Stochastic Environmental Research and Risk Assessment (2020) 34:1813–1830
https://doi.org/10.1007/s00477-020-01867-0

ORIGINAL PAPER

Sampling behavioral model parameters for ensemble-based sensitivity analysis using Gaussian process emulation and active subspaces

Daniel Erdal[1,2] · Sinan Xiao[3] · Wolfgang Nowak[3] · Olaf A. Cirpka[1]

Accepted: 28 August 2020 / Published online: 20 September 2020

Abstract

Ensemble-based uncertainty quantification and global sensitivity analysis of environmental models requires generating large ensembles of parameter-sets. This can already be difficult when analyzing moderately complex models based on partial differential equations because many parameter combinations cause an implausible model behavior even though the individual parameters are within plausible ranges. In this work, we apply Gaussian Process Emulators (GPE) as surrogate models in a sampling scheme. In an active-training phase of the surrogate model, we target the behavioral boundary of the parameter space before sampling this behavioral part of the parameter space more evenly by passive sampling. Active learning increases the subsequent sampling efficiency, but its additional costs pay off only for a sufficiently large sample size. We exemplify our idea with a catchment-scale subsurface flow model with uncertain material properties, boundary conditions, and geometric descriptors of the geological structure. We then perform a global-sensitivity analysis of the resulting behavioral dataset using the active-subspace method, which requires approximating the local sensitivities of the target quantity with respect to all parameters at all sampled locations in parameter space. The Gaussian Process Emulator implicitly provides an analytical expression for this gradient, thus improving the accuracy of the active-subspace construction. When applying the GPE-based preselection, 70–90% of the samples were confirmed to be behavioral by running the full model, whereas only 0.5% of the samples were behavioral in standard Monte-Carlo sampling without preselection. The GPE method also provided local sensitivities at minimal additional costs.

Keywords Global sensitivity analysis · Sampling behavioral models · Gaussian process emulation · Stochastic engine

1 Introduction

Numerical modeling of environmental processes is an important tool for many researchers and practitioners. With the increasing availability of computer power, we also see an increase in the size and complexity of the modeled systems (e.g., Kollet et al. 2010). For example, to describe flow and transport in surface-subsurface systems, it is common to solve systems of partial differential equations (*pde*) (e.g., Maxwell et al. 2015; Kollet et al. 2017). Aiming at a realistic representation of the physical

This work was supported by the Collaborative Research Center 1253 CAMPOS (Project 7: Stochastic Modeling Framework of Catchment-Scale Reactive Transport), funded by the German Research Foundation (DFG, Grant Agreement SFB 1253/1), the Sino-German (CSC-DAAD) Postdoc Scholarship Program 2018 (57395819) and Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy-EXC 2075-390740016.
All own-developed codes necessary to run the Stochastic Engine used in this work are available via http://hdl.handle.net/10900.1/6a66361b-b713-4312-819b-18f82f27aa18.

✉ Daniel Erdal
daniel.erdal@uni-tuebingen.de

[1] Center for Applied Geoscience, University of Tübingen, Hölderlinstr. 12, 72074 Tübingen, Germany

[2] Present Address: Tyréns AB, Lilla Badhusgatan 2, 41121 Göteborg, Sweden

[3] Institute for Modelling Hydraulic and Environmental Systems (LS3/SimTech), University of Stuttgart, Pfaffenwaldring 5a, 70569 Stuttgart, Germany

processes, these models are often highly uncertain due to uncertain material properties, boundary conditions, structural features, and geometric parameters. Commonly, all uncertain parameter values are inferred from sparse and indirect observations (a.k.a. model calibration), a field that has been studied intensively in subsurface hydrology (e.g., Vrugt et al. 2008; Shuttleworth et al. 2012; Yeh 2015). Before embarking into the calibration of a complex model, some form of sensitivity analysis (e.g., Saltelli et al. 2004, 2008; Xiao et al. 2018) is usually advisable to determine which parameters are sensitive to the data at hand. Sensitivity analysis is also a well-studied topic in hydrological science (e.g., Mishra et al. 2009; Song et al. 2015; Pianosi et al. 2016; Wagener and Pianosi 2019).

Both global-sensitivity analysis and statistical parameter inference may be performed using comparably large ensembles of model runs with different parameter values. A basic, but not trivial requirement is that every ensemble member should show a realistic behavior with respect to the modeled system. Typical examples of non-behavioral model runs include flooding of valleys that in reality are not wetlands, rivers falling dry that in reality are perennial, or the simulated reversal of observed flow directions, among others. Erdal and Cirpka (2019) analyzed the subsurface-flow model of a small valley, showing that the non-behavioral parameter-sets can take up a notable part of the parameter space. The behavioral parameter space may have non-trivial and unexpected boundaries, making unguided Monte-Carlo sampling computationally expensive. For large, computationally intensive models, the small and irregularly bounded behavioral parameter space may prohibit ensemble-based uncertainty quantification, global-sensitivity analysis, and model calibration.

An increasingly popular approach to decrease the computational effort of complex models is using surrogate models. Recent comprehensive reviews were given by Ratto et al. (2012), who considered environmental models, Razavi et al. (2012b), who considered hydrological models, and Asher et al. (2015) and Rajabi (2019), who both considered groundwater models. A surrogate-model, also known as a meta-model, proxy-model, emulator-model, or low-fidelity model, is in its most general form a simpler representation of a complex model. Owing to its relative simplicity, the surrogate model can be run much faster than the original complex model. In their review of surrogate models for groundwater applications, Asher et al. (2015) divided the surrogate models into three groups: (1) data-driven methods, in which the surrogate model mimics a given input-output relation based on training data originating from the original model, (2) projection-based methods, in which the original model is projected to a lower-order subspace to reduce parameter complexity, and (3) hierarchical methods, which directly simplify the actual model, e.g., by grid coarsening or assuming quasi-steady-state conditions (e.g von Gunten et al. 2014). Because in this work we are interested in approximating an input-output relationship, we will focus on data-driven methods. In studies related to subsurface flow and transport, commonly used data-driven methods are the polynomial chaos expansion (e.g., Laloy et al. 2013; Oladyshkin and Nowak 2012; Wu et al. 2014; Zhang et al. 2017), support vector machines (e.g., Yoon et al. 2011; Wu et al. 2015; Xu et al. 2017), and Gaussian Process Emulators, which have been used in the modeling of groundwater flow (e.g., Cui et al. 2018b, a), unsaturated flow (e.g., Zhang et al. 2018; Gadd et al. 2019; Zheng et al. 2019), subsurface transport (e.g., Ouyang et al. 2017; Zhang et al. 2018; Gadd et al. 2019; Zheng et al. 2019), saltwater intrusion (e.g., Rajabi and Ketabchi 2017; Kopsiaftis et al. 2019), and processes related to CO_2-sequestration (e.g., Espinet and Shoemaker 2013; Tian et al. 2017; Crevillén-García 2018).

Due to its wide-spread use, good results reported in the literature, relative simplicity, and inherent ability to provide not only predictions but also their uncertainty, we decided to use a Gaussian Process Emulator as our surrogate model in the present work. An additional advantage of Gaussian Process Emulators, which has not been emphasized in the literature so far, is that it can easily be extended to provide the gradient of the simulated quantity with respect to the parameters, a feature that we develop in Sects. 2.5 and 2.6. The gradients are needed when applying the active-subspace method (Constantine et al. 2014; Constantine and Doostan 2017) and other global-sensitivity analysis methods. Efficiently and reliably accessing gradients has been shown a difficult requirement of the active-subspace method in the past (e.g., Gilbert et al. 2016; Grey and Constantine 2018).

A key question in using surrogate models in the context of model calibration is how static the surrogate model should be. Razavi et al. (2012a) distinguished between a simple sequential approach, in which the surrogate model is trained once and no more changed, and an adaptive-recursive approach, in which the surrogate model is continuously updated when new runs of the full model become available. An example of the more common adaptive-recursive approach in subsurface modeling is the two-stage Markov-Chain Monte- Carlo (MCMC) method (Cui et al. 2011; Laloy et al. 2013). When considering the adaptive-recursive approach, a possible critical point is how to initialize the surrogate model. Any data-driven surrogate model needs a set of inputs and corresponding outputs (also known as snapshots or training samples) computed by the complex model it is about to mimic. However, as noted by Asher et al. (2015), many surrogate models lack frameworks for selecting these snapshots. If the aim is to create a sufficiently large ensemble of behavioral complex model runs, two main routes can be taken, that are analyzed in the present study.

The first route is to first systematically train the surrogate model on a non-random set of snapshots exploring the boundary between the behavioral and non-behavioral parameter space by so-called active learning (Cohn et al. 1996), until the surrogate model is good enough for predicting whether a parameter-set is behavioral. After the active training, the surrogate model is used for preselecting behavioral random parameter realizations before they are tested by the full model. The surrogate model is regularly updated as the ensemble size grows. In the field of reliability engineering, the surrogate model is often combined with active learning to find the separation boundary between safe and fail parameter regions (Echard et al. 2011; Cadini et al. 2014; Xiao et al. 2020).

The second route is to train the initial surrogate model from a small random sample of snapshots without active learning and start the selection of parameter realizations using the surrogate model for preselection. Initially, the surrogate model will not be very accurate in predicting whether a parameter-set is behavioral, but as the ensemble size of behavioral model runs grows, the surrogate model is updated and notably improved. Hence, in contrast to active learning, this learning is passive, non-targeted, or on-the-fly. To our best knowledge, there are few examples of proactively seeking the behavioral boundary with active learning, and no comparison has been made between active learning and on-the-fly learning for large-scale environmental models.

The scope of this paper is to illustrate the efficiency of Gaussian-Process-Emulator-based surrogate models for selecting behavioral parameter-sets in subsurface-flow applications in the context of ensemble-based uncertainty quantification and global-sensitivity analysis. We will compare active and passive training methods, targeting the plausibility (being behavioral) of model results, and show how Gaussian-Process-Emulator-based surrogate models can be used to construct local sensitivities needed in the active-subspace method of global-sensitivity analysis. Our application problem is a catchment-scale subsurface flow model with uncertain material properties, boundary conditions, and geometric descriptors of the geological structure.

The rest of the paper is structured as follows. In Sect. 2 we describe the theoretical background and methods used. This section includes both the theory of the surrogate model and the global sensitivity analysis, as well as the detailed description of our suggested sampling schemes. Following this, Sect. 3 outlines the two test-cases to which the sampling schemes are applied, while Sect. 4 presents the results. The paper finishes with discussions and conclusions in Sect. 5

2 Theory and method development

2.1 Subsurface flow

Our example application for the general approach presented in this work targets regional-scale subsurface flow. Variably saturated flow in the subsurface is commonly described by the Richards (1931) equation:

$$S_w S_s \frac{\partial h_p}{\partial t} + \theta_s \frac{\partial S_w}{\partial t} + \nabla \cdot \mathbf{q} = Q \tag{1}$$

$$\mathbf{q} = -\mathbf{K} k_r \nabla (h_p + z) \tag{2}$$

in which S_w [-] is the water saturation [-], S_s [1/L] is the specific storage coefficient, h_p [L] is the pressure head, θ_s the saturated water content, or porosity, Q [1/T] denotes volumetric sources and sinks, $\mathbf{K}$ [L/T] is the saturated-hydraulic-conductivity tensor, k_r [-] denotes the relative permeability, which depends on h_p, and z [L] is the vertical coordinate. In this work, the dependence of the water saturation S_w and the relative permeability k_r on the pressure head h_p is modelled by the standard Mualem-van Genuchten parameterization (Mualem 1976; Van Genuchten 1980):

$$S_w = \begin{cases} S_{wr} + (1 - S_{wr})\left(1 + \left|\alpha h_p\right|^n\right)^m & \text{if } h_p < 0 \\ 1 & \text{otherwise} \end{cases} \tag{3}$$

$$k_r = S_e^{0.5}\left(1 - \left(1 - S_e^{1/m}\right)^m\right)^2 \tag{4}$$

$$\text{with } S_e = \frac{S_w - S_{wr}}{1 - S_{wr}} \tag{5}$$

in which S_{wr} [-] is the residual water saturation, α [1/L], n [-], and m [-] are shape parameters with $m = 1 - 1/n$, and S_e [-] is the effective water saturation.

As boundary conditions, we either directly prescribe the pressure head at the boundary node ($h_{p,i}$), or prescribe/compute a flux Q_i across the boudary in three different ways:

$$h_{p,i} = h_{p,ref} \quad \text{on } \Gamma_D \tag{6}$$

$$Q_i = Q_{ref} \quad \text{on } \Gamma_N \tag{7}$$

$$Q_i = K_{ref} \cdot (h_{p,i} - h_{p,ref}) / \Delta x_{ref} \quad \text{on } \Gamma_R \tag{8}$$

$$Q_i = C_{ref} \cdot (h_{p,i} - h_{p,ref}) \cdot H(h_{p,i} - h_{p,ref}) \quad \text{on } \Gamma_{drain} \tag{9}$$

in which the subscript *ref* denotes a user-specified reference value, i is the index of the boundary node, K_{ref} [L/T] is the conductivity between the boundary and the reference node, Δx [L] is the corresponding separation distance, C [L^2/T] is the equivalent conductance, and $H(\cdot)$ is the Heaviside function. Equations 6–9 define all boundary conditions considered in this work with $\varGamma_j$ denoting the boundary section of type j. $\varGamma_D$, $\varGamma_N$, $\varGamma_R$, and $\varGamma_{drain}$ are known as Dirichlet, Neumann, Robin, and drainage boundaries, respectively, and the total boundary of the domain is $\varGamma = \varGamma_D \cup \varGamma_N \cup \varGamma_R \cup \varGamma_{drain}$.

2.2 Gaussian process emulator

Surrogate models replace computationally demanding models by a quick-to-evaluate approximate model that can predict a specific response of the full model without actually calling it. Gaussian Process Emulators (GPE) are widely used as surrogate models (Bastos and O'Hagan 2009; Busby 2009; Loeppky et al. 2009). In the hydrological community, a GPE model may best be explained as a kriging estimator in parameter space (see Kitanidis 1997).

We denote the vector of parameters $\mathbf{x}$ and the corresponding (here: scalar) model response $y(\mathbf{x})$. After training, the original model is replaced by an interpolation of the training data, assuming that the model $y(\mathbf{x})$ can be replaced by a multi-Gaussian field $g(\mathbf{x})$ over the parameters $\mathbf{x}$, conditioned to exactly meet the actual full-model evaluations $y_i(\mathbf{x}_i)$. All assumptions of ordinary kriging apply (in particular second-order stationary of the unconditional field and diffuse prior knowledge about deterministic trend coefficients), but the spatial coordinates used in interpolation by kriging are replaced by the parameter values.

Given a set $[\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{n_O}]^T$ of n_O training samples with the corresponding model responses $\mathbf{y} = [y_1, y_2, \ldots, y_{n_O}]^T$ of the full model, we estimate a deterministic uniform trend coefficient b and the vector of structural parameters θ of a covariance function $Q(\varDelta\mathbf{x}|\boldsymbol{\theta})$ of a given functional form by maximizing the likelihood:

$$p(b, \boldsymbol{\theta}|\mathbf{y}) \propto ||\mathbf{Q}||^{-1/2} \exp\left(-\frac{1}{2}(\mathbf{y} - \mathbf{1}b)^T \mathbf{Q}^{-1} (\mathbf{y} - \mathbf{1}b)\right) \tag{10}$$

in which $||\mathbf{Q}||$ is the determinant of the $n_O \times n_O$ covariance matrix $\mathbf{Q}$, the $n_O \times 1$ vector $\mathbf{1}$ consists only of unit entries, and the element $Q_{i,j}$ of $\mathbf{Q}$ is evaluated by the covariance-function model $Q(\mathbf{x}_i - \mathbf{x}_j|\boldsymbol{\theta})$. Typical structural parameters are the prior variance and a set of correlation lengths quantifying the distance in parameter space over which the correlation vanishes.

Assuming a uniform mean may be seen as the special case of a trend model $\mathbf{Xb}$ with the $n_O \times n_b$ matrix $\mathbf{X}$ of trend functions discretized at the points $\mathbf{x}_i$, and the $n_b \times 1$ vector of trend coefficients $\mathbf{b}$, in which n_b is the number of distinct trend functions considered. If the purpose of the surrogate model was to completely replace the full model, identifying the most suitable set of trend functions and covariance-function model would be advisable. In the given context, the surrogate model is only used in an intermediate step, and we deem the extra effort of optimizing the functional shape of trend models and covariance functions unnecessary. But of course, when there is good evidence that a well-selected trend model within $\mathbf{X}$ is helpful to better approximate the model with less training points for the GPE, then the framework here is easily extended.

The interpolation $g(\mathbf{x}_c)$ at the parameter point $\mathbf{x}_c$ is then achieved by solving the kriging system of equations, which we may write in its function-estimate form as (Kitanidis 1997):

$$\hat{\mu}_g(\mathbf{x}_c) = b + \mathbf{Q}_x \xi \tag{11}$$

$$\begin{bmatrix} \mathbf{Q} & \mathbf{1} \\ \mathbf{1}^T & 0 \end{bmatrix} \begin{bmatrix} \xi \\ b \end{bmatrix} = \begin{bmatrix} \mathbf{y} \\ 0 \end{bmatrix} \tag{12}$$

in which $\hat{\mu}_g(\mathbf{x}_c)$ is the best estimate, or conditional mean, of $g(\mathbf{x}_c)$, the $1 \times n_0$ vector $\mathbf{Q}_x$ has the elements $Q_x(i) = Q(\mathbf{x}_c - \mathbf{x}_i|\boldsymbol{\theta})$, and ξ is a $n_O \times 1$ vector of weights. The estimation variance $\hat{\sigma}_g^2(\mathbf{x}_c)$ is given by (Kitanidis 1997):

$$\hat{\sigma}_g^2(\mathbf{x}_c) = Q(\mathbf{0}|\boldsymbol{\theta}) - [\mathbf{Q}_x, 1] \begin{bmatrix} \mathbf{Q} & \mathbf{1} \\ \mathbf{1}^T & 0 \end{bmatrix}^{-1} [\mathbf{Q}_x, 1]^T \tag{13}$$

The kriging estimate $\hat{\mu}_g(\mathbf{x}_c)$ is the surrogate model, which is quickly evaluated for any point in parameter space if the structural parameters $\boldsymbol{\theta}$ of the covariance function are known.

The most computationally-expensive part is the estimation of $\boldsymbol{\theta}$, here written as the maximization of $p(b, \boldsymbol{\theta}|\mathbf{y})$ according to Eq. 10. In this work, we use the STK-toolbox for MATLAB (Bect et al. 2017) to construct the GPE surrogate model, which performs the estimation of $\boldsymbol{\theta}$ by the restricted maximum likelihood method (Patterson and Thompson 1971).

It may be worth noting, that the GPE procedure is often written in a form that is identical to the standard Kalman filter, or simple kriging, which requires that the trend coefficient b is known prior to considering any training data. Because we only have diffuse prior knowledge about b, the ordinary kriging equations listed above apply. As stated above, it would be possible to replace the uniform trend coefficient b with a deterministic trend model that depends on $\mathbf{x}$. However, already a linear trend would require estimating as many additional trend coefficients as structural parameters $\boldsymbol{\theta}$. As, in our approach, the GPE-models are re-trained at regular intervals throughout the sampling procedure, this could notably increase the computational demand for parameter estimation.

2.3 Probability of misclassification

The estimation variance $\hat{\sigma}_g^2(\mathbf{x}_c)$ expresses the uncertainty of the estimate and is needed in active training. In particular, we may be interested in the cumulative probability $P(y_t(\mathbf{x}_c))$ that the true model response $y(\mathbf{x}_c)$ is not greater than a target value y_t, which we approximate by the Gaussian distribution of the conditional estimate:

$$P(y_t(\mathbf{x}_c)) = \frac{1}{2}\left(1 + \operatorname{erf}\left(\frac{y_t - \hat{\mu}_g(\mathbf{x}_c)}{\hat{\sigma}_g(\mathbf{x}_c)\sqrt{2}}\right)\right) \tag{14}$$

Then the probability of misclassification $P_{mc}(\mathbf{x}_c)$ whether the approximated value of $y(\mathbf{x}_c)$ is greater or less than the target value y_t is:

$$P_{mc}(\mathbf{x}_c) = 2P(y_t(\mathbf{x}_c))(1 - P(y_t(\mathbf{x}_c))) \tag{15}$$

which is useful to evaluate the uncertainty of the boundary in parameter space where the model response meets an inequality condition $y(\mathbf{x}) \leq y_t$ or $y(\mathbf{x}) \geq y_t$. For applications using multiple observations (i.e. multiple GPEs), we consider the product of the individual probabilities.

2.4 Sampling schemes

In this work, we consider two different sampling schemes, both applying a two-stage acceptance procedure. For each of the targets found in Sect. 3.1, we set up one GPE-model. The GPE-surrogate models, mimicking the targets, are used for the stage-1 acceptance of candidate parameter-sets, and only the stage-1 accepted sets are run using the full model. Stage-2 acceptance is then achieved if also the full model shows a behavioral result. To judge the efficiency of the sampling schemes, we define an acceptance ratio as the number of stage-2 accepted samples divided by the number of stage-1 accepted samples. The two sampling schemes differ in the training of the GPE model. We call them "on-the-fly" (passive) and "active-learning" samplers. In the on-the-fly sampler, the GPE is trained during the subsequent actual sampling, implying that it will initially perform rather poorly and then gradually improve as the number of simulated stage-1 accepted samples, and hence the training data, grows. The active-learning sampling scheme, by contrast, starts with a non-random training period, in which the GPE learns about the boundary between accepted and rejected parameter-sets. Because the samples from the active-learning period are not drawn randomly, we must not use them in the actual sampling. This implies comparably high initial costs, but the efficiency of the actual sampling is expected to be high from the beginning on. The two sampling schemes are described in detail below.

2.4.1 Learning on-the-fly

This sampling scheme is based on the following steps:

1. Draw an initial set of random parameter-sets and run the flow model to obtain the observations. Here, we sample from a uniform prior and use Latin Hypercube sampling to draw 50 parameter-sets.
2. Set up and train one GPE for each target based on the current set of parameters and observations.
3. Suggest a new candidate point, drawn randomly from the allowed parameter space. Use the GPE to approximate the observations and their corresponding uncertainty. Compute, for each target, the probability of the candidate point being on the behavioral side of the targets. Finally, compute the joint probability of all targets being met by taking the product of the individual probabilities.

   (a) Accept the candidate point at stage-1 if the joint probability is larger than a given threshold and run the full flow model to check whether the candidate point is actually behavioral (stage-2 accepted). As a threshold, we use 0.5, which is chosen based on initial tests of the model. A higher value would generate a higher fraction of stage-1 accepted points that are also stage-2 accepted, but risk poor sampling quality in the boundary regions of the behavioral parameter space, while a lower number would risk resulting in low sampling efficiency.

   If not accepted,

   (b) Reject the candidate point. As the GPE at early times may not have a high quality, this decision may be wrong. Therefore, rejected candidates are stored in an archive, which is maintained throughout the full sampling, and the corresponding probabilities are updated whenever the GPE is retrained in this algorithm. If the probabilities exceed the threshold upon retraining of the GPE, they are included in the set of stage-1 accepted samples and are run with the full flow model. For practical reasons, the archive may be limited to a maximum number of parameter-sets with the highest (but still rejected) probabilities. In this work, we store 10,000 samples in the archive. This value was chosen since it is small enough to prevent storage or memory issues, but large enough to hold the rejected parameter-sets with the highest probabilities that may become stage-1 accepted upon retraining the GPE models. As long as the number is kept reasonably large, the choice is not critical and can be fitted to the user's available resources.

4. Redo point 3 until enough candidate points are accepted (stage-1) and run with the full model, then return to point 2 and retrain the GPE using the full collection of stage-1 accepted datasets sampled so far. Here, a variable threshold is used that ranges from every 10th accepted candidate (as long as the total number of accepted parameter-sets is below 500) to every 100th (when more than 3000 parameter-sets have been stage-1 accepted). The variable threshold is used to find a good balance between keeping the GPE updated and keeping the time spent on training it as low as possible.
5. Check whether the sample is large enough to finish the sampling. This is commonly done by checking the number of stage-2 accepted samples (here: 3000 samples are required, which is deemed large enough to easily perform the global sensitivity analysis with).

2.4.2 Active learning

In the active learning scheme, the actual sampling is preceded by a non-random sampling targeted at exploring the boundaries between behavioral and non-behavioral regions of the parameter space. Once this phase is finished, the resulting surrogate model should be of high quality and immediately make confident predictions on the behavioral status of a suggested parameter-set. After this initial learning boost, we return, for the remaining sampling process, to the on-the-fly sampling scheme described above. The active learning starts between points 1 and 2 of the scheme outlined for the on-the-fly sampler. The scheme consists of the following steps:

1. Setup and train one GPE for each target based on the current set of parameters and observations.
2. Find a parameter-set for which the misclassification probability according to Eq. 15 is the highest. This is done by drawing a large number of random candidate parameter-sets, evaluating the best estimate $\hat{\mu}_g$, then computing the misclassification probability P_{mc} for each candidate by Eq. 15, and picking the one with the highest value of P_{mc}. In this work, we use 100,000 random candidate parameter-sets. The size is not critical but should be large enough to allow for the identification of a candidate parameter-set with a high misclassification probability. 100,000 appeared to be a good compromise between the time required to evaluate the GPE-models and the chance to find good candidates.
3. Run the full flow model with the new parameter-set and return to point 1.
4. Keep adding new parameter-sets according to steps 2 and 3 until the surrogate model is good enough. In theory, this would be the case when the highest found misclassification probability does not exceed a preset threshold value. In practice, we found that both setting and achieving a good threshold value is difficult and requires an often too large number of model evaluations to be a usable statistic. In the present application, we therefore chose to finish the active-learning phase when the mean stage-2 acceptance ratio over the last 100 parameter-sets reached a value of 0.5. This represents a proxy for a stable sample, as half the samples are rejected and half accepted, which is what we would expect when correctly sampling the boundary between behavioral and non-behavioral space. Please note that we only check stage-2 acceptance to judge convergence, as this property is normally not needed during the actual active-learning phase, and that the acceptance criterion is measured as number of stage-2 accepted samples divided by the total number of model evaluations.
5. Continue with point 3 of the on-the-fly sampling scheme. The only differences to the description above is that (1) the training dataset now includes also the active-learning dataset, (2) the re-training of the GPE occurs on regular intervals (here: every 500 runs) and (3) the probability $P(y_t(\mathbf{x}_c))$ required to accept a candidate parameter-set $\mathbf{x}_c$ is raised to 0.7. It is important to note, however, that while the active-learning dataset is used for the training of the GPE, we don't include it in the final ensemble, even though it contains stage-2 accepted samples. The reason for this is that these samples are not drawn randomly; they are specifically targeted at the boundaries of the behavioral regions so that including the stage-2 accepted samples of the active-learning period would lead to a bias in the final ensemble towards these boundaries.

As a full model run is computationally expensive, we perform the computations on a mid-range cluster, running multiple instances of the flow model in parallel. However, flow models with different parameter-sets can take widely different computation times, so that waiting for a specific simulation to be finished before drawing new parameters would be inefficient. In the on-the-fly sampling scheme, this is just a matter of technical implementation. The active-learning scheme, by contrast, has originally been designed for sequential learning. To avoid waiting for model results of preceding parameter-sets, we thus temporarily approximate them by the expected value $\hat{\mu}_g$ of the GPE surrogate model and proceed as if this were the simulation results of the full model. As soon as the true observations are available, the expected values of the GPE model are replaced with true results. While this approach may deem the active learning sub-optimal in comparison to a sequential approach, the improvement in terms of wall-clock time is as large as the total number of jobs that can be run simultaneously, which is 100 in our case.

At this point, it is important to note that a global sensitivity analysis, like the active-subspace method described below, requires an independent sample of the behavioral parameter space. When using a surrogate model for preselection, we approximate the outcome of a full flow model with the surrogate model, which yields correct decisions only if the surrogate model is of high quality. It is essential that the user is fully aware of these assumptions. For a CPU-intensive full flow model, checking the correctness of a sample is difficult. In Sect. 4.2 we therefore provide a comparison between our sampling schemes and a pure Monte Carlo sample for a simplified testbed, to assess whether our proposed schemes capture the true parameter distributions, and therefore are valid to be used in a global sensitivity analysis. It is also important to note that storing stage-1 rejected parameter-sets described in point 3b above serves the purpose of correcting wrong decisions in the preselection scheme. When a falsely rejected parameter-set is re-evaluated by the updated GPE models, we can correct the wrong-negative sampling error and maintain a good coverage of the problematic regions in parameter space. Not re-evaluating rejected parameter-sets, by contrast, would lead to a smaller coverage of the parameter space also in the final (stage-2 accepted) sample, because sampling points had been discarded at a stage when the GPE models were still inaccurate. This would lead to a bias in the final sample, that can easily be avoided by re-evaluating the samples as the GPE-models evolve.

2.5 Global sensitivity analysis by active subspaces

The purpose of a global sensitivity analysis is to evaluate the relative impact of different parameters across the entire parameter space. In the last decades, various global-sensitivity-analysis methods have been proposed, including variance-based methods (Saltelli et al. 2010), derivative-based methods (Sobol' and Kucherenko 2009), classification-based methods (Spear and Hornberger 1980; Xiao and Lu 2017), and the methods based on active subspaces (Constantine and Diaz 2017). Each of these methods would benefit from an ensemble acceleration by active or passive learning. Following our past positive experience with active subspaces (Erdal and Cirpka 2019), and the added benefit of the GPE supplying also the required gradients (see below), we demonstrate the advantages of the GPE-based preselection scheme when using active subspaces for global sensitivity analysis. This method takes local sensitivities to construct a sorted basis of orthonormal directions in parameter space with decreasing influence on the model outcome. An active subspace is defined by the eigenvectors of the $n_P \times n_P$ matrix:

$$\mathbf{C} = \int \nabla f(\tilde{\mathbf{x}}) \otimes \nabla f(\tilde{\mathbf{x}}) \rho(\tilde{\mathbf{x}}) \mathrm{d}\tilde{\mathbf{x}} \approx \frac{1}{n} \sum_{i=1}^{n} \nabla f(\tilde{\mathbf{x}}_i) \otimes \nabla f(\tilde{\mathbf{x}}_i) \tag{16}$$

in which f denotes the model, $\tilde{\mathbf{x}}$ is the vector of normalized parameters, each scaled between 0 and 1, $\nabla f(\tilde{\mathbf{x}})$ is the vector of partial derivatives $\partial f / \partial \tilde{x}_i$, ρ is the prior probability density function of the normalized parameters, n is the number of observations, and the last part of the equation describes the approximation of the integral by Monte Carlo sampling. The eigen-decomposition of $\mathbf{C}$ reads as:

$$\mathbf{C} = \mathbf{W} \boldsymbol{\Lambda} \mathbf{W}^{-1} \tag{17}$$

in which $\boldsymbol{\Lambda}$ is the diagonal matrix of the eigenvalues λ_i, and $\mathbf{W}$ is the matrix of corresponding eigenvectors $\mathbf{w}_i$. From the eigen-decomposition, we can compute the square root of the activity score a_j (Constantine and Diaz 2017) which is used as a global sensitivity metric of parameter j:

$$a_j = \sqrt{\sum_{k=1}^{m} \lambda_k w_{j,k}^2}, \tag{18}$$

in which j is the parameter index, m is the number of subspace dimensions (i.e. number of eigenvectors), λ_k is the k-th eigenvalue and $w_{j,k}$ the value for parameter j in the k-th eigenvector.

A major issue in the use of active subspaces is evaluating the gradients of the observation f with respect to the scaled parameters $\tilde{\mathbf{x}}$. In theory, this would require a local sensitivity analysis at each evaluation point in parameter space. For large-scale complex models, these gradients are not readily available, and direct numerical differentiation is not an option if there are many parameters and an individual model run is computationally expensive. The common approach when using the active-subspace method in subsurface modeling is to approximate the gradients using a linear (Gilbert et al. 2016; Jefferson et al. 2015) or higher-order polynomial trend surface (Erdal and Cirpka 2019; Oladyshkin et al. 2012), which is fitted to all training data but may give a poor approximation of the local gradients.

In the given context, we use a GPE surrogate-model which not only yields a very reasonable approximation of the real model after sufficient training, but also a direct estimate of the sensitivity of the model with respect to the parameter vector $\mathbf{x}$ at any evaluation point $\mathbf{x}_c$ by taking the gradient of Eq. 11:

$$\nabla \hat{\mu}_g(\mathbf{x}_c) = (\nabla \otimes \mathbf{Q}_x)\boldsymbol{\xi} \tag{19}$$

in which $\nabla \otimes \mathbf{Q}_x$ is the matrix of gradient vectors of all elements of $\mathbf{Q}_x$ with respect to the normalized parameters $\tilde{\mathbf{x}}$, requiring the derivatives of the known, differentiable covariance function $Q(\varDelta\mathbf{x}|\boldsymbol{\theta})$ with respect to $\varDelta x_i$. The existence of these gradients restricts the GPE to be used with differentiable covariance functions.

2.6 Choice of the covariance function

The GPE surrogate model depends on the choice of the covariance function $Q(\varDelta\mathbf{x}|\boldsymbol{\theta})$, which determines the functional shape of the estimate. The same covariance functions are available that are used in geostatistical interpolation of spatial variables (e.g., the exponential and squared exponential covariance functions, the family of Matérn covariance functions, power-law functions, etc. (see Rasmussen and Williams 2006, Chapter 4)). As the interpolation is in parameter space, it is typical to choose a covariance function that has a derivative of zero at the origin to guarantee smoothness. Most likely, the squared exponential covariance function is the most widely used one in GPE modeling. Stein (1999), however, argues that the strong smoothness of the squared exponential function is not suitable for many physical processes, and recommends using a member of the Matérn family of covariance functions. Following the recommendations by Stein (1999) and the results of our preliminary testing, we chose the anisotropic Matérn covariance function of order 3/2. The covariance function is implemented in the STK toolbox (Bect et al. 2017) in the following way:

$$Q(\varDelta\mathbf{x}|\boldsymbol{\theta}) = \sigma^2\left(1 + 2\sqrt{\frac{3}{2}}d\right)\exp\left(-2\sqrt{\frac{3}{2}}d\right) \tag{20}$$

$$\text{with } d = \sqrt{\sum_j^{n_P}\left(\frac{\varDelta x_j}{\ell_j}\right)^2} \tag{21}$$

in which σ^2 denotes the variance, d is the scaled separation distance, n_P is the number of parameters, and ℓ_j is a scaling length for parameter x_j. The vector $\boldsymbol{\theta}$ of structural parameters, to be estimated in the GPE training, thus consists of σ^2 and the 32 scaling lengths ℓ_j, one for each parameter x_j (see further Sect. 3.1). The STK-toolbox estimates these parameter by the restricted maximum likelihood method (see further Sect. 2.2).

As pointed out in Sect. 2.5, we need the gradient of the model response $y(\mathbf{x})$ at all evaluation points of the model for the construction of the active subspace. We have already discussed that this can be approximated by the gradient of the GPE surrogate model according to Eq. 19. Substituting Eq. 20 into Eq. 19 yields:

$$\begin{aligned}
&\frac{\partial \hat{\mu}_g(\mathbf{x})}{\partial x_j} = -\sum_i^{no}\xi_i\frac{\partial (Q_x)_i}{\partial x_j} = -\sum_i^{no}\xi_i\frac{\partial (Q_x)_i}{\partial d_i}\frac{\partial d_i}{\partial x_j}\\
&\text{with}\,\frac{\partial (Q_x)_i}{\partial d_i} = -6\sigma^2 d_i\exp\left(-2\sqrt{(3/2)}d_i\right)\\
&\text{and}\,\frac{\partial d_i}{\partial x_j} = \frac{x_j - x_{i,j}}{d_i\ell_j^2}
\end{aligned} \tag{22}$$

in which d_i is the scaled distance between the evaluation point $\mathbf{x}$ and the training point $\mathbf{x}_i$ and $x_{i,j}$ is the j-th parameter in parameter-set $\mathbf{x}_i$. If the surrogate model is a good representation of the real model, the corresponding gradients are also expected to be good approximations.

3 Test cases

3.1 Subsurface model mimicking the Käsbach catchment

Throughout this manuscript, we use a catchment-scale flow model as a testbed for the sampling schemes presented later. We solve the Richards equation (Eq. 1) with HydroGeoSphere (Aquanty Inc. 2015), which uses lowest-order conforming finite elements on triangular prisms for spatial discretization and implicit Euler integration for temporal discretization. The test application is taken from Erdal and Cirpka (2019). The model domain mimics the Käsbach catchment close to Tübingen in southwest Germany. An illustration of the model domain and its setup is shown in Fig. 1.

As can be seen from Fig. 1, the subsurface consists of five different geological layers, which we assume to be internally homogeneous in the model. The bottom layer is the Middle Triassic Upper Muschelkalk, a karstified limestone formation, overlain by the lower Upper Triassic Erfurt formation, consisting of clayey mudstones and carbonate-rocks. The Erfurt formation is overlain by the middle Upper Triassic Grabfeld formation, made of interbedded mudstones and gypsum layers. While the original Grabfeld formation is comparably low-permeable, it becomes more conductive upon weathering so that we consider a weathered layer of uncertain thickness. In the valley center, unconsolidated Quaternary sediments persist. In the following we abbreviate the layers, from the bottom up, as mo, ku, km1, km1-w, and Q. A description of the regional settings can be found in D'Affonseca et al. (2020). Figure 1 shows that a fault line with an uncertain vertical offset passes approximately in the north-south direction through the model domain. Also, not all geological layers

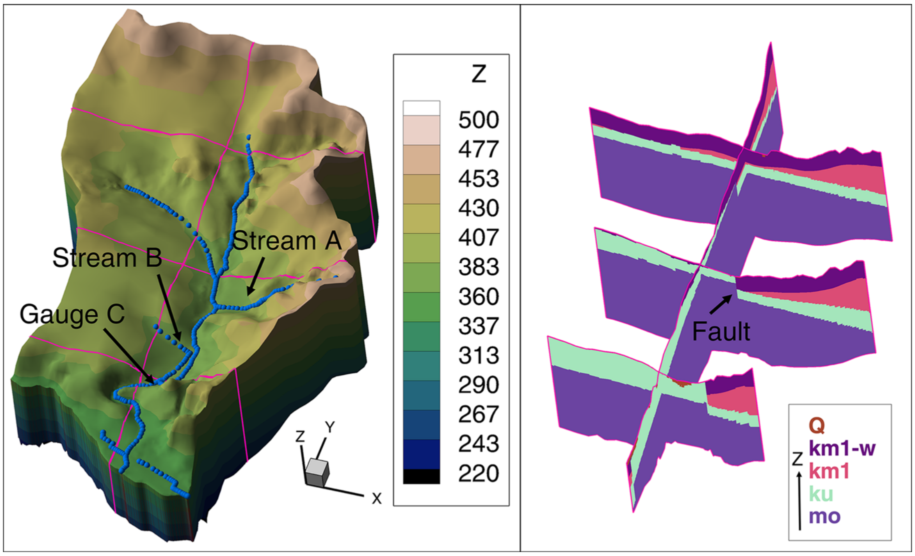

Fig. 1 Overview of the Käsbach catchment. Left: model domain with topography and streams; right: example of a geological realization

are present at all locations. We address the uncertainty in layer thicknesses and the fault offset by considering these geometric parameters as stochastic variables.

The model has a major stream and four tributaries, all modeled as drains (Eq. 9), implying that water can leave the groundwater domain if a reference pressure-head is exceeded, but it can never enter through this boundary. To account for the possibility of overland flow, we define a second drain boundary with a higher reference pressure-head at the top of the model domain wherever there are no streams. The other boundary conditions are: three Neumann boundary conditions (Eq. 7) on top of the domain, representing the incoming recharge for three different land uses (grassland, cropland, and forest), a Dirichlet boundary condition (Eq. 6) at the bottom of the domain representing inflow from the surroundings, and a Robin boundary condition (Eq. 8) at the southern vertical boundary representing the Ammer river into which the Käsbach river feeds. All other boundaries are modeled as no-flow boundaries.

In total, the model has 32 uncertain parameters, which are listed and explained in Table 1. Further information about the model and parameters are given by Erdal and Cirpka (2019). The model is run in transient mode with constant boundary conditions until steady state is reached. On a mid-range cluster, one model run takes about four CPU-hours.

A key problem in setting up the original model was that many parameter combinations, in which the individual parameter values were taken from plausible ranges, led to implausible model results. In an ensemble-based uncertainty analysis of the Käsbach model, we need to exclude such non-behavioral parameter combinations. Towards this end, we have defined a set of five target quantities, simulated by the model, that decide whether the parameter combination can be accepted (behavioral) or must be rejected (non-behavioral). In this work, we use the same targets as Erdal and Cirpka (2019), with slightly modified values to further decrease the behavioral part of the parameter space:

1. Maximum of 2×10^{-3} m^3/s of water leaving the domain on the top, outside of the streams (requesting no flooding).
2. Between 25 and 60% of the incoming water should leave the domain via the streams.
3. At Gauge C, the main stream should have a minimum discharge of 5×10^{-3} m^3/s
4. Stream A should have a maximum discharge of 3×10^{-3} m^3/s.
5. Stream B should have a minimum discharge of 5×10^{-6} m^3/s

3.2 Simplified testbeds

Although the HydroGeoSphere flow model presented above is reasonably fast compared to many similar and realistic catchment models (wall-clock time 1 h), the number of simulations that can be done in our study is limited to a few thousand. Hence, it is not possible to perform a pure Monte-Carlo rejection sampling as a true reference. To still allow for a more rigorous test of the suggested sampling schemes, we consider two simplified testbeds. The first is the surrogate model based on the active-subspace decomposition of the 10,000 flow simulations performed by Erdal and Cirpka (2019), while the second one is the GPE-surrogate model resulting from the on-the-fly sampling scheme performed and presented in this work. Hence, the simplified testbeds have the same

Table 1 List of uncertain parameters and ranges

ID	Parameter	Where	Equations	Unit	Max	Min
1.	Offset $h_{p,ref}$-fixed	Domain bottom	6	m	5	−5
2.	Fault height	mo-ku interface	–	m	100	0
3.	Interface offset	ku-km1 interface	–	m	20	−20
4.	Layer thickness	km1-weathered	–	m	50	5
5.	$h_{p,ref}$-drain	Streams	9	m	0.2	0.005
6.	$h_{p,ref}$-Robin	Southern exit	8	m	355	335
7–9.	Q_{ref}	Domain top	7	mm/year	150	80
10–14.	K	All geological units	1	m/s	10^{-5}	10^{-9}
15–17.	K_z-ratio	km1-weathered, km1, ku	–	–	50	1
18–22.	α	All geological units	3	1/m	5	0.5
23–27.	n	All geological units	3	–	9	1.5
28–32.	S_s	All geological units	1	1/m	10^{-4}	10^{-6}

input parameters as the real testbed, but the run time to obtain an observation is almost negligible. Arguably, the first idealized testbed is the simplest, as the active-subspace based surrogate model projects the 32 input parameter to two active variables which are subsequently used to obtain the observation from an estimated 2-D surface. Conversely, the GPE-based surrogate model remains an interpolation in 32-dimensional parameter space and could represent more complex relations. Using a GPE as a simplified testbed, while using the same GPE-setup in the proxy model, creates a simplified environment. In essence, it means that the true model can perfectly be modeled by the proxy model, which normally is not the case. This could lead to inflated acceptance rates, but should not affect the efficiency compared to the Monte-Carlo sampling.

3.3 Prior work

In a preceding study (Erdal and Cirpka 2019), we used the method of active subspaces (Constantine et al. 2014; Constantine and Diaz 2017) to both perform the global sensitivity analysis and aid the Monte-Carlo sampling of the behavioral parameter space. This resulted in a sample of 10,000 HydroGeoSphere simulations, out of which ≈ 2000 fulfilled all targets listed above. Our current goal will be to pursue a much higher efficiency, hoping to achieve some fraction close to 100% after training the GPE. The prior study showed that most of the 32 parameters were insensitive to the selected behavioral targets (e.g., all parameters related to the unsaturated flow regime), but that the sensitive ones had a large impact on subsurface flow in the model. The previous work also showed that the correlation between parameters strongly influenced the sampling, so that it is very difficult to predict the sampling result by expert knowledge. Hence, it is highly advisable to apply an automated search to obtain the joint distribution of behavioral parameters.

In this work, we will make use of the previous findings in three ways. First, we will compare the new GPE-based sampling schemes to the scheme of Erdal and Cirpka (2019), testing if they generate similar parameter distributions. Second, we will use the active-subspace surrogate model of Erdal and Cirpka (2019) to extensively test the GPE-based sampling scheme, by replacing the full HydroGeoSphere model with the active-subspace surrogate. This is done to show that the new sampling scheme can produce similar results as a pure Monte-Carlo sampling in the limit of extremely large ensembles, which is not possible to perform on the full HydroGeoSphere model (see further Sect. 3.2). Third, we compare the results of the global sensitivity analysis of Erdal and Cirpka (2019) to that based on the GPE-based sampling schemes to detect possible shifts in accuracy.

4 Results

4.1 Tests with the full subsurface-flow model

We start with the application to the real HydroGeoSphere flow model, aiming for 3000 stage-2 accepted parameter-sets. Figure 2 shows the acceptance statistics of the two GPE-based sampling schemes (blue: on-the-fly sampling, red: active learning) and the original scheme of Erdal and Cirpka (2019) (green). The latter scheme, which is based on a polynomial fit in two active subspaces, approaches an acceptance ratio of about 20%, while the two GPE-based sampling schemes end around 70%. In the GPE-scheme with active learning, the active-learning part extended over 680 samples. In this period, no samples can

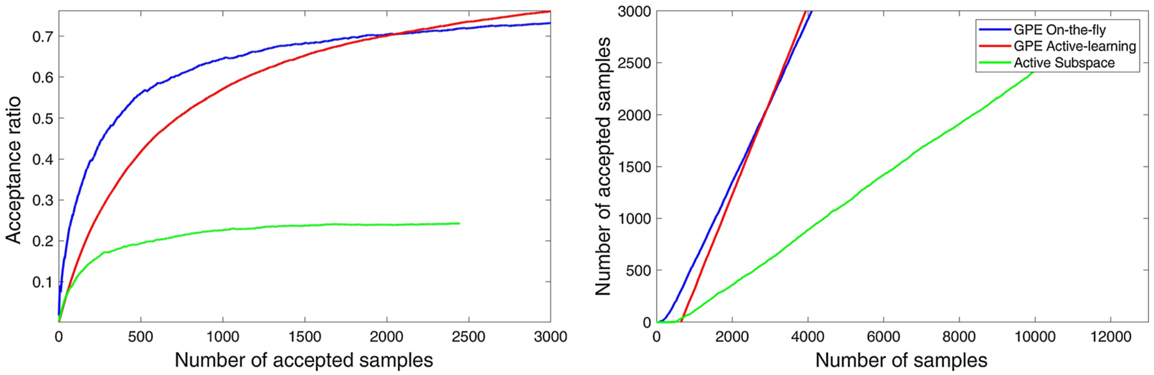

Fig. 2 Performance of the sampling methods in application to the full subsurface-flow model. Left: Stage-2 acceptance ratio as a function of the number of stage-2 accepted samples; right: number of stage-2 accepted samples as a function of the number of stage-1 accepted samples. Blue: GPE-based on-the-fly sampler; red: GPE sampler with preceding active-learning phase; green: active-subspace based sampler of Erdal and Cirpka (2019)

be accepted (zero acceptance in the right subplot), but these datasets are needed to assess the boundary of the behavioral parameter space. The crossover point between the two sampling schemes can be seen at ≈ 2000 stage-2 accepted samples (or ≈ 3000 stage-1 accepted samples). Before this point, the on-the-fly sampling scheme is more efficient, while afterward, the active-learning is more efficient. Hence, for our 3000 samples, the active learning is in principle more effective, but the difference between the schemes in the total number of full-model runs required to achieve the 3000 stage-2 accepted samples is rather small.

To assess the quality of the resulting posterior parameter distributions, it is not possible to compare the results of the sampling schemes to the true distribution that would be obtained by pure Monte-Carlo rejection sampling because the acceptance rates of the latter would be smaller than 1%. Instead, we compare the two GPE-based sampling schemes with the previously published scheme using active subspaces (Erdal and Cirpka 2019). Figure 3 shows the marginal parameter distributions resulting from the sampling schemes for three parameters with distinct distributions.

Table 2 (columns 8–10) lists the p-values of standard two-sided Kolmogorov–Smirnov tests for all 32 parameters, comparing the distributions resulting from the three sampling schemes among each other. To increase readability, p-values above 0.01 are set in bold (highlighting significant similarities), and the others are in italic. Although the Kolmogorov–Smirnov tests are not conclusive across all parameters, a visual comparison of the distributions indicates high degrees of similarity.

Figure 4 shows the square-root of activity scores (Eq. 18) for the ten most influential parameters with respect to streamflow at Gauge C (see Fig. 1) obtained by the global-sensitivity analysis using the different stage-2 accepted ensembles with 3000 samples each. We only show one observation here, since the results in the others are similar. For the results of the full sensitivity analysis

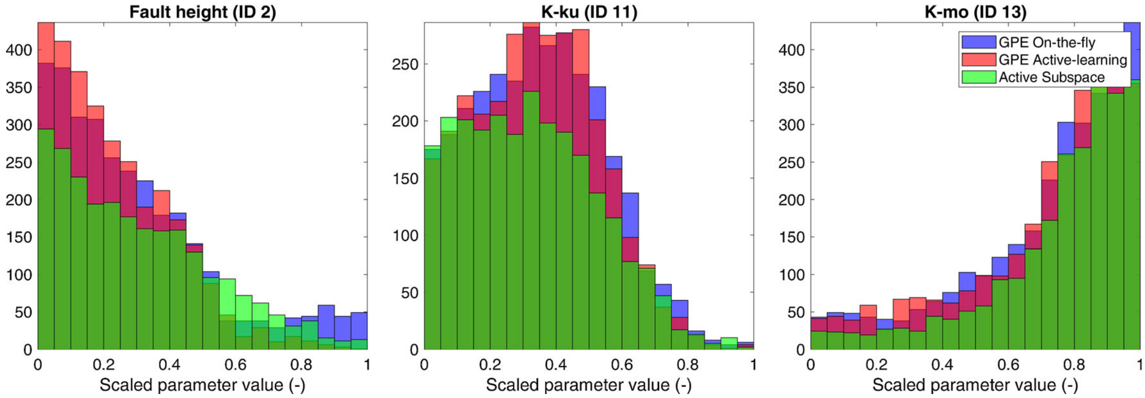

Fig. 3 Marginal distributions of three example parameters with distinct distributions, generated using the full flow model. Blue: GPE-based on-the-fly sampler; red: GPE sampler with preceding active-learning phase; green: active-subspace based sampler of Erdal and Cirpka (2019). Please note that, due to the slightly more conservative targets used in this work, the number of samples is a little bit lower for the active-subspace sampling scheme than in the two GPE-based schemes. Please also note that colors different to those in the legend are due to overlaying histogram-bars

Table 2 Quantitative comparison of the marginal distributions of all parameters by p-values of two-sided Kolmogorov–Smirnov tests. For the simplified testbeds, columns 2–3 and 5–6 compare the different GPE-based sampling schemes to their respective pure Monte Carlo sampling scheme, while columns 4 and 7 compare the pure MC sampling scheme to a uniform distribution. Columns 8–10 show the same test for the full flow model and compares different sampling schemes to one another

	Active subspace data			GPE-based data			Full flow model		
ID	OtF	AL	Uni	OtF	AL	Uni	OtF-AS	AL-AS	OtF-AL
1	**0.704**	*0.003*	*0*	**0.771**	**0.102**	*0*	**0.029**	*0.001*	*0.001*
2	**0.845**	*0.001*	*0*	*0.007*	*0*	*0*	*0*	*0*	*0*
3	**0.522**	**0.777**	*0*	**0.443**	**0.564**	*0*	*0.001*	**0.019**	**0.418**
4	**0.019**	*0.01*	*0*	**0.435**	**0.792**	*0*	*0*	*0*	**0.039**
5	**0.089**	**0.035**	*0*	**0.947**	**0.59**	*0.001*	*0.01*	*0.001*	**0.259**
6	**0.183**	**0.531**	*0*	**0.537**	**0.544**	*0*	**0.023**	**0.335**	**0.078**
7	**0.364**	**0.465**	*0*	**0.218**	**0.6**	*0*	*0*	*0*	**0.036**
8	**0.171**	*0.004*	*0*	**0.305**	**0.378**	*0*	**0.013**	**0.113**	**0.085**
9	**0.484**	**0.792**	*0.001*	**0.686**	**0.751**	*0.003*	*0*	*0*	**0.632**
10	**0.117**	**0.7**	*0*	**0.807**	**0.046**	*0*	**0.065**	**0.019**	**0.598**
11	**0.159**	*0*	*0*	**0.017**	*0*	*0*	*0*	*0*	**0.282**
12	**0.125**	**0.073**	*0*	**0.296**	**0.466**	**0.692**	**0.169**	**0.612**	**0.308**
13	**0.576**	*0.001*	*0*	**0.211**	**0.227**	*0*	**0.02**	**0.062**	**0.155**
14	**0.453**	**0.208**	*0*	**0.011**	**0.07**	*0*	**0.634**	**0.08**	**0.519**
15	**0.619**	**0.544**	*0*	**0.597**	**0.083**	*0*	**0.239**	**0.828**	**0.286**
16	**0.104**	**0.359**	*0*	**0.052**	**0.096**	**0.657**	*0.003*	*0.001*	**0.561**
17	**0.312**	**0.153**	**0.429**	**0.348**	**0.451**	**0.972**	**0.256**	**0.391**	**0.879**
18	**0.581**	**0.029**	**0.013**	**0.197**	**0.928**	**0.089**	**0.519**	**0.403**	**0.701**
19	**0.155**	**0.098**	**0.833**	**0.108**	**0.352**	**0.474**	**0.72**	**0.483**	**0.314**
20	*0*	**0.555**	**0.097**	**0.189**	**0.499**	**0.594**	**0.931**	**0.819**	**0.904**
21	**0.678**	**0.517**	**0.4**	**0.123**	**0.193**	**0.368**	*0.007*	**0.515**	*0*
22	**0.197**	**0.173**	**0.541**	**0.097**	**0.037**	**0.471**	**0.123**	**0.473**	**0.372**
23	**0.661**	**0.159**	**0.684**	**0.612**	**0.797**	**0.787**	**0.957**	**0.09**	**0.084**
24	**0.061**	**0.367**	**0.318**	**0.653**	**0.108**	**0.927**	**0.375**	**0.78**	**0.201**
25	**0.43**	**0.358**	**0.276**	**0.037**	**0.252**	**0.69**	**0.021**	**0.029**	**0.87**
26	**0.835**	**0.896**	**0.495**	**0.754**	**0.46**	**0.222**	**0.335**	**0.076**	**0.133**
27	**0.066**	**0.532**	**0.461**	**0.234**	**0.414**	**0.235**	**0.533**	**0.496**	**0.615**
28	**0.686**	**0.491**	**0.834**	**0.22**	**0.225**	**0.576**	**0.667**	**0.195**	**0.216**
29	**0.167**	**0.836**	**0.702**	**0.733**	**0.656**	**0.71**	*0.01*	*0*	**0.195**
30	**0.954**	**0.848**	**0.886**	**0.919**	**0.131**	**0.487**	**0.091**	**0.179**	**0.264**
31	**0.944**	**0.089**	**0.705**	**0.071**	**0.097**	**0.375**	**0.119**	**0.055**	**0.646**
32	**0.384**	**0.158**	**0.054**	**0.838**	**0.284**	**0.909**	**0.115**	**0.211**	**0.651**

ID refers to the ID in Table 1. To increase readability, p-values above 0.01 (i.e. acceptable results) are bold, and the others are italic. OtF, On-the-fly sampler; AL, Active-learning sampler; Uni, uniform distribution; AS, Active subspace; OtF-AS, comparison between OtF and AS (from prior work)

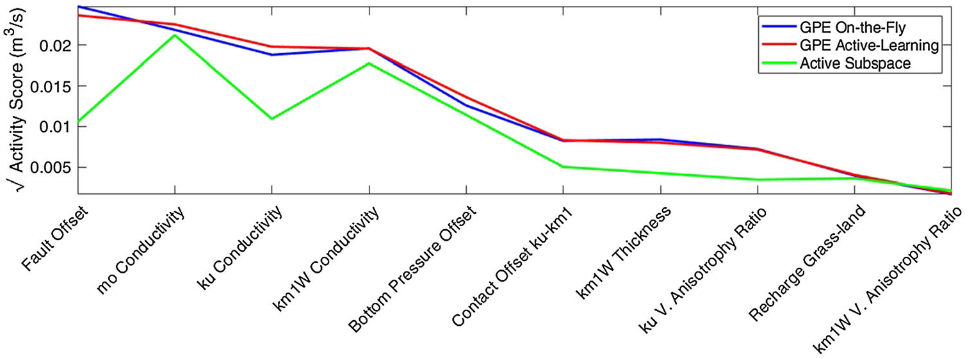

Fig. 4 Global sensitivity analysis of the full flow model. Square root of the activity score for the flow rate at gauge C as a metric of global sensitivity. Comparison of the different sampling schemes. The scores are restricted to the top ten most important parameters

and its discussion, the interested reader is referred to Erdal and Cirpka (2019). When performing the global sensitivity analysis using the two GPE-derived ensembles, we compute the gradients according to Eq. 19. Figure 4 shows the activity score for both the active-learning and the on-the-fly sampling schemes, as well as the active-subspace based sampling scheme of Erdal and Cirpka (2019). As can be seen, the two GPE-based sampling schemes yield nearly identical activity scores. The activity scores of Erdal and Cirpka (2019) are similar and give the same top five parameters. The notable differences between the active-subspace sampling scheme and the GPE-based ones are likely due to the slightly different marginal parameter distributions and the way how the local gradients are evaluated in the approach of Erdal and Cirpka (2019), in which a third-order polynomial was fitted through all behavioral parameter-sets. Independent tests of the active-subspace sampling scheme with a simplified testbed suggest that the results of the GPE-based sampling schemes are more correct. Above all, the global sensitivity analysis performed with the GPE-based ensembles and using the GPE-calculated gradients gives very plausible results at attractively low computational costs.

In summary, both sampling schemes give similar results when applied to the full flow model, with comparable final acceptance rates. While active learning is faster from a certain ensemble size onward, it adds complexity to the technical approach, and the length of the necessary active-learning phase is not known a-priori. We thus assess that the on-the-fly sampler to be the better and safer choice.

4.2 Simplified testbeds

In contrast to the full flow model, we can perform pure Monte-Carlo sampling when applying the sampling schemes to the simplified testbeds. This facilitates comparing the sampling schemes to the true posterior distribution obtained by the Monte-Carlo sampler. To achieve a sample of 3000 stage-2 accepted parameter-sets by pure Monte-Carlo sampling, the simplified testbed based on active-subspace decomposition required about 800,000 model evaluations in rejection sampling, while the simplified testbed based on the GPE-approximation required 600,000 model evaluations in rejection sampling, leading to efficiencies of 0.4% and 0.5%, respectively.

Figure 5 shows the acceptance rate of the two GPE-based sampling schemes together with that of the original active-subspace-based sampling scheme by Erdal and Cirpka (2019). As can be seen, the active-subspace sampling scheme approaches a stable acceptance rate of $\approx$ 15–20%, while the two GPE-based samplers reach acceptance ratios between 75 and 90%. The left plot of Fig. 5 shows the cross-over points, where the active-learning scheme becomes beneficial compared to the on-the-fly scheme. In both simplified testbeds, this occurs long before 3000 samples are accepted, which we defined as the requested number of accepted samples in our application. Hence, in these simplified testbeds, the active-learning scheme appears slightly preferable over the on-the-fly learning scheme. This is likely so because the boundary of the behavioral parameter space is explored at lower costs (active learning required only 200 samples) than when using the full flow model (requiring 680 samples). This notable difference in the active-learning period of the active learning GPE sampling scheme confirms that the simplified testbeds don't capture the complexity of the full flow model. The simplified testbed using a GPE surrogate model as virtual truth is in a way circular because we test whether the GPE approach can identify itself, which differs from representing a full subsurface-flow model. Another reason for the differences in the active training may be that the simulation times of the full model are fairly long. As discussed above, the full flow model needs to be run asynchronously and without waiting for any previous run to finish, whereas the original active-training procedure was designed for sequential processing of one parameter-set after the other, which is how the simplified test cases are run. This may contribute to the simplified test cases having a shorter active-learning phase than the full model.

Figure 6 shows the resulting marginal parameter distributions of the same three selected parameters as shown in Fig. 3. As can be seen, both GPE-based sampling schemes result in the correct (marginal) distributions. Table 2 (columns 2–3 and 5–6) lists the resulting p-values of standard two-sided Kolmogorov–Smirnov tests for all 32 parameters, comparing the distributions resulting from the GPE-based sampling schemes to those of pure Monte-Carlo sampling. To increase readability, p-values above 0.01 are again set in bold (highlighting significant similarities), and the others are in italic. Table 2 (columns 4 and 7) also shows the KS-metric comparing the pure Monte-Carlo samples with the uniform prior distribution (see Sect. 2.4), indicating which parameters have undergone significant selection upon the sampling. As can be seen, both GPE-based sampling schemes applied to both simplified testbeds result in marginal distributions of most parameters that are significantly similar to the true distributions. We thus conclude that both GPE-based sampling schemes work well for the simplified testbeds.

Another way of analyzing the results of the simplified testbeds is to check whether the surrogate models correctly predict whether a candidate parameter-set is behavioral or not. We test this here with the two GPE-based sampling schemes. Towards this end, we randomly choose 3000 parameter-sets that are behavioral and 3000 parameter-sets that are non-behavioral according to the truth of the

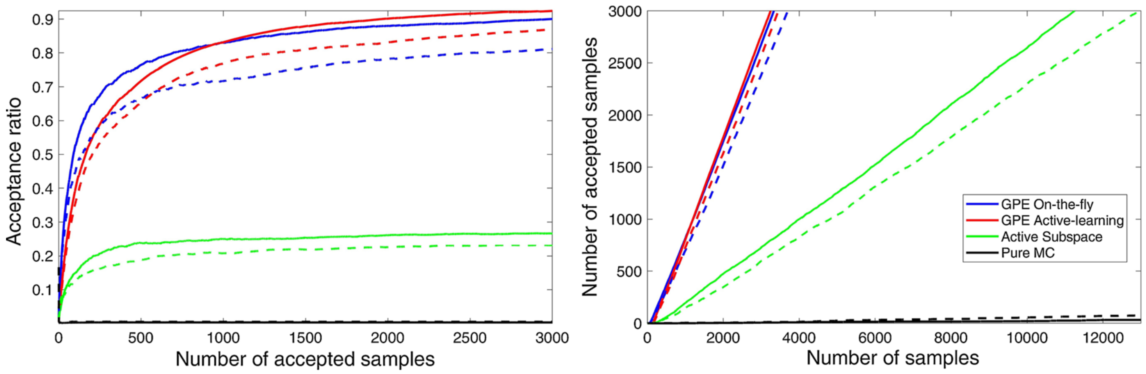

Fig. 5 Performance of the sampling methods in application to the simplified testbeds. Left: Stage-2 acceptance ratio as a function of the number of stage-2 accepted samples; right: number of stage-2 accepted samples as a function of number of stage-1 accepted samples. Blue: GPE-based on-the-fly learning; red: GPE-based active learning; green: Active-Subspace sampling; black: pure Monte-Carlo sampling. Solid lines: simplified testbed using GPE-based data; dashed lines; simplified testbed using Active-Subspace based data

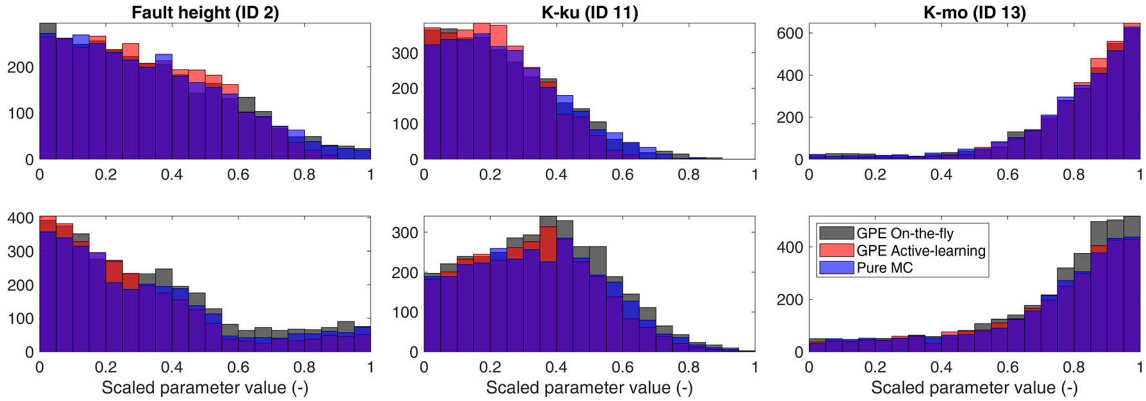

Fig. 6 Marginal distributions of the three example parameters with distinct distributions as shown in Fig. 3, here applied to the simplified testbeds. Upper row: simplified testbed with active-subspace based data; lower row: simplified testbed using GPE-derived data

simplified testbeds. We then ask the GPE surrogate models, either actively trained or trained on-the-fly, whether they predict these parameter-sets to be behavioral or not, resulting in true-positive or true-negative (correct), false-positive and false-negative predictions of the surrogate models. Table 3 shows the results. In all cases, 80% or more of the tested parameter-sets are correctly predicted as being behavioral or non-behavioral. It is also clear that barely any parameter-set is falsely predicted to be behavioral (false-positive in Table 3). The most common misclassification is false-negative, where the surrogate predicts a parameter-set to be non-behavioral while in fact, it is behavioral. Unfortunately, we consider this the least tolerable error, because we use the surrogate model only as a preselection tool. That is, a false-positive sample only implies that a full model run is performed on a parameter-set that needs to be discarded afterward, whereas a false-negative sample is not run at all, and we miss a part of the behavioral parameter space. With a ratio of false-negative predictions of about 10%, however, the error is not alarmingly large, especially considering that the tested datasets contained 50% truly behavioral samples, while a real sampling campaign would contain less than 1% behavioral sets. Because the rejected parameter-sets in such a setting are almost always truly non-behavioral, a realistic dataset would result in a better overall performance of the sampling schemes than in the test case. The dominance of the false-negative errors can be understood by the setup of

Table 3 Accuracy (%) of the GPE-based sampling schemes in predicting the behavioral status of 6000 parameter-sets in the simplified testbeds

	GPE-data		AS-data	
	OtF	AL	OtF	AL
Correct	95.2	91.6	93.0	87.2
False-negative	4.8	8.4	7.0	12.8
False-positive	0.02	0.02	0.03	0.02

GPE-data/AS-data, the true model is GPE-proxy model/Active subspace proxy model, OtF, on-the-fly; AL, active learning the sampling procedure. To obtain a false-negative, only one of the six targets need to generate a low probability. To obtain a false-positive, by contrast, all six targets must be falsely positive. Hence, the former is more likely to occur and therefore the false-negatives dominate over the false-positive errors. While there might be other approaches to combine the rejection criteria resulting in a balanced ratio between false-positive and false-negative errors, exploring such approaches would be outside of the scope of the current paper and is left for further research.

It may be worth noting that both Tables 3 and 2 indicate larger or more errors when using active learning rather than the on-the-fly sampling scheme. The larger differences to the reference distribution can come either from a suboptimal performance of the GPE-model resulting from the active learning, or, more likely, from the higher requirement during preselection use in this work for the active-learning scheme. This suggests that the improved sampling speed comes with a slight deterioration in the quality of the posterior distribution.

5 Discussion and conclusions

In this paper, we have used Gaussian Process Emulators as surrogate models for a surface-subsurface flow model, with the purpose of creating a large ensemble of behavioral parameter-sets to perform a global sensitivity analysis. We compared two different approaches: actively training the surrogate model prior to the sampling (active learning), or passively training the surrogate model as the sampling proceeds (on-the-fly). Both sampling schemes outperformed the pure Monte-Carlo sampling scheme (rejection sampling) by orders of magnitude, where a pure Monte-Carlo scheme only reached about 0.5% acceptance rate. The active learning sampling scheme showed the highest final acceptance rates with above 90%, while the on-the-fly sampling scheme improved with the increasing number of simulated samples and reached, in the best case, almost 90% acceptance rates. The on-the-fly scheme is most effective if the required ensemble size is smaller, in our case below 2000 samples for the full complex model. In a preceding study, we used polynomials of the first two active subspaces as surrogate models, achieving a sampling efficiency of just under 20%, so that the improvement of using the GPE is most notable. The GPE surrogate models not only predict the model outcome of the type as used in the training and its uncertainty, but also its gradient in parameter space, thus supporting the active-subspace approach as a global sensitivity analysis method.

To assess the performance of our sampling schemes, we estimate the time it would take to run the sampling without the GPE. The corresponding efficiency in the given application is about 0.5% (which results from the pure Monte-Carlo runs with the simplified testbeds), implying that already a minimum sample size of 300 behavioral flow models would require about 60,000 evaluations of the full flow model, at the computational cost of $\approx$ 240,000 CPU-hours. In contrast, any of our GPE-based sampling schemes would require less than 2000 CPU-hours to obtain the same number of behavioral samples, including the training time for the 6 GPE-surrogate models.

In the results section we could show that both GPE-based sampling schemes could reproduce the marginal distributions produced by a pure Monte-Carlo sampling scheme with high accuracy, even though this test could only be performed with two idealized testbeds. In this regard, the active-learning scheme sampled the target distributions slightly less accurate than the on-the-fly scheme.

The GPE-based sampling schemes require several tuning parameters. One of those that partly controls the resulting performance of the sampling schemes is the number of initial samples drawn before the first training of the surrogate model. In this work, we used an initial sample size of 50 Latin Hypercube sampled parameter-sets. If the requested size of the final ensemble is decently small, a large sample size before the first training of the GPE will cause many rejected full model runs (as, at least in our case, the pure Monte Carlo sampling has less than 1% acceptance ratio). However, a too-small number of initial runs will leave the surrogate model unable to make accurate predictions in the first iterations of the subsequent training. We have chosen 50 as this is roughly 1.5 times the number of parameters.

In designing a sampling scheme, two important and interlinked questions are always “what is the purpose of the sampling” and “how effective does it need to be”. In the present work, the aim is to explore the full posterior distribution of the behavioral parameter space. This means that we are interested in samples from the regions where we are very sure that they are accepted, as well as samples from the boundary between the behavioral and non-behavioral parts of the parameter space. The latter requirement is clearly more demanding, as a particularly effective sampling scheme (with very high acceptance rates) could potentially be so effective because it does not properly sample the uncertain boundary regions of the parameter space. Hence, it only samples the inner regions and thus causes a bias. In our work, we compared the two proposed GPE-based sampling schemes to pure Monte-Carlo sampling, showing that they both correctly sample the marginal parameter distributions. Contrasting to our current purpose, if we had been interested in just sampling a sufficiently large set of behavioral parameters, regardless of exploring the boundaries of the behavioral parameter space, the sampling would most likely be easier. Towards this end, we could raise the estimated required joint probability to accept a sample so that only samples are tested about which we are very sure that they are behavioral. In such a setting, the active-learning scheme would likely be the stronger candidate, as after the training we know rather certain where we do not want to be. In such a setting, an acceptance rate close to 100% would become achievable, while such a rate would be suspicious in our current setting where we want to explore the parameter space up to its behavioral boundaries.

A tempting thought to increase the acceptance rate, would be to use stage-2 accepted samples only to train the surrogate model. While this might indeed provide a higher acceptance rate, it might also limit the exploration of the parameter space to the inner part of the behavioral regions. As we here aim to explore the full behavioral parameter space, this approach is not feasible. For a further discussion on this topic, please see Erdal and Cirpka (2019).

When comparing the two GPE-based sampling schemes, the general conclusion of this study is to use the on-the-fly scheme, unless the required ensemble size is very large (notably larger than our current 3000 samples) or the user has reasons to believe that the active-learning part will be comparatively fast. For all other cases, the on-the-fly scheme is most likely the safer and better choice, as it produces usable simulation results already in the learning phase and seems to lead to more correct posterior parameter distributions.

We highly recommend using surrogate models to assist sampling the parameter space of complex models with unforeseeable boundaries of its behavioral regions. In this work, we have used the Gaussian Process Emulator as a surrogate model, primarily because of its proven good performance for subsurface flow (e.g., Cui et al. 2018b), and because it yields derivatives at minimal additional costs. Most likely, we could have used also other machine-learning tools with similarly good results (e.g., Yoon et al. 2011). In comparison to our previous study on the same model using a polynomial surrogate model of the first two active subspaces (Erdal and Cirpka 2019), by contrast, we see clear differences in both the performance and the ease of implementation. In the present study, we relied on a third-party code to perform all GPE-related tasks. This deems the methods a gray box: we know how it is supposed to work and what it is supposed to do, but the technical details are out of our control. The active-subspace based sampler, by contrast, is a complete white box, as the implementation is very simple and done in-house. If one just looks at the final performance metrics presented in studies like the current one, the choice for more complex models should be obvious. However, these figures are of course computed once the model framework is stable and operational, while the amount of time and number of model simulations spent getting there are commonly not discussed. In our case, the difference in complexity between the GPE and the active-subspace implementations resulted in a notable overhead for the former, including setup/learning time, unexplained errors, and crashes resulting from erroneous setups. Hence, one should consider the required ensemble size before choosing a method requiring a complicated implementation. Simple but less effective may be better for small ensembles, whereas complex and effective will be better when the required ensemble is larger.

Funding Open Access funding enabled and organized by Projekt DEAL.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

Aquanty Inc (2015) HydroGeoSphere user manual. Waterloo, ON

Asher MJ, Croke BF, Jakeman AJ, Peeters LJ (2015) A review of surrogate models and their application to groundwater modeling. Water Resour Res 51(8):5957–5973. https://doi.org/10.1002/2015WR016967

Bastos LS, O'Hagan A (2009) Diagnostics for Gaussian process emulators. Technometrics 51(4):425–438. https://doi.org/10.1198/TECH.2009.08019

Bect J, Vazquez E, et al (2017) STK: a small (matlab/octave) toolbox for kriging. Release 2.5. http://kriging.sourceforge.net

Busby D (2009) Hierarchical adaptive experimental design for Gaussian process emulators. Reliab Eng Syst Saf 94:1183–1193. https://doi.org/10.1016/j.ress.2008.07.007

Cadini F, Santos F, Zio E (2014) An improved adaptive Kriging-based importance technique for sampling multiple failure regions of low probability. Reliab Eng Syst Saf 131:109–117. https://doi.org/10.1016/j.ress.2014.06.023

Cohn DA, Ghahramani Z, Jordan MI (1996) Active learning with statistical models. Proc IEEE Int Symp Circuits Syst 3:129–145

Constantine PG, Diaz P (2017) Global sensitivity metrics from active subspaces. Reliab Eng Syst Saf 162(January):1–13. https://doi.org/10.1016/j.ress.2017.01.013

Constantine PG, Doostan A (2017) Time-dependent global sensitivity analysis with active subspaces for a lithium ion battery model. Stat Anal Data Min 10(5):243–262. https://doi.org/10.1002/sam.11347

Constantine PG, Dow E, Wang Q (2014) Active subspace methods in theory and practice: applications to kriging surfaces. SIAM J Sci Comput 36(4):A1500–A1524

Crevillén-García D (2018) Surrogate modelling for the prediction of spatial fields based on simultaneous dimensionality reduction of high-dimensional input/output spaces. R Soc Open Sci. https://doi.org/10.1098/rsos.171933

Cui T, Fox C, O'Sullivan MJ (2011) Bayesian calibration of a large-scale geothermal reservoir model by a new adaptive delayed acceptance Metropolis Hastings algorithm. Water Resour Res 47:W10521. https://doi.org/10.1029/2010WR010352

Cui T, Moore C, Raiber M (2018a) Probabilistic assessment of the impact of coal seam gas development on groundwater: Surat Basin, Australia. Hydrogeol J 26(7):2357–2377. https://doi.org/10.1007/s10040-018-1786-2

Cui T, Peeters L, Pagendam D, Pickett T, Jin H, Crosbie RS, Raiber M, Rassam DW, Gilfedder M (2018b) Emulator-enabled approximate Bayesian computation (ABC) and uncertainty analysis for computationally expensive groundwater models. J Hydrol 564(May):191–207. https://doi.org/10.1016/j.jhydrol.2018.07.005

D'Affonseca F, Finkel M, Cirpka OA (2020) Combining implicit geological modeling, field surveys, and hydrogeological modeling to describe groundwater flow in a karst aquifer. Hydrogeol J. https://doi.org/10.1007/s10040-020-02220-z

Echard B, Gayton N, Lemaire M (2011) AK-MCS: an active learning reliability method combining Kriging and Monte Carlo simulation. Struct Saf 33(2):145–154. https://doi.org/10.1016/j.strusafe.2011.01.002

Erdal D, Cirpka OA (2019) Global sensitivity analysis and adaptive stochastic sampling of a subsurface-flow model using active subspaces. Hydrol Earth Syst Sci 23(9):3787–3805. https://doi.org/10.5194/hess-23-3787-2019

Espinet AJ, Shoemaker CA (2013) Comparison of optimization algorithms for parameter estimation of multi-phase flow models with application to geological carbon sequestration. Adv Water Resour. https://doi.org/10.1016/j.advwatres.2013.01.003

Gadd C, Xing W, Nezhad MM, Shah AA (2019) A surrogate modelling approach based on nonlinear dimension reduction for uncertainty quantification in groundwater flow models. Transp Porous Media 126(1):39–77. https://doi.org/10.1007/s11242-018-1065-7

Gilbert JM, Jefferson JL, Constantine PG, Maxwell RM (2016) Global spatial sensitivity of runoff to subsurface permeability using the active subspace method. Adv Water Resour 92:30–42. https://doi.org/10.1016/j.advwatres.2016.03.020

Grey ZJ, Constantine PG (2018) Active subspaces of airfoil shape parameterizations. AIAA J 56(5):2003–2017. https://doi.org/10.2514/1.J056054

Jefferson JL, Gilbert JM, Constantine PG, Maxwell RM (2015) Active subspaces for sensitivity analysis and dimension reduction of an integrated hydrologic model. Comput Geosci 83:127–138. https://doi.org/10.1016/j.cageo.2015.07.001

Kitanidis PK (1997) The minimum structure solution to the inverse problem. Water Resour Res 33(10):2263–2272. https://doi.org/10.1029/97WR01619

Kollet S, Maxwell RM, Woodward CS, Smith S, Vanderborght J, Vereecken H, Simmer C (2010) Proof of concept of regional scale hydrologic simulations at hydrologic resolution utilizing massively parallel computer resources. Water Resour Res 46(4):W04201. https://doi.org/10.1029/2009WR008730

Kollet S, Sulis M, Maxwell RM, Paniconi C, Putti M, Bertoldi G, Coon ET, Cordano E, Endrizzi S, Kikinzon E, Mouche E, Mügler C, Park YJ, Refsgaard JC, Stisen S, Sudicky E (2017) The integrated hydrologic model intercomparison project, IH-MIP2: a second set of benchmark results to diagnose integrated hydrology and feedbacks. Water Resour Res 53(1):867–890. https://doi.org/10.1002/2016WR019191

Kopsiaftis G, Protopapadakis E, Voulodimos A, Doulamis N, Mantoglou A (2019) Gaussian process regression tuned by bayesian optimization for seawater intrusion prediction. Comput Intell Neurosci 2019:1–12. https://doi.org/10.1155/2019/2859429

Laloy E, Rogiers B, Vrugt JA, Mallants D, Jacques D (2013) Efficient posterior exploration of a high-dimensional groundwater model from two-stage Markov chain Monte Carlo simulation and polynomial chaos expansion. Water Resour Res 49(5):2664–2682. https://doi.org/10.1002/wrcr.20226

Loeppky JL, Sacks J, Welch WJ (2009) Choosing the sample size of a computer experiment: a practical guide. Technometrics 51(4):366–376. https://doi.org/10.1198/TECH.2009.08040

Maxwell RM, Putti M, Meyerhoff S, Delfs JO, Ferguson IM, Ivanov V, Kim J, Kolditz O, Kollet SJ, Kumar M, Lopez S, Niu J, Paniconi C, Park YJ, Phanikumar MS, Shen C, Sudicky EA, Sulis M (2015) Surface-subsurface model intercomparison: a first set of benchmark results to diagnose integrated hydrology and feedbacks. Water Resour Res 50:1531–1549. https://doi.org/10.1002/2013WR013725

Mishra S, Deeds N, Ruskauff G (2009) Global sensitivity analysis techniques for probabilistic ground water modeling. Ground Water 47(5):730–747. https://doi.org/10.1111/j.1745-6584.2009.00604.x

Mualem Y (1976) A new model for predicting the hydraulic conductivity of unsaturated porous media. Water Resour Res 12(3):513–522

Oladyshkin S, Nowak W (2012) Data-driven uncertainty quantification using the arbitrary polynomial chaos expansion. Reliab Eng Syst Saf 106:179–190. https://doi.org/10.1016/j.ress.2012.05.002

Oladyshkin S, de Barros FPJ, Nowak W (2012) Global sensitivity analysis: a flexible and efficient framework with an example from stochastic hydrogeology. Adv Water Resour 37:10–22. https://doi.org/10.1016/j.advwatres.2011.11.001

Ouyang Q, Lu W, Miao T, Deng W, Jiang C, Luo J (2017) Application of ensemble surrogates and adaptive sequential sampling to optimal groundwater remediation design at DNAPLs-contaminated sites. J Contam Hydrol 207(October):31–38. https://doi.org/10.1016/j.jconhyd.2017.10.007

Patterson HD, Thompson R (1971) Recovery of inter-block information when block sizes are unequal. Biometrika 58(3):545–554. https://doi.org/10.1093/biomet/58.3.545

Pianosi F, Beven K, Freer J, Hall JW, Rougier J, Stephenson DB, Wagener T (2016) Sensitivity analysis of environmental models: a systematic review with practical workflow. Environ Model Softw 79:214–232. https://doi.org/10.1016/j.envsoft.2016.02.008

Rajabi MM (2019) Review and comparison of two meta-model-based uncertainty propagation analysis methods in groundwater applications: polynomial chaos expansion and Gaussian process emulation. Stoch Environ Res Risk Assess 33(2):607–631. https://doi.org/10.1007/s00477-018-1637-7

Rajabi MM, Ketabchi H (2017) Uncertainty-based simulation-optimization using Gaussian process emulation: application to coastal groundwater management. J Hydrol 555:518–534. https://doi.org/10.1016/j.jhydrol.2017.10.041

Rasmussen CE, Williams CKI (2006) Gaussian processes for machine learning. The MIT Press, Cambridge

Ratto M, Castelletti A, Pagano A (2012) Emulation techniques for the reduction and sensitivity analysis of complex environmental models. Environ Model Softw 34:1–4. https://doi.org/10.1016/j.envsoft.2011.11.003

Razavi S, Tolson BA, Burn DH (2012a) Numerical assessment of metamodelling strategies in computationally intensive optimization. Environ Model Softw. https://doi.org/10.1016/j.envsoft.2011.09.010

Razavi S, Tolson BA, Burn DH (2012b) Review of surrogate modeling in water resources. Water Resour Res. https://doi.org/10.1029/2011WR011527

Richards LA (1931) Capillary conduction of liquids through porous mediums. Physics (College Park Md) 1(5):318–333. https://doi.org/10.1063/1.1745010

Saltelli A, Tarantola S, Campolongo F, Ratto M (2004) Sensitivity analysis in practice: a guide to assessing scientific models. Wiley, Chichester

Saltelli A, Ratto M, Andres T, Campolongo F, Cariboni J, Gatelli D, Saisana M, Tarantola S (2008) Global sensitivity analysis. Wiley, The Primer. https://doi.org/10.1002/9780470725184

Saltelli A, Annoni P, Azzini I, Campolongo F, Ratto M, Tarantola S (2010) Variance based sensitivity analysis of model output. Design and estimator for the total sensitivity index. Comput Phys Commun 181(2):259–270. https://doi.org/10.1016/j.cpc.2009.09.018

Shuttleworth WJ, Zeng X, Gupta HV, Rosolem R, de Gonçalves LGG (2012) Towards a comprehensive approach to parameter estimation in land surface parameterization schemes. Hydrol Process 27(14):2075–2097. https://doi.org/10.1002/hyp.9362

Sobol' IM, Kucherenko S (2009) Derivative based global sensitivity measures and their link with global sensitivity indices. Math Comput Simul 79(10):3009–3017. https://doi.org/10.1016/j.matcom.2009.01.023

Song X, Zhang J, Zhan C, Xuan Y, Ye M, Xu C (2015) Global sensitivity analysis in hydrological modeling: review of concepts, methods, theoretical framework, and applications. J Hydrol 523(225):739–757. https://doi.org/10.1016/j.jhydrol.2015.02.013

Spear R, Hornberger G (1980) Eutrophication in peel inlet-II. Identification of critical uncertainties via generalized sensitivity analysis. Water Res 14:43–49

Stein ML (1999) Interpolation of spatial data: some theory for kriging. Springer, Berlin

Tian L, Wilkinson R, Yang Z, Power H, Fagerlund F, Niemi A (2017) Gaussian process emulators for quantifying uncertainty in CO2 spreading predictions in heterogeneous media. Comput Geosci 105:113–119. https://doi.org/10.1016/j.cageo.2017.04.006

Van Genuchten M (1980) A closed-form equation for predicting the hydraulic conductivity of unsaturated soils. Soil Sci Soc Am J 8:892–898

Vrugt JA, Stauffer PH, Wöhling T, Robinson BA, Vesselinov VV (2008) Inverse modeling of subsurface flow and transport properties: a review with new developments. Vadose Zo J 7(2):843–864. https://doi.org/10.2136/vzj2007.0078

von Gunten D, Wöhling T, Haslauer C, Merchán D, Causapé J, Cirpka OA (2014) Efficient calibration of a distributed pde-based hydrological model using grid coarsening. J Hydrol 519:3290–3304. https://doi.org/10.1016/j.jhydrol.2014.10.025

Wagener T, Pianosi F (2019) What has global sensitivity analysis ever done for us? a systematic review to support scientific advancement and to inform policy-making in earth system modelling. Earth Sci Rev 194:1–18. https://doi.org/10.1016/j.earscirev.2019.04.006

Wu B, Zheng Y, Tian Y, Wu X, Yao Y, Han F, Liu J, Zheng C (2014) Systematic assessment of the uncertainty in integrated surface water-groundwater modeling based on the probabilistic collocation method. Water Resour Res 50(7):5848–5865. https://doi.org/10.1002/2014WR015366

Wu B, Zheng Y, Wu X, Tian Y, Han F, Liu J, Zheng C (2015) Optimizing water resources management in large river basins with integrated surface water-groundwater modeling: A surrogate-based approach. Water Resour Res 51(4):2153–2173. https://doi.org/10.1002/2014WR016653

Xiao S, Lu Z (2017) Structural reliability sensitivity analysis based on classification of model output. Aerosp Sci Technol 71:52–61. https://doi.org/10.1016/j.ast.2017.09.009

Xiao S, Lu Z, Wang P (2018) Multivariate global sensitivity analysis based on distance components decomposition. Risk Anal 38(12):2703–2721. https://doi.org/10.1111/risa.13133

Xiao S, Oladyshkin S, Nowak W (2020) Reliability analysis with stratified importance sampling based on adaptive Kriging. Reliab Eng Syst Saf. https://doi.org/10.1016/j.ress.2020.106852

Xu T, Valocchi AJ, Ye M, Liang F (2017) Quantifying model structural error: efficient Bayesian calibration of a regional groundwater flow model using surrogates and a data-driven error model. Water Resour Res 53(5):4084–4105. https://doi.org/10.1002/2016WR019831

Yeh WWG (2015) Review: optimization methods for groundwater modeling and management. Hydrogeol J 23(6):1051–1065. https://doi.org/10.1007/s10040-015-1260-3

Yoon H, Sc Jun, Hyun Y, Go Bae, Kk Lee (2011) A comparative study of artificial neural networks and support vector machines for predicting groundwater levels in a coastal aquifer. J Hydrol 396(1–2):128–138. https://doi.org/10.1016/j.jhydrol.2010.11.002

Zhang J, Li W, Lin G, Zeng L, Wu L (2017) Efficient evaluation of small failure probability in high-dimensional groundwater contaminant transport modeling via a two-stage Monte Carlo method. Water Resour Res 53:1948–1962. https://doi.org/10.1002/2016WR019518

Zhang J, Man J, Lin G, Wu L, Zeng L (2018) Inverse modeling of hydrologic systems with adaptive multifidelity Markov chain Monte Carlo simulations. Water Resour Res 54(7):4867–4886. https://doi.org/10.1029/2018WR022658

Zheng Q, Zhang J, Xu W, Wu L, Zeng L (2019) Adaptive multifidelity data assimilation for nonlinear subsurface flow problems. Water Resour Res 55(1):203–217. https://doi.org/10.1029/2018WR023615

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.