



**HAL**  
open science

## Recommendations for the tuning of rare event probability estimators

Mathieu Balesdent, Jérôme Morio, Julien Marzat

► **To cite this version:**

Mathieu Balesdent, Jérôme Morio, Julien Marzat. Recommendations for the tuning of rare event probability estimators. *Reliability Engineering and System Safety*, 2015, 133, pp.68-78. 10.1016/j.ress.2014.09.001 . hal-01064029

**HAL Id: hal-01064029**

**<https://hal.science/hal-01064029v1>**

Submitted on 15 Sep 2014

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

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

# Author's Accepted Manuscript

Recommendations for the tuning of rare event probability estimators

Mathieu Balesdent, Jérôme Morio, Julien Marzat



PII: S0951-8320(14)00211-7  
DOI: <http://dx.doi.org/10.1016/j.ress.2014.09.001>  
Reference: RESS5128

To appear in: *Reliability Engineering and System Safety*

Received date: 13 December 2011  
Revised date: 20 January 2014  
Accepted date: 1 September 2014

Cite this article as: Mathieu Balesdent, Jérôme Morio, Julien Marzat, Recommendations for the tuning of rare event probability estimators, *Reliability Engineering and System Safety*, <http://dx.doi.org/10.1016/j.ress.2014.09.001>

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting galley proof before it is published in its final citable form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

# Recommendations for the tuning of rare event probability estimators

Mathieu Balesdent<sup>a,\*</sup>, Jérôme Morio<sup>b</sup>, Julien Marzat<sup>a</sup>

<sup>a</sup>*Onera – The French Aerospace Lab, F-91761 Palaiseau, France*

<sup>b</sup>*Onera – The French Aerospace Lab, F-31055 Toulouse, France*

---

## Abstract

Being able to accurately estimate rare event probabilities is a challenging issue in order to improve the reliability of complex systems. Several powerful methods such as importance sampling, importance splitting or extreme value theory have been proposed in order to reduce the computational cost and to improve the accuracy of extreme probability estimation. However, the performance of these methods is highly correlated with the choice of tuning parameters, which are very difficult to determine. In order to highlight recommended tunings for such methods, an empirical campaign of automatic tuning on a set of representative test cases is conducted for splitting methods. This allows to provide a reduced set of tuning parameters that may lead to the reliable estimation of rare event probability for various problems. The relevance of the obtained result is assessed on a series of real-world aerospace problems.

## *Keywords:*

Tuning of simulation parameters, Adaptive importance Splitting, rare event probability estimation, Kriging-based optimization

---

\*Corresponding author, email : mathieu.balesdent@onera.fr

---

## 1. Introduction

Reliability of complex systems can depend on several events whose probabilities of occurrence are very low but however may induce serious consequences (*e.g.* destruction of the system, injuries, *etc.*). Being able to accurately estimate the probability of such rare events is challenging but mandatory in order to improve the reliability of complex systems. Several powerful methods, such as importance sampling [1, 2, 3, 4, 5], importance splitting [6, 7, 8, 9] or extreme value theory [10, 11, 12, 13] have been proposed to estimate such probabilities when the classical Crude Monte Carlo method (CMC) requires too many system simulations and is not accurate enough to be applied in this context. However, the performances of these algorithms are very dependent on some parameters (*e.g.* number of samples, kernel stepsizes, intermediary quantiles, *etc.*) which have to be finely tuned to ensure the consistency of the results. Indeed, different combinations of tuning parameters for the same method may lead to scattered performance. Moreover, intuition may not be sufficient for adjusting simultaneously a significant number of these *simulation parameters* and their automatic tuning should thus be considered. A possible way to reach this goal is to consider this problem as the global optimization of a criterion reflecting the performance of the considered method. In this paper, we perform the optimal determination of tuning parameters on several representative test cases in order to propose some recommendations for the tuning of rare event estimator based on the importance splitting algorithm (also called subset simulation [14]).

The main existing tools for tuning the simulation parameters are *cross-validation* and its variants ( $k$ -fold cross-validation, leave-one-out cross-validation, generalized cross-validation; see [15]). Cross-validation may be used to assess the performance level for a given value of the simulation parameter vector and then an optimization procedure may be called upon to find the best tuning for these parameters. In [16, 17], such approaches have been exploited via a discretization of the simulation parameter space. Bayesian networks have also been advocated in [18], by considering previous simulation runs as prior knowledge. Various other techniques have been employed for tuning purpose, such as Monte Carlo simulations [19], neural networks [20] or evolutionary algorithms [21, 22]. In particular, the use of genetic algorithms should be mentioned with applications in control design [23] or reliability assessment [24]. Nevertheless, these approaches reveal to be computationally intensive, since they require a large exploration of the simulation parameter space. This may be prohibitive when the evaluation of the method performance is achieved via costly computer simulations, and even more when the number of those simulations is severely limited.

We propose instead to rely on a *Kriging*-based global optimization algorithm, coming from the field of *Computer Experiments* (see [25]) to provide optimal simulation parameters at a limited computational cost. For example, these tools have been successfully applied to the optimal tuning of fault detection strategies in [26]. The present paper reports the application of this strategy to the tuning of methods for the estimation of rare event probability, which does not seem to have been addressed in the literature so far. We focus our paper on the tuning of one of the most used rare event estimation

technique: the Adaptive importance Splitting Technique (AST).

This paper is organized as follows. In Section 2, we present the Adaptive importance Splitting Technique, and describe its tunings and their impact on the estimator performances. Section 3 is devoted to the presentation of the automatic tuning methodology. In Section 4, we describe and perform the tuning campaign, using our proposed methodology on several representative test cases in order to propose tuning recommendations. Finally, we apply in Section 5 the recommended tunings on two real-world test cases relative to the launch vehicle fall-back safety zone estimation and missile safety.

## 2. Extreme probability estimation using the Adaptive importance Splitting Technique

### 2.1. Rare event probability estimation

Let us consider a  $d$ -dimensional random variable  $X$  with a probability density function (PDF)  $f_0$  and estimate the probability that  $P(\phi(X) > S)$  with  $\phi$ , a continuous “input-output” scalar function  $\phi : \mathbb{R}^d \rightarrow \mathbb{R}$  and  $S$  a threshold. The function  $\phi$  simply represents the input-output system in a considered application domain. We assume in this article that  $\phi(X)$  is a random variable. A simple way to estimate this probability is to consider Monte Carlo methods [27, 28, 29, 30, 31]. For that purpose, one generates independent and identically distributed samples  $X_1, \dots, X_N$  from the PDF  $f_0$  and then estimates the probability with

$$P^{MC} = \frac{1}{N} \sum_{i=1}^N \mathbf{1}_{\phi(X_i) > S} \quad (1)$$

where  $\mathbf{1}_{\phi(X_i) > S}$  is equal to 1 if  $\phi(X_i) > S$  and 0 otherwise. The relative deviation of the Monte Carlo estimator  $P^{MC}$  is given by the ratio  $\frac{\sigma_{P^{MC}}}{P^{MC}}$  with

$\sigma_{PMC}$ , the standard deviation of  $P^{MC}$ . Knowing the true probability  $P$  that  $(\phi(X) > S)$ , one has

$$\frac{\sigma_{PMC}}{P^{MC}} = \frac{1}{\sqrt{N}} \frac{\sqrt{P - P^2}}{P} \quad (2)$$

Considering rare event probability estimation, that is when  $P$  takes low values, one has

$$\lim_{P \rightarrow 0} \frac{\sigma_{PMC}}{P^{MC}} = \lim_{P \rightarrow 0} \frac{1}{\sqrt{NP}} = +\infty \quad (3)$$

The relative deviation of Monte Carlo estimation is very high and thus, one can conclude that Monte Carlo methods are not well suited to rare event probability estimation. Different alternatives to Monte Carlo failure can be considered such as importance sampling [1, 2, 3, 4, 5], importance splitting [6, 7, 8, 9] or extreme value theory [10, 11, 12, 13].

## 2.2. Adaptive importance Splitting Technique

In this article, we focus on Adaptive importance Splitting Technique (AST). This algorithm, which is also known as Subset Simulation [32, 33], has been successfully applied into numerous probability estimation problems (e.g. [6, 7, 34]). The splitting algorithm used in this article is completely described in [35]. In the present article, we just recall the different notations and the principle of this algorithm. For more details, one can consult [35].

### 2.2.1. Principle

Considering the set  $\mathbf{A} = \{x \in \mathbb{R}^d | \phi(x) > S\}$ , the objective of this method is to determine the probability in this set  $\mathbf{A}$  since one has  $P(X \in \mathbf{A}) = P(\phi(X) > S)$ . The principle of AST is to iteratively estimate supersets of  $\mathbf{A}$  and then to estimate  $P(X \in \mathbf{A})$  with conditional probabilities.

Let us define  $\mathbf{A}_0 = \mathbb{R}^d \supset \mathbf{A}_1 \supset \dots \supset \mathbf{A}_{n-1} \supset \mathbf{A}_n = \mathbf{A}$ , a decreasing sequence of  $\mathbb{R}^d$  subsets with smallest element  $\mathbf{A} = \mathbf{A}_n$ . The probability  $P(X \in \mathbf{A})$  can be then rewritten in the following way :

$$P(X \in \mathbf{A}) = \prod_{k=1}^n P(X \in \mathbf{A}_k | X \in \mathbf{A}_{k-1})$$

where  $P(X \in \mathbf{A}_k | X \in \mathbf{A}_{k-1})$  is the probability that  $X \in \mathbf{A}_k$  knowing that  $X \in \mathbf{A}_{k-1}$ .

### 2.2.2. Defining the $\mathbf{A}_k$ sequence

The subset  $\mathbf{A}_k$  sequence is easily evaluated. Indeed, it can be defined by  $\mathbf{A}_k = \{x \in \mathbb{R}^d | (\phi(x) > S_k)\}$  for  $k = 0, \dots, n$  with  $S = S_n > S_{n-1} > \dots > S_k > \dots > S_0$ . Determining the sequence  $\mathbf{A}_k$  is equivalent to choose some values for  $S_k$ , with  $k = 0, \dots, n$ . Let us define  $f_k$ , the density of  $X$  restricted to the set  $\mathbf{A}_k$  and  $X_1^{(k)}, \dots, X_N^{(k)}$ ,  $N$  samples generated from the density  $f_k$ . The values of  $S_k$  for  $k = 0, \dots, n$  can be determined in an adaptive manner using  $\beta$  quantiles of  $\phi(X_1^{(k)}), \dots, \phi(X_N^{(k)})$ .

The number  $n$  of subsets depends on the value of  $\beta$  and the sought probability is then estimated with  $P = \beta^{n-1} * P(X \in \mathbf{A} | X \in \mathbf{A}_{n-1})$ . To efficiently define the  $\mathbf{A}_k$  sequence adaptively, the quantile parameter  $\beta$  and the sample size  $N$  have to be adjusted depending on the number of samples that are available in the simulation.

### 2.2.3. Generating samples from the density $f_k$

Unfortunately, generating directly independent samples from the  $f_k$  conditional densities is in most cases impossible as they are usually unknown [8, 36]. Nevertheless, AST provides an iterative way to do it, yet in a dependent



fashion using a  $f_0$ -reversible Markovian kernel  $K(X, \cdot)$ . With such a kernel and  $X_k \sim f_k$ , one can distribute random variable  $\Xi_k$  according to  $f_k$  with the following proposal/refusal method:

$$\Xi_k = \Xi_k(X_k) = \begin{cases} K(X^k, \cdot) & \text{if } K(X^k, \cdot) \in A_k \\ X^k & \text{otherwise} \end{cases}$$

The generated set does unfortunately not contain independent variables identically distributed according to  $f_k$ . However, under mild conditions, it can be shown [36] that applying the proposal/refusal method several times may decrease the variance. Let us consequently define the variable  $N_{app}$ , the number of kernel applications to generate independent samples following the density  $f_k$ . The most general and arduous way to determine such kernel is to solve a reversibility equation or using a Metropolis-Hastings algorithm to simulate it [36]. Nevertheless, there are some results for multivariate Gaussian PDF. Actually, if  $X$  is a centered multivariate Gaussian

$$X \sim \mathcal{N}(0_d, I_d) \text{ and } K(X, \cdot) \sim \frac{X + \alpha \mathcal{N}(0_d, I_d)}{\sqrt{1 + \alpha^2}} \quad (4)$$

is a valuable reversible Markovian kernel [8, 36]. For that purpose, in the following of this article, one will assume that the input random variable  $X$  has a Gaussian PDF. In this case, one just has to choose an appropriate value for  $\alpha$ , which is still rather difficult.

#### 2.2.4. Implementation

Let us summarize from the previous sections the different steps of AST to estimate  $P(\phi(X) > S) = P(X \in \mathbf{A})$  for a given simulation budget  $N_{max}$ .

- 1) Set  $k = 0$ ,  $j = 0$ , the used simulation budget  $N_{used} = 0$  and define the value of  $N$ ,  $\beta$ ,  $\alpha$  and  $N_{app}$ .

- 2) Generate  $N$  samples  $X_1^{(0)}, \dots, X_N^{(0)}$  from  $f_0(X)$  and then the corresponding samples  $\phi(X_1^{(0)}), \dots, \phi(X_N^{(0)})$ . The parameter  $N_{used}$  is updated with  $N_{used} = N_{used} + N$ .
- 3) Estimate the  $\beta$ -quantile  $q^{(0)}$  of  $\phi(X_1^{(0)}), \dots, \phi(X_N^{(0)})$ .
- 4) While  $q^{(k)} < S$  and  $N_{used} < N_{max}$ , do:
  - a) Replace  $\{X_i^{(k)} \mid \phi(X_i^{(k)}) < q^{(k)}\}$  simulations with points uniformly selected with replacement within  $\{X_i^{(k)} \mid \phi(X_i^{(k)}) > q^{(k)}\}$  and call the sample set  $Z_1^0, \dots, Z_N^0$
  - b) for  $j = 1$  to  $N_{app}$ , estimate the set  $Z_1^j, \dots, Z_N^j$  with
 
$$Z^j = \begin{cases} \frac{Z^{j-1} + \alpha \mathcal{N}(0_d, I_d)}{\sqrt{1 + \alpha^2}} & \text{if } \phi\left(\frac{Z^{j-1} + \alpha \mathcal{N}(0_d, I_d)}{\sqrt{1 + \alpha^2}}\right) > q^{(k)} \\ Z^{j-1} & \text{otherwise} \end{cases}$$
  - c) set  $X_l^{(k+1)} = Z_l^{N_{app}} \forall l = 1, \dots, N$  and estimate the  $\beta$ -quantile  $q^{(k+1)}$  of  $\phi(X_1^{(k+1)}), \dots, \phi(X_N^{(k+1)})$
  - d) Set  $k = k + 1$  and update  $N_{used} = N_{used} + N \times N_{app}$
- 5) Estimate the probability with

$$P^{AST} = (1 - \beta)^k \times \frac{1}{N_{max} - N_{used}} \sum_{i=1}^{N_{max} - N_{used}} \mathbf{1}_{\phi(X_i^{(k)}) > S}$$

If the simulation budget is reached before  $q^{(k)} < S$ , we consider that the AST fails to estimate the probability.

### 2.3. Impact of the choice of simulation parameters on the estimator efficiency

The previous section has described the principle of AST. One can remark that several simulation parameters have to be tuned by the user in order to get a valuable estimation:

- the number of samples  $N$ . The quantile  $q$  is indeed better estimated when the number of samples  $N$  at each iteration is high, but a too high value of  $N$  reduces the number of intermediary steps and consequently cancels the adaptability advantage of AST.
- the quantile  $\beta$ . If the value of  $\beta$  is low, the convergence of AST may not be reached with the given simulation budget. On the contrary, if  $\beta$  is too high, the quantiles  $q^{(k)}$  can be tough to estimate for a given sample size  $N$ .
- the kernel stepsize  $\alpha$ . If too much kernel transitions are accepted, the simulation parameter  $\alpha$  is probably too high and *vice versa*.
- the number of kernel applications  $N_{app}$ . If  $N_{app}$  is too low, the samples are dependent and the splitting estimation can be biased. On the contrary, a large value of  $N_{app}$  is not very efficient. New samples are generated and consume the overall simulation budget even if the samples are already independent and thus do not really improve the probability estimation.

The trade-off between these different simulation parameters is very difficult to find. This choice is not obvious even in some very simple cases, as shown in the following subsection.

### 2.3.1. Illustration on a toy case

The toy case deals with the estimation of the probability  $P(\phi(X) > S)$  where  $\phi$  is the identity function and the threshold  $S$  is equal to 4 (Figure 1). The initial distribution is the unidimensional normal distribution with zero

mean and unit standard deviation  $\mathcal{N}(0, 1)$ . In order to point out the impact of the tuning choice on the estimator behavior, we have tested 100 tuning combinations determined using Latin Hypercube Sampling (LHS) [37]. The variation domains of the tunings are given in Table 1.

Parameter	Variation domain
$N$	[500,50000]
$N_{app}$	[1,20]
$\beta$	[0.1,0.95]
$\alpha$	[0.1,1]

Table 1: Variation domains of the tuning parameters

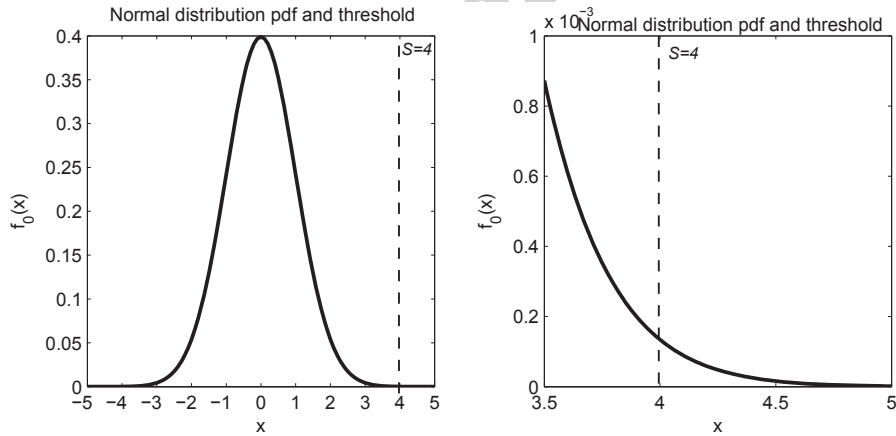


Figure 1: Normal distribution pdf and selected threshold

For each combination of tunings, we perform 100 estimations of  $P(\phi(X) > S)$  using AST and we compute the relative standard deviation with respect to the theoretical value of the desired probability using Equation 5 (performance

index). The available simulation budget  $N_{max}$  is equal to 500 000.

$$s(P^{AST}) = \frac{\sqrt{\frac{1}{m-1} \left( \sum_{i=1}^m (P_i^{AST}(\phi(X) > S) - P_{theory}(\phi(X) > S))^2 \right)}}{P_{theory}(\phi(X) > S)} \quad (5)$$

where  $m$  stands for the number of estimations ( $m = 100$  in the proposed case),  $P_i^{AST}$  is the  $i^{th}$  estimation of the rare event probability using the AST estimator and  $P_{theory}$  is the theoretical value of the probability ( $P_{theory}(\phi(X) > 4) = 3.17 \cdot 10^{-5}$ ). The obtained results for this toy case are given in Table 2.

Success rate	11%
Average probability (success)	$3.18 \cdot 10^{-5}$
Average $s$ (success)	31%

Table 2: Result synthesis of the toy case without adapted tunings

The success rate, standing for the percentage of cases for which AST estimator succeeds to estimate the probability, is only 11%. One considers that the estimator succeeds to estimate a probability when the condition  $q(k) \geq S$  is verified until the total simulation budget is reached. This rate is very weak given the fact that the toy case is very simple compared to real industrial cases (*i.e.* unimodal transfer function, not too rare probability to estimate, sizeable simulation budget), that confirms the need for a precise tuning policy in order to ensure the consistency of the results found by the AST estimator. When the AST achieves the estimation, the results show that the estimator allows to find a relatively good value of the considered rare event probability but with a high relative performance index  $s$ . The

performance index of 31% for this example is prior to the optimization of the simulation parameters.

### 3. Optimal determination of the probability estimator simulation parameters

#### 3.1. Problem formulation

Consider the numerical simulation of a representative test case on which a method (*e.g.* a rare event probability estimator) to be tuned is applied. This so-called “black-box” function, corresponding to the rare event estimator applied to the input-output function  $\phi$ , should return a scalar performance index  $s$ , as a function of a given choice of vector of simulation parameters  $\mathbf{h}$ . These parameters are only assumed to belong to a known bounded set  $\mathbb{H}$ . The tuning problem could then be formalized as the search for the optimal simulation parameters such that

$$\hat{\mathbf{h}} = \arg \min_{\mathbf{h} \in \mathbb{H}} s(\mathbf{h}). \quad (6)$$

This is a difficult problem, since the only available information is the performance value at sampled locations of  $\mathbf{h}$ . The proposed tuning methodology uses a Kriging model to approximate the unknown mapping from the simulation parameter space  $\mathbb{H}$  to the performance criterion  $s(\cdot)$ . The so-called Efficient Global Optimization (EGO) algorithm (see [38]) is then employed to explore areas of the simulation parameter space that might lead to better performance and finally find an optimal tuning.

In the case of rare-event probability estimation, the input-output function  $\phi$  is composed of the complex industrial code, and the parameters to be tuned

are those of the probability estimation strategy (see Table 1). The index  $s$  that measures the performance of the estimation is defined in Equation (5).

### 3.2. Kriging-based optimization

Consider that a small number,  $n$ , of possible simulation parameter tunings have already been experimented, resulting in the set  $\mathcal{H}_n = \{\mathbf{h}_1, \dots, \mathbf{h}_n\}$ . The corresponding performance indices are gathered in the  $n$ -dimensional vector  $\mathbf{s}_n = [s(\mathbf{h}_1), \dots, s(\mathbf{h}_n)]^\top$ . Basing on this initial knowledge, Kriging [39] makes it possible to build a surrogate approximation of the black-box function, by modeling it as a Gaussian process  $S(\cdot)$  with mean function  $m(\cdot)$  and covariance function  $\text{cov}(\cdot, \cdot)$  [40]. For any input  $\mathbf{h}$ ,

$$S(\mathbf{h}) = m(\mathbf{h}) + \zeta(\mathbf{h}), \quad (7)$$

where the mean function  $m(\cdot)$  is a classical known regression model (*e.g.*, polynomial in  $\mathbf{h}$ ) and  $\zeta(\cdot)$  is a zero-mean Gaussian process with covariance function  $\text{cov}(\cdot, \cdot)$ . The idea behind the Kriging surrogate model is to consider that the interpolation with the mean model  $m(\cdot)$  suffers errors at each point of the input space  $\mathbb{H}$ , and that these errors in two distinct points are correlated according to  $\text{cov}(\cdot, \cdot)$ . Since the actual covariance function is unknown in practice, it is usually modeled as

$$\text{cov}(\zeta(\mathbf{h}_i), \zeta(\mathbf{h}_j)) = \sigma_\zeta^2 R(\mathbf{h}_i, \mathbf{h}_j), \quad (8)$$

where  $\sigma_\zeta^2$  is the process variance and  $R(\cdot, \cdot)$  is a parametric correlation function. The widely-used correlation function adopted in this work is the *power exponential correlation function*,

$$R(\mathbf{h}_i, \mathbf{h}_j) = \exp\left(-\sum_{k=1}^{\dim \mathbf{h}} \left| \frac{\mathbf{h}_i(k) - \mathbf{h}_j(k)}{\theta_k} \right|^{p_k}\right) \quad (9)$$

where the parameters  $0 < p_k \leq 2$  reflect the regularity of the underlying Gaussian process, while the  $\theta_k$  are scale factors. Other correlation functions may also be appropriate [41].

The Kriging predictor [42] is given at any  $\mathbf{h} \in \mathbb{H}$ , by

$$\widehat{S}(\mathbf{h}) = m(\mathbf{h}) + \mathbf{r}(\mathbf{h})^T \mathbf{R}^{-1} (\mathbf{s}_n - \mathbf{m}_n), \quad (10)$$

where

$$\begin{cases} \mathbf{R}_{|ij} = R(\mathbf{h}_i, \mathbf{h}_j) \\ \mathbf{r}(\mathbf{h}) = [R(\mathbf{h}, \mathbf{h}_1), \dots, R(\mathbf{h}, \mathbf{h}_n)]^T \\ \mathbf{m}_n = [m(\mathbf{h}_1), \dots, m(\mathbf{h}_n)]^T \end{cases}. \quad (11)$$

The complexity of this surrogate model is very small, since it is linear in the available data  $\mathbf{s}_n$ . Another very interesting property is the ability to compute the *variance of the prediction error*, which could be seen as a confidence level of the prediction (10), as

$$\widehat{\sigma}^2 = \sigma_\zeta^2 \left( 1 - \mathbf{r}(\mathbf{h})^T \mathbf{R}^{-1} \mathbf{r}(\mathbf{h}) \right). \quad (12)$$

This statistical property provides an additional information, since the variance will be small near already sampled input values and much higher when the black-box function is not well known. This has contributed to the development of Kriging-based optimization algorithms that achieve a trade-off between local search (near the best known optimum) and global search (locations where the uncertainty on the surrogate is strong) [43]. One of these strategies is the Efficient Global Optimization (EGO) procedure [38], which proceeds as follows.

1. Sample  $n$  input vectors  $\mathbf{h}$  to compute  $\mathcal{H}_n$  and  $\mathbf{s}_n$ .



2. Fit a Kriging model on  $\mathcal{H}_n$  and  $\mathbf{s}_n$  via (10) and (11).
3. Find  $s_{\min}^n = \min_{i=1,\dots,n} \{s(\mathbf{h}_i)\}$ .
4. Find  $\hat{\mathbf{h}}_{n+1} = \arg \max_{\mathbf{h} \in \mathbb{H}} EI(\mathbf{h}, s_{\min}^n, \hat{S}, \hat{\sigma})$ .
5. Append  $\hat{\mathbf{h}}_{n+1}$  to  $\mathcal{H}_n$  and  $s(\hat{\mathbf{h}}_{n+1})$  to  $\mathbf{s}_n$ .
6. If  $n > n_{\max}$ , return  $s_{\min}^n$  as an optimal solution.

Else, go to Step 2 with  $n \leftarrow n + 1$ .

The working principle of this iterative algorithm is to replace the initial intractable optimization problem (6) by the repeated optimization of a much lighter function called the *Expected Improvement* (*EI* at Step 4), whose closed-form expression is

$$\begin{aligned}
 EI(\mathbf{h}, s_{\min}^n, \hat{S}, \hat{\sigma}) &= \left( s_{\min}^n - \hat{S}(\mathbf{h}) \right) \Psi \left( \frac{s_{\min}^n - \hat{S}(\mathbf{h})}{\hat{\sigma}(\mathbf{h})} \right) \\
 &+ \hat{\sigma}(\mathbf{h}) \psi \left( \frac{s_{\min}^n - \hat{S}(\mathbf{h})}{\hat{\sigma}(\mathbf{h})} \right), \tag{13}
 \end{aligned}$$

in which  $\psi$  and  $\Psi$  are respectively the probability density and cumulative distribution functions of the normal distribution. This function is computationally light, since it only involves the Kriging linear prediction (10) and variance (12). It could thus be optimized at each step via an auxiliary algorithm to be chosen. When the criterion to be optimized depends on both continuous and discrete variables, the search for the maximum expected improvement should be performed in a mixed-integer programming context. For problems involving only continuous variables, the DIRECT optimization algorithm [44] is used to find the point of maximum improvement. For mixed-integer problems, a combination between DIRECT and a classical recursive branch-and-bound strategy is used to deal with the combinatorial optimization problem (each branch consists then in solving a continuous optimization

problem with DIRECT). This approach can be used for problems involving few discrete variables, as is the case here. For problems with more discrete variables, a similar strategy can still be adopted by replacing the simple branch-and-bound approach by a more elaborate integer programming technique.

Thanks to the guidance of this sampling criterion, several candidate optima points are explored iteratively, which provides in the end a set of appropriate simulation parameter vectors for the application considered. The stopping condition of EGO is  $n_{\max}$ , which is the budget allotted for the black-box evaluations (it could also be expressed in terms of computation time). Note that other stopping criteria could also be considered, such as a threshold on the successive maximum values of  $EI$  obtained at Step 4 [45].

An illustration of the potential of EGO for the minimization of complex black-box functions is displayed in Figure 2 for a one-dimensional test case with multiple local optima. The initial sampling is composed of 8 points that have been randomly chosen, leading to an adverse situation where many local optima of the actual function have not been sampled. Although, after a few iterations of EGO, the global optimum is spotted and most of the local optima have also been explored. The Kriging interpolation also improves the sequential acquisition of these new samples.

#### 4. Application of the proposed strategy

In this section, we apply the proposed tuning optimization strategy on a benchmark of representative test cases in order to propose recommendations

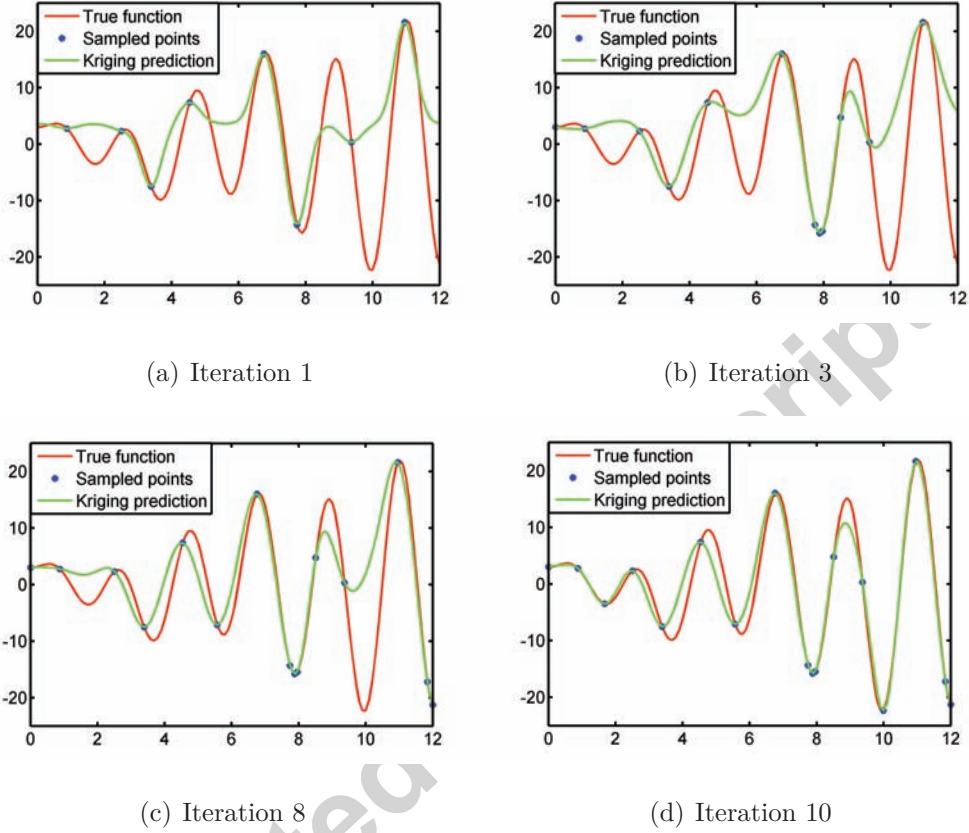


Figure 2: Optimization of a multimodal function by EGO

for the tuning of AST.

#### 4.1. Selection of a transfer function benchmark

In order to be representative of industrial complex system simulations, we have built a benchmark of transfer functions with several test cases, involving different problem dimensions and complexity levels. The transfer functions  $\phi$  taken into account in the proposed benchmark are the following:

- Case 1 : Identity function in dimension 1 (toy case)

- Case 2 : Ackley-like function in dimension 2
- Case 3 : Massively multimodal Ackley-like function in dimension 4
- Case 4 : Rayleigh function in dimension 15

We have chosen the identity function as the toy case in order to test the optimization strategy in the same case as the one presented in the second section. The test cases #2 and #3 involve modified Ackley functions in order to progressively increase the dimension of the input space and the complexity (number of modes and non linearities) of the transfer function. Finally, in the test case #4, we have chosen the Rayleigh function in order to be representative of high-dimensional systems. The considered initial distribution is a normal distribution with zero mean and unit standard deviation for all the test cases. Moreover, the simulation budget has been set to 500000 points.

#### 4.1.1. Toy case : unidimensional identity function

This case is the same as the one presented in Section 2 (Figure 1). It consists in determining the probability that the normal law exceeds a threshold which has been taken at 4. This case is very simple and has been addressed using AST in [35].

$$S_1 = 4 \tag{14}$$

$$X \sim \mathcal{N}(0_1, I_1) \tag{15}$$

$$\phi_1 : \begin{cases} \mathbb{R} \rightarrow \mathbb{R} \\ x \rightarrow x \end{cases} \tag{16}$$

#### 4.1.2. Case 2 : Ackley transfer function in dimension 2

The transfer function used in this test case is a modified Ackley-like function. The expression of the transfer function is given in Equation (19). In this equation,  $x_i$  stands for the  $i^{\text{th}}$  component of  $x$ .  $\phi_2$  is represented in Figure 3. This function presents numerous modes but only one is above the selected threshold ( $S_2 = 10$ ). The estimated theoretical probability, obtained by  $10^8$  CMC simulations is equal to  $7.2 \cdot 10^{-6}$ .

$$S_2 = 10 \quad (17)$$

$$X \sim \mathcal{N}(0_2, I_2) \quad (18)$$

$$\phi_2 : \begin{cases} \mathbb{R}^2 \rightarrow \mathbb{R} \\ x \rightarrow \Gamma - \frac{1}{\chi} \left( -\delta \exp \left( -\gamma \sqrt{\frac{1}{\epsilon} \sum_{i=1}^2 (x_i - \kappa)^2} \right) - \eta \exp \left( \frac{1}{\epsilon} \sum_{i=1}^2 \cos(2\pi(x_i - \kappa)) \right) + \omega \right) \end{cases} \quad (19)$$

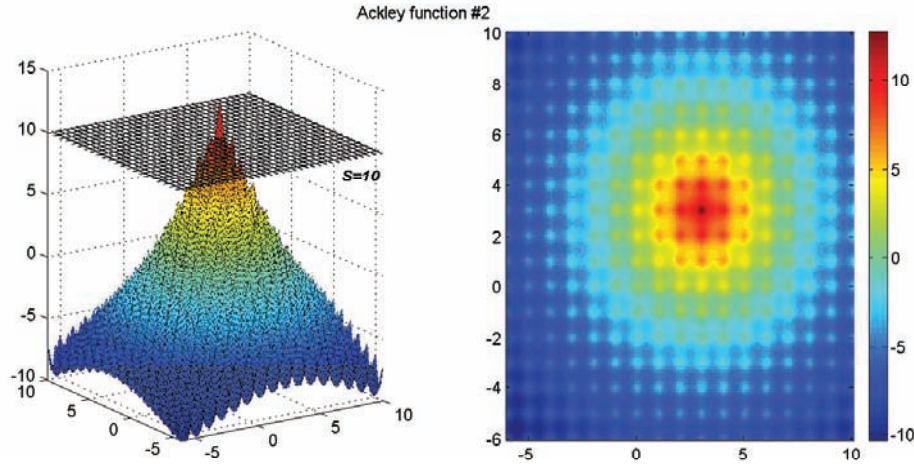


Figure 3: Ackley transfer function #2

#### 4.1.3. Case 3 : massively multimodal Ackley transfer function in dimension

4

The transfer function used in this test case is a modified Ackley-like function. The expression of the transfer function used in the previous test case has been modified to be massively multimodal (Equation 22). The complexity of the transfer function is increased by the presence of several zones which are above the selected threshold (Figure 4). Moreover, the dimension of the input data has been doubled with respect to the test case #2. The estimated theoretical probability, obtained by  $10^8$  CMC simulations is equal to  $3.2 \cdot 10^{-5}$ .

$$S_3 = 9 \quad (20)$$

$$X \sim \mathcal{N}(0_4, I_4) \quad (21)$$

$$\phi_3 : \begin{cases} \mathbb{R}^4 \rightarrow \mathbb{R} \\ x \rightarrow \Gamma - \frac{1}{\chi} \left( -\delta \exp \left( -\gamma \sqrt{\frac{1}{\epsilon} \sum_{i=1}^2 (x_i^2 - \kappa^2)^2} \right) - \eta \exp \left( \frac{1}{\epsilon} \sum_{i=1}^2 \cos(2\pi(x_i - \kappa)) \right) + \omega \right) \end{cases} \quad (22)$$

#### 4.1.4. Case 4 : Rayleigh transfer function in dimension 15

In order to generate the Rayleigh law, we use as the input a normal law and as the transfer function the Euclidean norm (Figure 5). The dimension of input data has been greatly increased with respect to the previous test cases. The estimated theoretical probability, obtained by  $10^8$  CMC simulations is equal to  $1.1 \cdot 10^{-6}$ .

$$S_4 = 7.5 \quad (23)$$

$$X \sim \mathcal{N}(0_{15}, I_{15}) \quad (24)$$

$$\phi_4 : \begin{cases} \mathbb{R}^{15} \rightarrow \mathbb{R} \\ x \rightarrow \|x\| \end{cases} \quad (25)$$

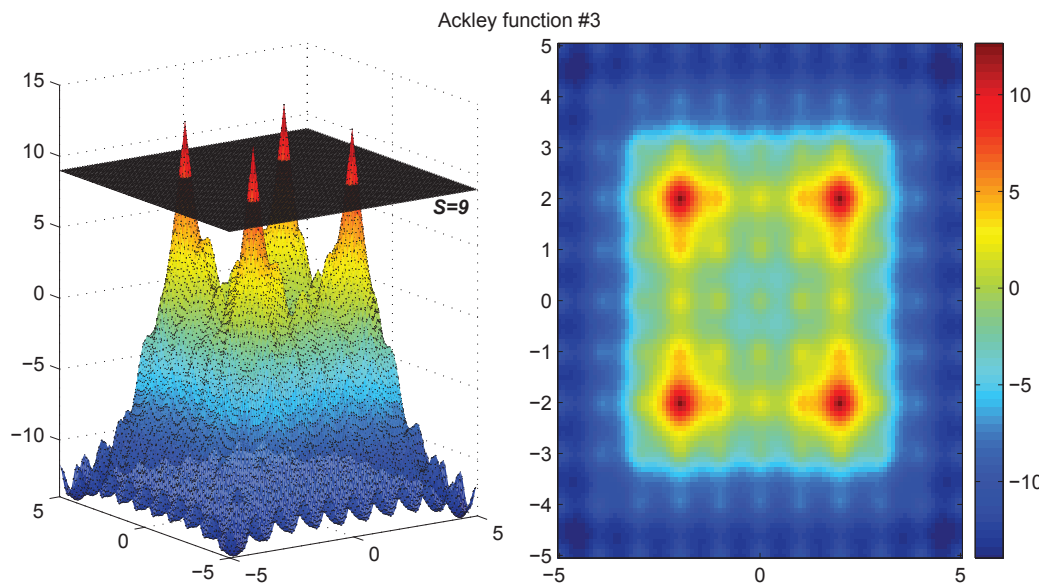


Figure 4: Representation of Ackley transfer function #3 in dimension 2

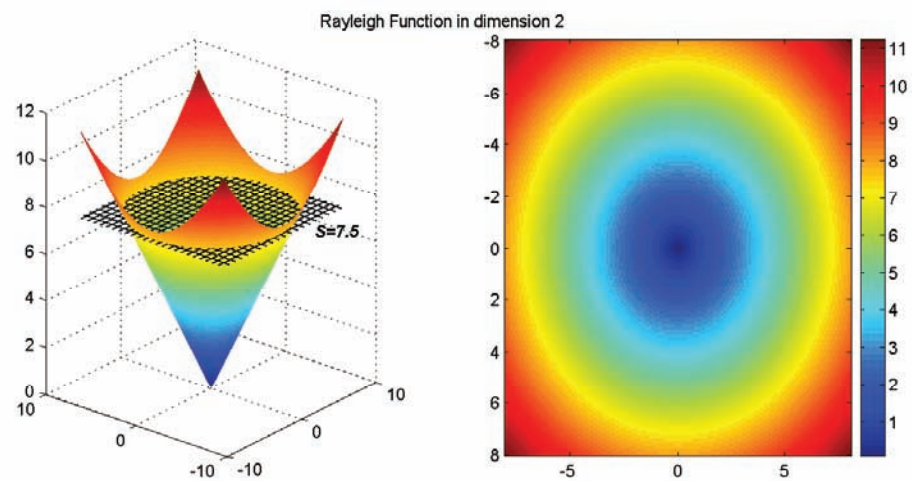


Figure 5: Rayleigh transfer function

Even if the expression of the transfer functions are analytical, AST considers them as input-output functions.

#### 4.2. Optimal tuning determination

In this section, we present the results obtained with the proposed optimization strategy. Each optimization has taken between two and four hours using MATLAB, 2.4Ghz DualCore Pentium/Windows XP. For each of the four performed optimizations, we consider that the tuning combinations that lead to a performance index of the AST estimator (Equation 5) lower than 10% are valuable and we keep them for determining the relevant tuning sets (Table 3). This is made possible since EGO allows the exploration of several good candidates. These different sets are represented in Figure 6. The intersection of these sets is not empty and is used to define the tuning recommendations (Table 4).

Test function/Parameter	$N$	$N_{app}$	$\alpha$	$\beta$
Gauss	[3000,15000]	[1,4]	[0.3,0.89]	[0.41,0.89]
Ackley #1	[3300,13500]	[2,5]	[0.35,0.85]	[0.60,0.85]
Ackley #2	[2000,16300]	[3,10]	[0.15,0.45]	[0.28,0.92]
Rayleigh	[3500,11000]	[1,6]	[0.3,0.65]	[0.42,0.95]

Table 3: Tuning intervals leading to a performance index lower than 10%

Parameter	$N$	$N_{app}$	$\alpha$	$\beta$
Recommendations	[3500,11000]	[3,4]	[0.35,0.45]	[0.60,0.85]

Table 4: Tuning recommendations



These recommendations are relatively consistent with respect to the domain literature. Indeed, in [9, 46], the authors indicate that an intermediary quantile level  $\beta$  between 0.75 and 0.8 and a Markov kernel stepsize  $\alpha$  around 0.3 are, from their experience, valuable parameters. However, in this paper, since the estimation has been performed with a restricted simulation budget, the proposed recommendation for the number of Markov kernel applications reveals to be lower than in [9].

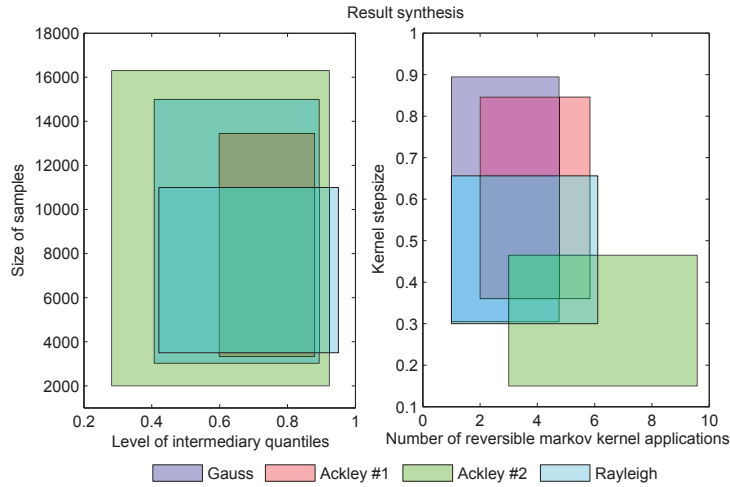


Figure 6: Result synthesis : each rectangle corresponds to valuable tuning parameter domain for a given test case

#### 4.3. Verification of the tuning recommendations on the benchmark

In order to check the consistency of the tuning recommendations on the benchmark, we have tested 40 combinations of tunings in the proposed tuning set, using a Latin Hypercube Sampling (Figure 7). The results obtained for the different test cases are summarized in Table 5. These results confirm the

relevance of the proposed recommendations because the sampled simulation parameter combinations in the recommended tuning set lead to a minimum success rate of the AST estimator equal to 95% and an average performance index which is always lower than 11%.

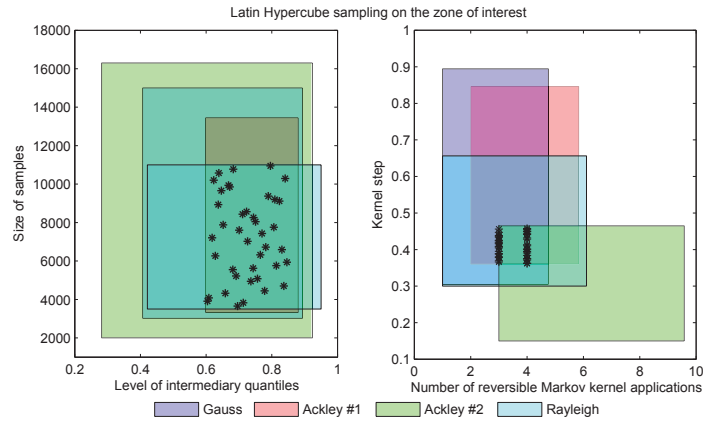


Figure 7: Latin Hypercube Sampling on valuable tunings

Test case	Success rate	Average $s$ (success)
#1	100%	7.29%
#2	95%	9.9%
#3	95%	10.7%
#4	95%	7.5%

Table 5: Verification on the benchmark

## 5. Experimentation on complex system cases

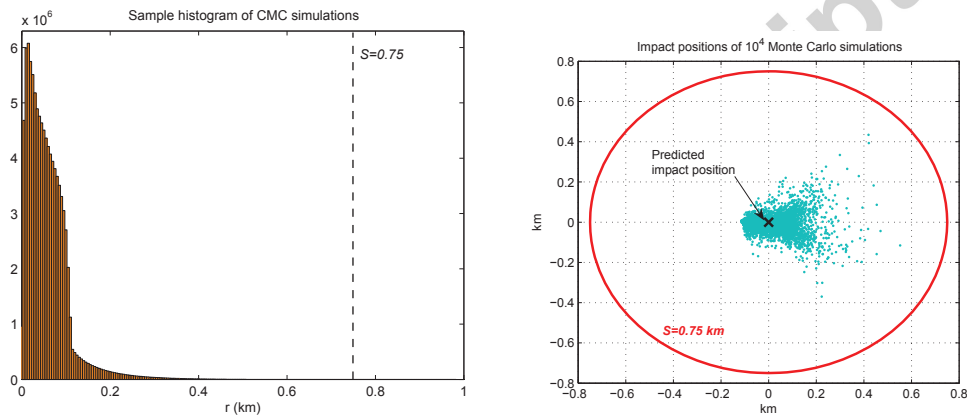
### 5.1. Spatial launch vehicle fall-back safety zone estimation

Spatial launch vehicle fall-back safety zone estimation is a very important problem in spatial application since the consequence of a mistake can be dramatic for the populations. We consider in this article a solid rocket booster that is the first stage of a launch vehicle. Its weight is about 35,000 kilograms and the launch point is at 112 kilometer height with a slope of 15 degrees. At the end of its mission, the rocket booster falls into the sea at some distance of a predicted position. Similar models have already been analyzed in [47].

The launch vehicle stage fall-back is thus modeled as an input-output function  $\phi$  with 12 Gaussian inputs  $X$  and one output  $Y = \phi(X)$ , representing the distance between the estimated launch stage fall-back position and the predicted one. In this study case, we aim at estimating the probability that the distance to the predicted impact position exceeds 0.75km  $P(\phi(X) > 0.75)$ . For this test-case, the estimated theoretical probability is equal to  $6.72 \cdot 10^{-6}$ . Figure 8 shows a sample histogram of CMC simulations and the corresponding positions of impacts. Several inputs of  $X$  can then influence its impact position:

- launch vehicle measurement unit uncertainty in position and speed (4 inputs). These parameters describe the performances of the launch vehicle accelerometers and gyros.
- meteorological conditions (2 inputs). The wind variations during the fall-back can influence the impact position.

- launch vehicle design and weight (5 inputs). The weight and the size of the different parts of the launch vehicle are also slightly random during the fall-back.
- the slope angle between the vertical axis and the velocity vector (1 input).



(a) Sample histogram of  $10^8$  CMC simulations (b) Impact positions of  $10^4$  CMC simulations

Figure 8: CMC simulations of the considered system

## 5.2. Missile safety

Safety consideration takes a more and more important place in the design and the evaluation of a missile. It is indeed necessary to characterize the risk that a missile does not reach its target in the evaluation of missile performances. In the second application case of this article, one considers an air launched missile. It is a supersonic stand-off missile powered by a liquid-fuel ramjet. It flies at Mach 2 to Mach 3, with a range between 80 km and

300 km depending on flight profile. More details on this computer code and its different inputs are given in the reference [34]. This missile case is highly multidimensional and is modeled with a continuous black-box computer code  $\phi$  with 51 independent Gaussian inputs  $X$  and one output  $Y = \phi(X)$  where  $Y$  is the distance between the missile and its target. Figure 9 shows a sample histogram of CMC simulations. The estimated theoretical probability for this test case is  $2.13 \cdot 10^{-5}$ . The different inputs of this black-box simulation code are the following :

- launch aircraft inertial measurement unit in position and speed (10 inputs),
- missile inertial measurement unit in position and speed (10 inputs),
- meteorological conditions (4 inputs),
- ramjet combustion process (8 inputs),
- missile design and weight (6 inputs),
- internal hydraulic cylinders efficiency (6 inputs),
- aerodynamic and flight mechanics parameters (7 inputs).

### *5.3. Application of the proposed tuning policy*

In order to test our proposed tuning policy on these real-world aerospace problems, we have applied the same 40 combinations of tunings using a Latin Hypercube Sampling taken from the recommended tuning set (Figure 10). In these two test-cases, the tuning parameters have not been optimized and

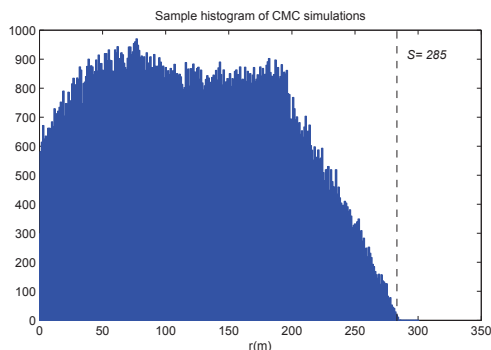


Figure 9: Missile safety sample histogram of  $10^5$  CMC simulations

have been simply selected according to the tuning recommendations (Table 4). The success rate of these real-world tests are respectively 92.5% and 75%, with corresponding average performances indices of 11.6% and 19%. In order to estimate the influence of the tunings on these problems, we have performed another test of 40 tuning combinations but sampled in a larger tuning set including the recommended one (Figure 10). As summarized in Table 6, the results show that for these examples, the performance of the AST estimator is greatly lessen when the tunings are chosen outside the recommended tuning set. In that way, in this two test-cases, taking into account the tuning recommendations allows to improve by a factor 1.5 the performances of the rare event probability estimator. We can note that the accuracy and the success rate of the spatial launch vehicle fall-back safety zone probability estimation are better than these of the missile-target distance probability estimation. This can be explained on the one hand by the very high non linearity of the missile safety computer code in comparison with the spacecraft fallback code. On the other hand, the dimensions of the input space involved in the two test

cases are different and vary by a factor 5 between the two cases. Therefore, the performance of the importance splitting algorithm is logically relatively worse in the missile safety case than in the spacecraft fall-back zone case.

		Success rate	Average $s$ (success)
Spacecraft fall-back zone	LHS with the recommendations	92.5%	11.6%
	LHS without the recommendations	67.5%	23.3%
Missile safety	LHS with the recommendations	75%	19%
	LHS without the recommendations	47%	33%

Table 6: Obtained results for the two aerospace test cases

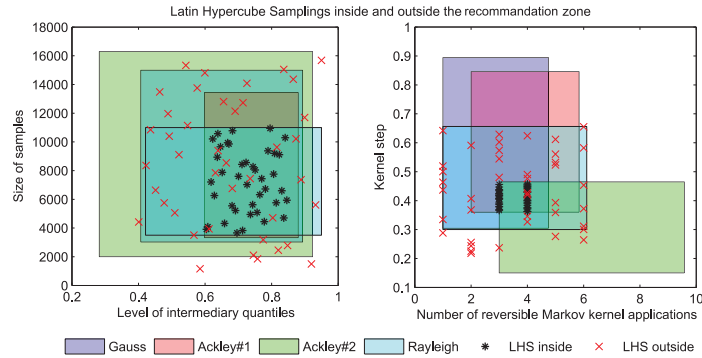


Figure 10: Latin Hypercube Sampling inside and outside the recommendation set

## 6. Conclusion

In this paper, we have proposed several recommendations for the tuning of the Adaptive Splitting Technique. To this end, a methodology for the determination of the optimal simulation parameters has been proposed and

applied to several test cases representative of complex system simulations. From the valuable tuning combinations obtained for these test cases, some recommendations have been proposed and verified in a series of real aerospace problems. The relevance of these tuning recommendations on other realistic study cases remains to be assessed. The proposed recommendations are the result of numerical global optimizations of the AST parameters. A theoretical verification of the proposed recommendations would be very interesting and will be tackled in future works. Moreover, it would also be interesting to apply the proposed tuning methodology to other rare event probability estimators such as Importance Sampling. For non Gaussian cases, using Metropolis Hastings algorithm or density transformations (Rosenblatt, Nataf, *etc.*) are interesting ways of AST improvements but still require further development to be used in industrial context.

## 7. Acknowledgement

The authors are thankful to Dr. Rudy Pastel (Onera), Dr. H el ene Piet-Lahanier (Onera) and Dr. Fran ois Le Gland (INRIA) for fruitful discussions. The authors thank for partial funding the Minist ere de l' conomie, de l'Industrie et de l'Emploi (Minister for the Economy, Industry and Employment) in the frame of CSDL (Complex System Design Lab) project of French competitiveness cluster System@tic - Paris R egion.

## References

- [1] Z. I. Botev, D. Kroese, T. Taimre, Generalized cross-entropy methods with applications to rare-event simulation and optimization, *Simulation*



- 11 (2007) 785 – 806.
- [2] P. Zhang, Nonparametric importance sampling, *Journal of the American Statistical Association* 91 (1996) 1245–1253.
- [3] R. Rubinstein, Optimization of computer simulation models with rare events, *European Journal of Operations Research* 99 (1997) 89–112.
- [4] P. Glynn, Importance sampling for Monte Carlo estimation of quantiles, Publishing House of Saint Petersburg University, 1996, pp. 180–185.
- [5] J. Morio, How to approach the importance sampling density, *European Journal of Physics* 31 (2010) 41–48.
- [6] P. Glasserman, P. Heidelberger, P. Shahabuddin, T. Zajic, Splitting for rare event simulation: analysis of simple cases, *Proceeding of the 1996 Winter Simulation Conference* (1996) 302–308.
- [7] P. L'Écuyer, V. Demers, B. Tuffin, Splitting for rare event simulation, *Proceeding of the 2006 Winter Simulation Conference* (2006) 137–148.
- [8] F. Cérou, P. Del Moral, T. Furon, A. Guyader, Rare event simulation for a static distribution, volume 7, *RESIM*, 2008, pp. 107–115.
- [9] F. Cérou, P. Del Moral, T. Furon, A. Guyader, Sequential Monte Carlo for rare event estimation, *Statistics and Computing* (2011) 1–14.
- [10] M. Gilli, E. Këllezi, An application of extreme value theory for measuring risk, *Computational Economics* 27 (2006) 207–228.

- [11] A. Davison, R. Smith, Models for exceedances over high thresholds (with discussion), *Journal of the Royal Statistical Society* 52 (1990) 393–442.
- [12] A. McNeil, T. Saladin, The peaks over threshold method for estimating high quantiles of loss distributions, *Proceedings of the 2nd International ASTIN Colloquium*, 1997.
- [13] J. Pickands, Statistical inference using extreme order statistics, *Annals of Statistics* 3 (1975) 119–131.
- [14] S.-K. Au, J. L. Beck, Estimation of small failure probabilities in high dimensions by subset simulation, *Probabilistic Engineering Mechanics* 16 (2001) 263 – 277.
- [15] G. H. Golub, M. Heath, G. Wahba, Generalized cross-validation as a method for choosing a good ridge parameter, *Technometrics* 21 (1979) 215–223.
- [16] R. Kohavi, G. H. John, Automatic parameter selection by minimizing estimated error, *Proceedings of the Twelfth International Conference on Machine Learning* (1995) 304–312.
- [17] F. Hutter, H. H. Hoos, T. Stutzle, Automatic algorithm configuration based on local search, *Proceedings of the National Conference on Artificial Intelligence* 22 (2007) 1152–1160.
- [18] R. Pavón, F. Díaz, V. Luzón, A model for parameter setting based on Bayesian networks, *Engineering Applications of Artificial Intelligence* 21 (2008) 14–25.

- [19] C. Gold, P. Sollich, Model selection for support vector machine classification, *Neurocomputing* 55 (2003) 221–249.
- [20] O. V. Korniyenko, M. S. Sharawi, D. N. Aloi, Neural network based approach for tuning Kalman filter, *Proceedings of the IEEE International Conference on Electro/Information Technology*, Lincoln, USA (2006).
- [21] T. D. Powell, Automated tuning of an extended Kalman filter using the downhill simplex algorithm, *Journal of Guidance, Control, and Dynamics* 25 (2002) 901–908.
- [22] S.-W. Lin, K.-C. Ying, S.-C. Chen, Z.-J. Lee, Particle swarm optimization for parameter determination and feature selection of support vector machines, *Expert Systems with Applications* 35 (2008) 1817–1824.
- [23] A. Varsek, T. Urbancic, B. Filipic, Genetic algorithms in controller design and tuning, *IEEE Transactions on Systems, Man and Cybernetics* 23 (1993) 1330–1339.
- [24] M. Gen, Y. Yun, Soft computing approach for reliability optimization: State-of-the-art survey, *Reliability Engineering & System Safety* 91 (2006) 1008–1026.
- [25] T. J. Santner, B. J. Williams, W. Notz, *The Design and Analysis of Computer Experiments*, Springer-Verlag, Berlin-Heidelberg, 2003.
- [26] J. Marzat, E. Walter, H. Piet-Lahanier, F. Damongeot, Automatic tuning via Kriging-based optimization of methods for fault detection and isolation, *Proceedings of the IEEE Conference on Control and Fault-Tolerant Systems*, Nice, France (2010) 505–510.

- [27] B. Silverman, *Density Estimation for statistics and data analysis*, Chapman and Hall, London, 1986.
- [28] G. A. Mikhailov, *Parametric Estimates by the Monte Carlo Method*, VSP, Utrecht (NED), 1999.
- [29] I. M. Sobol, *A Primer for the Monte Carlo Method*, CRC Press, Boca Raton, FL., 1994.
- [30] C. P. Robert, G. Casella, *Monte Carlo Statistical Methods*, Springer, New York, 2005.
- [31] H. Niederreiter, J. Spanier, *Monte Carlo and Quasi-Monte Carlo Methods*, Springer, 2000.
- [32] S. Au, J. Beck, Subset simulation - a new approach to calculating small failure probabilities, *Proceedings of the International Conference on Monte Carlo Simulation (2000)* 287–293.
- [33] S.-K. Au, J. L. Beck, Estimation of small failure probabilities in high dimensions by subset simulation, *Probabilistic Engineering Mechanics* 16 (2001) 263 – 277.
- [34] J. Morio, R. Pastel, F. L. Gland, Missile target accuracy estimation with importance splitting, *Aerospace Science and Technology* 25 (2013) 40 – 44.
- [35] J. Morio, R. Pastel, F. Le Gland, An overview of importance splitting for rare event simulation, *European Journal of Physics* 31 (2010) 1295–1303.

- [36] L. Tierney, Markov chains for exploring posterior distributions, *Annals of Statistics* 22 (1994) 1701–1762.
- [37] M. D. McKay, R. J. Beckman, W. Conover, A comparison of three methods for selecting values of input variables in the analysis of output from a computer code, *Technometrics* 21 (1979) 239 – 245.
- [38] D. R. Jones, M. J. Schonlau, W. J. Welch, Efficient global optimization of expensive black-box functions, *Journal of Global Optimization* 13 (1998) 455–492.
- [39] G. Matheron, Principles of geostatistics, *Economic Geology* 58 (1963) 1246.
- [40] J. Lefebvre, H. Roussel, E. Walter, D. Lecointe, W. Tabbara, Prediction from wrong models: the Kriging approach, *IEEE Antennas and Propagation Magazine* 38 (1996) 35–45.
- [41] C. E. Rasmussen, C. K. I. Williams, *Gaussian Processes for Machine Learning*, Springer-Verlag, New York, 2006.
- [42] M. J. Sasena, Flexibility and Efficiency Enhancements for Constrained Global Design Optimization with Kriging Approximations, PhD Thesis, University of Michigan, USA, 2002.
- [43] D. R. Jones, A taxonomy of global optimization methods based on response surfaces, *Journal of Global Optimization* 21 (2001) 345–383.
- [44] D. R. Jones, C. D. Perttunen, B. E. Stuckman, Lipschitzian optimization

without the Lipschitz constant, *Journal of Optimization Theory and Applications* 79 (1993) 157–181.

- [45] M. Schonlau, *Computer Experiments and Global Optimization*, PhD Thesis, University of Waterloo, Canada, 1997.
- [46] A. Lagnoux, Rare event simulation, *Probability in the Engineering and Informational science* 20 (2006) 45–66.
- [47] J. Morio, Non-parametric adaptive importance sampling for the probability estimation of a launcher impact position, *Reliability Engineering and System Safety* 96 (2011) 178 – 183.