Fast Computation of the Multi-points Expected Improvement with Applications in Batch Selection

Clément Chevalier, David Ginsbourger

Fast Computation of the Multi-points Expected Improvement with Applications in Batch Selection

Clément Chevalier[1,2] and David Ginsbourger[2]

[1] Institut de Radioprotection et de Sûreté Nucléaire (IRSN)
31, avenue de la Division Leclerc, 92260 Fontenay-aux-Roses, France
[2] IMSV, University of Bern,
Alpeneggstrasse 22, 3012 Bern, Switzerland
{clement.chevalier,ginsbourger}@stat.unibe.ch

Abstract. The Multi-points Expected Improvement criterion (or q-EI) has recently been studied in batch-sequential Bayesian Optimization. This paper deals with a new way of computing q-EI, without using Monte-Carlo simulations, through a closed-form formula. The latter allows a very fast computation of q-EI for reasonably low values of q (typically, less than 10). New parallel kriging-based optimization strategies, tested on different toy examples, show promising results.

Keywords: Computer Experiments, Kriging, Parallel Optimization, Expected Improvement

1 Introduction

In the last decades, *metamodeling* (or *surrogate modeling*) has been increasingly used for problems involving costly computer codes (or "black-box simulators"). Practitioners typically dispose of a very limited evaluation budget and aim at selecting evaluation points cautiously when attempting to solve a given problem.

In global optimization, the focus is usually put on a real-valued function f with d-dimensional source space. In this settings, [1] proposed the now famous *Efficient Global Optimization* (EGO) algorithm, relying on a kriging metamodel [2] and on the Expected Improvement (EI) criterion [3]. In EGO, the optimization is done by sequentially evaluating f at points maximizing EI. A crucial advantage of this criterion is its fast computation (besides, the analytical gradient of EI is implemented in [4]), so that the hard optimization problem is replaced by series of much simpler ones.

Coming back to the decision-theoretic roots of EI [5], a Multi-points Expected Improvement (also called "q-EI") criterion for batch-sequential optimization was defined in [6] and further developed in [7, 8]. Maximizing this criterion enables choosing batches of $q > 1$ points at which to evaluate f in parallel, and is of particular interest in the frequent case where several CPUs are simultaneously available. Even though an analytical formula was derived for the 2-EI in [7], the

Monte Carlo (MC) approach of [8] for computing q-EI when $q \geq 3$ makes the criterion itself expensive-to-evaluate, and particularly hard to optimize.

A lot of effort has recently been paid to address this problem. The pragmatic approach proposed by [8] consists in circumventing a direct q-EI maximization, and replacing it by simpler strategies where batches are obtained using an offline q-points EGO. In such strategies, the model updates are done using dummy response values such as the kriging mean prediction (Kriging Believer) or a constant (Constant Liar), and the covariance parameters are re-estimated only when real data is assimilated. In [9] and [10], q-EI optimization strategies were proposed relying on the MC approach, where the number of MC samples is tuned online to discriminate between candidate designs. Finally, [11] proposed a q-EI optimization strategy involving stochastic gradient, with the crucial advantage of *not* requiring to evaluate q-EI itself.

In this article we derive a formula allowing a fast and accurate approximate evaluation of q-EI. This formula may contribute to significantly speed up strategies relying on q-EI. The main result, relying on Tallis' formula, is given in Section 2. The usability of the proposed formula is then illustrated in Section 3 through benchmark experiments, where a brute force maximization of q-EI is compared to three variants of the Constant Liar strategy. In particular, a new variant (CL-mix) is introduced, and is shown to offer very good performances at a competitive computational cost. For self-containedness, a slightly revisited proof of Tallis' formula is given in appendix.

2 Multi-points Expected Improvement explicit formulas

In this section we give an explicit formula allowing a fast and accurate deterministic approximation of q-EI. Let us first give a few precisions on the mathematical settings. Along the paper, f is assumed to be one realisation of a Gaussian Process (GP) with known covariance kernel and mean known up to some linear trend coefficients, so that the conditional distribution of a vector of values of the GP conditional on past observations is still Gaussian (an improper uniform prior is put on the trend coefficients when applicable). This being said, most forthcoming derivations boil down to calculations on Gaussian vectors. Let $\mathbf{Y} := (Y_1, \ldots, Y_q)$ be a Gaussian Vector with mean $\mathbf{m} \in \mathbb{R}^q$ and covariance matrix Σ. Our aim in this paper is to explicitly calculate expressions of the following kind:

$$\mathbb{E}\left[\left(\max_{i\in\{1,\ldots,q\}} Y_i - T\right)_+\right] \tag{1}$$

where $(.)_+ := \max(., 0)$. In Bayesian optimization (say maximization), expectations and probabilities are taken conditional on response values at a given set of n points $(\mathbf{x}_1, \ldots, \mathbf{x}_n) \in \mathbb{X}^n$ where $\mathbb{X}$ is the input set of f (often, a compact subset of $\mathbb{R}^d$, $d \geq 1$), the threshold $T \in \mathbb{R}$ is usually the maximum of those n available response values, and $\mathbf{Y}$ is the vector of unknown responses at a given batch of q points, $\mathbf{X}^q := (\mathbf{x}_{n+1}, \ldots, \mathbf{x}_{n+q}) \in \mathbb{X}^q$. In such framework, the vector $\mathbf{m}$ and the

matrix Σ are the so-called "Kriging mean" and "Kriging covariance" at $\mathbf{X}^q$ and can be calculated relying on classical Kriging equations (see, e.g., [12]).

In order to obtain a tractable analytical formula for Expression (1), not requiring any Monte-Carlo simulation, let us first give a useful formula obtained by [13], and recently used in [14] for GP modeling with inequality constraints:

Proposition 1 (Tallis formulas) *Let $\mathbf{Z} := (Z_1, \ldots, Z_q)$ be a Gaussian Vector with mean $\mathbf{m} \in \mathbb{R}^q$ and covariance matrix $\Sigma \in \mathbb{R}^{q \times q}$. Let $\mathbf{b} = (b_1, \ldots, b_q) \in \mathbb{R}^q$. The expectation of any coordinate Z_k under the linear constraint $(\forall j \in \{1, \ldots, q\},\ Z_j \leq b_j)$ denoted by $\mathbf{Z} \leq \mathbf{b}$ can be expanded as follows:*

$$\mathbb{E}(Z_k | \mathbf{Z} \leq \mathbf{b}) = m_k - \frac{1}{p} \sum_{i=1}^{q} \Sigma_{ik}\ \varphi_{m_i, \Sigma_{ii}}(b_i)\ \Phi_{q-1}\left(\mathbf{c}_{.i}, \Sigma_{.i}\right) \tag{2}$$

where:

- $p := \mathbb{P}(\mathbf{Z} \leq \mathbf{b}) = \Phi_q(\mathbf{b} - \mathbf{m}, \Sigma)$
- $\Phi_q(\mathbf{u}, \Sigma)$ $(\mathbf{u} \in \mathbb{R}^q, \Sigma \in \mathbb{R}^{q \times q}, q \geq 1)$ *is the c.d.f. of the centered multivariate Gaussian distribution with covariance matrix Σ.*
- $\varphi_{m,\sigma^2}(.)$ *is the p.d.f. of the univariate Gaussian distribution with mean m and variance σ^2*
- $\mathbf{c}_{.i}$ *is the vector of $\mathbb{R}^{q-1}$ with general term $(b_j - m_j) - (b_i - m_i)\frac{\Sigma_{ij}}{\Sigma_{ii}}$, $j \neq i$*
- $\Sigma_{.i}$ *is a $(q-1) \times (q-1)$ matrix obtained by computing $\Sigma_{uv} - \frac{\Sigma_{iu}\Sigma_{iv}}{\Sigma_{ii}}$ for $u \neq i$ and $v \neq i$. This matrix corresponds to the conditional covariance matrix of the random vector $\mathbf{Z}_{-i} := (Z_1, \ldots, Z_{i-1}, Z_{i+1}, \ldots, Z_q)$ knowing Z_i.*

For the sake of brevity, the proof of this Proposition is sent in the Appendix. A crucial point for the practical use of this result is that there exist very fast procedures to compute the c.d.f. of the multivariate Gaussian distribution. For example, the work of [15], [16] have been used in many R packages (see, e.g., [17], [18]). The Formula (2) above is an important tool to efficiently compute Expression (1) as shown with the following Property:

Proposition 2 *Let $\mathbf{Y} := (Y_1, \ldots, Y_q)$ be a Gaussian Vector with mean $\mathbf{m} \in \mathbb{R}^q$ and covariance matrix Σ. For $k \in \{1, \ldots, q\}$ consider the Gaussian vectors $\mathbf{Z}^{(k)} := (Z_1^{(k)}, \ldots, Z_q^{(k)})$ defined as follows:*

$$\begin{aligned} Z_j^{(k)} &:= Y_j - Y_k\ ,\ j \neq k \\ Z_k^{(k)} &:= -Y_k \end{aligned}$$

Denoting by $\mathbf{m}^{(k)}$ and $\Sigma^{(k)}$ the mean and covariance matrix of $\mathbf{Z}^{(k)}$, and defining the vector $\mathbf{b}^{(k)} \in \mathbb{R}^q$ by $b_k^{(k)} = -T$ and $b_j^{(k)} = 0$ if $j \neq k$, the EI of $\mathbf{X}^q$ writes:

$$\boxed{EI(\mathbf{X}^q) = \sum_{k=1}^{q} \left((m_k - T)p_k + \sum_{i=1}^{q} \Sigma_{ik}^{(k)} \varphi_{m_i^{(k)}, \Sigma_{ii}^{(k)}}(b_i^{(k)}) \Phi_{q-1}\left(\mathbf{c}_{.i}^{(k)}, \Sigma_{.i}^{(k)}\right) \right)} \tag{3}$$

where:

- $p_k := \mathbb{P}(\mathbf{Z}^{(k)} \leq \mathbf{b}^{(k)}) = \Phi_q(\mathbf{b}^{(k)} - \mathbf{m}^{(k)}, \Sigma^{(k)})$.
  p_k *is actually the probability that* Y_k *exceeds* T *and* $Y_k = \max_{j=1,\dots,q} Y_j$.
- $\Phi_q(.,\Sigma)$ *and* $\varphi_{m,\sigma^2}(.)$ *are defined in Proposition 1*
- $\mathbf{c}_{.i}^{(k)}$ *is the vector of* $\mathbb{R}^{q-1}$ *constructed like in Proposition 1, by computing* $(b_j^{(k)} - m_j^{(k)}) - (b_i^{(k)} - m_i^{(k)})\frac{\Sigma_{ij}^{(k)}}{\Sigma_{ii}^{(k)}}$, *with* $j \neq i$
- $\Sigma_{.i}^{(k)}$ *is a* $(q-1)\times(q-1)$ *matrix constructed from* $\Sigma^{(k)}$ *like in Proposition 1. It corresponds to the conditional covariance matrix of the random vector* $\mathbf{Z}_{-i}^{(k)} := (Z_1^{(k)}, \dots, Z_{i-1}^{(k)}, Z_{i+1}^{(k)}, \dots, Z_q^{(k)})$ *knowing* $Z_i^{(k)}$.

Proof. Using that $\mathbb{1}_{\{\max_{i\in\{1,\dots,q\}} Y_i \geq T\}} = \sum_{k=1}^{q} \mathbb{1}_{\{Y_k \geq T,\ Y_j \leq Y_k\ \forall j\neq k\}}$, we get

$$\begin{aligned}
EI(\mathbf{X}^q) &= \mathbb{E}\left[\left(\max_{i\in\{1,\dots,q\}} Y_i - T\right)\sum_{k=1}^{q}\mathbb{1}_{\{Y_k\geq T,\ Y_j\leq Y_k\ \forall j\neq k\}}\right] \\
&= \sum_{k=1}^{q}\mathbb{E}\left((Y_k - T)\mathbb{1}_{\{Y_k\geq T,\ Y_j\leq Y_k\ \forall j\neq k\}}\right) \\
&= \sum_{k=1}^{q}\mathbb{E}\left(Y_k - T \middle| Y_k \geq T,\ Y_j \leq Y_k\ \forall j\neq k\right)\mathbb{P}\left(Y_k\geq T, Y_j\leq Y_k\ \forall j\neq k\right) \\
&= \sum_{k=1}^{q}\left(-T - \mathbb{E}\left(Z_k^{(k)}\middle|\mathbf{Z}^{(k)}\leq\mathbf{b}^{(k)}\right)\right)\mathbb{P}\left(\mathbf{Z}^{(k)}\leq\mathbf{b}^{(k)}\right)
\end{aligned}$$

Now the computation of $p_k := \mathbb{P}\left(\mathbf{Z}^{(k)} \leq \mathbf{b}^{(k)}\right)$ simply requires one call to the Φ_q function and the proof can be completed by applying Tallis formula (2) to the random vectors $\mathbf{Z}^{(k)}$ ($1 \leq k \leq q$).

Remark 1. From Properties (1) and (2), it appears that computing q-EI requires a total of q calls to Φ_q and q^2 calls to Φ_{q-1}. The proposed approach performs thus well when q is moderate (typically lower than 10). For higher values of q, estimating q-EI by Monte-Carlo might remain competitive.

Remark 2. In the particular case $q = 1$ and with the convention $\Phi_0(.,\Sigma) = 1$, Equation (3) corresponds to the classical EI formula proven in [5, 1].

3 Batch sequential optimization using Multi-points EI

Let us first illustrate Proposition 2 and show that the proposed q-EI calculation based on Tallis' formula is actually consistent with a Monte Carlo estimation. From a kriging model based on 12 observations of the Branin-Hoo function [1], we generated a 4-point batch (Figure 1, left plot) and calculated its q-EI value (middle plot, dotted line). The MC estimates converge to a value close to the latter, and the relative error after $5 * 10^9$ runs is less than 10^{-5}. 4-point batches generated from the three strategies detailed below are drawn on the right plot.

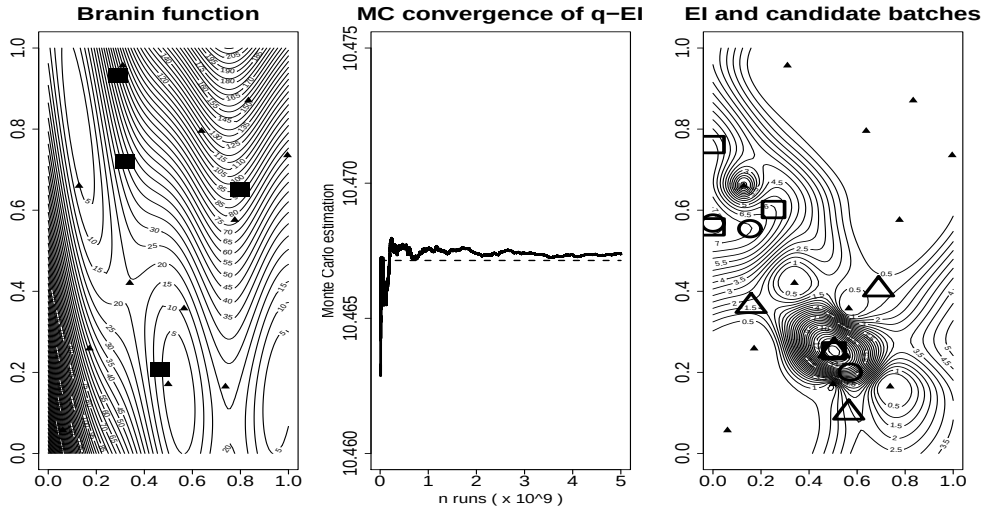

Fig. 1. Convergence (middle) of MC estimates to the q-EI value calculated with Proposition 2 in the case of a batch of four points (shown on the left plot). Right: candidate batches obtained by q-EI stepwise maximisation (squares), and the CL-min (circles) and CL-max (triangles) strategies.

We now compare a few kriging-based batch-sequential optimization methods on two different functions: the function $x \mapsto -\log(-\text{Hartman6}(x))$ (see, e.g., [1]), defined on $[0,1]^6$ and the Rastrigin function ([19, 20]) in dimension two restricted to the domain $[0, 2.5]^2$. The first function in dimension 6 is unimodal, while the second one has a lot of local optima (see: Figure 2). The Rastrigin function is one of the 24 noiseless test function of the Black-Box Optimization Benchmark (BBOB) [19].

For each runs, we start with a random initial Latin hypercube design (LHS) of $n_0 = 10$ (Rastrigin) or 50 (Hartman6) points and estimate the covariance parameters by Maximum Likelihood (here a Matérn kernel with $\nu = 3/2$ is chosen). For both functions and all strategies, batches of $q = 6$ points are added at each iteration, and the covariance parameters are re-estimated after each batch assimilation. Since the tests are done for several designs of experiments, we chose to represent, along the runs, the relative mean squared error:

$$\text{rMSE} = \frac{1}{M} \sum_{i=1}^{\text{M}} \left(\frac{y_{\min}^{(i)} - y_{\text{opt}}}{y_{\text{opt}}} \right)^2 \tag{4}$$

where $y_{\min}^{(i)}$ in the current observed minimum in run number i and y_{opt} is the real unknown optimum. The total number M of different initial designs of experiments is fixed to 50. The tested strategies are:

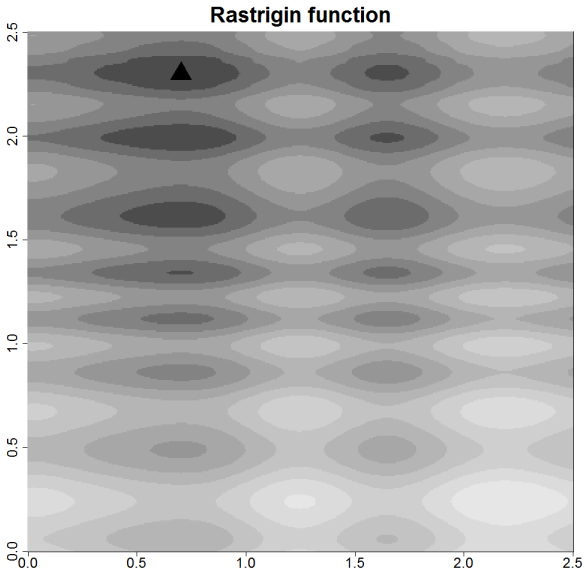

Fig. 2. Contour lines of the Rastrigin function (grayscale) and location of the global optimizer (black triangle)

- (1) q-EI stepwise maximization: q sequential d-dimensional optimizations are performed. We start with the maximization of the 1-point EI and add this point to the new batch. We then maximize the 2-point EI (keeping the first point obtained as first argument), add the maximizer to the batch, and iterate until q points are selected.
- (2) Constant Liar min (CL-min): We start with the maximization of the 1-point EI and add this point to the new batch. We then assume a dummy response (a"lie") at this point, and update the Kriging metamodel with this point and the lie. We then maximize the 1-point EI obtained with the updated kriging metamodel, get a second point, and iterate the same process until a batch of q points is selected. The dummy response has the same value over the $q-1$ lies, and is here fixed to the minimum of the current observations.
- (3) Constant Liar max (CL-max): The lie in this Constant Liar strategy is fixed to the maximum of the current observations.
- (4) Constant Liar mix (CL-mix): At each iteration, two batches are generated with the CL-min and CL-max strategies. From these two "candidate" batches, we choose the batch with the best actual q-EI value, calculated based on Proposition 2.
- (5) Random sampling.

Note that CL-min tends to explore the function near the current minimizer (as the lie is a low value and we are minimizing f) while CL-max is more exploratory. Thus, CL-min is expected to perform well on unimodal functions. On the contrary, CL-max may perform better on multimodal functions. For all the tests we use the DiceKriging and DiceOptim packages [4]. The optimizations of the different criteria rely on a genetic algorithm using derivatives, available in the rgenoud package [21]. Figure 3 represents the compared performances of these strategies.

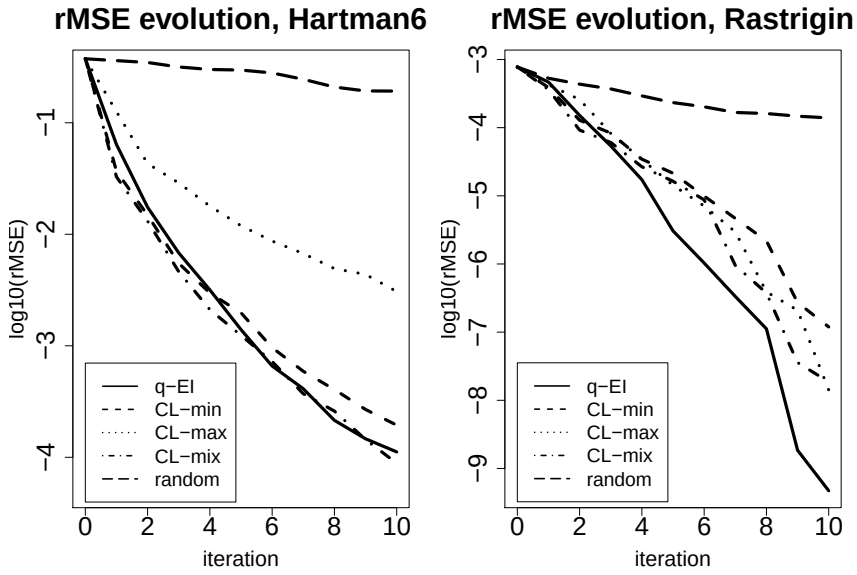

Fig. 3. Compared performances of the five considered batch-sequential optimization strategies, on two test functions.

From these plots we draw the following conclusions: first, the q-EI stepwise maximization strategy outperforms the strategies based on constant lies, CL-min and CL-max. However, the left graph of Figure 3 points out that the CL-min strategy seems particularly well-adapted to the Hartman6 function. Since running a CL is computationally much cheaper than a brute fore optimization of q-EI, it is tempting to recommend the CL-min strategy for Hartman6. However, it is not straightforward to know in advance which of CL-min or CL-max will perform better on a given test case. Indeed, for example, CL-max outperforms CL-min on the Rastrigin function.

Now, we observe that using q-EI in the CL-mix heuristic enables very good performances in both cases without having to select one of the two lie values in advance. For the Hartman6 function, CL-mix even outperforms both CL-

min and CL-max and has roughly the same performance as a brute force q-EI maximization. This suggests that a good heuristic might be to generate, at each iteration, candidate batches obtained with different strategies (e.g. CL with different lies) and to discriminate those batches using q-EI.

Conclusion

In this article we give a closed-form expression enabling a fast computation of the Multi-points Expected Improvement criterion for batch sequential Bayesian global optimization. This formula is consistent with the classical Expected Improvement formula and its computation does not require Monte Carlo simulations. Optimization strategies based on this criterion are now ready to be used on real test cases, and a brute maximization of this criterion shows promising results. In addition, we show that good performances can be achieved by using a cheap-to-compute criterion and by discriminating the candidate batches generated by such criterion with the q-EI. Such heuristics might be particularly interesting when the time needed to generate batches becomes a computational bottleneck, e.g. when $q \geq 10$ and calls to the Gaussian c.d.f. become expensive.

A perspective, currently under study, is to improve the maximization of q-EI itself, e.g. through a more adapted choice of the algorithm and/or an analytical calculation of q-EI's gradient.

Acknowledgments This work has been conducted within the frame of the ReDice Consortium, gathering industrial (CEA, EDF, IFPEN, IRSN, Renault) and academic (Ecole des Mines de Saint-Etienne, INRIA, and the University of Bern) partners around advanced methods for Computer Experiments. Clément Chevalier gratefully acknowledges support from the French Nuclear Safety Institute (IRSN). The authors also would like to thank Dr. Sébastien Da Veiga for raising our attention to Tallis' formula.

References

1. Jones, D.R., Schonlau, M., William, J.: Efficient global optimization of expensive black-box functions. Journal of Global Optimization **13**(4) (1998) 455–492
2. Santner, T.J., Williams, B.J., Notz, W.: The Design and Analysis of Computer Experiments. Springer Verlag (2003)
3. Mockus, J.: Bayesian Approach to Global Optimization. Theory and Applications. Kluwer Academic Publisher, Dordrecht (1989)
4. Roustant, O., Ginsbourger, D., Deville, Y.: DiceKriging, DiceOptim: Two R packages for the analysis of computer experiments by Kriging-Based Metamodelling and Optimization. Journal of Statistical Software **51 (1)** (2012) 1–55
5. Mockus, J., Tiesis, V., Zilinskas, A.: The application of Bayesian methods for seeking the extremum. In Dixon, L., Szego, E.G., eds.: Towards Global Optimization. Volume 2. Elsevier (1978) 117–129

6. Schonlau, M.: Computer Experiments and global optimization. PhD thesis, University of Waterloo (1997)
7. Ginsbourger, D.: Métamodèles multiples pour l'approximation et l'optimisation de fonctions numériques multivariables. PhD thesis, Ecole nationale supérieure des Mines de Saint-Etienne (2009)
8. Ginsbourger, D., Le Riche, R., L., C.: Kriging is well-suited to parallelize optimization. In: Computational Intelligence in Expensive Optimization Problems. Volume 2 of Adaptation Learning and Optimization. Springer (2010) 131–162
9. Janusevskis, J., Le Riche, R., Ginsbourger, D.: Parallel expected improvements for global optimization: summary, bounds and speed-up. (August 2011)
10. Janusevskis, J., Le Riche, R., Ginsbourger, D., Girdziusas, R.: Expected improvements for the asynchronous parallel global optimization of expensive functions : Potentials and challenges. In: LION 6 Conference (Learning and Intelligent OptimizatioN), Paris : France. (2012)
11. Frazier, P.I.: Parallel global optimization using an improved multi-points expected improvement criterion. In: INFORMS Optimization Society Conference, Miami FL. (2012)
12. Chilès, J.P., Delfiner, P.: Geostatistics: Modeling Spatial Uncertainty. Wiley, New York (1999)
13. Tallis, G.: The moment generating function of the truncated multi-normal distribution. J. Roy. Statist. Soc. Ser. B **23**(1) (1961) 223–229
14. Da Veiga, S., Marrel, A.: Gaussian process modeling with inequality constraints. Annales de la Faculté des Sciences de Toulouse **21 (3)** (2012) 529–555
15. Genz, A.: Numerical computation of multivariate normal probabilities. Journal of Computational and Graphical Statistics **1** (1992) 141–149
16. Genz, A., Bretz, F.: Computation of Multivariate Normal and t Probabilities. Springer-Verlag (2009)
17. Genz, A., Bretz, F., Miwa, T., Mi, X., Leisch, F., Scheipl, F., Bornkamp, B., Hothorn, T.: mvtnorm: Multivariate Normal and t Distributions. (2012) R package version 0.9-9992.
18. Azzalini, A.: mnormt: The multivariate normal and t distributions. (2012) R package version 1.4-5.
19. Finck, S., Hansen, N., Ros, R., Auger, A.: Real-parameter black-box optimization bencharking 2009: Presentation of the noiseless functions. Technical report, Research Center PPE, 2009 (2010)
20. Hansen, N., Finck, S., Ros, R., Auger, A.: Real-parameter black-box optimization bencharking 2009: Noiseless functions definitions. Technical report, INRIA 2009 (2010)
21. Mebane, W., Sekhon, J.: Genetic optimization using derivatives: The rgenoud package for r. Journal of Statistical Software **Vol. 42, Issue 11** (2011) 1–26
22. Cressie, N., Davis, A., Leroy Folks, J.: The moment-generating function and negative integer moments. The American Statistician **35 (3)** (1981) 148–150

Appendix: proof for Tallis formula (2)

The proof proposed here follows exactly the method given in [13] in the particular case of a centered Gaussian Vector with normalized covariance matrix (i.e. a covariance matrix equal to the *correlation* matrix). Here, the proof is slightly more detailed and applies in a more general case.

Let $\mathbf{Z} := (Z_1, \ldots, Z_q) \sim \mathcal{N}(\mathbf{m}, \Sigma)$ with $\mathbf{m} \in \mathbb{R}^q$ and $\Sigma \in \mathbb{R}^{q \times q}$. Let $\mathbf{b} = (b_1, \ldots, b_q) \in \mathbb{R}^q$. Our goal is to calculate: $\mathbb{E}(Z_k | \mathbf{Z} \leq \mathbf{b})$. The method proposed by Tallis consists in calculating the conditional joint moment generating function (MGF) of $\mathbf{Z}$ defined as follows:

$$M_{\mathbf{Z}}(\mathbf{t}) := \mathbb{E}(\exp(\mathbf{t}^\top \mathbf{Z}) | \mathbf{Z} \leq \mathbf{b}) \tag{5}$$

It is known (see, e.g., [22]) that the conditional expectation of Z_k can be obtained by deriving such MGF with respect to t_k, in $\mathbf{t} = \mathbf{0}$. Mathematically this writes:

$$\mathbb{E}(Z_k | \mathbf{Z} \leq \mathbf{b}) = \left. \frac{\partial M_{\mathbf{Z}}(\mathbf{t})}{\partial t_k} \right|_{\mathbf{t}=\mathbf{0}} \tag{6}$$

The main steps of this proof are then to calculate such MGF and its derivative with respect to any coordinate t_k.

Let us consider the **centered** random variable $\mathbf{Z}^c := \mathbf{Z} - \mathbf{m}$. Denoting $\mathbf{h} = \mathbf{b} - \mathbf{m}$, conditioning on $\mathbf{Z} \leq \mathbf{b}$ or on $\mathbf{Z}^c \leq \mathbf{h}$ are equivalent. The MGF of $\mathbf{Z}^c$ can be calculated as follows:

$$\begin{aligned} M_{\mathbf{Z}^c}(\mathbf{t}) :=& \mathbb{E}(\exp(\mathbf{t}^\top \mathbf{Z}^c) | \mathbf{Z}^c \leq \mathbf{h}) \\ =& \frac{1}{p} \int_{-\infty}^{h_1} \ldots \int_{-\infty}^{h_q} \exp(\mathbf{t}^\top \mathbf{u}) \varphi_{\mathbf{0},\Sigma}(\mathbf{u}) \mathrm{d}\mathbf{u} \\ =& \frac{1}{p} (2\pi)^{-\frac{q}{2}} |\Sigma|^{-\frac{1}{2}} \int_{-\infty}^{h_1} \ldots \int_{-\infty}^{h_q} \exp\left(-\frac{1}{2} \left(\mathbf{u}^\top \Sigma^{-1} \mathbf{u} - 2\mathbf{t}^\top \mathbf{u} \right) \right) \mathrm{d}\mathbf{u} \end{aligned}$$

where $p := \mathbb{P}(\mathbf{Z} \leq \mathbf{b})$ and $\varphi_{\mathbf{v},\Sigma}(.)$ denotes the p.d.f. of the multivariate normal distribution with mean $\mathbf{v}$ and covariance matrix Σ. The calculation can be continued by noting that:

$$\begin{aligned} M_{\mathbf{Z}^c}(\mathbf{t}) =& \frac{1}{p} (2\pi)^{-\frac{q}{2}} |\Sigma|^{-\frac{1}{2}} \exp\left(\frac{1}{2} \mathbf{t}^\top \Sigma \mathbf{t} \right) \int_{-\infty}^{h_1} \ldots \int_{-\infty}^{h_q} \exp\left(-\frac{1}{2} (\mathbf{u} - \Sigma \mathbf{t})^\top \Sigma^{-1} (\mathbf{u} - \Sigma \mathbf{t}) \right) \mathrm{d}\mathbf{u} \\ =& \frac{1}{p} \exp\left(\frac{1}{2} \mathbf{t}^\top \Sigma \mathbf{t} \right) \Phi_q(\mathbf{h} - \Sigma \mathbf{t}, \Sigma) \end{aligned}$$

where $\Phi_q(., \Sigma)$ is the c.d.f. of the centered multivariate normal distribution with covariance matrix Σ.

Now, let us calculate for some $k \in \{1, \ldots, q\}$ the partial derivative $\frac{\partial M_{\mathbf{Z}^c}(\mathbf{t})}{\partial t_k}$ in $\mathbf{t} = \mathbf{0}$, which is equal by definition to $\mathbb{E}(Z_k^c | \mathbf{Z}^c \leq \mathbf{h})$.

$$\begin{aligned} p\ \mathbb{E}(Z_k^c | \mathbf{Z}^c \leq \mathbf{h}) &= p\ \left. \frac{\partial M_{\mathbf{Z}^c}(\mathbf{t})}{\partial t_k} \right|_{\mathbf{t}=\mathbf{0}} \\ &= 0 + 1. \frac{\partial}{\partial t_k} \left(\Phi_q \left(\mathbf{h} - t_k \begin{pmatrix} \Sigma_{1k} \\ \vdots \\ \Sigma_{qk} \end{pmatrix}, \Sigma \right) \right) \Bigg|_{t_k=0} \\ &= -\sum_{i=1}^{q} \Sigma_{ik} \int_{-\infty}^{h_1} \ldots \int_{-\infty}^{h_{i-1}} \int_{-\infty}^{h_{i+1}} \ldots \int_{-\infty}^{h_q} \varphi_{\mathbf{0},\Sigma}(\mathbf{u}_{-i}, u_i = h_i) \mathrm{d}\mathbf{u}_{-\mathrm{i}} \end{aligned}$$

The last step is obtained applying the chain rule to $\mathbf{x} \mapsto \Phi_q(\mathbf{x}, \Sigma)$ at the point $\mathbf{x} = \mathbf{h}$. Here, $\varphi_{\mathbf{0},\Sigma}(\mathbf{u}_{-i}, u_i = h_i)$ denotes the c.d.f. of the centered multivariate normal

distribution at given points $(\mathbf{u}_{-i}, u_i = h_i) := (u_1, \ldots, u_{i-1}, h_i, u_{i+1}, \ldots, u_q)$. Note that the integrals in the latter Expression are in dimension $q-1$ and not q. In the i^{th} term of the sum above, we integrate with respect to all the q components except the component i. To continue the calculation we can use the identity:

$$\forall \mathbf{u} \in \mathbb{R}^q, \varphi_{\mathbf{0},\Sigma}(\mathbf{u}) = \varphi_{0,\Sigma_{ii}}(u_i)\varphi_{\Sigma_{ii}^{-1}\Sigma_i u_i, \Sigma_{-i,-i}-\Sigma_i\Sigma_{ii}^{-1}\Sigma_i^\top}(\mathbf{u}_{-i}) \tag{7}$$

where $\Sigma_i = (\Sigma_{1i}, \ldots, \Sigma_{i-1i}, \Sigma_{i+1i}, \ldots, \Sigma_{qi})^\top$ $(\Sigma_i \in \mathbb{R}^{q-1})$ and $\Sigma_{-i,-i}$ is the $(q-1) \times (q-1)$ matrix obtained by removing the line and column i from Σ. This identity can be proven using Bayes formula and Gaussian vectors conditioning formulas. Its use gives:

$$\begin{aligned} p\ \mathbb{E}(Z_k^c|\mathbf{Z}^c \leq \mathbf{h}) &= -\sum_{i=1}^{q} \Sigma_{ik}\varphi_{0,\Sigma_{ii}}(h_i)\Phi_{q-1}(\mathbf{h}_{-i} - \Sigma_{ii}^{-1}\Sigma_i h_i, \Sigma_{-i,-i} - \Sigma_i\Sigma_{ii}^{-1}\Sigma_i^\top) \\ &= -\sum_{i=1}^{q} \Sigma_{ik}\varphi_{m_i,\Sigma_{ii}}(b_i)\Phi_{q-1}(\mathbf{h}_{-i} - \Sigma_{ii}^{-1}\Sigma_i h_i, \Sigma_{-i,-i} - \Sigma_i\Sigma_{ii}^{-1}\Sigma_i^\top) \end{aligned}$$

which finally delivers the Tallis formula, see Equation (2).