EMPIRICAL EVALUATION OF BAYESIAN OPTIMIZATION IN PARAMETRIC TUNING OF CHAOTIC SYSTEMS

Mudassar Abbas,[1,*] Alexander Ilin,[1] Antti Solonen,[2] Janne Hakkarainen,[3] Erkki Oja,[1] & Heikki Järvinen[4]

[1]*Dept. of Computer Science, School of Science, Aalto University, Espoo, Finland*

[2]*Lappeenranta University of Technology, Lappeenranta, Finland*

[3]*Finnish Meterological Institute, Helsinki, Finland*

[4]*University of Helsinki, Helsinki, Finland*

Original Manuscript Submitted: 2016; Final Draft Received: 2016

In this work, we consider the Bayesian optimization (BO) approach for parametric tuning of complex chaotic systems. Such problems arise, for instance, in tuning the sub-grid scale parameterizations in weather and climate models. For such problems, the tuning procedure is generally based on a performance metric which measures how well the tuned model fits the data. This tuning is often a computationally expensive task. We show that BO, as a tool for finding the extrema of computationally expensive objective functions, is suitable for such tuning tasks. In the experiments, we consider tuning parameters of two systems: a simplified atmospheric model and a low-dimensional chaotic system. We show that BO is able to tune parameters of both the systems with a low number of objective function evaluations.

KEY WORDS: *Bayesian optimization, chaotic systems, data assimilation, ensemble Kalman filter*

1. INTRODUCTION

In climate and numerical weather prediction (NWP) models, accurate simulation and prediction depends upon the selection of optimal tuning parameters. A typical case is the tuning of closure parameters in climate models which describe processes that are imprecisely modeled due to the restrictive grid used for solving the differential equations (Järvinen et al., 2010; Schirber et al., 2013). Similarly, the tuning of parameters which control the stochastic physics components in ensemble prediction systems is a non-trivial task (Leutbecher and Palmer, 2008). Designing efficient procedures for tuning such model parameters is a topic of active research (see, e.g., Annan and Hargreaves, 2007; Hakkarainen et al., 2013; Hauser et al., 2012; Neelin et al., 2010; Solonen et al., 2012).

The tuning procedure is generally based on a performance metric which measures how well the tuned model fits the data. For example, in numerical weather prediction (NWP), tuning is done by optimizing measures related to forecast skill, while in climate models, tuning is based on optimization criteria which often compare some summary statistics (spatial and temporal averages) of the model simulation to observed statistics. Evaluating the performance metrics is computationally expensive, since it requires complex model simulations over the observation period.

One of the major difficulties of the tuning process is the high computational cost of the optimization procedure: for every candidate value of the tuning parameters, one has to perform computationally heavy simulations of a complex physical model. Another difficulty is that in many cases the objective function is noisy, that is two evaluations of the likelihood function with the same parameter values may generally lead to distinct function values. For example, when the goal is to optimize ensemble prediction systems (see, e.g., Solonen and Järvinen, 2013), a stochastic mechanism

is often used for ensemble member selection, which introduces randomness in the function evaluation. Another possible source of noise is the chaoticity of the tuned model. Small perturbations of the model parameters can result in significantly different simulation trajectories and therefore significant differences in the computed likelihood.

In this paper, we study the methodology called Bayesian optimization (BO) in the problem of parametric tuning of chaotic systems. In BO, the parameter values where the objective function is evaluated are carefully chosen so that we learn as much as possible about the objective function. As a result, the optimum can often be found with a small number of function evaluations. We discuss both deterministic and noisy tuning objectives.

We perform two studies: first, we consider parametric tuning of a simplified atmospheric model with a noiseless objective function. The tuned model is a two-layer quasi-geostrophic model with four tuned parameters which define the model error covariance of the corresponding data assimilation system. Second, we consider parametric tuning of a chaotic system with a noisy likelihood function. We use the parameterized Lorenz 95 model as a test model, similarly to previous studies. The goal is to explore the suitability of the BO methodology for parametric tuning of full scale climate and weather models. In the examples considered in this paper, the likelihood formulation is more relevant for (but not limited to) parametric tuning of NWP systems, as it is essentially built around the accuracy of short-term forecasts.

There are other approaches besides BO to accelerate computations with objective functions that are expensive to evaluate. Various surrogate modeling techniques attempt to describe the parameter-to-output dependence with empirical approximative models that are cheap to evaluate. Techniques range from polynomial chaos expansions (Marzouk and Xiu, 2009) to Gaussian processes (GP) models (Rasmussen and Williams, 2006), which are also applied in the BO method. In BO, instead of first building a surrogate model and then fixing it for further calculations, the goal is to design the points where the objective function is evaluated on the fly so that the potential of the new point in improving the current best value is maximized. That is, BO is directly built for solving optimization problems efficiently, not to represent the objective function efficiently in a selected region of the parameter space.

The BO method resembles classical response surface techniques for experimental optimization (Box and Draper, 1987), where local quadratic models are used to guide a sequence of experiments to obtain an optimal response. BO uses GP, which is more flexible in describing the behavior of the underlying objective. Also, BO uses a different way for selecting the next point where the objective function is evaluated. We use the GP based BO because it has been shown that it is a very efficient and flexible approach (see, e.g., Lizotte et al., 2012), especially, for computationally heavy to compute models (see, e.g., Brochu et al., 2010a).

The outline of the paper is as follows. In Sect. 2, we present the basic ideas behind the Bayesian optimization. In Sect. 3, we formulate the likelihood for a complex system represented as a state-space model. In Sect. 4, we consider the case of parametric tuning of a simplified atmospheric model with a noiseless objective function. In Sect. 5, we demonstrate parametric tuning of a chaotic system with a noisy likelihood. We conclude in Sect. 6.

2. BAYESIAN OPTIMIZATION

The goal of Bayesian optimization is to find the extrema of black-box functions, $f\colon \mathbb{R}^n \to \mathbb{R}$ that are expensive to evaluate (see, e.g., reviews by Brochu et al., 2010a; Snoek et al., 2012). Here, f which is also called the objective function typically does not have a closed form solution. In BO, the objective function is modeled as a random function whose distribution describes our knowledge of the function, given a set of function evaluations $\{\boldsymbol{\theta}_i, f(\boldsymbol{\theta}_i)\}_{i=1,\ldots,t}$. The posterior distribution over f is handled using the standard Gaussian process methodology which allows for evaluating the mean and variance of objective function values $f(\boldsymbol{\theta})$ in any location $\boldsymbol{\theta}$ at any optimization step t. This information is used to propose a new input location $\boldsymbol{\theta}_{t+1}$ which has the largest potential to improve the current best value of the objective function. In the following, we assume that the objective function is being maximized.

The search of the new point where the objective function has to be evaluated is done by optimizing a complementary function called *acquisition function*, that measures the potential to improve the current best point. The two statistics often utilized in designing acquisition functions are the predictive mean and the predictive variance of f at possible location $\boldsymbol{\theta}$. In designing new points where the function is evaluated, one typically has to choose between two extremes: sampling from locations of high predicted mean value (*exploitation* strategy) and locations of high uncertainty value (*exploration* strategy). BO provides a tool that is able to automatically trade off between exploration

and exploitation, which often yields a reduced number of objective function evaluations needed (Lizotte et al., 2012). It can also prove useful for objective functions with multiple local optima, and noise in the objective function can be handled in a straight-forward manner.

Even though BO was introduced in the seventies (Mockus et al., 1978), the methodology has been under active development in the recent years due to its successful application to a number of machine learning problems (Boyle, 2007; Brochu et al., 2010a; Frean and Boyle, 2008; Hutter et al., 2013; Lizotte, 2008; Osborne et al., 2009; Snoek et al., 2012). To our knowledge, BO has not been studied in connection with parameter tuning in complex dynamic models.

BO is a methodology that suits very well to the problem of complex system tuning. First, evaluation of the objective function in this task requires computationally expensive model simulations, typically requires several days to complete. Second, the sampling region of the parameters is often unknown and it is manually selected using expert knowledge. Third, the gradient information is unavailable and direct optimization is infeasible.

2.1 Gaussian processes

The Gaussian processes (GP) methodology is the key element of BO, as it is an elegant tool for describing distributions over unknown functions (Rasmussen and Williams, 2006). In this methodology, the prior distribution over f is chosen such that the function values $\mathbf{f}_\theta = [f(\theta_1), \ldots, f(\theta_t)]$ are assumed to be normally distributed:

$$\mathbf{f}_\theta \mid \boldsymbol{\eta} \;\sim\; \mathcal{N}(\mathbf{f}_\theta|\mathbf{0}, \mathbf{K}_f) \tag{1}$$

where the mean is typically taken to be zero and the covariance matrix $\mathbf{K}_f$ is constructed such that its ijth element is computed using covariance function $k(\theta_i, \theta_j|\boldsymbol{\eta})$ that depends on θ_i, θ_j and hyperparameters $\boldsymbol{\eta}$. The covariance function k is the key element of the GP modeling: it encodes our assumptions about the behavior of function f, such as its smoothness properties.

In our experiments, we use the squared exponential covariance function which is one of the most common covariance functions:

$$k(\theta_i, \theta_j; \boldsymbol{\eta}) = \sigma_f^2 \prod_{d=1}^{D} \exp\left(-\frac{(\theta_i^{(d)} - \theta_j^{(d)})^2}{2\, l_d^2}\right), \tag{2}$$

where l_d are the parameters defining the smoothness of the function in each dimension and σ_f^2 is the scaling parameter which specifies the magnitudes of the function values. Both belong to hyperparameters $\boldsymbol{\eta}$.

At every iteration of the BO algorithm, the properties of the unknown function f are learned by adapting the hyperparameters $\boldsymbol{\eta}$ to fit well the observed data $\{\theta_i, f(\theta_i)\}_{i=1,\ldots,t}$. This is typically done by maximizing the log marginal likelihood of the hyperparameters $\boldsymbol{\eta}$:

$$\log p(\mathbf{f}_n|\theta, \boldsymbol{\eta}) = -\frac{1}{2}\mathbf{f}_n^T(\mathbf{K}_f + \boldsymbol{\Sigma})^{-1}\mathbf{f}_n - \frac{1}{2}\log|\mathbf{K}_f + \boldsymbol{\Sigma}| - \frac{n}{2}\log 2\pi\,. \tag{3}$$

where $\mathbf{f}_n$ are the observed values of the objective function. They are assumed to be noisy such that $\mathbf{f}_n = f(\theta) + \epsilon,\ \epsilon \sim \mathcal{N}(0, \sigma_n^2)$. $\boldsymbol{\Sigma}$ is the covariance matrix of the noise in the objective function, which is often parameterized as $\sigma_n^2\mathbf{I}$ and estimated in the optimization procedure as well. Thus $\boldsymbol{\eta}$ consists of the following hyperparameters $\{\sigma_f, l_d, \sigma_n\}$. Note that σ_n is considered a hyperparameter and optimized only when the observed values of the objective function are noisy. Then, GP is used to evaluate the predictive distribution over the function values $f(\theta_{\text{new}})$ at any new location θ_{new}. Assuming that the observed values of the objective function are noisy and the noise is Gaussian, the predictive distribution is normal:

$$p(f(\theta_{\text{new}})|\mathbf{f}_n, \boldsymbol{\eta}) \sim \mathcal{N}\big(\mu(\theta_{\text{new}}), \sigma^2(\theta_{\text{new}})\big) \tag{4}$$

with the mean and variance given by

$$\mu(\theta_{\text{new}}) = \mathbf{k}_{\text{new}}^T\left(\mathbf{K}_f + \boldsymbol{\Sigma}\right)^{-1}\mathbf{f}_n \tag{5}$$

$$\sigma^2(\theta_{\text{new}}) = k(\theta_{\text{new}}, \theta_{\text{new}}) - \mathbf{k}_{\text{new}}^T\left(\mathbf{K}_f + \boldsymbol{\Sigma}\right)^{-1}\mathbf{k}_{\text{new}}\,, \tag{6}$$

where $\mathbf{k}_{\text{new}} = [k(\theta_{\text{new}}, \theta_1), \ldots, k(\theta_{\text{new}}, \theta_t)]^T$. For more details on training GP, see, for example, the book by Rasmussen and Williams (2006).

2.2 Acquisition functions

The acquisition functions are used to search for a new location θ_{new} which has the highest potential to improve the best value of the objective function obtained so far, denoted by

$$\mu^+ = \max_t \mu(\theta_t)\,.$$

At each BO iteration, the new sample is chosen to maximize the value of the acquisition function:

$$\theta_{\text{new}} = \arg\max_{\theta} g(\theta)\,,$$

where

$$g(\theta) = g(\mu^+, \mu(\theta), \sigma(\theta))\,.$$

High values of the acquisition function correspond to regions where the expected value $\mu(\theta)$ of the objective function value is high or where the prediction uncertainty $\sigma(\theta)$ is high or both. Deciding which areas have the largest potential is known as the exploration vs exploitation trade off (see, e.g., Jones, 2001).

The choice of possible acquisition criteria is quite large with developments still taking place (see, e.g., Brochu et al., 2010b; Lizotte et al., 2012). Here, we illustrate two of the most popular acquisition functions called *probability of improvement* (Kushner, 1964) and *expected improvement* (Mockus, 1989). The probability of improvement (PI) is formulated as

$$g_{\text{PI}}(\theta) = \Phi\left(\Delta/\sigma(\theta)\right) \tag{7}$$

$$\Delta = \mu(\theta) - \mu^+ - \xi \tag{8}$$

where $\mu(\theta)$ and $\sigma(\theta)$ are defined in Eq. (5) and Eq. (6), respectively. $\Phi(\cdot)$ is the normal cumulative distribution function. When $\xi = 0$, $g_{\text{PI}}(\theta)$ is simply the probability of improving the best value μ^+ by taking a sample at location θ. The problem with using $\xi = 0$ is that PI favors locations that have even a slight improvement over the current best μ^+. This means that in this setting PI has a higher tendency to exploit rather than explore and it practically always gets stuck at a local optimum (Lizotte et al., 2012). The parameter $\xi > 0$ allows for tuning PI in order to reduce this problem. However, the choice of ξ is always subjective, although it has a great impact on the performance. For a detailed study of the effect of ξ, we recommend the work of Lizotte et al. (2012).

$$g_{\text{EI}}(\theta) = \left\langle \mathbf{f}_\theta - \mu^+ \right\rangle \tag{9}$$

$$= \Delta\Phi\left(\Delta/\sigma(\theta)\right) + \sigma(\theta)\phi\left(\Delta/\sigma(\theta)\right) \tag{10}$$

where $\langle\cdot\rangle$ denotes expectation, Δ is defined in Eq. (8) and $\phi(\cdot)$ is the normal probability density function. The EI criterion is derived as the expected difference between the function value in a new location $f(\theta)$ and the current best μ^+. Thus, EI aims at maximizing f by the biggest margin and yields optimization which is less prone to getting stuck in a local optimum. Nevertheless, using a tuning parameter $\xi > 0$ allows for control of the exploration vs exploitation trade-off (Lizotte et al., 2012).

Figure 1 illustrates a one-dimensional maximization procedure using BO. We start with three function evaluations and show the sampled points, the GP fits (with the red line) and the posterior uncertainty (with the pink filled areas). The acquisition functions are shown along below the subplots of the objective function. The new location (marked with a magenta vertical line) is chosen so that it maximizes the acquisition function. As the optimization proceeds, we collect more samples and finally find the maximum of the objective function. One can notice that, compared to PI, EI favors exploration as it samples from regions with higher uncertainty. The objective function approximation obtained

with EI improves much faster compared to PI. EI is also able to find the global maximum earlier. In this case, one could argue that using a larger value of ξ in PI could result in more exploration and faster optimization. However, choosing the right ξ value for PI is generally difficult. In our experimental study, we use the EI acquisition function ($\xi = 0$) with BO for parametric tuning of the example models. A detailed study about the effect of ξ on EI is given in Lizotte et al. (2012).

As stated earlier, the acquisition function is maximized at every step of BO in order to find a sample with the best potential. The acquisition function typically has multiple local optima (see Fig. 1) and the ability to find the *global* optimum of the acquisition function is extremely important for the efficiency of BO. Thus, global optimization procedures are typically used, which can be computationally demanding. Nevertheless, this procedure is usually far cheaper computationally because the global optimization only evaluates the GP and does not touch the objective function, which is computationally the most expensive part. Any global optimization method can be used in this task. In this work we have used the *DIRECT* method by Jones et al. (1993). *DIRECT* stands for dividing rectangles, this method is based on derivative-free optimization (see, e.g., Rios, 2013).

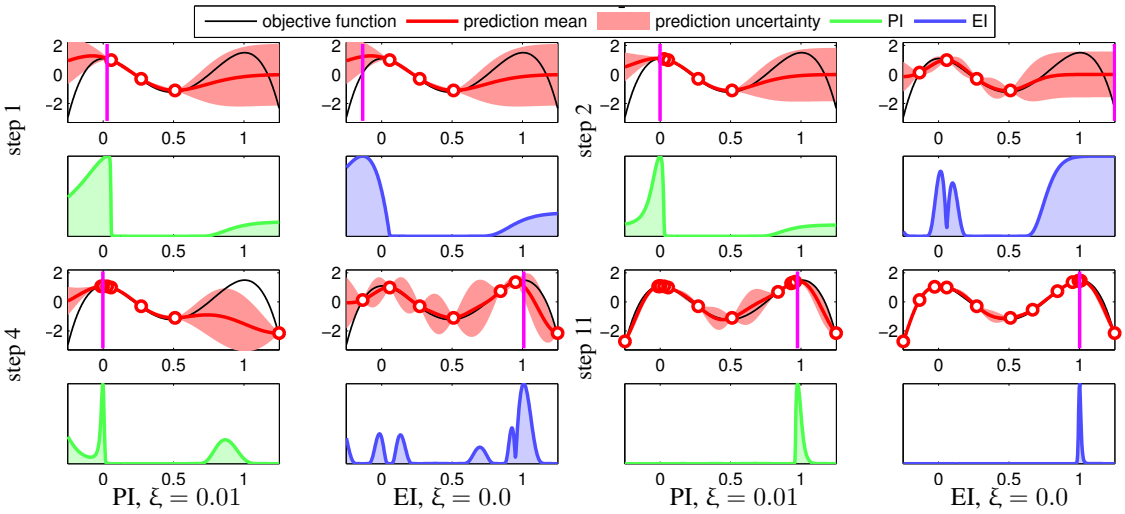

FIG. 1: One-dimensional demonstration of Bayesian optimization. The objective function f is shown with the black line. The circles represent sampled values of the objective function $\{f(\theta_i)\}_{i=1,\ldots,t}$. The red line is the prediction mean $\mu(\theta_i)$ and pink fill color is the uncertainty $\sigma^2(\theta_i)$ (± 2 standard deviation). The vertical magenta lines show the new sample locations θ_{t+1} proposed by BO so as to maximize the acquisition function. Note that all the horizontal axes show sample locations θ. The first and third column correspond to the PI (7) acquisition function. The second and fourth column correspond to the EI (9) acuisition function.

3. FILTERING METHODS FOR LIKELIHOOD EVALUATION

In tuning chaotic systems, we use the approach where the likelihood is computed using filtering techniques (Hakkarainen et al., 2012). Note that the likelihood from the filtering techniques is the objective function f. In section 3.1, we explain the inter-connection between the estimation of the filtering likelihood and BO. The tuned system is represented as a state-space model

$$\mathbf{s}_k = \mathcal{M}(\mathbf{s}_{k-1}) + \mathbf{E}_k \tag{11}$$

$$\mathbf{y}_k = \mathcal{K}(\mathbf{s}_k) + \mathbf{e}_k, \tag{12}$$

where $\mathbf{s}_k$ is the state of the model, $\mathbf{y}_k$ is the observation vector, $\mathcal{M}$ is the forward model which can be implemented by a solver of partial differential equations and $\mathcal{K}$ is the observation operator. $\mathbf{E}_k$ and $\mathbf{e}_k$ are noise terms which account

for model imperfection and observation noise. In climate science applications, model parameters θ usually appear in the formulation of $\mathcal{M}$ or/and they can govern the distribution of the model error term $\mathbf{E}_k$.

Filtering methods evaluate the likelihood by sequentially estimating the dynamically changing model state $\mathbf{s}_k$ for a given observation sequence $\mathbf{y}_{1:k} = \{\mathbf{y}_1, \ldots, \mathbf{y}_k\}$. Filters work by iterating two steps: prediction and update. In the prediction step, the current distribution of the state is evolved with the dynamical model to the next time step. The predictive distribution is given by the integral

$$p(\mathbf{s}_k|\mathbf{y}_{1:k-1}, \theta) = \int p(\mathbf{s}_k|\mathbf{s}_{k-1}, \theta) p(\mathbf{s}_{k-1}|\mathbf{y}_{1:k-1}, \theta) d\mathbf{s}_{k-1} \,. \tag{13}$$

As this integral generally does not have a closed form solution, it is usually approximated in one way or another. This yields different filtering techniques such as extended Kalman filter, ensemble Kalman filter, particle filter and so on.

The state distribution is updated using a new observation $\mathbf{y}_k$ using the Bayes rule:

$$p(\mathbf{s}_k|\mathbf{y}_{1:k}, \theta) \propto p(\mathbf{y}_k|\mathbf{s}_k, \theta) p(\mathbf{s}_k|\mathbf{y}_{1:k-1}, \theta). \tag{14}$$

This posterior is used inside the integral Eq. (13) to obtain the prior for the next time step.

The likelihood $p(\mathbf{y}_{1:n}|\theta)$ of the model parameters can be computed from the quantities evaluated in the filtering procedure:

$$p(\mathbf{y}_{1:K}|\theta) = p(\mathbf{y}_1|\theta) \prod_{k=2}^{K} p(\mathbf{y}_k|\mathbf{y}_{1:k-1}, \theta) \,, \tag{15}$$

where $p(\mathbf{y}_k|\mathbf{y}_{1:k-1}, \theta)$ is calculated based on the marginal posterior of the states:

$$p(\mathbf{y}_k|\mathbf{y}_{1:k-1}, \theta) = \int p(\mathbf{y}_k|\mathbf{s}_k, \theta) p(\mathbf{s}_k|\mathbf{y}_{1:k-1}, \theta) d\mathbf{s}_k \,.$$

Our goal is to search for parameters that maximize this likelihood $f(\theta) = p(\mathbf{y}_{1:K}|\theta)$ in Eq. (15).

Extended Kalman filter (EKF) is a filtering technique in which the integrals are approximated by linearization of the forward model $\mathcal{M}$ and the observation operator $\mathcal{K}$ around the current state estimate. Assuming that the observation error is normally distributed with zero mean and covariance matrix $\mathbf{R}_k$, the linearization yields:

$$p(\mathbf{y}_{1:n}|\theta) \propto \exp\left(-\frac{1}{2} \sum_{k=1}^{n} \mathbf{r}_k^T \left(\mathbf{C}_k^y(\theta)\right)^{-1} \mathbf{r}_k + \log|\mathbf{C}_k^y(\theta)| \right) \tag{16}$$

$$\mathbf{C}_k^y(\theta) = \mathbf{K}_k \left(\mathbf{M}_k \mathbf{C}_{k-1}^{\text{est}} \mathbf{M}_k^T + \mathbf{Q}_k(\theta) \right) \mathbf{K}_k^T + \mathbf{R}_k \tag{17}$$

where $\mathbf{r}_k = \mathbf{y}_k - \mathcal{K}(\mathbf{s}_k^p)$ are the prediction residuals, $\mathbf{M}_k$ and $\mathbf{K}_k$ are the linearization of $\mathcal{M}$ and $\mathcal{K}$ operators, respectively, $\mathbf{C}_{k-1}^{\text{est}}$ is the estimated covariance of $p(\mathbf{s}_k|\mathbf{y}_{1:k}, \theta)$ at time $k-1$ and $|\cdot|$ denotes the matrix determinant.

When the dimensionality of the tuned model is too large, the extended Kalman filter suffers from memory issues. Another problem is that linearization is often too cumbersome for highly complex models. In such scenarios, more sophisticated techniques like stochastic ensemble Kalman filters (EnKF) are often used for filtering.

The basic idea of EnKF is that the posterior distribution of the states is approximated using sample statistics, which are computed using a relatively small number of ensemble members propagated by the model at every assimilation step. Stochastic filters involve random perturbations of the model states and observations, which introduces randomness in the likelihood evaluation. More details on EnKF can be found, for example, in Evensen (2007).

3.1 BO for parametric tuning of chaotic systems using filtering likelihood

The evaluation of the likelihood requires running a data assimilation process for a computationally expensive model. Thus, each function evaluation is expensive and there can be noise in the likelihood, for instance, the noise caused by stochastic filtering techniques.

BO facilitates the optimization process by using a GP-based model as a surrogate of the tuned chaotic system, as shown in Fig. 2. Thus, in addition to the original tuning problem, that is the estimation of the parameters of a chaotic system, BO also requires the estimation of the parameters of the surrogate model.

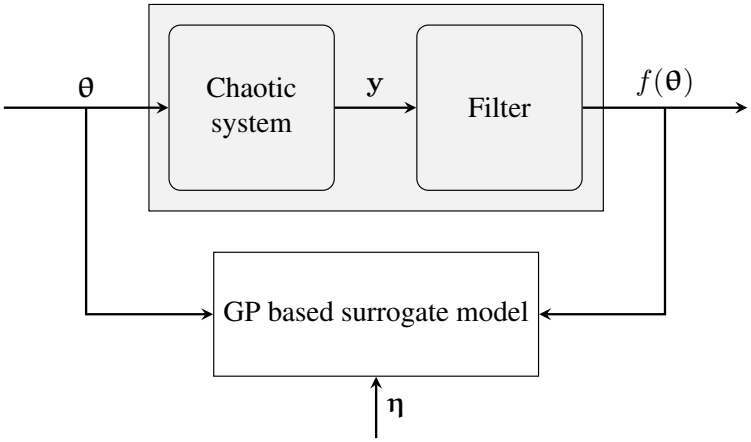

FIG. 2: A schematic illustration of the models tuned using BO. The chaotic system and filtering method combined together represent a data assimilation process. $\boldsymbol{\theta}$ is the input (parameters), $\mathbf{y}$ are observations and $f(\boldsymbol{\theta})$ is the output (log-likelihood function values) of the system. Pairs of input-output (data) are used to learn a GP based surrogate model. There are two estimation processes at play during BO: first is the estimation of the model parameters with the filtering likelihood technique. Second is a *meta-estimation* process which optimizes the GP based surrogate model. $\boldsymbol{\eta}$ represents hyperparameters of the GP model.

4. PARAMETRIC TUNING OF AN ATMOSPHERIC MODEL WITH "NOISELESS" LIKELIHOOD EVALUATIONS

In the following experiment, we tune a model of synoptic-scale chaotic dynamics. The likelihood is evaluated using the extended Kalman filter (EKF), which results in noiseless likelihood evaluations.

4.1 A two-layer quasi-geostrophic model

The quasi-geostrophic (QG) model simulates atmospheric flow for the geostrophic (slow) wind motions (see, e.g., Fisher et al., 2011). The chaotic nature of the dynamics generated by the QG model is shown in Vannitsem and Nicolis (1997). In our case, the system consists of two atmospheric layers and the geometrical domain of the model is specified by a cylindrical surface. The two atmospheric layers can interact through the interface between them (see left side in the Fig. 3). The system is simulated on a uniform grid for each atmospheric layer where the geometric structure of the model dictates periodic latitudinal boundary conditions and the values at the top and bottom of the cylindrical domain are set to pre-defined constant values.

When the geometrical domain of the two-layer QG model is mapped onto a plane, as shown in the right side in the Fig. 3, the two atmospheric layers are indicated as the top layer and the bottom layer with U_1 and U_2 denoting the mean zonal flows in the top and the bottom atmospheric layers, respectively. The orography in the model is such that there is a *hill* formation which affects the flow in the bottom atmospheric layer. This model generates features such as baroclinic instability common to operational weather models. Therefore, it can be used for the data assimilation process in numerical weather prediction (NWP) systems. The model dynamics are governed by the potential vorticity equations

$$\frac{\mathcal{D}_1}{\mathcal{D}t}\left(\nabla^2\psi_1 - F_1\left(\psi_1 - \psi_2\right) + \beta y\right) = 0, \tag{18}$$

$$\frac{\mathcal{D}_2}{\mathcal{D}t}\left(\nabla^2\psi_2 - F_2\left(\psi_2 - \psi_1\right) + \beta y + R_s\right) = 0, \tag{19}$$

where ψ_i denotes the model state vector called stream function and index i specifies the top atmospheric layer ($i = 1$)

and the bottom layer ($i = 2$). $\mathcal{D}_i$ denotes the substantial derivatives for latitudinal wind u_i and longitudinal wind v_i:

$$\frac{\mathcal{D}_i \cdot}{\mathcal{D}t} = \frac{\partial \cdot}{\partial t} + u_i \frac{\partial \cdot}{\partial x} + v_i \frac{\partial \cdot}{\partial y} .$$

Parameters R_s and β denote dimensionless orography component and the northward gradient of the Coriolis parameter f_0.

The relationship between the model physical attributes and parameters F_1 and F_2 in Eqs. (18) and (19) is defined by

$$F_1 = \frac{f_0^2 L^2}{\acute{g} D_1},\ F_2 = \frac{f_0^2 L^2}{\acute{g} D_2},\ \acute{g} = g \frac{\Delta\theta}{\bar{\theta}},$$
$$R_s = \frac{S\,(x,y)}{\upsilon D_2},\ \beta = \beta_0 \frac{L}{U},$$

where D_1 and D_2 are the depths of the two layers, $\Delta\theta$ defines the potential temperature change on the layer interface, $\bar{\theta}$ is the mean potential temperature, g is acceleration of gravity, $\upsilon = \frac{U}{f_0 L}$ is the Rossby number associated with the defined system, $S(x, y)$ is dimensional orography, and $S(x, y)$ and β_0 are dimensional representations of $R_s(x, y)$ and β, respectively. We used the implementation of the two-layer QG-model developed by Bibov (2011).

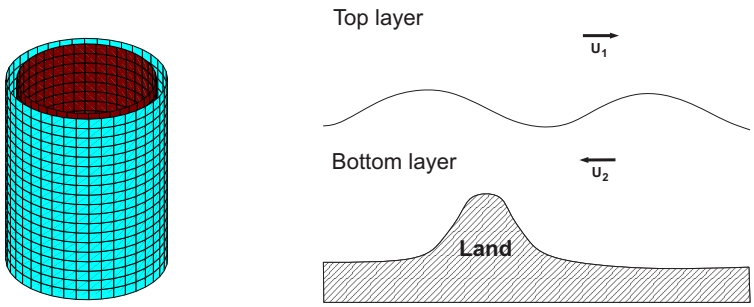

FIG. 3: Illustration of the two-layer quasi-geostrophic (QG) model. Left: schematic representation of the two atmospheric layers on a rotating cylinder. Right: the model layout at one of the longitudes.

4.2 Experimental setup

The described QG-model is used to formulate a synthetic problem of chaotic system tuning. The data used in the tuning process are generated by a QG model resolved on a dense 120×62 grid with the following parameters:

- layer depths are $D_1 = 6$ km and $D_2 = 4$ km
- distance between the grid points is 100 km.

This is our *true* system which is only used for generating the data.

The tuned system is a model which is governed by the same equations but it is resolved on a sparser grid 40×20 with the distance between grid points to be 300 km. This truncation of the grid size is a common practice in actual climate model testing. Thus, bias is being added to our tuning model because the fast processes affecting the observations on the finer scale will remain unmodeled.

The tuned system is represented as a state-space model Eqs. (11) and (12) where the forward model $\mathcal{M}$ is implemented by a solver of Eqs. (18) and (19). The state is the 1600-dimensional vector of the stream function values in

every point on the grid ($2 \times 40 \times 20$). In our scenario, the tuned quantity is the covariance matrix $\mathbf{Q}_k$ of the model error term $\mathbf{E}_k$ which is assumed to be normally distributed:

$$\mathbf{E}_k(\boldsymbol{\theta}) \sim N(\mathbf{0}, \mathbf{Q}(\boldsymbol{\theta}))\,. \tag{20}$$

The model error covariance matrix $\mathbf{Q}(\boldsymbol{\theta})$ is parameterized such that its ijth element is given by

$$\tau^2 \delta_{ij} + \sigma_q^2 \rho_{ij} \exp\left(-\frac{x_{ij}^2}{2\alpha^2}\right)\,, \tag{21}$$

where δ_{ij} equals 1 when $i = j$ and 0 otherwise. Component $\exp\left(-\frac{x_{ij}^2}{2\alpha^2}\right)$ represents the covariance function which models the dependency of the correlation on the distance between points: x_{ij} is the distance between points i and j projected on the same layer and α is a tuning parameter. ρ_{ij} defines correlations between the two layers and it is equal to 1 if i and j are in the same layer and $\rho_{ij} = \rho = \exp(-\gamma^2)$ otherwise (thus the actual tuning parameter is γ). Parameter σ_q^2 is the scaling parameter and τ is the nugget term often used to assure numerical stability.

Thus, there are four tuning parameters in total. The parameterization in Eq. (21) assures the positive-definiteness of $\mathbf{Q}$ for any combinations of the tuned parameters, which is important for the stability of BO. This corresponds to describing the model error as a Gaussian process with a covariance function separable in x and γ domains. In order to use a valid covariance function (see e.g., Banerjee, 2005; Gneiting, 2013), we computed the distance x_{ij} in the three-dimensional space, not on the cylinderical surface.

In the experiments, we assume that noisy observations of the simulated stream function are available at 50 randomly selected grid points every six hours. Thus, the observation operator $\mathcal{K}$ in Eq. (12) simply selects some of the elements of the state vector as being observed. The standard deviation of the iid Gaussian noise added to the simulated ψ values is $\sigma_y = 0.1$. The same value was used to form the covariance matrix of the observation noise in the tuned system in Eq. (12). The observation sequence contained 400 time instances.

We evaluate the likelihood using the extended Kalman filter, as described in Sect. 3.

4.3 Experimental results

We used BO with the EI acquisition function for tuning the parameters of the model error covariance matrix. Initially, we draw samples using the Latin hypercube sampling Lizotte et al. (LHS, see, e.g., 2012) in the region $\alpha \in [10\ \ 500]$, $\sigma_q^2 \in [0.01\ \ 0.81]$, $\rho \in [0.61\ 0.97]$ and $\tau^2 \in [0.25\ 0.81]$. The initialization consists of $(n = 10 \times 4 + 1)$ points (see pg. 473, Jones, 1998). In practice, we worked with the logarithm of the tuned parameters.

Figure 4 presents the results of BO using EI with $\xi = 0$. Here, we plot the approximation of the objective function using the mean of GP fitted after 200 iterations over all the data (initial 41 samples and BO 200 samples). Since there are four tuning parameters, each subplot presents the surface of the objective function when parameters α and $\log(\sigma_q)$ are varied and the other two parameters ρ and τ are fixed (see the corresponding fixed values above each subplot). The squares represent the samples used at the stage of initial sampling while the circles represent the samples gathered during the optimization procedure. Note that the locations of the samples are approximate: they are projected to the nearest plane corresponding to one of the subplots.

Figure 5 shows the behavior of BO at each iteration of the algorithm starting from the initial data. In Fig. 5, each point on the cross-marked line is computed by

$$\mathcal{L}_t = \log\left(f(\boldsymbol{\theta}_t) - \max(\mu^*, f^*)\right)\,, \tag{22}$$

where f^* is the maximum (log-likelihood function value) and μ^* is the GP mean value corresponding to the maximum found using BO in this experiment. The solid line shows $\max(\mathcal{L}_{1:t})$: the maximum obtained upto the current iteration t. Here, $\max(\mu^*, f^*) = \mu^*$. We observe that the number of log-likelihood function values required by the BO method to find the best point was 152. Here, we fixed the total number of iterations to 200. While other optimization performance criteria can be used as well (see, e.g., Huang et al., 2006, p. 457).

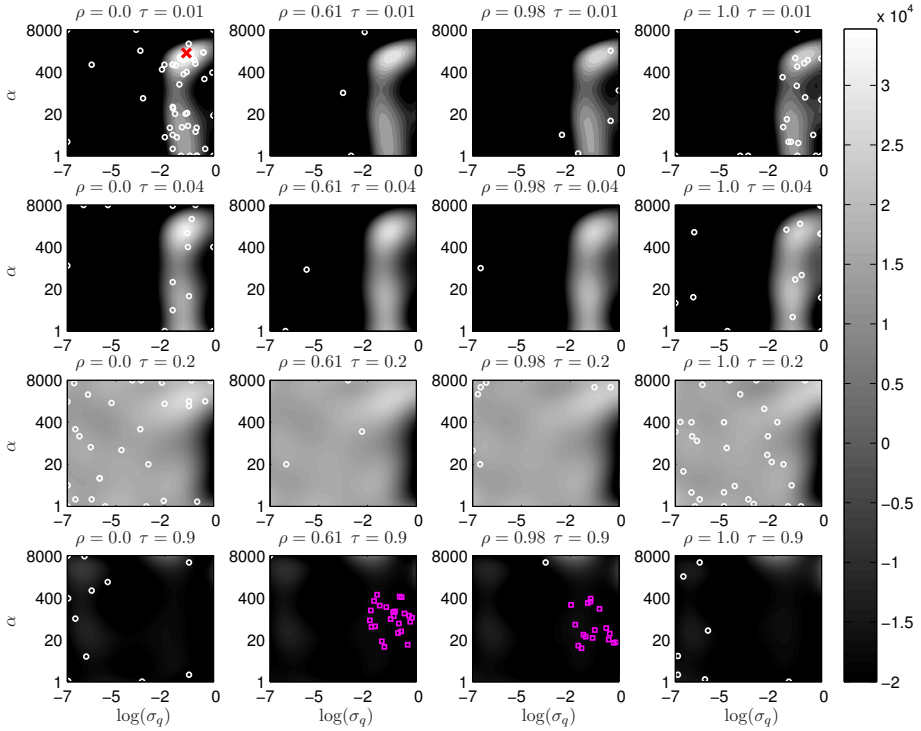

FIG. 4: The illustration of the surface of the objective function estimated during BO for parametric tuning of the QG model. Each subplot corresponds to a surface obtained by varying two parameters and keeping the other two parameters fixed. The (magenta) squares and (white) circles represent the sampled locations. Initial marked by the squares and the ones obtained during optimization marked by the circles. The (red) cross mark indicates the maximum of the likelihood function. Note the logarithmic scale for parameter σ_q. See text for a more thorough explanation.

This experiment shows that BO with the EI acquisition function is able to find the maximum of the posterior computed with GP while overcoming the local optima corresponding to small values of α. These values can be seen more clearly in the top four subplots (first row) of the Fig. 4. The exploration property of the method enables us to find the optima in a region away from where we draw the initial samples. Therefore, we can see from Fig. 5 that the best value found upto the current iteration gradually improves over time.

5. PARAMETRIC TUNING OF A CHAOTIC SYSTEM WITH "NOISY" LIKELIHOOD EVALUATIONS

In the following example, we simulate a scenario of tuning a large-scale chaotic system in which the evaluation of likelihood Eq. (15) using the extended Kalman filter is infeasible and therefore a stochastic EnKF is used. This results in noisy evaluations of the likelihood. As a tuned system we use a parameterized Lorenz 95 model.

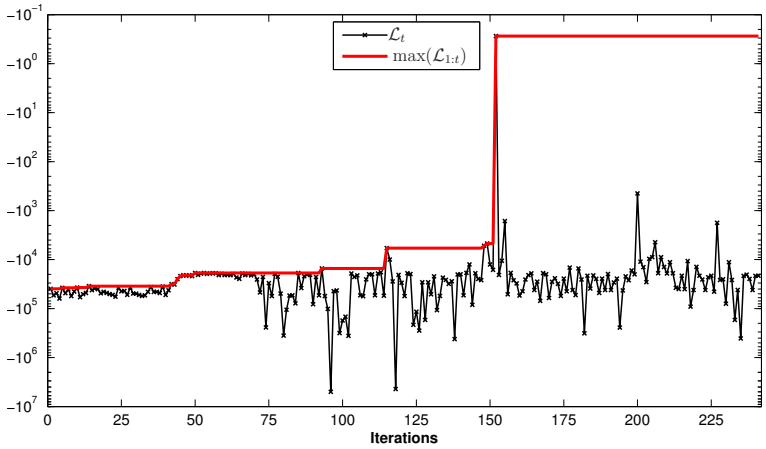

FIG. 5: BO applied to the parametric tuning of two-layer quasi-geostrophic model. The lines illustrate behavior of BO at each iteration of the algorithm starting from the initial data. Each point on the cross marked (black) line $\mathcal{L}_t$ is computed using (22). The solid (red) line shows $\max(\mathcal{L}_{1:t})$: the maximum obtained upto the current iteration t. See text for more details.

5.1 Parameterized Lorenz 95 model

The model generating the data is the classical Lorenz 95 model (Lorenz, 1995; Wilks, 2005) whose dynamics is given by

$$\frac{\mathrm{d}x_k}{\mathrm{d}t} = -x_{k-1}(x_{k-2} - x_{k+1}) - x_k + F - \frac{hc}{b} \sum_{j=J(k-1)+1}^{Jk} z_j, \tag{23}$$

$$\frac{\mathrm{d}z_j}{\mathrm{d}t} = -cbz_{j+1}(z_{j+2} - z_{j-1}) - cz_j + \frac{c}{b}F_z + \frac{hc}{b}x_{1+\lfloor \frac{j-1}{J} \rfloor} \tag{24}$$

where $k = 1, \ldots, K$ and $j = 1, \ldots, JK$. We use values $K = 40,\ J = 8,\ F = F_z = 10,\ h = 1$ and $c = b = 10$. In this model, the evolution of the slowly changing state variables x_i is affected by fast variables z_j and vice versa.

The tuned model is designed such that only the evolution of the slow variables is modeled and the net effect of the fast variables is represented with a deterministic component, such that

$$\frac{\mathrm{d}x_k}{\mathrm{d}t} = -x_{k-1}(x_{k-2} - x_{k+1}) - x_k + F - g(x_k, \theta), \tag{25}$$

and $g(x_k, \theta)$ is selected to be a polynomial $g(x_k, \theta) = \sum_{i=0}^{d} \theta_i x_k^i$, similarly to Hakkarainen et al. (2012). In our experiments, we use the polynomial of order $d = 1$ which corresponds to slope θ_1 and intercept θ_0. The forcing term remains unchanged, that is $F = F_z = 10$.

5.2 Experimental setup and likelihood formulation

The data is generated from the Lorenz 95 model Eqs. (23) and (24) with the discretization interval $\Delta t = 0.0025$. The state of the system is represented by a 40-dimensional vector of the slow variables x_k. We assume that noisy observations of the slow variables are available at 24 locations each day (one day corresponds to 0.2 time units). The last three state variables from every set of five states are picked and thus we observe the states $3, 4, 5, 8, 9, 10, \ldots, 38, 39, 40$. The standard deviation of the iid Gaussian noise added to the simulated x_j values is $(0.1\sigma_{\text{clim}})^2$, where $\sigma_{\text{clim}} = 3.5$ corresponds to a climatological standard deviation.

The tuned model is the parameterized Lorenz 95 model Eq. (25) simulated with the discretization interval $\Delta t = 0.025$ using 50, days of observations. The tuned system is formulated as a state-space model Eqs. (11) and (12) with a diagonal covariance matrix $\mathbf{Q} = \sigma_l^2 \mathbf{I}$ with σ_l^2 fixed to 0.0065, the value found to be optimal in our previous studies (Hakkarainen et al., 2012). Thus, the two tuned parameters are slope θ_0 and intercept θ_1 of the polynomial parameterization Eq. (25).

The objective function is the likelihood Eq. (15) computed via stochastic EnKF with 100 ensemble members (see, Hakkarainen et al., 2012) for details. Since we use a version of EnKF that involves random perturbations, the likelihood evaluations are noisy. Noisy likelihood introduces difficulties in standard methods that try to explore or optimize the likelihood surface.

5.3 Experimental results

We present at first the results of a single run of BO with the EI acquisition function ($\xi = 0$). Initially, we draw samples using the LHS design (Lizotte et al., 2012) in the region $4.0 \leq \theta_0 \leq 7.0$ and $0.57 \leq \theta_1 \leq 0.8$. The initialization consists of $(n = 10 \times 2 + 1)$ points (see pg. 473, Jones, 1998). These samples are represented with the squares in Fig. 6b. Figure 6b shows the GP mean approximation over all the data (initial 21 samples and BO 150 samples). The samples obtained using BO are shown with the circles and the cross mark indicates the found optimum. The found optimal values of the parameters are $\theta_0 = 1.99$ and $\theta_1 = 0.07$. The standard deviation value (σ_n) of the estimated noise in the total collected data was 482, which is rather large compared to the variability of the systematic component of the objective function. Figure 6a shows a scatter of uniformly random sampling of 1000 likelihood evaluations.

In Fig. 7b, we show the region closer to the optimum. We also perform uniformly random sampling in this smaller region and the scatter of 1000 samples is shown in Fig. 7a. At the found optimum the standard deviation of the noise is around 200. Note that we worked with the logarithm of the tuned parameters and the likelihood function is evaluated using the parameter values in the normal scale.

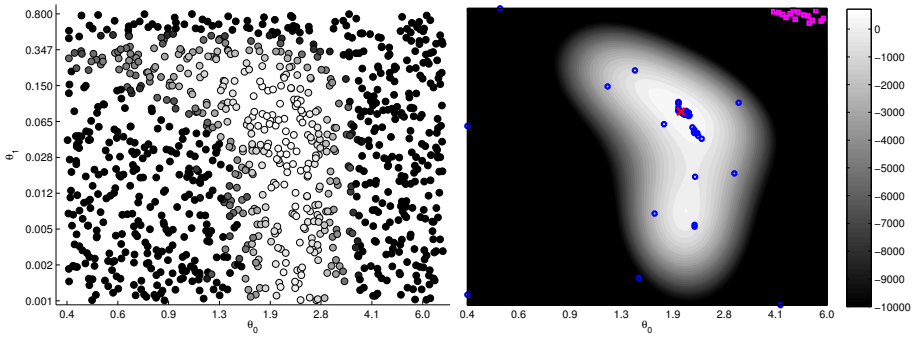

FIG. 6: (a) The scatter plot of the objective function values calculated on uniform random samples. **(b)** Single result of BO with EI for parametric tuning of the Lorenz 95 model using the EnKF likelihood. The (magenta) squares and (blue) circles represent the sampled locations. Initial marked by squares and the ones obtained during optimization marked by the circles. The (red) cross mark indicates the found maximum of the likelihood function. The contours represent the objective function approximation at the last iteration of BO.

Secondly, we present the results of the 200 experiments performed using BO for the parmeterized Lorenz 95 model. Each time the initial points were drawn using the LHS design in the same region $4.0 \leq \theta_0 \leq 7.0$ and $0.57 \leq \theta_1 \leq 0.8$. For the objective metric, the desired log-likelihood value is obtained by evaluating the likelihood function for different parameter values in an evenly spaced two-dimensional grid over the region $1.95 \leq \theta_0 \leq 2.05$ and $0.07 \leq \theta_1 \leq 0.11$. This is the region where the optimum value is highly likely to be. The likelihood function was evaluated over the grid 200 times and the average log-likelihood function value was selected for each grid point. The maximum of these selected values was then set as the desired log-likelihood value f_d^*. We computed the root mean

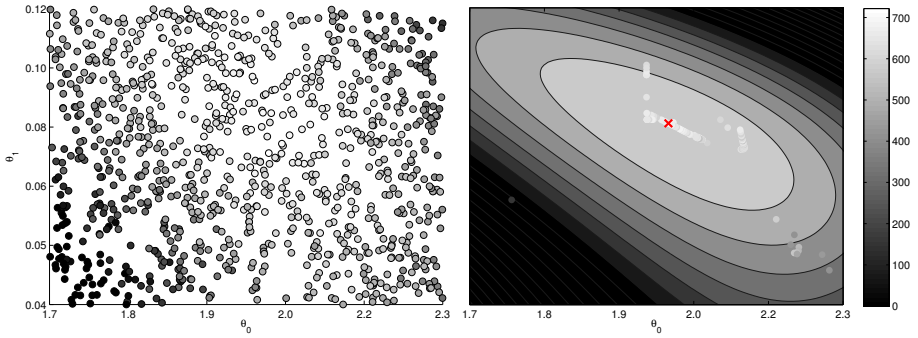

FIG. 7: The same result as in Fig. 6 but in a smaller region near the optimum. The BO samples in **(b)** are shown with circles whose gray-scale colors represent the evaluated objective function values. The (red) cross mark indicates the maximum.

square deviation by

$$\text{RMS} = \frac{1}{T f_d^*} \sqrt{\sum_{t=1}^{T} \|f_t^* - f_d^*\|^2}\,, \tag{26}$$

where the total number of likelihood function evaluations $T = 171$, and f_t^* is the maximum of the average log-likelihood function value upto the current iteration t obtained from BO experiments. For BO, the RMS value was 2.62 and the average standard deviation in the log-likelihood function values obtained from BO experiments normalized by f_d^* was ± 11.76.

Figure 8 shows the average behavior of BO at each iteration of the algorithm starting from the initial data (21 points). In Fig. 8, each circle is computed by

$$\mathcal{C}_t = \log\left(f(\boldsymbol{\theta}_t) - \max(f_b^*, f_c^*, f_d^*)\right)\,, \tag{27}$$

where f_b^* is the maximum (best log-likelihood function value) obtained from the 200 BO experiments and f_c^* is the maximum (best log-likelihood function value) obtained from 200 runs of the benchmark method (see details in Sect. 5.4). Each point on the plus marked line is the average value of the 200 samples of $\mathcal{C}_t$ and the solid line shows its maximum $\max(\mathcal{C}_{1:t})$: the maximum obtained upto the current iteration t. Here, $\max(f_b^*, f_c^*, f_d^*) = f_b^*$, hence, a small nugget 0.1 is added in Eq. (27) before taking the logarithm in order to avoid computing $\log(0)$. The dashed line is the desired log-likelihood value f_d^* which is plotted using Eq. (27). We explain the results and the comparison between BO and the benchmark method with respect to the objective metric in Sect.5.4.

5.4 Results of comaprison with the benchmark method

In this section, we present the experiments of the parametrized Lorenz 95 model for comparison of BO with the covariance matrix adaptation evolution strategy (CMA-ES) (see e.g., Hansen et al., 2009; Hansen and Ostermeier, 2001, 1996). CMA-ES is a powerful evolutionary (search) algorithm for difficult real-valued optimization problems. CMA-ES uses an iterative procedure to compute a covariance matrix, formulated as the inverse Hessian matrix (typically used in a quasi-Newton method). Unlike quasi-Newton methods, CMA-ES does not use or requires to compute gradients. CMA-ES can easily handle problems that have local optima and a noisy objective function.

With CMA-ES, we performed 200 runs, similarly to BO. Each time the initialization points were drawn uniformly randomly in the region $4.0 \leq \theta_0 \leq 7.0$ and $0.57 \leq \theta_1 \leq 0.8$. Unlike BO, CMA-ES does not require the log-likelihood function values at the initialization points because it uses this region only to compute a single mean point around which it starts generating the first population. The initial standard deviation (*step-size*) required by CMA-ES can also be computed with the initial points by the method itself. Hence, we do not provide an initial standard deviation

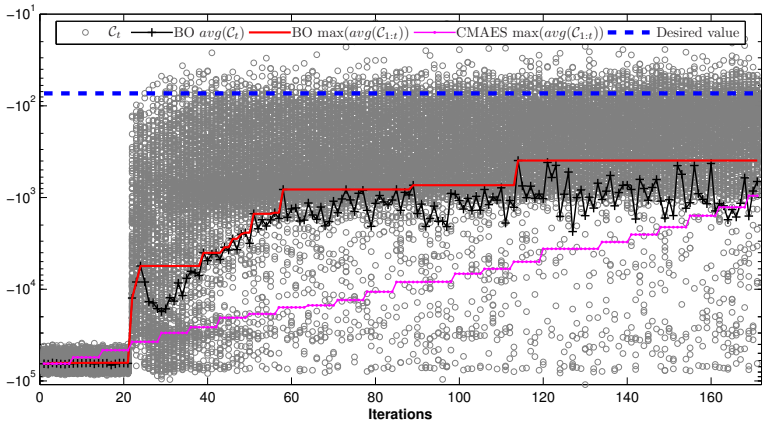

FIG. 8: Parametric tuning of the Lorenz 95 model. The solid (red) and the dot marked (magenta) lines illustrate the average behavior of BO and CMAES (benchmark method), respectively. A total of 200 experiments are performed using each method with random initializations of the initial parameter values in the same region in each experiment. This results in 200×171 likelihood function evaluations for each method. Each grey color circle $\mathcal{C}_t$ is computed using (27). Each point on the plus marked (black) line is the average value of the 200 samples of $\mathcal{C}_t$ and the solid (red) line shows its maximum $\max(\mathcal{C}_{1:t})$: the maximum obtained upto the current iteration t, for BO. The dot marked (magenta) line shows the same for CMAES method. The dashed (blue) line is the desired log-likelihood value in the objective metric (see text for details).

(*step-size*). CMA-ES generated a population size of 7 during each algorithmic iteration. After initialization, CMA-ES does require evaluating of the likelihood function for each member in the population at every algorithmic iteration. The method returns the mean and the median (sorted) of the log-likelihood function values at each iteration. In Fig. 8, the dot marked line shows the average behavior of CMA-ES at each iteration of the algorithm. The RMS value was 2.82 and the average standard deviation in the log-likelihood function values obtained from CMA-ES experiments normalized by f_d^* was ± 11.7. This shows that the performance gap between BO and CMA-ES is small, BO is able to maximize over the likelihood function faster and this is the edge it has when it is used for computationally expensive objective functions.

5.5 Accuracy of the tuned parameters

For the Lorenz 95 case, we test the goodness of the optimization scheme by computing the forecast accuracy, similarly as in Hakkarainen et al. (2012). We use a two-dimensional grid of the parameter space and compute the average forecast error for different parameter values. The average forecast error was computed using a 6 day forecast starting every 24h for 100 days. The average forecast error can be written as

$$F_{\text{avg}}(\theta) = \frac{1}{NK\sigma_{clim}^2} \sum_{i=1}^{N} \|\mathcal{M}_6(\mathbf{x}_i^{true}, \theta) - \mathbf{x}_{i+6}^{true}\|^2 \,, \tag{28}$$

where $N = 100$, $K = 40$ and $\sigma_{clim} = 3.5$. The notation $\mathcal{M}_6(\mathbf{x}_i^{true})$ means a 6 day prediction launched from the true state $\mathbf{x}_i^{true}$ with the parameter values θ. The relationship between the average forecast error and the tuned parameters is shown in Fig. 9(a). The contour surface represents the forecast error computed with Eq. (28). The squares indicate the parameter values used to initialize the BO method. The circles show the locations corresponding to samples obtained from BO. The cross indicates location of the maximum obtained using BO. Note that the result of BO shown in Fig. 9 is the same as discussed in section 5.3.

Similarly, the relationship between the EnKF likelihood function and the tuned parameters is shown in Fig. 9(b). The contour surface represents the EnKF log-likelihood function values obtained using a 50 days simulation length of

the model. The contours are drawn using the values computed on the same two-dimensional grid as in Fig. 9(a). Note that the contour lines are curly due to noisy values obtained with the EnKF likelihood function. The circles and cross mark correspond to the same result as shown in Fig. 9(a).

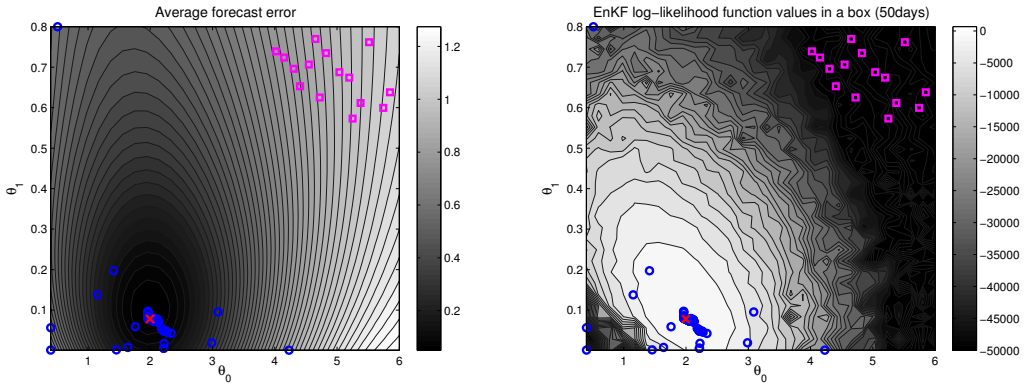

FIG. 9: (a) The contour surface represents the forecast error computed with (28). The contours are drawn using the error values computed on an evenly spaced two-dimensional grid. The (magenta) squares show the parameter values used to initialize the BO method. The (blue) circles show the locations corresponding to samples obtained from BO. The (red) cross indicates location of the maximum obtained using BO. The simulation length of the model used for the BO experiment was 50 days. (b) The contour surface represents the EnKF log-likelihood function values obtained using a 50 days simulation length of the model. The contours are drawn using the values computed on the same two-dimensional grid as in (a). Note that the contour lines are curly due to noisy values obtained with the EnKF likelihood function. The squares, circles and cross mark correspond to the same result as shown in (a) on the left.

With similar experiments, we tested the sensitivity of EnKF likelihood function and BO method to the simulation length of the Lorenz 95 model. We performed more experiments with larger number of days. The results of these experiments are shown in Fig. 10. These plots are similar to Fig. 9(b) except the differences that follow. In Fig. 10(a), the contour surface represents the EnKF log-likelihood function values obtained using a 100 days simulation length of the model. The simulation length of the model used for the BO experiment was also 100 days. In Fig. 10(b), the contour surface represents the EnKF log-likelihood function values obtained using a 500 days simulation length of the model. The simulation length of the model used for the BO experiment was also 500 days. From these results, we observed that there is a good agreement between the tuned parameters obtained with BO using the likelihood approach and the average forecast error when the simulation length is sufficient.

We performed the lead time analysis for the parameters obtained using the BO method. Figure 11 consists of three subplots corresponding to different simulation lengths (days) of the EnKF likelihood employed with BO. The BO parameters utilized for computing the lead time in each subplot of Fig 11 correspond to the same points as shown in Fig. 9(b), Fig. 10(a) and Fig. 10(b), respectively. We observe that the BO optimum in all three cases of 50, 100, and 500 days of EnKF likelihood simulation length has the lowest average forecast errors upto an acceptable lead time. The lead time analysis shows that the performance of BO is consistent over different lead times. It is also observed that the average forecast error saturates after a lead time of (5-6) days for some of the parameters, especially, for the initial points, which correspond to low log-likelihood function values (as shown in Fig. 9 and Fig. 10).

6. CONCLUSIONS

In this paper, we considered Bayesian optimization as a tool for parametric tuning of chaotic systems. We used two benchmark systems for testing the BO procedure: a simplified atmospheric model and a low-dimensional chaotic system. In the two-layer QG-model, the tuning parameters were four variables that constructed a model error covariance matrix used in the filtering with EKF. In the Lorenz 95 model, the tuning parameters were two variables that were used

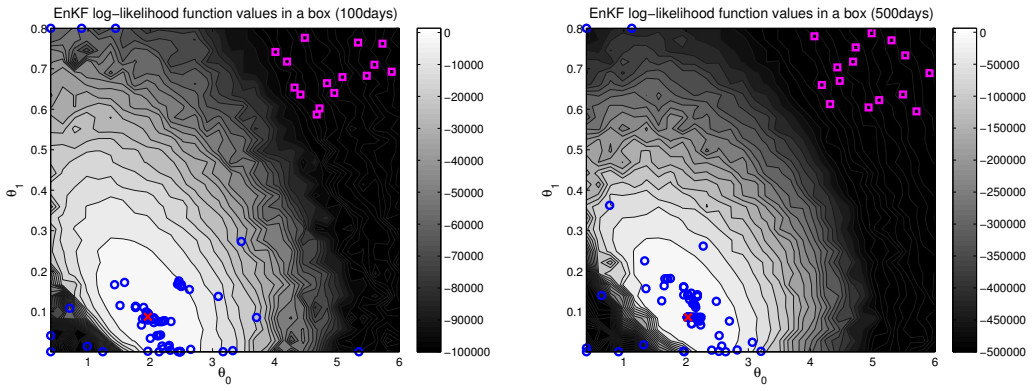

FIG. 10: Similar plots as shown in Fig. 9(b) except the differences that follow. In (a), the contour surface represents the EnKF log-likelihood function values obtained using a 100 days simulation length of the model. The simulation length of the model used for the BO experiment was also 100 days. In (b), the contour surface represents the EnKF log-likelihood function values obtained using a 500 days simulation length of the model. The simulation length of the model used for the BO experiment was also 500 days.

in the construction of a polynomial parameterization. For both experiments, the learning was based on the filtering likelihood.

The experiments showed that by using BO we were able to find the optimal solutions. The expensive models were tuned without the need of any gradient information. For the parametric tuning of Lorenz 95 model, we showed the performance of BO compared to CMA-ES using an objective metric. The results showed that the performace gap between BO and CMA-ES was small and that BO was able to maximize the likelihood function faster as compared to CMA-ES on average. The accuracy of the tuned parameters using the average forecast error showed the goodness of the optimization scheme. We showed the accuracy of the found parameters with respect to different simulation lengths of the model used in the EnKF likelihood function. We also showed the performance of the found parameters with respect to lead time.

The tested technique can be a practical tool in many tuning problems in climate sciences. Possible applications include parametric tuning of large-scale climate models (Schirber et al., 2013) and ensemble prediction systems (Solonen and Järvinen, 2013). However, there are some known issues when using the BO method for large systems: first, with the increase in the number of tuning parameters of the model the required number of samples to explore the domain would automatically increase. Second, with limited availability of training data due to expensive computational cost of the model, little is known about the objective function which means the design of the prior becomes more critical. BO is more suitable for computationally expensive black-box optimization tasks, especially, when the cost of evaluating the objective function is higher than the cost of fitting a GP. Third, with the exploration vs exploitation property of acquisition functions, one has to decide on how to handle such parameters. Although, some recent papers have shown that BO works well in real-world problems for the following dimensions: 12 in Brochu et al. (2010c), 9 in Snoek et al. (2012), or 9 in Brochu et al. (2010b), in general the problem of larger dimensionality still remains an active research topic, especially, in machine learning.

We see the above mentioned problems as a very interesting direction for researchers, especially, in weather and climate applications. Applying the BO technique to large-scale models like ECHAM5 is a future direction to consider, for example, this approach can be used for tuning four parameters of ECHAM5 that are related to clouds and precipitation, as done by Järvinen et al. (2010).

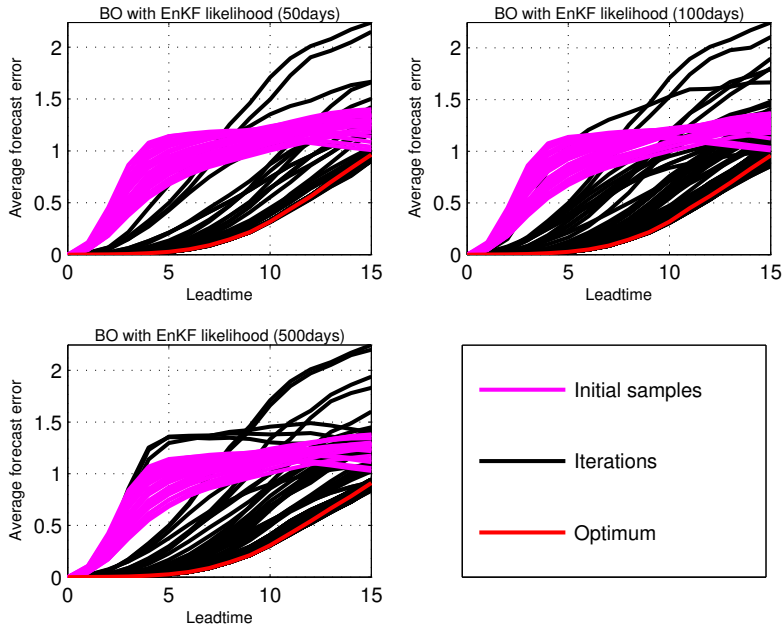

FIG. 11: Lead time analysis for the parameters obtained using BO method. The BO parameters utilized for computing the lead time in each subplot here correspond to the same points as shown in Fig. 9(b), Fig. 10(a) and Fig. 10(b), respectively.

ACKNOWLEDGMENTS

This work has been supported by the Academy of Finland (project number 134935) and the Finnish Centre of Excellence in Computational Inference Research (COIN).

REFERENCES

Annan, J. and Hargreaves, J., Efficient estimation and ensemble generation in climate modelling, Philos. T. R. Soc. A, vol. **365**, no. 1857, pp. 2077–2088, 2007.

Banerjee, S., On geodetic distance computations in spatial modeling, Biometrics, vol. **61**, no. 2, pp. 617–625, 2005

Bibov, A., Variational Kalman filter data assimilation for two-layer Quasi-Geostrophic model, M.S. thesis, Lappeenranta University of Technology, Finland, 2011.

Box, G. E. P. and Draper, N. R., Empirical model-building and response surfaces, New York: Wiley, pp. 304–316, 1987.

Boyle, P., Gaussian Processes for Regression and Optimisation, PhD, Victoria University of Wellington, New Zealand, 2007.

Brochu, E., Cora, V. M., and Freitas, N. D., A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning, Tech. rep., arXiv:1012.2599, arXiv.org, Cornell University Library, USA, 2010a.

Brochu, E., Hoffman, M., and Freitas, N. D, Portfolio Allocation for Bayesian Optimization, Tech. rep., arXiv:1009.5419, arXiv.org, Cornell University Library, USA, 2010b.

Brochu, E., Brochu, T., and Freitas, N. D, A Bayesian Interactive Optimization Approach to Procedural Animation Design, Proceedings of the 2010 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, pp. 103–112, Madrid, July 2010c.

Evensen, G., Data Assimilation: The Ensemble Kalman Filter, New York: Springer, pp. 119–194, 2007.

Fisher, M., Tremolet, Y., Auvinen, H., Tan, D., and Poli, P., Weak-constraint and long window 4DVAR, Tech. rep. 655, European Centre for Medium-Range Weather Forecasts, Reading, UK, 2011.

Frean, M. and Boyle, P.: Using Gaussian processes to optimize expensive functions, Proceedings of the 21st Australasian Joint Conference on Artificial Intelligence: Advances in Artificial Intelligence, pp. 258–267, 2008.

Gneiting, T., Strictly and non-strictly positive definite functions on spheres, Bernoulli, vol. **19**, no. 4, pp. 1327–1349, 2013

Hakkarainen, J., Ilin, A., Solonen, A., Laine, M., Haario, H., Tamminen, J., Oja, E., and Järvinen, H., On closure parameter estimation in chaotic systems, Nonlin. Processes Geophys., vol. **19**, no. 1, pp. 127–143, 2012.

Hakkarainen, J., Solonen, A., Ilin, A., Susiluoto, J., Laine, M., Haario, H., and Järvinen, H., A dilemma of the uniqueness of weather and climate model closure parameters, Tellus A, vol. **65**, pp. 20147, 2013.

Hansen, N., Niederberger, A. SP., Guzzella, L., and Koumoutsakos, P., A method for handling uncertainty in evolutionary optimization with an application to feedback control of combustion, Evolutionary Computation, IEEE Transactions on, vol. **13**, no. 1, pp. 180-197, 2009.

Hansen, N., and Ostermeier, A., Completely derandomized self-adaptation in evolution strategies, Evolutionary Computation, vol. **9**, no. 2, pp. 159-195, 2001.

Hansen, N., and Ostermeier, A., Adapting arbitrary normal mutation distributions in evolution strategies: The covariance matrix adaptation, In Proc. of IEEE Int. Conf. on Evolutionary Computation, pp. 312-317, May 1996.

Hauser, T., Keats, A., and Tarasov, L., Artificial neural network assisted Bayesian calibration of climate models, Clim. Dynam., vol. **39**, no. 1, pp. 137–154, 2012.

Huang, D., Allen, T. T., Notz, W. I., and Zeng, N., Global optimization of stochastic black-box systems via sequential kriging meta-models, J. Global Optim., vol. **34**, no. 3, pp. 441–466, 2006.

Hutter, F., Hoos, H., and Leyton-Brown, K., Bayesian optimization with censored response data, arXiv:1310.1947, arXiv.org, Cornell University Library, USA, 2013.

Järvinen, H., Räisänen, P., Laine, M., Tamminen, J., Ilin, A., Oja, E., Solonen, A., and Haario, H., Estimation of ECHAM5 climate model closure parameters with adaptive MCMC, Atmos. Chem. Phys., vol. **10**, no. 20, pp. 9993–10002, 2010.

Jones, D. R., A taxonomy of global optimization methods based on response surfaces, J. Global Optim., vol. **21**, no. 4, pp. 345–383, 2001.

Jones, D. R., Efficient Global Optimization of Expensive Black-Box Functions, J. Global Optim., vol. **13**, no. 4, pp. 455–492, 1998.

Jones, D. R., Perttunen, C. D., and Stuckman, B. E., Lipschitzian optimization without the Lipschitz constant, J. Optimiz. Theory App., vol. **79**, no. 1, pp. 157–181, 1993.

Kushner, H. J., A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise, ASME. J. Basic Eng., vol. **86**, no. 1, pp. 97–106, 1964.

Leutbecher, M. and Palmer, T., Ensemble forecasting, J. Comput. Phys., vol. **227**, no. 7, pp. 3515–3539, 2008.

Lizotte, D. J., Practical bayesian optimization, PhD, Edmonton, University of Alberta, Canada, 2008.

Lizotte, D. J., Greiner, R., and Schuurmans, D., An experimental methodology for response surface optimization methods, J. Global Optim., vol. **53**, no. 4, pp. 699–736, 2012.

Lorenz, E., Predictability: a problem partly solved, Proceedings of the Seminar on Predicability, European Center on Medium Range Weather Forecasting, Reading, UK, pp. 1–18, 1995.

Marzouk, Y. and Xiu, D., A stochastic collocation approach to Bayesian inference in inverse problems, Commun. Comput. Phys., vol. **6**, no. 4, pp. 826–847, 2009.

Mockus, J., Bayesian Approach to Global Optimization: Theory and Applications, Dordrecht: Kluwer Academic, pp. 79–116, 1989.

Mockus, J., Tiesis, V., and Zilinskas, A., Bayesian methods for seeking the extremum, Toward Global Optimization, North Holland, Amsterdam: Elsevier, vol. **2**, pp. 117–128, 1978.

Neelin, J. D., Bracco, A., Luo, H., McWilliams, J. C., and Meyerson, J. E., Considerations for parameter optimization and sensitivity in climate models, Proceedings of the National Academy of Sciences, vol. **107**, no. 50, pp. 21349-21354, 2010.

Osborne, M. A., Roman, G., and Roberts, S. J., Gaussian processes for global optimization, 3rd International Conference on Learning and Intelligent Optimization (LION3), pp. 1–15, 2009.

Rasmussen, C. E. and Williams, C. K. I., Gaussian Processes for Machine Learning, Cambridge MA: The MIT Press, pp. 79–193, 2006.

Rios, L. M. and Sahinidis, N. V., Derivative-free optimization: a review of algorithms and comparison of software implementations, J. Global Optim., vol. **56**, no. 3, pp. 1247–1293, 2013

Schirber, S., Klocke, D., Pincus, R., Quaas, J., and Anderson, J. L., Parameter estimation using data assimilation in an atmospheric general circulation model: from a perfect toward the real world, Journal of Advances in Modeling Earth Systems, vol. **5**, no. 1, pp. 58–70, 2013.

Snoek, J., Larochelle, H., and Adams, R. P., Practical Bayesian optimization of machine learning algorithms, arXiv:1206.2944, arXiv.org, Cornell University Library, USA, 2012.

Solonen, A. and Järvinen, H., An approach for tuning ensemble prediction systems, Tellus A, vol. **65**, pp. 20594, 2013.

Solonen, A., Ollinaho, P., Laine, M., Haario, H., Tamminen, J., and Järvinen, H., Efficient MCMC for climate model parameter estimation: parallel adaptive chains and early rejection, Bayesian Analysis, vol. **7**, no. 3, pp. 715–736, 2012.

Vannitsem, S. and Nicolis, C., Lyapunov vectors and error growth patterns in a T21L3 quasigeostrophic model, J. Atmos. Sci., vol. **54**, no. 2, pp. 347–361, 1997.

Wilks, D. S., Effects of stochastic parametrizations in the Lorenz '96 system, Q. J. Roy. Meteor. Soc., vol. **131**, no. 606, pp. 389–407, 2005.