpha_M) \in [0,1]^M$ a set of weights such $\sum_{k=1}^{M} \alpha_k = 1$
$\boldsymbol{\mu}^{(0)} = (\mu_1^{(0)}, ..., \mu_M^{(0)}) \in \mathbb{R}^{d \times M}$ s.t. $\forall k \in [\![1, M]\!],\ p(\mu_k^{(0)}) > 0$
$c^{(0)} \sim \mathcal{D}(\alpha)$
$z^{(0)} = (c^{(0)}, \boldsymbol{\mu}^{(0)})$
$\mathbf{x}^{(0)} = \mu_{c^{(0)}}^{(0)}$
$N \in \mathbb{N}$
Output: $\{\mathbf{x}^{(i)}\}_{i \in [\![1,N]\!]} \sim p$
$i = 0$ ;
while $i < N$ do
    Propose a new candidate $z^* = (c^*, \boldsymbol{\mu}^*) \sim q(\cdot|z^{(i)})$:
    $c^* \sim \mathcal{D}(\alpha)$
    $\mu_{c^*}^* \sim \mathcal{N}(\mu_{c^*}^{(i)}, \Sigma_{c^*})$
    $\boldsymbol{\mu}^* = \boldsymbol{\mu}^{(i)} + \boldsymbol{e}_{\boldsymbol{c}^*}(\mu_{c^*}^* - \mu_{c^*}^{(i)})$
    $z^* = (c^*, \boldsymbol{\mu}^*)$
    $\mathbf{x}^* = \mu_{c^*}^*$
    $r^{(i+1)} = \min(1; \frac{p(\mathbf{x}^*)q(z^{(i)}|z^*)}{p(\mathbf{x}^{(i)})q(z^*|z^{(i)})})$ Sample $u \sim \mathcal{U}[0,1]$ ;
    if $u < r^{(i+1)}$ then
        Accept $z^*$: $z^{(i+1)} \leftarrow z^*$
        $\mathbf{x}^{(i+1)} \leftarrow \mathbf{x}^*$;
    else
        Reject $z^*$. $z^{(i+1)} \leftarrow z^{(i)}$
        $\mathbf{x}^{(i+1)} \leftarrow \mathbf{x}^{(i)}$;
    $i \leftarrow i + 1$ ;
```

3 The MetaAL-OIS algorithm

In this section, we describe the method proposed in this paper, *i.e.* the MetaAL-OIS algorithm. A general sketch of the algorithm is given in Figure **??**.

The first step consists in building a metamodel refined on the limit-state region, which is described in Section 3.1. Secondly, a gaussian mixture ISD is used to compute a primary estimation of the probability of failure, based only on the metamodel:

$$\hat{p}_{\tilde{f}} = \mathbb{E}[\mathbb{1}_{\tilde{G}<0}(\mathbf{X})] \tag{22}$$

$$\hat{p}_{\pi} = \mathbb{E}[\pi(\mathbf{X})]. \tag{23}$$

This method is particularly fit for very low failure probabilities, so MCS method would involve a very large number of evaluations, so the computational cost on the metamodel would not be anymore negligible. It drastically reduces the metamodel calls. This is illustrated in Section 3.2. Third, in order to provide an unbiased estimation of p_f at a low computational cost, two estimators, described in Section 3.3, are computed: the first one is directly taken from MetaIS [1] algorithm, and the second uses both Control Variate method and IS. They both use the same points drawn from the multimodal MCMC sampler (see algorithm 2) with the ISD $h^*_\pi(\mathbf{x})$. The original performance function G is evaluated at those points, and the two estimators are computed. So, an *a posteriori* analysis of their respective variance estimation leads the user to select the best estimation. If the target coefficient of variation δ_{target} is not reached, new points are sampled, and the estimators are updated. Otherwise, the algorithm stops, and the estimator associated to the lowest coefficient of variation is returned.

Finally, Section 3.4 is devoted to a description about the parameters tuning of the proposed algorithm.

3.1 Metamodel Refinement Strategy

We propose here a new metamodel refinement strategy. In order to show the main features of this algorithm, let us introduce first the definition of the learning function U [13] associated to $\tilde{G}$. For a given $\mathbf{x}$, $U(\mathbf{x})$ is defined as

$$U(\mathbf{x}) = \frac{|\mu_{\hat{G}(\mathbf{x})}|}{\sigma_{\hat{G}(\mathbf{x})}}. \tag{24}$$

In this definition, $\phi(U(\mathbf{x}))$ is the probability that $\mathbf{x}$ is correctly classified by the predictor, where ϕ refers to the cumulative probability function of the standard Gaussian random variable. For a given set of points $\mathcal{X}$, $\mathbf{x}_0$, defined as follows

$$\mathbf{x}_0 = \underset{\mathbf{x}\in\mathcal{X}}{\arg\min}(U(\mathbf{x})), \tag{25}$$

indicates the point where the classification is the most uncertain among $\mathcal{X}$.

In AK-MCS [8], the D.o.E. is iteratively enriched at the point $\mathbf{x}_0$ by minimizing the learning function U (see Eq. 25) searching in a Monte Carlo population. The stopping criterion is $U(\mathbf{x}_0) \geqslant 2$, meaning that the sample, whose group is the most uncertain, displays a probability being correctly classified of at least $\phi(2) = 97.7\%$. Though, this method has several drawbacks. When the limit state $G(\mathbf{x}) = 0$ has different branches, the learning function U usually focuses on one of them first. Once the metamodel is accurate enough in this region, the learning criterion goes to another branch and carries on. If the refinement procedure is stopped, for instance because the DOE is too large, the metamodel can be accurately refined in some failure branches and too coarse on some others. Moreover, it can lead to clusters of points so numerical instabilities can arise during the metamodel building.

For these reasons, we propose here a new learning function,
$\mathcal{L}_U(\mathcal{X}, d_{min}, d_{max}, r_{max}, U_{max}, \mathcal{X}_{DOE})$:
It returns either a set containing a new point $\{\mathbf{x}_0\} \in \mathcal{X}$, or $\varnothing$. The algorithm has the following steps :

1. Compute the subset $\mathcal{Y} \subseteq \mathcal{X}$ such that $\forall \mathbf{y} \in \mathcal{Y}$:
   - $\min\limits_{\mathbf{z}\in\mathcal{X}_{DOE}} (\|\mathbf{y} - \mathbf{z}\|) \in [d_{min}, d_{max}]$,
   - $\|\mathbf{y}\| \leqslant r_{max}$
2. In $\mathcal{Y}$, select the point (if $\mathcal{Y}$ is not empty) minimizing the learning function U: $\mathbf{x}_0 = \underset{\mathbf{y}\in\mathcal{Y}}{\arg\min}(U(\mathbf{y}))$.
3. If $\mathcal{Y}$ is empty, or $U(\mathbf{x}_0) > U_{max}$, return $\varnothing$. Otherwise return $\{\mathbf{x}_0\}$.

Note that $\mathcal{X}_{DOE}$ represents the current set of D.o.E., $\mathcal{Y}$ contains points which satisfy geometrical conditions prescribing a minimal/maximal distance between the D.o.E. candidates and the current ones. Moreover, candidates which are too far from the center $0_{\mathbb{R}^d}$ are rejected. Minimizing the learning function U on Y permits to find the most uncertain point. Returning only points $\mathbf{x}_0$ such that $U(x_0) \leq U_{max}$ ensures that points too accurately predicted are not considered as D.o.E candidates. We suggest to take the following parameters:

- $d_{min} = 0.5$: if the performance function is expected to be very non-linear. Decreasing this value can lead to a more (costly) accurate metamodel.
- $d_{max} = 2$:
- $r_{max} = 5$: this parameter can be increased if the failure probability is expected to be very low, so the failure region can be distant from the center.
- $U_{max} = 2$: in this case, the accepted D.o.E. candidate is correctly classified by the predictor with a probability less than 97.7%. Increasing this parameter will increase both the global number of samples constituting the D.o.E. and the metamodel accuracy.

The refinement algorithm for the metamodel construction relies then on the following steps :

1. *Initial DOE and metamodel definition*: sample m_0 points generated by a Latin Hypercube procedure in the hypercube $[a_{min}, a_{max}]^d$. Build the metamodel from this set of points (Parameters $a_{min} = -5$, $a_{max} = 5$ give satisfactory results). Though, if p_f is expected to be very low, so the failure regions should be more distant from the standard space center, the hypercube box should be increased. Set i = 1.

2. *Sampling*: Sample a quite large population $\mathcal{N}_\gamma = \{\mathbf{x}_\gamma^{(1)}, ..., \mathbf{x}_\gamma^{(N_\gamma)}\}$ constituted by independent samples drawn from $\mathcal{N}(0, \gamma I_d)$. This step is aimed to capture all failure regions, with a relatively low number of samples (Parameters $N_\gamma = 10^4$ and $\gamma = 1.5$ give good results). The $p_{\tilde{f}}$ value is evaluated using IS with the PDF $f_{\mathcal{N}(0,\gamma I_d)}$. Then, N_γ is updated so a target $\hat{\delta}_{\tilde{G},\gamma}$ is reached. If $p_{\tilde{f}} = 0$, meaning that no metamodel failure points have been found, so $\tilde{G}$ might be too coarse at this stage, update the D.o.E at $\mathbf{x}_0 \in \mathcal{N}_\gamma$ minimizing the learning function U (AK-MCS step), and go to step 1. Otherwise go to step 3.

3. *Classification*: Compute $\mathcal{X}^- = \{\mathbf{x} \in \mathcal{N}_\gamma : \mu_{\hat{G}(\mathbf{x})} \leq 0\}$.

4. *Global Point Selection*: Refine and update the metamodel at $\mathcal{L}_U(\mathcal{X}^-, d_{min}, d_{max}, r_{max}, U_{max}, \mathcal{X}_{DOE})$, where $\mathcal{X}_{DOE}$ is the current D.o.E. ; if #{D.o.E.} $\geqslant N_{DOE}^{MAX}$, stop the metamodel refinement algorithm.

5. *Update $\mathcal{X}^-$*: Repeat steps 2 and 3 and go step 6.

6. *Seed Selection*: Use k-Means [14] clustering algorithm on the set $\mathcal{X}^-$. Its K clusters centroids are considered: $(\mathbf{x}_1^-, ..., \mathbf{x}_K^-)$. Set $k = 1$.

7. *Unimodal MCMC Sampling*: Sample a set of N_{MCMC} points $\mathcal{S}_k^{MCMC}$ drawn from the PDF $h_{\tilde{G}}^*$ using the modified MCMC Metropolis-Hastings sampler, with the seed initialized at $\mathbf{x}_k^-$. Burning and thining procedure are used. Remove the first N_{MCMC}^{min} values obtained, and then, accept only one sample every s_{MCMC} samples. Note that this technique can be applied regardless of the constant $p_{\tilde{f}}$.

8. *Enrich D.o.E. at step k*: Update the metamodel at $\tilde{\mathbf{x}}_k^- = \mathcal{L}_U(\mathcal{S}_k^{MCMC}, d_{min}, d_{max}, r_{max}, U_{max}, \mathcal{X}_{DOE})$. ; if $\#\{\text{D.o.E.}\} \geqslant N_{DOE}^{MAX}$, stop the metamodel refinement algorithm.

9. *Stopping criterion* If $k \neq K$: set $k = k + 1$ and loop back to step 7.
   If $k = K$ and $\bigcup_{k=1}^{K} \{\tilde{x}_k^-\} = \varnothing$, stop the metamodel refinement algorithm.
   If $i = N_{loop}^{MAX}$, stop the metamodel refinement algorithm.
   Otherwise, set $i = i + 1$ and loop back to step 2.

Finally, the metamodel refinement strategy depends on the following tuning parameters : $m_0, \gamma, N_\gamma, \hat{\delta}_{\tilde{f},\gamma}, K, N_{MCMC}, N_{MCMC}^{min}, s_{MCMC}, \alpha_{MCMC}, d_{min}, d_{max}, r_{max}, U_{max}, N_{loop}^{MAX}$, N_{DOE}^{MAX}.

Globally, the metamodel refinement strategy can be resumed under the following algorithm 3:

Algorithm 3: Metamodel Refinement Algorithm

```
Input: $m_0$, $N_{loop}^{MAX}$, $K$
Output: Metamodel $\tilde{G}$
Sample $m_0$ LHS points in $[a_{min}, a_{max}]^d$. Build $\tilde{G}$ s.t. $p_{\tilde{f}} > 0$.
for $i_{loop} = 1,\ N_{loop}^{MAX}$ do
    Sample $\mathcal{N}_\gamma$ ; Select $x_0 \in \mathcal{X}^-$ ; Update $\tilde{G}$.
    Run k-Means algorithm on $\mathcal{X}^-$ and find $(x_1^-, ..., x_K^-)$.
    for $k = 1,\ K$ do
        Sample $\mathcal{S}_k^{MCMC}$ using MCMC: the target PDF is $h_{\tilde{G}}^*$, the seed is $x_k^-$.
        Select (if exists) $\mathbf{x}_0 \in \mathcal{S}_k^{MCMC}$ ; Update $\tilde{G}$.
    if No points added among $\bigcup_{k=1}^{K} \mathcal{S}_k^{MCMC}$ then
        Return $\tilde{G}$.
    Return $\tilde{G}$.
```

3.2 Importance Sampling on the metamodel: Gaussian mixture ISD

We detail in this subsection the method used to compute p_π and $p_{\tilde{f}}$, starting with a refined metamodel $\tilde{G}$.

The p_π and $p_{\tilde{f}}$ values are computed using Importance Sampling, the instrumental PDF $h_\mathcal{N}$ being chosen as a mixture of gaussians as follows:

$$h_\mathcal{N}(\mathbf{x}) = \sum_{k=1}^{K^{IS}} \alpha_k^{IS} f_{\mathcal{N}(\mu_k^{IS}, \Sigma_k^{IS})}(\mathbf{x}). \tag{26}$$

Note that the computation of $\{\alpha_k^{IS},\ \mu_k^{IS}, \Sigma_k\}_k$ is the key of the efficiency of this method, resumed in algorithm 4. In a nutshell, failure points are sampled with MCS and classified in K^{init} groups. Centroids of these groups are iteratively used as the seed of the unimodal MCMC sampler, in order to sample failure points following the quasi-optimal ISD h_π^*, in *all failure regions*.

These final sampled failure points are again classified in K^{IS} groups, from which the empirical weight, mean and covariance are estimated and used as the gaussian mixture parameters.

Algorithm 4: Empirical estimation of $\{\alpha_k^{IS},\ \mu_k^{IS}, \Sigma_k\}_k$.

```
Input: $K^{init}, K^{IS}$
Output: Metamodel $\tilde{G}$
Sample a MC population $\mathcal{N}_\gamma$ and classify the failure points $\mathcal{X}^-$ according to $\tilde{G}$
Run k-Means algorithm on $\mathcal{X}^-$ and find the centroids $(x_1^-, ..., x_{K^{init}}^-)$.
for $k = 1,\ K^{init}$ do
    Sample $\mathcal{S}_k^{MCMC}$ using MCMC: the target PDF is $h_\pi^*$, the seed is $x_k^-$.
Run k-Means algorithm on $\bigcup_{k=1}^{K} \mathcal{S}_k^{MCMC}$ so it is classified in $K^{IS}$ sets $(\mathcal{S}_1^{IS}, ..., \mathcal{S}_{K^{IS}}^{IS})$.
$\forall k \in [\![1, K^{IS}]\!]$, $\mu_k^{IS}$ and $\Sigma_k$ are defined respectively as the empirical mean and the
empirical covariance of $\mathcal{S}_k^{IS}$. $\alpha_k^{IS}$ is defined as the empirical weight $\dfrac{\#\mathcal{S}_k^{IS}}{\#\bigcup_{k=1}^{K^{IS}} \mathcal{S}_k^{IS}}$.
```

During this step, the following tuning parameters should be chosen : γ, N_γ, $\hat{\delta}_{\tilde{f},\gamma}$, K^{init}, K^{IS}, N_{MCMC}, N_{MCMC}^{min}, s_{MCMC}, α_{MCMC}.

Using this IS method is not necessary when the estimated failure probability is relatively high. Though, even in the case of very low failure probabilities, it is possible to have sharp estimations of those quantities with a reasonable number of metamodel evaluations. The idea behind this IS technique, is to mimic the quasi-optimal multimodal ISD in order to decrease the required number of samples, with a gaussian mixture density which does not require any unknown parameters.

The resulting computations provide the estimations of $\hat{p}_\pi$ and $\hat{p}_{\tilde{f}}$, the corresponding variance estimations $\hat{\sigma}_{p_\pi}^2$ and $\hat{\sigma}_{p_{\tilde{f}}}^2$ and the coefficients of variation $\hat{\delta}_{p_\pi}$ and $\hat{\delta}_{p_{\tilde{f}}}$.

The $\hat{p}_{\tilde{f}}$ value can be used as a good estimation of p_f, but an error bound cannot be provided. However, it is in practice a very good estimation.

This IS procedure, aimed to provide the estimation of $\hat{p}_{\tilde{f}}$ based on a sampling technique applied to the metamodel, shares the same objective of the following metamodel-based methods, once the metamodel is refined: AK-IS, KAIS, MetaAK-IS^2, AK-SS. It aims to decrease the number of metamodel evaluations when the expected failure probability is very low.

3.3 Importance Sampling on the performance function: quasi-optimal ISD

In order to have a statistically consistent error for $\hat{p}_f$, we consider the instrumental PDF $h_\pi^*(\mathbf{x}) = \dfrac{\pi(x) f_{\mathbf{X}}(\mathbf{x})}{p_\pi}$, and we propose two estimators based on this quasi-optimal ISD.

3.3.1 Importance Sampling Estimator

This estimator is the one used in the second step of the metaIS [12]. The failure probability is re-written as:

$$\begin{aligned} p_f &= \int_{\mathbb{R}^n} \mathbb{1}_{G<0}(\mathbf{x}) \frac{f_{\mathbf{X}}(\mathbf{x})}{h_\pi^*(\mathbf{x})} h_\pi^*(\mathbf{x}) d\mathbf{x} \\ &= p_\pi \int_{\mathbb{R}^n} \frac{\mathbb{1}_{G<0}(\mathbf{x})}{\pi(\mathbf{x})} h_\pi^*(\mathbf{x}) d\mathbf{x} \\ p_f &= p_\pi \alpha_\pi \end{aligned} \tag{27}$$

(28)

where $\alpha_\pi = \mathbb{E}_{h_\pi^*}\left[\frac{\mathbb{1}_{G<0}(\mathbf{X})}{\pi(\mathbf{X})}\right]$ is a correction factor. Its estimator $\hat{\alpha}_\pi$ is evaluated using the importance sampling method described in Section 2.2, with the instrumental PDF h_π^*, regardless of the constant p_π. The points are sampled using the multimodal MCMC Metropolis Hastings sampler, initializing the seed and the covariances by using the empirical quantities $\{\alpha_k^{IS},\ \mu_k^{IS}, \Sigma_k\}_k$. The variance estimator $\hat{\sigma}^2_{\alpha_\pi}$ and the coefficient of variation $\hat{\delta}_{\alpha_\pi}$ are also computed.
An unbiased estimator of p_f, referred in the following as $\hat{p}_f$, provided by IS, reads

$$\hat{p}_f = \hat{p}_\pi \hat{\alpha}_\pi. \tag{29}$$

Its variance estimator reads

$$\begin{aligned} \hat{\sigma}_f^2 &= \mathbb{E}[\hat{p}_\pi^2 \hat{\alpha}_\pi^2] - \hat{p}_f^2 \\ &= \mathbb{E}[\hat{p}_\pi^2]\mathbb{E}[\hat{\alpha}_\pi^2] - \hat{p}_\pi^2 \hat{\alpha}_\pi^2 \\ &= (\hat{\sigma}^2_{p_\pi} + \hat{p}_\pi^2)(\hat{\sigma}^2_{\alpha_\pi} + \hat{\alpha}_\pi^2) - \hat{p}_\pi^2 \hat{\alpha}_\pi^2 \\ \hat{\sigma}_f^2 &= \hat{\sigma}^2_{p_\pi} \hat{\sigma}^2_{\alpha_\pi} + \hat{\sigma}^2_{p_\pi} \hat{\alpha}_\pi^2 + \hat{p}_\pi^2 \hat{\sigma}^2_{\alpha_\pi}, \end{aligned} \tag{30}$$

and its coefficient of variation can be formulated as follows:

$$\hat{\delta}_f = \sqrt{\hat{\delta}^2_{p_\pi} + \hat{\delta}^2_{\alpha_\pi} + \hat{\delta}^2_{p_\pi}\hat{\delta}^2_{\alpha_\pi}}. \tag{31}$$

3.3.2 Control Variate/Importance Sampling estimator

Here, we illustrate a novel estimator based on a combination between Control Variate and IS methods. The failure probability is re-written as:

$$\begin{aligned} p_f &= \mathbb{E}[\mathbb{1}_{\tilde{G}<0}(\mathbf{X})] + \int_{\mathbb{R}^n} \{\mathbb{1}_{G<0}(\mathbf{x}) - \mathbb{1}_{\tilde{G}<0}(\mathbf{x})\} \frac{f_{\mathbf{X}}(\mathbf{x})}{h_\pi^*(\mathbf{x})} h_\pi^*(\mathbf{x}) d\mathbf{x} \\ &= p_{\tilde{f}} + p_\pi \eta_\pi, \end{aligned} \tag{32}$$

where $\eta_\pi = \mathbb{E}_{h_\pi^*}\left[\frac{\mathbb{1}_{G<0}(\mathbf{x}) - \mathbb{1}_{\tilde{G}<0}(\mathbf{x})}{\pi(\mathbf{X})}\right]$.

Its estimator $\hat{\eta}_\pi$ is evaluated using the importance sampling method described in Subsection 2.2, with the instrumental PDF h_π^*, as for $\hat{\alpha}_\pi$, with exactly the same points drawn from the multimodal MCMC Metropolis Hastings sampler. The variance estimator $\hat{\sigma}^2_{\eta_\pi}$ and the coefficient of variation $\hat{\delta}_{\eta_\pi}$ are also given in a similar pattern. We also define $\epsilon_\pi = p_\pi \eta_\pi$.

The idea is to compute accurate estimations of $p_{\tilde{f}}$ and p_π using only metamodel evaluations. A correction constant ϵ_π is then computed using performance evaluations in order to obtain a unbiased estimation of p_f. An unbiased estimator of ϵ_π is given by

$$\hat{\epsilon}_\pi = \hat{p}_\pi \hat{\eta}_\pi. \tag{33}$$

Its variance estimator is given by

$$\hat{\sigma}^2_{\epsilon_\pi} = \hat{\sigma}^2_{p_\pi} \hat{\sigma}^2_{\eta_\pi} + \hat{\sigma}^2_{p_\pi} \hat{\eta}^2_\pi + \hat{p}^2_\pi \hat{\sigma}^2_{\eta_\pi}, \tag{34}$$

and its coefficient of variation by:

$$\hat{\delta}_{\epsilon_\pi} = \sqrt{\hat{\delta}^2_{p_\pi} + \hat{\delta}^2_{\eta_\pi} + \hat{\delta}^2_{p_\pi} \hat{\delta}^2_{\eta_\pi}}. \tag{35}$$

An unbiased estimator of p_f, referred in the following as $\hat{p}_f$ and provided by CV+IS, reads

$$\hat{p}_f = \hat{p}_{\tilde{G}} + \hat{\epsilon}_\pi. \tag{36}$$

where its variance estimator is

$$\hat{\sigma}^2_f = \hat{\sigma}^2_{p_{\tilde{f}}} + \hat{\sigma}^2_{\epsilon_\pi}, \tag{37}$$

and its coefficient of variation is defined by:

$$\hat{\delta}^2_f = \frac{\hat{\sigma}^2_f}{\hat{p}^2_f}. \tag{38}$$

Since the points required for the estimation of η_π and α_π are the same, it is possible during this step to compute with the same points two unbiased estimators (IS and CV+IS) for p_f, and then select the one that returns the lowest coefficient of variation.

3.4 Parameters tuning

In this section, we make some recommendations about the parameters to tune. The recommended parameters can be found in Table 1. The suggested values should give satisfactory results. According to the case, during the metamodel refinement, the user should be specifically aware of the following key parameters: d_{min}, K. In particular:

- If the failure probability is expected to be very low, the limit-state is more likely to be distant from the center, so γ, a_{min}, a_{max}, r_{max}, d_{max} should be increased.
- The exact number of failure regions should be less than K, K^{init} and K^{IS}. As a consequence, these parameters should be increased, if the number of failure regions observed on the metamodel is higher.
- U_{max} can be modified according to the metamodel confidence required by the user. A lower U_{max} should make the metamodel refinement algorithm to converge faster, but with a lowest accuracy.
- The metamodel refinement depends strongly on the d_{min} parameter. A larger d_{min} should lead to a coarser DoE. Re-starting the metamodel refinement strategy with a lower d_{min} could increase the accuracy of the metamodel.

a)							
N_{MCMC}	N_{MCMC}^{min}	s_{MCMC}	α_{MCMC}	γ	N_γ	$\hat{\delta}_{\tilde{f},\gamma}$	K
2000	5000	5	1	1.5	10^4	5%	5
m_0	a_{min}	a_{max}	r_{max}	d_{min}	d_{max}	U_{max}	N_{DOE}^{MAX}
15	-5	5	5	0.5	2	2	100
b)							
N_{MCMC}	N_{MCMC}^{min}	s_{MCMC}	α_{MCMC}	γ	N_γ	$\hat{\delta}_{\tilde{f},\gamma}$	K^{init}
2000	5000	5	1	1.5	10^4	5%	5
K^{IS}							
5							

Table 1: Recommended Tuning Parameters: a) Metamodel construction, b) Gaussian Mixture ISD Parameters.

4 Numerical results

In this section, several numerical examples are treated in order to illustrate the efficiency of the proposed method. The recommended tuning parameters described in Table 1 are systematically used, except if explicitly mentioned.

The proposed method is compared to those ones providing an unbiased estimation and to others including metamodel-based construction, where only an estimation of the failure probability based on the metamodel is provided. Comparisons in terms of accuracy and computational costs are illustrated on resuming tables.

In particular, the first part of the tables shows the p_f estimation and associated errors based only on the metamodel, including the number of calls done on the performance function in order to construct the metamodel. For computing $p_{\tilde{f}}$, the current practice consists in using Monte Carlo Sampling (MCS) directly on the metamodel. Though, when $p_{\tilde{f}}$ is very low, this method is not adapted since it would require a very large number of metamodel evaluations in order to obtain a high accuracy. In this paper, we show how using IS (see Section 3.2) with the gaussian mixture density as the ISD in order to obtain a sharp estimation of $p_{\tilde{f}}$ (less than 10^{-7}) in a reasonable number of metamodel calls. The second part of the tables shows the results based on the unbiased methods, including the total number of calls done on the performance function. Moreover, in order to assess the method, results are compared to a reference value, provided by MCS. Its variance estimation provides a $3-\sigma$ prediction interval in which the true failure probability p_f should lie. When p_f is very low, since a very large number of MCS points are necessary, this $3-\sigma$ interval should still be quite large. In order to reduce it, and to obtain a more precise $3-\sigma$ prediction interval, we use again the IS method mentioned in Section 3.2 using a gaussian mixture density as the ISD to compute reference value very accurately, until the associated coefficient of variation is below 0.10%. This method is labeled with *Perf + IS* in the tables.
In the papers where comparisons are provided, as for instance in [1] [8] [9], methods returning an unbiased failure probability estimation aim generally a COV target of 5%. In the following cases, this accuracy is most of the time attained (once the metamodel is built), by computing the unbiased estimators with only the first two hundred (200) performance calls. As a consequence, we show results for a COV target of 1%.

4.1 2D analytic example with a single failure region: Case 1

A 2D analytic example taken from [9] [1] is chosen to test the performances of the algorithm. This example is characterized by a low failure probability ($p_f \sim 3 \times 10^{-5}$), a very smooth limit state and a single failure region. The performance function in the standard space reads:

$$G(x_1, x_2) = \frac{1}{2}(x_1 - 2)^2 - \frac{3}{2}(x_2 - 5)^3 - 3 \tag{39}$$

where x_1, x_2 are the realizations of two independent standard Gaussian random variables.

To run the simulation, $m_0 = 10$ is used as tuning parameter.

Figure 1 illustrates how well the limit-state branch is approximated by the metamodel. Nine points are adaptively added, in order to fit the limit-state branch. The quasi-optimal ISD h^*_π seems to accurately approximate the optimal ISD $h^*_{\tilde{f}}$, and $\pi(\mathbf{x})$, to be an accurate surrogate for $\mathbb{1}_{\tilde{G}<0}(\mathbf{x})$.

In Table 2, we compare the results with those ones reported in [1], based on the following methods: Crude MC, FORM, FORM+IS, AK-IS, MetaAK-IS^2.

MetaAL-OIS method, based only on $\widetilde{G}$, returns a very good result, with a number of calls similar to the other metamodel-based methods, though with a much lower COV, *i.e.* $\hat{\delta}_{p_{\tilde{f}}}$. Indeed, applying the IS method described in 3.2 using a gaussian mixture density as the ISD, it is possible to have a very accurate estimation of $p_{\tilde{f}}$ in less than 10^7 metamodel calls. This remark applies systematically to each test-case presented in this Section 4.

Concerning the unbiased estimation, a reference solution is computed again using a gaussian mixture density as the ISD (labeled as Perf+IS in the table). In this case, a coefficient of variation of 0.10% is attained on the computation of a low probability $p_f = 2.87 \times 10^{-5}$ in only 1.05×10^6 evaluations. To reach the same accuracy, Crude MC method would require around 3×10^{10} evaluations. As a consequence, the 3-σ interval for Perf+IS is narrower than the one for Crude MC method, with a lower number of evaluations. Then, it should be more accurate. Again, this behavior is observed for each test-case presented in this section.

With the method proposed in this paper, i.e. MetaAL-OIS, only two hundred additional points are necessary in order to obtain a unbiased p_f estimation. Indeed, in this case, the metamodel is so well refined that the CV+IS estimator gives exactly the estimator $\hat{p}_f$. It means that the estimation of the correction constant $\hat{\epsilon}$ is zero: among the two hundred samples drawn, no one lies between the two limit-state $\{\widetilde{G}(\mathbf{x}) = 0\}$ and $\{G(\mathbf{x}) = 0\}$. We outline that the attained accuracy is very high: $\hat{\delta}_f = 0.10\%$. We can note that the IS estimator, for the same two hundred additional points, returns a unbiased estimation with $\hat{\delta}_f = 1.31\%$ which is already very low, but anyway greater than the one obtained with the CV+IS estimator. Note that the computation of the $3 - \sigma$ interval is obviously coherent with the result of $\hat{\delta}_f$. Note also that the results obtained with MetaAL-OIS are coherent with the reference solution (Perf+IS), presented earlier.

In FORM method, the Most Probable failure Point (MPP) is evaluated, assuming the case of a single failure region, so an estimation of p_f is returned assuming that G is locally linear. As it can be observed, the solution is wrong. In FORM+IS method, the standard gaussian distribution is shifted to this MPP and used as the ISD of an Importance Sampling method, so it is possible to obtain an unbiased estimation of p_f, but usually at a high computational cost.

Finally, in this case involving a one failure region characterized by a low probability, MetaAL-OIS is fully satisfactory, providing an accurate metamodel at a low cost, and a highly accurate unbiased failure probability estimation with only two hundred additional points.

	Metamodel-based Estimation				Unbiased Estimation			
Method	$\tilde{N}_{calls}$	$\hat{p}_{\tilde{G}}$	$\hat{\delta}_{p_{\tilde{f}}}$	$3-\hat{\sigma}_{p_{\tilde{f}}}$ Interval	N_{calls}	$\hat{p}_f$	$\hat{\delta}_f$	$3-\hat{\sigma}_f$ Interval
Crude MC					5×10^7	2.85×10^{-5}	2.64%	$[2.62, 3.08] \times 10^{-5}$
Perf + IS					*1.05×10^6*	*2.87×10^{-5}*	*0.10%*	*$[2.86, 2.89] \times 10^{-5}$*
FORM					19	4.21×10^{-7}		
FORM + IS					$19 + 10^4$	2.86×10^{-5}	2.39%	$[2.66, 2.95] \times 10^{-5}$
AK-IS	26	2.86×10^{-5}	2.39%	$[2.65, 3.07] \times 10^{-5}$				
MetaAK-IS^2	28	2.87×10^{-5}	2.39%	$[2.66, 3.08] \times 10^{-5}$				
MetaAL-OIS	**19**	$\mathbf{2.87 \times 10^{-5}}$	**0.10%**	$\mathbf{[2.86, 2.88] \times 10^{-5}}$	$\mathbf{19 + 200}$	$\mathbf{2.87 \times 10^{-5}}$	**0.10%**	$\mathbf{[2.86, 2.88] \times 10^{-5}}$
MetaAL-OIS(IS)					$19 + 200$	2.81×10^{-5}	1.31%	$[2.70, 2.93] \times 10^{-5}$
MetaAL-OIS(CV+IS)					$19 + 200$	2.87×10^{-5}	0.10%	$[2.86, 2.88] \times 10^{-5}$

Table 2: Comparison of the performances of the MetaAL-OIS with several algorithms of literature[1]: Case 1.

4.2 2D analytic example with two failure regions: Case 2

This example deals with a case of two non-connected failure regions, featuring failure probabilities lying from around 3×10^{-3} to 9×10^{-7}, according to the selected parameter c in Equation 40. The performance function [1] in the standard space reads:

$$G(x_1, x_2) = \min \left\{ \begin{array}{c} c - 1 - x_2 + e^{\frac{-x_1^2}{10}} + (\frac{x_1}{5})^4 \\ \frac{c^2}{2} - x_1 \cdot x_2 \end{array} \right\} \tag{40}$$

where x_1, x_2 are the realizations of two independent standard Gaussian random variables. We consider here three cases, with $c = 3, 4, 5$, respectively. Note that for $c = 5$, the case is challenging because it involves two non-connected failure regions and a very low failure probability.

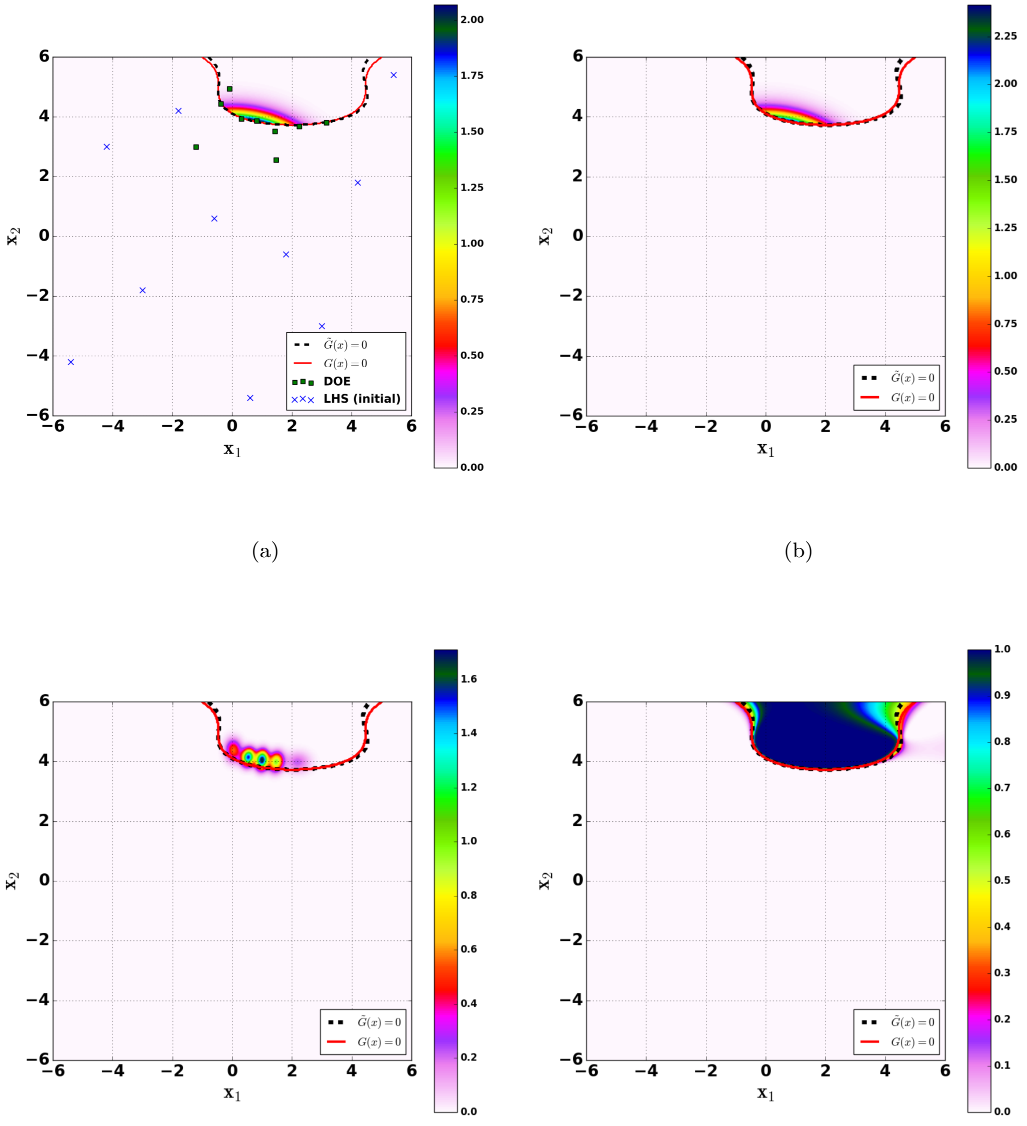
Concerning the parameters tuning, it is unchanged for $c = 3$. On the other hand, when $c = 4$ case is considered, $p_f \sim 9 \times 10^{-5}$, so the failure branches are expected to be further from the center. As a consequence, we set $\gamma = 1.5$, $r_{max} = 7$, $a_{min} = -6$ and $a_{max} = 6$. For the $c = 5$ case, we set $\gamma = 2$, $r_{max} = 8$, $a_{min} = -7$ and $a_{max} = 7$.

The results in Table 3, are compared to those one reported in [1] based on Crude MC, FORM, Subset, Au and Beck, Meta-IS and MetaAK-IS^2 methods.

As it can be observed, MetaAL-OIS behaves systematically much better than the other metamodel-based method available for this example (Meta-IS^2) for each c. In fact, the parameter $\hat{\delta}_{p_{\tilde{f}}}$ is much lower with much less points for building the metamodel. Indeed, the result is always very close to the reference value, computed with *Perf+IS* (considered as the reference in this paper).

Concerning the computation of the unbiased estimation, it can be noted that the CV+IS and IS unbiased estimators return similar COV. For $c = 3, 4, 5$, the CV+IS unbiased estimator is the one providing the lowest COV, at 1% for 524, 722 and 580 total performance calls, respectively. Note that these results are systematically better than the ones given by Meta-IS, which provides a 5% COV with 644 total performance calls for $c = 3, 4$, and with 2940 calls for $c = 5$. With $c = 3$ ($c = 4$), consider also that CV+IS estimator returns a $\hat{p}_f = 3.37 \times 10^{-3}$ ($\hat{p}_f = 8.87 \times 10^{-5}$) with a COV of 2.4% (4.2%) for only 244 (268) total performance calls. Note also that the result provided by MetaAL-OIS, is always very close to the reference value, computed with *Perf+IS* (considered as the reference in this paper), and involving much less points for building the metamodel.

We note that the bias in the result given by the IS estimator is quite high for $c = 5$, so the $3-\sigma$ interval prediction do not contain the *reference* value p_f. In general, the CV+IS estimator is more robust, even if sometimes the estimated error $\hat{p}_f$ is higher than the one provided by IS. This example suggests that the user should always check the $3-\sigma$ interval prediction returned by both estimator, and select the one provided by CV+IS if those intervals are very different. With the IS method mentioned in 3.2 using a gaussian mixture density as the ISD, a coefficient of variation



(a)

(b)

(c)

(d)

Figure 1: One Failure Example (Case 1): Metamodel limit-state $\{\tilde{G}(\mathbf{x}) = 0\}$ (dashed black line), Exact limit-state $\{G(\mathbf{x}) = 0\}$ (red line). (a) Contour of quasi-optimal density h^*_π, LHS points (blue crosses), DOE adaptively added (green square). (b) Optimal Density function h^*_G contour. (c) Gaussian Mixture Density contour $h_{\mathcal{N}}$. (d) π function contour.

of 0.10% is reached on the computation of the low probability (for $c = 5$) $p_{\tilde{f}} = 9.03 \times 10^{-7}$ in only 9.45×10^6 metamodel evaluations. To reach the same accuracy, MCS method would require around 10^{12} metamodel evaluations.

Figures 2 (resp. 3) illustrates the limit-state branch and the approximation with the metamodel for $c = 3$ (resp. $c = 5$). The metamodel captures very well the failure branches. The optimal ISD h^*_f seems to be very well approximated by the quasi-optimal ISD h^*_π. The gaussian mixture ISD mimics well the behaviour of the quasi-optimal ISD, involving a non-zero density mostly in the failure region, and covering the true limit-state region. Then, $\pi(\mathbf{x})$ is a good surrogate of $\mathbb{1}_{G<0}(\mathbf{x})$ except in too distant regions, which are not interesting since characterized by a too low density.

Finally, it is worth noting that in this test-case the MetaAL-OIS features better performances than any of the other methods considered here.

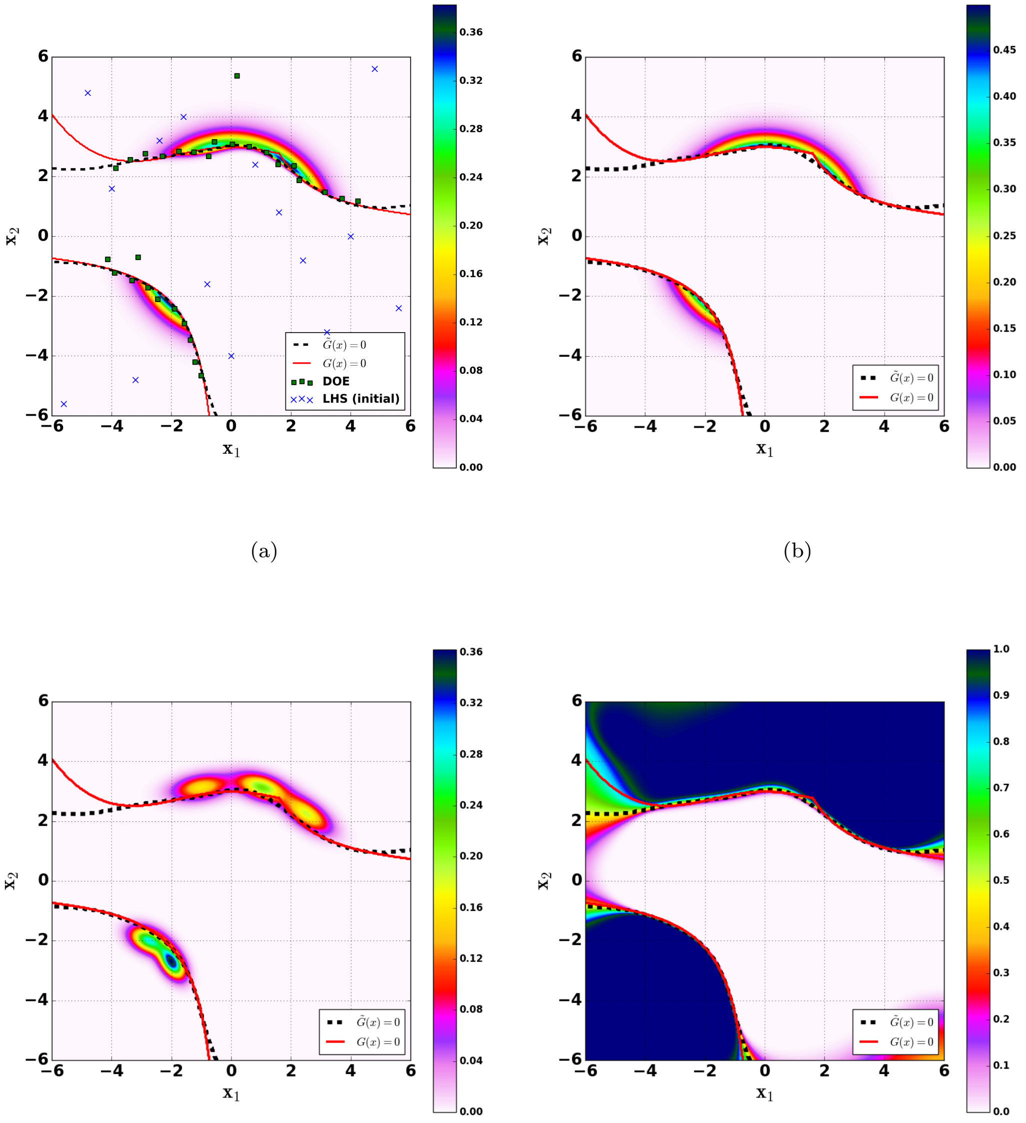
	Metamodel-based Estimation				Unbiased Estimation			
Method	N_{calls}	$\hat{p}_{\tilde{G}}$	$\hat{\delta}_{p_{\tilde{f}}}$	$3 - \hat{\sigma}_{p_{\tilde{f}}}$ Interval	N_{calls}	$\hat{p}_f$	$\hat{\delta}_f$	$3 - \hat{\sigma}_f$ Interval
c = 3								
Crude MC					$120,000$	3.35×10^{-3}	$< 5\%$	$\subseteq [2.85, 3.85] \times 10^{-3}$
Perf + IS					*1.3×10^6*	*3.47×10^{-3}*	*0.10%*	*$[3.47, 3.49] \times 10^{-3}$*
FORM					7	1.35×10^{-3}		
Subset					$300,000$	3.48×10^{-3}	$< 3\%$	$\subseteq [3.17, 3.80] \times 10^{-3}$
Au and Beck					600	2.47×10^{-3}	8%	$[1.88, 3.06] \times 10^{-3}$
Meta-IS					$44 + 600$	3.54×10^{-3}	$< 5\%$	$\subseteq [3.00, 4.07] \times 10^{-3}$
MetaAK-IS^2	117	3.47×10^{-3}	$< 5\%$	$\subseteq [2.94, 3.99] \times 10^{-3}$				
MetaAL-OIS	**44**	**3.53×10^{-3}**	**0.10%**	**$[3.52, 3.54] \times 10^{-3}$**	**$44 + 480$**	**3.46×10^{-5}**	**1.00%**	**$[3.35, 3.56] \times 10^{-5}$**
MetaAL-OIS (IS)					$44 + 480$	3.52×10^{-3}	1.02%	$[3.41, 3.63] \times 10^{-3}$
MetaAL-OIS (CV+IS)					$44 + 480$	3.46×10^{-3}	1.00%	$[3.35, 3.56] \times 10^{-3}$
c = 4								
Crude MC					$4,620,000$	8.68×10^{-5}	$< 5\%$	$\subseteq [7.38, 9.98] \times 10^{-5}$
Perf + IS					*1.6×10^6*	*8.99×10^{-5}*	*0.10%*	*$[8.97, 9.02] \times 10^{-5}$*
FORM					7	3.17×10^{-5}		
Subset					$500,000$	8.34×10^{-5}	$< 4\%$	$\subseteq [7.34, 9.34] \times 10^{-5}$
Au and Beck					600	6.51×10^{-5}	10%	$[4.56, 8.46] \times 10^{-5}$
Meta-IS					$64 + 600$	8.60×10^{-5}	$< 5\%$	$\subseteq [7.31, 9.89] \times 10^{-5}$
MetaAK-IS^2	118	8.49×10^{-5}	$< 5\%$	$\subseteq [7.22, 9.76] \times 10^{-5}$				
MetaAL-OIS	**68**	**8.86×10^{-5}**	**0.10%**	**$[8.83, 8.89] \times 10^{-5}$**	**722**	**8.92×10^{-5}**	**1.00%**	**$[8.65, 9.19] \times 10^{-5}$**
MetaAL-OIS (IS)					$68 + 722$	8.92×10^{-5}	1.00%	$[8.65, 9.19] \times 10^{-5}$
MetaAL-OIS (CV+IS)					$68 + 722$	8.88×10^{-5}	1.17%	$[8.57, 9.19] \times 10^{-5}$
c = 5								
Crude MC					$422,110,000$	9.48×10^{-7}	$< 5\%$	$\subseteq [8.06, 11.9] \times 10^{-7}$
Perf + IS					*1.8×10^6*	*8.97×10^{-7}*	*0.10%*	*$[8.95, 9.00] \times 10^{-7}$*
FORM					7	2.87×10^{-7}		
Subset					$700,000$	6.55×10^{-7}	$< 5\%$	$\subseteq [5.57, 7.53] \times 10^{-7}$
Au and Beck					600	6.54×10^{-7}	12%	$[4.19, 8.90] \times 10^{-7}$
Meta-IS			$< 5\%$		$40 + 2900$	9.17×10^{-7}	$< 5\%$	$\subseteq [7.80, 10.5] \times 10^{-7}$
MetaAK-IS^2	236	8.16×10^{-7}	$< 5\%$	$\subseteq [6.94, 9.38] \times 10^{-7}$				
MetaAL-OIS	**84**	**9.03×10^{-7}**	**0.10%**	**$[9.00, 9.05] \times 10^{-7}$**	**$84 + 496$**	**8.91×10^{-7}**	**1.00%**	**$[8.64, 9.18] \times 10^{-7}$**
MetaAL-OIS (IS)					$84 + 496$	9.61×10^{-7}	1.28%	$[9.24, 9.98] \times 10^{-7}$
MetaAL-OIS (CV+IS)					$84 + 496$	8.91×10^{-7}	1.00%	$[8.64, 9.18] \times 10^{-7}$

Table 3: Comparison of the performances of the MetaAL-OIS with several algorithms of literature[1]: Case 2.

4.3 2D analytic example with four failure regions: Case 3

Let us consider now a test-case with four failure regions [1, 12]. The performance function in the standard space reads:

$$G(x_1, x_2) = \min \left\{ \begin{array}{c} 3 + \frac{(x_1 - x_2)^2}{10} - \frac{x_1 + x_2}{\sqrt{2}} \\ 3 + \frac{(x_1 - x_2)^2}{10} + \frac{x_1 + x_2}{\sqrt{2}} \\ x_1 - x_2 + \frac{7}{\sqrt{2}} \\ -(x_1 - x_2) + \frac{7}{\sqrt{2}} \end{array} \right\} \tag{41}$$



(a)

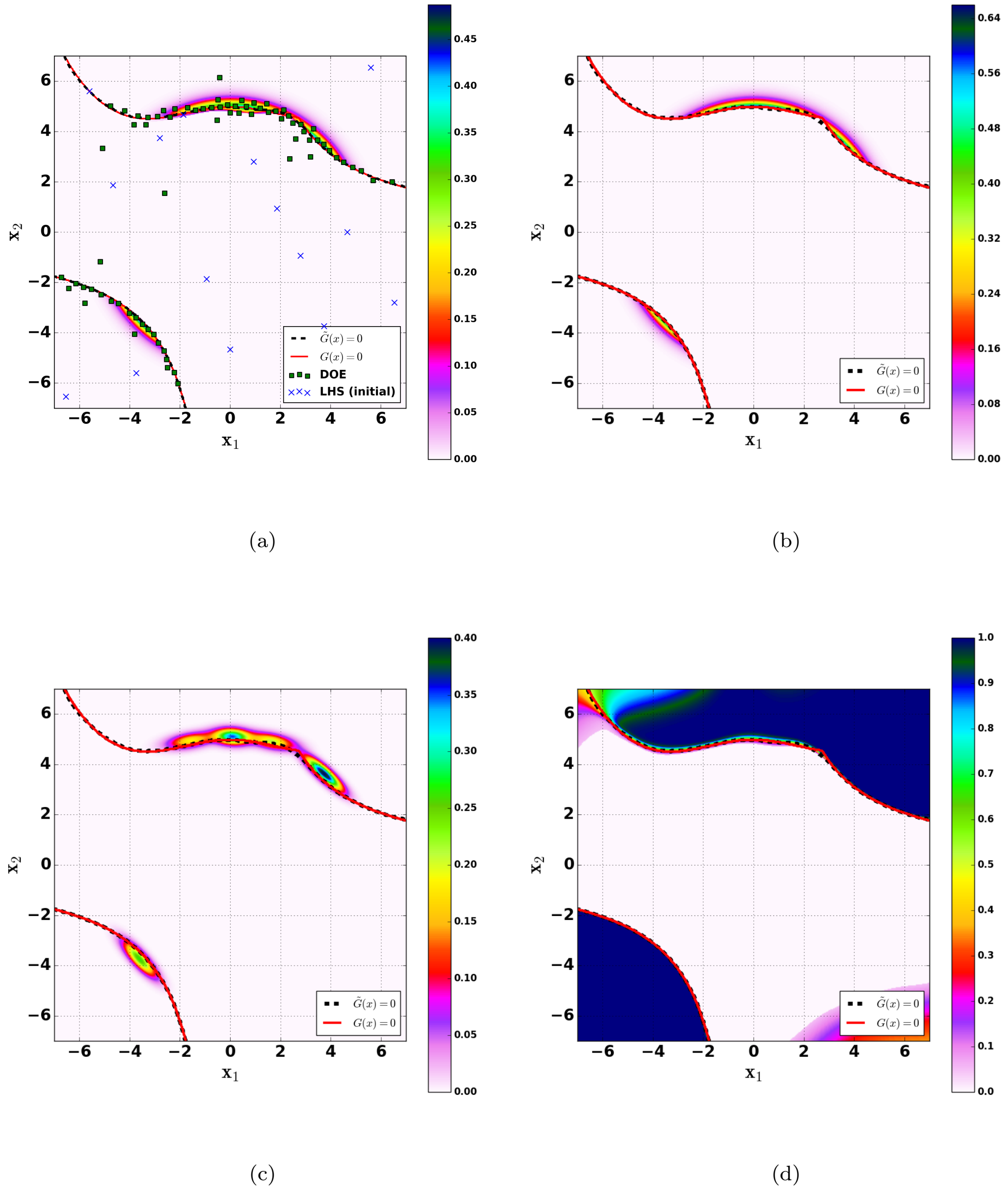
(b)

(c)

(d)

Figure 2: Two Failure Example - $c = 3$ (Case 2): Metamodel limit-state $\{\tilde{G}(\mathbf{x}) = 0\}$ (dashed black line), Exact limit-state $\{G(\mathbf{x}) = 0\}$ (red line). (a) Contour of quasi-optimal density h^*_π, LHS points (blue crosses), DOE adaptively added (green square). (b) Optimal Density function h^*_G contour. (c) Gaussian Mixture Density contour $h_{\mathcal{N}}$. (d) π function contour.

where x_1, x_2 are the realizations of two independent standard Gaussian random variables. The results in Table 4, are compared to those ones reported in [8][1] based on the following



(a)

(b)

(c)

(d)

Figure 3: Two Failure Example - $c = 5$ (Case 2): Metamodel limit-state $\{\tilde{G}(\mathbf{x}) = 0\}$ (dashed black line), Exact limit-state $\{G(\mathbf{x}) = 0\}$ (red line). (a) Contour of quasi-optimal density h^*_π, LHS points (blue crosses), DOE adaptively added (green square). (b) Optimal Density function h^*_G contour. (c) Gaussian Mixture Density contour $h_\mathcal{N}$. (d) π function contour.

methods: crude MC, FORM, DS, Subset, SMART, MetaAK-IS^2 and AK-MCS+U.

Compared with the other metamodel-based methods available for this example (MetaAK-IS^2

and AK-MCS+U), it seems that MetaAL-OIS features the best performances, in terms of COV and global number of calls. Moreover, the result is always very close to the reference value, involving less training points for building the metamodel.

In this case, the IS unbiased estimator returns the lowest COV: 1% for 575 total performance calls. Consider that with 269 total performance calls, it gives a $\hat{p}_f = 2.17 \times 10^{-3}$ with a COV of only 3.1%. Finally, note that MetaAL-OIS provides the best unbiased estimation with respect to the other methods considered here.

The capacity of the metamodel to capture the limit-state branch is illustrated in Figure 4. The parameters K, K^{IS}, K^{init} are fixed at 5, which is higher than the true number of failure regions. We note that the gaussian mixture ISD has five centers, whose two are located in one branch. Again, the surrogate π is well approximated except for distant regions characterized by negligible densities. The DOE used for the metamodel is well focused on the failure branches. The optimal ISD h_f^* seems to be very well approximated by the quasi-optimal ISD h_π^*.

	Metamodel-based Estimation				Unbiased Estimation			
Method	N_{calls}	$\hat{p}_{\tilde{G}}$	$\hat{\delta}_{p_{\tilde{f}}}$	$3-\hat{\sigma}_{p_{\tilde{f}}}$ Interval	N_{calls}	$\hat{p}_f$	$\hat{\delta}_f$	$3-\hat{\sigma}_f$ Interval
Crude MC					$781,016$	2.24×10^{-3}	2.23%	$[2.09, 2.39] \times 10^{-3}$
Perf + IS					2×10^6	2.22×10^{-3}	*0.10%*	$[2.21, 2.23] \times 10^{-3}$
FORM					7	1.35×10^{-3}		
DS					1800	2.22×10^{-3}		
Subset					$600,000$	2.22×10^{-3}	1.5%	$[2.12, 2.32] \times 10^{-3}$
SMART					1035	2.21×10^{-3}		
MetaAK-IS^2	138	2.22×10^{-3}	1.7%	$[2.11, 2.33] \times 10^{-3}$				
AK-MCS+U	96	2.23×10^{-3}						
MetaAL-OIS	**69**	$\mathbf{2.21 \times 10^{-3}}$	**0.10%**	$[\mathbf{2.20}, \mathbf{2.22}] \times \mathbf{10^{-3}}$	$\mathbf{69 + 506}$	$\mathbf{2.21 \times 10^{-3}}$	**1.00%**	$[\mathbf{2.15}, \mathbf{2.23}] \times \mathbf{10^{-3}}$
MetaAL-OIS (IS)					$69 + 506$	2.21×10^{-3}	1.00%	$[2.15, 2.23] \times 10^{-3}$
MetaAL-OIS (CV+IS)					$69 + 506$	2.21×10^{-3}	1.22%	$[2.13, 2.29] \times 10^{-3}$

Table 4: Comparison of the performances of the MetaAL-OIS with several algorithms of literature[1]: Case 3.

4.4 2D analytic "tricky" example with multiple failure regions: modified Rastrigin function. Case 4

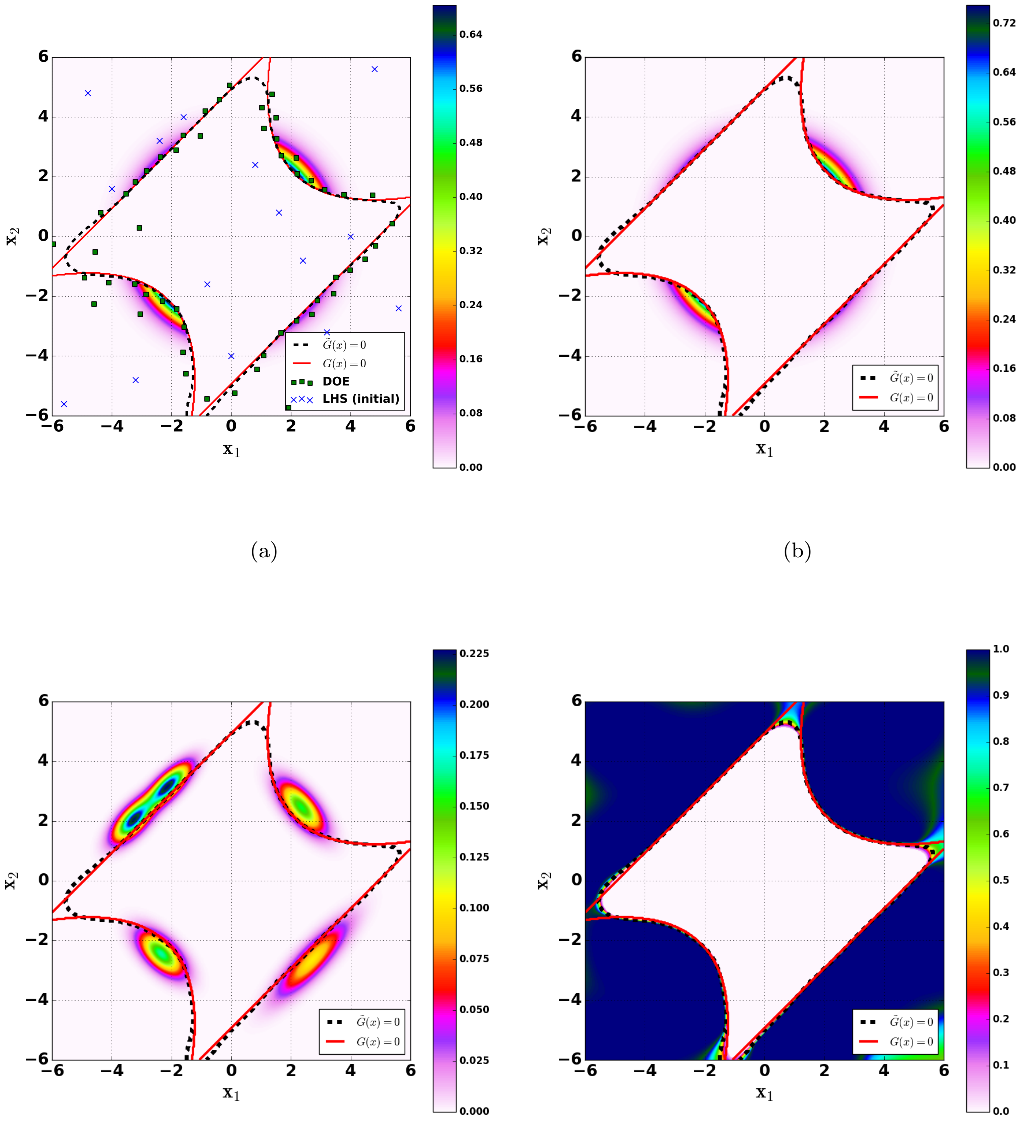
This test-case deals with a highly non-linear function involving non-convex and non-connected domains of failure (*i.e. "scattered gaps of failure"*). This example is tricky because of the numerous number of failure regions. However, they are close, and the failure probability is not very low. The performance function [1] [12] in the standard space reads as follows:

$$G(x_1, x_2) = 10 - \sum_{i=1}^{2} (x_i^2 - 5\cos(2\pi x_i)) \tag{42}$$

where x_1, x_2 are the realizations of two independent standard Gaussian random variables.

In this test-case, the tuning parameters K, K^{IS} and K^{init} are fixed at 50. Note that γ is set to 1, because the failure probability is not very low and a reasonable number of Monte-Carlo points should include failure points.

Table 5 illustrates the comparison between MetaAL-OIS and the methods reported in [1]: crude MC, FORM, AK-MCS and MetaAK-IS^2. It can be observed that MetaAL-OIS behaves better than the other metamodel-based methods available for this example (MetaAK-IS^2 and AK-MCS). In fact, a lower COV is associated to a lower number of functional evaluation as well. Moreover, the result is very close to the reference value.



(a) (b)

(c) (d)

Figure 4: Four Failure Regions Example (Case 3): Metamodel limit-state $\{\tilde{G}(\mathbf{x}) = 0\}$ (dashed black line), Exact limit-state $\{G(\mathbf{x}) = 0\}$ (red line). (a) Contour of quasi-optimal density h_π^*, LHS points (blue crosses), DOE adaptively added (green square). (b) Optimal Density function h_G^* contour. (c) Gaussian Mixture Density contour $h_\mathcal{N}$. (d) π function contour.

In this case, the CV+IS unbiased estimator returns the lowest COV, *i.e.* 0.10% for 351 total performance calls, while the IS estimator features a worse performance (*i.e.* 2.15% for the same

351 calls).

As done in the other test-cases, we then compare the limit-state branch and the metamodel behavior (see Figure 5). Also in this challenging case, a very good performances can be observed. In particular, the metamodel is able to capture and represent several domains of failure.

	Metamodel-based Estimation				Unbiased Estimation			
Method	N_{calls}	$\hat{p}_{\tilde{G}}$	$\hat{\delta}_{p_{\tilde{f}}}$	$3-\hat{\sigma}_{p_{\tilde{f}}}$ Interval	N_{calls}	$\hat{p}_f$	$\hat{\delta}_f$	$3-\hat{\sigma}_f$ Interval
Crude MC [1]					$25,000$	7.43×10^{-2}	2.23%	$[6.93, 7.93]\times10^{-2}$
Perf + IS					$\mathit{1.3\times10^{6}}$	$\mathit{7.29\times10^{-2}}$	*0.10%*	$\mathit{[7.27, 7.32]\times10^{-2}}$
FORM					20	6.83×10^{-6}		
AK-MCS	391	7.43×10^{-2}	2.23%	$[6.93, 7.93]\times10^{-2}$				
MetaAK-IS^2	480	7.35×10^{-2}	2.5%	$[6.80, 7.90]\times10^{-2}$				
MetaAL-OIS	**151**	$\mathbf{7.31\times10^{-2}}$	**0.10%**	$\mathbf{[7.28, 7.33]\times10^{-2}}$	$\mathbf{151+200}$	$\mathbf{7.31\times10^{-2}}$	**0.10%**	$\mathbf{[7.28, 7.33]\times10^{-2}}$
MetaAL-OIS (IS)					$151+200$	7.11×10^{-2}	2.15%	$[6.65, 7.57]\times10^{-2}$
MetaAL-OIS (CV+IS)					$151+200$	7.31×10^{-2}	0.10%	$[7.28, 7.33]\times10^{-2}$

Table 5: Comparison of the performances of the MetaAL-OIS with several algorithms of literature[1]: Case 4.

4.5 Case 5: 6D analytic example in the physical space: dynamic response of a non-linear oscillator

This test-case is a problem with six random variables in the physical space. The performance function is smooth with respect to the considered inputs. It consists of a non-linear undamped single degree of freedom system [8]. In particular, the performance function is given as follows:

$$G(c_1, c_2, m, r, t_1, F_1) = 3r - \left| \frac{2F_1}{m\omega_0^2} sin\left(\frac{\omega_0 t_1}{2}\right) \right| \tag{43}$$

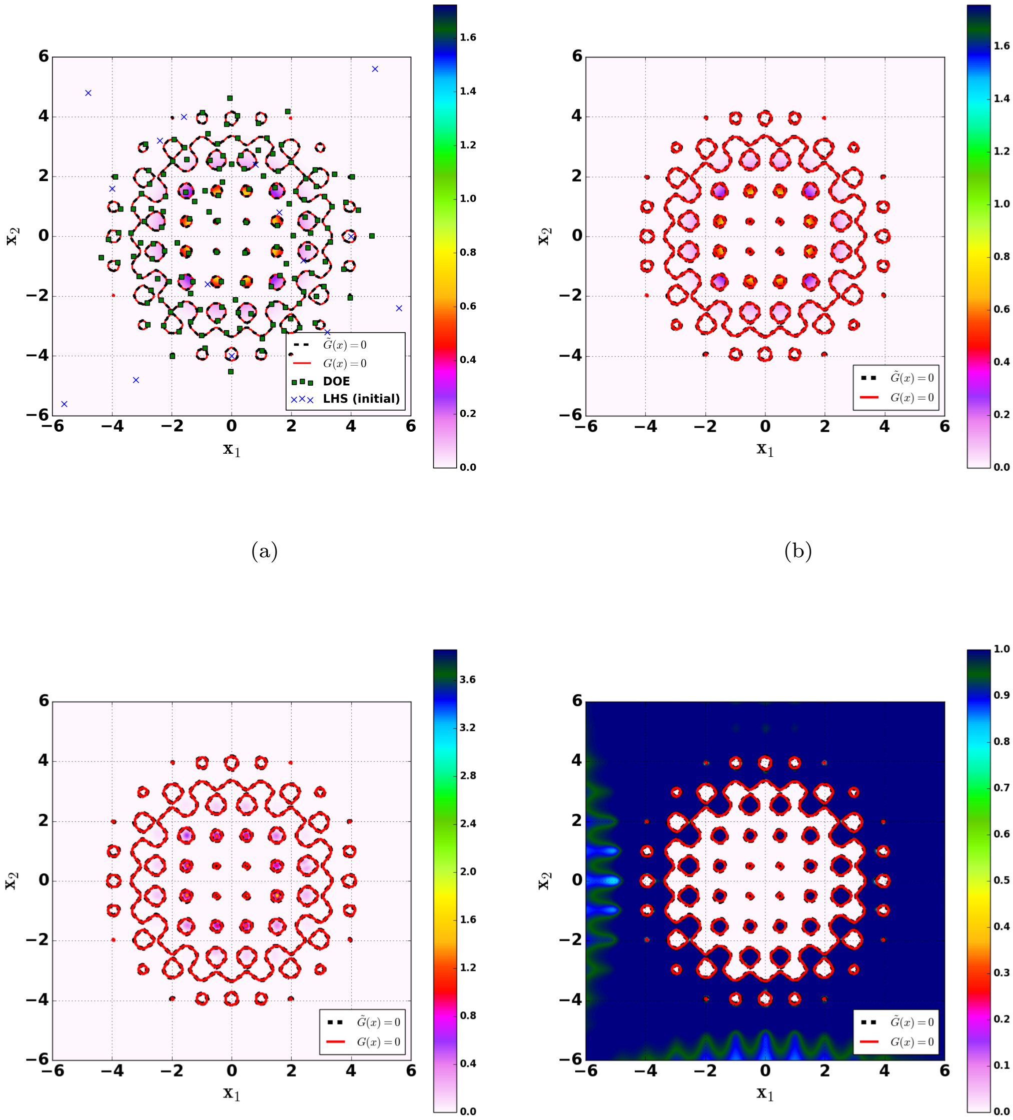
with $\omega_0 = \sqrt{\dfrac{c_1 + c_2}{m}}$. The six random variables are listed in Table 6. The difficulty here comes from the relatively high dimension of the problem, adding the fact that the input variable belong to the physical space.

The proposed approach MetaAL-OIS is compared to MCS and other metamodels based methods [8]; only the ones that returned a fairly good estimation of p_f are mentioned. Note that C.O.V are not available in [8] for the computation associated to $p_{\tilde{f}}$, estimated with metamodel-based methods, so not quoted in Table 7. MetaAL-OIS metamodel requires 70 performance evaluations, which is slightly more than for the AK-MCS+EFF (only 45), with a very good accuracy. In fact, the COV is 0.10% for MetaAL-OIS metamodel, which is very low. Finally, performances of the MetaAL-OIS method seem very good also in this case, even if a proper comparison can not be done since only partial data about the performances of the other methods are available.

Concerning the unbiased estimation, note that only 228 additional evaluations lead to a sharp unbiased estimation with $\hat{\delta}_f = 1\%$. Again, MetaAL-OIS gives very satisfactory results with respect to other methods, for a low computational effort.

5 Conclusion

In this paper, we have proposed a metamodel-based method for the computation of tail probabilities, suitable for very low probability and/or multiple failure regions, also able to return an accurate unbiased estimation of the failure probability with few additional performance function calls.



(a)

(b)

(c)

(d)

Figure 5: Multiple Failure Regions Example (Case 4): Metamodel limit-state $\{\tilde{G}(\mathbf{x}) = 0\}$ (dashed black line), Exact limit-state $\{G(\mathbf{x}) = 0\}$ (red line). (a) Contour of quasi-optimal density h_π^*, LHS points (blue crosses), DOE adaptively added (green square). (b) Optimal Density function h_G^* contour. (c) Gaussian Mixture Density contour $h_\mathcal{N}$. (d) π function contour.

In particular, we have proposed a significant improvement of the Meta-IS algorithm developed by Dubourg [12]. The main improvement is obtained by modifying the metamodel construction.

Variable	P.D.F.	Mean	Standard Deviation
m	Normal	1	0.05
c_1	Normal	1	0.1
c_2	Normal	0.1	0.01
r	Normal	0.5	0.05
F_1	Normal	1	0.2
t_1	Normal	1	0.2

Table 6: Random Variables [8] of Case 5.

	Metamodel-based Estimation		Unbiased Estimation			
Method	N_{calls}	$\hat{p}_{\tilde{G}}$	N_{calls}	$\hat{p}_f$	$\hat{\delta}_f$	$3-\hat{\sigma}_f$ Interval
MCS			$70,000$	2.834×10^{-2}	2.2%	$[2.647, 3.021] \times 10^{-2}$
Perf + IS			*2.9×10^6*	*2.856×10^{-2}*	*0.10%*	*$[2.846, 2.865] \times 10^{-2}$*
AK-MCS+U	58	2.834×10^{-2}				
AK-MCS+EFF	45	2.851×10^{-2}				
DS + Neural Network	86	2.8×10^{-2}				
Importance Sampling (IS)			6144	2.7×10^{-2}		
IS + Response Surface	109	2.5×10^{-2}				
IS + Spline	67	2.7×10^{-2}				
IS + Neural Network	68	3.1×10^{-2}				
MetaAL-OIS	**70**	$\mathbf{2.847 \times 10^{-2}}$	$\mathbf{70 + 228}$	$\mathbf{2.848 \times 10^{-2}}$	**1.00%**	$\mathbf{[2.763, 2.934] \times 10^{-2}}$
MetaAL-OIS (IS)			$70 + 228$	2.848×10^{-2}	1.00%	$[2.763, 2.934] \times 10^{-2}$
MetaAL-OIS (CV+IS)			$70 + 228$	2.880×10^{-2}	1.16%	$[2.781, 2.981] \times 10^{-2}$

Table 7: Comparison of the performances of the MetaAL-OIS with several algorithms of literature[8]: Case 5.

This refinement strategy permits to deal with a metamodel which has a similar accuracy as the one provided using AK-MCS [8], with significantly less points, though. It is suitable for multiple failure regions. Moreover, an original IS technique is developed, leading to very good estimation of p_f, only by sampling the surrogate. Unlike in AK-IS [9] or MetaAK-IS^2 [1], the gaussian mixture ISD $h^*_{\mathcal{N}}$ is most likely to sample points in regions in which the optimal ISD h^* would do. As a consequence, very low probability can be handled. Eventually, in order to quantify the error made by the metamodel, IS involving the ISD h^*_π is a very efficient variance reduction technique since $\tilde{G}$ fits G in the limit-state more accurately than in Meta-IS.

Finally, the way the metamodel is built offers robustness in case of complex limit-states, because it avoids points clustering (then building the metamodel is easier), and refines back and forth the different failure branches. If the number of sample of the DOE is so large that the metamodel can not be refined anymore, the user can stop the metamodel refinement strategy but still have quite a good representation of the original performance function.

Additionally, to obtain an unbiased failure estimation, we have used directly the second part of the Meta-IS [12], and proposed another unbiased estimator. It is slightly different and does not induce additional computations. Moreover, in some cases, it can further significantly reduce the number of performance function calls. Among methods returning an unbiased estimation, we have showed that MetaAL-OIS permits a lower computational cost for a better accuracy.

We have showed the ability of the method to deal with very low probability and multiple failure regions by performing several test-cases. The construction of the metamodel requires in general significantly less computational calls than other metamodel-based methods, and preserves the global accuracy. Then, the metamodel construction and the gaussian mixture IS method can be an accurate and a general alternative to AK-MCS, MetaAK-IS^2, KAIS and AK-SS. In fact, it provides a surrogate-based analysis, using the metamodel instead of the original limit-state function, dealing with multiple failure regions and low probability.

Moreover, we have also described and assessed a method allowing to reduce significantly the number of metamodel calls, when estimating the failure probability from the metamodel (which can be very large if p_f is very low), with respect to MCS.

Future work will be directed towards the automatic tuning of the proposed algorithm's parameters, especially the ones with the major impact on the efficiency of the method: d_{min} and K.

References

[1] E. Zio F. Cadini, F. Santos. An improved adaptive kriging-based importance technique for sampling multiple failure regions of low probability. *Reliability Engineering and System Safety*, 131:109–117, 2014.

[2] R. Y. Rubinstein. *Simulation and the Monte-Carlo method.* New York: Wiley, 1981.

[3] G. S. Fishman. *Monte-Carlo: concepts, algorithms and applications.* New York: Wiley, 1996.

[4] U. Bourgund C. Bucher. A fast and efficient response surface approach for structural reliability problems. *Struct Saf*, 7:57–66, 1990.

[5] P. S. Spanos R. G. Ghanem. *Stochastic finite elements: a spectral approach.* Berlin: Springer, 1991.

[6] N. Lagaros M. Papadrakakis. Reliability-based structural optimization using neural networks and monte carlo simulation. *Comput Methods Appl Mech Eng*, 191(32):3491–507, 2002.

[7] L. Swiler B. Bichon, M. Eldred and S. Mahadevan. Efficient global reliability analysis for nonlinear implicit performance functions. *AIAA J*, 46(10):2459–68, 2005.

[8] M. Lemaire B. Echard, N. Gayton. Ak-mcs: An active learning reliability method combining kriging and monte carlo simulation. *Reliability Engineering and System Safety*, 33:145–154, 2011.

[9] M. Lemaire B. Echard, N. Gayton and N. Relun. A combined inportance sampling and kriging reliability method for small failure probabilities with time-demanding numerical models. *Reliability Engineering and System Safety*, 111:232–240, 2013.

[10] Y. Liu H. Zhao, Z. Yue and Z. Gao. An efficient reliability method combining adaptive importance sampling and kriging metamodel. *Applied Mathematical Modelling*, 39:1853–66, 2015.

[11] J. Chen X. Huang. Assessing small failure probabilities by akâĂŞss: An active learning method combining kriging and subset simulation. *Structural Safety*, 59:86–95, 2016.

[12] F. Deheeger V. Dubourg, B. Sudret. Metamodel-based importance sampling for structural reliability analysis. *Probabilistic Engineering Mechanics*, 33:47–57, 2013.

[13] E. Zio F. Cadini, A. Gioletta. Improved metamodel-based importance sampling for the performance assessment of radiactive waste repositories. *Reliability Engineering and System Safety*, 134:188–197, 2015.

[14] J. MacQueen. *Some methods for classification and analysis of multivariate observations.* Proceedings of the Berkeley symposium on mathematical statistics and probability, vol.1. Berkeley, 1967.

[15] W. Notz T. Santner, B. Williams. *The design and analysis of computer experiments.* New York: Springer, 2013.

[16] C. Williams C. E. Rasmussen. *Gaussian Processes for Machine Learning.* the MIT Press, 2006.

[17] GPy. GPy: A gaussian process framework in python. `http://github.com/SheffieldML/GPy`, since 2012.

[18] J. Beck S. K. Au. Estimation of small failure probabilities in high dimensions by subset simulation. *Prob. Eng. Mech.*, 16:263–277, 2001.

RESEARCH CENTRE
BORDEAUX – SUD-OUEST

351, Cours de la Libération
Bâtiment A 29
33405 Talence Cedex

Publisher
Inria
Domaine de Voluceau - Rocquencourt
BP 105 - 78153 Le Chesnay Cedex
inria.fr

ISSN 0249-6399