AGU ADVANCING EARTH AND SPACE SCIENCES

Water Resources Research

RESEARCH ARTICLE

10.1029/2024WR037735

Extending GLUE With Multilevel Methods to Accelerate Statistical Inversion of Hydrological Models

Max Gustav Rudolph[1], Thomas Wöhling[2,3], Thorsten Wagener[4], and Andreas Hartmann[1]

[1]Institute of Groundwater Management, TUD Dresden University of Technology, Dresden, Germany, [2]Chair of Hydrology, Institute of Hydrology and Meteorology, TUD Dresden University of Technology, Dresden, Germany, [3]Lincoln Agritech, Lincoln, New Zealand, [4]Institute of Environmental Science and Geography, University of Potsdam, Potsdam, Germany

Key Points:

- We extend the Generalized Likelihood Uncertainty Estimation methodology to a strategy that considers multiple resolution levels (MLGLUE)
- We demonstrate the acceleration with MLGLUE for different spatial (groundwater flow model) and temporal (rainfall-runoff model) resolutions
- We show that inversion results from multilevel and conventional methods align and discuss factors affecting computational efficiency

Correspondence to:

M. G. Rudolph,
max_gustav.rudolph@tu-dresden.de

Citation:

Rudolph, M. G., Wöhling, T., Wagener, T., & Hartmann, A. (2024). Extending GLUE with multilevel methods to accelerate statistical inversion of hydrological models. *Water Resources Research*, *60*, e2024WR037735. https://doi.org/10.1029/2024WR037735

Received 12 APR 2024
Accepted 30 SEP 2024

Author Contributions:

Conceptualization: Max Gustav Rudolph
Formal analysis: Max Gustav Rudolph
Methodology: Max Gustav Rudolph
Resources: Andreas Hartmann
Software: Max Gustav Rudolph
Supervision: Thomas Wöhling, Thorsten Wagener, Andreas Hartmann
Writing – original draft: Max Gustav Rudolph
Writing – review & editing: Thomas Wöhling, Thorsten Wagener, Andreas Hartmann

Abstract Inverse problems aim at determining model parameters that produce observed data to subsequently understand, predict or manage hydrological or other environmental systems. While statistical inversion is especially popular, its sampling-based nature often inhibits its application to computationally costly models, which has compromised the use of the Generalized Likelihood Uncertainty Estimation (GLUE) methodology, for example, for spatially distributed (partial) differential equation based models. In this study we introduce multilevel GLUE (MLGLUE), which alleviates the computational burden of statistical inversion by utilizing a hierarchy of model resolutions. Inspired by multilevel Monte Carlo, most parameter samples are evaluated on lower levels with computationally cheap low-resolution models and only samples associated with a likelihood above a certain threshold are subsequently passed to higher levels with costly high-resolution models for evaluation. Inferences are made at the level of the highest-resolution model but substantial computational savings are achieved by discarding samples with low likelihood already on levels with low resolution and low computational cost. Two example inverse problems, using a rainfall-runoff model and groundwater flow model, demonstrate the substantially increased computational efficiency of MLGLUE compared to GLUE as well as the similarity of inversion results. Findings are furthermore compared to inversion results from Markov-chain Monte Carlo (MCMC) and multilevel delayed acceptance MCMC, a corresponding multilevel variant, to compare the effects of the multilevel extension. All examples demonstrate the wide-range suitability of the approach and include guidelines for practical applications.

1. Introduction

Inverse problems are ubiquitous in hydrological modeling, emerging in the context of parameter estimation, system understanding, sustainable water resources management, and the operation of digital twins (e.g., Leopoldina, 2022). Computational models are often highly parameterized and non-linear, posing substantial challenges to parameter inversion approaches. Furthermore, observations of system states are affected by measurement uncertainty and the knowledge about the underlying system is incomplete, resulting in uncertainties associated with computational models (Beven, 1993; Wagener & Gupta, 2005; Carrera et al., 2005; Beven, 2006; Vrugt, ter Braak, Gupta, & Robinson, 2009; Laloy & Vrugt, 2012; Zhou et al., 2014; Mai, 2023). We need to quantify these uncertainties if models should be used for scientific inquiry or in support of decision-making (Blöschl et al., 2019; Page et al., 2023). While process-based spatially distributed models are increasingly used to guide decision-making and to sustainably manage water resources, such modeling approaches are computationally costly (Doherty, 2015; Herrera et al., 2022), making uncertainty quantification (UQ) and statistical inversion especially challenging (Erdal & Cirpka, 2020; Kuffour et al., 2020; White, Hunt, et al., 2020). There is a need to develop computationally efficient approaches to UQ and statistical inversion to overcome the pressing challenges associated with climate change and their impact on water resources.

Various approaches to UQ have been developed and applied in that respect; the Bayesian approach to statistical inversion and UQ, however, is especially popular due to the ability to comprehensively treat uncertainties in state variables, parameters, and model output (Linde et al., 2017; Montanari, 2007; Page et al., 2023; Vrugt, 2016). Generalized Likelihood Uncertainty Estimation (GLUE) (Beven & Binley, 1992, 2014) - as an informal Bayesian approach - and Markov-chain Monte Carlo sampling (MCMC) (Brunetti et al., 2023; Cui et al., 2024; Dodwell et al., 2019; Gallagher et al., 2009; Lykkegaard et al., 2023; Vrugt, 2016) - as a formal Bayesian approach - are frequently applied in the environmental sciences for statistical inversion. Bayesian frameworks consider model parameters to be random variables that are associated with prior distributions, which are conditioned on system

state observations using a likelihood function to posterior distributions. The likelihood function may either be defined formally or informally, depending on the belief and assumptions made about sources of error and the intended properties of the likelihood function itself, and many different approaches exist to define such functions (Beven, 2016; Beven & Binley, 1992; Beven & Freer, 2001; Nott et al., 2012; Sadegh & Vrugt, 2013; Schoups & Vrugt, 2010; Vrugt & Beven, 2018).

Approaches to statistical inversion generally rely on repeatedly running the computational model with different parameter values (i.e., repeatedly solving the forward problem) to obtain outputs that can be compared to observations of the same variable, if available. With computationally costly models, this approach quickly becomes intractable and there is a need to develop more efficient sampling approaches for statistical inversion. Different approaches have been developed to reduce computational cost of inversion, such as using data-driven surrogate or reduced-order models during inversion, which are then often run instead of the computationally costly high-fidelity model (Allgeier, 2022; Asher et al., 2015; Burrows & Doherty, 2015; Doherty & Christensen, 2011; Gosses & Wöhling, 2019, 2021; Linde et al., 2017). Reducing model spatial resolution can reduce model complexity and computational cost in general and the effect of horizontal (Reinecke et al., 2020; Savage et al., 2016; Wildemeersch et al., 2014) as well as vertical (White, Knowling, & Moore, 2020) discretization in model performance has been studied before, also in the context of accelerating inversion (von Gunten et al., 2014).

Multilevel methods and multilevel Monte Carlo (MLMC) (Cliffe et al., 2011; Giles, 2008, 2015; Heinrich, 2001; Peherstorfer et al., 2018), with extensions to multilevel MCMC and multilevel delayed acceptance MCMC (MLMCMC and MLDA, respectively) (Cui et al., 2024; Dodwell et al., 2019; Lykkegaard et al., 2023), were previously introduced with the motivation of reducing the computational cost of Monte Carlo estimators. For spatially distributed models, multilevel methods utilize multiple levels of spatial domain resolution. Together with the most finely discretized highest-level model, several more coarsely discretized lower-level models are considered. Most solutions to the forward problem are then found using lower-level models while the highest-level model is executed far less frequently, harboring the potential for large savings in overall computation time. Contrary to surrogate- or reduced-order-model-aided approaches to UQ, multilevel methods make no simplifying assumptions about the model and the relevant processes are simulated directly on all resolution levels. Another contrast is that the coarsely discretized models are not used instead of the high-fidelity model but they are synergistically used together. Linde et al. (2017) summarize first applications of MLMC for the forward propagation of uncertainties in hydrogeology and hydrogeophysics. Multilevel methods can be used with all types of models where a notion of model resolution exists. Typically, multilevel methods are applied to models based on (partial) differential equations (PDEs) using different spatial grid resolutions (e.g., in numerical groundwater flow models) or different temporal resolutions (e.g., in rainfall-runoff models). We note that multifidelity methods exist as well, where the notion of model resolution is generalized as compared to multilevel methods (Peherstorfer et al., 2018). A low-fidelity model does not need to be associated with lower resolution but can be any computationally cheaper approximation to the problem, such as a regression-based model, and we refer the reader to Ng and Willcox (2014), Peherstorfer et al. (2016), and Peherstorfer et al. (2018) for detailed discussions.

Previous applications of multilevel methods focussed on models with different spatial resolutions (Cliffe et al., 2011; Cui et al., 2024; Dodwell et al., 2019; Linde et al., 2017; Lykkegaard et al., 2023), entailing challenges when transferring parameter fields from one spatial resolution to another. To this end, utilizing point measurements of parameters or the combination with other predictor variables, Gaussian process regression is frequently used to generate conditioned parameter fields on any desired spatial resolution (Doherty, 2003; Kitanidis & Vomvoris, 1983; Zhou et al., 2014; Zimmerman et al., 1998). Unconditioned random fields are also utilized, where parameter fields are generated on any desired spatial resolution (Y. Liu et al., 2019); using uncorrelated and spatially independent random variables, the Karhunen-Loéve expansion is frequently employed to parameterize the random field (Cliffe et al., 2011; Cui et al., 2024; Dodwell et al., 2019; Lykkegaard et al., 2023). The definition of hydrological response units or internally homogeneous zones of parameters represents another strategy for parameterization (Anderson et al., 2015; Kumar et al., 2013; White, 2018; Zhou et al., 2014). Parameter scaling can be used to transfer parameter fields from one spatial resolution to another. While there is no generally valid theory for upscaling (i.e., from fine to coarse grids) (Binley et al., 1989; Samaniego et al., 2010), various upscaling operators are used in practice (Binley et al., 1989; Colecchio et al., 2020; Samaniego et al., 2010).

Geostatistical approaches, such as Gaussian process regression, are often used to (initially) assign parameters for spatially distributed groundwater flow- or other hydrological models. This simultaneously reduces over-parameterization as the number of geostatistical parameters is much lower than the number of parameters of the computational model. Also reducing effects of overparameterization (such as non-uniqueness, equifinality, and reduced identifiability) and to better constrain the parameter space during inversion, regularization can be employed in combination with different parameterization strategies (Moore & Doherty, 2006; Moore et al., 2010; Pokhrel et al., 2008; Tonkin & Doherty, 2005).

While multilevel methods have previously been used to accelerate MCMC algorithms (Cui et al., 2024; Dodwell et al., 2019; Lykkegaard & Dodwell, 2022; Lykkegaard et al., 2023) in a formal Bayesian framework, they have not yet been applied in connection with GLUE. In this study, we utilize ideas from multilevel Monte Carlo strategies to accelerate statistical inversion of hydrological models with the GLUE methodology. After introducing multilevel GLUE (MLGLUE), two example inverse problems are considered. We subsequently apply conventional GLUE and MLGLUE as well as MCMC and MLDA to those problems and compare the results.

While we are aware of the debate about the (non-) formality of statistical inference paradigms, such as GLUE, this discussion is not subject of the present work. Vrugt, ter Braak, Gupta, and Robinson (2009), Nott et al. (2012), and Sadegh and Vrugt (2013) made efforts to integrate contradicting views and we continue on this track. One needs to make pragmatic choices for any kind of inference and ours is to make statistical inversion - and GLUE specifically - more accessible for cases where it would previously be computationally intractable.

2. Methods

2.1. The Inverse Problem

Consider observations $\widetilde{\mathbf{Y}} = \left[\tilde{y}_1, \ldots, \tilde{y}_k\right]^T \in \mathcal{Y} \subseteq \mathbb{R}^k$ of a real system and consider a model $\mathcal{F}$ that simulates the system response $\mathbf{Y} = \left[y_1, \ldots, y_k\right]^T \in \mathcal{Y}$ corresponding to $\widetilde{\mathbf{Y}}$. With fixed initial and boundary conditions, the model output depends on model parameters $\boldsymbol{\theta} \in \mathcal{X} \subseteq \mathbb{R}^n$

$$\widetilde{\mathbf{Y}} = \mathcal{F}(\boldsymbol{\theta}) + \boldsymbol{\varepsilon}. \tag{1}$$

The parameter vector $\boldsymbol{\theta}$ is considered a random vector with an associated prior distribution $p_p(\boldsymbol{\theta})$ (Kavetski et al., 2006; Vrugt, ter Braak, Gupta, & Robinson, 2009). $\boldsymbol{\varepsilon} \in \mathbb{R}^k$ in this context represents the combined effect of conceptual model error and measurement error (e.g., M. C. Kennedy & O'Hagan, 2001; Plumlee, 2017); subsequently we refer to $\boldsymbol{\varepsilon}$ simply as error and refer to the aforementioned references for more detailed discussions on errors.

Solving the inverse problem in a Bayesian statistical framework means to obtain the posterior distribution of the parameters $p\left(\boldsymbol{\theta}|\widetilde{\mathbf{Y}}\right)$ via Bayes' theorem

$$p\left(\boldsymbol{\theta}|\widetilde{\mathbf{Y}}\right) = \frac{p_p(\boldsymbol{\theta}) p\left(\widetilde{\mathbf{Y}}|\boldsymbol{\theta}\right)}{p\left(\widetilde{\mathbf{Y}}\right)} \propto p_p(\boldsymbol{\theta}) p\left(\widetilde{\mathbf{Y}}|\boldsymbol{\theta}\right) \tag{2}$$

where $p\left(\widetilde{\mathbf{Y}}|\boldsymbol{\theta}\right)$ is the likelihood function and $p\left(\widetilde{\mathbf{Y}}\right)$ is the proportionality factor called model evidence, which is the integrated, prior-weighted likelihood of observing the data.

Assuming that errors $r_i = y_i - \tilde{y}_i$ are mutually independent, identically distributed (i.i.d.) and follow a Gaussian distribution with constant variance σ_r^2, the log-likelihood takes the form

$$\mathcal{L}\left(\boldsymbol{\theta}|\widetilde{\mathbf{Y}}\right) := \log p\left(\widetilde{\mathbf{Y}}|\boldsymbol{\theta}\right) = -\frac{k}{2}\ln(2\pi) - \frac{k}{2}\ln\left(\sigma_r^2\right) - \frac{1}{2}\sigma_r^{-2} \cdot \sum_{i=1}^{k} \left(y_i - \tilde{y}_i\right)^2. \tag{3}$$

The assumptions of i. i.d. errors, however, usually does not hold as these errors of hydrological models often exhibit strong autocorrelation and heteroscedasticity (see, e.g., Beven (2006) for a discussion). Beven and

Freer (2001) and Vrugt, ter Braak, Gupta, and Robinson (2009) give alternative likelihood formulations for non-Gaussian errors that often come at the cost of additional hyperparameters.

2.2. Multilevel Methods

Proposed by Heinrich (2001) and Giles (2008) to reduce the variance of Monte Carlo estimators and to make them computationally more efficient, multilevel methods rely on a simple concept. Instead of computing a Monte Carlo estimate (e.g., the expectation of a scalar model output) using a model with high-resolution, high accuracy, and high computational cost directly, most work is done using models with lower computational cost, lower resolution, and lower accuracy. Considering approaches to sampling such as Markov-chain Monte Carlo (see Section 2.3), the lower-level models with lower computational cost filter out poor samples before they are evaluated with higher-level models with higher computational cost, effectively increasing the acceptance rate of samples and therefore increasing computational efficiency (Lykkegaard et al., 2023). Consequently, most samples are evaluated or drawn on lower levels, which are associated with computationally cheaper low-resolution models. The computationally costly high-resolution model is evaluated far less frequently, harboring the potential for substantial computational savings; we refer to Cliffe et al. (2011), Giles (2015), and Lykkegaard et al. (2023) for more detail. Nevertheless, we now introduce some technical aspects which will be used later.

Consider a scalar quantity, $\mathbf{Q} = \mathcal{Q}(\mathcal{F}(\boldsymbol{\theta}))$, where $\mathcal{Q}$ represents some function of the model output. As an example, consider $\mathbf{Q}$ to represent the groundwater level at a fixed location in the model domain. Instead of one single model for a system, assume that there is a hierarchy of models and related quantities, which are denoted by $\{\mathcal{F}_\ell\}_{\ell=0}^L$ and $\{\mathbf{Q}_\ell\}_{\ell=0}^L$, respectively, where ℓ is the level index. In the context of PDE-based models, ℓ may be related to the grid size or time step length of the model. A larger ℓ corresponds to a higher domain resolution with smaller computational cells or smaller time steps, for example. $\mathcal{F}_L$ is considered the target model, associated with a certain (pre-defined) model resolution. As the level index increases, we assume that model computational cost also increases while the approximation error decreases. For ℓ close to 0, evaluations of $\mathbf{Q}_\ell$ are computationally cheap but inaccurate; for larger values of ℓ, evaluations of $\mathbf{Q}_\ell$ are more accurate but computationally more expensive. Reconsidering the idea from above, it follows that overall computational cost for sampling can be reduced if most model evaluations are performed on lower levels (small ℓ) and higher-level models (larger ℓ) are evaluated less frequently.

To assess computational efficiency and variance reduction of multilevel Monte Carlo (MLMC) estimators, Cliffe et al. (2011) analyze individual levels as well as the relationships between subsequent levels regarding $\mathbf{Q}_\ell$. While such analyses are not formally required for multilevel inversion (Cliffe et al., 2011; Lykkegaard et al., 2023), they offer valuable insights (see Section 2.4.2). $\mathbb{V}[\mathbf{Q}_\ell]$ and $\mathbb{E}[\mathbf{Q}_\ell]$ should be approximately constant as $\ell \to L$, ensuring that $\mathbf{Q}_\ell$ is a good enough approximation even on the coarsest level $\ell = 0$. Furthermore, $\mathbb{V}[\mathbf{Q}_\ell - \mathbf{Q}_{\ell-1}]$ and $\mathbb{E}[\mathbf{Q}_\ell - \mathbf{Q}_{\ell-1}]$ should decay rapidly and be smaller than $\mathbb{V}[\mathbf{Q}_\ell]$ and $\mathbb{E}[\mathbf{Q}_\ell]$, respectively, as $\ell \to L$, ensuring that the approximation error decreases with increasing level, which requires $\mathbf{Q}_\ell$ and $\mathbf{Q}_{\ell-1}$ to be sufficiently correlated. We discuss further practical aspects regarding the design of the model hierarchy in more detail in Section 2.4.2.

2.3. Multilevel Markov-Chain Monte Carlo

The multilevel delayed acceptance (MLDA) MCMC algorithm was developed by Lykkegaard et al. (2023) as an extension to the (fixed-length) surrogate transition method of J. S. In Liu (2008), which can operate on model hierarchies with more than two levels. The main functionality of MLDA is shown in Figure 1 for a case with two levels. We use the Python implementation of MLDA by Lykkegaard (2022) with fixed-length subchains and the option of running a number of n_{chains} chains in parallel. In the remainder we also assume that the parameter vectors $\{\boldsymbol{\theta}_\ell\}_{\ell=0}^L$ are comprised of the same model parameters, that is, we do not consider level-dependent or different coarse and fine (or nested) model parameter vectors.

While other MCMC algorithms sample from a single (posterior) distribution as given in Equation 2, MLDA considers a hierarchy of distributions $p_0(\cdot), \ldots, p_\ell(\cdot), \ldots, p_L(\cdot)$ that are computationally cheap approximations of the target density $p(\cdot)$, where each $p_\ell(\cdot)$ may be defined according to Equation 2 corresponding to each model in $\{\mathcal{F}_\ell\}_{\ell=0}^L$. The MLDA algorithm then gets called on the highest-level density $p_L(\cdot)$. By recursively calling the MLDA algorithm on level $\ell - 1$, subchains with length J_ℓ are generated on levels $1 \le \ell \le L$ until level $\ell = 0$ is

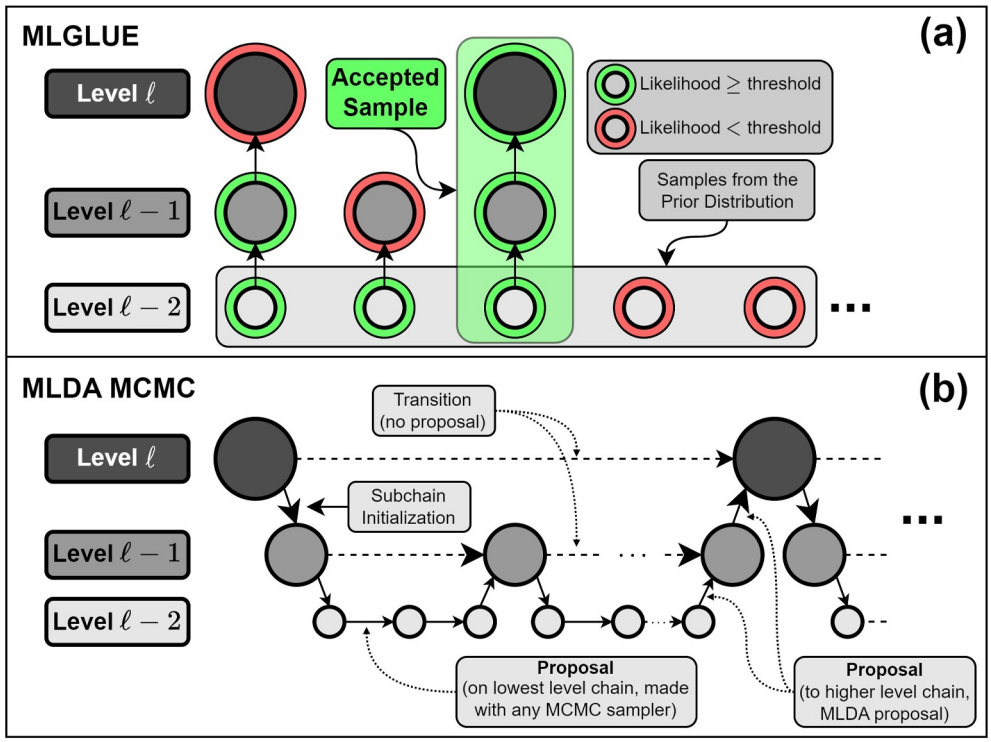

Figure 1. Schematic representation of multilevel sampling strategies for the case of three levels; (a) MLGLUE approach, green rings indicate a likelihood that is above the level-dependent threshold, red rings indicate the contrary; (b) Multilevel Delayed Acceptance MCMC; circles represent the state or current parameter sample.

reached. We note that different subchain lengths may be used on different levels but the analysis here is restricted to the same $J_\ell = J$ on all levels. On the lowest level $\ell = 0$, a conventional MCMC sampler is invoked. The final state of a subchain on level $\ell - 1$, $\boldsymbol{\theta}_{\ell-1}^{J_\ell}$, is finally passed as a proposal to the higher-level chain on level ℓ. Subsequently, only samples from the highest level are considered for inference. A conventional single-level MCMC sampler may be obtained with using MLDA if only the highest-level model is considered. We note that for MLDA the relation between different levels is not formally required to show decaying variance and mean as described in Section 2.2. Aspects of the design of the model (or posterior) hierarchy are discussed in more detail in Section 2.4.2.

To assess convergence of the Markov-chains on the highest level, the Gelman-Rubin statistic $\hat{R}$ is frequently used for multi-chain samplers (Gelman & Rubin, 1992; Lykkegaard et al., 2023). In hydrological applications, a value of $\hat{R} \leq 1.2$ is often deemed sufficient to ensure convergence (e.g., Vrugt, ter Braak, Gupta, & Robinson, 2009; Vrugt, 2016; Zhang et al., 2020). We note, however, that stricter choices for $\hat{R}$ can be found in the literature and that there still exists an ongoing debate about this choice. MCMC (and MLDA) samples from converged chains are naturally correlated and may show dependence on initial samples. Therefore, an initial number of samples is often discarded (burn-in) and remaining samples are thinned (only every $\mathcal{K}$-th sample is considered for subsequent analysis, reducing sample autocorrelation) to obtain approximately independent samples (e.g., Brunetti et al., 2023; Gallagher et al., 2009; Lykkegaard et al., 2023; Vrugt, 2016). The number of approximately independent samples is termed the estimated effective sample size and can be calculated as shown in Geyer (1992, 2011). Because we observed good chain convergence for both examples without discarding any burn-in (except for MCMC sampling for the groundwater flow example, where only the initial sample was discarded; see Text S3 and Text S4 in Supporting Information S1), we obtain effective samples by thinning such that the resulting number of samples is approximately equal to the estimated effective sample size. We denote this set of effective samples by matrix $\mathbf{B}$ with each column representing a single variable and each of the N_b rows representing a single sample.

2.4. Multilevel Generalized Likelihood Uncertainty Estimation

2.4.1. The MLGLUE Algorithm

The Generalized Likelihood Uncertainty Estimation (GLUE) methodology rejects the formal (Bayesian) statistical basis of inference and instead seeks to identify a set of system representations (combinations of model inputs, model structures, model parameters, errors) that are sufficiently consistent with the observations of that system (Beven & Freer, 2001; Vrugt, ter Braak, Gupta, & Robinson, 2009; Beven & Binley, 2014; Mirzaei et al., 2015). GLUE has furthermore been shown to represent a special case of Approximate Bayesian Computation, where summary statistics of simulations and observations are used instead of likelihood functions, and we refer to Nott et al. (2012) and Sadegh and Vrugt (2013) for more details regarding differences and similarities.

The likelihood function in GLUE aggregates all aspects of error and consistency as a generalized fuzzy belief. It serves as a decision threshold to separate behavioral (i.e., good agreement between $\mathbf{Y}$ and $\widetilde{\mathbf{Y}}$) and non-behavioral (i.e., poor agreement between $\mathbf{Y}$ and $\widetilde{\mathbf{Y}}$) simulations. Beven and Binley (1992) and (Beven & Freer, 2001) introduced a number of different functions for this purpose. The following likelihood is frequently used (Vrugt, ter Braak, Gupta, & Robinson, 2009):

$$\tilde{\mathcal{L}}\left(\boldsymbol{\theta}|\widetilde{\mathbf{Y}}\right) := \left(\sigma_r^2\right)^{-W} = \left(\frac{\sum_{i=1}^{k}\left(y_i - \tilde{y}_i\right)^2}{k-2}\right)^{-W}, \tag{4}$$

where W is a shape parameter of the likelihood function defined by the user. Note that for $W = 0$, every simulation will have an equal likelihood and for $W \to \infty$, the emphasis will be placed on a single best simulation while the other solutions are assigned a negligible likelihood.

Parameter and model output uncertainty is estimated in GLUE by running the model with N parameter samples, $\left\{\boldsymbol{\theta}^{(j)}\right\}_{j=1}^{N}$, randomly drawn from the prior distribution and evaluating the likelihood function for each sample. The likelihood threshold may either be defined a-priori (as a certain value above which a model realization is considered behavioral) or may be defined as a percentage based on the set of all likelihood corresponding to the evaluated parameter samples (by setting the threshold to, e.g., the top 10% of the likelihood values) (Beven & Binley, 1992; Beven & Freer, 2001; Vrugt, ter Braak, Gupta, & Robinson, 2009). Using only behavioral solutions (cumulative) probability distributions of model outputs are generated, from which uncertainty estimates are finally computed. Behavioral parameter samples are used to estimate the posterior distribution of model parameters.

MLGLUE is generally similar to MLDA (or MLMCMC) as shown in Figure 1. As with MLDA, a parameter sample $\boldsymbol{\theta}^{(j)}$ is only finally stored if it is accepted on the highest level. While MLDA makes use of an acceptance probability on all levels (as it is typical in MCMC algorithms), MLGLUE uses a level-dependent likelihood threshold on all levels to distinguish between samples being accepted (i.e., behavioral solutions) and samples being discarded (i.e., non-behavioral solutions).

MLGLUE requires that likelihood thresholds are available for every level prior to sampling, although pre-defined likelihood thresholds can optionally be used. MLGLUE considers a simple Monte Carlo estimator to compute likelihood thresholds, where the same set of parameter samples is evaluated on each level using the likelihood function. The number of those parameter samples, N_t, should be substantially smaller than the overall number of samples being evaluated with MLGLUE, N. We denote the set of corresponding likelihoods on a single level by $\left\{\tilde{\mathcal{L}}^{(i,\ell)}\right\}_{i=1}^{N_t}$ and the combined set for all levels by $\left\{\left\{\tilde{\mathcal{L}}^{(i,\ell)}\right\}_{i=1}^{N_t}\right\}_{\ell=0}^{L}$. The likelihood thresholds on the different levels are then obtained by computing a pre-defined percentile estimate from the level-dependent likelihood samples (e.g., for a threshold corresponding to the top 5% the 95%-percentile is computed). We denote the set of likelihood thresholds on each level by $\left\{\tilde{\mathcal{L}}_{T,\ell}\right\}_{\ell=0}^{L}$. We refer to this step as *tuning*. For two example problems we discuss the choice of N_t (see Section 4). We also note that the tuning phase can be omitted entirely if level-dependent likelihood thresholds can be pre-defined, for example, from expert knowledge.

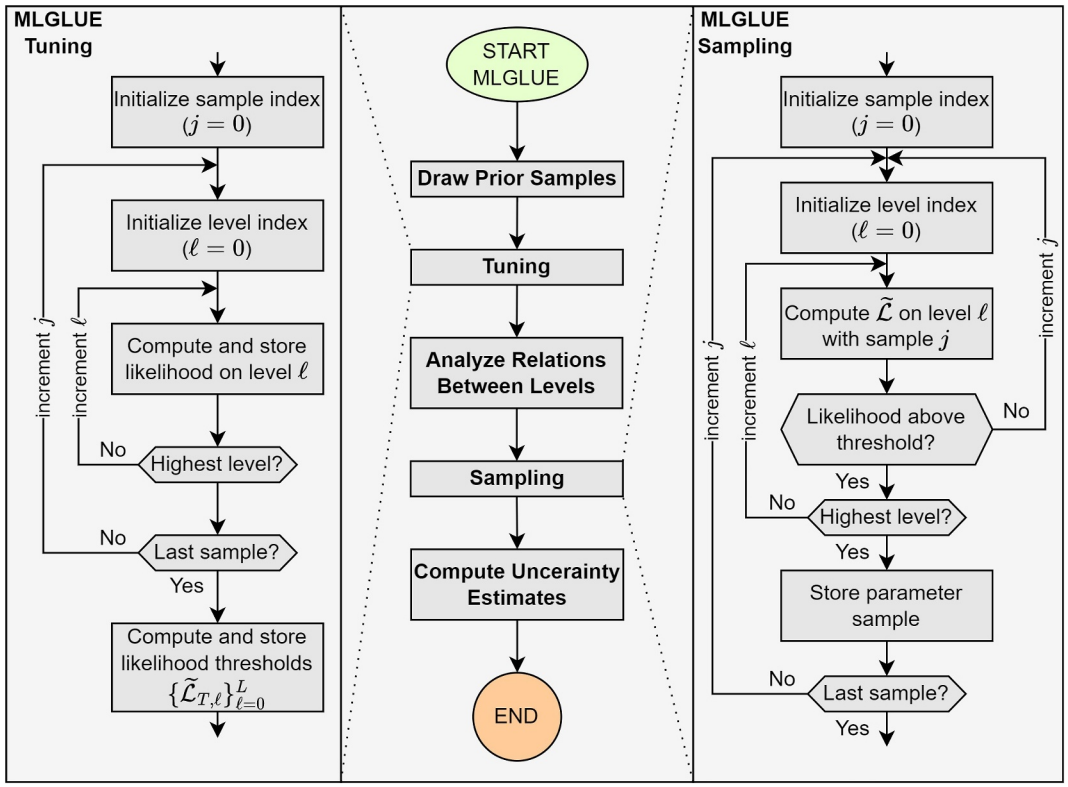

Figure 2. Schematic representation of the multilevel Generalized Likelihood Uncertainty Estimation algorithm; tuning refers to the (optional) Monte Carlo estimation of likelihood thresholds, sampling refers to the repeated evaluation of parameter samples (see the description of algorithm steps).

From the set of likelihood values on each level, $\left\{\{\tilde{\mathcal{L}}^{(i,\ell)}\}_{i=1}^{N_t}\right\}_{\ell=0}^{L}$, sample estimates of $\mathbb{V}\left[\tilde{\mathcal{L}}_\ell\right]$, $\mathbb{E}\left[\tilde{\mathcal{L}}_\ell\right]$, $\mathbb{V}\left[\tilde{\mathcal{L}}_\ell - \tilde{\mathcal{L}}_{\ell-1}\right]$, and $\mathbb{E}\left[\tilde{\mathcal{L}}_\ell - \tilde{\mathcal{L}}_{\ell-1}\right]$ for $\ell = 0,\ldots,L$ are computed to analyze the relation between levels regarding the likelihood. This is equivalent to setting $\mathbf{Q}_\ell = \tilde{\mathcal{L}}_\ell$, bridging the gap between MLMC and MLGLUE in this context (see Section 2.2).

Afterward, *sampling* is started and parameter samples $\boldsymbol{\theta}^{(j)}$ are initially evaluated with the model on the coarsest level, $\ell = 0$. If the corresponding likelihood is greater or equal to the level-dependent threshold, the sample is passed to the next higher level and is evaluated again. This process is repeated until the highest level is reached and the sample is finally considered behavioral or non-behavioral. If the likelihood is smaller than the level-dependent threshold on any level, the sample is immediately regarded as non-behavioral and rejected. Therefore, samples with low likelihoods are already disregarded on lower levels, leading to substantial computational savings. In the Supporting Information, the reasoning for using level-dependent likelihood thresholds as well as the structure of the algorithm is clarified in more detail. The MLGLUE algorithm is presented in algorithm 1 with tuning excluded and schematically shown in Figure 2.

Algorithm 1 Multilevel Generalized Likelihood Uncertainty Estimation

Draw a sample $\mathbf{\Theta}_0$ of N points from the (typically uniform) prior distribution $p_p(\boldsymbol{\theta})$
for $j = 0, \ldots, N$ **do**
  for $\ell = 0, \ldots, L$ **do**
    Compute the (log-) likelihood $\widetilde{\mathcal{L}}^{(j,\ell)} = \widetilde{\mathcal{L}}(\boldsymbol{\theta}^{(j)}|\widetilde{\mathbf{Y}})$ with sample $\boldsymbol{\theta}^{(j)}$ from $\mathbf{\Theta}_0$ and with the model on level ℓ
    if $\ell = L$ *and* $\widetilde{\mathcal{L}}^{(j,\ell)} \geq \widetilde{\mathcal{L}}_{T,\ell}$ **then**
      Store $\boldsymbol{\theta}^{(j)}$ in matrix $\mathbf{B}$, store the corresponding simulation results $\mathbf{Y}$ in $\mathbf{S}$, increment $j \leftarrow j+1$, and break the loop over the levels
    if $\widetilde{\mathcal{L}}^{(j,\ell)} \geq \widetilde{\mathcal{L}}_{T,\ell}$ **then**
      Increment $\ell \leftarrow \ell + 1$, continuing the loop over the levels for sample $\boldsymbol{\theta}^{(j)}$
    if $\widetilde{\mathcal{L}}^{(j,\ell)} < \widetilde{\mathcal{L}}_{T,\ell}$ **then**
      Increment $j \leftarrow j+1$, breaking the loop over the levels
for $\mathbf{b}^{(i)},\ i = 1, \ldots, N_b$ *in* $\mathbf{B}$ **do**
  Normalize the corresponding likelihood as $\widetilde{\mathcal{L}}'(\mathbf{b}^{(i)}|\widetilde{\mathbf{Y}})$ such that $\sum_{i=1}^{N_b} \widetilde{\mathcal{L}}'(\mathbf{b}^{(i)}|\widetilde{\mathbf{Y}}) = 1$, e.g., via $\widetilde{\mathcal{L}}'(\mathbf{b}^{(i)}|\widetilde{\mathbf{Y}}) = \widetilde{\mathcal{L}}(\mathbf{b}^{(i)}|\widetilde{\mathbf{Y}}) / \sum_{i'=1}^{N_b} \widetilde{\mathcal{L}}(\mathbf{b}^{(i')}|\widetilde{\mathbf{Y}})$
for $\mathbf{Y}^{(i)}, i = 1, \ldots, N_b$ *in* $\mathbf{S}$ **do**
  Assign the corresponding weight $\widetilde{\mathcal{L}}'(\mathbf{b}^{(i)}|\widetilde{\mathbf{Y}})$
Sort the $\mathbf{Y}^{(i)}, i = 1, \ldots, N_b$ increasingly according to their weights and create uncertainty estimates from the empirical distribution obtained this way (e.g., as quantiles)

2.4.2. Designing the Model Hierarchy

During multilevel inversion, no explicit approach exists yet to optimally pre-define the number of levels or the difference in resolution between the levels. In their example applications of MLMCMC and MLDA, Dodwell et al. (2019) and Lykkegaard et al. (2023) arbitrarily pre-define the coarsening as well as the number of levels considered but give some analysis of the effect regarding the number of levels. In similar examples to our subsequently considered benchmark example of groundwater flow (see Section 3.2), Cliffe et al. (2011) consider 5 levels for MLMC, Dodwell et al. (2019) consider up to 5 levels for MLMCMC, Lykkegaard and Dodwell (2022) consider 2 levels with MLDA, and Lykkegaard et al. (2023) consider 3 levels with MLDA. In the following we give guidelines on how to design a hierarchy of models and also show directions for further research.

A geometric series of resolutions for the computational grids (in space or time or both) is often most suitable in the context of MLMC (also see Section 2.2), where the factor of grid refinement (when going from ℓ to $\ell + 1$) or coarsening (when going from ℓ to $\ell - 1$) between subsequent levels is constant (Giles, 2015). We also adopt this method in this study.

In MLGLUE, a parameter sample that is accepted on the highest level with the highest resolution model is evaluated on all lower levels with lower-resolution models before. Therefore, the number of levels in the model hierarchy should be as low as possible and the coarsening factor as large as possible to obtain a high computational efficiency of the multilevel hierarchy. Those aspects are then restricted by the quality of the coarsest-level model being sufficiently high, by the required resolution on the highest level, and by the requirement for

sufficiently high correlation between subsequent levels. Those criteria can be analyzed via the relations between levels regarding $\left\{\left\{\tilde{\mathcal{L}}^{(i,\ell)}\right\}_{i=1}^{N_t}\right\}_{\ell=0}^{L}$ (see also Section 2.2).

In this study we consider cases where a target resolution is given for the highest-level model and lower-resolution models are obtained by subsequent coarsening. Afterward, in practical applications, the coarsest possible model resolution for the lowest level should be determined approximately. With the highest and lowest resolutions specified, the number of levels is determined through finding an appropriate coarsening factor that results in sufficiently high correlation between the levels (see Section 2.2). We investigate and discuss those aspects in more detail for the results of the test problems in Section 4.

An alternative strategy for the design of the hierarchy is presented in Vidal-Codina et al. (2015) and Giles (2015) for non-geometric MLMC. It relies on generating a set of test models for a large number of levels, $\{\mathcal{F}_\ell\}_{\ell=0}^{L}$, and then selecting a subset of levels that satisfy some conditions on the relation between levels, similar to the conditions used in the tuning phase of MLGLUE. In any case, this approach requires additional computational resources to optimize the hierarchy, being associated with a large number of degrees of freedom in the design. This strategy can potentially be applied for MLGLUE as well but is not the focus of the current study. This approach is left open for further research as it has become apparent in this study that a geometric series generally serves as a robust starting point under various conditions.

Besides improving the design of the model hierarchy itself, an error model can be employed to increase efficiency. The error model introduced by Lykkegaard et al. (2023) accounts for differences in the model outputs between different levels using a bias term. Lower-level approximations are improved by estimating bias using model evaluations on different levels for the same parameter sets. In order to focus on the effects of model hierarchy design without any disturbance, no error model is used with MLDA in this study. We note, however, that such approaches can increase efficiency, potentially also of MLGLUE, and therefore pose an interesting direction for further research.

2.4.3. Parallelization

Like the conventional formulation of GLUE, MLGLUE can be parallelized in a straightforward manner to accelerate computation. We utilize `Ray v2.2.0` (Moritz et al., 2018) for parallelization with its `multiprocessing.Pool` API. Parallelization is achieved by using `Ray Actors` instead of local processes. For MLGLUE and GLUE, the function (or task) being parallelized corresponds to the evaluation of a single parameter sample, starting on $\ell = 0$ and including all subsequent model runs on higher levels (see the MLGLUE algorithm). MLGLUE considers running the hierarchy of models $\{\mathcal{F}_0(\theta_i),\ldots,\mathcal{F}_L(\theta_i)\}$ for a single parameter sample θ_i as one iteration. As the parallelization is implemented on the level of these iterations, it allows for evaluating multiple parameter samples in parallel. For the case of using MLGLUE with a single level (i.e., conventional GLUE), the iteration reduces to running the target model, $\{\mathcal{F}_L(\theta_i)\}$, for multiple parameter samples in parallel.

For MLDA and MCMC, however, the parallelization is implemented on the level of individual chains. The MLDA implementation (`tinyDA v0.9.8`, Lykkegaard (2022)) also relies on `Ray Actors` for parallelization, implemented via remote functions. Therefore, the underlying mechanism for parallelization are identical for GLUE, MLGLUE, MCMC, and MLDA. Still, differences regarding the increase in computational efficiency may be observed when comparing sequential and parallelized algorithm run times for GLUE and MLGLUE with those for MCMC and MLDA. This is due to (a) the differences in the implementation of parallelization and (b) the differences in the algorithms themselves.

2.5. Analysis of Posterior Convergence

In order to compare the different methods of statistical inference in our study, we assess the convergence to a stable posterior distribution and monitor the number of model evaluations and the computational time required for convergence. We introduce a simple way of assessing convergence that works for any method that returns a - possibly ordered - sequence of values in $\mathbb{R}^n$, which are assumed here to be samples from a probability distribution. In the context of MCMC, the introduced methodology is not to be mistaken for a way of assessing the convergence of (Markov-) chains.

The central concept of the methodology is to analyze the ratio of mean and variance of the (marginal) posterior distribution, estimated from a subset of the set of all available samples, to mean and variance estimated from the set of all available samples (N_b samples in **B**). As the subset gets larger, and eventually becomes equal to **B**, this quantity allows for the analysis of convergence behavior. The subset is taken to be the first s samples from the posterior samples returned by a method of statistical inference. We denote the estimate of the mean or any higher-order moment around the mean by μ_m^s, where s represents the size of the subset and m represents the moment order. We define the relative deviation $\mathcal{D}_m^s$ of moment m, computed with a subset of size s, from the globally estimated moment as

$$\mathcal{D}_m^s := \frac{\mu_m^s}{\mu_m^{N_b}} - 1 \tag{5}$$

By definition, $\mathcal{D}_m^s \to 0$ as $s \to N_b$; however, the analysis regarding *how* and *how quickly* $\mathcal{D}_m^s$ tends toward zero as s increases allows for the analysis of convergence behavior. We assume convergence at $s = s_c$ if $-0.05 \leq \mathcal{D}_m^s \leq 0.05$ for all $s \geq s_c$. Assuming that the samples are obtained uniformly over time during inference or computation enables the assessment of convergence against computation time instead of sample size.

3. Test Problems

The test problems discussed in Sections 3.1 and 3.2 are used to illustrate the differences between the methods of statistical inference (MLGLUE, GLUE, MLDA, MCMC) regarding obtained posterior distributions, uncertainty estimates for model outputs, and computational efficiency. An identical number of prior parameter samples is used for GLUE and MLGLUE. The MCMC sampler takes the same number of steps across all chains. Similarly, the MLDA sampler takes the same number of steps on the lowest level across all chains; due to the subsampling rate, a smaller number of steps is then taken across all chains on the higher levels. This ensures comparability as all methods evaluate the same number of parameter samples on the lowest level (i.e., the only level for GLUE and MCMC). For GLUE and MLGLUE, an informal likelihood function (Equation 4) is used for each problem. MCMC and MLDA are used with a formal likelihood function (Equation 3). We analyze the tuning phase separately for both examples using two threshold settings (selecting the top 2 % and 7 %) for different N_t.

All methods of inference are implemented in the Python programming language. The `tinyDA v0.9.8` (Lykkegaard, 2022) package is used for MLDA and MCMC sampling, using `Ray v2.2.0` (Moritz et al., 2018) for parallelization. We apply a DREAM(Z)-sampler, which is similar to the DREAM(ZS)-sampler but it uses fully independent chains and no snooker updates (Lykkegaard, 2022; Vrugt, 2016). Samplers from the DREAM family have been developed with a focus on hydrological sciences (Vrugt, 2016) and have been frequently used for Bayesian inversion of, i. a., rainfall-runoff models (Vrugt et al., 2008; Vrugt, ter Braak, Diks, et al., 2009; Vrugt, ter Braak, Gupta, & Robinson, 2009; Vrugt, 2016) and groundwater flow models (Laloy & Vrugt, 2012; Laloy et al., 2013; J. Kennedy et al., 2016; Thiros et al., 2022).

`ArviZ v0.12.1` (Kumar et al., 2019) is used for the analysis of MLDA and MCMC results regarding chain convergence and effective sample size (see Section 2.3); in `tinyDA`, the initial sample is returned additionally to the N drawn samples. MLGLUE is implemented as a Python package and also enabled for parallel computing with `Ray v2.2.0` (Moritz et al., 2018). We note that we subsequently refer to the processed posterior samples from MCMC and MLDA (i.e., after burn-in and thinning, see Section 2.3) as effective samples. The same term is also used for unprocessed GLUE and MLGLUE posterior samples, which are independent without further processing. Therefore, the results of both groups (GLUE and MLGLUE and MCMC and MLDA) are compared on an equal basis using (approximately) independent samples.

3.1. Rainfall-Runoff Modeling

The first case study considers rainfall-runoff modeling using the conceptual model HYMOD (Boyle, 2001), which is schematically shown in Figure 3. The model has five parameters (explained in Figure 3), takes time series of precipitation, $P(t)\ \left[LT^{-1}\right]$, and potential evaporation, $PET(t)\ \left[LT^{-1}\right]$, as inputs and outputs a time series of discharge, $Q(t)\ \left[LT^{-1}\right]$. This model has been frequently and similarly used in the context of statistical inference, uncertainty analysis, and sensitivity analysis (Boyle, 2001; Wagener et al., 2001; Vrugt et al., 2003, 2005; Blasone et al., 2008; Vrugt et al., 2008; Vrugt, ter Braak, Gupta, & Robinson, 2009; Herman et al., 2013).

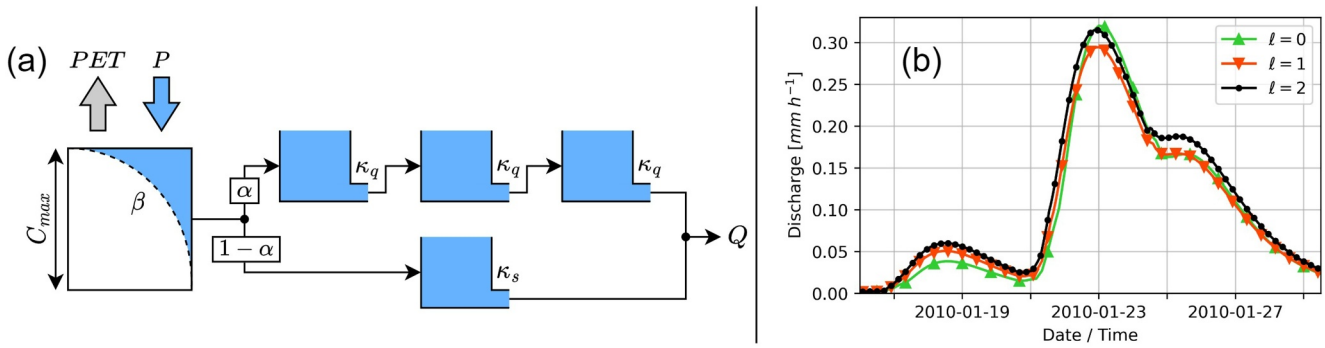

Figure 3. (a) Schematic representation of the HYMOD model (Vrugt, ter Braak, Gupta, & Robinson, 2009); $C_{\max}$ $[L]$ is the maximum catchment storage, β $[-]$ is the spatial variability of soil moisture storage, α $[-]$ is the distribution factor between reservoirs, and κ_q $\left[T^{-1}\right]$ and κ_s $\left[T^{-1}\right]$ are discharge coefficients of the quick-flow and slow-flow reservoirs, respectively; (b) discharge simulated by models on all three levels for two consecutive events, only every fifth time step is marked.

We apply the model to data from the Leaf River catchment near Collins, Mississippi, USA, which has been studied with the same model multiple times before (Wagener et al., 2001; Vrugt et al., 2003, 2005; Blasone et al., 2008; Vrugt et al., 2008; Vrugt, ter Braak, Gupta, & Robinson, 2009). We refer the reader to the aforementioned references for detailed descriptions of the HYMOD model and the study area. Contrary to other studies we consider time series with hourly instead of daily resolution (Gauch et al., 2020, 2021) and use the hydrological year of data from 2009 to 10-01 to 2010-09-30. The first 25 days are considered a warm-up period, being simulated but not used to calculate likelihoods.

The model is implemented in the Python programming language following Knoben et al. (2019); Trotter et al. (2022); Trotter and Knoben (2022) and the differential equations are solved using the explicit Euler method (e.g., Braun, 1993). The highest-level model uses an hourly time step equal to the data time steps. Two additional lower-level models are considered with time steps of two and 4 hours, respectively (i.e., time step lengths are doubled when going to the next lower level). On levels $\ell = 0$ and $\ell = 1$, resulting time series of discharge are linearly interpolated to the time steps of the model on level $\ell = 2$ to allow for the calculation of likelihoods with the original data time steps.

The prior distribution $p_p(\boldsymbol{\theta})$ is chosen to be a uniform distribution over the parameters $\boldsymbol{\theta} = [C_{\max}, \beta, \alpha, \kappa_s, \kappa_q]^T$ with lower bounds $\boldsymbol{\theta}_l = [1.0, 0.1, 0.0, 0.0, 0.0]$ and upper bounds $\boldsymbol{\theta}_u = [1000.0, 2.0, 1.0, 0.1, 0.5]$. Length units of $[mm]$ and time units of $[h]$ are used throughout the model and for all data sets. A total number of $N_t + N = 5{,}000 + 995{,}000 = 1{,}000{,}000$ samples are drawn from $p_p(\boldsymbol{\theta})$ for GLUE and MLGLUE, where $N_t = 5{,}000$ samples are used to estimate the level-dependent likelihood thresholds (see Section 2.4) and to analyze the relations between the levels (see Section 2.2) in MLGLUE. With MCMC and MLDA samplers, also $1{,}000{,}000$ steps are taken on the lowest level (i.e., the only level in MCMC). The choice of N_t is discussed in Section 4.1. A constant variance equal to the constant additive Gaussian noise variance $\left(\sigma^2 = 1.0\ \mathrm{mm}^2\mathrm{h}^{-2}\right)$ is used for the Gaussian likelihood (see Equation 3); for the likelihood used in MLGLUE and GLUE (see Equation 4) $W = 1$ is used. The likelihood thresholds are estimated to correspond to the best 2% of simulations. For MLDA, the sub-sampling rate is set to 5. MLDA and MCMC are run with 10 independent chains. All methods are run on 32 dual-core CPUs (64 total threads).

3.2. Groundwater Flow

The second example considers steady-state two-dimensional groundwater flow in an aquifer with inhomogeneous horizontal hydraulic conductivity, Dirichlet-type (fixed potentials), Neumann-type (no-flow conditions, recharge), Robin-type (river), and nodal sink type (wells) boundary conditions.

$$\frac{\partial}{\partial x}\left(K_{xx}\frac{\partial h}{\partial x}\right) + \frac{\partial}{\partial y}\left(K_{yy}\frac{\partial h}{\partial y}\right) + R = 0 \tag{6}$$

$$h = h_c \quad \forall\, y \in \partial\Omega,\ x = 0\ m \tag{7}$$

$$\frac{\partial h}{\partial y} = 0 \quad \forall x \in \partial\Omega,\ y \in \{0\ m,\ 5{,}000\ m\} \tag{8}$$

$$\frac{\partial h}{\partial x} = 0 \quad \forall y \in \partial\Omega, x = 10{,}000\ m \tag{9}$$

$$f_{riv} = c_{riv}\Delta h \quad \forall\, 0\ m \le x \le 10{,}000\ m,\ y = 1{,}000\ m, \tag{10}$$

Where $K\left[LT^{-1}\right]$ is the hydraulic conductivity field, $h\ [L]$ is the hydraulic head field, $R\left[LT^{-1}\right]$ is the recharge flux, $f_{riv}\left[LT^{-1}\right]$ is river inflow, and $c_{riv}\left[T^{-1}\right]$ is riverbed conductance. The model is set up with the finite-differences code `MODFLOW-NWT` and the reader is referred to Harbaugh (2005) and Niswonger et al. (2011) for a detailed description of the model and boundary condition implementations.

The reference model is discretized as a regular structured grid with a cell-size of 25 $m \times 25\ m$, having 200 rows and 400 columns. The aquifer bottom is horizontal at 10.0 m above the reference datum; the aquifer top represents a tilted plane falling linearly from 55.0 m on the left side of the domain to 45.0 m above the reference datum on the right side of the domain. A river crosses the domain along a single row, having a constant water level at 6.0 m below the aquifer top and a river bottom at 9.0 m below the aquifer top. 5 wells are placed in the model domain with a total extraction rate of 700 md^{-1}. Spatially uniform recharge is applied with a rate of $2 \cdot 10^{-5}\ \mathrm{md}^{-1}$. A constant head of 45.0 m above the reference datum is assigned to the leftmost column of cells. 12 observation points as well as 1 prediction point are placed in the domain.

The hydraulic conductivity in every cell is obtained in the reference model using a regular grid of pilot points (e.g., Doherty, 2003), linearly spaced (5 along columns, 10 along rows) starting on the domain boundaries. Reference values of pilot point $\log_{10}$-hydraulic conductivities are obtained by sampling from a log-normal distribution with $\mu = 0.3$ and $\sigma = 0.7$. Gaussian process regression (GPR), as implemented in `scikit-learn v1.2.0` (Pedregosa et al., 2011), is used to interpolate $\log_{10}$-hydraulic conductivities at cell centers of the reference model with a radial basis function kernel with a fixed length scale of 600 m. The model domain and its main characteristics are shown in Figure 4 for the models on levels $\ell = 0$ and $\ell = 3$.

The reference model is also the highest-level model. Besides this model, three lower-level models are considered, resulting in $\ell = 0,1,2,3$. Lower-level models are obtained via grid coarsening, where cell sizes are doubled going from ℓ to $\ell - 1$. Lower-level hydraulic conductivity values at each cell are obtained by using the geometric mean of corresponding higher-level cells.

Besides the 50 pilot point parameters, the GPR length scale is considered a model parameter as well; $\boldsymbol{\theta} = [\theta_{1,PP}, \ldots, \theta_{50,PP}, \theta_{51,GPR}]^T$. We denote the parameter-to-observable map (i.e., Equations 6–10) by $\mathcal{F}(\boldsymbol{\theta})$. Adding Gaussian random noise to the observations then leads to $\widetilde{\mathbf{Y}} = \mathcal{F}(\boldsymbol{\theta}) + \varepsilon$, $\varepsilon \sim \mathcal{N}(\mu = 0, \sigma = 0.5)$.

As a prior distribution $p_p(\boldsymbol{\theta})$, a uniform distribution is chosen with lower bounds $\boldsymbol{\theta}_l = \left[1 \cdot 10^{-2}, \ldots, 1 \cdot 10^{-2}, 5 \cdot 10^{2}\right]$ and upper bounds $\boldsymbol{\theta}_u = \left[1 \cdot 10^{1}, \ldots, 1 \cdot 10^{1}, 1 \cdot 10^{3}\right]$. A total number of $N_t + N = 2{,}000 + 998{,}000 = 1{,}000{,}000$ samples are drawn from $p_p(\boldsymbol{\theta})$ for GLUE and MLGLUE, where $N_t = 2{,}000$ samples are used to estimate the level-dependent likelihood thresholds (see Section 2.4) and to analyze the relations between the levels (see Section 2.2) in MLGLUE. With MCMC and MLDA samplers, also 1,000,000 steps are taken on the lowest level (i.e., the only level in MCMC). The choice of N_t is discussed in Section 4.2. A constant variance equal to the constant additive Gaussian noise variance $\left(\sigma^2 = 1.0\ m^2\right)$ is used for the Gaussian likelihood (see Equation 3); for informal likelihoods (see Equation 4) $W = 1$ is used. The likelihood thresholds are estimated to correspond to the best 7% of all simulations. The threshold is greater compared to the rainfall-runoff modeling example as the dimension of the parameter space is substantially higher in this example. As the same number of samples is used, they are less densely distributed, leading to smaller acceptance probability in MLGLUE and GLUE for smaller thresholds. To counteract this effect, the threshold is increased, leading to a higher acceptance probability at the cost of obtaining posterior samples associated with lower likelihood. For MLDA, the sub-sampling rate is set to 5. All methods are run on 32 dual-core CPUs (64 total threads).

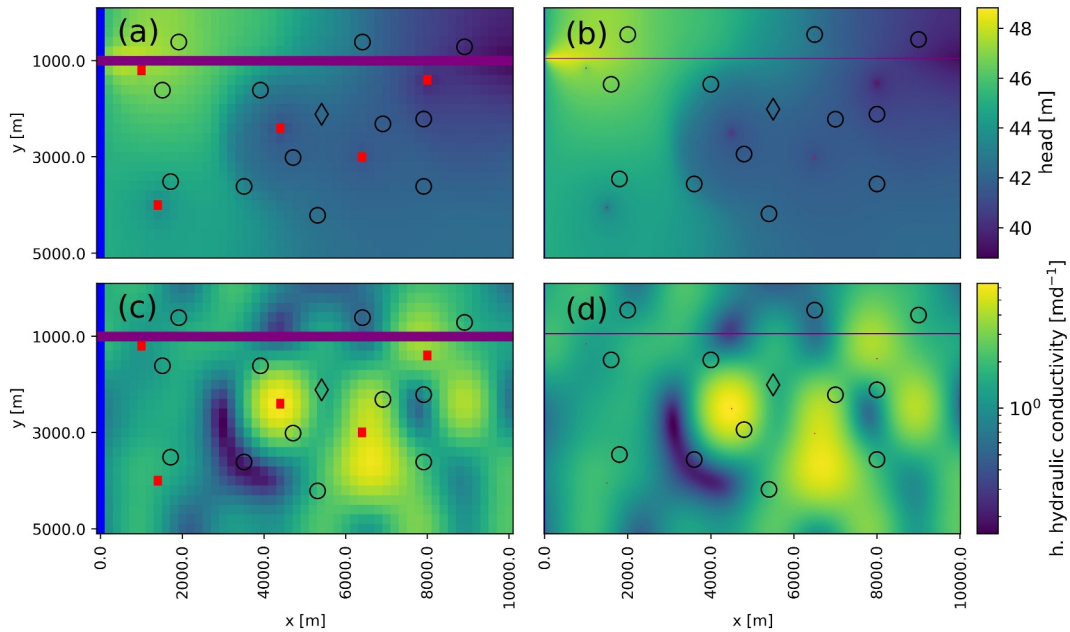

Figure 4. Groundwater flow model domain; head contours obtained with true parameters on level $\ell = 0$ (a) and on level $\ell = 3$ (b); horizontal hydraulic conductivity field on level $\ell = 0$ (c) and on level $\ell = 3$ (d); specific characteristics are: constant head cells (blue), river cells (purple), wells (red), observation points (circles), prediction point (diamond).

4. Results

For the two examples considered, we now present results of inversion with the methodologies of MLGLUE, GLUE, MLDA, and MCMC. We analyze how models on different levels are related and how the results obtained with a multilevel approach differ from the conventional approach using a single model. Differences between MLGLUE and GLUE on one hand, and between MLDA and MCMC on the other hand, are discussed regarding posterior parameter and model output distributions, as well as computational efficiency.

4.1. Rainfall-Runoff Modeling

In this example, likelihood thresholds are not pre-defined but are estimated during the tuning phase of the MLGLUE algorithm. For two threshold settings the estimated likelihood thresholds are shown in Figure S1 in Supporting Information S1 for different numbers of tuning samples, N_t. For the smaller threshold setting of 2 % (i.e., higher likelihood threshold values), likelihood thresholds stabilize at $N_t = 5{,}000$ after showing initial oscillations. For the larger threshold setting of 7 %, likelihood values tend to decrease successively, stabilizing at $N_t = 2{,}000$. The ratio of the likelihood thresholds on the three levels, however, remains approximately equal for both threshold settings, even for smaller N_t. From this analysis and with the threshold setting being 2 %, we set $N_t = 5{,}000$ in this example.

The relations between the three levels are shown in Figure S2 in Supporting Information S1. $\mathbb{V}\left[\tilde{\mathcal{L}}_\ell\right]$ and $\mathbb{E}\left[\tilde{\mathcal{L}}_\ell\right]$ are approximately constant across all levels and $\mathbb{V}\left[\tilde{\mathcal{L}}_\ell - \tilde{\mathcal{L}}_{\ell-1}\right]$ and $\mathbb{E}\left[\tilde{\mathcal{L}}_\ell - \tilde{\mathcal{L}}_{\ell-1}\right]$ decay across all levels. The correlation coefficients are 0.9102 between levels $\ell = 0$ and $\ell = 1$ and 0.9958 between levels $\ell = 1$ and $\ell = 2$ and therefore increase with increasing level index. Consequently, the approximation error of the likelihoods decreases as $\ell \to L$.

The sampling efficiencies of all methods are shown in Figure 5; detailed results of MLDA and MCMC chain convergence (Gelman-Rubin statistic), the recovery of effective samples, and proposal hyperparameters are described in Text S3 in Supporting Information S1. With MLGLUE the overall computation time is reduced by ≈58 % and the number of effective samples per minute is ≈74 % higher compared to GLUE. With MLDA the overall computation time is reduced by ≈18 % and the number of effective samples per minute is ≈39 % lower

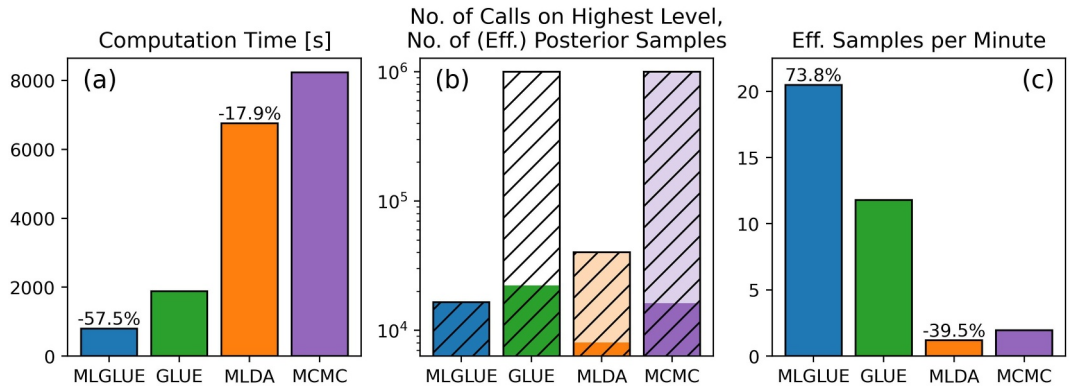

Figure 5. Sampling efficiencies for the rainfall-runoff modeling example; (a) computation times with percentual reductions with multilevel methods compared to conventional methods; (b) No. of model calls on the highest level (dashed), No. of posterior samples (light colors), No. of effective posterior samples (dark colors); (c) No. of effective posterior samples per minute with percentual increase compared to conventional methods.

compared to conventional MCMC. While the number of effective samples per minute is lower for MLDA compared to MCMC, the ratio between the number of effective samples to the total number of posterior samples on the highest level is higher, indicating lower sample autocorrelation before thinning. More detailed analyses of MLDA and MCMC results are presented in Supporting Information S1.

We note that the low correlation between levels $\ell = 0$ and $\ell = 1$ in this example potentially reduces the efficiency of MLDA, leading to reduced acceptance rates on level $\ell = 1$. Such effects can be observed in MLGLUE as well, although they are not as pronounced in this example. An error model Lykkegaard et al. (2023) can potentially be used to increase efficiency for MLDA, and potentially MLGLUE as well, which we discuss in Section 2.4.2.

The results of convergence analysis (see Section 2.5) are shown in Figure 6. Results are obtained by splitting the original sets of effective parameter samples into 200 consecutive subsets, independently of the method of inference. Multilevel approaches (MLGLUE and MLDA) generally converge after a shorter computation time compared to their conventional counterparts (GLUE and MCMC), respectively. The deviation of mean and variance, however, is larger for small sample sizes with MLGLUE compared to GLUE with the set of prior

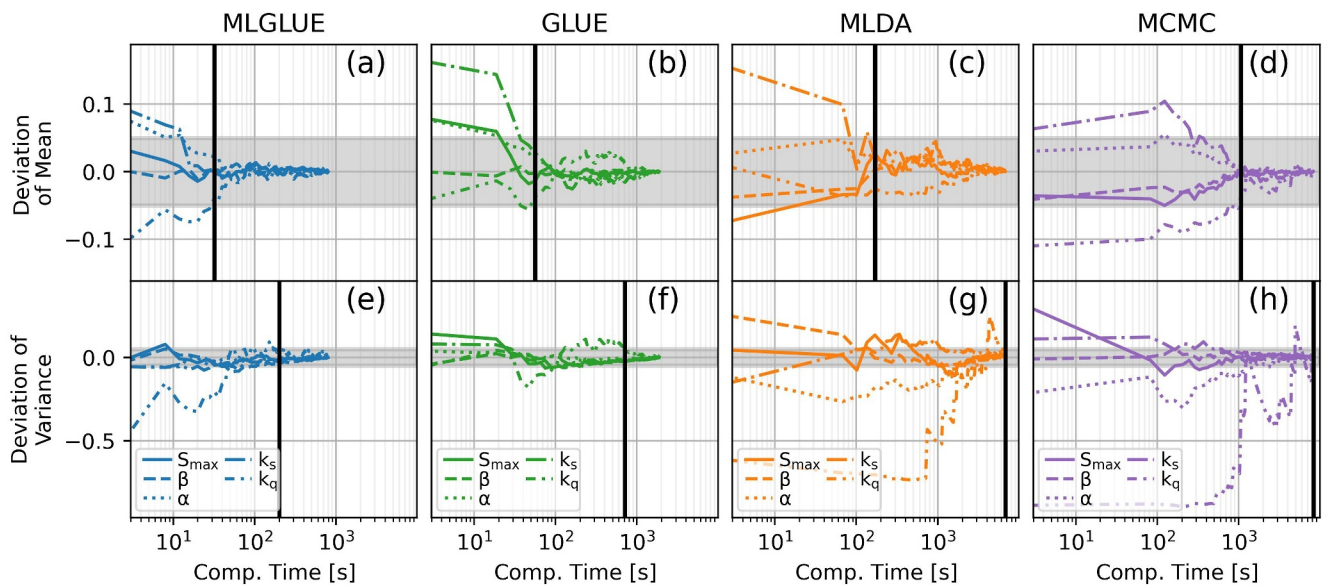

Figure 6. Convergence analysis for the rainfall-runoff modeling example (Equation 5); for the different methods of inference (a)–(d) shows the deviation of the mean and (e)–(h) shows the deviation of the variance; gray regions represent the region where convergence is achieved; black vertical lines represent the computational time at which convergence is achieved for all parameters.

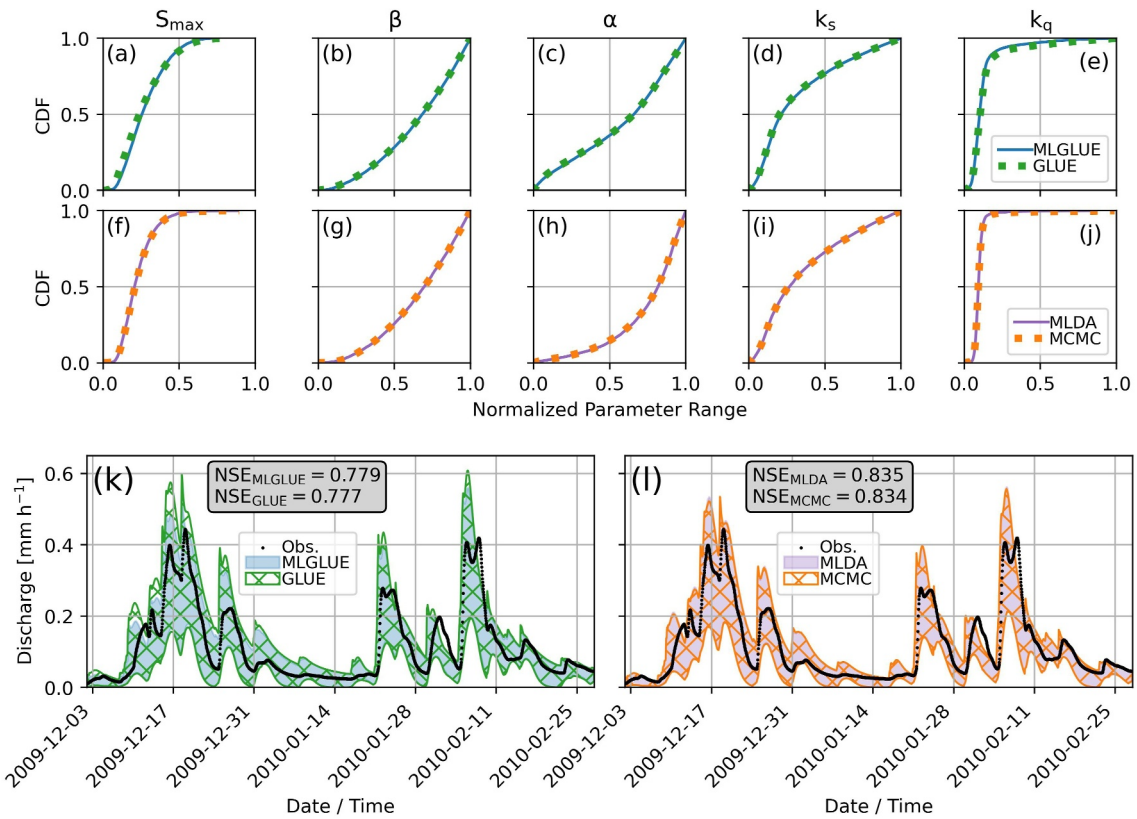

Figure 7. CDFs of model parameters for the rainfall-runoff modeling example for MLGLUE and GLUE (a to e), for MLDA and MCMC (f to j) and 99% − 1% uncertainty estimates around the median value for MLGLUE and GLUE (k) and for MLDA and MCMC (l).

samples being equal for MLGLUE and GLUE. Compared to MLDA, MCMC results show a larger deviation of the mean even for larger sample sizes.

Estimated cumulative distribution functions (CDFs) of the parameter posteriors are shown in Figures 7(a)–7(d). Posteriors obtained with multilevel methods (MLGLUE and MLDA) are virtually identical to their conventional counterparts (GLUE and MCMC). Uncertainty estimates of MLGLUE are different from those of GLUE in that they have smaller range, which is particularly visible at peak flow events (e.g., around 2009-12-17). Uncertainty estimates from MLDA and MCMC are virtually identical, also at peak flow events. The Nash-Sutcliffe model efficiency (Nash & Sutcliffe, 1970), computed with the median of the simulations, is virtually identical for MLGLUE and GLUE and slightly higher for MLDA compared to MCMC.

4.2. Groundwater Flow

In this example, likelihood thresholds are not pre-defined but are estimated during the tuning phase of the MLGLUE algorithm. For two threshold settings the estimated likelihood thresholds are shown in Figure S3 in Supporting Information S1 for different numbers of tuning samples, N_t. For the smaller threshold setting (2 %, corresponding to a higher likelihood threshold), the likelihood thresholds on all levels generally increase as N_t increases and stabilize at $N_t = 5{,}000$. For the setting with a larger threshold setting (7 %), the likelihood values also increase as N_t increases but remain at smaller values compared to the smaller threshold setting and stabilize at $N_t = 2{,}000$. The ratio of the likelihood thresholds on the four levels remains approximately equal only for the larger threshold setting, even for smaller N_t. See Section 4.1 for a more detailed discussion on the tuning phase. With the threshold setting being set to 7 % in this example, we set $N_t = 2{,}000$ here to keep N_t as small as possible to reduce overall computational cost but ensure reasonably stable likelihood threshold estimates.

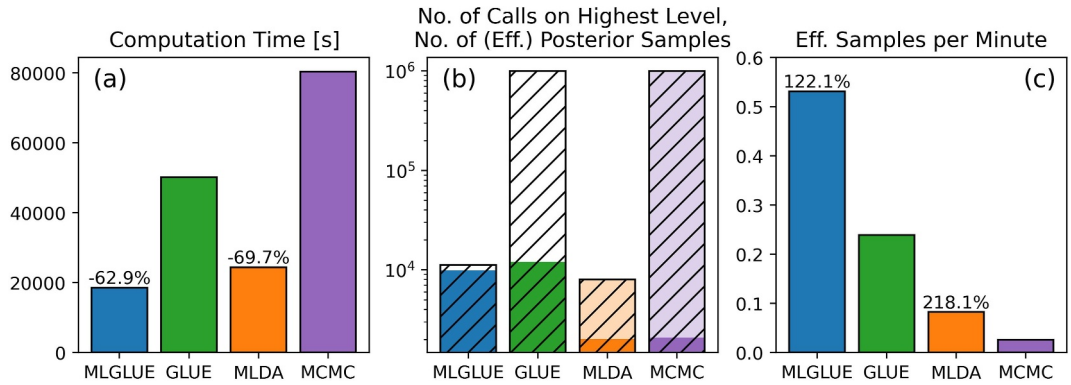

Figure 8. Sampling efficiencies for the groundwater flow example; (a) computation times with percentual reductions with multilevel methods compared to conventional methods; (b) No. of model calls on the highest level (dashed), No. of posterior samples (light colors), No. of effective posterior samples (dark colors); (c) No. of effective posterior samples per minute with percentual increase compared to conventional methods.

The relations between the three levels are shown in Figure S4 in Supporting Information S1. $\mathbb{V}\left[\tilde{\mathcal{L}}_\ell\right]$ and $\mathbb{E}\left[\tilde{\mathcal{L}}_\ell\right]$ are approximately constant and $\mathbb{V}\left[\tilde{\mathcal{L}}_\ell - \tilde{\mathcal{L}}_{\ell-1}\right]$ and $\mathbb{E}\left[\tilde{\mathcal{L}}_\ell - \tilde{\mathcal{L}}_{\ell-1}\right]$ decay across all levels. The variance of the sampled likelihoods on level $\ell = 0$, however, is smaller than on higher levels. The correlation coefficients are 0.9954 between levels $\ell = 0$ and $\ell = 1$, 0.9989 between levels $\ell = 1$ and $\ell = 2$, and 0.9997 between levels $\ell = 2$ and $\ell = 3$ and therefore increase with increasing level index.

The sampling efficiencies of all methods are shown in Figure 8; detailed results of MLDA and MCMC chain convergence (Gelman-Rubin statistic), the recovery of effective samples, and proposal hyperparameters are described in Text S4 in Supporting Information S1. The overall computation time is reduced by ≈63 % and the number of effective samples per minute is ≈122 % higher with MLGLUE compared to GLUE. The overall computation time is reduced by ≈70 % and the number of effective samples per minute is ≈206 % higher with MLDA compared to conventional MCMC. The ratio between the number of effective samples to the total number of posterior samples on the highest level is substantially higher for MLDA compared to MCMC, indicating lower sample autocorrelation before thinning. More detailed analyses of MLDA and MCMC results are presented in Supporting Information S1.

The results of convergence analysis (see Section 2.5) are shown in Figure 9. Results are obtained by splitting the original sets of effective parameter samples into 200 consecutive subsets, independently of the method of inference. Multilevel approaches (MLGLUE and MLDA) generally converge after a shorter computation time compared to their conventional counterparts (GLUE and MCMC), respectively. The deviation of mean and variance is larger with MLGLUE compared to GLUE, especially for small sample sizes, although the set of prior samples is equal for MLGLUE and GLUE. MLDA and MCMC results show similar convergence behavior, except for the length scale parameter. MLDA results show larger deviations of the length scale mean and variance for smaller sample sizes.

Estimated CDFs of the parameter posteriors are shown in Figures 10(a)–10(d). Posteriors obtained with MLGLUE are substantially more conditioned than GLUE posteriors (indicated by the deviations of the cumulative distributions from the straight line representing a uniform distribution). The length scale posterior, however, is similar for MLGLUE and GLUE. MLDA and MCMC posteriors are virtually identical. Uncertainty estimates of MLGLUE are different from those of GLUE as they show slightly larger ranges and less bias toward higher values, which can be attributed to the differences in the posterior distributions. Uncertainty estimates from MLDA and MCMC are similarly different in that they have smaller range and less bias toward higher values for MLDA. As evaluated with the squared Pearson correlation coefficient (denoted by R^2), MLGLUE results are slightly more accurate compared to GLUE. Similarly, MLDA results are slightly more accurate compared to MCMC.

10.1029/2024WR037735

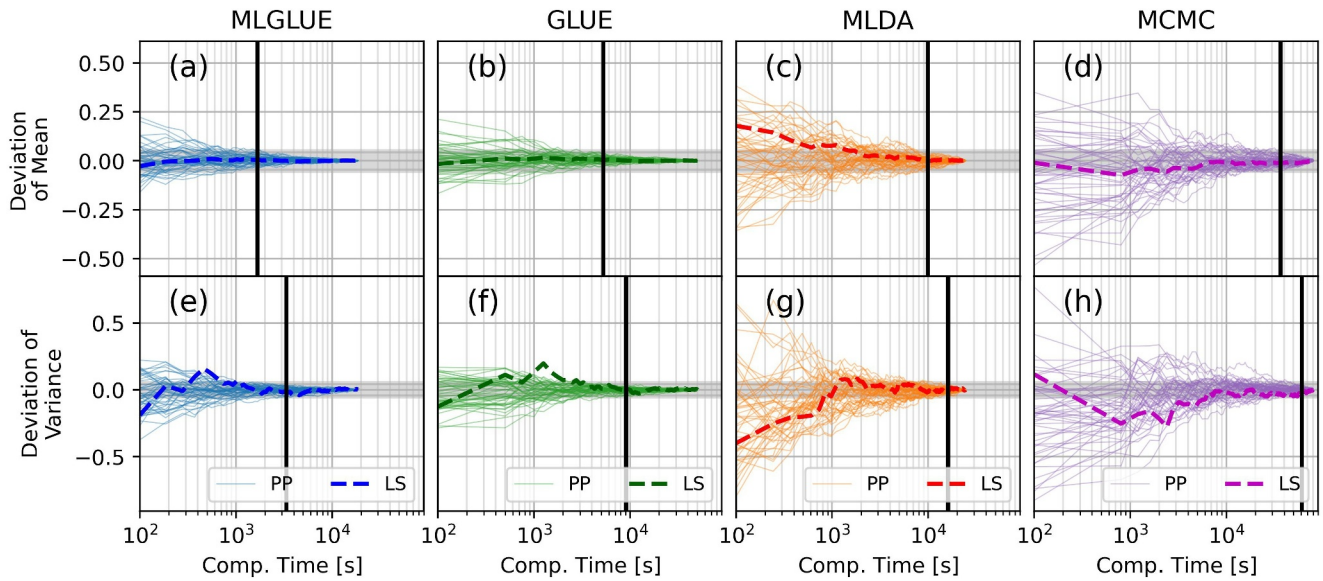

Figure 9. Convergence analysis for the groundwater flow example (Equation 5); for the different methods of inference (a)–(d) shows the deviation of the mean and (e)–(h) shows the deviation of the variance; gray regions represent the region where convergence is achieved; black vertical lines represent the computational time at which convergence is achieved for all parameters.

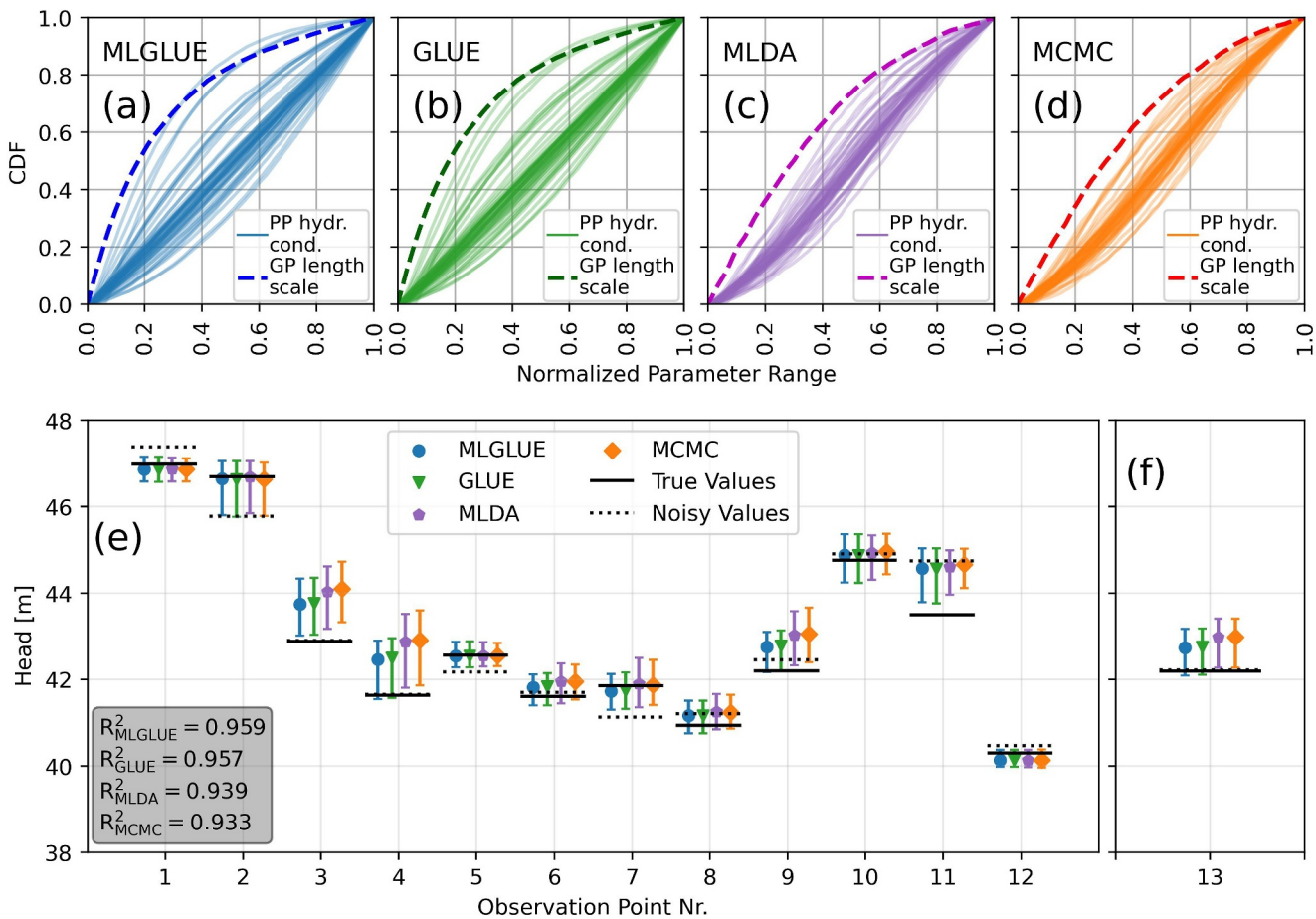

Figure 10. CDFs of model parameters for the groundwater flow example (a, b, c, d) and 99% − 1% uncertainty estimates around the median value for observation points (e) and for the prediction point(f).

5. Discussion

We applied MLGLUE to two test problems and subsequently compared the results to conventional GLUE as well as to MCMC and MLDA. These applications illustrate the capabilities of the multilevel extension but also identifies aspects that need careful consideration for practical applications. The examples considered here are comparable to other examples used to study multilevel methods found in, for example, Cliffe et al. (2011), Dodwell et al. (2019), Lykkegaard et al. (2023), and (Cui et al., 2024). However, although groundwater flow is a frequently used example case, the system used here (see Section 3.2) is far more complex compared to previous applications. Additionally, other previous studies only considered synthetic cases where the underlying truth is known; our rainfall-runoff modeling example considers a real system.

We do not want to debate about the non-formality of GLUE or MLGLUE and the comparison to formal Bayesian approaches such as MCMC or MLDA and refer to other studies for detailed discussions (Vrugt, ter Braak, Gupta, & Robinson, 2009; Li et al., 2010; Jin et al., 2010; Nott et al., 2012; Sadegh & Vrugt, 2013). However, especially in the context of multilevel methods, there is practical relevance in comparing MLGLUE and MLDA as our examples show. While MLDA (and other MCMC-type approaches) effectively *search* for "acceptable" parameter sets (Vrugt, ter Braak, Gupta, & Robinson, 2009), GLUE and MLGLUE *evaluate* samples from a (prior) distribution. With its increased computational efficiency, MLGLUE now can efficiently balance the need to widely sample the parameter space with the typically low acceptance probability of such Monte Carlo-type methods. While the associated computational cost was a limiting factor in the application of GLUE (e.g., Tran & Kim, 2022), MLGLUE now allows for a wider applicability. Furthermore, MLGLUE may be applied to computationally expensive models which do not allow for an explicit definition of a likelihood function (Sadegh & Vrugt, 2013), which is also described in Section 2.4.

For both examples it was identified that the number of tuning samples, N_t, required to obtain stable and accurate estimates of likelihood thresholds in GLUE and MLGLUE increases with decreasing threshold percentage, although the parameter space dimensions were greatly different ($n = 5$ for rainfall-runoff modeling and $n = 51$ for groundwater flow). For a threshold setting of 2 %, $N_t = 5{,}000$ tuning samples were needed for accurate estimation in both examples. For a threshold setting of 7 %, however, only $N_t = 2{,}000$ tuning samples were required for accurate estimation in both examples. This behavior is in agreement with the fact that Monte-Carlo estimators generally do not perform well at rare event estimation (e.g., Beck & Zuev, 2015), which can be translated to the present case of estimating values in the tails of the distribution of likelihood values (i.e., estimating large percentiles). We hypothesize that using a Latin hypercube design or quasi-Monte Carlo sampling during the tuning phase increases robustness as well as computational efficiency.

The model hierarchies were designed for both examples using a coarsening factor of 2. While for the rainfall-runoff modeling this choice resulted in increased computational efficiency of MLGLUE compared to GLUE, a coarsening factor of 3 (results not shown) resulted in a substantially reduced acceptance rate. This was especially evident from a large difference between highest-level model runs and finally accepted samples. The consideration of a fourth level, being even coarser than the current level $\ell = 0$, was not successful as the correlation between the two lowest levels then was found to be very low, again leading to low acceptance rates. Similar behavior was identified for the groundwater flow example, where the likelihood variance on the lowest level with the coarsest resolution was smaller than on subsequently higher levels. As described by Cliffe et al. (2011), further hypothetical grid coarsening beyond the current level $\ell = 0$ for such a case can result in the graphs of $\mathbb{V}\left[\tilde{\mathcal{L}}_\ell\right]$ and $\mathbb{V}\left[\tilde{\mathcal{L}}_\ell - \tilde{\mathcal{L}}_{\ell-1}\right]$ to eventually intersect, resulting in $\mathbb{V}\left[\tilde{\mathcal{L}}_\ell - \tilde{\mathcal{L}}_{\ell-1}\right] > \mathbb{V}\left[\tilde{\mathcal{L}}_\ell\right]$ for some ℓ. In the context of MLMC (forward problems), this then leads to an increased computational cost compared to conventional MC. Insufficient correlation between the likelihood values on subsequent levels in MLGLUE would then result in lower acceptance rates on levels $\ell > 0$, affecting the overall computational efficiency of the algorithm. Therefore, the characteristics of the relation between levels as described in Sections 2.2 and 2.4.2 should also be considered for MLGLUE to ensure computational efficiency. We hypothesize at this point that a non-geometric construction of the hierarchies can potentially further increase computational efficiency (Giles, 2015; Vidal-Codina et al., 2015). The analysis required for this, however, demands additional computational resources to optimize the design as it is associated with a large number of degrees of freedom. Including an error model to account for differences between levels also poses a potentially valuable extension (see Section 2.4.2).

Differences exist in the number of posterior samples between MLGLUE and GLUE. This can be attributed to parameter samples being occasionally discarded on lower levels with lower-resolution models, although they would be accepted on higher levels. This is due to the fact that the likelihoods on subsequent levels are not perfectly correlated in both example applications. This effect is reduced as the correlation between subsequent levels increases; it can be controlled through careful design of the model hierarchy (see Section 2.4.2). This behavior is also reflected in the convergence analysis where, using the same set of prior samples, MLGLUE initially shows larger deviations of posterior mean and variance. Differences in posterior samples also result in small deviations regarding posterior parameter distributions and uncertainty estimates of model outputs.

6. Conclusions

In the hydrological sciences, the popularity of statistical inference and inversion has remained high. However, the applicability of corresponding approaches to more complex models and in the context of digital twins has been limited by the associated computational cost of solving inverse problems. The goal of our study was to introduce and test an extension to the GLUE methodology for approximate Bayesian inversion that alleviates the problems associated with computationally costly models through considering multiple levels of model resolution (MLGLUE). Inspired by multilevel Monte Carlo, in MLGLUE most parameter samples are evaluated on lower levels with computationally cheaper low-resolution models instead of using a (data-driven) surrogate model that is decoupled from the high-resolution or target model. Only samples associated with a likelihood above a certain threshold, which can optionally be estimated during a tuning phase of the algorithm, are subsequently passed to higher levels with costly high-resolution models for evaluation. Inferences are made at the level of the highest-resolution model but substantial computational savings are achieved by discarding samples with low likelihood already on levels with low resolution and low computational cost.

MLGLUE is evaluated using example inverse problems involving a rainfall-runoff model and a groundwater flow model. The results of approximate Bayesian inversion with MLGLUE are compared to the results from GLUE. Findings are furthermore compared to results of exact Bayesian inversion using single-level Markov-chain Monte Carlo (MCMC) as well as multilevel delayed acceptance (MLDA) MCMC. However, this comparison is made without generally debating differences between approximate and exact Bayesian inversion (see also Section 5). Identical numbers of prior samples are considered for GLUE and MLGLUE. Similarly, the same number is used to pre-define the number of steps the MCMC and MLDA samplers take. We show that the results (parameter posteriors, uncertainty estimates, convergence behavior) obtained with multilevel approaches (MLGLUE and MLDA) are highly similar to conventional approaches (GLUE and MCMC), respectively.

We identified in both example applications that MLGLUE and MLDA generally result in less precise estimates of parameter posteriors for small effective sample sizes compared to GLUE and MCMC, respectively. This effect, however, vanishes for larger sample sizes required in practical applications. For both examples, MLGLUE resulted in the lowest computational time for inversion and the highest number of effective samples per minute compared to all other methods. We expect the computational benefit of using MLGLUE to increase as the computational cost of a single model call increases, which has been previously identified for multilevel Monte Carlo and multilevel inversion (Cliffe et al., 2011; Dodwell et al., 2019; Giles, 2015; Lykkegaard et al., 2023).

Our results demonstrate that.

- By considering a hierarchy of models with decreasing (spatial) resolution, MLGLUE can substantially reduce the computational cost of statistical inversion for different kinds of hydrological models.
- MLGLUE is most effective for differential-equation-based models, such as they are often encountered in the hydrological sciences; notions of grid or time-step refinement and coarsening are well understood in such cases and MLGLUE may be directly applied.
- Although rigorous criteria on the choice of the number of levels and the coarsening factor do not exist, for MLGLUE there should be as few levels as possible with differences in resolution being as large as possible. Those aspects are restricted by the quality of the coarsest-level model being sufficiently high, the required resolution on the highest level, and the requirement for sufficiently high correlation between subsequent levels. A non-geometric construction of the hierarchy promises to be an alternative, however being associated with elevated computational cost to optimize the hierarchy (see Section 2.4.2).

- Statistical analysis of model outputs on all levels can potentially reveal various aspects such as the impact of model resolution on quantities of interest or the possibility for model simplification. This offers an interesting direction for future research with multilevel methods.

Conflict of Interest

The authors declare no conflicts of interest relevant to this study.

Data Availability Statement

Relevant resources needed to reproduce the results as well as figures are openly available and can be found under the DOI 10.5281/zenodo.13122535 (Rudolph et al., 2024). The MLGLUE algorithm is available as a Python package under https://github.com/iGW-TU-Dresden/MLGLUE.

Acknowledgments
Funding for Thorsten Wagener has been provided by the Alexander von Humboldt Foundation in the framework of the Alexander von Humboldt Professorship endowed by the German Federal Ministry of Education and Research. The authors thank Robert Scheichl, Aretha Teckentrup, and Anastasia Istratuca for fruitful discussions and inspiration. The authors also thank the Associate Editor and three anonymous reviewers who provided helpful comments that improved this manuscript. We gratefully acknowledge the support by the Open Access Publishing Fund of TU Dresden. Open Access funding enabled and organized by Projekt DEAL.

References

Allgeier, J. T. (2022). Analytical and stochastic numerical methods for the simulation of subsurface flow in floodplains. https://doi.org/10.15496/publikation-76913

Anderson, M. P., Woessner, W. W., & Hunt, R. J. (2015). *Applied groundwater modeling: Simulation of flow and advective transport* (2nd ed.). Academic Press. (OCLC: ocn921253555).

Asher, M. J., Croke, B. F. W., Jakeman, A. J., & Peeters, L. J. M. (2015). A review of surrogate models and their application to groundwater modeling: Surrogates of groundwater models. *Water Resources Research*, *51*(8), 5957–5973. https://doi.org/10.1002/2015WR016967

Beck, J. L., & Zuev, K. M. (2015). Rare-event simulation. In R. Ghanem, D. Higdon, & H. Owhadi (Eds.), *Handbook of uncertainty quantification* (pp. 1–26). Springer International Publishing. https://doi.org/10.1007/978-3-319-11259-6_24-1

Beven, K. (1993). Prophecy, reality and uncertainty in distributed hydrological modelling. *Advances in Water Resources*, *16*(1), 41–51. https://doi.org/10.1016/0309-1708(93)90028-E

Beven, K. (2006). A manifesto for the equifinality thesis. *Journal of Hydrology*, *320*(1–2), 18–36. https://doi.org/10.1016/j.jhydrol.2005.07.007

Beven, K. (2016). Facets of uncertainty: Epistemic uncertainty, non-stationarity, likelihood, hypothesis testing, and communication. *Hydrological Sciences Journal*, *61*(9), 1652–1665. https://doi.org/10.1080/02626667.2015.1031761

Beven, K., & Binley, A. (1992). The future of distributed models: Model calibration and uncertainty prediction. *Hydrological Processes*, *6*(3), 279–298. https://doi.org/10.1002/hyp.3360060305

Beven, K., & Binley, A. (2014). Glue: 20 years on. *Hydrological Processes*, *28*(24), 5897–5918. https://doi.org/10.1002/hyp.10082

Beven, K., & Freer, J. (2001). Equifinality, data assimilation, and uncertainty estimation in mechanistic modelling of complex environmental systems using the GLUE methodology. *Journal of Hydrology*, *19*.

Binley, A., Beven, K., & Elgy, J. (1989). A physically based model of heterogeneous hillslopes: 2. Effective hydraulic conductivities. *Water Resources Research*, *25*(6), 1227–1233. https://doi.org/10.1029/WR025i006p01227

Blasone, R.-S., Vrugt, J. A., Madsen, H., Rosbjerg, D., Robinson, B. A., & Zyvoloski, G. A. (2008). Generalized likelihood uncertainty estimation (GLUE) using adaptive Markov chain Monte Carlo sampling. *Advances in Water Resources*, *31*(4), 630–648. https://doi.org/10.1016/j.advwatres.2007.12.003

Blöschl, G., Bierkens, M. F., Chambel, A., Cudennec, C., Destouni, G., Fiori, A., et al. (2019). Twenty-three unsolved problems in hydrology (UPH) – A community perspective. *Hydrological Sciences Journal*, *64*(10), 1141–1158. https://doi.org/10.1080/02626667.2019.1620507

Boyle, D. P. (2001). *Multicriteria calibration of hydrologic models*. University of Arizona, Retrieved from http://hdl.handle.net/10150/290657

Braun, M. (1993). *Differential equations and their applications: An introduction to applied mathematics* (Vol. 11). Springer New York. https://doi.org/10.1007/978-1-4612-4360-1

Brunetti, G., Šimunek, J., Wöhling, T., & Stumpp, C. (2023). An in-depth analysis of Markov-Chain Monte Carlo ensemble samplers for inverse vadose zone modeling. *Journal of Hydrology*, *624*, 129822. https://doi.org/10.1016/j.jhydrol.2023.129822

Burrows, W., & Doherty, J. E. (2015). Efficient calibration/uncertainty analysis using paired complex/surrogate models. *Ground Water*, *53*(4), 531–541. https://doi.org/10.1111/gwat.12257

Carrera, J., Alcolea, A., Medina, A., Hidalgo, J., & Slooten, L. J. (2005). Inverse problem in hydrogeology. *Hydrogeology Journal*, *13*(1), 206–222. https://doi.org/10.1007/s10040-004-0404-7

Cliffe, K. A., Giles, M. B., Scheichl, R., & Teckentrup, A. L. (2011). Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients. *Computing and Visualization in Science*, *14*(1), 3–15. https://doi.org/10.1007/s00791-011-0160-x

Colecchio, I., Boschan, A., Otero, A. D., & Noetinger, B. (2020). On the multiscale characterization of effective hydraulic conductivity in random heterogeneous media: A historical survey and some new perspectives. *Advances in Water Resources*, *140*, 103594. https://doi.org/10.1016/j.advwatres.2020.103594

Cui, T., Detommaso, G., & Scheichl, R. (2024). Multilevel dimension-independent likelihood-informed MCMC for large-scale inverse problems. *Inverse Problems*, *40*(3), 035005. https://doi.org/10.1088/1361-6420/ad1e2c

Dodwell, T. J., Ketelsen, C., Scheichl, R., & Teckentrup, A. L. (2019). *Multilevel Markov chain Monte Carlo*. SIAM / ASA Journal of Uncertainty Quantification.

Doherty, J. E. (2003). Ground water model calibration using pilot points and regularization. *Ground Water*, *41*(2), 170–177. https://doi.org/10.1111/j.1745-6584.2003.tb02580.x

Doherty, J. E. (2015). *Calibration and uncertainty analysis for complex environmental models*. Watermark Numerical Computing.

Doherty, J. E., & Christensen, S. (2011). Use of paired simple and complex models to reduce predictive bias and quantify uncertainty. *Water Resources Research*, *47*(12). https://doi.org/10.1029/2011WR010763

Erdal, D., & Cirpka, O. A. (2020). Technical Note: Improved sampling of behavioral subsurface flow model parameters using active subspaces. *Hydrology and Earth System Sciences*, *24*(9), 4567–4574. https://doi.org/10.5194/hess-24-4567-2020

Gallagher, K., Charvin, K., Nielsen, S., Sambridge, M., & Stephenson, J. (2009). Markov chain Monte Carlo (MCMC) sampling methods to determine optimal models, model resolution and model choice for Earth Science problems. *Marine and Petroleum Geology*, *26*(4), 525–535. https://doi.org/10.1016/j.marpetgeo.2009.01.003

Gauch, M., Kratzert, F., Klotz, D., Nearing, G., Lin, J., & Hochreiter, S. (2020). Data for "rainfall-runoff prediction at multiple timescales with a single long short-term memory network. *Zenodo*. https://doi.org/10.5281/zenodo.4072701

Gauch, M., Kratzert, F., Klotz, D., Nearing, G., Lin, J., & Hochreiter, S. (2021). Rainfall–runoff prediction at multiple timescales with a single Long Short-Term Memory network. *Hydrology and Earth System Sciences*, *25*(4), 2045–2062. https://doi.org/10.5194/hess-25-2045-2021

Gelman, A., & Rubin, D. B. (1992). Inference from iterative simulation using multiple sequences. *Statistical Science*, *7*(4). https://doi.org/10.1214/ss/1177011136

Geyer, C. J. (1992). Practical Markov chain Monte Carlo. *Statistical Science*, *7*(4). https://doi.org/10.1214/ss/1177011137

Geyer, C. J. (2011). Introduction to Markov chain Monte Carlo. In *Handbook of Markov chain Monte Carlo* (1st ed., pp. 3–48). Chapman and Hall/CRC. https://doi.org/10.1201/b10905-2

Giles, M. B. (2008). Multilevel Monte Carlo path simulation. *Operations Research*, *56*(3), 607–617. https://doi.org/10.1287/opre.1070.0496

Giles, M. B. (2015). Multilevel Monte Carlo methods. *Acta Numerica*, *24*, 259–328. https://doi.org/10.1017/S096249291500001X

Gosses, M., & Wöhling, T. (2019). Simplification error analysis for groundwater predictions with reduced order models. *Advances in Water Resources*, *125*, 41–56. https://doi.org/10.1016/j.advwatres.2019.01.006

Gosses, M., & Wöhling, T. (2021). Robust data worth analysis with surrogate models. *Ground Water*, *59*(5), 728–744. https://doi.org/10.1111/gwat.13098

Harbaugh, A. W. (2005). *MODFLOW-2005, the U.S. Geological survey modular ground-water model—the ground-water flow process (tech. Rep. No. U.S. Geological survey techniques and methods 6–A16)*. USGS.

Heinrich, S. (2001). Multilevel Monte Carlo methods. In S. Margenov, J. Waśniewski, & P. Yalamov (Eds.), *Large-scale scientific computing* (pp. 58–67). Springer Berlin Heidelberg.

Herman, J. D., Reed, P. M., & Wagener, T. (2013). Time-varying sensitivity analysis clarifies the effects of watershed model formulation on model behavior. *Water Resources Research*, *49*(3), 1400–1414. https://doi.org/10.1002/wrcr.20124

Herrera, P. A., Marazuela, M. A., & Hofmann, T. (2022). Parameter estimation and uncertainty analysis in hydrological modeling. *WIREs Water*, *9*(1), e1569. https://doi.org/10.1002/wat2.1569

Jin, X., Xu, C.-Y., Zhang, Q., & Singh, V. P. (2010). Parameter and modeling uncertainty simulated by GLUE and a formal Bayesian method for a conceptual hydrological model. *Journal of Hydrology*, *383*(3–4), 147–155. https://doi.org/10.1016/j.jhydrol.2009.12.028

Kavetski, D., Kuczera, G., & Franks, S. W. (2006). Bayesian analysis of input uncertainty in hydrological modeling: 1. Theory: Input uncertainty in hydrology, 1. *Water Resources Research*, *42*(3). https://doi.org/10.1029/2005WR004368

Kennedy, J., Ferré, T. P. A., & Creutzfeldt, B. (2016). Time-lapse gravity data for monitoring and modeling artificial recharge through a thick unsaturated zone. *Water Resources Research*, *52*(9), 7244–7261. https://doi.org/10.1002/2016WR018770

Kennedy, M. C., & O'Hagan, A. (2001). Bayesian calibration of computer models. *Journal of the Royal Statistical Society - Series B: Statistical Methodology*, *63*(3), 425–464. https://doi.org/10.1111/1467-9868.00294

Kitanidis, P. K., & Vomvoris, E. G. (1983). A geostatistical approach to the inverse problem in groundwater modeling (steady state) and one-dimensional simulations. *Water Resources Research*, *19*(3), 677–690. https://doi.org/10.1029/WR019i003p00677

Knoben, W. J. M., Freer, J., Fowler, K. J. A., Peel, M. C., & Woods, R. A. (2019). Modular assessment of rainfall–runoff models toolbox (MARRMoT) v1.2: An open-source, extendable framework providing implementations of 46 conceptual hydrologic models as continuous state-space formulations. *Geoscientific Model Development*, *12*(6), 2463–2480. https://doi.org/10.5194/gmd-12-2463-2019

Kuffour, B. N. O., Engdahl, N. B., Woodward, C. S., Condon, L. E., Kollet, S., & Maxwell, R. M. (2020). Simulating coupled surface–subsurface flows with ParFlow v3.5.0: Capabilities, applications, and ongoing development of an open-source, massively parallel, integrated hydrologic model. *Geoscientific Model Development*, *13*(3), 1373–1397. https://doi.org/10.5194/gmd-13-1373-2020

Kumar, R., Carroll, C., Hartikainen, A., & Martin, O. (2019). ArviZ a unified library for exploratory analysis of Bayesian models in Python. *Journal of Open Source Software*, *4*(33), 1143. https://doi.org/10.21105/joss.01143

Kumar, R., Samaniego, L., & Attinger, S. (2013). Implications of distributed hydrologic model parameterization on water fluxes at multiple scales and locations: Distributed hydrologic model parameterizations. *Water Resources Research*, *49*(1), 360–379. https://doi.org/10.1029/2012WR012195

Laloy, E., Rogiers, B., Vrugt, J. A., 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: Speeding up MCMC Simulation of a Groundwater Model. *Water Resources Research*, *49*(5), 2664–2682. https://doi.org/10.1002/wrcr.20226

Laloy, E., & Vrugt, J. A. (2012). High-dimensional posterior exploration of hydrologic models using multiple-try DREAM(ZS) and high-performance computing: Efficient MCMC for high-dimensional problems. *Water Resources Research*, *48*(1). https://doi.org/10.1029/2011WR010608

Leopoldina, G. N. A. o. S. (Ed.) (2022)., *Earth system science: Discovery, diagnosis, and solutions in times of global change: Report on tomorrow's science*. https://doi.org/10.26164/leopoldina_03_00591

Li, L., Xia, J., Xu, C.-Y., & Singh, V. (2010). Evaluation of the subjective factors of the GLUE method and comparison with the formal Bayesian method in uncertainty assessment of hydrological models. *Journal of Hydrology*, *390*(3–4), 210–221. https://doi.org/10.1016/j.jhydrol.2010.06.044

Linde, N., Ginsbourger, D., Irving, J., Nobile, F., & Doucet, A. (2017). On uncertainty quantification in hydrogeology and hydrogeophysics. *Advances in Water Resources*, *110*, 166–181. https://doi.org/10.1016/j.advwatres.2017.10.014

Liu, J. S. (Ed.) (2008)., *Monte Carlo strategies in scientific computing*. Springer.

Liu, Y., Li, J., Sun, S., & Yu, B. (2019). Advances in Gaussian random field generation: A review. *Computational Geosciences*, *23*(5), 1011–1047. https://doi.org/10.1007/s10596-019-09867-y

Lykkegaard, M. B. (2022). tinyDA v0.9.8 [Software]. Retrieved from https://pypi.org/project/tinyda/

Lykkegaard, M. B., & Dodwell, T. J. (2022). Where to drill next? A dual-weighted approach to adaptive optimal design of groundwater surveys. *Advances in Water Resources*, *164*, 104219. https://doi.org/10.1016/j.advwatres.2022.104219

Lykkegaard, M. B., Dodwell, T. J., Fox, C., Mingas, G., & Scheichl, R. (2023). Multilevel delayed acceptance MCMC. *SIAM/ASA Journal on Uncertainty Quantification*, *11*(1), 1–30. https://doi.org/10.1137/22M1476770

Mai, J. (2023). Ten strategies towards successful calibration of environmental models. *Journal of Hydrology*, *620*, 129414. https://doi.org/10.1016/j.jhydrol.2023.129414

Mirzaei, M., Huang, Y. F., El-Shafie, A., & Shatirah, A. (2015). Application of the generalized likelihood uncertainty estimation (GLUE) approach for assessing uncertainty in hydrological models: A review. *Stochastic Environmental Research and Risk Assessment*, *29*(5), 1265–1273. https://doi.org/10.1007/s00477-014-1000-6

Montanari, A. (2007). What do we mean by 'uncertainty'? The need for a consistent wording about uncertainty assessment in hydrology. *Hydrological Processes*, *21*(6), 841–845. https://doi.org/10.1002/hyp.6623

Moore, C., & Doherty, J. E. (2006). The cost of uniqueness in groundwater model calibration. *Advances in Water Resources*, *29*(4), 605–623. https://doi.org/10.1016/j.advwatres.2005.07.003

Moore, C., Wöhling, T., & Doherty, J. (2010). Efficient regularization and uncertainty analysis using a global optimization methodology: REGULARIZATION, uncertainty and global optimization. *Water Resources Research*, *46*(8). https://doi.org/10.1029/2009WR008627

Moritz, P., Nishihara, R., Wang, S., Tumanov, A., Liaw, R., Liang, E., et al. (2018). Ray: A distributed framework for emerging AI applications. In *13th USENIX symposium on operating systems design and implementation (OSDI 18)* (pp. 561–577).

Nash, J., & Sutcliffe, J. (1970). River flow forecasting through conceptual models part I — A discussion of principles. *Journal of Hydrology*, *10*(3), 282–290. https://doi.org/10.1016/0022-1694(70)90255-6

Ng, L. W. T., & Willcox, K. (2014). Multifidelity approaches for optimization under uncertainty: Multifidelity approaches for optimization under uncertainty. *International Journal for Numerical Methods in Engineering*, *100*(10), 746–772. https://doi.org/10.1002/nme.4761

Niswonger, R. G., Panday, S., & Ibaraki, M. (2011). MODFLOW-NWT, A Newton formulation for MODFLOW-2005 (tech. Rep. No. U.S. Geological survey techniques and methods 6–A37).

Nott, D. J., Marshall, L., & Brown, J. (2012). Generalized likelihood uncertainty estimation (GLUE) and approximate Bayesian computation: What's the connection? Technical note. *Water Resources Research*, *48*(12). https://doi.org/10.1029/2011WR011128

Page, T., Smith, P., Beven, K., Pianosi, F., Sarrazin, F., Almeida, S., et al. (2023). Technical note: The CREDIBLE uncertainty estimation (CURE) toolbox: Facilitating the communication of epistemic uncertainty. *Hydrology and Earth System Sciences*, *27*(13), 2523–2534. https://doi.org/10.5194/hess-27-2523-2023

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Peherstorfer, B., Willcox, K., & Gunzburger, M. (2016). Optimal model management for multifidelity Monte Carlo estimation. *SIAM Journal on Scientific Computing*, *38*(5), A3163–A3194. https://doi.org/10.1137/15M1046472

Peherstorfer, B., Willcox, K., & Gunzburger, M. (2018). Survey of multifidelity methods in uncertainty propagation, inference, and optimization. *SIAM Review*, *60*(3), 550–591. https://doi.org/10.1137/16M1082469

Plumlee, M. (2017). Bayesian calibration of inexact computer models. *Journal of the American Statistical Association*, *112*(519), 1274–1285. https://doi.org/10.1080/01621459.2016.1211016

Pokhrel, P., Gupta, H. V., & Wagener, T. (2008). A spatial regularization approach to parameter estimation for a distributed watershed model. *Water Resources Research*, *44*(12), 2007WR006615. https://doi.org/10.1029/2007WR006615

Reinecke, R., Wachholz, A., Mehl, S., Foglia, L., Niemann, C., & Döll, P. (2020). Importance of spatial resolution in global groundwater modeling. *Ground Water*, *58*(3), 363–376. https://doi.org/10.1111/gwat.12996

Rudolph, M. G., Wöhling, T., Wagener, T., & Hartmann, A. (2024). Extending GLUE with multilevel methods to accelerate statistical inversion of hydrological models - code and data. *Zenodo*. https://doi.org/10.5281/zenodo.13122535

Sadegh, M., & Vrugt, J. A. (2013). Bridging the gap between GLUE and formal statistical approaches: Approximate Bayesian computation. *Hydrology and Earth System Sciences*, *17*(12), 4831–4850. https://doi.org/10.5194/hess-17-4831-2013

Samaniego, L., Kumar, R., & Attinger, S. (2010). Multiscale parameter regionalization of a grid-based hydrologic model at the mesoscale: Multiscale parameter regionalization. *Water Resources Research*, *46*(5). https://doi.org/10.1029/2008WR007327

Savage, J. T. S., Pianosi, F., Bates, P., Freer, J., & Wagener, T. (2016). Quantifying the importance of spatial resolution and other factors through global sensitivity analysis of a flood inundation model. *Water Resources Research*, *52*(11), 9146–9163. https://doi.org/10.1002/2015WR018198

Schoups, G., & Vrugt, J. A. (2010). A formal likelihood function for parameter and predictive inference of hydrologic models with correlated, heteroscedastic, and non-Gaussian errors. *Water Resources Research*, *46*(10), 2009WR008933. https://doi.org/10.1029/2009WR008933

Thiros, N. E., Gardner, W. P., Maneta, M. P., & Brinkerhoff, D. J. (2022). Quantifying subsurface parameter and transport uncertainty using surrogate modelling and environmental tracers. *Hydrological Processes*, *36*(11), e14743. https://doi.org/10.1002/hyp.14743

Tonkin, M. J., & Doherty, J. E. (2005). A hybrid regularized inversion methodology for highly parameterized environmental models: Hybrid regularization methodology. *Water Resources Research*, *41*(10). https://doi.org/10.1029/2005WR003995

Tran, V. N., & Kim, J. (2022). Robust and efficient uncertainty quantification for extreme events that deviate significantly from the training dataset using polynomial chaos-kriging. *Journal of Hydrology*, *609*, 127716. https://doi.org/10.1016/j.jhydrol.2022.127716

Trotter, L., & Knoben, W. J. M. (2022). MARRMoT v2.1. *Zenodo*. https://doi.org/10.5281/zenodo.6484372

Trotter, L., Knoben, W. J. M., Fowler, K. J. A., Saft, M., & Peel, M. C. (2022). Modular assessment of rainfall–runoff models toolbox (MARRMoT) v2.1: An object-oriented implementation of 47 established hydrological models for improved speed and readability. *Geoscientific Model Development*, *15*(16), 6359–6369. https://doi.org/10.5194/gmd-15-6359-2022

Vidal-Codina, F., Nguyen, N., Giles, M., & Peraire, J. (2015). A model and variance reduction method for computing statistical outputs of stochastic elliptic partial differential equations. *Journal of Computational Physics*, *297*, 700–720. https://doi.org/10.1016/j.jcp.2015.05.041

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

Vrugt, J. A. (2016). Markov chain Monte Carlo simulation using the DREAM software package: Theory, concepts, and MATLAB implementation. *Environmental Modelling and Software*, *75*, 273–316. https://doi.org/10.1016/j.envsoft.2015.08.013

Vrugt, J. A., & Beven, K. J. (2018). Embracing equifinality with efficiency: Limits of Acceptability sampling using the DREAM(LOA) algorithm. *Journal of Hydrology*, *559*, 954–971. https://doi.org/10.1016/j.jhydrol.2018.02.026

Vrugt, J. A., Diks, C. G. H., Gupta, H. V., Bouten, W., & Verstraten, J. M. (2005). Improved treatment of uncertainty in hydrologic modeling: Combining the strengths of global optimization and data assimilation. *Water Resources Research*, *41*(1), 2004WR003059. https://doi.org/10.1029/2004WR003059

Vrugt, J. A., Gupta, H. V., Bouten, W., & Sorooshian, S. (2003). A shuffled complex evolution metropolis algorithm for optimization and uncertainty assessment of hydrologic model parameters: Efficient method for estimating parameter uncertainty. *Water Resources Research*, *39*(8). https://doi.org/10.1029/2002WR001642

Vrugt, J. A., ter Braak, C. J. F., Clark, M. P., Hyman, J. M., & Robinson, B. A. (2008). Treatment of input uncertainty in hydrologic modeling: Doing hydrology backward with Markov chain Monte Carlo simulation. *Water Resources Research*, *44*(12). https://doi.org/10.1029/2007WR006720

Vrugt, J. A., ter Braak, C. J. F., Diks, C., Robinson, B. A., Hyman, J. M., & Higdon, D. (2009a). Accelerating Markov chain Monte Carlo simulation by differential evolution with self-adaptive randomized subspace sampling. *International Journal of Nonlinear Sciences and Numerical Stimulation*, *10*(3). https://doi.org/10.1515/IJNSNS.2009.10.3.273

Vrugt, J. A., ter Braak, C. J. F., Gupta, H. V., & Robinson, B. A. (2009b). Equifinality of formal (DREAM) and informal (GLUE) Bayesian approaches in hydrologic modeling? *Stochastic Environmental Research and Risk Assessment*, *23*(7), 1011–1026. https://doi.org/10.1007/s00477-008-0274-y

Wagener, T., Boyle, D. P., Lees, M. J., Wheater, H. S., Gupta, H. V., & Sorooshian, S. (2001). A framework for development and application of hydrological models. *Hydrology and Earth System Sciences*, *5*(1), 13–26. https://doi.org/10.5194/hess-5-13-2001

Wagener, T., & Gupta, H. V. (2005). Model identification for hydrological forecasting under uncertainty. *Stochastic Environmental Research and Risk Assessment*, *19*(6), 378–387. https://doi.org/10.1007/s00477-005-0006-5

White, J. T. (2018). A model-independent iterative ensemble smoother for efficient history-matching and uncertainty quantification in very high dimensions. *Environmental Modelling and Software*, *109*, 191–201. https://doi.org/10.1016/j.envsoft.2018.06.009

White, J. T., Hunt, R. J., Fienen, M. N., & Doherty, J. E. (2020a). Approaches to highly parameterized inversion: PEST++ version 5, a software suite for parameter estimation, uncertainty analysis, management optimization and sensitivity analysis (report No. 7-C26). Reston, VA. https://doi.org/10.3133/tm7C26

White, J. T., Knowling, M. J., & Moore, C. R. (2020b). Consequences of groundwater-model vertical discretization in risk-based decision-making. *Ground Water*, *58*(5), 695–709. https://doi.org/10.1111/gwat.12957

Wildemeersch, S., Goderniaux, P., Orban, P., Brouyère, S., & Dassargues, A. (2014). Assessing the effects of spatial discretization on large-scale flow model performance and prediction uncertainty. *Journal of Hydrology*, *510*, 10–25. https://doi.org/10.1016/j.jhydrol.2013.12.020

Zhang, J., Vrugt, J. A., Shi, X., Lin, G., Wu, L., & Zeng, L. (2020). Improving simulation efficiency of MCMC for inverse modeling of hydrologic systems with a Kalman-Inspired proposal distribution. *Water Resources Research*, *56*(3). https://doi.org/10.1029/2019WR025474

Zhou, H., Gómez-Hernández, J. J., & Li, L. (2014). Inverse methods in hydrogeology: Evolution and recent trends. *Advances in Water Resources*, *63*, 22–37. https://doi.org/10.1016/j.advwatres.2013.10.014

Zimmerman, D. A., de Marsily, G., Gotway, C. A., Marietta, M. G., Axness, C. L., Beauheim, R. L., et al. (1998). A comparison of seven geostatistically based inverse approaches to estimate transmissivities for modeling advective transport by groundwater flow. *Water Resources Research*, *34*(6), 1373–1413. https://doi.org/10.1029/98WR00003