ECAI 2010
H. Coelho et al. (Eds.)
IOS Press, 2010

doi:10.3233/978-1-60750-606-5-249

Bayesian Monte Carlo for the Global Optimization of Expensive Functions

Perry Groot and **Adriana Birlutiu** and **Tom Heskes**[1]

Abstract. In the last decades enormous advances have been made possible for modelling complex (physical) systems by mathematical equations and computer algorithms. To deal with very long running times of such models a promising approach has been to replace them by stochastic approximations based on a few model evaluations. In this paper we focus on the often occuring case that the system modelled has two types of inputs $\boldsymbol{x} = (\boldsymbol{x}_c, \boldsymbol{x}_e)$ with $\boldsymbol{x}_c$ representing control variables and $\boldsymbol{x}_e$ representing environmental variables. Typically, $\boldsymbol{x}_c$ needs to be optimised, whereas $\boldsymbol{x}_e$ are uncontrollable but are assumed to adhere to some distribution. In this paper we use a Bayesian approach to address this problem: we specify a prior distribution on the underlying function using a Gaussian process and use Bayesian Monte Carlo to obtain the objective function by integrating out environmental variables. Furthermore, we empirically evaluate several active learning criteria that were developed for the deterministic case (i.e., no environmental variables) and show that the ALC criterion appears significantly better than expected improvement and random selection.

1 Introduction

Optimisation of expensive functions is one of the core problems in many of the most challenging problems in computing. Mathematical computer models are frequently used to explore the design space to reduce the need for expensive hardware prototypes, but are often hampered by very long running times. Much emphasis has therefore been on optimising a model using as few function evaluations as possible. A very promising approach has been to develop a stochastic approximation of the expensive function to optimise – a surrogate model – and use that approximation as replacement in optimisation and to determine the next best function value to evaluate according to some criteria in model fitting. This approach is well known as *response surface modelling* [11, 9].

In this paper we consider a situation often observed in practice in which there are two types of input variables: $\boldsymbol{x} = (\boldsymbol{x}_c, \boldsymbol{x}_e)$ with $\boldsymbol{x}_c$ a set of *control* variables and $\boldsymbol{x}_e$ a set of *environmental* variables. The control variables are the variables that we can control whereas the environmental variables are assumed to have values governed by some distribution that we cannot manipulate. For example, in [3, 2] a hip prosthesis is designed where the control variables specify its shape and the environmental variables account for the variability in patient population like bone density and activity. In [29] a VLSI circuit is designed where the control variables are the widths of six transistors and the environmental variables are qualitative indicators. In [12] a compressor blade design is improved where the control variables specify the geometry of the blade and the environmental variables are manufacturing variations in chord, camber, and thickness.

In this article we focus on optimising a real-valued objective function that only depends on the control variables, but its value for each setting of the control variables is the mean over the distribution of the environmental variables. Hence, we seek to optimise the control variables in order to obtain the best average response of the objective function over the distribution of environmental variables

$$\boldsymbol{x}_c^* = \operatorname*{argmax}_{\boldsymbol{x}_c} \ell(\boldsymbol{x}_c) = \operatorname*{argmax}_{\boldsymbol{x}_c} \int_{\boldsymbol{x}_e} f(\boldsymbol{x}_c, \boldsymbol{x}_e) p(\boldsymbol{x}_e) \,\mathrm{d}\boldsymbol{x}_e \qquad (1)$$

with f some real-valued utility function and $p(\cdot)$ some known measure over the environmental variables $\boldsymbol{x}_e$. In particular, we focus on the problem of active learning in this context – how to choose the ith sample point as a function of the sample points seen so far in order to obtain a good prediction for $\boldsymbol{x}_c^*$ using as few function evaluations of f as possible.

Our contribution is a computational framework for optimising functions that depend on both control and environmental variables. We describe in detail how the problem can be addressed by integrating Gaussian processes, Bayesian Monte Carlo, and active learning criteria. Additionally, we empirically validate several well-known active learning criteria on a cake mix case study and show that the ALC criterion appears significantly better than expected improvement and random selection.

The rest of the paper is structured as follows. Section 2 describes some background. Section 3 describes the framework we use to address the problem formulated in Eq. (1) step-by-step: integrating out environmental variables using Bayesian Monte Carlo to obtain a stochastic approximation to the objective function (Section 3.1), reformulating the optimisation problem in term of the stochastic approximation (Section 3.2), and active learning criteria for efficiently finding the maximum of the stochastic approximation (Section 3.3). Section 4 gives empirical results of our approach. Section 5 describes related work. Section 6 gives conclusions.

Notation. Boldface notation is used for vectors and matrices. Normal fonts and subscripts are used for the components of vectors and matrices or scalars. The notation $\mathcal{N}(\boldsymbol{\theta}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is used for a multivariate Gaussian with mean $\boldsymbol{\mu}$ and variance $\boldsymbol{\Sigma}$. The transpose of a matrix $\boldsymbol{M}$ is denoted by $\boldsymbol{M}^T$. The zero vector and identity matrix are denoted by $\boldsymbol{0}$ and $\boldsymbol{I}$, respectively. We use f to denote a function that depends on both control and environmental variables, and h to denote a deterministic function, i.e., it only depends on control variables.

[1] Radboud University Nijmegen, Institute for Computing and Information Sciences, the Netherlands, email: pcgroot@gmail.com

2 Background

Section 2.1 describes Gaussian process regression. Section 2.2 describes Bayesian Monte Carlo, which is a Bayesian approach for evaluating integrals using a Gaussian process to specify a prior distribution over functions.

2.1 Gaussian process regression

To simplify notation, we don't make a distinction between control and environmental variables at this point. Let $\boldsymbol{x} \in \mathbb{R}^N$ be an input point, $y \in \mathbb{R}$ an output point. Let $\mathcal{D}_n = \{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)\}$ be a set of n input-output pairs. Let $X = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ and let $Y = \{y_1, \ldots, y_n\}$ be the set of inputs and outputs, respectively, occurring in $\mathcal{D}_n$. We assume that $\mathcal{D}_n$ is generated by an unknown function $h : \mathbb{R}^N \to \mathbb{R}$ and the goal is to learn h given $\mathcal{D}_n$.

To learn h we model h using a zero mean Gaussian process (GP), $h \sim \mathcal{GP}(\boldsymbol{0}, \boldsymbol{K})$, which defines a prior distribution over functions. The covariance matrix $\boldsymbol{K}$ is given by a kernel function k. For example, the quadratic exponential covariance function is defined as

$$\begin{aligned} K_{ij} &= \operatorname{cov}(h(\boldsymbol{x}_i), h(\boldsymbol{x}_j)) = k(\boldsymbol{x}_i, \boldsymbol{x}_j) \\ &= w_0 \exp\left(-\frac{1}{2}(\boldsymbol{x}_i - \boldsymbol{x}_j)^T \boldsymbol{A}^{-1}(\boldsymbol{x}_i - \boldsymbol{x}_j)\right) \end{aligned} \tag{2}$$

with $\boldsymbol{A} = \operatorname{diag}(w_1^2, \ldots, w_N^2)$ and w_i hyperparameters. Given a GP prior over functions $h \sim \mathcal{GP}(\boldsymbol{0}, \boldsymbol{K})$ and a set of observations $\mathcal{D}_n$, a posterior distribution $p(h|\mathcal{D}_n)$ can computed that can be used to make predictions at new test points $\boldsymbol{x}$, $\boldsymbol{x}'$. The standard predictive equations for GP regression are given by [24]:

$$\begin{aligned} \overline{h}_{\mathcal{D}_n}(\boldsymbol{x}) &= k(\boldsymbol{x}, X)\boldsymbol{Q}^{-1}Y \\ \operatorname{cov}_{\mathcal{D}_n}(h(\boldsymbol{x}), h(\boldsymbol{x}')) &= k(\boldsymbol{x}, \boldsymbol{x}') - k(\boldsymbol{x}, X)\boldsymbol{Q}^{-1}k(X, \boldsymbol{x}') \end{aligned} \tag{3}$$

with $\boldsymbol{Q} = (\boldsymbol{K} + \sigma_n^2 \boldsymbol{I})$ the kernel matrix with a tiny constant added to its diagonal in order to improve numerical stability.

A 1-D illustration of GP regression is shown in Figure 1, left panel. The true function $h(x) = \sin(x) + \frac{1}{3}x$ (dashed line) is approximated with a GP using four sample observations (dots). The solid line is the GP mean function $\overline{h}_{\mathcal{D}_n}$ and the two standard pointwise error bars are obtained from $\operatorname{cov}_{\mathcal{D}_n}(h)$ given in Eq. (3).

2.2 Bayesian Monte Carlo

In practice, evaluating a function h is often *expensive*, meaning that we are able to only obtain a small number of function evaluations. This leads to uncertainty about h because of incomplete knowledge. Furthermore, often we are not interested in h, but in evaluating the integral $H = \int_{\boldsymbol{x}} h(\boldsymbol{x})p(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$ (with respect to some measure $p(\boldsymbol{x})$ denoting the importance of the inputs). Because of the uncertainty in h, determining H can be considered an inference problem [20].

The Bayesian Monte Carlo (BMC) method is a Bayesian approach for evaluating integrals [23]. BMC starts with defining a prior distribution over h and updates this distribution using a set of n observations $\mathcal{D}_n = \{(\boldsymbol{x}_i, y_i)\}_{i=1,\ldots,n}$ to obtain a posterior distribution $p(h|\mathcal{D}_n)$. When h is modelled with a GP prior and the posterior $p(h|\mathcal{D}_n)$ is or can be approximated with an (infinite-dimensional joint) Gaussian, the distribution of H has a Gaussian distribution, $H \sim \mathcal{N}(\overline{H}, \operatorname{cov}(H))$, and is fully characterised by its mean and variance [23]

$$\begin{aligned} \overline{H} &= \int_{\boldsymbol{x}} \overline{h}_{\mathcal{D}_n}(\boldsymbol{x})p(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x} \\ \operatorname{cov}(H) &= \int_{\boldsymbol{x}}\int_{\boldsymbol{x}'} \operatorname{cov}(h_{\mathcal{D}_n}(\boldsymbol{x}), h_{\mathcal{D}_n}(\boldsymbol{x}'))p(\boldsymbol{x})p(\boldsymbol{x}')\,\mathrm{d}\boldsymbol{x}\,\mathrm{d}\boldsymbol{x}' \end{aligned} \tag{4}$$

with $\overline{h}_{\mathcal{D}_n}(\boldsymbol{x})$ and $\operatorname{cov}(h_{\mathcal{D}_n}(\boldsymbol{x}), h_{\mathcal{D}_n}(\boldsymbol{x}'))$ the posterior mean and posterior variance, respectively, as given in Eq. (3). The integrals in Eq. (4) can be reformulated as follows

$$\overline{H} = \boldsymbol{z}\boldsymbol{Q}^{-1}Y, \qquad \operatorname{cov}(H) = c - \boldsymbol{z}\boldsymbol{Q}^{-1}\boldsymbol{z}^T \tag{5}$$

where we used the following integrals

$$\begin{aligned} c &= \int_{\boldsymbol{x}} p(\boldsymbol{x}) \int_{\boldsymbol{x}'} p(\boldsymbol{x}')k(\boldsymbol{x}, \boldsymbol{x}')\,\mathrm{d}\boldsymbol{x}\,\mathrm{d}\boldsymbol{x}' \\ z_l &= \int_{\boldsymbol{x}} p(\boldsymbol{x})k(\boldsymbol{x}, \boldsymbol{x}_l)\,\mathrm{d}\boldsymbol{x} \end{aligned} \tag{6}$$

with k the kernel function and $\boldsymbol{x}_l \in X$ the l-th input point in the data set. Both c and z_l depend on the data as the kernel function k can have a number of hyperparameters that are optimised with respect to the data (cf. Eq. (2)).

In some cases these multi-dimensional integrals can be reduced to products of one dimensional integrals, which are usually easier to solve. If the density $p(\boldsymbol{x})$ and the kernel function are both Gaussian we obtain analytic expressions. In particular, when $p(\boldsymbol{x}) \sim \mathcal{N}(\boldsymbol{b}, \boldsymbol{B})$ and when using the common quadratic exponential covariance function in Eq. (2) we obtain the following analytical expressions [23]:

$$\begin{aligned} c &= w_0|2\boldsymbol{A}^{-1}\boldsymbol{B} + \boldsymbol{I}|^{-1/2} \\ z_l &= w_0|\boldsymbol{A}^{-1}\boldsymbol{B} + \boldsymbol{I}|^{-1/2}\cdot \\ &\quad \exp\left(-\frac{1}{2}(\boldsymbol{x}_l - \boldsymbol{b})^T(\boldsymbol{A} + \boldsymbol{B})^{-1}(\boldsymbol{x}_l - \boldsymbol{b})\right) \end{aligned} \tag{7}$$

Some other choices that lead to analytical expressions are Gaussian mixtures for $p(\boldsymbol{x})$ and polynomial kernels.

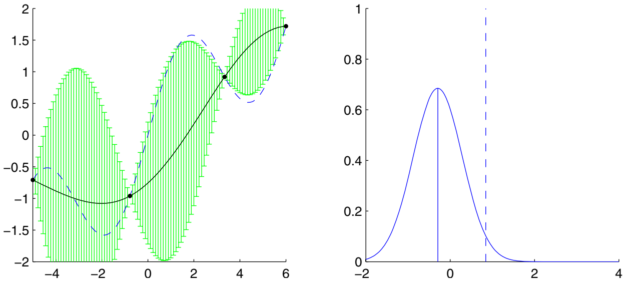

Figure 1. Left: Gaussian process regression. The GP mean prediction (solid line) of the function $h(x) = \sin(x) + \frac{1}{3}x$ (dashed line) with two standard error pointwise error bars after four observations (dots). Right: Bayesian Monte Carlo. The normal distribution representing $\int_x h(x)p(x)\,\mathrm{d}x$ with $p(x) \sim \mathcal{N}(1, 1)$ and the true integral value represented by a dashed line.

A 1-D illustration of BMC is shown in Figure 1. On the left we have a GP fit of the function $h(x) = \sin(x) + \frac{1}{3}x$. On the right we have the corresponding Gaussian distribution for $\int_x h(x)p(x)\,\mathrm{d}x$ with $p(x) \sim \mathcal{N}(1, 1)$ calculated using BMC. True values are shown with a dashed line, approximations with a solid line. By obtaining more function evaluations for h, the GP fit will improve and the Gaussian predictive distribution for the integral will become more peaked and will converge to the true value.

3 Framework

Below we describe our approach step-by-step to address the problem formulated in Eq. (1).

3.1 Integrating out environmental variables

In the rest of the paper we reintroduce the distinction between control and noise variables $\boldsymbol{x} = (\boldsymbol{x}_c, \boldsymbol{x}_e)$. We consider the case where we only integrate out $\boldsymbol{x}_e$ from $f(\boldsymbol{x}_c, \boldsymbol{x}_e)$ using the BMC method described above. Because of uncertainty about f, we model $f(\boldsymbol{x}_c, \boldsymbol{x}_e)$ with a Gaussian process (Section 2.1) and given data $\mathcal{D}_n$ will write it as $f(\boldsymbol{x}_c, \boldsymbol{x}_e|\mathcal{D}_n)$. Using BMC (Section 2.2) to integrate out $\boldsymbol{x}_e$ from $f(\boldsymbol{x}_c, \boldsymbol{x}_e|\mathcal{D}_n)$ we obtain a stochastic approximation L to our objective function ℓ (Eq. (1)). The stochastic objective function L is described by a collection of random variables $L(\boldsymbol{x}_c)$:

$$L(\boldsymbol{x}_c|\mathcal{D}_n) = \int_{\boldsymbol{x}_e} f(\boldsymbol{x}_c, \boldsymbol{x}_e|\mathcal{D}_n) p(\boldsymbol{x}_e)\, \mathrm{d}\boldsymbol{x}_e \tag{8}$$

for which we assume a Gaussian measure $p(\boldsymbol{x}_e) \sim \mathcal{N}(\boldsymbol{x}_e|\boldsymbol{b}, \boldsymbol{B})$ on inputs $\boldsymbol{x}_e$. When we model f using a GP with a kernel function defined in Eq. (2) it follows from Eq. (5) that L is a GP with a mean and covariance function defined by

$$\begin{aligned} \overline{L}(\boldsymbol{x}_c) &= \boldsymbol{z}(\boldsymbol{x}_c)\boldsymbol{Q}^{-1}Y \\ \mathrm{cov}(L(\boldsymbol{x}_c), L(\boldsymbol{x}'_c)) &= c(\boldsymbol{x}_c, \boldsymbol{x}'_c) - \boldsymbol{z}(\boldsymbol{x}_c)\boldsymbol{Q}^{-1}\boldsymbol{z}(\boldsymbol{x}'_c)^T \end{aligned} \tag{9}$$

where we omitted the dependence on $\mathcal{D}_n$ for readability and used the shorthand notation $\boldsymbol{z}(\boldsymbol{x}_c)_l = w_0^{-1} k_c(\boldsymbol{x}_c, \boldsymbol{x}_{c,l})\boldsymbol{z}_l$ and $c(\boldsymbol{x}_c, \boldsymbol{x}'_c) = w_0^{-1} k_c(\boldsymbol{x}_c, \boldsymbol{x}'_c)c$, which follows from Eq. (6) and the fact that the kernel function k factorises, i.e.,

$$k((\boldsymbol{x}_c, \boldsymbol{x}_e), (\boldsymbol{x}'_c, \boldsymbol{x}'_e)) = w_0^{-1} k_c(\boldsymbol{x}_c, \boldsymbol{x}'_c) k_e(\boldsymbol{x}_e, \boldsymbol{x}'_e) \tag{10}$$

with k_c, k_e the kernel function k restricted to the domain of $\boldsymbol{x}_c$ and $\boldsymbol{x}_e$, respectively.

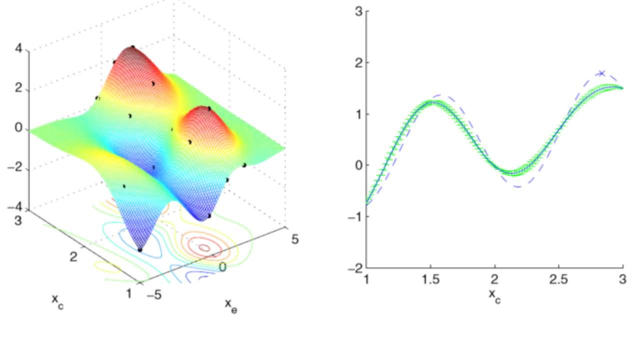

Figure 2. Left: GP mean prediction of the function $f(x_c, x_e) = \sin(x_e) + \frac{1}{3}x_e + \sin(5x_c) + \frac{1}{3}x_c - 1$ given a small number of observations (dots). Right: GP prediction of $\int_{x_e} f(x_c, x_e)p(x_e)\, \mathrm{d}x_e$ with $p(x) \sim \mathcal{N}(x; 1, 1)$ and the true function shown by a dashed line computed using numerical integration.

A 2-D illustration of integrating out environmental variables using the BMC approach is shown in Figure 2. On the left we have the mean GP fit of $f(x_c, x_e) = \sin(x_e) + \frac{1}{3}x_e + \sin(5x_c) + \frac{1}{3}x_c - 1$ with $1 \leq x_c \leq 3$ and $-5 \leq x_e \leq 5$. On the right we have the GP fit for $\int_{x_e} f(x_c, x_e)p(x_e)\, \mathrm{d}x_e$ using BMC. By obtaining more function evaluations for f, the GP fit shown on the left and right will improve.

3.2 Optimisation

So far, we have modelled our objective function ℓ with a Gaussian process L. The goal, however, is to find the value $\boldsymbol{x}_c^*$ such that $\ell(\boldsymbol{x}_c^*)$ is maximised (cf. Eq. (1)) as illustrated by the small cross in Figure 2, right panel. The idea is to request more information about the true objective function ℓ (through f), update our stochastic approximation L, and use the resulting model to make a prediction:

$$\begin{aligned} \tilde{\boldsymbol{x}}_c^* &= \operatorname*{argmax}_{\boldsymbol{x}_c} \overline{L}(\boldsymbol{x}_c|\mathcal{D}_n) \\ &= \operatorname*{argmax}_{\boldsymbol{x}_c} \int_{\boldsymbol{x}_e} \overline{f}(\boldsymbol{x}_c, \boldsymbol{x}_e|\mathcal{D}_n) p(\boldsymbol{x}_e)\, \mathrm{d}\boldsymbol{x}_e. \end{aligned} \tag{11}$$

This problem formulation is quite different from earlier work on optimising expensive functions. Previous work is typically of the form shown in Figure 1, left panel. Our work is of the form shown in Figure 2, right panel. There are two key aspects that distinguishes our problem formulation from previous work. First, we do not optimise f, but the average that f takes over a distribution of the environment variables (i.e., the difference between left and right panels in Figures 1 and 2). Second, our objective function L is a collection of stochastic variables which is only observed indirectly through $f(\boldsymbol{x}_c, \boldsymbol{x}_e)$ whereas previous work almost exclusively focusses on optimising a *deterministic* function h that is directly observed through observations $h(\boldsymbol{x})$ (i.e., the difference between Figures 1 and 2).

Since f is expensive to evaluate, we would like to select function evaluations of f in such a way that the $\tilde{\boldsymbol{x}}_c^*$ obtained in Eq. (11) results in a value $\ell(\tilde{\boldsymbol{x}}_c^*)$ that is close to the global optimum $\ell(\boldsymbol{x}_c^*)$ using as few function evaluations as possible. This is known in the literature as active learning or infill sampling criteria. Below we describe several active learning criteria.

3.3 Active learning

Much work has already been done on optimising expensive functions by optimising a Gaussian process based surrogate model (see [28, 15] for a detailed overview). Below we describe some well-known criteria for active learning that are applicable to our problem formulation before empirically validating them in Section 4. The criteria can be split into two categories: (1) criteria that are specifically geared towards finding a maximum of a function, but not necessarily a good global model fit, and (2) criteria that improve the global model fit and thereby indirectly also the predicted maximum value, which we describe in Sections 3.3.1 and 3.3.2, respectively. Furthermore, in Section 3.3.3, we propose extensions of these criteria that are applicable to our problem formulation.

3.3.1 Criteria for obtaining a maximum

Expected Improvement. One of the early influential papers is the work by Jones *et al.* [11] who studied the problem of finding a maximum of a *deterministic* function using a GP (i.e., finding the maximum of the true underlying function as shown in Figure 1, left panel, by a dashed line). Based on the currently best observed value $y_{\max} = \max\{h(\boldsymbol{x}_1), \ldots, h(\boldsymbol{x}_n)\}$ given n observed function evaluations $\{h(\boldsymbol{x}_i)\}_{i=1,\ldots,n}$ Jones *et al.* define the improvement $I(\hat{\boldsymbol{x}})$ at a new input $\hat{\boldsymbol{x}}$ as

$$I(\hat{\boldsymbol{x}}) = \max\{0, h(\hat{\boldsymbol{x}}) - y_{\max}\} \tag{12}$$

Of course, this value cannot be computed as h is unknown, but the expected improvement $E[I(\hat{\boldsymbol{x}})]$ can be computed using the GP model for h. The expected improvement can be used as infill criteria by requesting a new observation at the location where the $E[I(\hat{\boldsymbol{x}})]$ obtains its maximum value. When compared with work on expected improvement, in our work the known value $y_{\max}$ is replaced by a probabilistic value obtained from L.

Generalised Expected Improvement. As the expected improvement criteria was found to often get stuck in local optima, a generalisation was proposed in [26] that introduces a parameter that controls the local-global balance. Let $L^n_{\max} = \max_{\boldsymbol{x}_c}\{\overline{L}(\boldsymbol{x}_c|\mathcal{D}_n)\}$ be the maximum (over the means) of the predicted values of our objective function given the n data samples collected so far.[2] As the objective function is a Gaussian process, the predictive distribution in a new point is Gaussian distributed, i.e., $L(\hat{\boldsymbol{x}}_c) \sim \mathcal{N}(m(\hat{\boldsymbol{x}}_c), s^2(\hat{\boldsymbol{x}}_c))$. For readability the dependence on $\hat{\boldsymbol{x}}_c$ is left out, denoting $m(\hat{\boldsymbol{x}}_c)$ as m and $s^2(\hat{\boldsymbol{x}}_c)$ as s^2. The generalised improvement [26] over the current best value is defined as

$$I(\hat{\boldsymbol{x}}_c)^g = \begin{cases} (\ell(\hat{\boldsymbol{x}}_c) - L^n_{\max})^g & \text{if } \ell(\hat{\boldsymbol{x}}_c) > L^n_{\max} \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

with g a non-negative parameter controlling the local-global balance.

Analogously to the improvement function of Jones *et al.* (cf. Section 3.3), this value cannot be computed as ℓ is unknown, but the expectation of the generalised improvement can be computed by using the GP predictive distribution at $L(\hat{\boldsymbol{x}}_c)$ for $\ell(\hat{\boldsymbol{x}}_c)$. The expected generalised improvement can be shown to take the following form

$$E[I(\hat{\boldsymbol{x}}_c)^g] = \sum_{k=0}^{g} \binom{g}{k} s^g (-u)^{g-k} T_k \tag{14}$$

with $u = (L^n_{\max} - m)/s$ and where

$$T_0 = 1 - \Phi(u) \text{ and } T_1 = \phi(u) \tag{15}$$

with Φ the standard normal cumulative distribution function and ϕ the standard normal probability density function. Each T_k for $k > 1$ can be computed recursively from

$$T_k = u^{k-1}\phi(u) + (k-1)T_{k-2} \tag{16}$$

Higher values of g result in more global search. The standard (expected) improvement function uses $g = 1$.

3.3.2 Criteria for obtaining a global model fit

Variance reduction. We consider two other active learning criteria based on variance reduction. The first method denoted ALM, developed by MacKay [16], maximises the expected information gain about parameter values of the model by selecting data where the predictor has maximum variance. This is directly applicable to a Gaussian process as it provides a variance estimate for each test point (cf. Eq. (3)). The second method denoted ALC, developed by Cohn [4], is motivated from the goal of minimising the generalisation error. It computes how the output variance of the predictor changes (averaged over a set of reference data points Λ) when a new test point $\hat{\boldsymbol{x}}$ would be added to the data set. Formally,

$$\Delta\sigma^2_\lambda(\hat{\boldsymbol{x}}) = \frac{(K(X,\lambda)\boldsymbol{K}^{-1}\boldsymbol{m} - K(\hat{\boldsymbol{x}},\lambda))^2}{K(\hat{\boldsymbol{x}},\hat{\boldsymbol{x}}) - \boldsymbol{m}^T\boldsymbol{K}^{-1}\boldsymbol{m}} \tag{17}$$

with $\boldsymbol{m} = K(X,\hat{\boldsymbol{x}})$, $\boldsymbol{K}^{-1} = K(X,X)^{-1}$, and $\lambda \in \Lambda$. In [27], both methods are compared on the average variance and mean-squared error and ALC was found to consistently perform better (but much harder to evaluate) than ALM and random selection.

Latin Hypercube Sampling. A k-dimensional Latin Hypercube Design (LHD) [17, 8] is a design of n points $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ with $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{ik})$ such that for each dimension j, $1 \leq j \leq k$, all x_{ij}, $i = 1, \ldots, n$ are distinct. In the literature, LHDs are typically used to initialise the statistical model, before switching to an active learning criterion. Note that LHDs choose a design beforehand, without using any information about the acquired samples so far.

3.3.3 Combined criteria

The generalised expected improvement criterion will result in a new point $\hat{\boldsymbol{x}}_c$, to be used to request more observations about the objective function ℓ. Observations about ℓ, however, can only be obtained by evaluating $f(\boldsymbol{x}_c, \boldsymbol{x}_e)$. Hence, in the context of this paper, the generalised expected improvement criterion needs to be extended to obtain a pair $(\hat{\boldsymbol{x}}_c, \hat{\boldsymbol{x}}_e)$ for the function f to be evaluated at. In this paper, we combine the generalised improvement criterion with the ALC criterion of Section 3.3.2: we apply the generalised expected improvement on $L(\boldsymbol{x}_c)$ to obtain $\hat{\boldsymbol{x}}_c$ followed by ALC on $f(\hat{\boldsymbol{x}}_c, \boldsymbol{x}_e)$, i.e., with $\hat{\boldsymbol{x}}_c$ fixed, to obtain $\hat{\boldsymbol{x}}_e$.[3] This extends the generalised expected improvement, which aims at finding a maximum, to our case of functions dependent on both control and environmental variables. Analogously, any of the global criteria of Section 3.3.2 can be used on the integrated objective function $\ell(\boldsymbol{x}_c)$ to obtain $\hat{\boldsymbol{x}}_c$ and then combined with another criterion on the function $f(\hat{\boldsymbol{x}}_c, \boldsymbol{x}_e)$ with fixed $\hat{\boldsymbol{x}}_c$. In this paper, we only consider the ALC criterion combined with itself, denoted ALC-ALC. Thus, ALC uses Eq. (17) with the covariance function K as defined in Eq. (2) whereas ALC-ALC uses Eq. (17) with the covariance function K as defined in Eq. (9), resulting in a $\hat{\boldsymbol{x}}_c$, followed by the ALC criterion on $f(\hat{\boldsymbol{x}}_c, \boldsymbol{x}_e)$.

4 Experiments

The following case study is taken from [1]. Suppose we were to introduce a new cake mix into the consumer market that we like to be robust against inaccurate settings of oven temperature (T) and baking time (t). We would like to design experiments varying the control variables – the amount of flour (F), the amount of sugar (S), and the amount of egg powder (E) – and environmental variables (oven temperature and baking time) to see if we could create a cake mix that is better with respect to the environmental variables than the standard recipe so far produced by the product development laboratory.

Given a number of data samples we fitted a Gaussian process and used its mean function as the true underlying model. We used the

[2] In [26] the criteria is defined for a deterministic function and $L^n_{\max}$ is defined to be the maximum over the known n observed values.

[3] In this paper we only combine with ALC as this criterion turned out to give better performance than the ALM and random criteria.

same hyperparameters of this model in the experiments. We set

$$p \sim \mathcal{N}(\boldsymbol{b}, \boldsymbol{B}), \quad \boldsymbol{b} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \boldsymbol{B} = \begin{bmatrix} 2/3 & -1/3 \\ -1/3 & 2/3 \end{bmatrix} \tag{18}$$

based on the variance observed in the data and assuming a negative correlation between oven temperature and baking time. We used the divided rectangles (DIRECT) algorithm [6, 10] as global optimiser and the maximum value of the corresponding function ℓ as defined in Eq. (1) was found to be 5.5330 and was obtained at $\boldsymbol{x}_c^* = (F^*, S^*, E^*) = (1.1852, -0.7407, 1.1084)$, which implies an improved cake mix by using a higher amount of flour, a lower amount of sugar, and a higher amount of egg powder than the standard recipe set at $(0, 0, 0)$.

The goal of the various active learning criteria is to find the value $\boldsymbol{x}_c^*$ that maximises ℓ as quickly as possible using properties of the stochastic approximation L. Therefore, let $\tilde{\boldsymbol{x}}_c^*$ be the value that maximises $\overline{L}$ the current mean GP estimate of ℓ. We take as error measure ϵ the distance between the true maximum value and the true value at the predicted location $\tilde{\boldsymbol{x}}_c^*$:[4]

$$\epsilon = |\max_{\boldsymbol{x}_c} \ell(\boldsymbol{x}_c) - \ell(\tilde{\boldsymbol{x}}_c^*)|, \qquad \tilde{\boldsymbol{x}}_c^* = \operatorname*{argmax}_{\boldsymbol{x}_c} \overline{L}(\boldsymbol{x}_c) \tag{19}$$

We first evaluated the random, ALM, ALC, and LHD criterion on the cake mix case study. Besides LHD, we started for each active learning criteria from a random initial sample and iteratively selected new samples according to the criteria. At each iteration we updated the model and computed the error measure given in Eq. (19). We iteratively selected up to 50 samples and averaged the results over 50 runs. The set of random initial starting points were the same for each of the random, ALM, and ALC active learning criteria. Because of the computational complexity of the ALC criterion we limited the set Λ to 500 reference samples that were drawn according to the distribution specified by p on the environmental variables and uniform distribution on the control variables.

The results are shown in Figure 3 in which we plot the error measure from Eq. (19) and the standard deviation of the mean over 50 runs. Clearly, the ALM method performs is even worse than random sampling. The LHD approach performs better than ALM, but its performance is initially very similar to random sampling and after about 25 samples it is even outperformed by random sampling. Although LHDs are typically used as initialisation method in the literature these results suggest that an LHD is unnecessary and may lead to worse performance. Similar results for LHDs and deterministic functions have also been reported recently in [15]. The ALC criterion performs very well on the cake mix study. The downside, however, is that ALC is computationally more challenging and tries to optimise the global model fit, but not specifically the predicted maximum of the objective function. The ALC-ALC criterion is a bit worse than the ALC criterion, but performs quite well. It has the advantage that optimisation in a high dimensional space of both control and environmental variables can be split into two sequential optimisation steps in two lower dimensional spaces.

Besides evaluating the active learning criteria that are aimed at improving the global model fit we also evaluated the generalised expected improvement criterion which aims at finding the maximum. As already mentioned in Section 3.3.1 we used the generalised expected improvement to obtain an $\hat{\boldsymbol{x}}_c$ which was then kept fixed in one of the criteria that aim for a global fit to obtain a pair $(\hat{\boldsymbol{x}}_c, \hat{\boldsymbol{x}}_e)$ for further evaluation. We only investigated the combination of the generalised expected improvement criterion with the ALC criterion as ALC clearly outperformed the random, ALM, and LHD criteria.

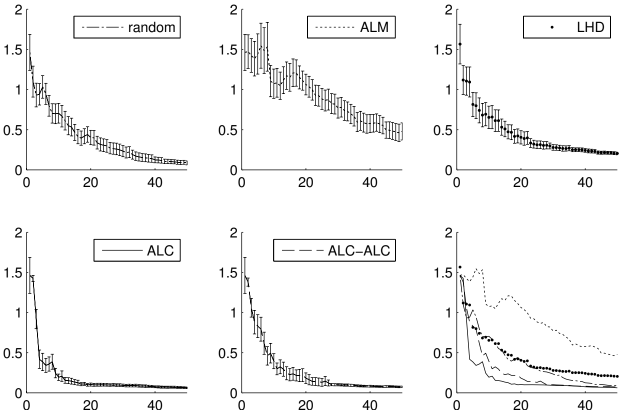

Figure 3. For each active learning criteria we computed a sequence of samples and observations to be added. At each step we computed the distance between the true maximum value and the value at the location where we predict the maximum value to be. We computed the mean performance and standard deviation of the mean of each active learning criterion over 50 runs. The bottom right subfigure superimposes the means.

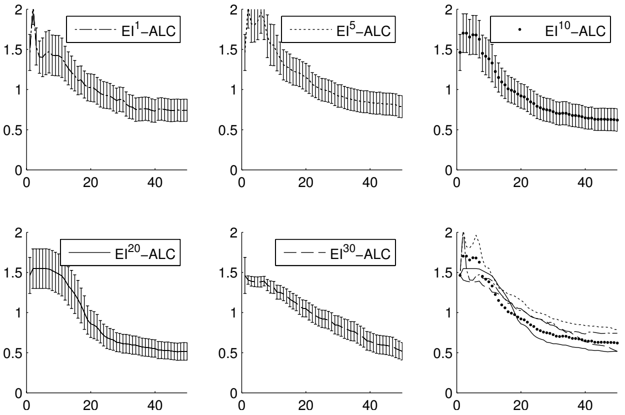

Figure 4. The results of the generalised expectation criterion combined with the ALC variance reduction strategy. For each active learning criteria we computed a sequence of samples and observations to be added. At each step we computed the distance between the true maximum value and the value at the location where we predict the maximum value to be. We computed the mean performance and standard deviation of the mean of each active learning criteria over 50 runs. The bottom right subfigure superimposes the means.

The results are shown in Figure 4. For the cake mix case study, the results of the generalised expected improvement criterion are pretty bad when compared to the results shown in Figure 3. In all cases evaluated, the generalised expected improvement criterion is outperformed by random sampling. The generalised expected improvement criterion has originally been developed for deterministic functions and these results show that the criterion cannot easily be augmented to be used for the optimisation of functions that are dependent on both control variables and environmental variables.

[4] Alternatively, if there is only one dominating global optimum, one can take as error measure the distance $\|\boldsymbol{x}_c^* - \tilde{\boldsymbol{x}}_c^*\|$. In our case study, however, there are multiple local optima that are almost as good as the global optimum.

5 Related work

Optimisation of expensive *deterministic* functions (which may include noisy observations) using response surfaces is an active field of research. Recently, further developments of the theory have appeared (e.g., multi-step lookahead, use of derivative information [21]) as well as some new domains of application (e.g., robotics [14, 13] and controllers [7]). Designing better infill criteria is also still an active topic, e.g., [26, 25, 22, 5].

Less work has been done in the area of optimizing functions dependent on both control and environmental variables. The earliest ideas can be contributed to Genichi Taguchi in the 1980s who coined the term *robust parameter design*, but their inefficiency have often been criticised [18]. Recently, some progress has been made using response surfaces applied to integrated objective functions, but restricted to finite measures on the environmental variables [19, 30]. The current paper extends this work to Gaussian measures.

We showed that the well-known generalized expected improvement criterion performed badly on the case study investigated and that the ALC criterion performed quite well. Nevertheless, there is room for further improvement. The authors are unaware of active learning criteria specifically designed for the type of problems considered in this paper.

6 Conclusions and future work

In this paper we demonstrated a step-by-step approach for optimising functions that depend on both control and environmental variables. We described in detail how the problem can be addressed by integrating Gaussian processes, Bayesian Monte Carlo, and active learning criteria. Furthermore, we empirically validated several well-known active learning criteria on a cake mix case study.

An issue for further research is the design of better active learning criteria as the expected improvement criteria, which is often advocated in this field for deterministic functions, performed quite badly. For example, we could and probably should take into account the variance of $L^n_{\max}$ in the generalised expected improvement criterion. Other issues wanting further investigation is the scalability of the approach in terms of control and environmental input dimensions as well as the use of Gaussian mixtures as distributions over environmental variables.

ACKNOWLEDGEMENTS

We thank the anonymous referees for their comments. The current research was funded by STW project 07605 (HearClip) and NWO VICI grant 639.023.604.

REFERENCES

[1] G. Box, S. Bisgaard, and C. Fung, 'An explanation and critique of Taguchi's contributions to quality engineering', *Quality and Reliability Engineering International*, **4**, 123–131, (1988).

[2] P.B. Chang, B.J. Williams, K.S.B. Bhalla, T.W. Belknap, T.J. Santner, W.I. Notz, and D.L. Bartel, 'Design and analysis of robust total joint replacements: finite element model experiments with enviromental variables', *J. Biomechanical Engineering*, **123**, 239–246, (2001).

[3] P.B. Chang, B.J. Williams, W.I. Notz, T.J. Santner, and D.L. Bartel, 'Robust optimization of total joint replacements incorporating environmental variables', *J. Biomech. Eng.*, **121**, 304–310, (1999).

[4] D.A. Cohn, 'Neural networks exploration using optimal experimental design', *Neural Networks*, **6**(9), 1071–1083, (1996).

[5] K. Crombecq and D. Gorissen, 'A novel sequential design strategy for global surrogate modeling', in *Proceedings of the 41th Conference on Winter Simulation*, pp. 731–742, (2009).

[6] D.E. Finkel, 'DIRECT optimization algorithm user guide', Technical report, Center for Research in Scientific Computation, North Carolina State University, (2003).

[7] M. Frean and P. Boyle, 'Using Gaussian processes to optimize expensive functions', in *Proceedings of the 21st Australasian Joint Conference on Artificial Intelligence: Advances in Artificial Intelligence*, volume 5360 of *LNAI*, pp. 258–267, (2008).

[8] R.L. Iman, J.M. Davenport, and D.K. Zeigler, 'Latin hypercube sampling (program users guide)', Technical Report SAND79-1473, Sandia National Laboratories, Albuquerque, NM, (1980).

[9] D.R. Jones, 'A taxonomy of global optimization methods based on response surfaces', *Journal of Global Optimization*, **21**, 345–383, (2001).

[10] D.R. Jones, 'The DIRECT global optimization algorithm', in *Encyclopedia of Optimization*, 725–735, Springer, (2009).

[11] D.R. Jones, M. Schonlau, and J.W. Welch, 'Efficient global optimization of expensive black-box functions', *Journal of Global Optimization*, **13**, 455–492, (1998).

[12] A. Kumar, P.B. Nair, A.J. Keane, and S. Shahpar, 'Robust design using Bayesian Monte Carlo', *International Journal for Numerical Methods in Engineering*, **73**, 1497–1517, (2008).

[13] D. Lizotte, T. Wang, M. Bowling, and D. Schuurmans, 'Gaussian process regression for optimization', in *NIPS Workshop on Value of Information*, (2005).

[14] D. Lizotte, T. Wang, M. Bowling, and D. Schuurmans, 'Automatic gait optimization with gaussian process regression', in *International Joint Conference on Artificial Intelligence (IJCAI-07)*, (2007).

[15] D.J. Lizotte, *Practical Bayesian optimization*, Ph.D. dissertation, University of Alberta, 2008.

[16] D.J.C. MacKay, 'Information-based objective functions for active data selection', *Neural Computation*, **4**(4), 589–603, (1992).

[17] M.D. MacKay, R.J. Beckman, and W.J Conover, 'A comparison of three methods for selecting values of input variables in the analysis of output from a computer code', *Technometrics*, **21**(2), 239–245, (1979).

[18] R.H. Myers, A.I. Khuri, and G. Vining, 'Response surface alternatives to the Taguchi robust parameter design approach', *The American Statistician*, **46**(2), 131–139, (1992).

[19] Marin O., *Designing computer experiments to estimate integrated respose functions*, Ph.D. dissertation, The Ohio State University, 2005.

[20] A. O'Hagan, 'Bayes-Hermite quadrature', *Journal of Statistical Planning and Inference*, **29**, 245–260, (1991).

[21] M.A. Osborne, R. Garnett, and S.J. Roberts, 'Gaussian processes for global optimization', in *3rd International Conference on Learning and Intelligent Optimization (LION3)*, Trento, Italy, (January 2009).

[22] W. Ponweiser, T. Wagner, and M. Vincze, 'Clustered multiple generalized expected improvement: A novel infill sampling criterion for surrogate models', in *Congress on Evolutionary Computation*, pp. 3515–3522, (2008).

[23] C.E. Rasmussen and Z. Ghahramani, 'Bayesian Monte Carlo', in *Advances in Neural Information Processing Systems*, volume 15, pp. 505–512. MIT Press, (2003).

[24] C.E. Rasmussen and C.K.I. Williams, *Gaussian Processes for Machine Learning*, MIT Press, Cambridge, MA, 2006.

[25] M.J. Sasena, P. Papalambros, and P. Goovaerts, 'Exploration of metamodeling sampling criteria for constrained global optimization', *Eng. Opt.*, **34**, 263–278, (2002).

[26] M. Schonlau, W.J. Welch, and D.R. Jones, *Global versus local search in constrained optimization of computer models*, volume 34 of *Lecture Notes - Monograph Series*, 11–25, IMS, 1998.

[27] S. Seo, M. Wallat, T. Graepel, and K. Obermayer, 'Gaussian process regression: active data selection and test point rejection', in *Proc. International Joint Conf. Neural Networks*, volume 3, pp. 241–246, (2000).

[28] B. Settles, 'Active learning literature survey', Technical Report Computer Sciences Technical Report 1648, University of Wisconsin-Madison, (January 2009).

[29] W.J. Welch, T.-K. Yu, S.M. Kang, and J. Sacks, 'Computer experiments for quality control by parameter design', *J. Quality Technology*, **22**, 15–22, (1990).

[30] B.J. Williams, T.J. Santner, and W.I. Notz, 'Sequential design of computer experiments to minimize integrated response functions', *Statistica Sinica*, **10**, 1133–1152, (2000).