Bayesian inference of a lake water quality model by emulating its posterior density

A. Dietzel and P. Reichert

1Eawag, Swiss Federal Institute of Aquatic Science and Technology, Dübendorf, Switzerland

Abstract We use a Gaussian stochastic process emulator to interpolate the posterior probability density of a computationally demanding application of the biogeochemical-ecological lake model BELAMO to accelerate statistical inference of deterministic model and error model parameters. The deterministic model consists of a mechanistic description of key processes influencing the mass balance of nutrients, dissolved oxygen, organic particles, and phytoplankton and zooplankton in the lake. This model is complemented by a Gaussian stochastic process to describe the remaining model bias and by Normal, independent observation errors. A small subsample of the Markov chain representing the posterior of the model parameters is propagated through the full model to get model predictions and uncertainty estimates. We expect this approximation to be more accurate at only slightly higher computational costs compared to using a Normal approximation to the posterior probability density and linear error propagation to the results as we did in an earlier paper. The performance of the two techniques is compared for a didactical example as well as for the lake model. As expected, for the didactical example, the use of the emulator led to posterior marginals of the model parameters that are closer to those calculated by Markov chain simulation using the full model than those based on the Normal approximation. For the lake model, the new technique proved applicable without an excessive increase in computational requirements, but we faced challenges in the choice of the design data set for emulator calibration. As the posterior is a scalar function of the parameters, the suggested technique is an alternative to the emulation of a potentially more complex, structured output of the simulation model that allows for the use of a less case-specific emulator. This is at the cost that still the full model has to be used for prediction (which can be done with a smaller, approximately independent subsample of the Markov chain).

1. Introduction

Environmental models are often complex, computationally demanding, and affected by bias. A general methodology has been developed to deal with these difficulties [Sacks et al., 1989; Currin et al., 1991; Kennedy and O’Hagan, 2001; Santner et al., 2003; Higdon et al., 2004; O’Hagan, 2006; Bayarri et al., 2007a, 2007b; Zhou et al., 2011]. In this methodology, bias is considered by describing it by a Gaussian stochastic process. Computational limitations are addressed by replacing the demanding computer code by a statistical model, in this case another Gaussian stochastic process, conditioned to design runs, to get a fast emulator approximating the simulation results. Such a replacement of a dynamic simulation model by a faster response surface approximation is also known as surrogate, meta, or proxy modeling and many different techniques have been suggested to get such approximations (see Razavi et al. [2012] for a recent review focusing on water resources modeling).

However, many environmental simulation models are dynamic, which makes these standard emulation techniques difficult to apply. There are (at least) the following three options to deal with this problem:

1. Extend emulators to cope explicitly with dynamic models. Several methodologies have been suggested to do so that are based on different conceptual approaches to emulating dynamic models [Castelletti et al., 2012]. One of these approaches are model reduction techniques that were originally developed for linear models [see Antoulas et al., 2001, for a review] and later extended to nonlinear models [Hinze and Volkwein, 2005]. Also Gaussian stochastic process-based emulators mentioned above were adapted to cope with dynamic models. A first approach to do this relies on statistical, linear dynamic models [Liu and West, 2009; Young and Ratto, 2011] as a basis for the emulator or on dynamic models that approximate...
the mechanisms of the full model [Reichert et al., 2011; Albert, 2012]. A second approach is based on the emulation of the coefficients of functional series to approximate model output [Bayarri et al., 2007a, 2007b; Higdon et al., 2008; Morris, 2012]. A third approach uses multivariate emulators to describe the time series [Rougier, 2008; Conti and O'Hagan, 2010; Paulo et al., 2012]. And, finally, a fourth approach emulates the time-stepping mechanism of the full model to get an emulator of the time series [Bhattacharya, 2007; Conti et al., 2009].

2. Use simulations of the full model to derive an analytical approximation to the posterior probability density, often a multivariate Normal distribution or a linear combination of multiple Normal distributions that is then used for statistical inference and prediction [Gelman et al., 2004; Bornkamp, 2011; Dietzel and Reichert, 2012].

3. Do not emulate the full model, but emulate the log of the unnormalized posterior density of the model. As this is a scalar function of the model parameters, it needs a less complex structure of the emulator compared to emulating potentially multivariate time series. This emulator can then be used to calculate Markov chains of the posterior very quickly. However, in contrast to the case when emulating the full model, this emulator cannot be used for prediction. Prediction is then done by propagating a nearly independent subsample of the posterior Markov chain through the full model. This is the approach suggested in this paper to complement the other approaches.

Despite the number and diversity of approaches that have been suggested for the emulation of dynamic models (listed under item 1 above), more research is needed to guide the selection of a methodology for a specific problem and to make these approaches easier to apply. On the other hand, the much simpler approach of approximating the posterior density by a multivariate Normal distribution (listed under item 2 above) is not satisfying as such a simple parametric shape can hardly be expected to accurately approximate the posterior. For this reason, in parallel to further developing dynamic emulators, it is of interest to expand the toolbox of modelers with a technique that is potentially easier to transfer to other applications because it does not depend on the output structure of the model, even if it may be slightly less efficient for prediction. Obviously, the shape of the posterior to be emulated depends on the model and the data and can be very complicated. It is the goal of this paper to explore the potential of emulating this shape of the posterior instead of the model output. This is done for a didactical example as well as a real-world case study. The didactical example is a simple microbial growth model that was already used by Reichert and Schuwirth [2012] and Dietzel and Reichert [2012]. This makes a direct comparison possible between the full Bayesian approach, the use of the emulator of the posterior, and the use of a Normal approximation to the posterior combined with linearized error propagation to the model results. For the real-world case study, the same lake model was used as in Dietzel and Reichert [2012], where a Normal approximation to the posterior was used. The lake model BELAMO is a specifically interesting application case for the proposed emulation method. Due to its relatively long computation time of 1 min for the simulation of yearly nutrient and plankton dynamics in four lake compartments, standard parameter calibration techniques are difficult to apply for long-term simulations. As this model is intended to summarize the knowledge about the relation between lake nutrient inputs and planktic growth and to potentially support management decisions, a thorough parameter calibration and uncertainty analysis is needed.

This paper is structured as follows. The methodology is described in section 2. In the following section 3 it is applied to the didactical example. To demonstrate its feasibility for a realistic model application, section 4 describes the use of the methodology to the lake water quality model. Finally, we draw our conclusions in section 5.

2. Methods

The methodology applied in this paper consists of the following steps:

1. Formulating a (unnormalized) posterior based on a likelihood function that considers bias in the form of a Gaussian stochastic process and on a prior of the parameters. Evaluating this posterior for a design, parameter set spread over a reasonable domain of parameter values. This domain can be chosen as a neighborhood of the numerical approximation to the parameter set that leads to the maximum posterior value. This step is slow, as simulations of the deterministic model, for which the posterior is calculated, are required.
2. Conditioning a Gaussian Stochastic Process (GASP) emulator to interpolate the log of the unnormalized posterior as a function of the parameters between the points of the design parameter set and estimate interpolation accuracy. This step is fast, as there is no need for further evaluations of the deterministic model. It requires a simple, scalar emulator that is independent of the output structure of the deterministic model.

3. Applying a Metropolis Markov chain Monte Carlo technique to derive a Markov chain sample of the emulated posterior. This step is again fast, as only the emulator, not the deterministic model has to be evaluated. This computational efficiency makes it possible to generate long Markov chains that sample the approximate, emulated posterior quite accurately.

4. Propagating a subsample of the Markov chain through the deterministic and error model to estimate prediction uncertainty. This step is slow, as it requires the full model, including the potentially slow deterministic model, to be evaluated. To improve computational efficiency, it may be useful to only propagate a relatively small, approximately independent, subsample of the Markov chain.

These four steps are discussed in more detail in the following four subsections.

2.1. Derivation of Posterior

As suggested by Craig et al. [1996, 2001], Kennedy and O’Hagan [2001], Higdon et al. [2004], and Bayarri et al. [2007a, 2007b], we describe the observations as follows:

\[ \mathbf{Y}_M^{\ell}(x, \theta, \psi, \xi) = \mathbf{y}_M^{\ell}(x, \theta) + \mathbf{B}_M^{\ell}(x, \xi) + \mathbf{E}_M^{\ell}(\psi). \]  (1)

The vector \( \mathbf{Y}_M^{\ell} \) of \( n_l \) random variables represents the observations as described by the full model \( M \) consisting of the deterministic model, \( \mathbf{y}_M^{\ell}(x, \theta) \), the bias correction, \( \mathbf{B}_M^{\ell}(x, \xi) \), and observation error, \( \mathbf{E}_M^{\ell}(\psi) \), at the observation layout \( L \). The observation layout defines the output variables and the time points and locations at which they are observed or evaluated. Its length \( n_l \) defines the number of combinations of different output variables, time points, and locations of interest. \( \mathbf{Y}_M^{\ell} \) depends on the external influence factors, \( x \), unknown parameters of the deterministic model, \( \theta \), and additional parameters, \( \psi \) and \( \xi \), of the error terms. Equation (1) leads to a hierarchical model with the bias, \( \mathbf{B}_M^{\ell}(x, \xi) \), as an intermediate variable.

We assume Normally distributed observation errors with mean zero and covariance matrix \( \Sigma_{\mathbf{E}_M}(\psi) \)

\[ f_{\mathbf{E}_M|\psi}(x|\psi) = \frac{1}{\sqrt{2\pi \sigma_{\mathbf{E}_M}}} \exp \left( -\frac{1}{2} \left( x - \mu_{\mathbf{E}_M}(\psi) \right)^T \Sigma_{\mathbf{E}_M}^{-1} \left( x - \mu_{\mathbf{E}_M}(\psi) \right) \right). \]  (2)

and a zero-mean Gaussian stochastic process for the bias with covariance matrix \( \Sigma_{\mathbf{B}_M}(\xi, x) \)

\[ f_{\mathbf{B}_M|\xi}(b|\xi) = \frac{1}{\sqrt{2\pi \sigma_{\mathbf{B}_M}}} \exp \left( -\frac{1}{2} \left( b - \mu_{\mathbf{B}_M}(\xi, x) \right)^T \Sigma_{\mathbf{B}_M}^{-1} \left( b - \mu_{\mathbf{B}_M}(\xi, x) \right) \right). \]  (3)

The diagonal of the covariance matrix of the observation error is assumed to consist of the elements

\[ \Sigma_{\mathbf{E}_M,\ell}(\psi) = \sigma_{\mathbf{E}_M,\ell}(\psi)^2, \]  (4)

all other matrix elements are assumed to be zero. The covariance matrix of the bias is assumed to consist of the matrix elements

\[ \Sigma_{\mathbf{B}_M,\ell}(\xi, x) = \sigma_{\mathbf{B}_M,\ell}(\xi, x)^2 \exp \left( -\frac{(t - t_j) \tau}{\tau} \right) \]  (5)

depending on points in time \( t \) and the correlation time \( \tau \) (note that this choice of the covariance matrix does not depend on the input \( x \); however, such a dependence may be important for systems the dynamics of which is more directly forced by the input, see, e.g., Honti et al. [2013] or Del Giudice et al. [2013]). For output variables \( y_j^\ell \) and \( y_j^l \) that are elements of different time series, the (off diagonal) matrix elements are assumed to be zero. In our didactical example and our real model application, these different time series represent concentrations of microbes and substrate in a mixed reactor, and phytoplankton, zooplankton, nutrients, and dissolved oxygen in the epilimnion or the hypolimnion of the lake, respectively. The equations (2) and (3) make it possible to separate model bias and observation error through different
assumptions about the correlations. The observation errors are assumed to be independent, whereas the bias is characterized by considerable autocorrelation due to systematic model errors and propagation of errors through a deterministic model that usually leads to memory effects.

Note that the model formulation given by equation (1) leads to identifiability problems at two different levels. First, the description of output pattern by the sum \( y_M(x, \theta) + B_i(x, \xi) \) makes it possible to describe these predominantly by one or the other of these terms. The objective is to describe as much of this pattern as possible by the deterministic model, \( y_M(x, \theta) \), and use the bias correction, \( B_i(x, \xi) \), only to the degree needed. Second, the correlation structure parameters of the Gaussian stochastic process have a poor identifiability, as has been shown in the context of emulators [Li and Sudjianto, 2005]. Both of these problems have to be addressed by the choice of the prior distribution of the parameters. To keep the bias as small as possible, the prior density of its standard deviation should have its maximum at zero and decrease toward larger values. The characteristic width of this decrease has to be chosen accordingly to past experiences with the typical size of model bias. The poor identifiability of the Gaussian stochastic process parameters is mainly caused by a lack of sensitivity if the correlation length (in our case time) decreases below the observation interval [Li and Sudjianto, 2005]. This case, which makes the bias at subsequent time points approximately independent, leads to additional identifiability problems with the observation error, \( E_1(\psi) \). Such short correlation lengths are obviously not adequate for the description of systematic model deviations that should be described by the bias term. This has also to be addressed by an adequate choice of the prior of this correlation length. In practice, this is often not that difficult as it has to relate to characteristic times characterizing the memory effects in the model that are usually known. With such priors, the inference process will be able to address independent errors by \( E_1(\psi) \) and correlated errors by \( B_i(x, \xi) \) and it will keep the bias as small as possible.

Under the assumptions specified above, using observations, \( y^i_L \), for the observation layout \( L \), and using the joint prior, \( f_\Theta \xi \psi T \{ \theta, \psi, \xi | y^i_L, x \} \), of the parameters of the deterministic model, \( \Theta \), of the observation error parameters, \( \Psi \), and of the bias parameters, \( \Xi \), the posterior of the parameters is given by a multivariate Normal distribution as follows [see Reichert and Schuwirth, 2012; Dietzel and Reichert, 2012, for more details]:

\[
\begin{align*}
 f_{\Theta \xi \psi T \{ \theta, \psi, \xi | y^i_L, x \}} & = \frac{f_{\Theta \xi \psi T \{ \theta, \psi, \xi \} y^i_L, x}}{\sqrt{\det(\Sigma_{\Theta} + \Sigma_{\Xi})}} \\
 & \cdot \exp \left( -\frac{1}{2} \| y^i_L - y_M(x, \theta) \|^2 (\Sigma_{\Theta} + \Sigma_{\Xi})^{-1} \| y^i_L - y_M(x, \theta) \| \right).
\end{align*}
\]

The joint prior distribution of all parameters was assumed to be the independent combination of all required marginals.

To account for heteroscedasticity in deterministic model output of individual time series, as in Dietzel and Reichert [2012], we applied the technique to Box-Cox transformed data and model results. To account for heteroscedasticity between time series of different output variables, we used different variances for different output variables.

### 2.2. Gaussian Stochastic Process Emulator

In a previous study [Dietzel and Reichert, 2012], a Normal distribution was used to approximate the posterior (6) at its maximum to reduce the computational demand resulting from the evaluation of the deterministic model, \( y_M(x, \theta) \), during the inference process. This approximating Normal distribution was constructed based on evaluations of the posterior for a parameter sample from a Normal or Uniform distribution in a neighborhood of the maximum of the posterior. The maximum had been estimated before using a global optimization algorithm [Dietzel and Reichert, 2012].

In this paper, we try to improve this approximation by emulating the log of the posterior instead of approximating it by a parameterized (in our case Normal) distribution. This emulator approximates the dependence of the posterior, \( f_\Theta \xi \psi T \{ \theta, \psi, \xi \} \), given by equation (6), as a function of the parameters of the deterministic model, \( \Theta \), those of the observation model, \( \Psi \), and those of the bias correction term, \( \Xi \). It emulates this scalar function directly, bypassing the intermediate dependence of the posterior on the output of the deterministic model, \( y_M(x, \theta) \), that has a much more complicated structure (multivariate time series). As the emulator interpolates between the evaluations of the log posterior for a given design data set, we can expect a better representation of its shape than can be achieved by the Normal distribution, if the size of the design data
set is sufficiently large. As a basis for conditioning the emulator, we draw a random sample, \( \gamma_D = (\gamma_D^1, \ldots, \gamma_D^{n_d}) \), of size \( n_D \) from a multivariate Normal distribution around the maximum of the posterior as a design parameter set (this may require a suitable transformation of the parameter range to an unlimited range or a truncation of the Normal distribution). Here \( \gamma = (\Theta^T, \Psi^T, \Xi^T)^T \) is used as an abbreviation for all parameters. Using the same notation for the corresponding random vectors, \( \Gamma = (\Theta^T, \Psi^T, \Xi^T)^T \), this design data set can be written in the form

\[
D = (\gamma_D, f_D) = \left( \gamma_D^0, f_{\Gamma | \gamma_D^0}(\gamma_D) \right),
\]

with the posterior densities \( f_D = f_{\Gamma | \gamma_D^0}(\gamma_D) \) calculated according to equation (6).

For the special case of a univariate emulator, the expected value \( \log f \) of the multivariate Normal distribution describing the GASP emulator conditioned on the log of the unnormalized posterior density values at the design data points is given by [see O’Hagan, 1992; Reichert et al., 2011, equation (17)]:

\[
\log f(\gamma, D) = \log(f_{\gamma | \gamma_D}(\gamma) + k^{(\alpha)}(\gamma, \gamma_D)) - \left( \begin{array}{c} \log(f_D^0) - \log(f_{\gamma | \gamma_D}(\gamma_D^0)) \\ \vdots \\ \log(f_D^{n_D}) - \log(f_{\gamma | \gamma_D}(\gamma_D^{n_D})) \end{array} \right) \left( \begin{array}{c} k^{(\alpha)}(\gamma_D^0, \gamma_D^1) \\ \vdots \\ k^{(\alpha)}(\gamma_D^{n_D}, \gamma_D^{n_D}) \end{array} \right)^{-1}.
\]

In this equation, \( \log(f_{\gamma | \gamma_D}(\gamma_D)) \) represents the prior mean of the emulator interpolating the log posterior, while \( f_D \) represents the exact posterior values. For the application to a didactical example, we use the log values of the Normal density approximating the posterior distribution resulting from the importance sampling technique described in Dietzel and Reichert [2012] as the deterministic prior mean of the emulator (shifted by a constant to account for the lack of normalization of the posterior density). For the lake water quality model application, we used a constant mean (set to the mean of the log posterior density values of the design data set), but allowed for a correction by a similar shift parameter. The choice of a constant mean as a prior for the emulator has the disadvantage that it does not lead to an integrable function. The emulator will return to this constant value when being evaluated far away from the design data points. However, as we only need a good representation of the posterior within the coverage area of the Markov chain, this property may be an advantage regarding the diagnosis of problems of coverage of the posterior by design data points. If the Markov chain moves far away from the design points, the posterior will be evaluated to this mean and the Markov chain will diverge. If this divergence can be resolved by a better coverage with design points, it may be an indication of such a numerical problem rather than a true inference result. With this choice of a prior mean, nondiverging Markov chains are thus an indication of a sufficient coverage of the high-probability density area of the posterior. Obviously, this still does not guarantee a sufficiently dense coverage to get an accurate emulation. The use of the prior density of the parameters could be a valuable alternative to our choices for the prior of the mean of the emulator. However, we would loose the test mentioned above because a close representation of the prior by the emulated posterior is more difficult to be identified as being a true result of the inference process or as being a numerical artifact resulting from an insufficient coverage of the posterior by the design points of the emulator.

The matrix \( K^{(\alpha)} \) in equation (8) defines the covariance structure of the unconditioned Gaussian process for an arbitrary set of \( n \) parameter vectors:

\[
K^{(\alpha)}(\gamma^1, \ldots, \gamma^n) = \begin{pmatrix} K(\gamma^1, \gamma^1) & \cdots & K(\gamma^1, \gamma^n) \\ \vdots & \ddots & \vdots \\ K(\gamma^n, \gamma^1) & \cdots & K(\gamma^n, \gamma^n) \end{pmatrix},
\]

with

\[
K(\gamma', \gamma') = \sigma^2 \cdot \exp \left( -\sum_{k=1}^{n} \left( \frac{\gamma'_k - \gamma_k'}{\lambda_k} \right)^2 \right).
\]

Equation (10) defines the correlation structure of the covariance matrix and defines the standard deviation \( \sigma \) of the prior Gaussian process. For the applications in this paper, we consider only the mean of the
conditioned Gaussian process to interpolate the posterior probability density. The value of $\sigma$ does not influence these results (it would influence, however, the variance of the predicted log posterior density). The parameters $k_k$ are the correlation lengths of all model parameters $c_k$; $n_c$ is the number of model parameters, in our case parameters of the deterministic model and of the error model. The larger these correlation lengths, the smoother the interpolation by the emulator. The minimum size of these correlation lengths is bounded by a typical distance between design points in the corresponding dimension. If the correlation lengths are chosen to be too small, the emulator parameters become poorly identifiable and the emulation result will approach the prior mean when being far from design points even within the coverage area of the posterior by design points (see Li and Sudjianto [2005] and also discussion below, in particular the top row of plots in Figure 1 below). Additionally, the correlation matrix depends on the roughness parameter $\alpha$. The mean of the emulator is (infinitely often) differentiable at the design data points only for the choice $\alpha = 2$. This choice is susceptible to oscillations of the mean of the emulator, if large correlation lengths are chosen (see Figure 1 below). Smaller values of $\alpha$ resolve this problem, but, due to the nondifferentiability at the design points, the emulation response surface can become more rugged. In equation (8), the matrix $K^{(n_c)}$.
only depends on the design parameter set and hence can be calculated according to equation (9) and inverted already during the set up of the emulator. The same is true for the last factor in the same equation. The value of \( f_M(\gamma) \) is given by the result of the prior emulator mean for new parameter values \( \gamma \). The vector \( k \) is the only other component of equation (8) which depends on the new parameter values. It is given by:

\[
k^{(n)}(\gamma, \gamma^1, \ldots, \gamma^n) = \left( \begin{array}{c} K(\gamma, \gamma^1) \\ \vdots \\ K(\gamma, \gamma^n) \end{array} \right),
\]

with components as given by equation (10).

The emulation method described above can be considered as kriging in parameter space [Krige, 1951; Cressie, 1993; Kleijnen, 2009]. Figure 1 illustrates the behavior of the interpolator for different values of \( x \) and \( \lambda \).

The figure shows the emulation of the model

\[
y = \max \left( 80 - 10(\theta - 2.5)^2, 10 \right),
\]

with the only parameter \( \theta \) and the GASPE emulator based on a linear regression as prior emulator mean. The design parameter set consists of all integer numbers between 0 and 10. The emulation was conducted for \( x \) equals to 2, 1.9, and 1 and \( \lambda \) equals to 0.08, 0.8, and 8. It becomes obvious that for a small correlation length, the emulator is unable to provide a good interpolation as the “influence” of the design results does not extend “far enough” and the emulator switches to its prior mean. For larger correlation lengths, the emulation is smoother and less dependent on its prior mean, but for \( x = 2 \) and a large correlation length, this can lead to oscillations between the design data points. The results shown in Figure 1 indicate that the choice of the correlation length depends more on the coverage of the parameter space by design data points than on the behavior of the model to be emulated as long as a random sample is used for the design data. It may, however, be useful to account for regions of stronger variability by increasing the density of the design data points. They also show that an emulator can approximate a function much better than a Normal distribution, as also stated by O’Hagan [1992].

### 2.3. Markov Chain Monte Carlo Sampling Based on Emulator

The emulator conditioned to the design data of the unnormalized posterior as described in section 2.2 was used to calculate a Markov chain sample of the (approximate) posterior calculated by \( \log f(\gamma, D) \) according to equation (8) using a Metropolis algorithm [Gelman et al., 2004; Gamerman, 1997]. The proposal distribution was optimized by four iterations with shorter Markov chains before the final Markov chain was calculated resulting in a sample of the emulated posterior. For visualization of the results, a density estimator was applied to this sample to provide estimates of posterior marginal distributions of the parameters \( \gamma \).

### 2.4. Model Predictions

Model predictions for new observations (sum of the deterministic model output, bias and observation error) not used for calibration defined by layout \( L_2 \) are distributed according to [Reichert and Schwirch, 2012]

\[
f_{E, \gamma}^{(l)}(y^i_j | y^i_j, x) = \frac{1}{\sqrt{2\pi c^2}} \exp \left( -\frac{1}{2} \left( y^i_j - E[Y_M^{(l)} | Y_M^{(l)}, \Theta, \Psi, \Xi] \right)^T \cdot \text{Var}[Y_M^{(l)} | Y_M^{(l)}, \Theta, \Psi, \Xi]^{-1} \cdot \left( y^i_j - E[Y_M^{(l)} | Y_M^{(l)}, \Theta, \Psi, \Xi] \right) \right) 
\]

with

\[
E[Y_M^{(l)} | Y_M^{(l)}, \Theta, \Psi, \Xi] = Y_M^{(l)}(x, \theta) + \left( \Sigma_{e^{(l)}} + \Sigma_{b^{(l)}} \right)_{L_2, L_1} (\Sigma_{e^{(l)}} + \Sigma_{b^{(l)}})^{-1} (y^i_j - y_M^{(l)}(x, \theta)),
\]

and

\[
\text{Var}[Y_M^{(l)} | Y_M^{(l)}, \Theta, \Psi, \Xi] = \Sigma_{e^{(l)}} + \Sigma_{b^{(l)}} - \left( \Sigma_{e^{(l)}} + \Sigma_{b^{(l)}} \right)_{L_2, L_1} (\Sigma_{e^{(l)}} + \Sigma_{b^{(l)}})^{-1} (\Sigma_{e^{(l)}} + \Sigma_{b^{(l)}})^T_{L_2, L_1}.
\]
In these equations, $L_1$ refers to the observation layout (state variables and points in time and space) used for calibration, $L_2$ to the layout for which results are to be predicted [see Reichert and Schuwirth, 2012, for more details]. In practical applications, the observation layout will typically correspond to the past and the prediction layout to the future, but this is not formally required. From these results, approximate uncertainty bounds can be derived for the results of the deterministic model considering the contributions of uncertainty due to model bias and observation error. Omitting the terms $\Sigma_{E_2}$ and $(\Sigma_{E_1})_{L_2,L_2}$ in equations (14) and (15) results in a representation of our knowledge of the "true" system output before observation (results of the deterministic model plus bias) in equation (13).

Numerically, a sample from this distribution is drawn by first choosing a subsample (for increasing the numerical efficiency) of the Markov chain derived as described in section 2.3. Then, for each parameter vector, $(\theta, \psi, \xi)$, of this subsample, a sample is drawn from the Normal distribution defined by the equations (14) and (15) while either using or omitting the terms $\Sigma_{E_2}$ and $(\Sigma_{E_1})_{L_2,L_2}$. This results in sampling either from $y_M^l | Y_M^1$ or from $y_M^l + B_M^l | Y_M^1$, respectively.

The emulation technique was implemented with the statistics and graphics software R [R Core Team, 2013].

3. Didactical Example

We use the same microbial growth model as used in Reichert and Schuwirth [2012] and Dietzel and Reichert [2012], as a simple example for the application of the suggested emulation approach and for an informative comparison to the calibration and uncertainty analysis approaches described in the other two papers.

3.1. Model Description, Observation Data, and Prior Parameter Distributions

This deterministic model describes the degradation of a substrate of concentration $C_S$ by microbes with concentration $C_M$ in a batch reactor (without inflow or outflow). This system can be described by the following mass balance equations:

$$\frac{dC_M}{dt} = \mu \frac{C_S}{K + C_S} C_M \exp \left( c(T - T_0) \right) - B C_M - q C_M,$$

$$\frac{dC_S}{dt} = -\frac{\mu}{K + C_S} C_M \exp \left( c(T - T_0) \right) + q (C_{S,in} - C_S),$$

with the initial conditions

$$C_M(0) = C_{M,in},$$

$$C_S(0) = C_{S,in},$$

and the temperature variations

$$T = T_0 + A \sin \left( 2\pi (t - 0.25d) \right).$$

In these equations, $C_M$ is the concentration of microbes in the reactor, $C_S$ is the concentration of substrate in the reactor, $t$ is time, $\mu$ is the maximum specific growth rate of the microbes, $K$ is the concentration of substrate at which the microbes grow with half of their maximum specific growth rate, $B$ is the specific death rate of the microbes, $q$ is the volumetric flow rate through the reactor per unit of reactor volume, $Y$ is the yield (produced amount of biomass per consumed amount of substrate), $C_{S,in}$ is the concentration of substrate in the inflow (the concentration of microbes in the inflow is assumed to be zero), $T$ refers to temperature, $T_0$ is a reference temperature assumed to be 20°C, $A$ is the amplitude of temperature variations, $c$ is a temperature dependence coefficient assumed to be 0.046 per °C, $C_{M,in}$ is the initial concentration of microbes in the reactor, and $C_{S,in}$ is the initial concentration of substrate in the reactor. The initial concentrations of microbes and substrate were chosen in a way that allows for a longer dynamic phase before the substrate is taken up.

With this deterministic model, we could produce synthetic observations for a simulation period of 4 days (10 observations of both, $C_M$ and $C_S$, per day), an amplitude of temperature variation of 4°C and the following parameter values: $\mu = 4$, $K = 10$, $b = 1$, $q = 1$, $Y = 0.6$, $C_{S,in} = 100$, $C_{M,in} = 10$, $C_{S,in} = 40$, and $c_{E_2} = 0.5$ and $c_{E_1} = 0.5$ (in model output units) as standard deviations of Normally distributed observation errors in both.
model output variables (concentration of microbes and of substrate). As the generated data does not contain heteroscedasticity, no Box-Cox transformation was applied.

The first 2 days out of these data were used to estimate the parameters of a deterministic model that describes the same system but contains structural error due to neglecting the temperature dependence of microbial growth (and associated substrate consumption) as described by the equations (16). The vector \( \theta = (\mu, K, b, q_e, Y, C_{sm}, C_{ini}, C_{ini})^T \) gives the parameter vector of the deterministic model. The parameters \( \phi = (\sigma_{E_{eq}}, \sigma_{E_{ini}})^T \) consist of the standard deviations \( \sigma_{E_{eq}} \) and \( \sigma_{E_{ini}} \) of independent, zero-mean Normal distributions characterizing the observation error of \( C_M \) and \( C_S \) at all points in time. Finally, the vector \( \xi = (\sigma_{B_{eq}}, \sigma_{B_{ini}} \cdot \tau)^T \) combines the standard deviations \( \sigma_{B_{eq}} \) and \( \sigma_{B_{ini}} \) for the bias in each of the deterministic model output variables \( C_M \) and \( C_S \) and their (joint) correlation time \( \tau \). The priors of \( \sigma_{B_{eq}} \) and \( \sigma_{B_{ini}} \) were chosen as Normal distributions with mean zero and a standard deviation of 0.5, truncated at zero to avoid negative values. For the parameters \( \theta \) of the deterministic model, we used independent Lognormal priors with the means at the correct values given above and standard deviations of 50% of the corresponding prior means. For the parameters \( \phi \) and the correlation time \( \tau \), Lognormal priors with the means at the correct values in case of the observation error parameters (given above) and at 0.3 for \( \tau \) and with standard deviations of 10% were used. The joint prior parameter distribution was the independent combination of these individual marginals. The choice of the prior distribution was made according to the previous paper and was based on our experience with such models. It fulfills the criteria outlined in sections 2.1 and 2.2 required for the identifiability of the Gaussian process parameters. Different to the application in Dietzel and Reichert [2012], it was not needed to calibrate the logarithms of the parameters instead of the parameters themselves.

For more details about the didactical example, we refer to Reichert and Schuwirth [2012] and Dietzel and Reichert [2012]. For the emulation of the log of the unnormalized posterior probability density of the microbial growth model and its error model, a design data set was used that consisted of 1000 parameter vectors and the respective unnormalized log posterior densities calculated according to equation (6). All parameters \( \gamma = (\theta, \phi, \xi) \) were used. The design parameter set was randomly drawn from a Normal distribution representing a simple approximation to the posterior distribution of the model. This approximating Normal distribution was derived from a sequence of iterative importance sampling steps starting from a random sample from a Uniform distribution in a neighborhood of the numerical maximum of the posterior as described in Dietzel and Reichert [2012]. The emulator was conditioned on this design data set and the same Normal distribution was used as a prior mean of the emulator. It directly emulates the dependence of the posterior on deterministic and error model parameters without explicitly addressing the intermediate step of calculating the results of the deterministic growth model. Hence, it is also independent of, in this didactical example synthetic, observation data. We then ran a Markov chain Monte Carlo sampling with this emulator. Four iterations with chains of length 10,000 were run for the optimization of the proposal distribution, the final chain was calculated with a length of 100,000. The chain was thinned by a factor of 5 and every 100th element of the resulting chain was propagated through the microbial growth model for the prediction of model results and derivation of uncertainty bounds as described in section 2.4. This means that only 200 additional simulations with the microbial growth model were needed for prediction.

We evaluated the performance of the emulator for different setups by applying the emulator to additional new input data also drawn from the basic Normal distribution and again calculating the unnormalized log posterior value by applying equation (6) to the new parameter set and comparing the results to the emulator results.

### 3.2. Emulator Setup and Performance

In this section, we discuss aspects of the most suitable setup of the emulator, mainly focused on the choice of the roughness parameter \( \zeta \), the correlation lengths \( \lambda_k \) (see equation (10)), the design data set, and the prior mean of the emulator. Some problems appeared during the set up of the emulator and the subsequent calculation of a Markov chain Monte Carlo sample with the presented emulator giving the result of the log of the unnormalized posterior density of the microbial growth model and its error model. A first difficulty arose from the choice of a suitable design data set. One critical aspect is the size of the design data set. A sample size of 100 parameter vectors and the resulting unnormalized log posterior densities led to numerical problems already for the initialization of the emulator during the inversion of the covariance matrix.
matrix. Although the didactical example is relatively simple, it is comprised of a parameter space with already 13 dimensions. This makes an interpolation in all dimensions difficult if only few points are available. A design data set of size 1000 appeared to be more adequate.

Furthermore, tests with a parameter sample from a Uniform distribution enveloping a neighborhood of the posterior mode failed. We used such samples [Reichert and Schuwirth, 2012] to initiate sequential importance sampling to find an approximation of the posterior by a Normal distribution [Dietzel and Reichert, 2012]. With the sample from a Uniform distribution in a neighborhood of the maximum, it was possible to initialize the emulator and train it with this design data set. However, the calculation of a Markov chain failed, typically because the rejection frequency was too large, while the parameter space was poorly explored. This is an indication that, due to the high dimension of the parameter space, only a few points of the design data set are close to the posterior maximum for a Uniform sample in its neighborhood. Hence, the emulator provided a poor approximation to the log posterior with local maxima close to the design points but poor emulation in between. It became obvious that a design parameter set drawn from a Normal distribution with higher probability density around the maximum of the posterior is much more appropriate and avoids this effect. After having made these experiences, we used a random parameter sample of size 1000 from the Normal distribution achieved from iterative importance sampling for an approximation to the posterior from an earlier study [Dietzel and Reichert, 2012]. The same Normal distribution was used as the prior mean of the emulator.

As a second step, we aimed for a good performance of the emulator by choosing the optimal combination of the roughness parameter \(\alpha\) and correlation lengths \(\lambda_k\). We did so by analyzing the root-mean-square error between the emulator interpolating between the design data points for a new set of parameters and the true log posterior density of the same parameter set. As we are not limited by computational time for the didactical example, we were able to explore the results of the method for different emulator setups compared to the results of the real model using the simulations of the deterministic growth model. For this purpose, we drew a sample of 1000 parameter vectors from the approximate Normal distribution and calculated the unnormalized log posterior densities for those parameter combinations, once with the emulator trained on a different design data set and once by the true model according to equation (6), for which the results of the deterministic model \(y_D(x, \theta)\) are needed. The root-mean-square error (RMSE) of emulation could then be calculated and it could be compared for different combinations of values of \(\alpha\) and \(\lambda_k\) (see Bastos and O’Hagan [2009] for other diagnostics). For the exponent \(\alpha\), we tested seven discrete values between 1 and 2. For the 13 correlation lengths \(\lambda_k\) for each parameter, we only tested different correlation lengths \(\lambda_k\) between 0.0001 and 75, which were multiplied with the range of the parameters to get the absolute correlation lengths. In general, it would be more desirable to choose the correlation length for each parameter separately. This could also enhance the performance of the emulator. However, the choice of correlation lengths is a difficult task. As discussed above, a key element of this choice is the typical length between sampling points in each dimension. This can be considered by multiplying a joint correlation length factor with the individual ranges of all parameter to get the individual correlation lengths. Figure 2 shows the RMSEs of emulation for all tested correlation length factors for \(\alpha\) equals 1, 1.5, 1.9, and 2.

It is important to point out that the results depend on the random choice of new input parameter sets for which the emulator and the true log posterior are calculated. Changing these sets slightly changes the results, while the overall pattern stays the same, as we saw in several replicates of the same calculation. The results shown in Figure 2 indicate that for small correlation lengths, the root-mean-square error between emulation and true value decreases with increasing correlation lengths. This is meaningful, as a larger correlation length should smooth the interpolation between design data points and decrease the influence of the prior emulator mean (see Figure 1 for an illustration). However, there seems to be an optimum for the correlation lengths, after which the RMSE increases again for even larger correlation lengths. Additionally, we can see that for increasing values of \(\alpha\), the RMSE is decreasing more strongly. We have to emphasize that an emulation with \(\alpha = 2\) in combination with large correlation lengths leads to the problem that the inversion of the covariance matrix \(K^{-1}(\nu)\) (see equation (8)) leads to numerical difficulties. All in all, the smallest RMSE can be achieved with \(\alpha = 2\) and \(\lambda_{\text{fact}} = 5\). The first tries with the emulator were therefore done with this emulator setup. However, for this combination, the Markov chain Monte Carlo sampling resulted in diverging chains causing bimodal and sometimes unrealistically wide posterior parameter distributions. This also caused unrealistic best estimates for the model parameters leading to poor model predictions. A
possible explanation for these phenomena is that for \( \alpha = 2 \), the domains of good emulation and of poor emulation due to oscillations are very close to each other regarding the \( \lambda_k \) and are not even clearly distinguishable in the space at emulated parameters (see Figure 1). Although \( \alpha = 1.9 \) and a relative correlation length of \( \lambda_{\text{fact}} = 1 \) shows a slightly higher RMSE in the example above, this seemed to be a better choice due to its higher robustness against oscillations. Similarly good results could even be achieved with \( \alpha = 1 \) and \( \lambda_{\text{fact}} = 1 \).

### 3.3. Results and Discussion

In the following, we show the results for the calibration and uncertainty analysis of the microbial growth model with the help of the GASP emulator as introduced in section 2. We focus on the results for \( \alpha = 1.9 \) and \( \lambda_{\text{fact}} = 1 \). Figure 3 shows the marginal prior and posterior distributions of all model parameters. The posterior distributions are derived from the Markov chain with 100,000 elements based on the calculation of the log of the unnormalized posterior by the GASP emulator. They are compared with the approximation by a Normal distribution (in that case their logarithm was approximated by a Normal distribution) as shown in Dietzel and Reichert [2012] and by a Markov chain Monte Carlo technique based on simulations with the deterministic model as presented in Reichert and Schuwirth [2012].

In general, all marginal distributions derived from emulation show a better accordance with the full MCMC method than the ones of the Normal approximation. Moreover, the best estimates of the parameters represented by the mode of the posterior distribution are closer to the true parameter values (vertical lines). The results of a Kolmogorov-Smirnov test show distances between the marginals derived by the Normal
approximation and the full MCMC method that vary between 0.05 and 0.69 and have a median of 0.39. The distances between those derived from emulation and the full MCMC method vary between 0.06 and 0.4 with a median of 0.24. These findings clearly indicate an improved posterior approximation by the GASP emulator than by the Normal distribution, but also the improved estimates are still not completely satisfying. For the GASP emulator approach, every 500th element of the full Markov chain (every 100th element of the thinned chain), likely to represent a random sample, was propagated through the model (compared to the full MCMC method for which the whole Markov chain was used) for prediction. The results for the prediction as described in section 2.4 are presented in Figure 4, again as a comparison of the three different prediction approaches and for both, the calibration and the validation time intervals.

Overall, the results for the three different approaches are very similar. As the best parameter estimates for the Markov chain on basis of the emulator are closer to the full Markov chain Monte Carlo results and to the true values, the deterministic model results and the bias-corrected output are also closer to the full Markov chain Monte Carlo results than by the Normal distribution.
chain results than for the approximation of the posterior by a Normal distribution. The 95% credibility intervals for model predictions resulting from parameter uncertainty, bias, and observation error as well as for the bias and of the observation error of the two output variables \(C_M\) and \(C_S\) are very similar for all three approaches. Due to the smaller subsample of the Markov chain Monte Carlo method based on the GASP emulator, (b) the middle column of posterior approximation by a Normal distribution from importance sampling [Dietzel and Reichert, 2012], and (c) the right column the results of the full MCMC method [Reichert and Schuwirth, 2012].

**Figure 4.** Results of the median and 95% credibility intervals for both variables (top row) \(C_M\) and \(C_S\) (observations used for calibration are marked by solid symbols, those not used for calibration by open symbols; output of the deterministic model (long-dashed line), bias-corrected output (deterministic model plus bias; median as solid line, 95% credibility bounds as dark gray area bounded by dashed lines), and prediction for new observations (including observation error; median as solid line (same as for bias-corrected output), 95% credibility bounds shown as dark and light gray areas bounded by dotted lines)) and results of the median and 95% credibility intervals for the bias (solid line and dark gray areas bounded by dashed lines) and of the observation error (dots and vertical line segments for the variables (middle row) \(C_M\) and (bottom row) \(C_S\). (a) The left column shows the results of a Markov chain Monte Carlo method based on the GASP emulator, (b) the middle column of posterior approximation by a Normal distribution from importance sampling [Dietzel and Reichert, 2012], and (c) the right column the results of the full MCMC method [Reichert and Schuwirth, 2012].

**Table 1.** Prior Marginals for the Calibrated Parameters \(\theta\) of the Deterministic Model

<table>
<thead>
<tr>
<th>Name</th>
<th>Unit</th>
<th>Distribution</th>
<th>Mean</th>
<th>Standard Deviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(k_{\text{gro}, \text{ALG}, T_0})</td>
<td>day(^{-1})</td>
<td>Lognormal</td>
<td>1.79</td>
<td>0.5</td>
<td>Growth rate of phytoplankton at reference temp./sat. light intensity</td>
</tr>
<tr>
<td>(k_{\text{death}, \text{ALG}, T_0})</td>
<td>day(^{-1})</td>
<td>Lognormal</td>
<td>0.0721</td>
<td>0.05</td>
<td>Death rate of phytoplankton at reference temperature</td>
</tr>
<tr>
<td>(k_{\text{gro}, \text{ZOO}, T_0})</td>
<td>day(^{-1})</td>
<td>Lognormal</td>
<td>0.397</td>
<td>0.25</td>
<td>Growth rate of zooplankton at reference temperature</td>
</tr>
<tr>
<td>(k_{\text{death}, \text{ZOO}, T_0})</td>
<td>day(^{-1})</td>
<td>Lognormal</td>
<td>0.0675</td>
<td>0.05</td>
<td>Death rate of zooplankton at reference temperature</td>
</tr>
<tr>
<td>(k_{\text{feed}})</td>
<td>gDM m(^{-3})</td>
<td>Lognormal</td>
<td>0.295</td>
<td>0.3</td>
<td>Threshold phytoplankton concentration when zooplankton feeding switches to Monod limit.</td>
</tr>
<tr>
<td>(f_{\text{X, in}}) (transformed)</td>
<td></td>
<td>Normal</td>
<td>2</td>
<td>1</td>
<td>Inert fraction of allochthonous organic particles</td>
</tr>
<tr>
<td>(k_{\text{miner, aer}, T_0})</td>
<td>day(^{-1})</td>
<td>Lognormal</td>
<td>0.295</td>
<td>0.3</td>
<td>Aerobic mineralization rate at reference temperature</td>
</tr>
<tr>
<td>(k_{\text{miner, ane}, T_0})</td>
<td>day(^{-1})</td>
<td>Lognormal</td>
<td>0.295</td>
<td>0.3</td>
<td>Anaerobic mineralization rate at reference temperature</td>
</tr>
<tr>
<td>(k_{\text{miner, anox}, T_0})</td>
<td>day(^{-1})</td>
<td>Lognormal</td>
<td>0.295</td>
<td>0.3</td>
<td>Anoxic mineralization rate at reference temperature</td>
</tr>
</tbody>
</table>
the emulator application, the boundaries of the credibility intervals are slightly less smooth than for the other two approaches. All approaches show the separation into calibration and validation time intervals, apparent by the widening of the credibility intervals after time point 2. This reflects a significantly better posterior knowledge during the calibration period than during the prediction period.

4. Application to Biogeochemical and Ecological Lake Model

We applied the emulation technique described in section 2 also to a real case study, the calibration of a biogeochemical and ecological lake model. Several lake models have already been developed with different key application areas in mind [Arhonditsis and Brett, 2004; Jørgensen, 2010; Mooij et al., 2010]. These include SALMO [Benndorf and Recknagel, 1982], its further developments SALMO-1D and SALMO-HR [Baumert and Benndorf, 2005; Petzoldt et al., 2005], (DYRESM-) CAEDYM [Bruce et al., 2006; Hamilton and Schladow, 1997; Rinke et al., 2009; Romero et al., 2004; Schladow and Hamilton, 1997; Tanentzap et al., 2007; Trolle et al., 2008], and PROTECH [Elliott et al., 1999a, 1999b, 2000, 2005, 2006, 2007, 2010; Elliott and Thackeray, 2004; Reynolds et al., 2001]. Unlike these lakes models [Mooij et al., 2010], the Biogeochemical and Ecological Lake Model (BELAMO) [Omlin et al., 2001a, 2001b; Mieleitner and Reichert, 2006, 2008; Dietzel et al., 2013] aims for a mechanistic description of the mineralization processes in the sediment layers in addition to those in the water column. This also guarantees closing of the element cycles of phosphorus and nitrogen.

Due to its high computational demand, a full Markov chain Monte Carlo sampling cannot be performed for BELAMO. The possibility to calibrate this model and estimate parameter and prediction uncertainty relying on a Normal approximation of the posterior distribution to reduce the computational demand was explored in an earlier study [Dietzel and Reichert, 2012]. The posterior approximation by a Normal distribution seemed to be a crude simplification and the need to gain more information about the shape of the posterior distribution. This is done by the application of a GASP emulator in this section. There are few applications of emulators to lake water quality models, one application of a different response surface approach is by the application of a GASP emulator in this section. There are few applications of emulators to lake water quality models, one application of a different response surface approach.

4.1. BELAMO: Model Description

The Biogeochemical and Ecological Lake Model (BELAMO), implemented in AQUASIM (version 2.1f, http://www.aquasim.eawag.ch), a computer program for the identification and simulation of aquatic systems [Reichert, 1994, 1998], describes concentrations of nutrients, oxygen, organic particles, phytoplankton, and zooplankton in a lake. It is based on differential equations formulating mass balances of these constituents. Originally, it was implemented as a 1-D model for Lake Zurich [Omlin et al., 2001a], then analyzed for its transferability to the lakes Greifensee and Walensee [Mieleitner and Reichert, 2006] and subsequently spatially simplified to a

| Table 2. Prior Marginals for the Parameters $\phi$ of the Error Modela |
|-----------------|-----------------|-------------|---------------|----------------------------------|
| Name            | Unit            | Distribution | Mean          | Standard Deviation   | Description                                    |
| $\sigma_{ALG}$  | gWM$^{-1}/m^{1/2}$ | Lognormal    | 0.07          | 0.02               | Standard deviation of measurement error in phytoplankton |
| $\sigma_{ZOO}$  | gWM$^{-1}/m^{1/2}$ | Lognormal    | 0.08          | 0.02               | Standard deviation of measurement error in zooplankton    |
| $\sigma_{NO3}$  | gN$^{-1}/m^{1/2}$ | Lognormal    | 0.022         | 0.002              | Standard deviation of measurement error in nitrate       |
| $\sigma_{O2}$   | gO$^{-1}/m^{1/2}$ | Lognormal    | 0.011         | 0.001              | Standard deviation of measurement error in oxygen         |
| $\sigma_{POM}$  | gP$^{-1}/m^{1/2}$ | Lognormal    | 0.007         | 0.0002             | Standard deviation of measurement error in phosphate      |

*a*On a Box-Cox transformed scale.

| Table 3. Prior Marginals for the Calibrated Parameters $\zeta$ of the Biasa |
|-----------------|-----------------|-------------|---------------|----------------------------------|
| Name            | Unit            | Distribution | Mean          | Description                                    |
| $\zeta_{ALG}$   | gWM$^{-1}/m^{1/2}$ | Exponential  | 0.3           | Standard deviation of bias in phytoplankton          |
| $\zeta_{ZOO}$   | gWM$^{-1}/m^{1/2}$ | Exponential  | 0.41          | Standard deviation of bias in zooplankton            |
| $\zeta_{NO3}$   | gN$^{-1}/m^{1/2}$ | Exponential  | 0.11          | Standard deviation of bias in nitrate               |
| $\zeta_{O2}$    | gO$^{-1}/m^{1/2}$ | Exponential  | 0.5           | Standard deviation of bias in oxygen                |
| $\zeta_{POM}$   | gP$^{-1}/m^{1/2}$ | Exponential  | 0.035         | Standard deviation of bias in phosphate             |

*a*On a Box-Cox transformed scale.
Dietzel and Reichert [2012] applied a calibration technique to BELAMO that makes it possible to estimate the prediction uncertainty of structurally uncertain and computationally demanding models, while considering and exploring the uncertainties in the model parameters. This approach allows for a more robust assessment of the model's performance and reliability. The figure illustrates the prior and posterior distributions of several key model parameters, including the growth rate of phytoplankton in the oligotrophic (k_{growth_{OL}}) and eutrophic (k_{growth_{EO}}) zones, the death rate of phytoplankton in the oligotrophic (k_{death_{OL}}) and eutrophic (k_{death_{EO}}) zones, the nutrient (k_{mineral_{OL}}) and light (k_{light_{OL}}) limitation factors, and the nitrogen fixation rates (k_{fixation}).

Figure 5. Prior (dotted, supported by light gray shading of the areas below the lines) and posterior (supported by dark gray shading of the areas below the lines) marginals of the model parameters from the posterior approximation by the GASP emulator (solid) and a Normal approximation from a Uniform neighborhood/Normal importance sampling (dashed).

A four-box version for an analysis of functional phytoplankton groups [Mieleitner and Reichert, 2008]. Dietzel and Reichert [2012] applied a calibration technique to BELAMO that makes it possible to estimate the prediction uncertainty of structurally uncertain and computationally demanding models, while considering and exploring the uncertainties in the model parameters.
quantifying the contribution of model bias. This study was only done for Lake Zurich, but was followed by an uncertainty estimation and joint calibration of the lakes Greifensee, Lake Zurich, and Walensee [Dietzel et al., 2013]. Dietzel et al. (2013) also modified aspects of the model to reduce the bias to use the model for long-term simulations of the three lakes during a time with decreasing nutrient input loads.

The current version of BELAMO calculates concentrations of ammonium, nitrate, phosphate, oxygen, degradable and inert dead organic particles, and the total biomass of phytoplankton and zooplankton in the epilimnion, hypolimnion, and two sediment compartments of the lakes. These four model compartments are considered as well-mixed boxes, the model accounts for inflow to and outflow from the water column, gas exchange between the water column and the atmosphere, mixing between the two water compartments, sedimentation of particles, advection and diffusion of dissolved substances in the water column and in the pore water of the sediment compartments, sediment accumulation, and permanent burial. Within the model compartments, the model includes the description of growth, respiration, and death of phytoplankton and zooplankton, aerobic, anoxic, and anaerobic mineralization, nitrification, methane oxidation, and phosphate adsorption to sinking degradable and inert organic particles.

To learn about lake biogeochemistry and ecology from a model like BELAMO or to even use it for environmental decision support in the area of water quality and plankton dynamics, it is needed to assess the uncertainty of model predictions. As stated by Dietzel and Reichert (2012), despite the complexity of the lake model, it still represents a clearly simplified description of the real system that causes a considerable amount of bias. Furthermore, simulations are time consuming. During the development of the model and while conducting different application studies, it became obvious that the calibration of the lake model is a nontrivial problem. This is especially true for the application to a long-term period of up to 30 years of simulation time and for a joint calibration of three lakes of different trophic regime with keeping most parameters universal [Dietzel et al., 2013]. The time-consuming and difficult calibration process in the high-dimensional parameter and output spaces brought up the idea to use an emulator for a better approximation of the posterior distribution than the coarse approximation by Normal distribution that
was used in the previous studies [Dietzel and Reichert, 2012; Dietzel et al., 2013]. In the following, we apply the emulator described in section 2 to the calibration of BELAMO based on long-term observations of Lake Zurich.

4.2. Study Area
Lake Zurich is a deep mesotrophic lake in the northeastern part of the Swiss plateau. It is separated into two parts by a natural dam, lower Lake Zurich, the larger and deeper part of the two, served as a study lake. In lower Lake Zurich, measurements were taken at the location of largest depth (136 m). The watershed of the lake is mainly used for settlements and to a smaller amount for agriculture. More detailed descriptions of the properties of the lake can be found in Mieleitner and Reichert [2006] and Dietzel et al. [2013], both show a map of the study area as well.

4.3. Observation Data
Observations from Lake Zurich of the years 1976–2005 (zooplankton only since 1985) are available. We used the data up to 1995 for model calibration and the data from 1996 to 2005 for model validation. Ammonium, nitrate, phosphate, oxygen, and temperature were measured as profiles, consisting of samples in 19 different depths. Phytoplankton data consists of samples from 14 different depths, while zooplankton was only measured as two integrated samples over the upper and lower parts of the lake depth. These data were collected monthly by the Water Supply Authority of Zurich (Wasserversorgung Zürich, WVZ) and were aggregated to average concentrations in the epilimnion (upper 10 m of Lake Zurich) and hypolimnion (remaining 126 m of the lake). For inference and prediction, the five state variables nitrate, phosphate, oxygen, phytoplankton, and zooplankton were used. The observation and prediction layouts, $L_1$ and $L_2$, define combinations of these five state variables, locations (epilimnion or hypolimnion), and monthly time points between 1976 and 2005. As lake model input data, information about the nutrient input loads from inflowing rivers, waste water treatment plants, and wet deposition, as well as data about meteorological forcing could be obtained from federal and cantonal agencies and from technical reports. For a detailed description of data compilation and processing, we refer to Dietzel et al. [2013], specifically to its supporting information.

We assume that observation and structural errors are smaller for smaller concentrations of chemical and biological variables. To account for this heteroscedasticity, a Box-Cox transformation of data and deterministic model results were performed (see section 2.1). The first transformation parameter was chosen as 0.5, the second as 0. These parameters also could be estimated, which, however, would increase the dimension of the calibration problem.

4.4. Prior Distributions
To guarantee comparability with the results based on the Normal approximation of the posterior, the same prior distributions for the parameters was used as in Dietzel and Reichert [2012]. Tables 1–3 specify the marginals; the joint distribution was taken to be the product of these marginals.

The choice which parameters of the deterministic model, $\theta$, should be calibrated was based on sensitivity and identifiability analyses (see Dietzel et al. [2013] and partly Omlin et al. [2001b] for more details).
The priors are given in Table 1. We assumed log-normal priors for the standard deviations of the observation errors representing the parameter vector $\psi$. These values are listed in Table 2 in Box-Cox transformed units. Table 3 contains the priors for the parameters $\xi$ of the bias in Box-Cox transformed units. We chose Exponential distributions for which the probability density increases with decreasing value of the standard deviation. This reflects our desire to avoid bias if possible (see also the discussion in section 2.1). The correlation time $\tau$ was set to about 4 months, as this is a typical order of magnitude of structures in the yearly pattern of the state variables in the lake. This parameter was not included in the calibration procedure, as this parameter would be difficult to identify and anyway an informative prior would be needed to inform the inference procedure about which time scales to represent with the two error terms $B_{E}(x, \xi)$ and $E_{B}(\psi)$ in equation (1) (see also the discussion in section 2.1).

The product of these priors was used for the calculation of the unnormalized posterior density according to equation (6) at the design data points with which the emulator was trained.

### 4.5. Emulator Setup

A similar setup of the emulator was chosen for the lake model as for the didactical example, in particular, the emulator parameters were set to $x = 1.9$ and $\lambda_{emt} = 1$. Note that the computational requirements of the lake model did not allow an extensive study of the approximation accuracy of the posterior by the emulator as we could perform for the didactical example (see Figure 2 and the corresponding discussion in the text). However, as the results shown in Figure 1 indicate that a good choice of the correlation lengths seems to be primarily dependent on the resolution of the design parameter set and not on the behavior of the deterministic model, it seemed reasonable to use the same relative length scale for the correlation lengths for the lake model as well. As a design data set, we drew a random sample of 2000 parameter vectors from an uncorrelated Normal distribution with the mean at the maximum of the posterior that was derived from an earlier calibration [Dietzel and Reichert, 2012]. A constant, the mean of the unnormalized log posterior densities of the design data set, was used as the prior emulator mean. Again, the emulator describes the

![Figure 8](attachment:figure8.png)

Figure 8. (a and b) Nitrate and (c and d) phosphate concentrations in the epilimnion (Figures 8a and 8c) and hypolimnion (Figures 8b and 8d) of Lake Zurich. Data points (dots), output of the deterministic model (long dashed), median (solid) and 95% credibility bounds (dark gray area with dashed boundaries) of bias-corrected output, and median (solid; same as for bias-corrected output) and 95% credibility bounds (dark and light gray areas with dotted boundaries) of predictions of new observations (including observation error) for the whole simulation time.
dependence of the posterior on the parameters of the lake model and its error model, bypassing the intermediate step of calculating results of a deterministic model and constructing a likelihood function based on these. A single evaluation of the emulated log posterior (needed for one step of the Markov chain) takes 0.05 s compared to 20 min for an evaluation with the full model. This increase could even be extended by implementing the emulator in an efficient programming language rather than in R (which was more convenient for this feasibility study). With this emulator setup, it was possible to calculate a Markov chain of length 100,000 of the emulated unnormalized log posterior with autocorrelation lengths with a median of 50. The proposal distribution of the Markov chain had before been adjusted using four shorter Markov chains of length 10,000. To derive prediction uncertainty bands, every 20th element from the final Markov chain thinned by a factor of 5 (i.e., the final chain has the length 20,000) was propagated through the model, which resulted in 1000 simulations with the lake model.

4.6. Results and Discussion

For the emulator setup described in the previous section, Figure 5 shows the marginal prior and posterior distributions of all lake model and error model parameters. The results of the emulator are compared to the posterior marginals derived from a Normal approximation by an importance sampling technique shown in
Dietzel and Reichert [2012]. In many cases, both techniques show similar results, for many but not for all parameters, the emulation leads to wider posterior marginals.

The model results and their prediction uncertainties for the considered deterministic model output variables phytoplankton, zooplankton, nitrate, phosphate, and oxygen derived from the emulation of the posterior distribution and propagation of a Markov chain subsample are presented in Figures 6–8. The results show large similarities with the results calculated by a Normal approximation to the posterior shown in Dietzel and Reichert [2012]. The uncertainty bands shown in these figures represent the posterior knowledge of the “true” values of the plotted variables (dark gray bands) and of potential observations (light gray bands). As only the data up to 1995 were used for calibration, this posterior knowledge is much less uncertain for the time period from 1976 (1985, respectively) to 1995 than during the extrapolation and validation period from 1996 to 2005. This leads to a considerable widening of uncertainty bands in 1996. This is also obvious in Figure 9, where the median posterior estimates and the uncertainty of the bias and the observation error are shown. Additionally, the autocorrelated character of the bias and the randomness of the observation error of the model output variables are evident.

Several problems emerged during the set up of the emulator and the subsequent Markov chain Monte Carlo sampling. Some problems, the suitable choice of the exponent $\alpha$ and correlation lengths factors $k_{fact}$, as well as the choice of the design data set, had already been explored by the didactical example. Additionally, the size of the design data set had to be increased for the lake model application due to the higher dimension of the parameter space (19 parameters: 9 lake model parameters, 5 observation error parameters, 5 bias parameters). For the lake model application, it was not possible to derive the design data set from the Normal distribution found as an approximation of the posterior distribution in an earlier study [Dietzel and Reichert, 2012], neither to use this approximation as the prior mean of the emulator. These problems resulted from the large correlations and narrow form of the Normal distribution derived from the importance sampling. This problem was solved by sampling from an uncorrelated Normal distribution drawn at a larger neighborhood of the posterior mode as a design data set. Using a different, even simpler prior emulator mean, just a constant value, improved the emulator performance as well. Furthermore, a larger subsample of the resulting Markov chain was propagated through the lake and error model to enhance the estimation of prediction uncertainty bands. For the inference process, the emulator was able to reduce the original 20 min of simulation time for calculating the posterior density for an observation period of 20 years to 0.05 s.

5. Conclusions

In the current study, we suggest to replace an approximate parametric description of the posterior of the parameters of a computationally demanding model by an approximation of the posterior by a GASP emulator to be able to do Markov chain Monte Carlo sampling for Bayesian inference. This can be particularly interesting for models with structurally complex output, such as multivariate time series, for which it is not easy to construct an emulator for the output. A subsample of the Markov chain can then be propagated through the full model to obtain model predictions. We illustrated the application of this technique to a model consisting of a deterministic model, an additive bias correction term, and independent observation errors. However, the suggested technique is not limited to this case. Due to the simple structure of the emulator, a single scalar function of the model parameters, the computational gain for parameter inference can be expected to be considerable. The main disadvantages of the proposed technique are the need of the full model for prediction and potential difficulties in selecting an adequate design parameter set and emulator parameter priors for emulator conditioning.

We could show for a simple example that the presented method results in a better posterior approximation than achieved when using a Normal approximation to the posterior. The technique has the advantage to not make any assumptions about the shape of the posterior distribution. The results could be gained at only slightly higher computational costs than for the Normal approximation to the posterior and linear error propagation. A first maximization of the posterior distribution is needed for both techniques. For statistical inference of model parameters, Markov chain Monte Carlo sampling of the emulator, rather than the full model, is needed, which is computationally very cheap. In our second, more complex, example with the lake model BELAMO, Markov chain Monte Carlo runs were faster by a factor of more than 10,000 when
using the emulator compared to using the full model. However, runs of the full model are still required for a design parameter set of size 2000 to condition the emulator and for a subsample of the Markov chain of size 1000 for estimating prediction uncertainty. However, these simulations are easier to do in parallel than MCMC sampling.

Besides these positive results of the presented emulation technique, we also discovered some problems with the optimal setup of the GASP emulator. It became obvious that the evaluation of emulator performance by its root mean square error (RMSE) for a small test sample can be misleading. Due to the small size of the test sample, it is possible to miss oscillations of the emulator leading to poor behavior of the Markov chain even for emulator parameters for which the emulator seems to approximate the posterior well. In particular, the choice of the design data set, the prior emulator mean, and the emulator parameters \( \alpha \) and \( \lambda k \) are crucial factors for the quality of the method that need more investigation. These choices are difficult as, in situations in which emulation is interesting, the computational demand of the full model does not allow an extensive testing of emulator performance.

The suggested emulation technique may replace parametric approximations of the posterior of computationally demanding models, which is still the dominant approximation technique described in textbooks about Bayesian inference. The main advantage over the parametric approximation is the better performance at only slightly increased computational effort. On the other hand, the difficulties in finding an adequate design data set and optimal emulator parameters show the limitations of this approach and indicate that more research in this and alternative techniques is still required.

Acknowledgments

This research was funded by the Swiss National Science Foundation (SNSF). We are grateful to Richard Forster and Oliver Köster from WZ2, Barbara Känel, Pius Niederhauser, Irene Purschert, and Peter Spohn from AWEL, and Daniel Streit and Christian Leu from BAFU for the kind provision of data on chemistry, biology, and physics of Lake Zurich, discharging waste water treatment plants and inflowing rivers. We thank Carlo Albert, Dario Del Giudice, Andreas Scheidegger, and Nele Schuwirth from Eawag for helpful comments to earlier versions of the manuscript and specifically one reviewer for the careful consideration of the manuscript.

References


Li, R., and A. Sudijanto (2005), Analysis of computer experiments using penalized likelihood in Gaussian kriging models, Technometrics, 47(2), 111–120.


Santner, T. J., B. J. Williams, and W. Notz (2003), The Design and Analysis of Computer Experiments, Springer, N. Y.


