# Fast kriging-based stepwise uncertainty reduction with application to the identification of an excursion set

Clément Chevalier, Julien Bect, David Ginsbourger, Emmanuel Vazquez, Victor Picheny, Yann Richet

# Fast kriging-based stepwise uncertainty reduction with application to the identification of an excursion set

**Clément Chevalier**[⋆] **· Julien Bect · David Ginsbourger · Victor Picheny · Yann Richet · Emmanuel Vazquez**

November 14, 2011

**Abstract** A Stepwise Uncertainty Reduction (SUR) strategy aims at constructing a sequence $X_1(f), X_2(f), \ldots$ of evaluation points of a function $f : \mathbb{R}^d \to \mathbb{R}$ in such a way that the residual uncertainty about a quantity of interest $S(f)$ given the information provided by the evaluation results is small. In Bect, Ginsbourger, Li, Picheny and Vazquez, Statistics and Computing, 2011, several SUR approaches have been shown to be particularly efficient for the problem of estimating the volume of an excursion set of a function $f$ above a threshold. Here, we build upon these results and we present fast implementations of some SUR strategies, which are based on two ideas. The first idea is to take advantage of update formulas for kriging. The second idea is to derive closed-form expressions for some integrals that appear in the SUR criteria. We are able to demonstrate significant speed-ups and we illustrate our algorithms on a nuclear safety application.

**Keywords** Computer experiments · Gaussian processes · Sequential design · Probability of failure estimation · Active learning · Metamodel-based inversion

⋆ corresponding author

C. Chevalier · D. Ginsbourger
Institute of Mathematical Statistics and Actuarial Science
University of Bern, Switzerland.
E-mail: {firstname}.{lastname}@stat.unibe.ch

J. Bect and E. Vazquez
Supelec, Gif-sur-Yvette, France.
E-mail: {firstname}.{lastname}@supelec.fr

V. Picheny
CERFACS, Toulouse, France.
E-mail: picheny@cerfacs.fr

Y. Richet
Institut de Radioprotection et de Sûreté Nuclaire (IRSN),
Fontenay-aux-Roses, France.
E-mail: yann.richet@irsn.fr

## 1 Introduction

Whether in natural sciences, engineering, or economics, the study of complex phenomena is increasingly relying on numerical simulations. From an end user's perspective, a numerical simulator can often be considered as a black box taking a number of real-valued parameters as inputs and returning one or several quantities of interest after a post-processing stage—typically, the response is a cost or a performance. Formally, the space of inputs is a set $\mathbb{X} \subset \mathbb{R}^d$ and the simulator can be viewed as a function $f : \mathbb{X} \to \mathbb{R}$ that maps the inputs to a cost or a performance indicator. In many practical applications, the objective is to obtain information about the simulator from a number of runs or in other words, to infer a quantity of interest $S(f)$ from a number of evaluations of $f$. For instance, in presence of uncertain input parameters, a problem that is often at stake is the estimation of the probability that a cost exceeds a given threshold. This problem corresponds to the estimation of the volume $\alpha(f)$ of the excursion set $\Gamma = \{x \in \mathbb{X} : f(x) \geq T\}$, with $T$ a given threshold, under a probability measure $\mathsf{P}_{\mathbb{X}}$ on $\mathbb{X}$ that models the uncertainty about the input parameters.

In this paper we consider the problem of estimating $S(f)$ using Stepwise Uncertainty Strategies (Vazquez and Piera-Martinez, 2007; Vazquez and Bect, 2009; Bect et al., 2011), and our objective is to propose efficient algorithms for their implementation. A Stepwise Uncertainty Reduction (SUR) strategy aims at constructing a sequence $X_1(f), X_2(f), \ldots$ of evaluation points of $f$ in such a way that the residual uncertainty about $S(f)$ given the information provided by the evaluation results is small. More precisely, SUR strategies are based on three main ideas. The first (Bayesian) idea is to consider $f$ as a sample path of a random process, which is assumed Gaussian for the sake of tractability. Doing so entails that any quantity depending on $f$ is formally a random variable (in the following, we choose not to make a formal distinction between the function $f$ and its random process model). The second idea is to introduce a measure of the uncertainty about the quantity of interest conditioned on the $\sigma$-algebra $\mathcal{A}_n$ generated by $\{(X_i(f), f(X_i(f))),\ 1 \leq i \leq n\}$. We will denote by $H_n(f)$ such a measure of uncertainty, which is an $\mathcal{A}_n$-measurable random variable. The third idea is to choose evaluation points sequentially in order to minimize at each step $n$ the *expected* value of the future uncertainty $H_{n+1}(f)$ with respect to the random outcome of the new evaluation of $f$:

$$X_{n+1}(f) \;=\; \operatorname*{argmin}_{x_{n+1} \in \mathbb{X}} J_n(x_{n+1}) := \mathbb{E}_n\left(H_{n+1}(f) \;\middle|\; X_{n+1}(f) = x_{n+1}\right), \qquad (1)$$

where $\mathbb{E}_n(\,\cdot\,)$ stands for the conditional expectation $\mathbb{E}(\,\cdot \mid \mathcal{A}_n)$.

For example, the conditional variance $\mathrm{Var}_n\left(\alpha(f)\right)$ of $\alpha(f)$ given $\mathcal{A}_n$ is a natural choice for $H_n$ to quantify the (residual) uncertainty about $\alpha(f)$. Then, a SUR strategy to estimate $\alpha(f)$ would consist, at step $n$, in choosing the next evaluation point by looking at how much sampling at $x_{n+1} \in \mathbb{X}$ will affect, in expectation, the variance of the volume of excursion.

As stated above the main objective of this paper is to present numerically efficient implementations of some SUR strategies. For example, in Bect et al. (2011) the

numerical computation of $J_n$ in (1) was based on quadrature formulas written as

$$J_n(x_{n+1}) \approx \sum_{q=1}^{Q} \sum_{m=1}^{M} w_q v_{n+1}(x^{(m)}; x_{n+1}, y_q), \tag{2}$$

where $Q$ is the number of points used to approximate the conditional expectation with respect to the random outcome of the evaluation at $x_{n+1}$ using a standard Gauss-Hermite quadrature, and $M$ is the number of points used to obtain a Monte-Carlo approximation of $H_{n+1}(f)$. (The $x^{(m)}$s are i.i.d. according to $\mathsf{P}_{\mathbb{X}}$; $(y_1, \ldots, y_q)$ and $(w_1, \ldots, w_q)$ stand for the quadrature points and quadrature weights of the Gauss-Hermite quadrature.) As will be detailed in the following sections, the computation of $v_{n+1}(x^{(m)}; x_{n+1}, y_q)$ in (2) involves the determination of the kriging mean and the kriging variance at $x^{(m)}$ from the evaluations of $f$ at $X_1(f), \ldots, X_n(f)$ and $x_{n+1}$. It follows that the computation of the computation of $J_n$ at one point has a complexity of $O(n^3 + M.n^2 + M.Q)$ operations. Since we need to evaluate $J_n$ several times to carry out the minimization in (1), the computational cost of a SUR sampling strategy implemented using (2) can be very large.

**Outline of the paper.** In order to decrease the numerical complexity of SUR strategies, we propose two ideas. The first idea is to take advantage of update formulas for kriging to avoid unnecessary computations of the kriging weights when computing criteria such as (2) at $p$ different locations. After having recalled classical numerical considerations about kriging in Section 2, we will show how to obtain a first improvement toward efficient SUR algorithms in Section 3. The second idea is to revisit the numerical approximation of $H_n$. In some cases, it appears that parts of the numerical approximation of $H_n$ can be carried out with the help of closed-form expressions, which makes it possible to obtain, again, a substantial gain with respect to what was done in previous studies. In particular, Section 4 derives closed form expressions for some criteria in Bect et al. (2011) and Gramacy and Lee (2010). Finally, in Section 5, we illustrate our algorithms on a nuclear safety application, where we are able to show considerable speed-ups with respect to implementations proposed in Bect et al. (2011).

**Simplified notations.** In the rest of the paper, we shall drop the dependence on $f$ for random variables that are a function of $f$. For instance, $\alpha(f)$ will be denoted by $\alpha$. Evaluation points $X_1(f), X_2(f), \ldots$ will be denoted by $X_1, X_2, \ldots$

## 2 Kriging update formulas

### 2.1 Kriging mean and variance

In this section, we shall recall how to obtain the kriging mean and variance without using update formulas. Let $f \sim \mathrm{GP}(m, k)$ be a Gaussian random process with mean function $m(\cdot) = \mathbb{E}(f(\cdot))$ and covariance function $k(\cdot, \cdot) = \mathrm{cov}(f(\cdot), f(\cdot))$. We assume that the mean can be written as a linear combination

$$m(\cdot) = \sum_{i=1}^{l} \beta_i \, p_i(\cdot) \tag{3}$$

of basis functions $p_1, \ldots, p_l$ (very often, these are monomials), where $\beta_1, \ldots, \beta_l$ are *unknown* parameters. The covariance $k$ is assumed to be a given symmetric *strictly* positive function.

The kriging predictor of $f$ at a point $x \in \mathbb{X}$ from $n$ observations $f(x_1), \ldots, f(x_n)$ is the best linear unbiased predictor (BLUP) of $f(x)$ from the observations, that we shall denote by

$$m_n(x) \;=\; \lambda(x;\underline{x}_n)^{\mathsf{T}} \begin{pmatrix} f(x_1) \\ \vdots \\ f(x_n) \end{pmatrix}. \tag{4}$$

The vector of *kriging weights* $\lambda(x;\underline{x}_n) \in \mathbb{R}^n$ can be obtained by solving the linear system

$$\underbrace{\begin{pmatrix} K(\underline{x}_n) & p(\underline{x}_n)^{\mathsf{T}} \\ p(\underline{x}_n) & \mathbf{0} \end{pmatrix}}_{:=\widetilde{K}(\underline{x}_n)} \cdot \underbrace{\begin{pmatrix} \lambda(x;\underline{x}_n) \\ \mu(x;\underline{x}_n) \end{pmatrix}}_{:=\widetilde{\lambda}(x;\underline{x}_n)} \;=\; \underbrace{\begin{pmatrix} k(x,\underline{x}_n) \\ p(x) \end{pmatrix}}_{:=\widetilde{k}(x,\underline{x}_n)} \tag{5}$$

where $K(\underline{x}_n)$ is the $n \times n$ covariance matrix of the random vector $(f(x_1), \ldots, f(x_n))^{\mathsf{T}}$, $p(\underline{x}_n)$ is the $l \times n$ matrix with general term $p_i(x_j)$, $k(x,\underline{x}_n)$ is the column vector with general term $k(x,x_i)$ and $\mu(x;\underline{x}_n)$ is a vector of $l$ Lagrange multipliers associated to the unbiasedness constraint.

The covariance function of the prediction error

$$k_n(x,y) \;:=\; \mathbb{E}_n\left(\left(f(x) - m_n(x)\right)\left(f(y) - m_n(y)\right)\right), \tag{6}$$

also called *kriging covariance*, can be written using the notations of equation (5) as

$$k_n(x,y) \;=\; k(x,y) - \widetilde{k}(x,\underline{x}_n)^{\mathsf{T}}\,\widetilde{\lambda}(y;\underline{x}_n). \tag{7}$$

The conditional variance of the prediction error at a point $x \in \mathbb{X}$, also called *kriging variance*, will be denoted by $s_n^2(x) := k_n(x,x)$.

*Remark 1* To ensure that the conditional process $f|\mathcal{A}_n$ is still Gaussian when the mean function is of the form (3), with an unknown vector of parameters $\beta$, it is necessary to adopt a Bayesian approach and to use an (improper) uniform distribution over $\mathbb{R}^l$ as a prior distribution for $\beta$ (see Bect et al. (2011), Section 2.3, Proposition 2, and the references therein for more details).

### 2.2 Update formulas

Consider the complexity of computing the kriging mean $m_{n+1}(x)$ and the kriging variance $s_{n+1}^2(x)$ at $M$ different values of $x \in \mathbb{X}$. A system of the form (5) has to be solved $M$ times with the same left-hand side $\widetilde{K}(\underline{x}_{n+1})$, which is typically performed with a cost of $O(n^3 + Mn^2)$ operations: the matrix inverse $\widetilde{K}_{n+1}^{-1}$ (alternatively, a QR or LU factorization) is first computed in $O(n^3)$ operations, and a matrix-vector multiplication is then performed for each $x$ in $O(n^2)$ operations (see, e.g., Golub and Van Loan, 1996, for more information about the complexity of standard algorithms in

linear algebra). It turns out that this complexity can be improved upon, in the case where the inverse (or factorization) of $\widetilde{K}(\underline{x}_n)$ is already available, together with the kriging means and variances from $n$ observations. Indeed, consider the following kriging update formulas (Barnes and Watson, 1992):

$$m_{n+1}(x) \;=\; m_n(x) \;+\; \frac{k_n(x, x_{n+1})}{s_n^2(x_{n+1})} \;\cdot\; \Big[f(x_{n+1}) - m_n(x_{n+1})\Big], \tag{8}$$

$$s_{n+1}^2(x) \;=\; s_n^2(x) \;-\; \frac{k_n^2(x, x_{n+1})}{s_n^2(x_{n+1})}, \tag{9}$$

where $k_n(x, x_{n+1})$ is the kriging covariance between $f(x)$ and $f(x_{n+1})$, which can be computed using (7), i.e.,

$$k_n(x, x_{n+1}) \;=\; k(x, x_{n+1}) \;-\; \widetilde{\lambda}(x_{n+1}; \underline{x}_n)^{\mathsf{T}}\, \widetilde{k}(x, \underline{x}_n). \tag{10}$$

The kriging means and variances from $n+1$ observations can thus be obtained in $O(n^2+Mn)$ operations : $\widetilde{\lambda}(x_{n+1}; \underline{x}_n)$ is first obtained using $\widetilde{K}(\underline{x}_n)^{-1}$ (or a factorization of $\widetilde{K}(\underline{x}_n)$) in $O(n^2)$ operations and, for each $x$, $k_n(x, x_{n+1})$ is computed from (10) in $O(n)$ operations.

*Remark 2* The variance update formula (9) has been used to obtain computational savings in various kriging-based algorithms (see, e.g., Barnes and Watson, 1992; Gao et al., 1996; Emery, 2009; Gramacy and Lee, 2010).

*Remark 3* The coefficient $k_n(x, x_{n+1})/s_n^2(x_{n+1})$ appearing in (8) can be seen as the last component of the vector of weights $\lambda(x; \underline{x}_{n+1})$, as observed by Emery (2009).

## 3 Improving the numerical complexity of SUR strategies using update formulas for kriging

### 3.1 A general family of kriging-based sampling criteria

The kriging update formulas given in Section 2.2 makes it possible to produce efficient algorithms for a large family a kriging-based sampling criteria. More precisely, we will consider the computation of expressions of the form

$$C_n(x_{n+1}) \;=\; \mathbb{E}_n\left(\mathcal{H}(m_{n+1}, s_{n+1}^2) \mid X_{n+1} = x_{n+1}\right), \qquad x_{n+1} \in \mathbb{X}, \tag{11}$$

where $m_{n+1}(\,\cdot\,)$ and $s_{n+1}^2(\,\cdot\,)$ are the updated kriging mean and kriging variance functions assuming that $x_{n+1}$ has been added to the design of experiment, and $\mathcal{H}(\,\cdot\,)$ is a functional depending on a finite number $M$ of values of the updated kriging mean and variance. Note that, given $\mathcal{A}_n$, $m_{n+1}(\,\cdot\,)$ is a random field while $s_{n+1}^2(\,\cdot\,)$ is a deterministic function. The randomness of $m_{n+1}(\,\cdot\,)$ comes from the unknown response $f(x_{n+1})$ and we know that, conditionally on $\mathcal{A}_n$, $f(x_{n+1}) \sim \mathcal{N}(m_n(x_{n+1}), s_n^2(x_{n+1}))$.

*Example 1.* Consider the problem of estimating the volume $\alpha$ of an excursion set of a function $f$ above a threshold $T$ introduced in Section 1. Then, a SUR strategy to estimate $\alpha$ would consist in choosing the next evaluation point, at each step $n$, by minimizing the expectation of the variance of the volume of excursion:

$$J_n^{(\alpha)}(x_{n+1}) \;=\; \mathbb{E}_n\left(\mathrm{Var}_{n+1}(\alpha) \;\middle|\; X_{n+1} = x_{n+1}\right) . \tag{12}$$

This sampling criterion is considered in Bect et al. (2011) but has not been implemented because the numerical complexity of the approximation of $\mathrm{Var}_{n+1}(\alpha)$ was deemed too high. In fact, $J_n$ may be rewritten under the form

$$J_n^{(\alpha)}(x_{n+1}) \;=\; r_n \;-\; \mathbb{E}_n\left(\int_{\mathbb{X}\times\mathbb{X}} p_{n+1}(x)\, p_{n+1}(y)\, \mathsf{P}_{\mathbb{X}}(\mathrm{d}x)\, \mathsf{P}_{\mathbb{X}}(\mathrm{d}y) \;\middle|\; X_{n+1} = x_{n+1}\right),$$

where $r_n$ is a term that does not depend on $x_{n+1}$ and for $n \geq 1$

$$\begin{aligned} p_n : \mathbb{X} &\to [0,1] \\ x &\mapsto \mathbb{E}_n(\ _{\{f(x)>T\}}) \end{aligned} \tag{13}$$

($_{\{f(x)>T\}}$ equals one if $f(x) > T$ and zero otherwise), which can be easily computed from the kriging mean and variance (see Section 4). The second term is the only one that matters as far as choosing $x_{n+1}$ is concerned, and gives rise to a criterion of the form (11) after discretization of the double integral over $\mathbb{X} \times \mathbb{X}$.

*Example 2.* Consider a threshold $T \in \mathbb{R}$ and the problem of estimating the excursion set $\Gamma = \{x \in \mathbb{X} : f(x) \geq T\}$, which can be characterized by the random variable $_{\{f(x)>T\}}$. Since $_{\{f(x)>T\}}$ has variance $p_n(x)(1-p_n(x))$ given $\mathcal{A}_n$, $\int_{\mathbb{X}} p_n(1-p_n)\mathrm{d}\mathsf{P}_{\mathbb{X}}$, where $\mathsf{P}_{\mathbb{X}}$ is a probability measure on $\mathbb{X}$, can serve as a criterion to quantify the uncertainty about $\Gamma$. This leads us to define a SUR sampling criterion written as

$$J_n^{(\Gamma)}(x_{n+1}) \;=\; \mathbb{E}_n\left(\int_{\mathbb{X}} p_{n+1}(1-p_{n+1})\mathrm{d}\mathsf{P}_{\mathbb{X}} \;\middle|\; X_{n+1} = x_{n+1}\right), \tag{14}$$

which is again of the form (11) when the integral with respect to $\mathsf{P}_{\mathbb{X}}$ is approximated by Monte-Carlo:

$$J_n^{(\Gamma)}(x_{n+1}) \;\approx\; \mathbb{E}_n\left(\sum_{m=1}^{M} p_{n+1}(x^{(m)})\left(1 - p_{n+1}(x^{(m)})\right) \;\middle|\; X_{n+1} = x_{n+1}\right), \qquad x^{(1)}, \ldots, x^{(M)} \overset{\text{iid}}{\sim} \mathsf{P}_{\mathbb{X}}, \tag{15}$$

since $p_{n+1}(x^{(m)})$ is a function of $m_{n+1}(x^{(m)})$ and $s^2_{n+1}(x^{(m)})$. Note that $J_n^{(\Gamma)}$ is considered in Bect et al. (2011).

*Example 3.* Consider the problem of estimating the global maximum of $f$ over a bounded domain $\mathbb{X}$. Denote by $M_n = \max(f(X_1), \ldots, f(X_n))$ the maximum of evaluation results at step $n$, and

$$J_n^{(\mathrm{EI})}(x_{n+1}) = \mathbb{E}_n\left(M_{n+1} - M_n \;\middle|\; X_{n+1} = x_{n+1}\right)$$

the classical expected improvement (EI) criterion (see, e.g., Jones et al., 1998). Then, the *two-step EI* criterion (Osborne et al., 2009; Ginsbourger and Le Riche, 2010) is defined as

$$J_n^{(\mathrm{EI},\,2)}(x_{n+1}) = \mathbb{E}_n\left(\max_{x_{n+2}\in\mathbb{X}} \mathbb{E}_{n+1}\left(M_{n+2} - M_n \mid X_{n+2} = x_{n+2}\right) \;\middle|\; X_{n+1} = x_{n+1}\right),$$

and can be rewritten as

$$J_n^{(\mathrm{EI},\,2)}(x_{n+1}) \;=\; J_n^{(\mathrm{EI})}(x_{n+1}) + \mathbb{E}_n\left(\max_{x_{n+2}\in\mathbb{X}} J_{n+1}^{(\mathrm{EI})}(x_{n+2}) \mid X_{n+1} = x_{n+1}\right). \quad (16)$$

The second term of (16) is of the form (11) when the search for $x_{n+2}$ is restricted to a finite set of candidate points, since $J_{n+1}^{(\mathrm{EI})}(x_{n+2})$ is a function of $m_{n+1}(x_{n+2})$ and $s_{n+1}^2(x_{n+2})$.

*Example 4.* In a recent paper, Gramacy and Lee introduced the Integrated Expected Conditional Improvement criterion (see Gramacy and Lee (2010)). This criterion corresponds to a SUR strategy for carrying out constrained or non-constrained optimization. The expression of the IECI criterion for a maximization problem is as follows:

$$J_n^{(\mathrm{IECI})}(x_{n+1}) = \mathbb{E}_n\left\{ \int_{\mathbb{X}} \Big[(m_{n+1}(x) - \widehat{f}_{\max})\Phi(\gamma_{n+1}(x)) + s_{n+1}(x)\varphi(\gamma_{n+1}(x))\Big] \mathsf{P}_{\mathbb{X}}(dx) \;\Big|\; X_{n+1} = x_{n+1} \right\} \quad (17)$$

where

$$\widehat{f}_{\max} = \max_{x\in\mathbb{X}}(m_n(x)) \quad \text{and} \quad \gamma_{n+1}(x) := \frac{m_{n+1}(x) - \widehat{f}_{\max}}{s_{n+1}(x)}\,.$$

The integrand of $J_n^{(\mathrm{IECI})}(x_{n+1})$ is called the expected conditional improvement. This sampling criterion is again of the form (11) after discretization of the integral over $\mathbb{X}$.

### 3.2 Algorithms for the approximation of $C_n$, with or without update formulas

In this section we define and compare two algorithms for the approximate computation of $C_n\big(x_{n+1}^l\big)$, $1 \le l \le p$, for a certain number $p$ of candidate points, $x_{n+1}^1, \ldots, x_{n+1}^p$. The integration of the random variable $f(x_{n+1}^l)$ is carried out in both algorithms using a Gauss-Hermite quadrature formula. More precisely, if we denote by $(z_k, w_k)$, $1 \le k \le$

$Q$ the quadrature points and weights of the $Q$-points Gauss-Hermite quadrature, then both algorithms are actually computing

$$\widehat{C}_n\left(x_{n+1}^l\right) \;:=\; \sum_{k=1}^{Q} w_k\, \mathcal{H}\left(m_{n+1,l,k}, s_{n+1,l}^2\right), \qquad 1 \le l \le p\,, \tag{18}$$

where $m_{n+1,l,k}$ and $s_{n+1,l}^2$ are the updated kriging mean and variances, assuming that $x_{n+1}^l$ is added to the design and that $f(x_{n+1}^l) = y_k^l := m_n(x_{n+1}^l) \,+\, z_k\, s_n(x_{n+1}^l)$.

The first algorithm (see Algorithm 1) is a straightforward generalization of the existing algorithm for SUR criterions (Vazquez and Bect, 2009; Bect et al., 2011), which does not make use of the update formulas. The complexity for the evaluation of $\widehat{C}_n$ at $p$ candidate points using this algorithm is $O\left(p\left(n^3 + Mn^2 + MQ + \kappa_M Q\right)\right)$, where $\kappa_M$ denotes the complexity of the algorithm that computes $\mathcal{H}$. Note that the computations of the kriging weights and variances can be done outside the inner loop (i.e., the loop on quadrature points) thanks to the fact that neither the weights nor the variances depend on $f(x_{n+1}^l)$.

The second algorithm (see Algorithm 2) is a new algorithm, which brings the complexity down to $O\big((n^3 + Mn^2 \,+ p(n^2 + Mn + MQ + \kappa_M Q)\big)$ using the update formulas presented in Section 2.2. Note that Algorithm 2 has the same complexity as Algorithm 1 when used for a single candidate point (i.e., with $p = 1$). The superiority of Algorithm 2 becomes apparent only when $p$ is large. Indeed, a significant part of the linear algebraic computations—with a complexity of $O\big(n^3 + Mn^2\big)$—is now performed outside the outer loop, i.e., outside the loop on candidate points. As a result, each additional candidate point comes with a cost of $O\big(n^2 + Mn + MQ + \kappa_M Q\big)$ using Algorithm 1, a significant improvement with respect to the $O\big(n^3 + Mn^2 + MQ + \kappa_M Q\big)$ of Algorithm 1.

*Remark 4* The complexity $\kappa_M$ grows at least linearly with $M$ in all the examples of this paper. In this case, the complexity Algorithm 1 can be simplified to $O\big(p(n^3 + Mn^2 + \kappa_M Q)\big)$ and that of Algorithm 2 to $O\big((n^3 + Mn^2 \,+ p(n^2 + Mn + \kappa_M Q)\big)$.

## 4 Additional simplifications and speed-ups for the SUR strategies

In this section, we provide additional analytical formulas allowing to speed-up the calculation time of several criteria of the form (11).

### 4.1 General idea to simplify a criterion of the form (11)

Any criterion of the form (11) consist of the expectation at step $n$ of a random variable $\mathcal{H}(m_{n+1}, s_{n+1}^2)$ depending on the unknown $f(x_{n+1})$. In the kriging framework, conditionally on $\mathcal{A}_n$, we know that $f(x_{n+1}) \sim \mathcal{N}(m_n(x_{n+1}), s_n^2(x_{n+1}))$, or equivalently

$$f(x_{n+1}) \;=\; m_n(x_{n+1}) + U\, s_n(x_{n+1}) \tag{19}$$

**Algorithm 1** Computation of $\widehat{C}_n$ at $p$ candidate points (without the update formulas). The complexity of each step is indicated in bold.

**Require:** $n$ observations of the simulator: $\mathcal{A}_n$
**Require:** $M$ integration points $x^{(1)}, ..., x^{(M)}$
**Require:** $p$ candidate points $x^1_{n+1}, ..., x^p_{n+1}$
**Require:** quadrature points and weights: $(z_1, ..., z_Q)$, $(w_1, ..., w_Q)$

**for** $l = 1 \to p$ **do**
  Compute the inverse (or factorize) the matrix $\widetilde{K}(\underline{x}_{n+1})$ defined by (5) assuming that the point $x^l_{n+1}$ has been added to the design of experiments. $\mathbf{O(n^3)}$
  **for** $j = 1 \to M$ **do**
    compute the vector $\widetilde{\lambda}(x^{(j)}; \underline{x}_{n+1})$ using equation (5). $\mathbf{O(n^2)}$
    compute the kriging variance $s^2_{n+1}(x^{(j)})$ using equation (7). $\mathbf{O(n)}$
    precompute the sum $\sum_{i=1}^n \lambda_i(x^{(j)}).f(x_i)$, where $\lambda_i(x^{(j)})$ is the kriging weight of observation $i$ for the kriging prediction at point $x^{(j)}$. $\mathbf{O(n)}$
    **for** $k = 1 \to Q$ **do**
      compute the kriging mean: $m_{n+1}(x^{(j)})$ assuming that $f(x_{n+1}) = y^l_k$. The precomputation is useful as: $m_{n+1}(x^{(j)}) = \sum_{i=1}^{n+1} \lambda_i(x^{(j)}).f(x_i)$. $\mathbf{O(1)}$
    **end for**
  **end for**
  **for** $k = 1 \to Q$ **do**
    compute the functional $\mathcal{H}(m_{n+1,k}, s^2_{n+1})$. $\mathbf{O}(\kappa_\mathbf{M})$
  **end for**
  compute the criterion $\widehat{C}_n(x^l_{n+1})$. $\mathbf{O(Q)}$
**end for**
Total cost: $\mathbf{O(p(n^3 + M.n^2 + M.Q + \kappa_M.Q))}$

in distribution, with $U$ a $\mathcal{N}(0,1)$ random variable that is independent of $\mathcal{A}_n$. Also, one can see using (8) that the updated kriging mean $m_{n+1}$ is a random field having a very simple dependency with respect to the random variable $f(x_{n+1})$. Indeed, combining equations (8) and (19) results in the functional equality

$$m_{n+1} = \; m_n + U\, \nu_{n+1}, \tag{20}$$

with $\nu_{n+1}(x) := k_n(x, x_{n+1})/s_n(x_{n+1})$. Pluging (20) into (11) yields

$$\mathbb{E}_n\left(\mathcal{H}(m_{n+1}, s^2_{n+1}) \mid X_{n+1} = x_{n+1}\right) = \frac{1}{\sqrt{2\pi}} \int_{\mathbb{R}} \mathcal{H}\left(m_n + u\,\nu_{n+1}, s^2_{n+1}\right) e^{-\frac{u^2}{2}}\, \mathrm{d}u. \tag{21}$$

For specific functionals $\mathcal{H}$, integrals of the form (21) can be calculated analytically. Three examples of such functionals will be given in the forthcoming Section 4.2.

**Algorithm 2** Computation of $\widehat{C}_n$ at $p$ candidate points (using the update formulas). The complexity of each step is indicated in bold.

**Require:** $n$ observations of the simulator: $\mathcal{A}_n$
**Require:** $M$ integration points $x^{(1)}, ..., x^{(M)}$
**Require:** $p$ candidate points $x_{n+1}^1, ..., x_{n+1}^p$
**Require:** quadrature points and weights: $(z_1, ..., z_Q)$, $(w_1, ..., w_Q)$

evaluate the matrix $\widetilde{K}(\underline{x}_n)^{-1}$ given in equation (5). $\mathbf{O(n^3)}$
evaluate the current kriging variances and means $s_n^2(x^{(j)}), m_n(x^{(j)})$ at the $M$ integration points. $\mathbf{O(M.n^2)}$
**for** $l = 1 \to p$ **do**
  compute the vector $\widetilde{\lambda}(x_{n+1}^l, \underline{x}_n)$ using equation (5): $\mathbf{O(n^2)}$
  compute the kriging variance $s_n^2(x_{n+1}^l)$. $\mathbf{O(n)}$
  **for** $j = 1 \to M$ **do**
    compute $k_n(x^{(j)}, x_{n+1}^l)$ using precomputations and equation (7). $\mathbf{O(n)}$
    compute the updated kriging variance $s_{n+1}^2(x^{(j)})$ using update formula (9). $\mathbf{O(1)}$
    **for** $k = 1 \to Q$ **do**
      assume that $f(x_{n+1}^l) = y_k^l$ and compute the updated kriging mean $m_{n+1}(x^{(j)})$ using update formula (8). $\mathbf{O(1)}$
    **end for**
  **end for**
  **for** $k = 1 \to Q$ **do**
    compute the functional $\mathcal{H}(m_{n+1,k}, s_{n+1}^2)$. $\mathbf{O}(\kappa_\mathbf{M})$
  **end for**
  compute the criterion $\widehat{C}_n(x_{n+1}^l)$. $\mathbf{O(Q)}$
**end for**
Total cost: $\mathbf{O(n^3 + M.n^2 + p(n^2 + M.n + M.Q + \kappa_M.Q))}$

We first give some analytical formulas that will help to get simplified expressions using equation (21). The proofs of these results are given in Appendix A.

$$K_{a,b} := \int_{\mathbb{R}} \varphi(a+bu)\, e^{-\frac{u^2}{2}} du = \frac{1}{\sqrt{1+b^2}} \exp\left(-\frac{1}{2}a^2\left(1-\widetilde{b}^2\right)\right), \tag{22}$$

$$L_{a,b} := \int_{\mathbb{R}} \Phi(a+bu)\, e^{-\frac{u^2}{2}} du = \sqrt{2\pi}\, \Phi(\widetilde{a}), \tag{23}$$

$$M_{a,b} := \int_{\mathbb{R}} u\, \Phi(a+bu)\, e^{-\frac{u^2}{2}} du = b\, K_{a,b}, \tag{24}$$

$$N_{a,b,c,d} := \int_{\mathbb{R}} \Phi(a+bu)\, \Phi(c+du)\, e^{-\frac{u^2}{2}} du = \sqrt{2\pi}\, \Phi_2\left(\widetilde{a}, \widetilde{c}, \widetilde{b}\,\widetilde{d}\right), \tag{25}$$

where $\Phi$ and $\varphi$ are respectively the c.d.f. and the p.d.f. of the standard univariate Gaussian distribution, $\Phi_2(u_1, u_2, \rho)$ is the c.d.f. of the standard bivariate Gaussian distribution with correlation $\rho$, and $\widetilde{a} := \frac{a}{\sqrt{1+b^2}}$, $\widetilde{b} := \frac{b}{\sqrt{1+b^2}}$, $\widetilde{c} := \frac{c}{\sqrt{1+d^2}}$, $\widetilde{d} := \frac{d}{\sqrt{1+d^2}}$.

Very efficient numerical procedures are available to compute the multivariate Gaussian c.d.f. : Genz (1992) wrote routines in Fortran77 which have been wrapped in many R Packages (mnormt, pbivnorm, mvtnorm available on CRAN). In our code, we use the pbivnorm package which is dedicated to the bivariate case and is slightly faster when many calls of the bivariate c.d.f. are done at a time.

4.2 Closed-form expressions for several SUR strategies

In this section, we give a simplified expression of three sampling criteria of the form (11) using the ideas presented in the previous section. These expressions are useful because they allow us to by-pass the numerical computation of an integral over $\mathbb{R}$, with $Q$ quadrature points and weights. The integrals are rewritten in terms of Gaussian c.d.f. and its multivariate version, which can be implemented efficiently. The proofs are based on the numerical results given in Appendix B.

*Criterion* $J_n^{(\alpha)}$. A first idea to compute the conditional variance in $J_n^{(\alpha)}$ is to rely on the simulation of sample paths of $f$. Simulating $N$ (unconditioned) sample path of a Gaussian Process at $M$ points requires the Cholesky factorization of a matrix of size $M \times M$ and $N$ matrix-vector products. The complexity of this task alone is $O(M^3 + M^2.N)$. In fact, the numerical computation of $J_n^{(\alpha)}$ can be greatly simplified by re-writing $J_n^{(\alpha)}$ as

$$J_n^{(\alpha)}(x_{n+1}) = \; c_n - \int_{\mathbb{X}\times\mathbb{X}} \Phi_2\left(\widetilde{a}(x), \widetilde{a}(y), \widetilde{b}(x)\widetilde{b}(y)\right) \mathsf{P}_{\mathbb{X}}(\mathrm{d}x)\mathsf{P}_{\mathbb{X}}(\mathrm{d}y) \tag{26}$$

where $c_n$ is a quantity not depending of $x_{n+1}$, $\widetilde{a}(x) := \frac{a(x)}{\sqrt{1+b^2(x)}}$, and $\widetilde{b}(x) := \frac{b(x)}{\sqrt{1+b^2(x)}}$. Expression (26) can also be approximated by Monte-Carlo. If a sample of size $M^2$ is used to carry out the Monte Carlo integration on $\mathbb{X} \times \mathbb{X}$, the cost for computing $J_n^{(\alpha)}$ is $O(n^2 + M.n + M^2)$, which makes the criterion affordable for reasonable values of $M$ (see Section 5.3.1).

*Criterion* $J_n^{(\Gamma)}$. Using the same ideas, $J_n^{(\Gamma)}$ can be rewritten as

$$J_n^{(\Gamma)}(x_{n+1}) = \int_{\mathbb{X}} \Phi_2\left(\frac{a(x)}{\sqrt{1+b^2(x)}}, \frac{-a(x)}{\sqrt{1+b^2(x)}}, \frac{-b^2(x)}{1+b^2(x)}\right) \mathsf{P}_{\mathbb{X}}(\mathrm{d}x) \tag{27}$$

where $a(x) := (m_n(x) - T)/s_{n+1}(x)$ and $b(x) := \nu_{n+1}(x)/s_{n+1}(x)$. The integral over $\mathbb{X}$ in (27) can be approximated by Monte-Carlo sampling. Using $M$ integration points $x^{(1)}, ..., x^{(M)}$, $J_n^{(\Gamma)}(x_{n+1})$ can be computed in $O(n^2 + M.n)$ operations instead of $O(n^2 + M.n + M.Q)$ operations (using Algorithm 2), because the approximation of an integral over $\mathbb{R}$ relying on $Q$ quadrature points and weights has been removed.

*Criterion* $J_n^{(\mathrm{IECI})}$. The Integrated Expected Conditional Improvement criterion given by (17) can be rewritten as

$$J_n^{(\mathrm{IECI})}(x_{n+1}) = \frac{1}{\sqrt{2\pi}} \int_{\mathbb{X}} \Big( v_1(x) L_{v_3(x),v_4(x)} + \Big(\frac{v_2^2(x)}{s_{n+1}(x)} + s_{n+1}(x)\Big) K_{v_3(x),v_4(x)} \Big) \mathsf{P}_{\mathbb{X}}(\mathrm{d}x) \tag{28}$$

$$\text{where } \begin{cases} v_1(x) := m_n(x) - \widehat{f}_{\max}, \\ v_2(x) := \nu_{n+1}(x), \\ v_3(x) := v_1(x)/s_{n+1}(x), \\ v_4(x) := v_2(x)/s_{n+1}(x). \end{cases} \tag{29}$$

and $K_{.,.}$, $L_{.,.}$, $M_{.,.}$ are given by (22), (23) and (24). The gain in complexity for this criterion is the same than for $J_n^{(\Gamma)}$: from $O(n^2 + M.n + M.Q)$ to $O(n^2 + M.n)$.

## 5 Application : nuclear safety test case

In this section, we illustrate our algorithms on a nuclear safety application. The aim of this section is twofold. First, we show the improvements provided by Algorithm 2 and the techniques presented Section 4 in terms of computation time. Second, we test and compare two SUR strategies (using $J_n^{(\alpha)}$ and $J_n^{(\Gamma)}$) to estimate the excursion set of a quantity related to the safety of a system involving fissile material.

### 5.1 Presentation of the test case

Any system involving fissile materials may produce a chain reaction based on neutrons, which are both a product and an initiator of fission reactions. Besides a possible over-production of neutrons, a high quantity of energy is also generated and therefore needs to be controlled. Nuclear criticality safety assessment aims at avoiding such "criticality accident" within the range of operational conditions of a given system, described by several physical and chemical parameters affecting the chain reaction efficiency.

In order to check subcriticality of the studied system, the neutrons multiplication factor (also known as k-effective, denoted $k_{\text{eff}}$) — a parameter describing the neutrons production ratio — is estimated using a simulator. While a $k_{\text{eff}}$ of 1.0 — meaning that the neutron population is constant — is the target of a nuclear power plant, all fuel management applications (transport, storage) aim at a $k_{\text{eff}}$ strictly lower than this critical value.

In the present application, the French Institute for Radiological Protection and Nuclear Safety (IRSN) uses a Markov Chain Monte Carlo (MCMC) code to calculate the $k_{\text{eff}}$ of a system depending on many parameters (quantity and density of fissile material, geometry of the system, quantity of water in the system, temperature,...). The key task is to locate the set of parameters $\Gamma^\star$ where the $k_{\text{eff}}$ is higher than 0.95. The threshold $T = 0.95$ corresponds to the critical value of 1.0 minus a regulation margin. The MCMC code calculating the $k_{\text{eff}}$, called MORET, may be considered expensive as the calculation of $k_{\text{eff}}$ for one set of parameters takes approximately 5 to 30 minutes on a 2.5GHz processor, depending on the numerical conditioning of the model. An exhaustive evaluation strategy (a full factorial design for example) is consequently out of reach when the number of parameters is high (up to ten here).

In the present application case, the system is a storage facility of plutonium powder, whose criticality is controlled by two real valued input parameters:

- the mass of plutonium (MassePu) in the system

– the logarithm of the concentration of plutonium (logConcPu).

The input domain $\mathbb{X}$ is:
(MassePu,logConcPu) $\in \mathbb{X} = [0.1, 2] \times [2, 9.6]$ and mathematically our goal is locate the excursion set $\Gamma^\star$ having a cautious evaluation strategy of our simulator $f$, where $f(\mathbf{x}) = k_{\text{eff}}$(MassePu,logConcPu).
In this particular example in dimension two we were able to run a high number of evaluations in order to identify accurately the excursion set and its volume. Figure 1 shows the set $\Gamma^\star$ obtained after 300 evaluations of the code on a space filling design of experiment. The volume of the excursion set is equal to 22.68% of the volume of $\mathbb{X}$. Note that in Section 5.3.2, our aim is to determine whether the SUR strategy can identify accurately the excursion set with a small fraction of this budget of 300 evaluations.

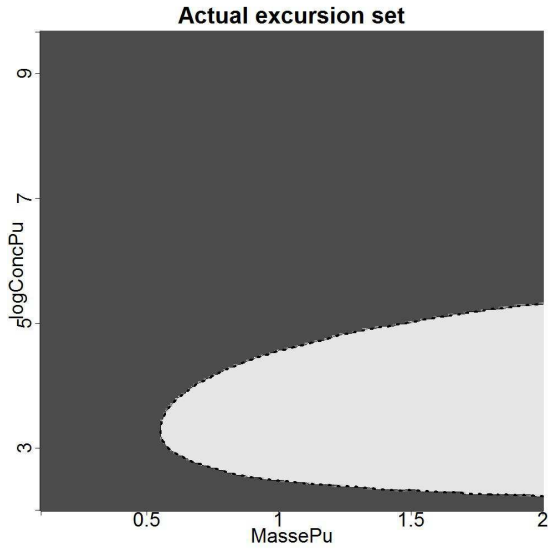


**Fig. 1** Excursion set (in white) of the function $k_{\text{eff}}$ = f(Masse Pu, logConcPu) for a threshold $T = 0.95$.

## 5.2 Gaussian Process modeling

Prométhée is a workbench for parametric computing developed by the IRSN allowing to run any code like MORET. The results presented in the next two subsections are obtained using our algorithms written in R language and wrapped by Promethee. All our algorithms rely on the objects and functions of the DiceKriging package (Roustant et al. (2011)).
Among the possible covariance kernels available in DiceKriging, we choose to work with a separable Matérn covariance with parameters estimated by maximum likelihood from

the observations and re-estimated after each iteration. Recall that Matérn covariance kernels depend of a parameter $\nu$ that we here choose to be equal to 3/2. In this particular case the separable Matérn covariance has the following expression:

$$k_\theta(x,y) = \sigma^2 \prod_{i=1}^{d} \left(1 + \frac{\sqrt{3}h_i}{\theta_i}\right) \exp\left(-\frac{\sqrt{3}h_i}{\theta_i}\right), \tag{30}$$

where $\theta = (\theta_1, ..., \theta_d)$ are scale parameters, $\sigma^2$ is the variance of the non-conditioned Gaussian process and $h_i := x_{[i]} - y_{[i]}$ where $x_{[i]}$ denotes the $i^{th}$ coordinate of $x$. In our test case we have $d = 2$ and the estimated parameters are denoted $\widehat{\sigma}, \widehat{\theta}_{\text{MassePu}}, \widehat{\theta}_{\text{logConcPu}}$. We also estimate the trend which is considered as an unknown constant $\widehat{\beta}_1$.

## 5.3 Results

### *5.3.1 Performance tests: computation time of the criteria $J_n^{(\Gamma)}$ and $J_n^{(\alpha)}$*

We present here some pure performance tests done with our $R$ code, on the nuclear safety test function $k_{\text{eff}}$(MassePu,logConcPu). The objective is to quantify the computation time improvement given by Algorithm 2 and by the formulae in Section 4 for computing the $J_n^{(\Gamma)}$ criterion. We run only one iteration of the $J_n^{(\Gamma)}$ or $J_n^{(\alpha)}$ criterion starting with an initial design of experiment of $n$ points, and look at the computation time, which is simply the time of $p$ evaluations of the criterion. Such tests are done using $M$ integration points. We choose $p = 2500$ (corresponding to a $50 \times 50$ grid) and let $n$ and $M$ vary. We test only one value for $p$ as the complexity in $p$ is simply an affine function of $p$. A value of 2500 for $p$ corresponds to a realistic budget for optimizing the criteria on the whole input set provided that its dimension $d$ is not too large. Also, note that when $p$, $n$ and $M$ are fixed, the computation time of one iteration do not depend of $d$.
For the $J_n^{(\Gamma)}$ criterion, two implementations of the criterion are tested:

- one implementation with Algorithm 1 (using $q = 10$ quadrature points and weights),
- one implementation with Algorithm 2 and with the formulae of Section 4.

The $J_n^{(\alpha)}$ criterion is implemented with Algorithm 2 and with the formulae of Section 4. We do not propose a "slow" implementation of $J_n^{(\alpha)}$ (using Gaussian Process simulations) and thus have only the "fast" algorithm for this criterion. Recall that, for each evaluation of $J_n^{(\alpha)}$ we do $M^2$ calls to the function $\Phi_2$ instead of $M$, which affects the computation time when $M$ is large. In order to keep a reasonable computation time for $J_n^{(\alpha)}$, we did not test the speed of this criterion for values of $M$ higher than 1000. For the $J_n^{(\Gamma)}$ criterion we took a maximum value of 10000 for $M$.

The theoretical complexities of our algorithms are indicated in Table 1. We denote the computation times as follows: $J^{(\Gamma)}_{n,M,\text{slow}}$, $J^{(\Gamma)}_{n,M,\text{fast}}$ and $J^{(\alpha)}_{n,M,\text{fast}}$ for respectively the slow implementation of $J_n^{(\Gamma)}$, the fast one and the implementation of $J_n^{(\alpha)}$. A computation time comparison is given in Figure 2 for different values of $n$ and $M$. The tested values for $M$ are going from 100 to 10000.

We can see that the improvement provided by Algorithm (2) with respect to Algorithm (1) for the implementation of $J_n^{(\Gamma)}$ is important, especially when $n$ is large. This is mainly due to the $O(M.n)$ complexity versus $O(M.n^2)$ to compute the updated kriging variances. Also the $J_n^{(\alpha)}$ criterion, which was previously considered computationally unaffordable, has a rather quick computation when $M$ is not large, i.e. not higher than 600. When $M$ is large the $M^2$ calls to the $\Phi_2$ function make the $J_n^{(\alpha)}$ criterion expensive to evaluate. This is a major drawback when a large value is required for $M$ (e.g., when $d$ is large). However a "naive" implementation of $J_n^{(\alpha)}$ using Gaussian process simulations also prevents using high values for $M$ because of the $M \times M$ matrix inversion (with a $O(M^3)$ cost) required to simulate conditional Gaussian Processes realisations in $M$ points.

**Table 1** Theoretical complexity for computing one iteration of a SUR strategy using the $J_n^{(\alpha)}$ or $J_n^{(\Gamma)}$ sampling criterion

| algorithm | complexity |
|---|---|
| Standard implementation of $J_n^{(\Gamma)}$, (Algorithm 1) | $O(p(n^3 + M.n^2 + M.q))$ |
| Improved implementation of $J_n^{(\Gamma)}$, (Algorithm 2) | $O(n^3 + M.n^2 + p(n^2 + M.n))$ |
| Improved implementation of $J_n^{(\alpha)}$, (Algorithm 2) | $O(n^3 + M.n^2 + p(n^2 + M.n + M^2))$ |

*5.3.2 Sequential inversion applied to nuclear safety*

In this section, we run a stepwise inversion of the nuclear safety function shown on Figure 1. The $J_n^{(\Gamma)}$ sampling criterion is used with an initial design of experiment of $n = 6$ points, $M = 625$ integration points and $p = 2500$ points evaluated for the optimisation of $J_n^{(\Gamma)}$. As it is now possible to compute efficiently the $J_n^{(\alpha)}$ criterion in a reasonable time we also calculate its value on the $p$ points in order to see if a strategy using the $J_n^{(\Gamma)}$ criterion would provide the same sample than a strategy with the $J_n^{(\alpha)}$ criterion. Here, the calculation of the $J_n^{(\alpha)}$ criterion is only indicative; the new sample is chosen at each iteration at the point that minimizes $J_n^{(\Gamma)}$. We consider the following outputs:

- the evolution of the maximum likelihood estimator of the covariance parameters and of the trend (shown on Figure 4)
- the uncertainty $H_n^{(\Gamma)}$, which is the quantity $\int_{\mathbb{X}} p_n(1-p_n)\mathrm{d}\mathsf{P}_{\mathbb{X}}$, given the $n$ observations. The $J_n^{(\Gamma)}$ criterion aims at reducing this uncertainty.
- the uncertainty $H_n^{(\alpha)}$, which is the quantity $Var_n(\alpha)$. The $J_n^{(\alpha)}$ criterion aims at reducing this uncertainty.
- the volume error, denoted by $\varepsilon$, which is equal to the difference between the real volume and the estimated volume $\int_{\mathbb{X}} p_n \mathrm{d}\mathsf{P}_{\mathbb{X}}$, given the $n$ observations. In this example, the real volume is equal to 0.2268.
- $J_n^{(\Gamma)}(x^\star)$ and $J_n^{(\alpha)}(x^\star)$, which are the values of the $J_n^{(\Gamma)}$ and $J_n^{(\alpha)}$ criteria at the point $x^\star$ minimizing (on a 50×50 grid) the $J_n^{(\Gamma)}$ criterion at the considered iteration. Note that the minimizer of the $J_n^{(\alpha)}$ criterion, can be different from $x^\star$.

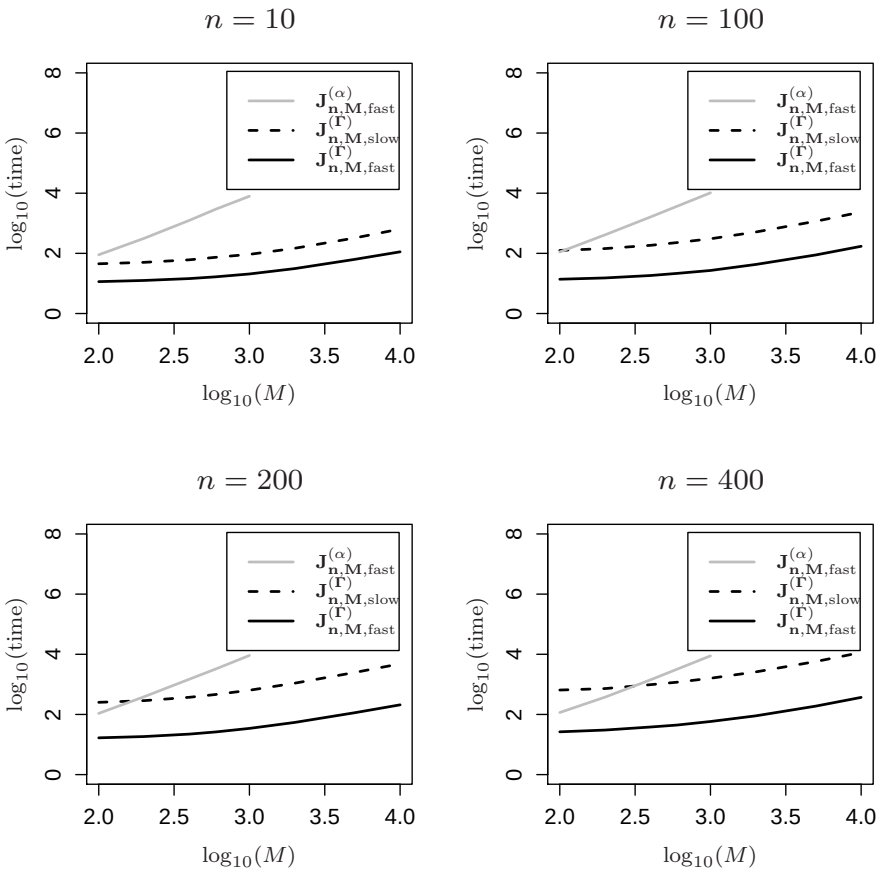


**Fig. 2** Computation times (in seconds) as a function of $n$ and $M$ for two implementations of the $J_n^{(\Gamma)}$ criterion and for a fast implementation of the $J_n^{(\alpha)}$ criterion

Figure 3 shows that at each iteration, $J_n^{(\Gamma)}(x^\star)$ is lower than $H_n^{(\Gamma)}$. This is not a surprise as $J_n^{(\Gamma)}(x^\star)$ is the expectation of the future uncertainty $H_{n+1}^{(\Gamma)}$ if $x^\star$ is added to the design of experiments. The same applies for respectively $J_n^{(\alpha)}(x^\star)$ and $H_n^{(\alpha)}$. Also, we can note that at some iterations (e.g. iteration two) the point $x^\star$ is not the minimizer of the $J_n^{(\alpha)}$ criterion, as $\min(J_n^{(\alpha)})$ is lower than $J_n^{(\alpha)}(x^\star)$.

We can also note that $H_n^{(\Gamma)}$ converges almost monotonically toward zero. This could be anticipated as the goal of the $J_n^{(\Gamma)}$ criterion is precisely to reduce $H_n^{(\Gamma)}$. It is however possible to have an increasing $H_n^{(\Gamma)}$ (e.g. iteration six and seven). This can be due to the re-evaluation of the covariance parameters at each iteration, but it is also possible to observe such phenomenon even with no covariance parameters re-estimation. Regarding the volume error, we can note that $H_n^{(\alpha)}$ is closely related to the square of the volume error. Indeed, if we denote $\alpha^\star$ the real volume and $\alpha$ the

random volume of the excursion set (recall that $\mathbb{E}_n(\alpha) = \int_{\mathbb{X}} p_n \mathrm{dP}_{\mathbb{X}}$) we have:

$$\begin{aligned} \varepsilon^2 =& (\alpha^\star - \mathbb{E}_n(\alpha))^2 \\ H_n^{(\alpha)} =& \mathbb{E}_n\left((\alpha - \mathbb{E}_n(\alpha))^2\right) \end{aligned}$$

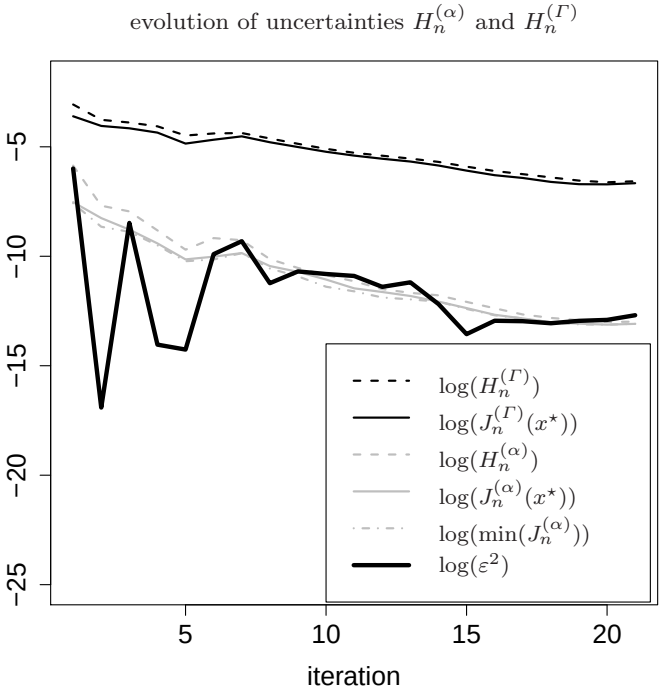


**Fig. 3** Evolution of $H_n^{(\Gamma)}$ and $H_n^{(\alpha)}$ during the inversion. The optimum $J_n^{(\Gamma)}(x^\star)$ of the $J_n^{(\Gamma)}$ criterion is usually lower than $H_n^{(\Gamma)}$ and corresponds to the expectation of $H_n^{(\Gamma)}$ at the next iteration. Regarding the $J_n^{(\alpha)}$ criterion, the curves $min(J_n^{(\alpha)})$ and $J_n^{(\alpha)}(x^\star)$ coincide only when $x^\star$ is also the minimizer of $J_n^{(\alpha)}$.

The evolutions of the inversions and of the $J_n^{(\Gamma)}$ and $J_n^{(\alpha)}$ criteria evaluated on the whole domain are shown on Figures 5, 6 and 7. One can see that here, the excursion set is accurately identified in few iterations of the SUR strategy. After 15 evaluations (i.e. 9 iterations), the function $p_n(x)$ does not depart much from the true function $_{x\in\Gamma}$. The plot of the $J_n^{(\Gamma)}$ criterion shows the areas of interest which are basically the areas with both a high kriging variance and a value near 0.5 for $p_n(x)$ (meaning that $m_n(x)$ is near the threshold). The plot of the $J_n^{(\alpha)}$ criterion is informative only and is provided to show that, most of the times, the $J_n^{(\Gamma)}$ criterion a good proxy of the much more expensive $J_n^{(\alpha)}$ criterion. Indeed, after 6, 9, 12 and 18 evaluations of MORET, the minimizer of $J_n^{(\Gamma)}$ is located in the same region than the minimizer of $J_n^{(\alpha)}$. However,

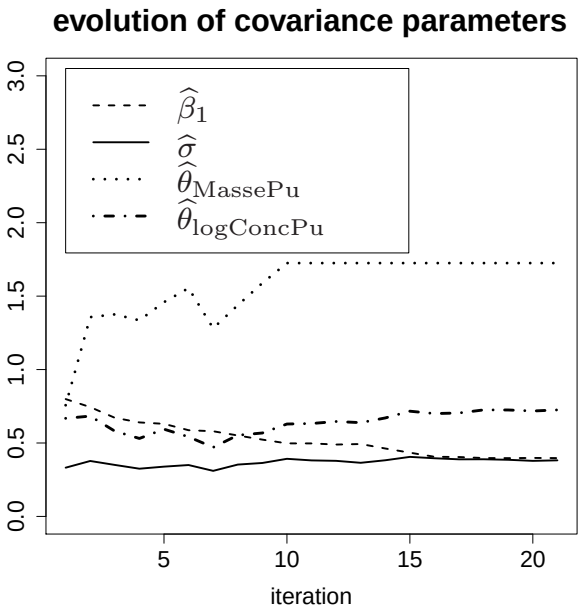


**Fig. 4** Estimated covariance parameters and trend during the sequential inversion. The Maximum Likelihood Estimator of these parameters has little variations after 10 iterations.

the minimizer of $J_n^{(\alpha)}$ often seems to be a clear optimum whereas for $J_n^{(\Gamma)}$ we often have several local optimum with almost the same value (e.g. after 9 or 12 evaluations). According to these plots, the $J_n^{(\Gamma)}$ criterion appears to be more exploratory than $J_n^{(\alpha)}$, with a larger area of interest.

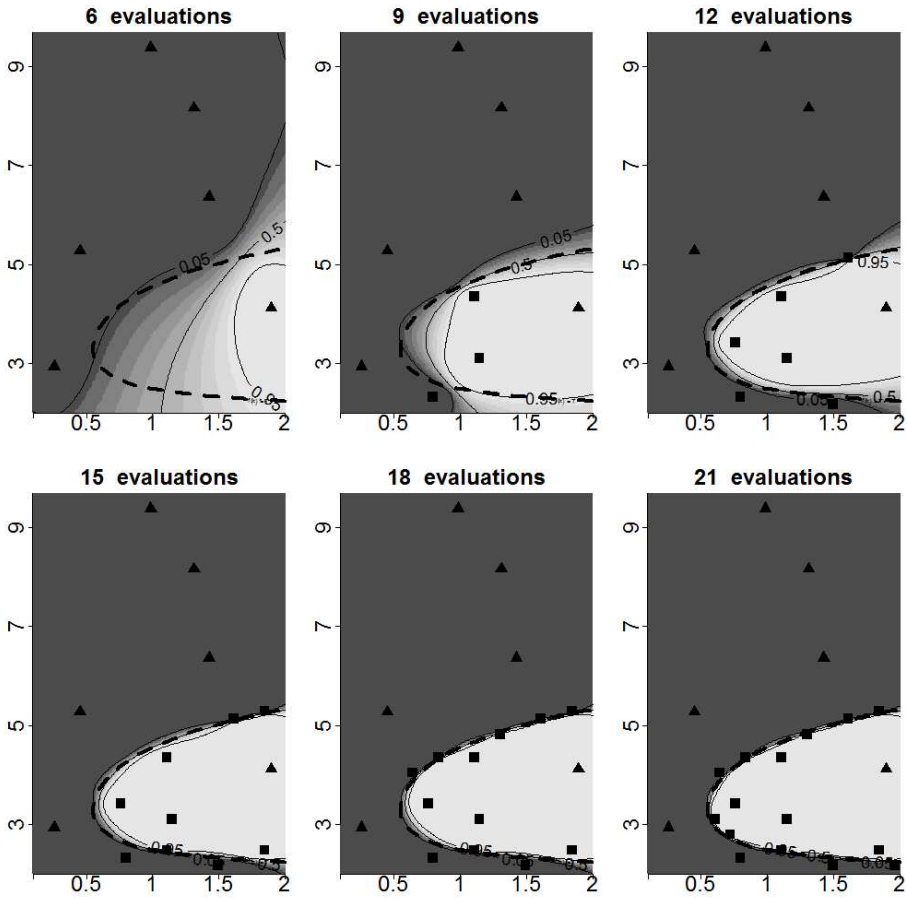


**Fig. 5** Plot of the function $p_n(x) = P_n(x \in \Gamma) = \Phi\left(\frac{m_n(x)-T}{s_n(x)}\right)$ after $n$ evaluations of the MORET code. The triangles are the six points of the initial DOE. The squares are the points sampled using the $J_n^{(\Gamma)}$ criterion. Areas in black correspond to $p_n(x) \approx 0$ and areas in white correspond to $p_n(x) \approx 1$. The dotted line indicates the true excursion set (see Figure 1). The contour lines indicate the three level sets $p_n(x) = 0.05, 0.5$ and $0.95$.

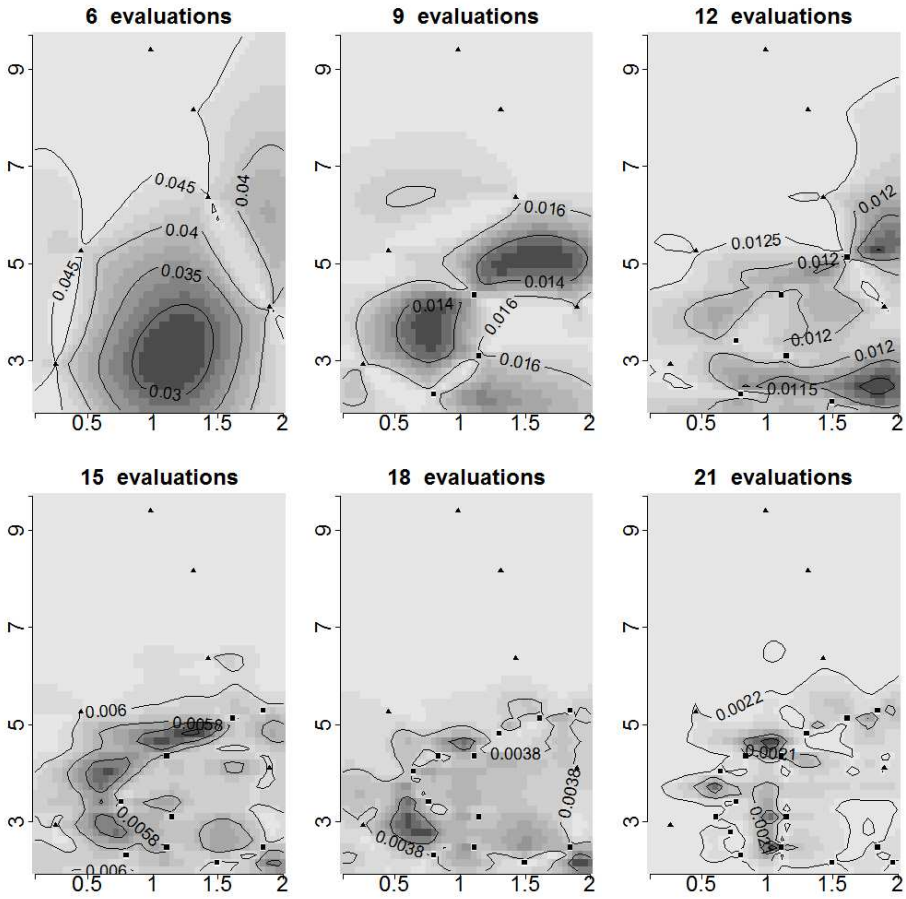


**Fig. 6** Plot of the value of the criterion $J_n^{(\Gamma)}(x) = \mathbb{E}_n\left(\int_{\mathbb{X}} p_{n+1}(1-p_{n+1})\mathrm{d}\mathrm{P}_{\mathbb{X}}\right)$ after $n$ evaluations of the MORET code. At each iteration, the point chosen for sampling is the optimum of this function. Areas in black correspond to potentially interesting areas, where the uncertainty $H_n^{(\Gamma)}$ is reduced the most in expectation.

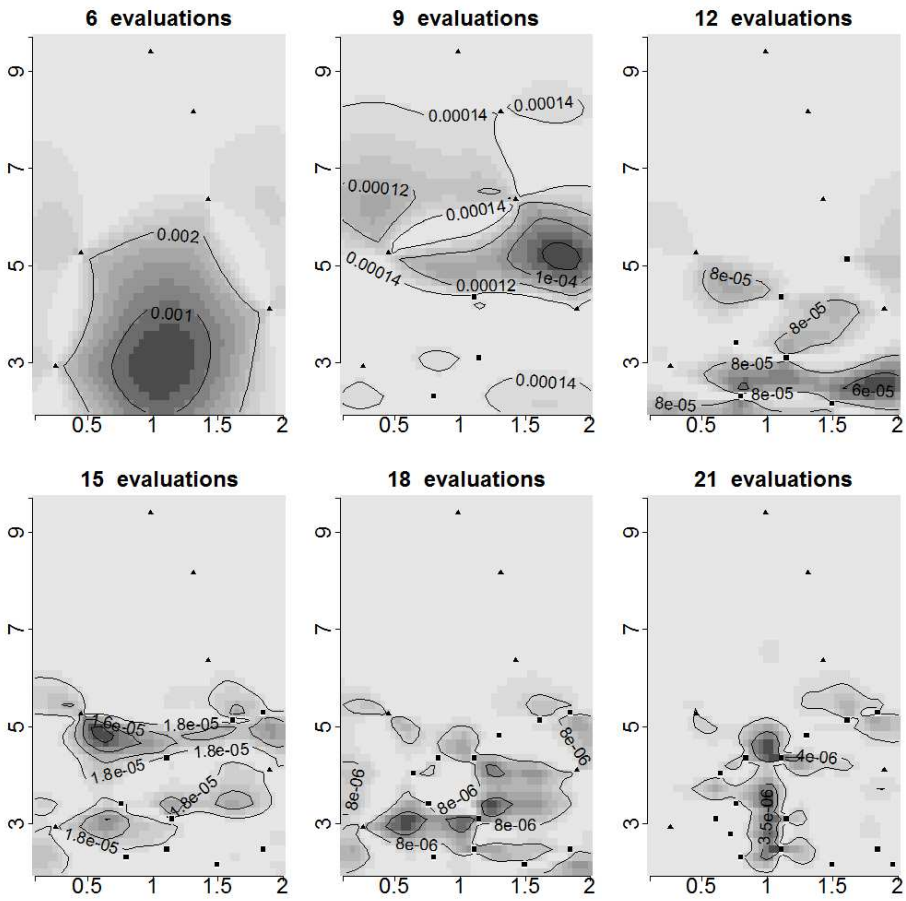


**Fig. 7** Plot of the value of the criterion $J_n^{(\alpha)}(x) = \mathbb{E}_n(Var_{n+1}(\alpha))$ after $n$ evaluations of the MORET code. At each iteration, the point chosen for sampling is the optimum of the $J_n^{(\Gamma)}$ function, and not $J_n^{(\alpha)}$. The goal is then simply to compare the $J_n^{(\Gamma)}$ and $J_n^{(\alpha)}$ criterion. In particular we are interested to see if they have the same optimum. Areas in black correspond to potentially interesting areas, where the uncertainty $H_n^{(\alpha)}$ is reduced the most in expectation.

## 6 Conclusion and future work

In this paper, we presented an algorithm for the enhanced computation of a wide class of Kriging-based infill sampling criteria, relying on state-of-the art kriging update formulas. The proposed complexity analysis showed that the use of such update formulas can indeed speed up significantly the computation of criteria of this class, including the Stepwise Uncertainty Reduction (SUR) criteria proposed in Bect et al. (2011), but also the $IECI$ criterion of Gramacy and Lee (2010), the two-step EI criterion of Osborne et al. (2009) and Ginsbourger and Le Riche (2010), and more. In addition, we presented further analytical results for SUR criteria, allowing significant additionnal computational savings. Those results also enabled a fast evaluation of the $J_n$ criterion without conditional simulations, which was considered so far as out of reach.

As confirmed by the very good experimental results obtained on a real-world two-dimensional nuclear safety example, the proposed algorithms and global methodology constitute a very sensible option for the identification of excursion sets and the estimation of their volume, especially when dealing with highly non-linear phenomena studied under limited time budget. However, the efficiency of the method needs to be relativized to the dimensionality of the problem considered, and the ability of the Kriging metamodel (including its trend and covariance kernel specifications) to capture correctly the objective function within the evaluation budget limitations. Indeed, whatever the degree of sophistication of the proposed criteria and their efficient computation, SUR strategies are essentially "model believer", meaning that the criteria are calculated under the hypothesis that the unknown function is one realization of the Gaussian process underlying the Kriging model. Bayesian approaches may mitigate but not solve this issue (the model uncertainty being then partially transferred to the prior distribution of the covariance parameters), at the price of an additional complexity and further computational costs. A mild option could be to perform some kind of pessimistic Bayesian analysis on the final model, in order to support with more confidence conclusions regarding the probability of excursion and potential undiscovered regions of the excursion set.

Future work includes the delivery of an updated version of the KrigInv R package, with a full open source implementation of the presented algorithms. The latter package might then be used to investigate the applicability and the limits of SUR strategies on test-cases in higher dimensions. Concerning the exploitation of such algorithms in industrial frameworks, an effort for providing parallelizable variants of SUR is needed. One of our perspectives of future work is to extend the proposed speed-ups and simplifications to parallel SUR criteria, currently in development. Finally, from a more theoretical perspective, approches directly based on random set notions (considering a "variance" of the excursion set itself, rather than the variance of the excursion volume) may provide elegant alternative sampling criteria for inversion and related problems.

## A Proof of numerical results

*Proof of equation (22)*

$$\begin{aligned}
K_{a,b} :=& \int_{\mathbb{R}} \varphi(a+b.u)e^{-\frac{u^2}{2}}.du \\
=& \int_{\mathbb{R}} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left((a+b.u)^2+u^2\right)\right).du \\
=& \exp\left(-\frac{1}{2}a^2.\left(1-\frac{b^2}{1+b^2}\right)\right)\int_{\mathbb{R}} \varphi\left(\frac{a.b}{\sqrt{1+b^2}}+u.\sqrt{1+b^2}\right).du \\
=& \frac{1}{\sqrt{1+b^2}}.\exp\left(-\frac{1}{2}a^2.\left(1-\frac{b^2}{1+b^2}\right)\right)
\end{aligned}$$

*Proof of equation (23)*

$$\begin{aligned}
L_{a,b} :=& \int_{\mathbb{R}} \Phi(a+b.u)e^{-\frac{u^2}{2}}.du \\
=& \sqrt{2\pi}\int_{\mathbb{R}} P(N<a+b.u).\frac{e^{-\frac{u^2}{2}}}{\sqrt{2\pi}}.du \\
=& \sqrt{2\pi}.P(N<a+b.U)
\end{aligned}$$

where $N$ and $U$ are two independent random variables with standard univariate Gaussian distribution. Now, using the fact that a sum of random variable with Gaussian distribution has still a Gaussian distribution:

$$P(N<a+b.U) = P(N-b.U<a) = \Phi\left(\frac{a}{\sqrt{1+b^2}}\right)$$

*Proof of equation (24)* We use an integration by part to obtain:

$$\begin{aligned}
M_{a,b} :=& \int_{\mathbb{R}} u.\Phi(a+b.u)e^{-\frac{u^2}{2}}.du \\
=& -1.\left[\Phi(a+b.u).e^{-\frac{u^2}{2}}\right]_{-\infty}^{\infty} + b.\int_{\mathbb{R}} \varphi(a+b.u)e^{-\frac{u^2}{2}}.du \\
=& b.K_{a,b}
\end{aligned}$$

*Proof of equation (25)*

$$\begin{aligned}
N_{a,b,c,d} :=& \int_{\mathbb{R}} \Phi(a+b.u)\Phi(c+d.u)e^{-\frac{u^2}{2}}.du \\
=& \sqrt{2\pi}\int_{\mathbb{R}} P(N_1<a+b.u, N_2<c+d.u).\frac{e^{-\frac{u^2}{2}}}{\sqrt{2\pi}}.du \\
=& \sqrt{2\pi}P(N_1<a+b.U, N_2<c+d.U) \\
=& \sqrt{2\pi}P(N_1-b.U<a, N_2-d.U<c)
\end{aligned}$$

where $N_1$, $N_2$ and $U$ are three independent random variables with standard univariate Gaussian distribution. The variance of $N_1 - b.U$ is equal to $1+b^2$ and the covariance between $N_1 - b.U$ and $N_2 - d.U$ is equal to $b.d$. Thus:

$$\begin{aligned}
P(N_1-b.U<a, N_2-d.U<c) &= P\left(\frac{N_1-b.U}{\sqrt{1+b^2}}<\frac{a}{\sqrt{1+b^2}}, \frac{N_2-d.U}{\sqrt{1+d^2}}<\frac{c}{\sqrt{1+d^2}}\right) \\
&= \Phi_2\left(\frac{a}{\sqrt{1+b^2}}, \frac{c}{\sqrt{1+d^2}}, \frac{b.d}{\sqrt{1+b^2}\sqrt{1+d^2}}\right)
\end{aligned}$$

where $\Phi_2(.,.,\rho)$ is the c.d.f. of a Gaussian vector which marginals are standard Gaussian variables with a correlation $\rho$.

## B Simplified expressions for three criteria

*Proof of equation (27)* The proof to obtain the new expression for $J_n^{(\Gamma)}$ uses equation (25).

$$\begin{aligned} J_n^{(\Gamma)}(x_{n+1}) &= \int_{\mathbb{X}} \mathbb{E}_n\left(p_{n+1}(x)(1-p_{n+1}(x))\mathsf{P}_{\mathbb{X}}(dx)\right) \\ &= \int_{\mathbb{X}} \mathbb{E}_n\left(\Phi\left(\frac{m_{n+1}(x)-T}{s_{n+1}(x)}\right)\Phi\left(\frac{T-m_{n+1}(x)}{s_{n+1}(x)}\right)\right)\mathsf{P}_{\mathbb{X}}(dx) \end{aligned}$$

Now we use equation (20) to write:

$$m_{n+1}(x) = m_n(x) + U\,\nu_{n+1}(x),$$

where $U \sim N(0,1)$. Thus, denoting:

$$\begin{aligned} a(x) &:= (m_n(x)-T)/s_{n+1}(x), \\ b(x) &:= \nu_{n+1}(x)/s_{n+1}(x), \end{aligned}$$

we can write:

$$\begin{aligned} J_n^{(\Gamma)}(x_{n+1}) =& \int_{\mathbb{X}}\left(\int_{\mathbb{R}} \Phi(a(x)+b(x).u)\Phi(-a(x)-b(x).u)\frac{e^{-u^2/2}}{\sqrt{2\pi}}.du\right)\mathsf{P}_{\mathbb{X}}(dx) \\ =& \frac{1}{\sqrt{2\pi}}\int_{\mathbb{X}} N_{a(x),b(x),-a(x),-b(x)}\mathsf{P}_{\mathbb{X}}(dx). \end{aligned}$$

We then use our expression of $N_{a,b,c,d}$ given in equation (25) to conclude.

*Proof of equation (26)* We will first work on the expression of $J_n$ and will then use equation (25) again.

$$\begin{aligned} J_n(x_{n+1}) &= \mathbb{E}_n(Var_{n+1}(\alpha)) \\ &= \mathbb{E}_n\left(\mathbb{E}_{n+1}\left(\int_{\mathbb{X}}(\ _{\{f(x)>T\}} - p_{n+1}(x))\mathsf{P}_{\mathbb{X}}(dx)\right)^2\right) \\ &= \mathbb{E}_n\left(\mathbb{E}_{n+1}\iint_{\mathbb{X}\times\mathbb{X}}(\ _{\{f(x)>T\}} - p_{n+1}(x))(\ _{\{f(y)>T\}} - p_{n+1}(y))\mathsf{P}_{\mathbb{X}}(dx)\mathsf{P}_{\mathbb{X}}(dy)\right) \\ &= \mathbb{E}_n\left(\iint_{\mathbb{X}\times\mathbb{X}}\left(\mathbb{E}_{n+1}(\ _{\{f(x)>T\}}\ _{\{f(y)>T\}}) - p_{n+1}(x)p_{n+1}(y)\right)\mathsf{P}_{\mathbb{X}}(dx)\mathsf{P}_{\mathbb{X}}(dy)\right) \end{aligned}$$

If we apply the law of total expectation, we see that for any $(x,y) \in \mathbb{X}^2$:

$$\mathbb{E}_n(\mathbb{E}_{n+1}(\ _{\{f(x)>T\}}\ _{\{f(y)>T\}})) = \mathbb{E}_n(\ _{\{f(x)>T\}}\ _{\{f(y)>T\}}) = P(f(x)>T, f(y)>T|\mathcal{A}_n)$$

Thus this quantity does not depend of $x_{n+1}$ and we can write:

$$c_n := \iint_{\mathbb{X}\times\mathbb{X}} P(f(x)>T, f(y)>T|\mathcal{A}_n)\mathsf{P}_{\mathbb{X}}(dx)\mathsf{P}_{\mathbb{X}}(dy)$$

Consequently we have a new expression for $J_n$:

$$J_n(x_{n+1}) = c_n - \iint_{\mathbb{X}\times\mathbb{X}} \mathbb{E}_n(p_{n+1}(x)p_{n+1}(y))\mathsf{P}_{\mathbb{X}}(dx)\mathsf{P}_{\mathbb{X}}(dy)$$

Then we can proceed like in the previous paragraph by writing again the expression of $p_{n+1}$ and expanding $m_{n+1}$ with equation (20) and end up the proof using equation (25).

*Proof of equation (28)* This proof simply uses again the expansion of the expression of $m_{n+1}$ using equation (20). We then use the numerical results (22),(23) and (24). Noting

$$\begin{aligned} v_1(x) &:= m_n(x) - \widehat{f}_{\max}, \\ v_2(x) &:= \nu_{n+1}(x), \\ v_3(x) &:= v_1(x)/s_{n+1}(x), \\ v_4(x) &:= v_2(x)/s_{n+1}(x), \end{aligned}$$

an expansion with equation (20) leads to:

$$\begin{aligned} IECI(x_{n+1}) = &\int_{\mathbb{X}} \left( \int_{\mathbb{R}} ((v_1(x) + v_2(x)u)\Phi(v_3(x) + v_4(x)u) \right. \\ &\left. + \; s_{n+1}(x)\varphi(v_3(x) + v_4(x)u)) \, \frac{e^{-u^2/2}}{\sqrt{2\pi}} du \right) \mathsf{P}_{\mathbb{X}}(dx) \\ = &\frac{1}{\sqrt{2\pi}} \int_{\mathbb{X}} \left( v_1(x) L_{v_3(x),v_4(x)} + \left( \frac{v_2^2(x)}{s_{n+1}(x)} + s_{n+1}(x) \right) K_{v_3(x),v_4(x)} \right) \mathsf{P}_{\mathbb{X}}(dx) \end{aligned}$$

## References


Barnes, R.J., Watson, A.: Efficient updating of kriging estimates and variances. Mathematical geology **24**(1), 129–133 (1992)

Bect, J., Ginsbourger, D., Li, L., Picheny, V., Vazquez, E.: Sequential design of computer experiments for the estimation of a probability of failure. Statistics and Computing pp. 1–21 (2011). DOI: 10.1007/s11222-011-9241-4

Emery, X.: The kriging update equations and their application to the selection of neighboring data. Computational Geosciences **13**(1), 211–219 (2009)

Gao, H., Wang, J., Zhao, P.: The updated kriging variance and optimal sample design. Mathematical Geology **28**, 295–313 (1996)

Genz, A.: Numerical computation of multivariate normal probabilities. Journal of Computational and Graphical Statistics **1**, 141–149 (1992)

Ginsbourger, D., Le Riche, R.: Towards gaussian process-based optimization with finite time horizon. In: mODa 9 — Advances in Model-Oriented Design and Analysis, Contributions to Statistics, pp. 89–96. Springer (2010)

Golub, G.H., Van Loan, C.F.: Matrix computations. Third Edition, vol. 3. Johns Hopkins University Press (1996)

Gramacy, R.B., Lee, H.K.H.: Optimization under unknown constraints. In: Bayesian Statistics 9 – Proceedings of the Ninth Valencia International Meeting, pp. 229–256 (2010)

Jones, D.R., Schonlau, M., William, J.: Efficient global optimization of expensive black-box functions. Journal of Global Optimization **13**(4), 455–492 (1998)

Osborne, M.A., Garnett, R., Roberts, S.J.: Gaussian processes for global optimization. In: 3rd International Conference on Learning and Intelligent Optimization (LION3). Trento, Italy (2009)

Roustant, O., Ginsbourger, D., Deville, Y.: Dicekriging, diceoptim: Two r packages for the analysis of computer experiments by kriging-based metamodelling and optimization (2011). In revision for Journal of Statistical Software

Vazquez, E., Bect, J.: A sequential Bayesian algorithm to estimate a probability of failure. In: Proceedings of the 15th IFAC Symposium on System Identification, SYSID 2009 15th IFAC Symposium on System Identification, SYSID 2009. Saint-Malo France (2009)

Vazquez, E., Piera-Martinez, M.: Estimation du volume des ensembles d'excursion d'un processus gaussien par krigeage intrinsèque. In: 39ème Journées de Statistiques Conférence Journée de Statistiques. Angers France (2007)