Parallel expected improvements for global optimization: summary, bounds and speed-up

Technical report

OMD2 deliverable Nmbr. 2.1.1-B

Janis Janusevskis · Rodolphe Le Riche · David Ginsbourger

Abstract The sequential sampling strategies based on Gaussian processes are widely used for optimization of time consuming simulators. In practice, such computationally demanding problems are solved by increasing number of processing units. This has therefore induced extensions of sampling criteria which consider the framework of parallel calculation.

This report further studies expected improvement criteria for parallel and asynchronous computations. A unified parallel asynchronous expected improvement criterion is formulated. Bounds and strategies for comparing criteria values at various design points are discussed. Finally, the impact of the number of available computing units on the performance is empirically investigated.

Keywords Kriging based optimization · Gaussian process · Expected improvement · Optimization using computer grids · Distributed calculations · Monte Carlo optimization

1 Introduction

Many design problems are nowadays based on computationally costly simulator codes. The current technology for adressing computationally intensive tasks is based on an increasing number of processing units (processors, cores, CPUs, GPUs). For example,

Janis Janusevskis
LSTI / CROCUS team
Ecole Nationale Supérieure des Mines de Saint-Etienne
Saint-Etienne, France
E-mail: janisj@latnet.lv

Rodolphe Le Riche
CNRS UMR 5146 and LSTI / CROCUS team
Ecole Nationale Supérieure des Mines de Saint-Etienne
Saint-Etienne, France
E-mail: leriche@emse.fr

David Ginsbourger
University of Bern
Bern, Switzerland
E-mail: david.ginbourger@stat.unibe.ch

the next generation of exascale computers (10^{18} floating point operations per second) should have of the order of 10^9 computing nodes [13]. Distributing calculations on several nodes is particularly relevant in optimization where a natural parallelization level exists, that of calculating concurrent designs in parallel. However, most current optimization methods have been thought sequentially. Today, there is a need for optimization strategies that operate within the framework of heterogeneous parallel computation, i.e. algorithms designed for being used on computing grids.

The Efficient Global Optimization (EGO) algorithm [8] has become a popular choice for optimizing computationally expensive simulation models. EGO is based on kriging (Gaussian process regression) metamodels, that allow to formulate the Expected improvement (EI) criterion, which provides a compromise between global exploration of the domain space and local exploitation of the best known solutions. At each optimization step the EI is maximized to obtain the next simulation point. By definition the EI is a sequential one point strategy [5] and lacks means for efficiently accounting for parallel processing capabilities.

The EI criterion has been adapted for parallel computation in [6],[12] leading to the so called "q steps EI" (qEI), that provides the synchronous selection of q new points for the next simulation run. This qEI criterion assumes that the computation time is constant for all design points, i.e. once a set of points is sent for simulation the results are available at the same time for all the points in the set. In general this assumption may not be correct. The simulations at different points may have different algorithmic complexities or run on nodes with different performances or loads, therefore the simulations will finish successively, i.e. new computing nodes become available asynchronously.

The selection of points for the next calculation should be made as soon as computational resources are available, i.e. asynchronously. In [4] the qEI criterion has been extended and the so called "Expected Expected Improvement" (EEI) criterion has been proposed. In the framework of Gaussian processes one assumes that the simulator response at *busy* points (points that have been sent to simulator but have not returned yet) are conditional Gaussian random variables. Maximizing expectation of the EI with respect to busy points allows selection of a new set of points accounting for the points that are being simulated.

As already noted in [6] and [4], the implementation of these criteria faces several problems. Firstly, in general qEI and EEI can not be obtained analytically. They should instead be estimated using numerical procedures such as Monte Carlo (MC). Secondly, selecting the next point in EGO is a global maximization of the EI problem which has the same dimension as the design variables. In contrast, maximizing qEI and EEI has for dimension the number of the design variables multiplied by the number of new points (the number of free computing nodes). Therefore brute optimization of qEI and EEI using simple MC may not be cost effective. These problems have been partially addressed in [6], [4], [1] by introducing heuristics.

This report further studies expected improvement criteria for parallel and asynchronous computational environments. Firstly, the $EI^{(\mu,\lambda)}$ criterion is formulated in a more general form. Secondly, in section 4, bounds on $EI^{(\mu,\lambda)}$ are provided in section 3 and strategies for comparing $EI^{(\mu,\lambda)}$ at various design points are discussed. Finally, the impact of the number of available computing units, λ, on the performance, i.e., the method speed-up, is investigated in section 5.

2 A unified presentation of parallel expected improvements

The traditional improvement in the objective value [8], [9] at point x is defined as

$$I_{(\omega)}(x) = \max(0, f_{min} - \min(\mathbf{Y}_{(\omega)})) = \big(f_{min} - \min(\mathbf{Y}_{(\omega)})\big)^+ \tag{1}$$

where $f_{min} = min(\mathbb{Y})$ is the minimum of the observations, x is the new point for the next simulation and $\mathbf{Y}_{(\omega)} = \big(Y_{(\omega)}(x)\big)\,|\mathbb{Y}$ is the joint Gaussian vector (kriging metamodel) at the point of interest x conditioned on the past observations $\mathbb{X}, \mathbb{Y}$.

The *multi-points* improvement [6] can be rewritten as

$$I^{(\lambda)}_{(\omega)}(\mathbf{x}) = \max(0, f_{min} - \min(\mathbf{Y}^{\lambda}_{(\omega)})) = \Big(f_{min} - \min(\mathbf{Y}^{\lambda}_{(\omega)})\Big)^+ \tag{2}$$

where $f_{min} = min(\mathbb{Y})$ is the minimum of the previous observations, $\mathbf{x} = (x_1, ..., x_\lambda)$ are the λ new points for the next simulation and $\mathbf{Y}^{\lambda}_{(\omega)} = (Y_1, ..., Y_\lambda) = \big(Y_{(\omega)}(x_1), ..., Y_{(\omega)}(x_\lambda)\big)\,|\mathbb{Y}$ is joint Gaussian vector (kriging metamodel) at the new points $\mathbf{x} = (x_1, ..., x_\lambda)$ conditioned on the past observations $\mathbb{X}, \mathbb{Y}$.

The *multi-points asynchronous* improvement is defined as

$$I^{(\mu,\lambda)}_{(\omega)}(\mathbf{x}) = \max(0, \min(f_{min}, \mathbf{Y}^{\mu}_{(\omega)}) - \min(\mathbf{Y}^{\lambda}_{(\omega)})) = \Big(\min(f_{min}, \mathbf{Y}^{\mu}_{(\omega)}) - \min(\mathbf{Y}^{\lambda}_{(\omega)})\Big)^+ \tag{3}$$

where $f_{min} = min(\mathbb{Y})$ is the minimum of the previous observations, $\mathbf{x} = (x_{\mu+1}, ..., x_{\mu+\lambda})$ are the λ points for the next simulation and
$\mathbf{Y}^{\lambda}_{(\omega)} = (Y_{\mu+1}, ..., Y_{\mu+\lambda}) = \big(Y_{(\omega)}(x_{\mu+1}), ..., Y_{(\omega)}(x_{\mu+\lambda})\big)\,|\mathbb{Y}$ is again the Gaussian vector at the points of interest conditioned on the past observations $\mathbb{X}, \mathbb{Y}$, and
$\mathbf{Y}^{\mu}_{(\omega)} = (Y_1, ..., Y_\mu) = \big(Y_{(\omega)}(x_1), ..., Y_{(\omega)}(x_\mu)\big)\,|\mathbb{Y}$ is the joint Gaussian vector at the points where the simulator has not yet provided responses (i.e., the busy points). The multi-points asynchronous improvement has first been introduced in [4], [3]. As it can be readily seen from eq. (3), it is a natural measure of the progress made between points being or already calculated (the $\min(f_{min}, \mathbf{Y}^{\mu}_{(\omega)})$ term) and future points (the $\min(\mathbf{Y}^{\lambda}_{(\omega)})$ term). It has an important feature for asynchronous parallel calculation: it is null at the busy points. Indeed,

$$\text{If } \min(\mathbf{Y}^{\mu}_{(\omega)}) \leq f_{min} \quad , \quad I^{(\mu,\lambda)}_{(\omega)}(\mathbf{x}) = \Big(\min(\mathbf{Y}^{\mu}_{(\omega)}) - \min(\mathbf{Y}^{\mu}_{(\omega)})\Big)^+ = 0$$

$$\text{else } \min(\mathbf{Y}^{\mu}_{(\omega)}) > f_{min} \quad , \quad I^{(\mu,\lambda)}_{(\omega)}(\mathbf{x}) = \Big(f_{min} - \min(\mathbf{Y}^{\mu}_{(\omega)})\Big)^+ = 0$$

When used in the context of optimization, a natural choice for the sampling criterion is to use the expectation of the improvement, thus for multi-points asynchronous improvement in eq. (3) we have

$$EI^{(\mu,\lambda)}(\mathbf{x}) = \mathbb{E}_{\Omega}(I^{(\mu,\lambda)}_{(\omega)}(\mathbf{x})). \tag{4}$$

In the special case when $\mu = 0$ and $\lambda = 1$, it reduces to classical EI, with a known analytical expression. When $\mu = 0$ and $\lambda \geq 2$, it is equivalent to the qEI and we will further write it as $EI^{(\lambda)}$ (note that for $\mu = 0$ and $\lambda = 2$, an analytical expression is given in [6]). Note also that by applying the law of total expectations

eq. (4) is equivalent to EEI [4]. In general calculating eq. (4) amounts to estimating a multidimensional integral (where the number of dimensions depends on the number of available and busy nodes) with respect to Gaussian density and must be based on numerical procedures, in particular MC methods.

MC based estimation procedure samples the Gaussian vector $(\mathbf{Y}^{\lambda}, \mathbf{Y}^{\mu})$ and estimates the criteria values. Even though the calculation of the improvement (eq. (2) and eq. (3)) is not complex, the estimation of the criteria may become time consuming. The error of MC estimate depends on the number of samples, which may be large if a high precision of the estimate is needed. Such cases are very likely to occur at the end of the optimization when the points $\mathbf{x}$ proposed by the mainly converged optimizer are close to each other. Additionally, the crude MC sampling becomes inefficient with the number of EGO iterations because the probability of improvement becomes small.

2.1 Illustration of criteria

For illustrating the criteria we use the one dimensional test function

$$y = \sin(3x) - \exp\left(-\frac{(x+0.1)^2}{0.01}\right). \tag{5}$$

The test function together with predicted kriging mean and $2\sigma_{MC}$ confidence intervals, one point EI, 2 points EI or $EI^{(0,2)}$ and 2 points asynchronous EI or $EI^{(1,2)}$ are shown in fig. 1. It can be well seen that the criteria are symmetrical with respect to the line $x_1 = x_2$. Also notice that the maximum of $EI^{(0,2)}(\mathbf{x})$ is close to $\mathbf{x} = (x_1, x_2)$ where (x_1, x_2) corresponds to the modes of $EI(x)$.

The points obtained from $\max EI^{(\lambda)}$ for several iterations are shown in fig. 2. In this case of 'deceptive' function, the region of the true optimum is located faster with increasing λ. Larger λ values induce a better exploration of the design space allowing construction of a globally more accurate regression model and thus preventing EI from stagnation.

Figure 3 provides an example of points created through $\max EI^{(1,1)}$ for 3 iterations (6 time steps), assuming that calculation of the objective takes two time steps, and that the second computing node becomes available after one time step.

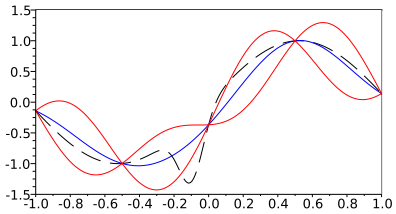
3 $EI^{(\mu,\lambda)}$ criteria bounds

In this section upper and lower bounds for the criteria of eq. (4) are derived. Firstly, in section 3.1, we address the parallel case $EI^{(\lambda)}$ and after, in section 3.2, we look at the asynchronous case $EI^{(\mu,\lambda)}$.

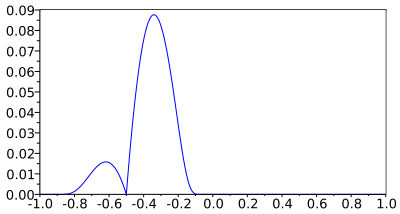
3.1 Bounds on the multipoints expected improvement

From the definition of multi-points improvement

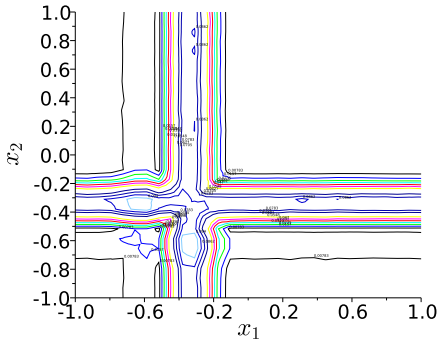
$$\begin{aligned} I^{(\lambda)}_{(\omega)}(\mathbf{x}) &= \max(0, f_{min} - \min(Y_1, ..., Y_\lambda)) \\ &\geq \max(0, f_{min} - Y_i) = I^{(\lambda=1)}_{(\omega)}(x_i) \end{aligned}$$



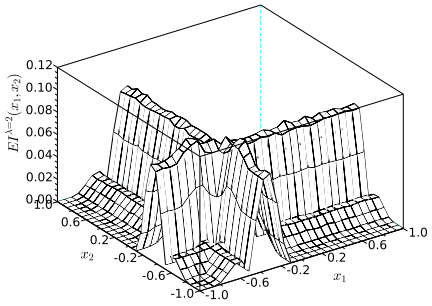
(a) Test function eq. (5) (dashed), kriging model based on 5 observations - predicted mean (blue), 2σ predicted confidence intervals (red).



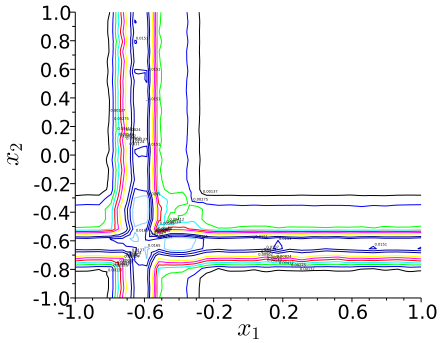
(b) One point EI, $EI(x)$.



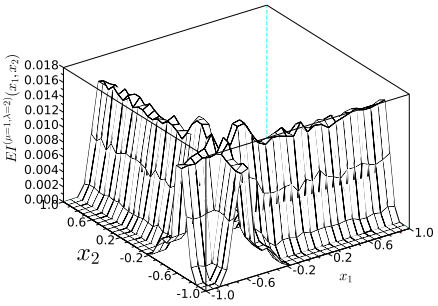
(c) Contour lines of 2 points EI, $EI^{(0,2)}(\mathbf{x})$



(d) 3D plot of 2 points EI, $EI^{(0,2)}(\mathbf{x})$



(e) Contour lines of 2 points async. EI (busy point $x_{busy} = -0.34$), $EI^{(1,2)}(\mathbf{x})$



(f) 3D plot of 2 points async. EI (busy point $x_{busy} = -0.34$), $EI^{(1,2)}(\mathbf{x})$

Fig. 1: Illustrations of the simple, the two-points and the two-points asynchronous Expected Improvements using the analytical test function eq. (5). $EI^{(0,2)}$ and $EI^{(1,2)}$ are calculated with 10000 MC simulations. Notice the maximum of $EI^{(0,2)}(\mathbf{x})$ is close to $\mathbf{x} = (x_1, x_2)$ where (x_1, x_2) corresponds to the modes of $EI(x)$.

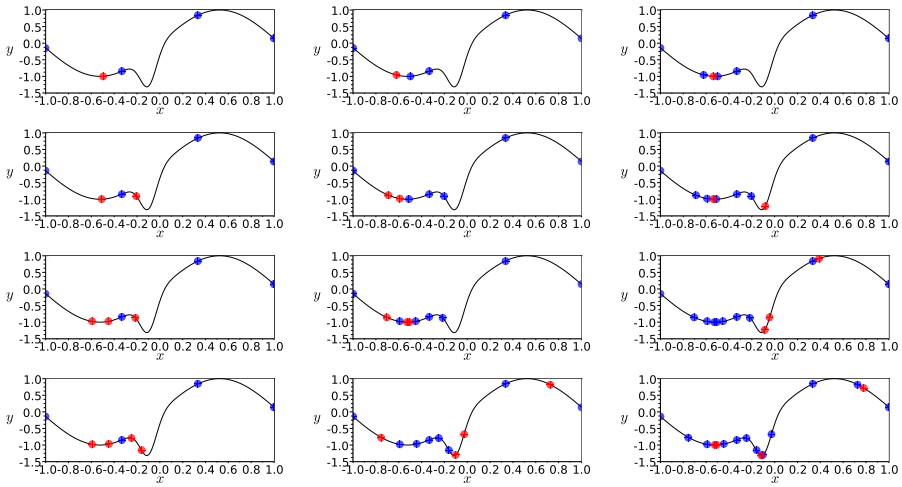


Fig. 2: Example of points generated by $\max EI^{(\lambda)}$ (red) at different iterations, true function (solid line), points used for kriging model (blue). First line $\lambda = 1$, second line $\lambda = 2$, third line $\lambda = 3$, forth line $\lambda = 4$. First column – first iteration, second column – second iteration, third column – third iteration. For the first iteration $\theta = 0.3$, for iterations 2 and 3, θ is estimated by maximizing the kriging model likelihood.

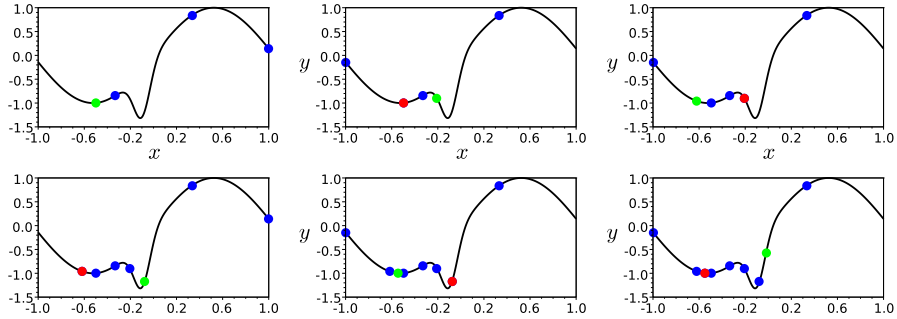


Fig. 3: Example of points generated by $\max EI^{(1,1)}$ during 6 time steps. It is assumed that calculation of the objective takes two time steps. Objective function (solid line), points used for kriging model (blue), points where the response value has been calculated (red), busy points (sent for calculation but not known yet – green). Initially and after the first time step $\theta = 0.3$, for subsequent steps (after data arrives) θ is estimated by maximizing the kriging model likelihood.

and therefore the lower bound on multi-points EI is

$$EI^{(\lambda)}(\mathbf{x}) \;\geq\; \max_{i=1,\lambda} EI^{(\lambda=1)}(x_i). \tag{6}$$

The upper bound on the expectation of multi-points improvement can also be derived from its definition

$$
\begin{aligned}
EI^{(\lambda)}_{(\omega)}(\mathbf{x}) &= \mathbb{E}[\max(0, f_{min} - \min(Y_1, ..., Y_\lambda))] \qquad (7)\\
&= \sum_{i=1}^{\lambda} \int_{-\infty}^{f_{min}} \int_{y_i}^{+\infty} \cdots \int_{y_i}^{+\infty} (f_{min} - y_i)\ f_{\mathcal{N}(\mu,\Sigma)}(y_1, ..., y_\lambda)(dy_j)_{j\neq i}\ dy_i \\
&\leq \sum_{i=1}^{\lambda} \int_{-\infty}^{f_{min}} \int_{-\infty}^{+\infty} \cdots \int_{-\infty}^{+\infty} (f_{min} - y_i)\ f_{\mathcal{N}(\mu,\Sigma)}(y_1, ..., y_\lambda)(dy_j)_{j\neq i}\ dy_i \\
&= \sum_{i=1}^{\lambda} \int_{-\infty}^{f_{min}} (f_{min} - y_i) \int_{-\infty}^{+\infty} \cdots \int_{-\infty}^{+\infty} f_{\mathcal{N}(\mu,\Sigma)}(y_1, ..., y_\lambda)(dy_j)_{j\neq i}\ dy_i \\
&= \sum_{i=1}^{\lambda} \int_{-\infty}^{f_{min}} (f_{min} - y_i) f_{\mathcal{N}(\mu_i,\sigma_i^2)}(y_i) dy_i \\
&= \sum_{i=1}^{\lambda} EI^{(\lambda=1)}(x_i) \qquad (8)
\end{aligned}
$$

where $f_{\mathcal{N}(\mu,\Sigma)}$ is the density of a multivariate normal distribution with mean μ and covariance matrix Σ, $(dy_j)_{j\neq i}$ is the sequence from dy_1 to dy_λ except dy_i.

The above relations provide the upper bound

$$
EI^{(\lambda)}(\mathbf{x}) \ \leq\ \sum_{i}^{\lambda} EI^{(\lambda=1)}(x_i). \qquad (9)
$$

The plots in fig. 4 show upper and lower bounds versus $EI^{(\lambda)}$ ($\lambda = 2$) for 100 random points. The test case is the same as in the previous example of fig. 1 with the function eq. (5). $EI^{(\lambda)}$ is calculated using 100 and 10000 MC simulations. It can be seen that for these points bounds are seemingly tight.

3.2 Bounds on the asynchronous multi-points expected improvement

From the definition of asynchronous multi-points improvement, one has

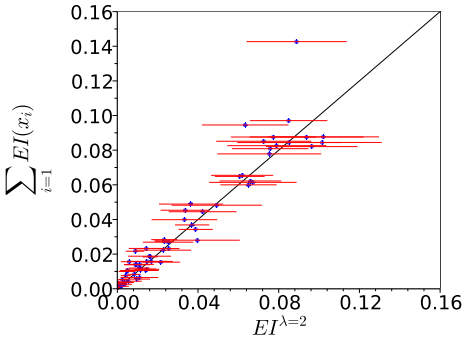
$$
\begin{aligned}
I^{(\mu,\lambda)}_{(\omega)}(\mathbf{x}) &= \max(0, \min(f_{min}, Y_1^\mu, ..., Y_\mu^\mu) - \min(Y_1^\lambda, ..., Y_\lambda^\lambda)) \qquad (10)\\
&\leq \begin{cases} \max(0, f_{min} - \min(Y_1^\lambda, ..., Y_\lambda^\lambda)) \\ \max(0, Y_i^\mu - \min(Y_1^\lambda, ..., Y_\lambda^\lambda)) \end{cases} \qquad (11)
\end{aligned}
$$

and the expectation is

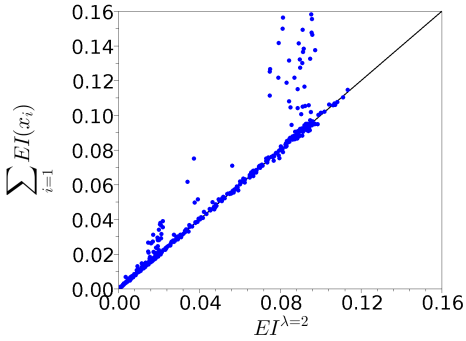
$$
EI^{(\mu,\lambda)}_{(\omega)}(\mathbf{x}) = \qquad \leq \begin{cases} \mathbb{E}[\max(0, f_{min} - \min(Y_1^\lambda, ..., Y_\lambda^\lambda))] = EI^{(\lambda)}_{(\omega)}(\mathbf{x}) \\ \mathbb{E}[\max(0, Y_i^\mu - \min(Y_1^\lambda, ..., Y_\lambda^\lambda))] \end{cases} \qquad (12)
$$

If we define

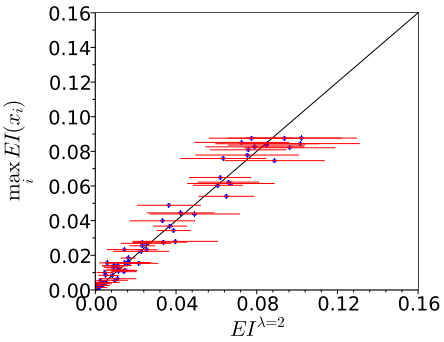
$$
I^{*(i,j)}_{(\omega)}(\mathbf{x}) = \max(0, Y_i^\mu - Y_j^\lambda) \qquad (13)
$$



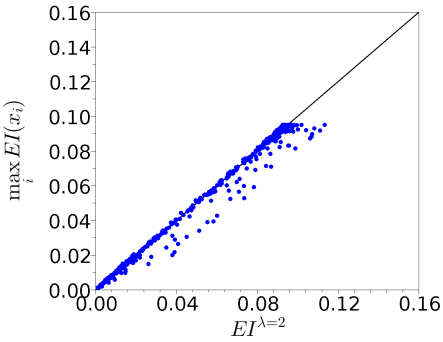
(a) Upper bound versus $EI^{(\lambda)}$ with estimated 2σ confidence intervals. $EI^{(\lambda)}$ estimated using 100 MC samples.



(b) Upper bound versus $EI^{(\lambda)}$ with estimated 2σ confidence intervals. $EI^{(\lambda)}$ estimated using 10000 MC samples.



(c) Lower bound versus $EI^{(\lambda)}$ with estimated 2σ confidence intervals. $EI^{(\lambda)}$ estimated using 100 MC samples.



(d) Lower bound versus $EI^{(\lambda)}$ with estimated 2σ confidence intervals. $EI^{(\lambda)}$ estimated using 10000 MC samples.

Fig. 4: Upper and lower bounds versus $EI^{(\lambda)}$ ($\lambda = 2$) for random chosen points with 2σ confidence intervals. The test case is based on eq. (5) as in fig. 1. $EI^{(\lambda)}$ is calculated using 100 and 10000 MC simulations. Notice that for the points in this example bounds are seemingly tight.

and

$$EI^{*(i,j)}(\mathbf{x}) = \mathbb{E}_{\Omega}(I^{*(i,j)}_{(\omega)}(\mathbf{x})) \tag{14}$$

then using a similar reasoning as in eq. (7) one can say that

$$\mathbb{E}[\max(0, Y_i^{\mu} - \min(Y_1^{\lambda}, ..., Y_{\lambda}^{\lambda}))] \leq \sum_{j=1}^{\lambda} EI^{*(i=1,j)}(x_j). \tag{15}$$

The upper bound is

$$EI^{(\mu,\lambda)}(\mathbf{x}) \;\leq\; \min\left(\sum_j^{\lambda} EI^{(\lambda=1)}(x_j), \sum_j^{\lambda} EI^{*(i=1,j)}(x_j), ..., \sum_j^{\lambda} EI^{*(i=\mu,j)}(x_j)\right). \tag{16}$$

Notice that all components on the right hand side of eq. (16) have analytical expressions since $Y_i^{\mu} Y_j^{\lambda} \mid \mathbb{Y}$ is Gaussian with mean and variance known from kriging.

To estimate the lower bound on the asynchronous expected improvement one can write

$$\begin{aligned} I_{(\omega)}^{(\mu,\lambda)}(\mathbf{x}) &= \max(0, \min(f_{min}, Y_1^{\mu}, ..., Y_{\mu}^{\mu}) - \min(Y_1^{\lambda}, ..., Y_{\lambda}^{\lambda})) \\ &= \max(0, \min(f_{min}, Y_1^{\mu}, ..., Y_{\mu}^{\mu}) - Y_1^{\lambda}, ..., \min(f_{min}, Y_1^{\mu}, ..., Y_{\mu}^{\mu}) - Y_{\lambda}^{\lambda}) \\ &\geq (\min(f_{min}, Y_1^{\mu}, ..., Y_{\mu}^{\mu}) - Y_j^{\lambda})^{+}. \end{aligned}$$

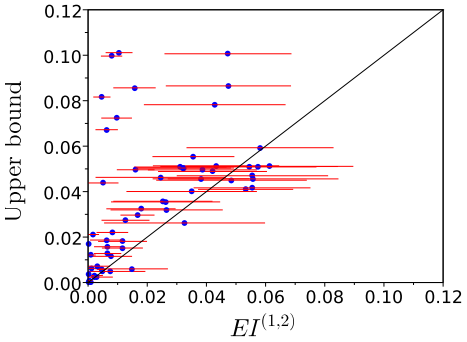
The expectations of type $\mathbb{E}[(\min(f_{min}, Y_1^{\mu}, ..., Y_{\mu}^{\mu}) - Y_j^{\lambda})^{+}]$ are again integrals on the $R^{\mu+1}$ hyperspace truncated by several hyperplanes. As noted previously, the estimation of such integrals is based on numerical procedures and are not trivial. Instead a trivial lower bound of the asynchronous expected improvement from the definition of the improvement can be used

$$EI^{(\mu,\lambda)}(\mathbf{x}) \geq 0. \tag{17}$$

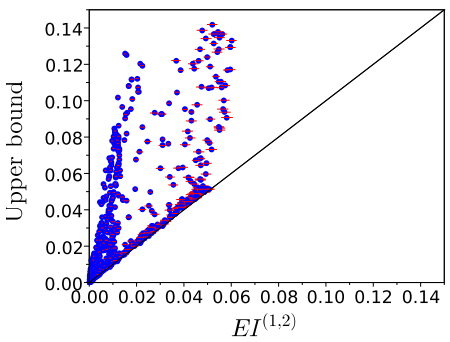
The plots in fig. 5 show upper bounds versus $EI^{(\mu,\lambda)}$ ($\mu = 1, \lambda = 2$) for a set of random points using the same test function eq. (5) as in the previous examples fig. 4 and fig. 1. $EI^{(\mu,\lambda)}$ is calculated using 100 and 10000 MC simulations. It can be seen that there are clusters of points where the upper bound is considerably overestimated.

4 $EI^{(\mu,\lambda)}$ estimation and selection of the candidate point

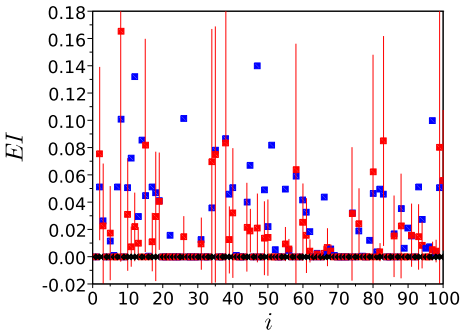
The optimization criteria of eq. (4) are intended to choose the most promising set of points for the next simulations. As already noted above, the computation of these criteria is not trivial, so that a simple heuristic strategy would be to use the bounds derived in the previous section. Because the bounds do not account for couplings between points (to the exception of the EI^* values which account for pairwise couplings through $Y_i^{\mu} Y_j^{\lambda}$ in eq. (13)), the criteria values may be considerably under or over estimated (for multi-points $EI^{(\lambda)}$ fig. 4 and especially asynchronous multi-points $EI^{(\mu,\lambda)}$ fig. 5). Therefore, we propose a mixed strategy which utilizes bounds together with MC estimations.



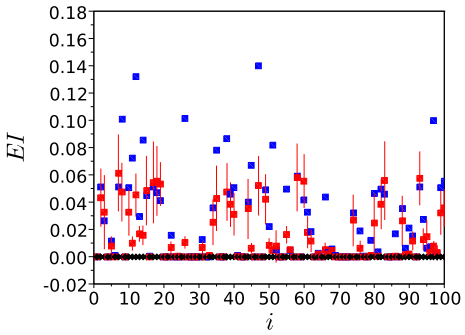
(a) Upper bound versus $EI^{(\mu,\lambda)}$ for 100 random points with estimated 2σ confidence intervals. $EI^{(\mu,\lambda)}$ estimates using 100 MC samples.



(b) Upper bound versus $EI^{(\mu,\lambda)}$ for 1000 random points. $EI^{(\lambda)}$ estimates using 10000 MC samples.



(c) Lower bound (diamond), upper bound (square), estimated $EI^{(\mu,\lambda)}$ (red dot) with estimated 2σ confidence intervals (red lines) on 100 random points. $EI^{(\mu,\lambda)}$ estimates using 10 MC samples.



(d) Lower bound (diamond), upper bound (square), estimated $EI^{(\mu,\lambda)}$ (red dot) with estimated 2σ confidence intervals (red lines) on 100 random points. $EI^{(\mu,\lambda)}$ estimates using 100 MC samples.

Fig. 5: Upper and lower bounds, $EI^{(\mu,\lambda)}$ ($\mu = 1, \lambda = 2$) for 100 and 1000 random points with 2σ confidence intervals. The test case is based on eq. (5) like in fig. 1. $EI^{(\mu,\lambda)}$ is calculated using 100 and 10000 MC simulations. Notice in this example that for many points the upper bounds are overestimated.

4.1 Estimation and confidence intervals

The MC estimation allows us to estimate the expectations of improvements in eq. (4). By using crude MC the estimated $EI^{(\mu,\lambda)}$ (denoted in this section as EI) is

$$EI_{MC}(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{N} I_{(\omega)}(\mathbf{x})$$

where N is the number of samples and $I_{(\omega)}(\mathbf{x})$ is the improvement calculated from the sampled trajectory of the conditional Gaussian process at the points of interest.

"Points" has a plural here because, as the reader may remember, in the case of the parallel asynchronous expected improvement, expectation is calculated over joint random variables located at $\mu + \lambda$ points in the optimization variable space.

The error variance of this estimate also can be estimated from data as

$$\sigma^2_{MC} = \frac{1}{n(n-1)} \sum_{i=1}^{N} (I_{(\omega)}(\mathbf{x}) - EI_{MC})^2.$$

Therefore $(EI_{MC}|EI)$ follows Student's t-distribution with mean EI and variance σ^2_{MC} with $n-1$ degrees of freedom. Bounds discussed in previous chapter provide us with prior information on EI. Lets assume that $\alpha \leq EI \leq \beta$ and EI uniformly distributed between its bounds, $EI \sim \mathcal{U}(\alpha, \beta)$. Using Bayes theorem it is possible to show that $(EI|EI_{MC})$ follows a truncated Student's t-distribution

$$p(EI|EI_{MC}) = \frac{p(EI_{MC}|EI)p(EI)}{\int_-^+ p(EI_{MC}|EI)p(EI)dEI} \tag{18}$$

$$= \frac{1}{\sigma_{MC}} f_\nu \left(\frac{EI - EI_{MC}}{\sigma_{MC}} \right) \left[F_\nu \left(\frac{\alpha - EI_{MC}}{\sigma_{MC}} \right) - F_\nu \left(\frac{\beta - EI_{MC}}{\sigma_{MC}} \right) \right]^{-1} \tag{19}$$

where $F_\nu(.)$ and $f_\nu(.)$ are the c.d.f. and p.d.f. of the Student's t- variable with $\nu = n-1$ degrees of freedom. The moments of truncated t-distribution are given for example in [10] and the formulas are copied in appendix A. Numerical tests have shown that these formulas become numerically unstable as the bounds are on the same side and far from the mean.

However it is also know that as ν increases the t-distribution is well approximated by the Gaussian distribution. The density function of truncated Gaussian distribution is equivalent to eq. (18) except that instead of $F_\nu(.)$ and $f_\nu(.)$ we have $\Phi(.)$ and $\phi(.)$ which are the c.d.f. and p.d.f. of the standard normal distribution. The mean and variance of truncated Gaussian random are known and if we denote by $u_1 = (\alpha - EI_{MC})/\sigma_{MC}$ and $u_2 = (\beta - EI_{MC})/\sigma_{MC}$, then

$$\mathbb{E}(EI|EI_{MC}) = EI_{MC} + \frac{\phi(u_1) - \phi(u_2)}{\Phi(u_2) - \Phi(u_1)} \sigma_{MC} \tag{20}$$

and

$$\mathbb{VAR}(EI|EI_{MC}) = \sigma^2_{MC} \left[1 + \frac{u_1\phi(u_1) - u_2\phi(u_2)}{\Phi(u_2) - \Phi(u_1)} - \left(\frac{\phi(u_1) - \phi(u_2)}{\Phi(u_2) - \Phi(u_1)} \right)^2 \right]. \tag{21}$$

The $[\mathbb{E}(EI|EI_{MC}) - k\sqrt{\mathbb{VAR}(EI|EI_{MC})}, \mathbb{E}(EI|EI_{MC}) + k\sqrt{\mathbb{VAR}(EI|EI_{MC})}]$ when $k = 1$ gives a 68% confidence that EI is inside this interval.

4.2 Ranking of candidate points

The confidence intervals from the previous section allow to compare two points. Let say that we have a set of points $\mathbf{x}_i$, $i = 1..n$, MC estimates $EI_{MC}(\mathbf{x}_i)$ and $\sigma^2_{MC}(\mathbf{x}_i)$, let's denote by $\mu_i = \mathbb{E}(EI|EI_{MC}(\mathbf{x}_i))$ and $\sigma_i = \sqrt{\mathbb{VAR}(EI|EI_{MC}(\mathbf{x}_i))}$.

Now, with over 60% confidence ($k = 1$),

$$\text{if} \quad \mu_i - k\sigma_i \geq \mu_j + k\sigma_j \quad \text{then} \quad EI(\mathbf{x}_i) \geq EI(\mathbf{x}_j) \tag{22}$$

$$\text{if} \quad \mu_i + k\sigma_i < \mu_j - k\sigma_j \quad \text{then} \quad EI(\mathbf{x}_i) < EI(\mathbf{x}_j). \tag{23}$$

If neither eq. (22) nor eq. (23) hold then we do not have enough information to safely discriminate between the two points and additional data is necessary. In order to reduce the confidence intervals $N = N_i + N_j$ additional MC samples are computed, where N_i and N_j are the number of samples at points $\mathbf{x}_i$ and $\mathbf{x}_j$, respectively. A simple strategy is to add a number of samples proportional to the variance of the estimate,

$$N_i = \frac{\sigma_i^* N}{\sigma_i^* + \sigma_j^*}, \quad N_j = N - N_i$$

where $\sigma_i^* = \sigma_i, \sigma_j^* = \sigma_j$, if $\sigma_{MC}(x_i) > 0, \sigma_{MC}(x_j) > 0$.

Special care must be taken when none of the MC samples actually obtains trajectory where improvement is positive. In such cases the $\sigma_{MC} = 0$ and $EI_{MC} = 0$ are underestimated. On the other hand increasing number of total MC samples with no hits (trajectory providing positive improvement) indicates that true value of EI is small. To safely discriminate between two candidate points one needs to account for such situations.

If point x_j has no MC hit but x_i has at least one sample where improvement is positive it is possible to estimate the improvement variance $\sigma^2_{MC}(x_i)$ and use this variance to overestimate the improvement variance at the other point x_j as $N_i/N_j\sigma^2_{MC}(x_i)$, where N_i and N_j are the number of already computed MC samples.Therefore if $\sigma_i > 0$ and $\sigma_j = 0$ then $\sigma_i^* = \sigma_i$ and σ_j^* is calculated using eq. (21) where $\sigma_{MC}(x_j) = \sigma_{MC}(x_i)\sqrt{\frac{n_i}{n_j}}$.

However if no MC samples have positive improvement at both points .i.e. $\sigma_{MC}(x_i) = 0, \sigma_{MC}(x_j) = 0$ then the number of samples should be increased equally $N_i = N_j = N/2$, until the probability of improvement is small and we can assume that the true value of EI at both points is very small and further sampling is unnecessary, or until the MC budget is exceeded.

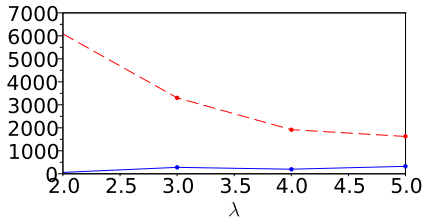
4.3 Illustration of the impact of bounds

In fig. 6 we show the 80th percentile of the number of MC simulations versus λ for comparing random pairs of points. For each λ setting, 100 random trajectories of Gaussian process are generated (and act as sample functions), the kriging models are built from 4 evenly spaced points using the same covariance form as that used in the process generation.

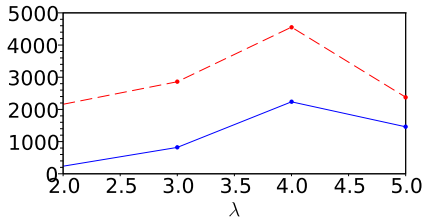
For each trajectory a pair of points is randomly selected (i.e., a set of μ points and two sets of λ points) and their $EI^{(\mu,\lambda)}$ values are compared using MC based procedures

with and without bounds. The initial number of MC samples is 10 and at each step $N = 20$. The maximum number of MC samples for each point is limited to 10^5. If more MC samples are necessary one concludes that for the two points values of $EI^{(\mu,\lambda)}$ are very similar and the points cannot be compared. The percentiles in fig. 6 are computed only on points that can be compared. For test cases where $\mu = 0$, on the average 99 pairs of points where comparable within this number of maximum MC evaluations. For cases where $\mu = 1$ and $\mu = 3$, on the average only 75 and 69 pairs where comparable.

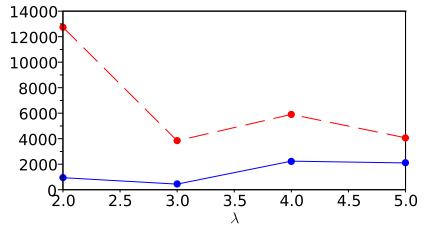
The results indicate that the bounds reduce the necessary number of MC evaluations, however (especially in asynchronous case $\mu > 0$), the discrimination of points often requires a very large number of MC evaluations. Furthermore, in practice one is interested in selecting points with maximum $EI^{(\mu,\lambda)}$: the optimization of $EI^{(\mu,\lambda)}$ eventually leads to regions where the values of $EI^{(\mu,\lambda)}$ become similar and a much larger number of MC simulations are needed for the ranking of points i.e., the non comparable cases occur more often. Therefore the strategy based on mixing crude MC with bounds is effective only at the initial optimization steps when the global exploration of $EI^{(\mu,\lambda)}$ resembles random sampling.



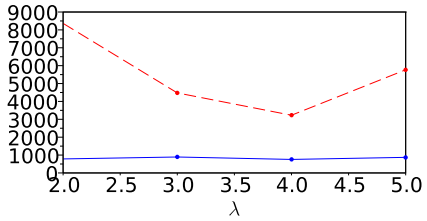
(a) $\mu = 0$. Sample trajectories generated on 1000 point grid with Matern 5/2 kernel and $\theta = 0.3$.



(b) $\mu = 0$. Sample trajectories generated on 1000 point grid with Matern 5/2 kernel and $\theta = 0.15$.



(c) $\mu = 1$. Sample trajectories generated on 1000 point grid with Matern 5/2 kernel and $\theta = 0.3$.



(d) $\mu = 3$. Sample trajectories generated on 1000 point grid with Matern 5/2 kernel and $\theta = 0.3$.

Fig. 6: 80th percentile of the number of MC simulations necessary to discriminate between $EI^{(\mu,\lambda)}$ at two random points versus λ. The number of MC simulations without bounds (dashed line), number of MC simulations with bounds (solid line).

5 An empirical study of $EI^{(0,\lambda)}$ scale-up properties

To empirically compare the efficiency of the parallel improvement criteria (eq. (4)) to classical EI, we use random test functions in one dimension, i.e. we sample Gaussian process trajectories on a 1000 point grid using Matern 5/2 kernel functions [11] with fixed scaling parameters $\theta = 0.3$ (fig. 8, fig. 10) and $\theta = 0.15$ (fig. 9, fig. 11).

5.1 Optimization strategy

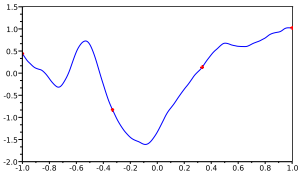
For the maximization of $EI^{(\lambda)}(\mathbf{x})$, we use the Covariance Matrix Adaptation Evolution Strategy algorithm (CMA-ES, [7]) with some changes. Firstly, as we have data only at discrete locations, every point sampled by CMA-ES is mapped to the nearest grid point. Secondly, the ranking of points in a given population is based on the comparison procedure described in section 4. This MC based ranking makes the objective function noisy, which is compatible with the stochastic CMA-ES algorithm, but adds randomness in the tests.

When comparing two points $\mathbf{x}_i$ and $\mathbf{x}_j$ (two sets of points in actual space) which have close EI values (for example the points are close to each other) the necessary number of MC simulations for safe comparison may be very large. In order to keep the optimization procedure computationally feasible, the maximum number of MC simulation for estimating EI at one point is 10^5. If the numbers of MC simulations for both points exceed this budget, we assume that it is not possible to discriminate between the two points (as explained in section 4) because their criteria values are very close. This situation occurs during the optimization, especially towards the end of the CMA-ES iterations, when the variance of the population is small. Furthermore, the values of EI (and probability of improvement) decreases with the number of iterations. This further increases the number of MC samples needed to discriminate between two points, as most of the sampled trajectories will be above the best observed point. These problems illustrate the necessity for more efficient criteria estimation procedures (such as MC variance reduction techniques), where more trajectories that are interesting for calculating the criteria (below the best point) are used.

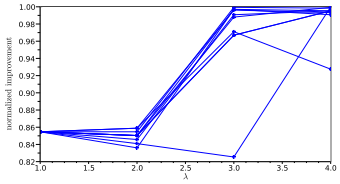
5.2 Results

The impact of λ on the speed-up of the parallel expected improvements is examined by comparing the actual improvements (difference between best objective values at the points of the initial DOE and at the λ points provided by criteria) after one time step. The assumption is made that the simultaneous calculation of the objective is possible on multiple computing nodes, however the actual number of utilized processors depends on the criterion, i.e. $EI^{(1)}$ provides one new point, $EI^{(2)}$ two new points, etc.

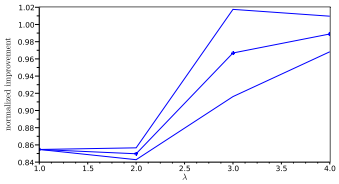
The variation in actual improvement due to the EI optimization strategy is illustrated in fig. 7, where 10 runs of EI maximization are performed on a single test trajectory fig. 7a and the mean actual improvement with 1σ confidence bounds fig. 7c and mean improvement rank together with 1σ confidence fig. 7d. The improvement rank is obtained for each trajectory by ordering the best actual improvement values with respect to λ. It is used as a normalization method for better visualization of the speed-up related to parallel EIs.



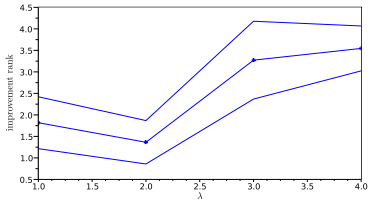
(a) Sample trajectory on 1000 point grid with Matern 5/2 kernel and $\theta = 0.3$.



(b) Actual improvement of $EI^{(\lambda)}$ for 10 independent maximizations, $\lambda = 1...4$.



(c) Actual mean improvement, 1σ confidence intervals of $EI^{(\lambda)}$, $\lambda = 1...4$.



(d) Mean improvement rank, 1σ confidence intervals of $EI^{(\lambda)}$, $\lambda = 1...4$.

Fig. 7: Variation in actual improvement due to the stochastic maximization of $EI^{(\lambda)}$. Statistics calculated for 10 runs on a fixed test function.

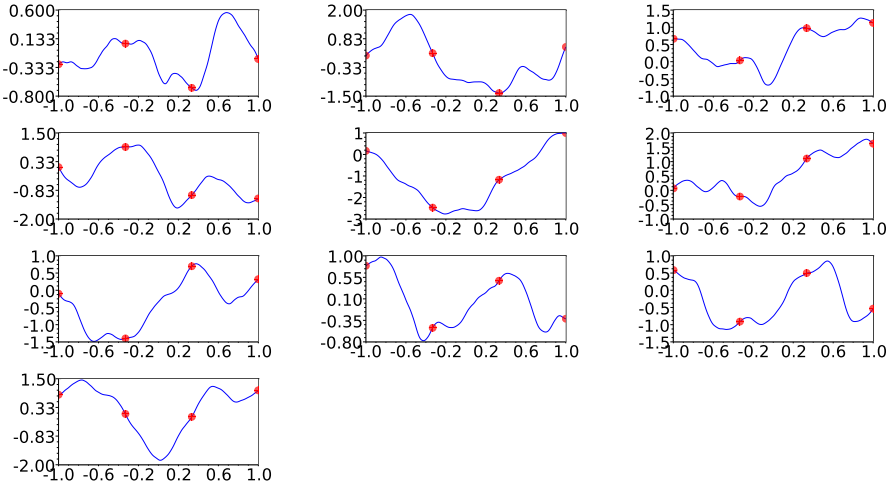
The mean improvement, mean rank with confidence intervals for actual improvement on 10 random test functions (fig. 8a and fig. 9a) at points of $\max_{\mathbf{x}} EI^{(\lambda)}(\mathbf{x})$ are illustrated in fig. 8c and fig. 9c. The initial kriging model is built using 4 evenly spaced points and the covariance hyper-parameter θ is fixed equal to the value used in trajectory generation. The results of this test indicate that the $EI^{(\lambda)}$ has a sub-linear improvement with respect to λ.

The λ impact on improvement is also studied in a more realistic scenario, where points of maximum EI are added *sequentially* to the DOE as they would during an optimization. The following typical three steps optimization strategy is investigated :

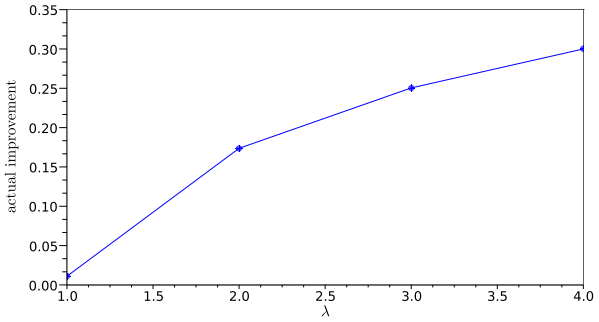
1. the kriging model is built from 4 initial points;
2. kriging covariance parameters are not known a priori and are estimated by maximizing likelihood;
3. $EI^{(\lambda)}(\mathbf{x})$ is maximized by CMA-ES;
4. the λ points together with $f(\mathbf{x})$ are added to the DOE.

Steps 2, 3, 4 are repeated for 2 iterations.

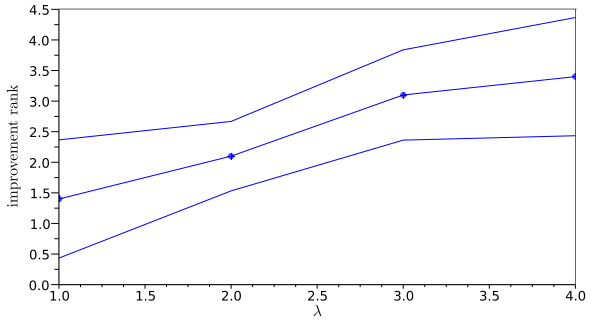
The statistics of the overall best improvement (the best improvement after three iterations with respect to initial 4 point DOE) versus λ over 10 random test functions using the three step optimization strategy are shown in fig. 10 and fig. 11. As in the previous cases, the results suggest that the $EI^{(\lambda)}$ provides a sub-linear improvement with respect to λ.



(a) Sample trajectories on 1000 point grid with Matern 5/2 kernel and $\theta = 0.3$.

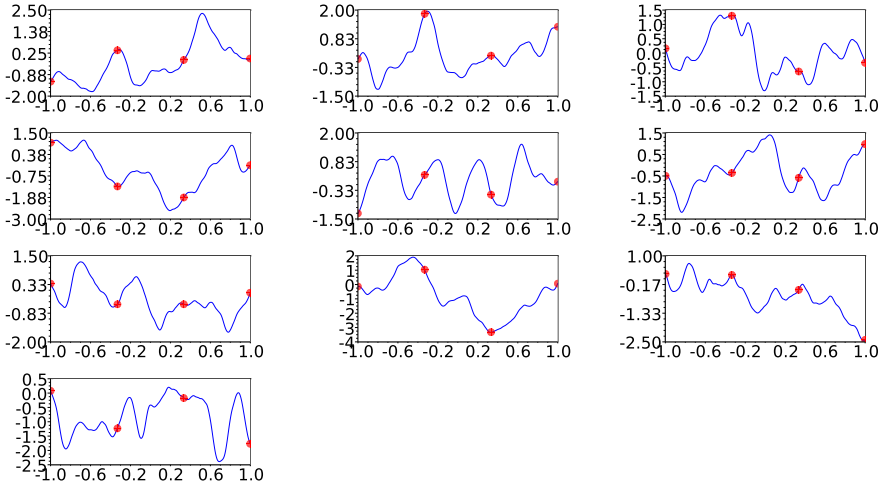


(b) Actual mean improvement over 10 trajectories for $\lambda = 1...4$.

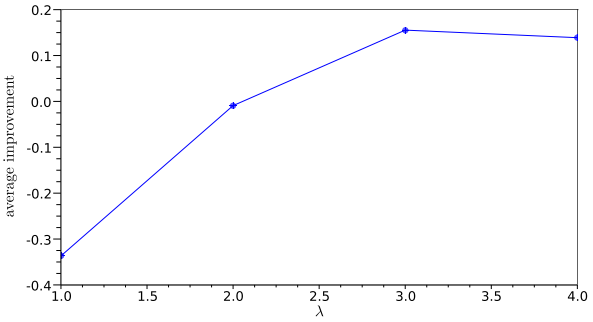


(c) Mean improvement rank, 1σ confidence intervals over 10 trajectories for $\lambda = 1...4$.

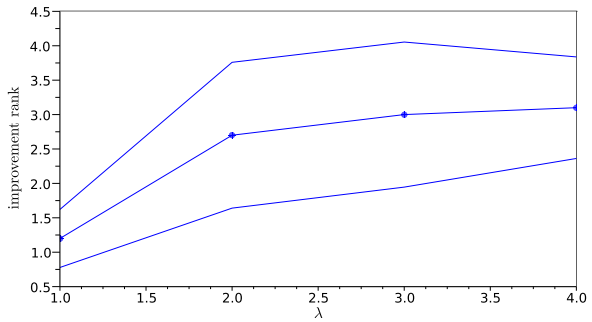
Fig. 8: λ impact on performance observed on 10 relatively smooth ($\theta = 0.3$) random functions



(a) Sample trajectories on 1000 point grid with Matern 5/2 kernel and $\theta = 0.15$.

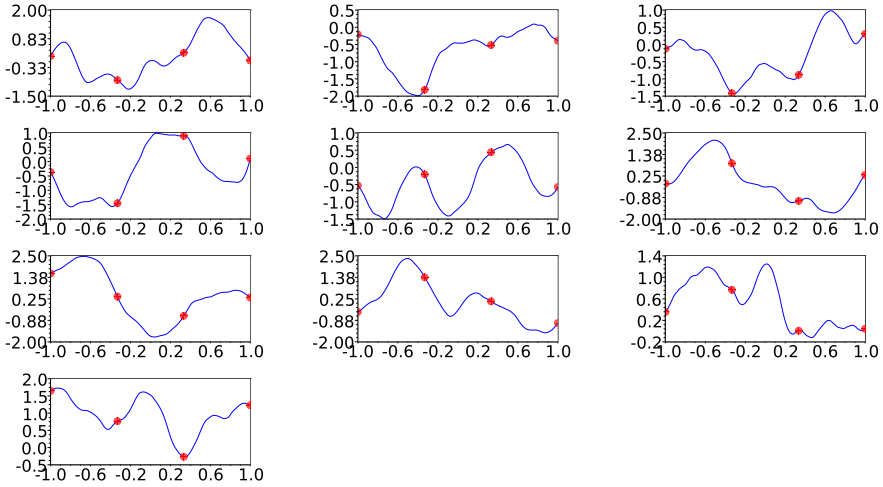


(b) Actual mean improvement over 10 trajectories for $\lambda = 1...4$.

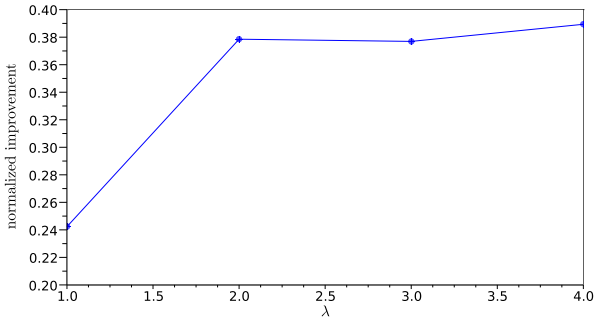


(c) Actual average improvement over 10 trajectories for $\lambda = 1...4$.

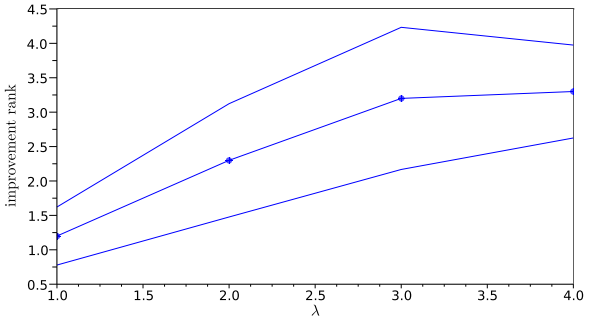
Fig. 9: λ impact on performance observed on 10 shaky ($\theta = 0.15$) random functions



(a) Sample trajectories on 1000 point grid with Matern 5/2 kernel and $\theta = 0.3$.

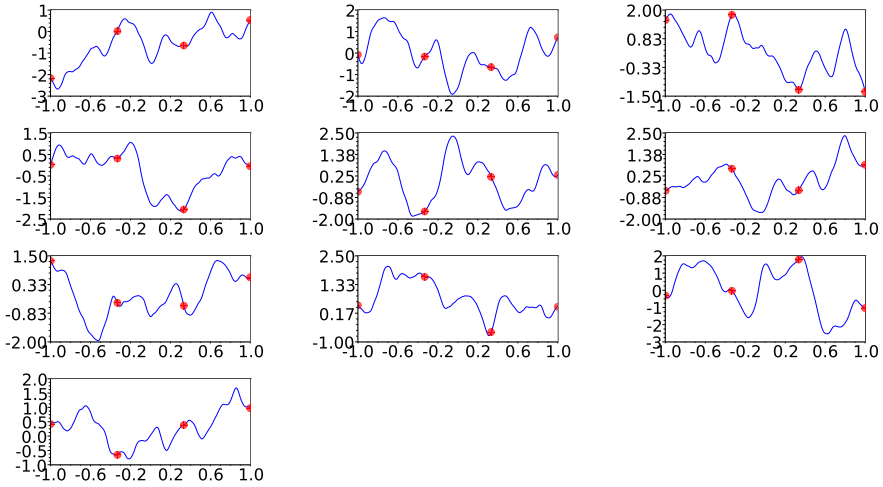


(b) Actual mean overall improvement after 3 iterations, over 10 random trajectories, $\lambda = 1...4$.

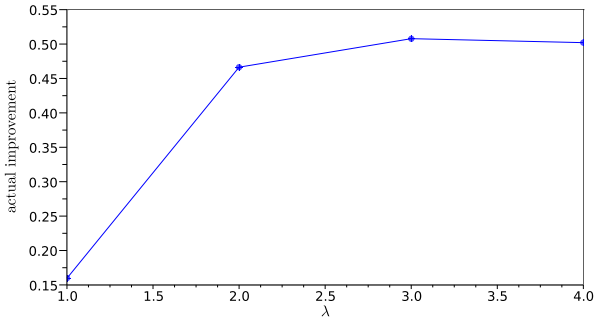


(c) Mean overall improvement rank, 1σ confidence intervals over 10 random trajectories, $\lambda = 1...4$.

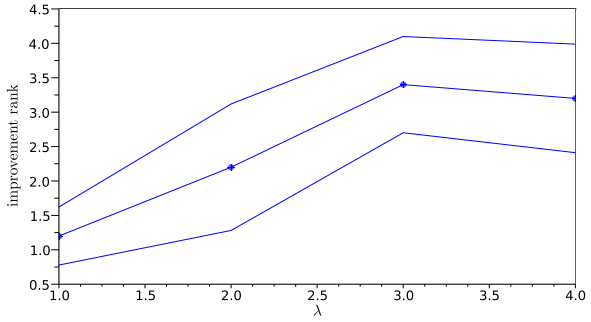
Fig. 10: λ impact on performance observed on 10 relatively smooth ($\theta = 0.3$) random functions over 3 iterations.



(a) Sample trajectories on 1000 point grid with Matern 5/2 kernel and $\theta = 0.15$.



(b) Actual mean overall improvement after 3 iterations, over 10 random trajectories, $\lambda = 1...4$.



(c) Mean overall improvement rank, 1σ confidence intervals over 10 random trajectories, $\lambda = 1...4$.

Fig. 11: λ impact on performance observed on 10 shaky ($\theta = 0.15$) random functions over 3 iterations.

6 Conclusion

In this study, the $EI^{(\mu,\lambda)}$ criterion has been formulated by combining parallel and asynchronous versions of the expected improvement criterion. $EI^{(\mu,\lambda)}$ measures the expected improvement brought by λ new points when the value of the objective function at μ already chosen points is not yet known but will be.

Bounds on $EI^{(\mu,\lambda)}$ have been provided for use in MC based estimation and comparison strategy. The easily calculable bounds may be useful at the initial steps of $EI^{(\mu,\lambda)}$ maximization when comparing points at distant locations and the coupling between design points is weak.

Finally, the effect of the number of available nodes, λ, on the optimization performance has been investigated in 1D test cases. The results indicate that parallel EIs provide sub-linear speed-up. The observed speed-up is in agreement with the intuition that the ability to obtain points in parallel should provide better and perhaps "safer" results. Note that in the test cases studied here the actual type of kriging covariance kernel is known. It seems, however, that in general the parallel calculation of λ points provides better exploration of the design space. Therefore it may partially solve the problem of deceptive functions [8],[2]. Better exploration of the design space allows globally more accurate construction of kriging models and therefore reduces the risk of stagnating searches in local regions due to deceptive initial states.

The implementation of the $EI^{(\mu,\lambda)}(\mathbf{x})$ for optimization indicates several difficulties. Firstly, the necessary number of MC simulations increases a lot in order to be able to discriminate close points. Secondly, crude MC sampling is often inefficient because trajectories that are better than the best observation become seldom as the optimization proceeds. Thirdly, the number of dimensions of $\mathbf{x}$ increases proportionally to the number of nodes. These challenges call for further studies on specialized optimization heuristics and MC procedures.

References

1. Vincent Dubourg, Bruno Sudret, and Jean-Marc Bourinet. Reliability-based design optimization using kriging surrogates and subset simulation. *Structural and Multidisciplinary Optimization*, pages 1–18, 2011. 10.1007/s00158-011-0653-8.
2. Alexander I. Forrester and Donald R. Jones. Global optimization of deceptive functions with sparse sampling. 2008.
3. D. Ginsbourger, J. Janusevskis, R. Le Riche, and C. Chevalier. Dealing with asynchronicity in kriging-based parallel global optimization. In *Second World Congress on Global Optimization in Engineering & Science (WCGO-2011)*, July 3-7 2011.
4. David Ginsbourger, Janis Janusevskis, and Rodolphe Le Riche. Dealing with asynchronicity in parallel Gaussian Process based global optimization. Technical report, July 2010. Deliverable no. 2.1.1-A of the ANR / OMD2 project available as `http://hal.archives-ouvertes.fr/hal-00507632`.
5. David Ginsbourger and Rodolphe Le Riche. Towards GP-based optimization with finite time horizon. In Alessandra Giovagnoli, Anthony C. Atkinson, Bernard Torsney, and May Caterina, editors, *mODa 9 Advances in Model-Oriented Design and Analysis*, pages 89–96. Springer, 2010.
6. David Ginsbourger, Rodolphe Le Riche, and Laurent Carraro. Kriging is well-suited to parallelize optimization. In Yoel Tenne and Chi-Keong Goh, editors, *Computational Intelligence in Expensive Optimization Problems*, Springer series in Evolutionary Learning and Optimization, pages 131–162. springer, 08 2009.
7. N. Hansen. The CMA evolution strategy: a comparing review. In J.A. Lozano, P. Larranaga, I. Inza, and E. Bengoetxea, editors, *Towards a new evolutionary computation. Advances on estimation of distribution algorithms*, pages 75–102. Springer, 2006.

8. Donald R. Jones. A taxonomy of global optimization methods based on response surfaces. *Journal of Global Optimization*, 21:345–383, 2001.
9. Donald R. Jones, Matthias Schonlau, and William J. Welch. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13(4):455–492, 1998.
10. Hea-Jung Kim. Moments of truncated student-t distribution. *Journal of the Korean Statistical Society*, 37(1):81 – 87, 2008.
11. Carl E. Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, December 2005.
12. Matthias Schonlau. *Computer experiments and global optimization*. PhD thesis, Waterloo, Ont., Canada, Canada, 1997. AAINQ22234.
13. William JALBY. Scientific directions for exascale computing research. In *forum Ter@tech*, Palaiseau, France, June 2011. Ecole Polytechnique.

A Moments of Students *t*-truncated distribution

If we denote by $T = (EI - EI_{MC})/\sigma_{MCMC}$ and $a = (\alpha - EI_{MC})/\sigma_{MCMC}$ and $b = (\beta - EI_{MC})/\sigma_{MCMC}$ then the moments of truncated t-distribution in interval $[a, b]$ are given for example in [10] and

$$\mathbb{E}(T) = G_\nu(1)(A_{(\nu)}^{-(\nu-1)/2} - B_{(\nu)}^{-(\nu-1)/2})$$

$$\mathbb{E}(T^2) = \frac{\nu}{\nu-2} + G_\nu(1)(aA_{(\nu)}^{-(\nu-1)/2} - bB_{(\nu)}^{-(\nu-1)/2})$$

where

$$G_\nu(l) = \frac{\Gamma((\nu-l)/2)\nu^{\nu/2}}{2[F_\nu(b) - F_\nu(a)]\Gamma(\nu/2)\Gamma(1/2)}$$

and $A_{(\nu)} = \nu + a^2$ and $B_{(\nu)} = \nu + b^2$. In our case

$$\mathbb{E}(EI|EI_{MC}) = EI_{MC} + \sigma_{MCMC}\mathbb{E}(T)$$

and

$$\mathbb{VAR}(EI|EI_{MC}) = \sigma_{MCMC}^2(\mathbb{E}(T^2) - \mathbb{E}(T)^2)$$