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

M. G. Rudolph[1], T. Wöhling[2,3], T. Wagener[4], A. Hartmann[1]

[1]Institute of Groundwater Management, Technische Universität Dresden, Dresden, Germany
[2]Chair of Hydrology, Institute of Hydrology and Meteorology, Technische Universität Dresden, 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 setting with multiple levels of model resolution (MLGLUE)
- We demonstrate the acceleration with MLGLUE for different spatial (groundwater flow model) and temporal (rainfall-runoff model) resolutions
- We find that MLGLUE acceleration is comparable to or more efficient than a multilevel extension of Markov-chain Monte Carlo

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

Abstract

Inverse problems are ubiquitous in hydrological modelling for parameter estimation, system understanding, sustainable water resources management, and the operation of digital twins. 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, e.g., 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 modelling, 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 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 et al., 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). While process-based spatially distributed models are increasingly used to guide decision-making and to sustainably manage water resources, such modelling 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 (Montanari, 2007; Vrugt, 2016; Linde et al., 2017; Page et al., 2023). Generalized Likelihood Uncertainty Estimation (GLUE) (Beven & Binley, 1992, 2014) - as an informal Bayesian approach - and Markov-chain Monte Carlo sampling (MCMC) (Gallagher et al., 2009; Vrugt, 2016; Dodwell et al., 2019; Brunetti et al., 2023; Lykkegaard et al., 2023; Cui et al., 2024) - as a formal Bayesian approach - are frequently applied in the environmental sciences for statistical inversion. The Bayesian framework considers 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 & Binley, 1992; Beven & Freer, 2001; Schoups & Vrugt, 2010; Nott et al., 2012; Sadegh & Vrugt, 2013; Beven, 2016; 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 the often often run instead of the computationally costly high-fidelity model (Doherty & Christensen, 2011; Asher et al., 2015; Burrows & Doherty, 2015; Linde et al., 2017;

Gosses & Wöhling, 2019, 2021; Allgeier, 2022). Reducing model spatial resolution can reduce model complexity and computational cost in general and the effect of horizontal (Wildemeersch et al., 2014; Savage et al., 2016; Reinecke et al., 2020) 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) (Heinrich, 2001; Giles, 2008; Cliffe et al., 2011; Giles, 2015), with extensions to multilevel MCMC and multilevel delayed acceptance MCMC (MLMCMC and MLDA, respectively) (Dodwell et al., 2019; Lykkegaard et al., 2023; Cui et al., 2024), 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, harbouring 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 synergetically used together. Linde et al. (2017) summarize first applications of MLMC for the forward propagation of uncertainties in hydrogeology and hydrogeophysics. We note that 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).

Previous applications of multilevel methods focussed on models with different spatial resolutions (Cliffe et al., 2011; Linde et al., 2017; Dodwell et al., 2019; Lykkegaard et al., 2023; Cui et al., 2024), entailing challenges when transferring parameter fields from one spatial resolution to another. Geostatistical approaches are often used to (initially) assign parameters for spatially distributed groundwater flow- or other hydrological models. This simultaneously reduces overparameterization as the number of geostatistical parameters is much lower than the number of parameters of the computational model. 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 (Kitanidis & Vomvoris, 1983; Zimmerman et al., 1998; Zhou et al., 2014; Doherty, 2003). 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; Dodwell et al., 2019; Lykkegaard et al., 2023; Cui et al., 2024). The definition of hydrological respose units or internally homogeneous zones of parameters represents another strategy for parameterization (Kumar et al., 2013; Zhou et al., 2014; Anderson et al., 2015; White, 2018). To better constrain the parameter space during inversion and to reduce the aggravating effect of overparameterization, regularization can be employed in combination with different parameterization strategies (Tonkin & Doherty, 2005; Moore & Doherty, 2006; Pokhrel et al., 2008; Moore et al., 2010). 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; Samaniego et al., 2010; Colecchio et al., 2020).

While multilevel methods have previously been used to accelerate MCMC algorithms (Dodwell et al., 2019; Lykkegaard & Dodwell, 2022; Lykkegaard et al., 2023; Cui et al., 2024) 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.

2 Methods

2.1 The Inverse Problem

Consider observations $\widetilde{\mathbf{Y}} = [\tilde{y}_1, \ldots, \tilde{y}_k]^T \in \mathcal{Y} \subseteq \mathbb{R}^k$ of a real system and consider a model $\mathcal{F}$ that simulates the system response $\mathbf{Y} = [y_1, \ldots, y_k]^T \in \mathcal{Y}$ corresponding to $\widetilde{\mathbf{Y}}$. The model output also depends on initial and boundary conditions $\mathcal{C}_i$ and $\mathcal{C}_b$,

respectively, as well as on model parameters $\boldsymbol{\theta} \in \mathcal{X} \subseteq \mathbb{R}^n$

$$\widetilde{\mathbf{Y}} = \mathcal{F}(\boldsymbol{\theta}, \mathcal{C}_i, \mathcal{C}_b) + \boldsymbol{\varepsilon} := \mathcal{F}(\boldsymbol{\theta}) + \boldsymbol{\varepsilon} \tag{1}$$

$\mathcal{F} : \mathcal{C}_i, \mathcal{C}_b \rightarrow \mathbf{Y} \in \mathcal{Y}$ is closed by the parameter vector $\boldsymbol{\theta}$ (Kavetski et al., 2006; Vrugt et al., 2009), which is considered a random vector with an associated prior distribution $p_p(\boldsymbol{\theta})$. $\boldsymbol{\varepsilon} \in \mathbb{R}^k$ in this context represents the combined effect of conceptual model error and measurement error (e.g., 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 $p(\boldsymbol{\theta}|\widetilde{\mathbf{Y}})$ via Bayes' theorem

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

where $p(\widetilde{\mathbf{Y}}|\boldsymbol{\theta})$ is the likelihood function and $p(\widetilde{\mathbf{Y}})$ is the proportionality factor called model evidence, which is the average likelihood of the model to have generated 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

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

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 et al. (2009) give alternative likelihood formulations for non-Gaussian errors that often come at the cost of additional hyperparameters.

2.2 Multilevel Monte Carlo

We will discuss the notion of multilevel methods from the perspective of multilevel Monte Carlo (MLMC), which is a method to efficiently compute the expectation of a quantity of interest that depends on (model) parameters (Heinrich, 2001; Giles, 2008; Cliffe et al., 2011; Giles, 2015). Consider the situation where we are given a distribution of model parameters, $p(\boldsymbol{\theta})$, and want to compute the expected value of some scalar quantity related to the model output, $\mathbf{Q} = \mathcal{Q}(\mathcal{F}(\boldsymbol{\theta}))$, with respect to $p(\boldsymbol{\theta})$. Here, $\mathcal{Q}$ represents some

function of the model output, e.g., it yields the system state at a certain location, or a more abstract quantity. As an example, consider $\mathbf{Q}$ to represent the groundwater level at some location in the model domain. Propagating the uncertainty contained in the parameter distributions through the model to represent the uncertainty in $\mathbf{Q}$ is considered a problem of forward propagation of uncertainty, which is the opposite of the inverse problem described in section 2.1. Yet, MLMC builds on a simple intuition that illustrates the idea behind MLGLUE.

For simplicity and without loss of generality consider $\mathbf{Q} \in \mathbb{R}$ for the remainder of this section. Instead of one single model for the system, assume that there is a hierarchy of models (approximations of the real system), which are denoted by $\{\mathcal{F}_\ell\}_{\ell=0}^{\infty}$, where ℓ is the level index. Associated with each model in the hierarchy are values for the quantity of interest, $\{\mathbf{Q}_\ell\}_{\ell=0}^{\infty}$, such that $\widetilde{\mathbf{Q}} = \lim_{\ell\to\infty} \mathbf{Q}_\ell$, where $\widetilde{\mathbf{Q}}$ represents the true value. In the context of PDE-based models, ℓ may be related to the grid size or time step length of the model, i.e., a larger ℓ corresponds to a higher domain resolution with smaller computational cells or smaller time steps, for example. We assume that the computational cost for evaluating $\mathcal{F}_\ell$ (or $\mathbf{Q}_\ell$) increases while the approximation error decreases as $\ell \to L$. Here L is the index of the highest level, which is often associated with the target model and all lower levels have lower resolution. We note that the most common form of the model hierarchy is a geometric series of computational grids, where the factor of refinement or coarsening between subsequent levels is constant across all levels (Cliffe et al., 2011; Giles, 2015). To estimate the expectation of $\mathbf{Q}$ efficiently, MLMC avoids the direct estimation of $\mathbb{E}[\mathbf{Q}_L]$ on the highest level $\ell = L$. Instead, the correction of the estimation with respect to the next lower level is computed, based on the linearity of expectation:

$$\mathbb{E}[\mathbf{Q}_L] = \mathbb{E}[\mathbf{Q}_0] + \sum_{\ell=1}^{L} \mathbb{E}[\mathbf{Q}_\ell - \mathbf{Q}_{\ell-1}] \tag{4}$$

This approach generally results in substantial computational savings and different multilevel estimators for $\mathbb{E}[\mathbf{Q}_L]$ exist (Giles, 2008; Cliffe et al., 2011; Giles, 2015; Dodwell et al., 2019; Lykkegaard et al., 2023; Cui et al., 2024). The original MLMC algorithm of Giles (2008) (as well as subsequently applied algorithms) takes a bottom-up approach, i.e., sampling is started on $\ell = 0$ and ℓ is only incremented if the algorithm has not yet converged on level ℓ. There, efficiency and variance reduction regarding the ex-

pectation of $\mathbf{Q}$ may be optimized by choosing an optimal refinement (e.g., the decrease of cell or time step size when going from ℓ to $\ell + 1$).

In the context of MLMC, the behaviour of the variances $\mathbb{V}[\mathbf{Q}_\ell]$ and $\mathbb{V}[\mathbf{Q}_\ell - \mathbf{Q}_{\ell-1}]$ and expectations $\mathbb{E}[\mathbf{Q}_\ell]$ and $\mathbb{E}[\mathbf{Q}_\ell - \mathbf{Q}_{\ell-1}]$ as $\ell \to L$ gives an indication of the overall quality and efficiency of the hierarchy $\{\mathbf{Q}_\ell\}_{\ell=0}^{L}$ (Cliffe et al., 2011). $\mathbb{V}[\mathbf{Q}_\ell]$ 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. $\mathbb{V}[\mathbf{Q}_\ell - \mathbf{Q}_{\ell-1}]$ may be expanded as

$$\mathbb{V}[\mathbf{Q}_\ell - \mathbf{Q}_{\ell-1}] = \mathbb{V}[\mathbf{Q}_\ell] + \mathbb{V}[\mathbf{Q}_{\ell-1}] - 2 \cdot \mathrm{Cov}(\mathbf{Q}_\ell, \mathbf{Q}_{\ell-1}), \tag{5}$$

showing that it should be given that $2 \cdot \mathrm{Cov}(\mathbf{Q}_\ell, \mathbf{Q}_{\ell-1}) > \mathbb{V}[\mathbf{Q}_{\ell-1}]$, which requires $\mathbf{Q}_\ell$ and $\mathbf{Q}_{\ell-1}$ to be sufficiently correlated.

While those relations between levels are not formally required to hold for inversion, they ensure that the multilevel estimator for the expectation of $\mathbf{Q}$ has reduced variance and is computationally more efficient compared to a single-level estimator (Cliffe et al., 2011; Lykkegaard et al., 2023). While a deviation of the previously described optimal relations between levels does not necessarily indicate a poorly performing model hierarchy, without such a deviation the hierarchy may be said to be well behaved. We discuss 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) on the basis of the delayed acceptance algorithm coupled with the randomized-length-subchain surrogate transition (Christen & Fox, 2005; J. S. Liu, 2008) and includes many concepts similar to MLMC described in section 2.2. Delayed acceptance MCMC has been employed by Laloy et al. (2013) to accelerate Bayesian inversion for groundwater flow models using a generalized polynomial chaos surrogate model. The main functionality of MLDA is shown in Fig. 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, i.e., 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 Eq. 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 Eq. 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 recusively calling the MLDA algorithm on level $\ell - 1$, subchains with length J_ℓ are generated on levels $1 \leq \ell \leq L$ until level $\ell = 0$ is 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 $\widehat{R}$ is frequently used for multi-chain samplers (Gelman & Rubin, 1992; Lykkegaard et al., 2023). A value of $\widehat{R} \leq 1.2$ is often deemed sufficient to ensure convergence (e.g., Vrugt, 2016). MCMC (and MLDA) samples from converged chains are naturally correlated and may show dependence on initial samples, requiring that an initial number of samples is burned and that that samples are thinned (e.g., every other sample may be omitted to reduce autocorrelation) to obtain approximately independent samples (e.g., Vrugt, 2016; Lykkegaard et al., 2023). The number of approximately intependent samples is termed the estimated effective sample size and can be calculated as shown in Geyer (1992, 2011). We obtain effective samples by burning initial samples such that $\widehat{R} \leq 1.2$ for all chains, followed by thinning such that the resulting number of samples is approximately equal to the estimated effective sample size. We denote this set of effective sam-

ples 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 et al., 2009; Beven & Binley, 2014; Mirzaei et al., 2015).

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 behavioural (i.e., good agreement between $\mathbf{Y}$ and $\widetilde{\mathbf{Y}}$) and non-behavioural (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 et al., 2009):

$$\widetilde{\mathcal{L}}(\boldsymbol{\theta}|\widetilde{\mathbf{Y}}) := (\sigma_r^2)^{-W} = \left(\frac{\sum_{i=1}^{k} (y_i - \tilde{y}_i)^2}{k-2} \right)^{-W} \tag{6}$$

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, $\{\boldsymbol{\theta}^{(j)}\}_{j=1}^{N}$, 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 behavioural) 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 et al., 2009). Using only behavioural solutions, (cumulative) probability distributions of model out-

puts are generated, from which uncertainty estimates are finally computed. Behavioural parameter samples are used to estimate the posterior distribution of model parameters.

MLGLUE is generally similar to MLDA (or MLMCMC) as shown in Fig. 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., behavioural solutions) and samples being discarded (i.e., non-behavioural solutions).

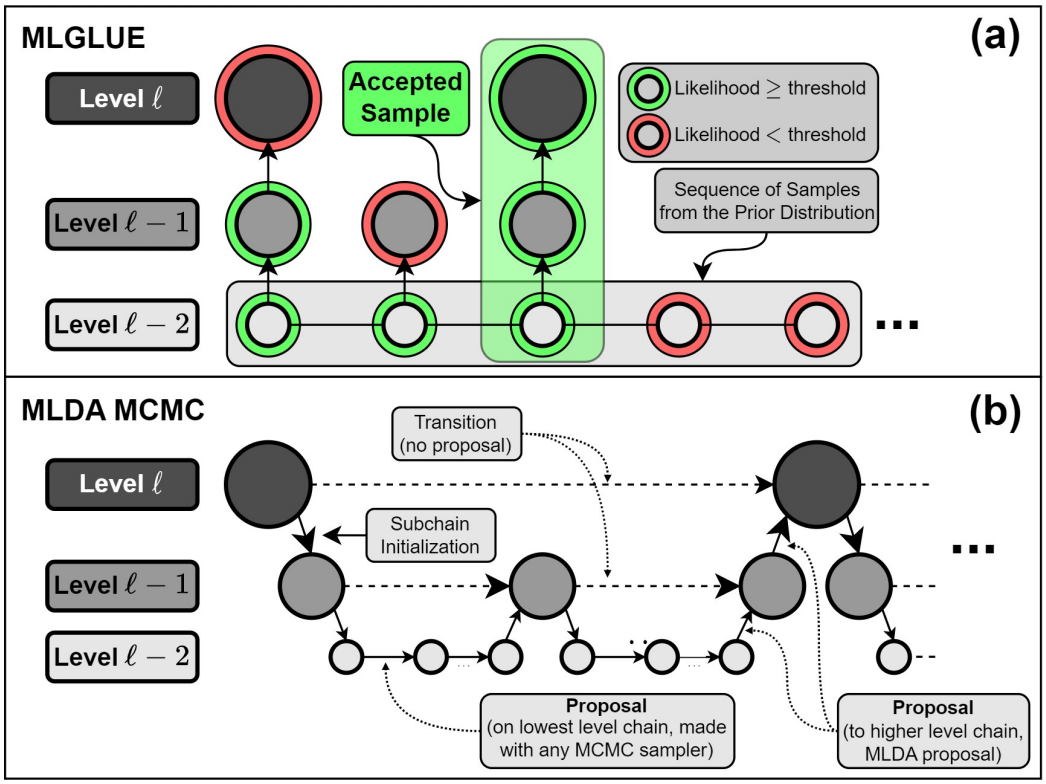


Figure 1. Schematic representation of multilevel sampling strategies for the case of three levels; (a) MLGLUE approach, green rings indicate a likelihood that is above the level-dependent threshold, red rings indicate the contrary; (b) Multilevel Delayed Acceptance MCMC; circles represent the state or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

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 $\{\widetilde{\mathcal{L}}^{(i,\ell)}\}_{i=1}^{N_t}$ and the combined set for all levels by $\{\{\widetilde{\mathcal{L}}^{(i,\ell)}\}_{i=1}^{N_t}\}_{\ell=0}^{L}$. The likelihood thresholds on the different levels are then obtained by computing a pre-defined percentile estimate from the level-dependent likelihood samples (for example, for a threshold corresponding to the top 5% the 95%-percentile is computed). We denote the set of likelihood thresholds on each level by $\{\widetilde{\mathcal{L}}_{T,\ell}\}_{\ell=0}^{L}$. 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, e.g., from expert knowledge.

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

Afterwards, *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 behavioural or non-behavioural. If the likelihood is smaller than the level-dependent threshold on any level, the sample is immediately regarded as non-behavioural 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 Fig. 2.

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 multilevel MCMC 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

Algorithm 1: Multilevel Generalized Likelihood Uncertainty Estimation

Draw a sample $\boldsymbol{\Theta}_0$ 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 likelihood $\widetilde{\mathcal{L}}^{(j,\ell)} = \widetilde{\mathcal{L}}(\boldsymbol{\theta}^{(j)}|\widetilde{\mathbf{Y}})$ with sample $\boldsymbol{\theta}^{(j)}$ from $\boldsymbol{\Theta}_0$ 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)

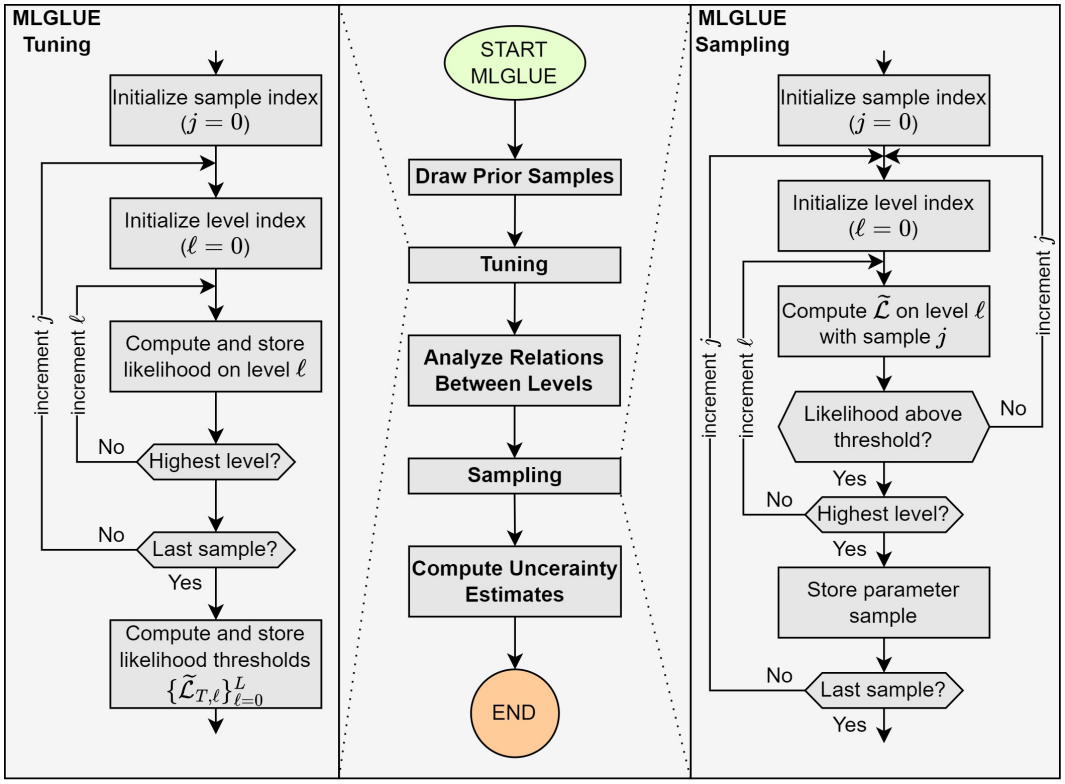


Figure 2. Schematic representation of the multilevel Generalized Likelihood Uncertainty Estimation algorithm; tuning refers to the (optional) Monte Carlo estimation of likelihood thresholds, sampling refers to the repeated evaluation of parameter samples (see the description of algorithm steps)

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 multilevel MCMC, 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 be-

fore. 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 $\{\{\widetilde{\mathcal{L}}^{(i,\ell)}\}_{i=1}^{N_t}\}_{\ell=0}^{L}$ (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. Afterwards, 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.

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` (Team, 2022) 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. While the MLDA implementation (`tinyDA v0.9.8`, Lykkegaard (2022)) does not use the `multiprocessing.Pool` API, it still 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 (1) the differences in the implementation of parallelization and (2) 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 $\mathbf{B}$). As the subset gets larger, and eventually becomes equal to $\mathbf{B}$, this quantity allows for the analysis of convergence behaviour. 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

$$\mathcal{D}_m^s := \frac{\mu_m^s}{\mu_m^{N_b}} - 1 \tag{7}$$

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 towards zero as s increases allows for the analysis of convergence behaviour. 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 all methods to ensure comparability. For GLUE and MLGLUE, an informal likelihood function (Eq. 6) is used for each problem. MCMC and MLDA are used with a formal likelihood function (Eq. 3). We analyze the tuning phase separately for both examples using two threshold settings (selecting the top 2 % and 7 %) for different N_t.

For reasons of reproducibility, seeds are used for pseudo-random number generation, which is used in multiple places (e.g., drawing samples form a distribution); for each problem, the same seeds are used for all methods of inference in the example under study.

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 with a DREAM(Z)-sampler, which is similar to the DREAM(ZS)-sampler (Vrugt, 2016; Lykkegaard, 2022), using `Ray v2.2.0` (Team, 2022) for parallelization. `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` (Team, 2022). 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.

3.1 Rainfall-Runoff Modelling

The first case study considers rainfall-runoff modelling using the conceptual model HYMOD (Boyle, 2001), which is schematically shown in Fig. 3. The model has five parameters (explained in Fig. 3), takes time series of precipitation, $P(t)$ $[LT^{-1}]$, and potential evaporation, $PET(t)$ $[LT^{-1}]$, as inputs and outputs a time series of discharge, $Q(t)$ $[LT^{-1}]$. This model has been frequently and similarly used in the 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, 2009; Herman et al., 2013). 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, 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 2009-10-01 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 four 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 $p_0(\boldsymbol{\theta})$ 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 $[h]$

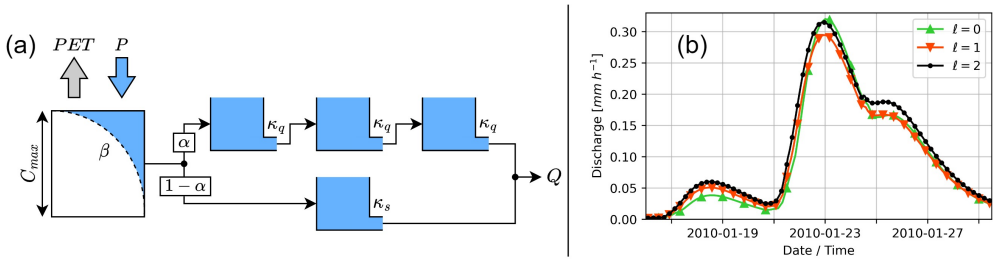


Figure 3. (a) Schematic representation of the HYMOD model (Vrugt et al., 2009); C_{max} $[L]$ is the maximum catchment storage, β $[-]$ is the spatial variability of soil moisture storage, α $[-]$ is the distribution factor between reservoirs, and κ_q $[T^{-1}]$ and κ_s $[T^{-1}]$ are discharge coefficients 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

are used throughout the model and for all datasets. A total number of $N_t+N = 5,000+995,000 = 1,000,000$ samples are drawn from $p_p(\boldsymbol{\theta})$ with each inference method, 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. The choice of N_t is discussed in section 4.1. A constant variance equal to the constant additive Gaussian noise variance ($\sigma^2 = 1.0\ mm^2h^{-2}$) is used for the Gaussian likelihood (see Eq. 3); for the likelihood used in MLGLUE and GLUE (see Eq. 6) $W = 1$ 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:

$$\frac{\partial}{\partial x}\left(K_{xx}\frac{\partial h}{\partial x}\right)+\frac{\partial}{\partial y}\left(K_{yy}\frac{\partial h}{\partial y}\right)+R=0 \tag{8}$$

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

$$\frac{\partial h}{\partial y}=0 \quad \forall x\in\partial\Omega, y\in\{0\ m, 5,000\ m\} \tag{10}$$

$$\frac{\partial h}{\partial x}=0 \quad \forall y\in\partial\Omega, x=10,000\ m \tag{11}$$

$$f_{riv}=c_{riv}\Delta h \quad \forall 0\ m\leq x\leq 10,000\ m, y=1,000\ m \tag{12}$$

where K $[LT^{-1}]$ is the hydraulic conductivity field, h $[L]$ is the hydraulic head field, R $[LT^{-1}]$ is the recharge flux, f_{riv} $[LT^{-1}]$ is river inflow, and c_{riv} $[T^{-1}]$ 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 × 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 $2\cdot 10^{-5}$ 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 Fig. 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.

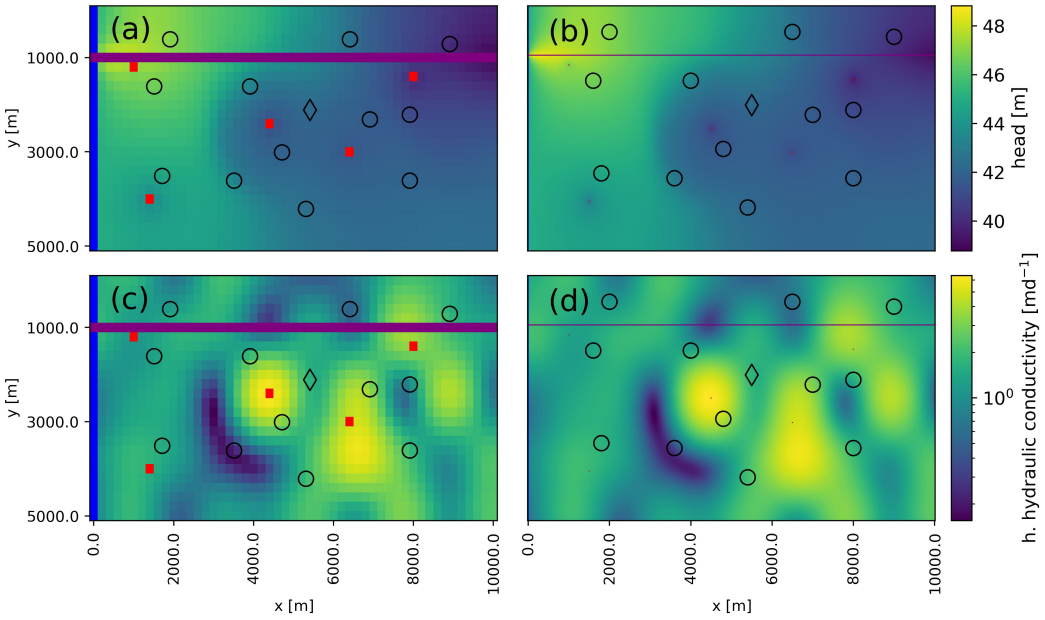


Figure 4. Groundwater flow model domain; head contours obtained with true parameters on level $\ell = 0$ (a) and on level $\ell = 3$ (b); horizontal hydraulic conductivity field on level $\ell = 0$ (c) and on level $\ell = 3$ (d); specific characteristics are: constant head cells (blue), river cells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

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., Eqs. 8 to 12) 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 $\boldsymbol{\theta}_l = [1 \cdot 10^{-2}, \ldots, 1 \cdot 10^{-2}, 5 \cdot 10^2]$ and upper bounds $\boldsymbol{\theta}_u = [1 \cdot 10^1, \ldots, 1 \cdot 10^1, 1 \cdot 10^3]$. A total number of $N_t + N = 2,000 + 98,000 = 100,000$ samples are drawn from $p_p(\boldsymbol{\theta})$ with each inference method, 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. The choice of N_t is discussed in section 4.2. A constant variance equal to the constant additive Gaussian noise variance ($\sigma^2 = 1.0\ m^2$) is used for the Gaussian likelihood (see Eq. 3); for informal likelihoods (see Eq. 6) $W = 1$ is used. The likelihood thresholds are estimated to correspond to the best 7% of all simulations. For MLDA, the sub-sampling rate is set to 5. All methods are run on 32 dual-core CPUs (64 total threads).

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.

MCMC chains typically exhibit a transition period where the samples approach the posterior distribution. The samples of this transition period are discarded as *burn-in* (Gallagher et al., 2009; Brunetti et al., 2023). GLUE and MLGLUE both result in independent posterior samples, while MCMC and MLDA result in correlated posterior samples. To compare both groups (GLUE & MLGLUE and MCMC & MLDA) on an equal basis, independent samples are obtained from MCMC and MLDA samples via *thinning*; only every $\mathcal{K}$-th sample is considered for subsequent analysis. We apply thinning such that the thinned number of samples is approximately equal to the estimated effective sample size of unthinned samples (see section 2.3).

4.1 Rainfall-Runoff Modelling

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 Fig. S1 in the supplementary information 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 Fig. S2 in the supplementary information. $\mathbb{V}[\widetilde{\mathcal{L}}_\ell]$ and $\mathbb{E}[\widetilde{\mathcal{L}}_\ell]$ are approximately constant across all levels and $\mathbb{V}[\widetilde{\mathcal{L}}_\ell - \widetilde{\mathcal{L}}_{\ell-1}]$ and $\mathbb{E}[\widetilde{\mathcal{L}}_\ell - \widetilde{\mathcal{L}}_{\ell-1}]$ 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 Fig. 5; detailed results of MLDA and MCMC chain convergence (Gelman-Rubin statistic) and the recovery of effective samples is described in Text S3 in the supplementary information. 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 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 alayses of MLDA and MCMC results are presented in the supporting information.

The results of convergence analysis (see section 2.5) are shown in Fig. 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 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 Fig. 7 (a) - (d). Posteriors obtained with multilevel methods (MLGLUE and MLDA) are virtually identical to their conventional counterparts (GLUE and MCMC).

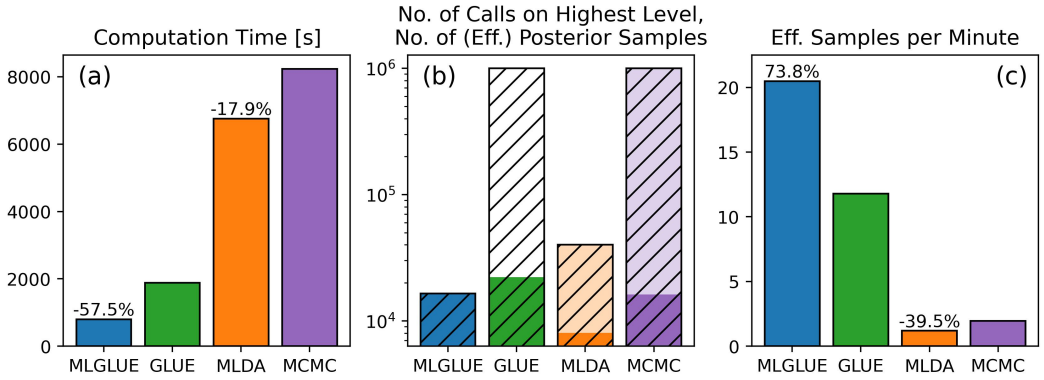


Figure 5. Sampling efficiencies for the rainfall-runoff modelling example; (a) computation times with percentual reductions compared to conventional methods; (b) No. of model calls on the highest level (dashed), No. of posterior samples (light colors), No. of effective posterior samples (dark colors);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

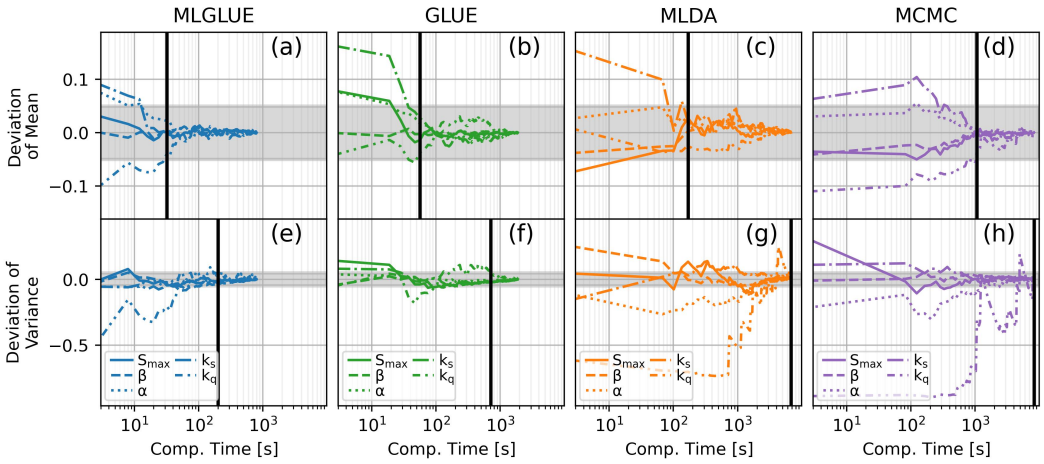


Figure 6. Convergence analysis for the rainfall-runoff modelling example (Eq. 7); for the different methods of inference (a) - (d) shows the deviation of the mean and (e) - (h) shows the deviation of the variance; grey regions represent the region where convergence is achieved; black vertical lines represent the computational time at which convergence is achieved for all parameters

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.

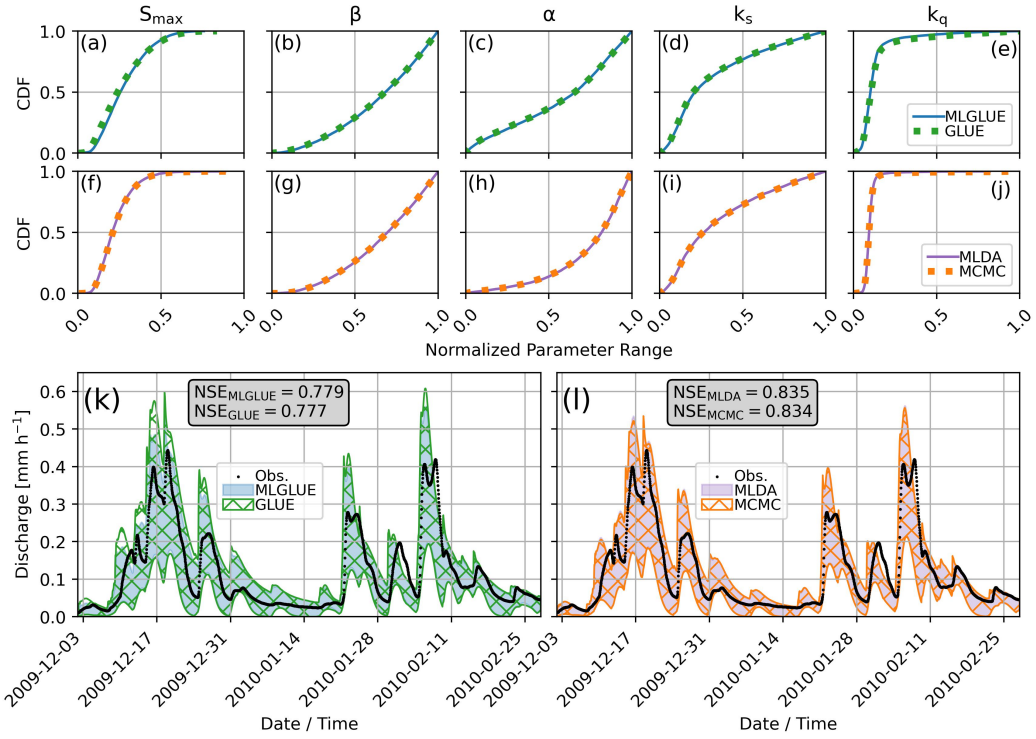


Figure 7. CDFs of model parameters for the rainfall-runoff modelling example for MLGLUE and GLUE (a to e), for MLDA and MCMC (f to j) and 99% − 1% uncertainty estimates around the median value for MLGLUE and GLUE (k) and for MLDA and MCMC (l)

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 Fig. S3 in the supplementary information 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 threhsold 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.

The relations between the three levels are shown in Fig. S4 in the supplementary information. $\mathbb{V}[\widetilde{\mathcal{L}}_\ell]$ and $\mathbb{E}[\widetilde{\mathcal{L}}_\ell]$ are approximately constant and $\mathbb{V}[\widetilde{\mathcal{L}}_\ell - \widetilde{\mathcal{L}}_{\ell-1}]$ and $\mathbb{E}[\widetilde{\mathcal{L}}_\ell - \widetilde{\mathcal{L}}_{\ell-1}]$ 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 Fig. 8; detailed results of MLDA and MCMC chain convergence (Gelman-Rubin statistic) and the recovery of effective samples is described in Text S4 in the supplementary information. 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 the supporting information.

The results of convergence analysis (see section 2.5) are shown in Fig. 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 behaviour, except for the length scale parameter. MLDA results show larger deviations of the length scale mean and variance for smaller sample sizes.

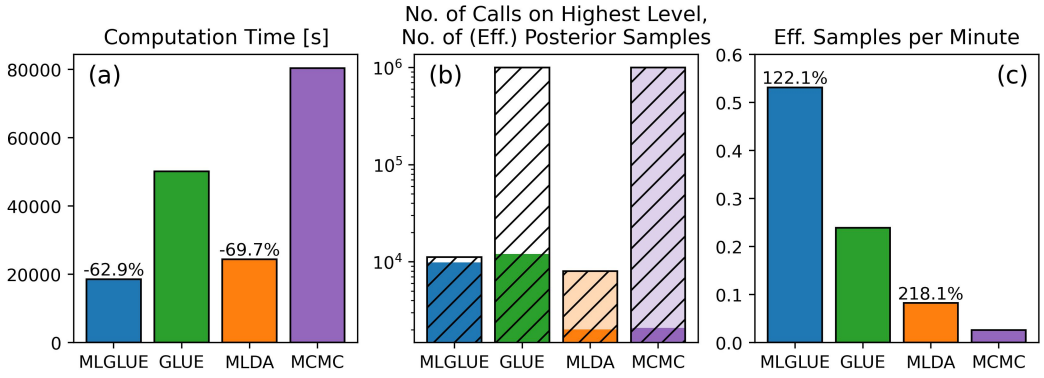


Figure 8. Sampling efficiencies for the groundwater flow example; (a) computation times with percentual reductions compared to conventional methods; (b) No. of model calls on the highest level (dashed), No. of posterior samples (light colors), No. of effective posterior samples (dark colors); (c) No.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

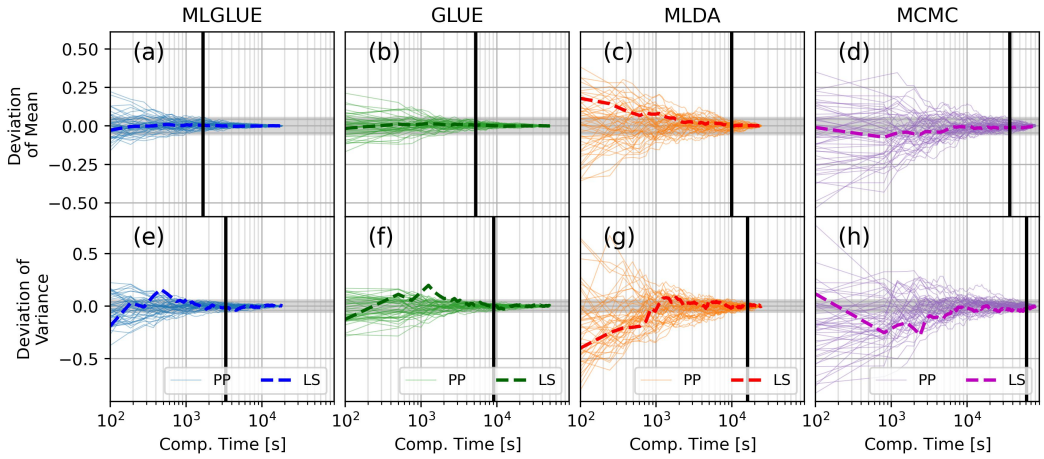


Figure 9. Convergence analysis for the groundwater flow example (Eq. 7); for the different methods of inference (a) - (d) shows the deviation of the mean and (e) - (h) shows the deviation of the variance; grey regions represent the region where convergence is achieved; black vertical lines represent the computational time at which convergence is achieved for all parameters

Estimated CDFs of the parameter posteriors are shown in Fig. 10 (a) - (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 towards 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 towards higher values for MLDA. As evaluated with the coefficient of determination (R^2), MLGLUE results are slightly more accurate compared to GLUE. Similarly, MLDA results are slightly more accurate compared to MCMC.

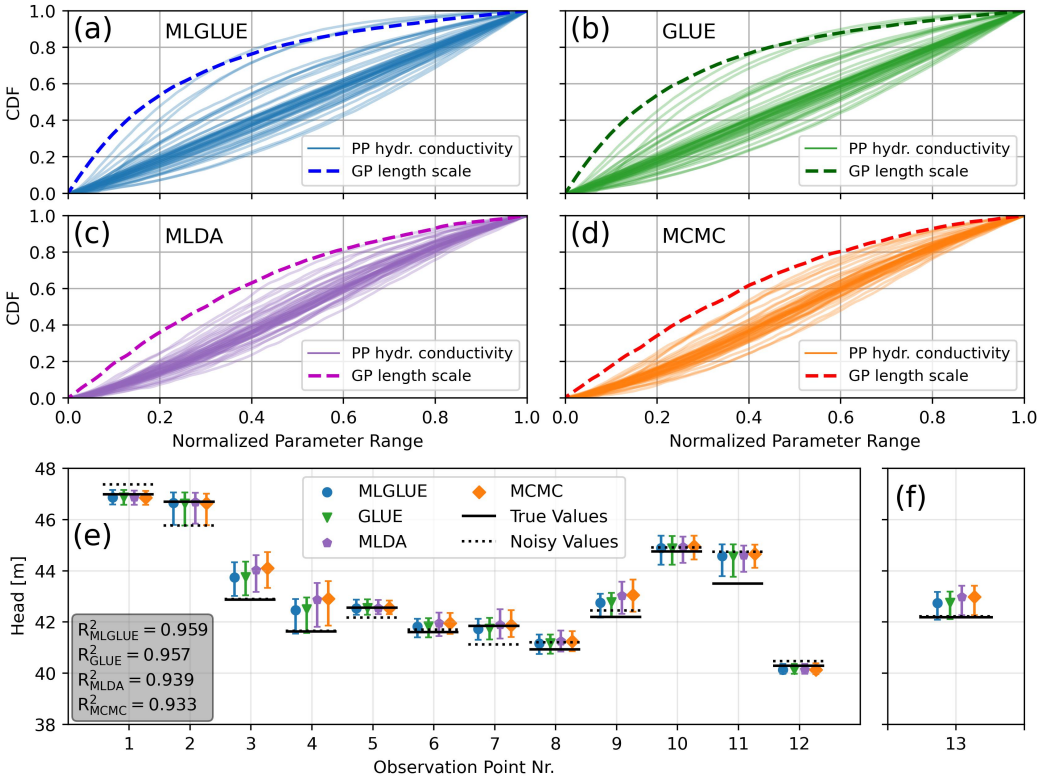


Figure 10. CDFs of model parameters for the groundwater flow example (a, b, c, d) and 99% − 1% uncertainty estimates around the median value for observation points (e) and for the prediction point(f)

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, e.g., 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 modelling example considers a real system.

For both examples it was identified that the number of tuning samples, N_t, required to obtain stable and accurate estimates of likelihood thresholds increases with decreasing threshold percentage although the parameter space dimensions were greatly different ($n = 5$ for rainfall-runoff modelling 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 behaviour 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 modelling 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 ratio. 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 rations. Similar behaviour 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 $\mathbb{V}[\widetilde{\mathcal{L}}_\ell]$ and $\mathbb{V}[\widetilde{\mathcal{L}}_\ell - \widetilde{\mathcal{L}}_{\ell-1}]$ to eventually intersect, resulting in $\mathbb{V}[\widetilde{\mathcal{L}}_\ell - \widetilde{\mathcal{L}}_{\ell-1}] > \mathbb{V}[\widetilde{\mathcal{L}}_\ell]$ for some ℓ. In the context of MLMC (forward problems), this then leads to an increased computational cost compared to conventional MC. As in-

dicated by Eq. 5, if $\mathbb{V}[\widetilde{\mathcal{L}}_\ell]$ decreases and $\mathbb{V}[\widetilde{\mathcal{L}}_\ell - \widetilde{\mathcal{L}}_{\ell-1}]$ increases with decreasing ℓ, then $\mathrm{Cov}(\widetilde{\mathcal{L}}_\ell, \widetilde{\mathcal{L}}_{\ell-1})$ must decrease as well. 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 for MLMC in section 2.2 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 (Vidal-Codina et al., 2015; Giles, 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.

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 behaviour 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 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-fidelity 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 statistical inversion with MLGLUE are compared to the results from GLUE, Markov-chain Monte Carlo (MCMC), as well as multilevel delayed acceptance (MLDA) MCMC. Identical numbers of prior samples are considered for all methods to ensure comparability. We show that the results (parameter posteriors, uncertainty estimates, convergence behaviour) obtained with multilevel approaches (MLGLUE and MLDA) are highly similar to conventional approaches (GLUE and MCMC), respectively. MLGLUE showed the resulted in the lowest computation time and the highest number of posterior samples per minute for both example problems and compared to all other methods of inference.

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; Giles, 2015; Dodwell et al., 2019; 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.

Open Research Section

Relevant resources needed to reproduce the results as well as figures are openly available and can be found under the DOI `10.5281/zenodo.10963983` (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. We gratefully acknowledge the support by the Open Access Publishing Fund of TU Dresden. Open Access funding is enabled and organized by the DEAL project. The authors have no conflict of interest with respect to the results of this study.

References

Allgeier, J. T. (2022). *Analytical and Stochastic Numerical Methods for the Simulation of Subsurface Flow in Floodplains* (Doctoral dissertation, Eberhard Karls Universität Tübingen, Tübingen). Retrieved from `http://dx.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* (Second edition ed.). London ; San Diego, CA: Academic Press. (OCLC: ocn921253555)

Asher, M. J., Croke, B. F. W., Jakeman, A. J., & Peeters, L. J. M. (2015, August). A review of surrogate models and their application to groundwater modeling: SURROGATES OF GROUNDWATER MODELS. *Water Resources Research*, *51*(8), 5957–5973. Retrieved 2022-07-13, from http://doi.wiley.com/10.1002/2015WR016967 doi: 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). Cham: Springer International Publishing. Retrieved 2024-02-21, from https://link.springer.com/10.1007/978-3-319-11259-6_24-1 doi: 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. Retrieved 2023-05-23, from https://linkinghub.elsevier.com/retrieve/pii/030917089390028E doi: 10.1016/0309-1708(93)90028-E

Beven, K. (2006, March). A manifesto for the equifinality thesis. *Journal of Hydrology*, *320*(1-2), 18–36. Retrieved 2022-02-25, from https://linkinghub.elsevier.com/retrieve/pii/S002216940500332X doi: 10.1016/j.jhydrol.2005.07.007

Beven, K. (2016, July). Facets of uncertainty: epistemic uncertainty, non-stationarity, likelihood, hypothesis testing, and communication. *Hydrological Sciences Journal*, *61*(9), 1652–1665. Retrieved 2023-10-04, from http://www.tandfonline.com/doi/full/10.1080/02626667.2015.1031761 doi: 10.1080/02626667.2015.1031761

Beven, K., & Binley, A. (1992, July). The future of distributed models: Model calibration and uncertainty prediction. *Hydrological Processes*, *6*(3), 279–298. Retrieved 2022-05-25, from https://onlinelibrary.wiley.com/doi/10.1002/hyp.3360060305 doi: 10.1002/hyp.3360060305

Beven, K., & Binley, A. (2014, November). GLUE: 20 years on. *Hydrological Processes*, *28*(24), 5897–5918. Retrieved 2023-01-09, from https://onlinelibrary.wiley.com/doi/10.1002/hyp.10082 doi:

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, June). A physically based model of heterogeneous hillslopes: 2. Effective hydraulic conductivities. *Water Resources Research*, *25*(6), 1227–1233. Retrieved 2023-05-26, from http://doi.wiley.com/10.1029/WR025i006p01227 doi: 10.1029/WR025i006p01227

Blasone, R.-S., Vrugt, J. A., Madsen, H., Rosbjerg, D., Robinson, B. A., & Zyvoloski, G. A. (2008, April). Generalized likelihood uncertainty estimation (GLUE) using adaptive Markov Chain Monte Carlo sampling. *Advances in Water Resources*, *31*(4), 630–648. Retrieved 2022-06-16, from https://linkinghub.elsevier.com/retrieve/pii/S0309170807001856 doi: 10.1016/j.advwatres.2007.12.003

Blöschl, G., Bierkens, M. F., Chambel, A., Cudennec, C., Destouni, G., Fiori, A., ... Zhang, Y. (2019, July). Twenty-three unsolved problems in hydrology (UPH) – a community perspective. *Hydrological Sciences Journal*, *64*(10), 1141–1158. Retrieved 2023-10-04, from https://www.tandfonline.com/doi/full/10.1080/02626667.2019.1620507 doi: 10.1080/02626667.2019.1620507

Boyle, D. P. (2001). *Multicriteria calibration of hydrologic models* (Doctoral dissertation, 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). New York, NY: Springer New York. Retrieved 2024-02-21, from http://link.springer.com/10.1007/978-1-4612-4360-1 doi: 10.1007/978-1-4612-4360-1

Brunetti, G., Šimunek, J., Wöhling, T., & Stumpp, C. (2023, September). An in-depth analysis of Markov-Chain Monte Carlo ensemble samplers for inverse vadose zone modeling. *Journal of Hydrology*, *624*, 129822. Retrieved 2023-10-04, from https://linkinghub.elsevier.com/retrieve/pii/S0022169423007643 doi: 10.1016/j.jhydrol.2023.129822

Burrows, W., & Doherty, J. E. (2015, July). Efficient Calibration/Uncertainty Analysis Using Paired Complex/Surrogate Models. *Groundwater*, *53*(4), 531–

541. Retrieved 2022-05-16, from https://onlinelibrary.wiley.com/doi/10.1111/gwat.12257 doi: 10.1111/gwat.12257

Carrera, J., Alcolea, A., Medina, A., Hidalgo, J., & Slooten, L. J. (2005, March). Inverse problem in hydrogeology. *Hydrogeology Journal*, *13*(1), 206–222. Retrieved 2023-05-22, from http://link.springer.com/10.1007/s10040-004-0404-7 doi: 10.1007/s10040-004-0404-7

Christen, J. A., & Fox, C. (2005). Markov Chain Monte Carlo Using an Approximation. *Journal of Computational and Graphical Statistics*, *14*(4). doi: 10.1198/106186005X76983

Cliffe, K. A., Giles, M. B., Scheichl, R., & Teckentrup, A. L. (2011, January). Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients. *Computing and Visualization in Science*, *14*(1), 3–15. Retrieved 2022-11-23, from http://link.springer.com/10.1007/s00791-011-0160-x doi: 10.1007/s00791-011-0160-x

Colecchio, I., Boschan, A., Otero, A. D., & Noetinger, B. (2020, June). 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. Retrieved 2023-05-26, from https://linkinghub.elsevier.com/retrieve/pii/S0309170819310681 doi: 10.1016/j.advwatres.2020.103594

Cui, T., Detommaso, G., & Scheichl, R. (2024, March). Multilevel dimension-independent likelihood-informed MCMC for large-scale inverse problems. *Inverse Problems*, *40*(3), 035005. Retrieved 2024-02-28, from https://iopscience.iop.org/article/10.1088/1361-6420/ad1e2c doi: 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, March). Ground Water Model Calibration Using Pilot Points and Regularization. *Groundwater*, *41*(2), 170–177. Retrieved 2022-07-11, from https://doi.org/10.1111/j.1745-6584.2003.tb02580.x (Publisher: John Wiley & Sons, Ltd) doi: 10.1111/j.1745-6584.2003.tb02580.x

Doherty, J. E. (2015). *Calibration and Uncertainty Analysis for Complex Environ-*

mental Models. Brisbane: Watermark Numerical Computing.

Doherty, J. E., & Christensen, S. (2011, December). Use of paired simple and complex models to reduce predictive bias and quantify uncertainty. *Water Resources Research*, *47*(12). Retrieved 2022-02-28, from http://doi.wiley.com/10.1029/2011WR010763 doi: 10.1029/2011WR010763

Erdal, D., & Cirpka, O. A. (2020, September). Technical Note: Improved sampling of behavioral subsurface flow model parameters using active subspaces. *Hydrology and Earth System Sciences*, *24*(9), 4567–4574. Retrieved 2023-10-04, from https://hess.copernicus.org/articles/24/4567/2020/ doi: 10.5194/hess-24-4567-2020

Gallagher, K., Charvin, K., Nielsen, S., Sambridge, M., & Stephenson, J. (2009, April). 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. Retrieved 2023-10-04, from https://linkinghub.elsevier.com/retrieve/pii/S0264817209000075 doi: 10.1016/j.marpetgeo.2009.01.003

Gauch, M., Kratzert, F., Klotz, D., Nearing, G., Lin, J., & Hochreiter, S. (2020, October). *Data for "Rainfall-Runoff Prediction at Multiple Timescales with a Single Long Short-Term Memory Network".* Zenodo. Retrieved from https://doi.org/10.5281/zenodo.4072701 doi: 10.5281/zenodo.4072701

Gauch, M., Kratzert, F., Klotz, D., Nearing, G., Lin, J., & Hochreiter, S. (2021, April). Rainfall–runoff prediction at multiple timescales with a single Long Short-Term Memory network. *Hydrology and Earth System Sciences*, *25*(4), 2045–2062. Retrieved 2024-02-20, from https://hess.copernicus.org/articles/25/2045/2021/ doi: 10.5194/hess-25-2045-2021

Gelman, A., & Rubin, D. B. (1992, November). Inference from Iterative Simulation Using Multiple Sequences. *Statistical Science*, *7*(4). Retrieved 2022-11-08, from https://projecteuclid.org/journals/statistical-science/volume-7/issue-4/Inference-from-Iterative-Simulation-Using-Multiple-Sequences/10.1214/ss/1177011136.full doi: 10.1214/ss/1177011136

Geyer, C. J. (1992, November). Practical Markov Chain Monte Carlo. *Statistical Science*, *7*(4). Retrieved 2024-04-02, from https://projecteuclid.org/journals/statistical-science/volume-7/issue-4/Practical-Markov

-Chain-Monte-Carlo/10.1214/ss/1177011137.full doi: 10.1214/ss/1177011137

Geyer, C. J. (2011, May). Introduction to Markov Chain Monte Carlo. In *Handbook of Markov Chain Monte Carlo* (1st ed., pp. 3–48). New York: Chapman and Hall/CRC. Retrieved 2024-04-02, from https://www.taylorfrancis.com/books/9780429138508/chapters/10.1201/b10905-2 doi: 10.1201/b10905-2

Giles, M. B. (2008, June). Multilevel Monte Carlo Path Simulation. *Operations Research*, *56*(3), 607–617. Retrieved 2023-05-02, from https://doi.org/10.1287/opre.1070.0496 (Publisher: INFORMS) doi: 10.1287/opre.1070.0496

Giles, M. B. (2015, May). Multilevel Monte Carlo methods. *Acta Numerica*, *24*, 259–328. Retrieved 2022-11-24, from https://www.cambridge.org/core/product/identifier/S096249291500001X/type/journal_article doi: 10.1017/S096249291500001X

Gosses, M., & Wöhling, T. (2019, March). Simplification error analysis for groundwater predictions with reduced order models. *Advances in Water Resources*, *125*, 41–56. Retrieved 2022-02-17, from https://linkinghub.elsevier.com/retrieve/pii/S030917081830575X doi: 10.1016/j.advwatres.2019.01.006

Gosses, M., & Wöhling, T. (2021, September). Robust Data Worth Analysis with Surrogate Models. *Groundwater*, *59*(5), 728–744. Retrieved 2022-05-12, from https://onlinelibrary.wiley.com/doi/10.1111/gwat.13098 doi: 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). Reston, VA: USGS.

Heinrich, S. (2001). Multilevel Monte Carlo Methods. In S. Margenov, J. Waśniewski, & P. Yalamov (Eds.), *Large-Scale Scientific Computing* (pp. 58–67). Berlin, Heidelberg: Springer Berlin Heidelberg.

Herman, J. D., Reed, P. M., & Wagener, T. (2013, March). Time-varying sensitivity analysis clarifies the effects of watershed model formulation on model behavior. *Water Resources Research*, *49*(3), 1400–1414. Retrieved 2023-12-16, from https://doi.org/10.1002/wrcr.20124 (Publisher: John Wiley & Sons, Ltd) doi: 10.1002/wrcr.20124

Herrera, P. A., Marazuela, M. A., & Hofmann, T. (2022, January). Parameter estimation and uncertainty analysis in hydrological modeling. *WIREs Water*, *9*(1), e1569. Retrieved 2023-10-04, from https://wires.onlinelibrary.wiley.com/doi/10.1002/wat2.1569 doi: 10.1002/wat2.1569

Kavetski, D., Kuczera, G., & Franks, S. W. (2006, March). Bayesian analysis of input uncertainty in hydrological modeling: 1. Theory: INPUT UNCERTAINTY IN HYDROLOGY, 1. *Water Resources Research*, *42*(3). Retrieved 2022-11-30, from http://doi.wiley.com/10.1029/2005WR004368 doi: 10.1029/2005WR004368

Kennedy, M. C., & O'Hagan, A. (2001, September). Bayesian Calibration of Computer Models. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, *63*(3), 425–464. Retrieved 2024-03-01, from https://academic.oup.com/jrsssb/article/63/3/425/7083367 doi: 10.1111/1467-9868.00294

Kitanidis, P. K., & Vomvoris, E. G. (1983, June). A geostatistical approach to the inverse problem in groundwater modeling (steady state) and one-dimensional simulations. *Water Resources Research*, *19*(3), 677–690. Retrieved 2023-05-25, from http://doi.wiley.com/10.1029/WR019i003p00677 doi: 10.1029/WR019i003p00677

Knoben, W. J. M., Freer, J., Fowler, K. J. A., Peel, M. C., & Woods, R. A. (2019, June). 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. Retrieved 2024-02-21, from https://gmd.copernicus.org/articles/12/2463/2019/ doi: 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, March). 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. Retrieved 2023-10-04, from https://gmd.copernicus.org/articles/13/1373/2020/ doi: 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. Retrieved from https://doi.org/10.21105/joss.01143 (Publisher: The Open Journal) doi: 10.21105/joss.01143

Kumar, R., Samaniego, L., & Attinger, S. (2013, January). 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. Retrieved 2023-05-23, from http://doi.wiley.com/10.1029/2012WR012195 doi: 10.1029/2012WR012195

Laloy, E., Rogiers, B., Vrugt, J. A., Mallants, D., & Jacques, D. (2013, May). 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. Retrieved 2023-09-15, from http://doi.wiley.com/10.1002/wrcr.20226 doi: 10.1002/wrcr.20226

Laloy, E., & Vrugt, J. A. (2012, January). 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). Retrieved 2022-07-25, from http://doi.wiley.com/10.1029/2011WR010608 doi: 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.* Retrieved from https://levana.leopoldina.org/receive/leopoldina_mods_00591 (ISBN: 978-3-8047-4256-7 Url: https://doi.org/10.26164/leopoldina_03_00590 Url: https://doi.org/10.26164/leopoldina_03_00591) doi: 10.26164/leopoldina_03_00591

Linde, N., Ginsbourger, D., Irving, J., Nobile, F., & Doucet, A. (2017, December). On uncertainty quantification in hydrogeology and hydrogeophysics. *Advances in Water Resources*, *110*, 166–181. Retrieved 2023-05-16, from https://linkinghub.elsevier.com/retrieve/pii/S0309170817304608 doi: 10.1016/j.advwatres.2017.10.014

Liu, J. S. (Ed.). (2008). *Monte Carlo Strategies in Scientific Computing*. New York: Springer.

Liu, Y., Li, J., Sun, S., & Yu, B. (2019, October). Advances in Gaussian random field generation: a review. *Computational Geosciences*, *23*(5), 1011–1047. Retrieved 2022-08-31, from http://link.springer.com/10.1007/s10596-019-09867-y doi: 10.1007/s10596-019-09867-y

Lykkegaard, M. B. (2022, November). *tinyDA*. Retrieved from https://pypi.org/project/tinyda/

Lykkegaard, M. B., & Dodwell, T. J. (2022, June). Where to drill next? A dual-weighted approach to adaptive optimal design of groundwater surveys. *Advances in Water Resources*, *164*, 104219. Retrieved 2023-01-16, from https://linkinghub.elsevier.com/retrieve/pii/S0309170822000914 doi: 10.1016/j.advwatres.2022.104219

Lykkegaard, M. B., Dodwell, T. J., Fox, C., Mingas, G., & Scheichl, R. (2023, March). Multilevel Delayed Acceptance MCMC. *SIAM/ASA Journal on Uncertainty Quantification*, *11*(1), 1–30. Retrieved 2023-03-16, from https://epubs.siam.org/doi/10.1137/22M1476770 doi: 10.1137/22M1476770

Mai, J. (2023, May). Ten strategies towards successful calibration of environmental models. *Journal of Hydrology*, *620*, 129414. Retrieved 2023-06-01, from https://linkinghub.elsevier.com/retrieve/pii/S0022169423003566 doi: 10.1016/j.jhydrol.2023.129414

Mirzaei, M., Huang, Y. F., El-Shafie, A., & Shatirah, A. (2015, July). 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. Retrieved 2023-05-03, from http://link.springer.com/10.1007/s00477-014-1000-6 doi: 10.1007/s00477-014-1000-6

Montanari, A. (2007, March). What do we mean by 'uncertainty'? The need for a consistent wording about uncertainty assessment in hydrology. *Hydrological Processes*, *21*(6), 841–845. Retrieved 2022-11-30, from https://onlinelibrary.wiley.com/doi/10.1002/hyp.6623 doi: 10.1002/hyp.6623

Moore, C., & Doherty, J. E. (2006, April). The cost of uniqueness in groundwater model calibration. *Advances in Water Resources*, *29*(4), 605–623. Re-

trieved 2022-07-13, from https://linkinghub.elsevier.com/retrieve/pii/S0309170805001752 doi: 10.1016/j.advwatres.2005.07.003

Moore, C., Wöhling, T., & Doherty, J. (2010, August). Efficient regularization and uncertainty analysis using a global optimization methodology: REGULARIZATION, UNCERTAINTY AND GLOBAL OPTIMIZATION. *Water Resources Research*, *46*(8). Retrieved 2023-10-16, from http://doi.wiley.com/10.1029/2009WR008627 doi: 10.1029/2009WR008627

Nash, J., & Sutcliffe, J. (1970, April). River flow forecasting through conceptual models part I — A discussion of principles. *Journal of Hydrology*, *10*(3), 282–290. Retrieved from https://www.sciencedirect.com/science/article/pii/0022169470902556 doi: 10.1016/0022-1694(70)90255-6

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, December). Generalized likelihood uncertainty estimation (GLUE) and approximate Bayesian computation: What's the connection?: TECHNICAL NOTE. *Water Resources Research*, *48*(12). Retrieved 2023-05-03, from http://doi.wiley.com/10.1029/2011WR011128 doi: 10.1029/2011WR011128

Page, T., Smith, P., Beven, K., Pianosi, F., Sarrazin, F., Almeida, S., ... Wagener, T. (2023, July). Technical note: The CREDIBLE Uncertainty Estimation (CURE) toolbox: facilitating the communication of epistemic uncertainty. *Hydrology and Earth System Sciences*, *27*(13), 2523–2534. Retrieved 2023-10-04, from https://hess.copernicus.org/articles/27/2523/2023/ doi: 10.5194/hess-27-2523-2023

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

Plumlee, M. (2017, July). Bayesian Calibration of Inexact Computer Models. *Journal of the American Statistical Association*, *112*(519), 1274–1285. Retrieved from https://doi.org/10.1080/01621459.2016.1211016 (Publisher: Taylor & Francis) doi: 10.1080/01621459.2016.1211016

Pokhrel, P., Gupta, H. V., & Wagener, T. (2008, December). A spatial regulariza-

tion approach to parameter estimation for a distributed watershed model. *Water Resources Research*, *44*(12), 2007WR006615. Retrieved 2023-10-16, from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2007WR006615 doi: 10.1029/2007WR006615

Reinecke, R., Wachholz, A., Mehl, S., Foglia, L., Niemann, C., & Döll, P. (2020, May). Importance of Spatial Resolution in Global Groundwater Modeling. *Groundwater*, *58*(3), 363–376. Retrieved 2022-06-08, from https://onlinelibrary.wiley.com/doi/10.1111/gwat.12996 doi: 10.1111/gwat.12996

Rudolph, M. G., Wöhling, T., Wagener, T., & Hartmann, A. (2024, April). *Extending GLUE with multilevel methods to accelerate statistical inversion of hydrological models - code and data.* Zenodo. Retrieved from https://doi.org/10.5281/zenodo.10963983 doi: 10.5281/zenodo.10963983

Sadegh, M., & Vrugt, J. A. (2013, December). Bridging the gap between GLUE and formal statistical approaches: approximate Bayesian computation. *Hydrology and Earth System Sciences*, *17*(12), 4831–4850. Retrieved 2023-05-03, from https://hess.copernicus.org/articles/17/4831/2013/ doi: 10.5194/hess-17-4831-2013

Samaniego, L., Kumar, R., & Attinger, S. (2010, May). Multiscale parameter regionalization of a grid-based hydrologic model at the mesoscale: MULTISCALE PARAMETER REGIONALIZATION. *Water Resources Research*, *46*(5). Retrieved 2023-05-23, from http://doi.wiley.com/10.1029/2008WR007327 doi: 10.1029/2008WR007327

Savage, J. T. S., Pianosi, F., Bates, P., Freer, J., & Wagener, T. (2016, November). 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. Retrieved 2023-10-04, from https://agupubs.onlinelibrary.wiley.com/doi/10.1002/2015WR018198 doi: 10.1002/2015WR018198

Schoups, G., & Vrugt, J. A. (2010, October). 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. Retrieved 2023-01-09, from https://onlinelibrary.wiley

.com/doi/abs/10.1029/2009WR008933 doi: 10.1029/2009WR008933

Team, R. (2022). *Ray.* Retrieved from https://pypi.org/project/ray/2.2.0/

Tonkin, M. J., & Doherty, J. E. (2005, October). A hybrid regularized inversion methodology for highly parameterized environmental models: HYBRID REGULARIZATION METHODOLOGY. *Water Resources Research*, *41*(10). Retrieved 2022-05-16, from http://doi.wiley.com/10.1029/2005WR003995 doi: 10.1029/2005WR003995

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

Trotter, L., Knoben, W. J. M., Fowler, K. J. A., Saft, M., & Peel, M. C. (2022, August). 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. Retrieved 2024-02-21, from https://gmd.copernicus.org/articles/15/6359/2022/ doi: 10.5194/gmd-15-6359-2022

Vidal-Codina, F., Nguyen, N., Giles, M., & Peraire, J. (2015, September). A model and variance reduction method for computing statistical outputs of stochastic elliptic partial differential equations. *Journal of Computational Physics*, *297*, 700–720. Retrieved 2024-02-29, from https://linkinghub.elsevier.com/retrieve/pii/S0021999115003757 doi: 10.1016/j.jcp.2015.05.041

von Gunten, D., Wöhling, T., Haslauer, C., Merchán, D., Causapé, J., & Cirpka, O. A. (2014, November). Efficient calibration of a distributed pde -based hydrological model using grid coarsening. *Journal of Hydrology*, *519*, 3290–3304. Retrieved 2023-03-03, from https://linkinghub.elsevier.com/retrieve/pii/S0022169414008191 doi: 10.1016/j.jhydrol.2014.10.025

Vrugt, J. A. (2016, January). Markov chain Monte Carlo simulation using the DREAM software package: Theory, concepts, and MATLAB implementation. *Environmental Modelling & Software*, *75*, 273–316. Retrieved 2022-07-19, from https://linkinghub.elsevier.com/retrieve/pii/S1364815215300396 doi: 10.1016/j.envsoft.2015.08.013

Vrugt, J. A., & Beven, K. J. (2018, April). Embracing equifinality with efficiency: Limits of Acceptability sampling using the DREAM(LOA) algo-

rithm. *Journal of Hydrology*, *559*, 954–971. Retrieved 2023-10-04, from https://linkinghub.elsevier.com/retrieve/pii/S0022169418301021 doi: 10.1016/j.jhydrol.2018.02.026

Vrugt, J. A., Diks, C. G. H., Gupta, H. V., Bouten, W., & Verstraten, J. M. (2005, January). Improved treatment of uncertainty in hydrologic modeling: Combining the strengths of global optimization and data assimilation. *Water Resources Research*, *41*(1), 2004WR003059. Retrieved 2024-02-14, from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2004WR003059 doi: 10.1029/2004WR003059

Vrugt, J. A., Gupta, H. V., Bouten, W., & Sorooshian, S. (2003, August). 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). Retrieved 2023-03-03, from http://doi.wiley.com/10.1029/2002WR001642 doi: 10.1029/2002WR001642

Vrugt, J. A., ter Braak, C. J. F., Clark, M. P., Hyman, J. M., & Robinson, B. A. (2008, December). Treatment of input uncertainty in hydrologic modeling: Doing hydrology backward with Markov chain Monte Carlo simulation. *Water Resources Research*, *44*(12). Retrieved 2022-07-25, from http://doi.wiley.com/10.1029/2007WR006720 doi: 10.1029/2007WR006720

Vrugt, J. A., ter Braak, C. J. F., Gupta, H. V., & Robinson, B. A. (2009, October). Equifinality of formal (DREAM) and informal (GLUE) Bayesian approaches in hydrologic modeling? *Stochastic Environmental Research and Risk Assessment*, *23*(7), 1011–1026. Retrieved 2022-03-07, from http://link.springer.com/10.1007/s00477-008-0274-y doi: 10.1007/s00477-008-0274-y

Wagener, T., Boyle, D. P., Lees, M. J., Wheater, H. S., Gupta, H. V., & Sorooshian, S. (2001, March). A framework for development and application of hydrological models. *Hydrology and Earth System Sciences*, *5*(1), 13–26. Retrieved 2024-01-03, from https://hess.copernicus.org/articles/5/13/2001/ doi: 10.5194/hess-5-13-2001

Wagener, T., & Gupta, H. V. (2005, December). Model identification for hydrological forecasting under uncertainty. *Stochastic Environmental Re-*

search and Risk Assessment, *19*(6), 378–387. Retrieved 2023-05-08, from http://link.springer.com/10.1007/s00477-005-0006-5 doi: 10.1007/s00477-005-0006-5

White, J. T. (2018, November). A model-independent iterative ensemble smoother for efficient history-matching and uncertainty quantification in very high dimensions. *Environmental Modelling & Software*, *109*, 191–201. Retrieved from https://www.sciencedirect.com/science/article/pii/S1364815218302676 doi: 10.1016/j.envsoft.2018.06.009

White, J. T., Hunt, R. J., Fienen, M. N., & Doherty, J. E. (2020). *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. Retrieved from http://pubs.er.usgs.gov/publication/tm7C26 doi: 10.3133/tm7C26

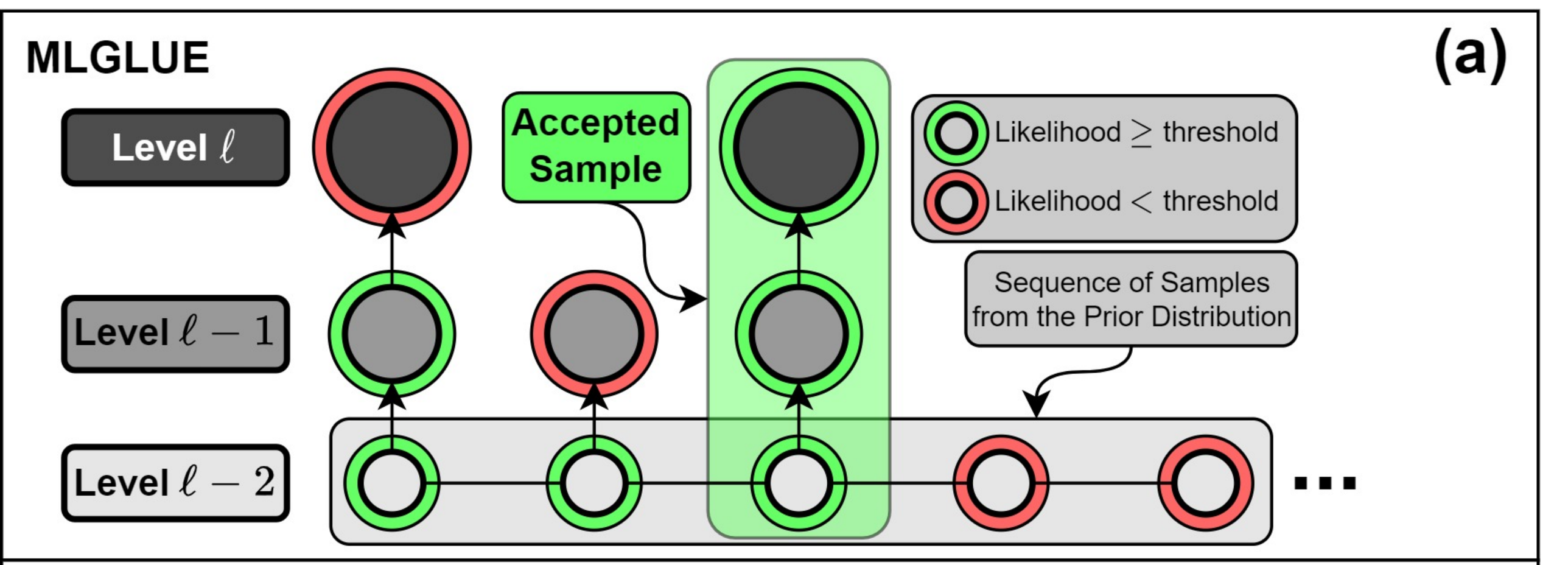
White, J. T., Knowling, M. J., & Moore, C. R. (2020, September). Consequences of Groundwater-Model Vertical Discretization in Risk-Based Decision-Making. *Groundwater*, *58*(5), 695–709. Retrieved 2022-07-06, from https://doi.org/10.1111/gwat.12957 (Publisher: John Wiley & Sons, Ltd) doi: 10.1111/gwat.12957

Wildemeersch, S., Goderniaux, P., Orban, P., Brouyère, S., & Dassargues, A. (2014, March). Assessing the effects of spatial discretization on large-scale flow model performance and prediction uncertainty. *Journal of Hydrology*, *510*, 10–25. Retrieved 2023-03-17, from https://linkinghub.elsevier.com/retrieve/pii/S0022169413009177 doi: 10.1016/j.jhydrol.2013.12.020

Zhou, H., Gómez-Hernández, J. J., & Li, L. (2014, January). Inverse methods in hydrogeology: Evolution and recent trends. *Advances in Water Resources*, *63*, 22–37. Retrieved 2022-04-22, from https://linkinghub.elsevier.com/retrieve/pii/S0309170813002017 doi: 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., ... Rubin, Y. (1998, June). 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. Retrieved 2022-05-25, from http://doi.wiley.com/10.1029/98WR00003 doi: 10.1029/98WR00003

Figure 1.


MLGLUE
(a)
Level ℓ
Level $\ell - 1$
Level $\ell - 2$
Accepted Sample
Likelihood $\geq$ threshold
Likelihood $<$ threshold
Sequence of Samples from the Prior Distribution
...

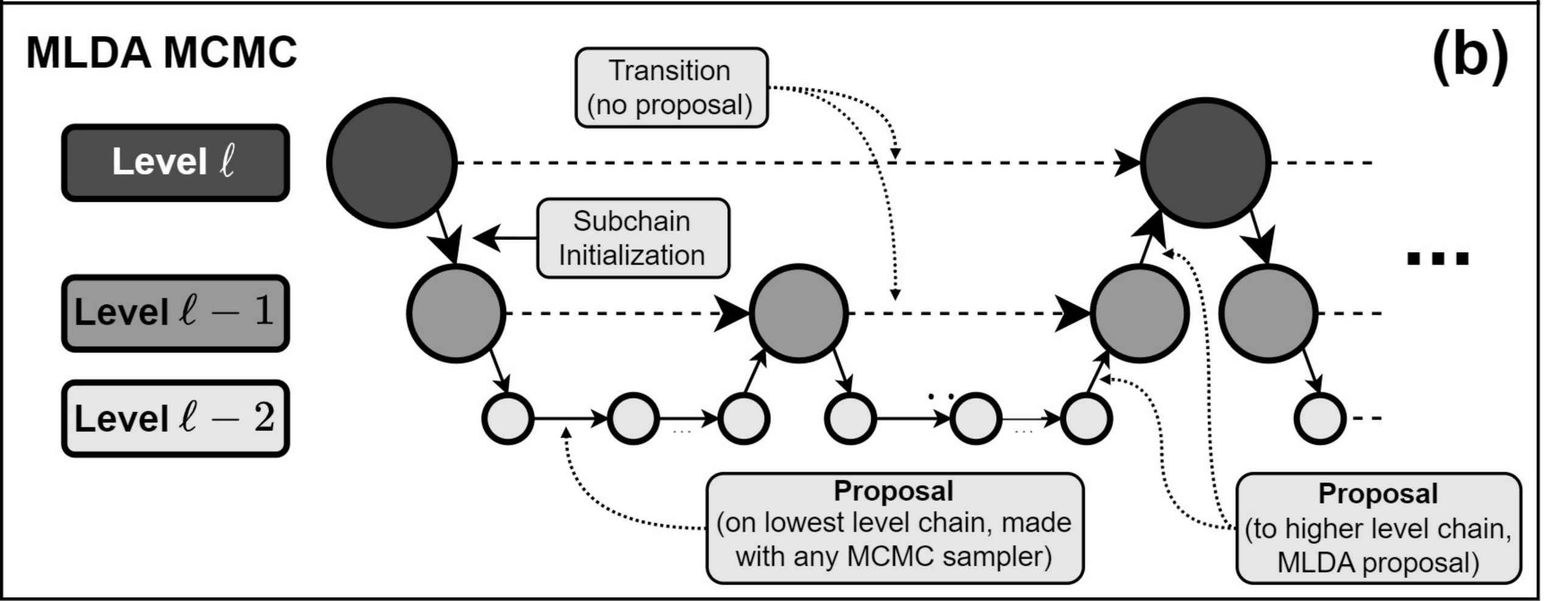

MLDA MCMC
(b)
Level ℓ
Level $\ell - 1$
Level $\ell - 2$
Transition (no proposal)
Subchain Initialization
...
Proposal (on lowest level chain, made with any MCMC sampler)
Proposal (to higher level chain, MLDA proposal)

Figure 2.

MLGLUE Tuning

Initialize sample index $(j = 0)$

Initialize level index $(\ell = 0)$

Compute and store likelihood on level ℓ

Highest level?

No

increment ℓ

Last sample?

No

increment j

Yes

Compute and store likelihood thresholds $\{\widetilde{\mathcal{L}}_{T,\ell}\}_{\ell=0}^{L}$

START MLGLUE

Draw Prior Samples

Tuning

Analyze Relations Between Levels

Sampling

Compute Uncerainty Estimates

END

MLGLUE Sampling

Initialize sample index $(j = 0)$

Initialize level index $(\ell = 0)$

Compute $\widetilde{\mathcal{L}}$ on level ℓ with sample j

Likelihood above threshold?

No

increment j

Yes

Highest level?

No

increment ℓ

Yes

Store parameter sample

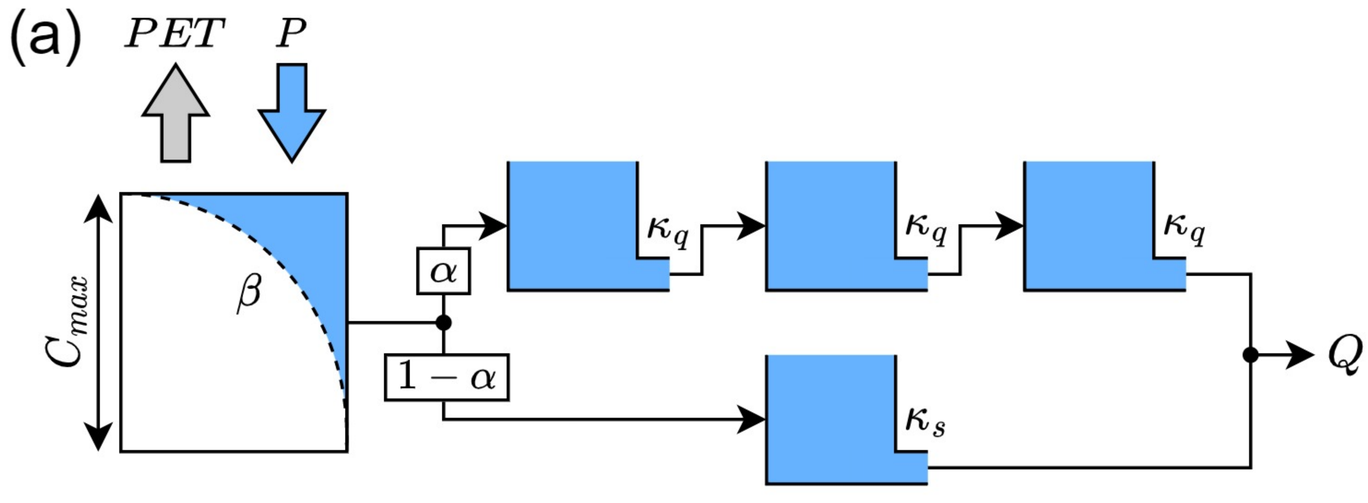
Last sample?

No

increment j

Yes

Figure 3.


(a)
PET
P
C_{max}
β
α
$1-\alpha$
κ_q
κ_q
κ_q
κ_s
Q

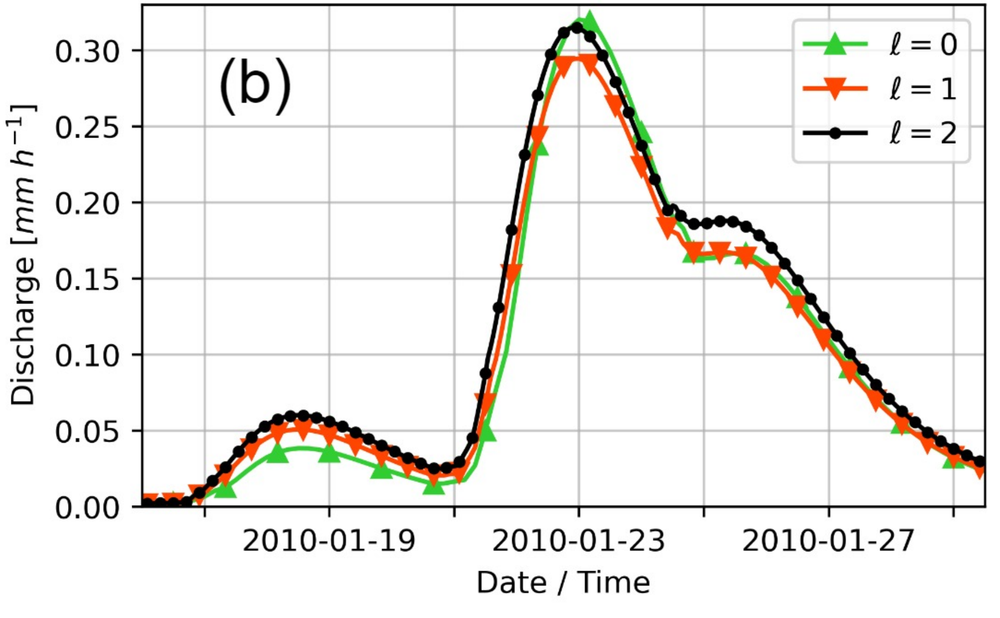

(b)
$\ell = 0$
$\ell = 1$
$\ell = 2$
Discharge [$mm\ h^{-1}$]
0.00
0.05
0.10
0.15
0.20
0.25
0.30
2010-01-19
2010-01-23
2010-01-27
Date / Time

Figure 4.

(a)

(b)

(c)

(d)

y [m]

1000.0

3000.0

5000.0

0.0

2000.0

4000.0

6000.0

8000.0

10000.0

x [m]

48

46

44

42

40

head [m]

10^0

h. hydraulic conductivity [md^{-1}]

Figure 5.

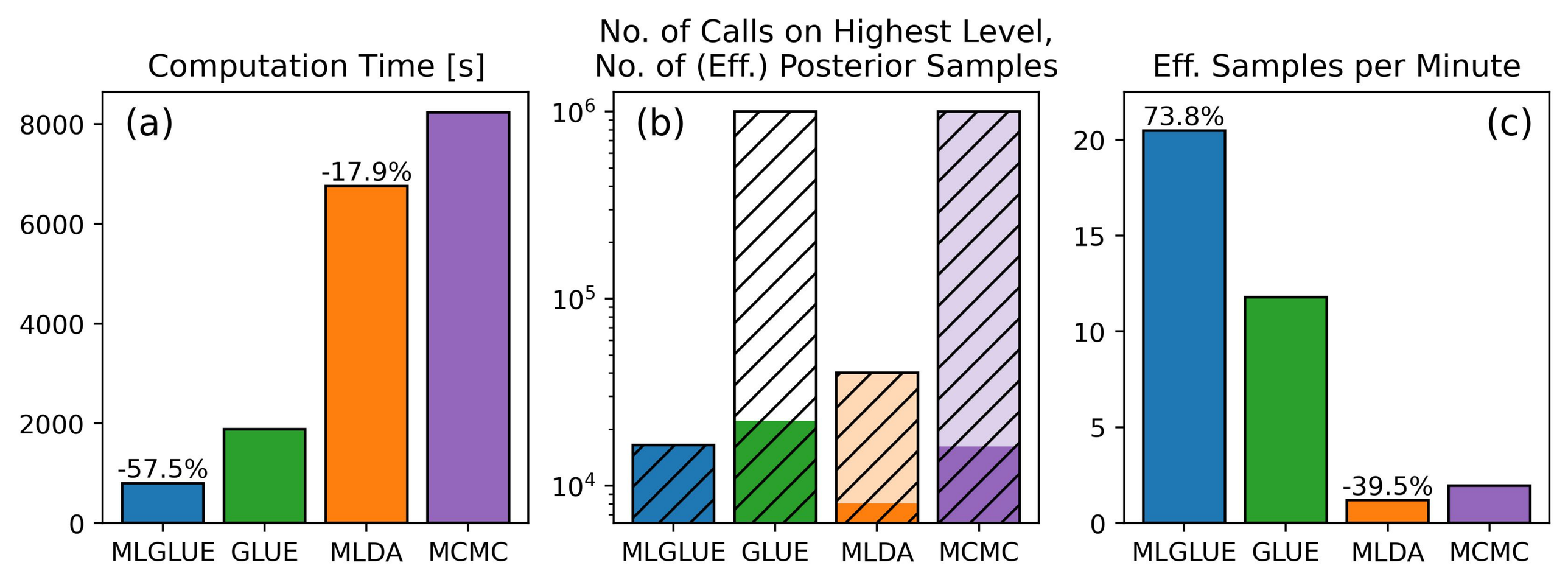

Computation Time [s]
(a)
8000
6000
4000
2000
0
-57.5%
-17.9%
MLGLUE
GLUE
MLDA
MCMC
No. of Calls on Highest Level,
No. of (Eff.) Posterior Samples
(b)
10^6
10^5
10^4
MLGLUE
GLUE
MLDA
MCMC
Eff. Samples per Minute
(c)
73.8%
-39.5%
20
15
10
5
0
MLGLUE
GLUE
MLDA
MCMC

Figure 6.

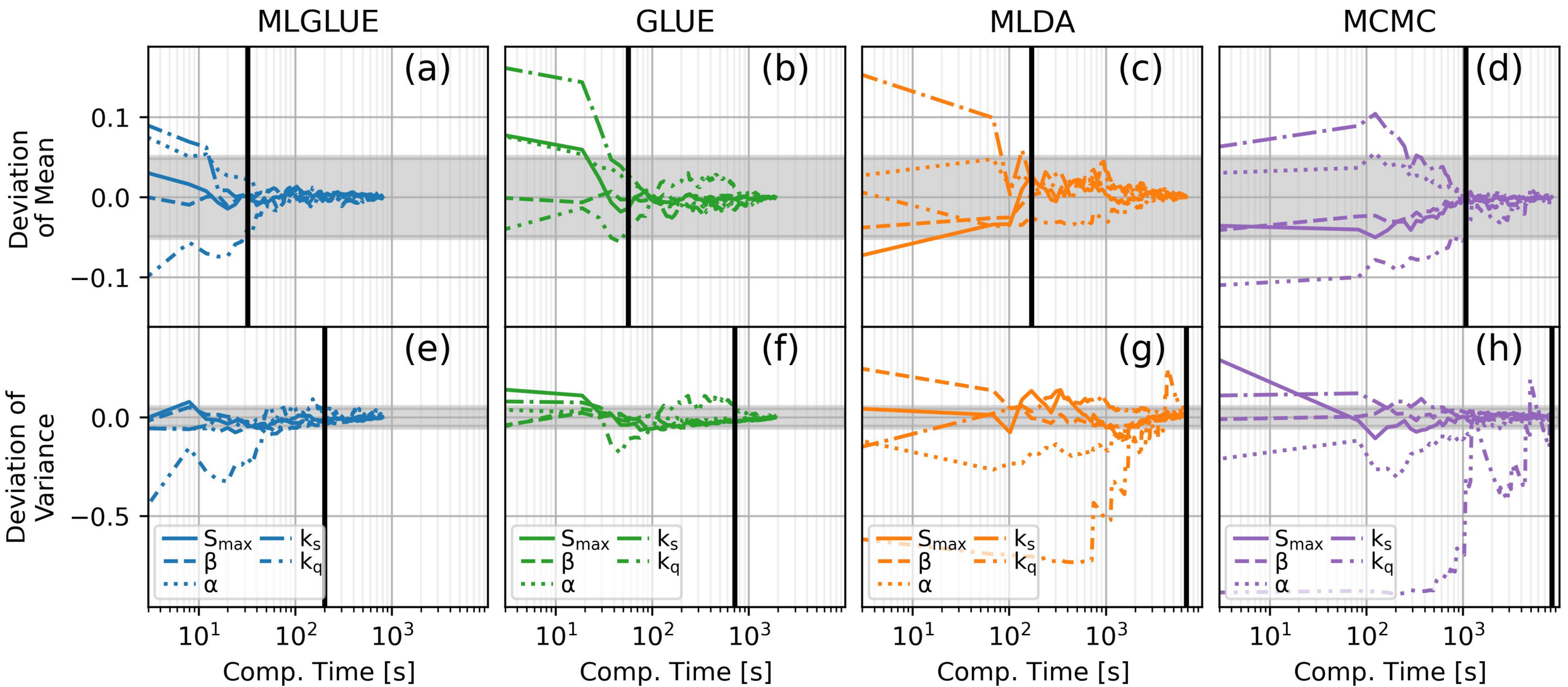

MLGLUE
GLUE
MLDA
MCMC
(a)
(b)
(c)
(d)
(e)
(f)
(g)
(h)
Deviation of Mean
Deviation of Variance
0.1
0.0
−0.1
0.0
−0.5
S_{max}
k_s
β
k_q
α
10^1
10^2
10^3
Comp. Time [s]

Figure 7.

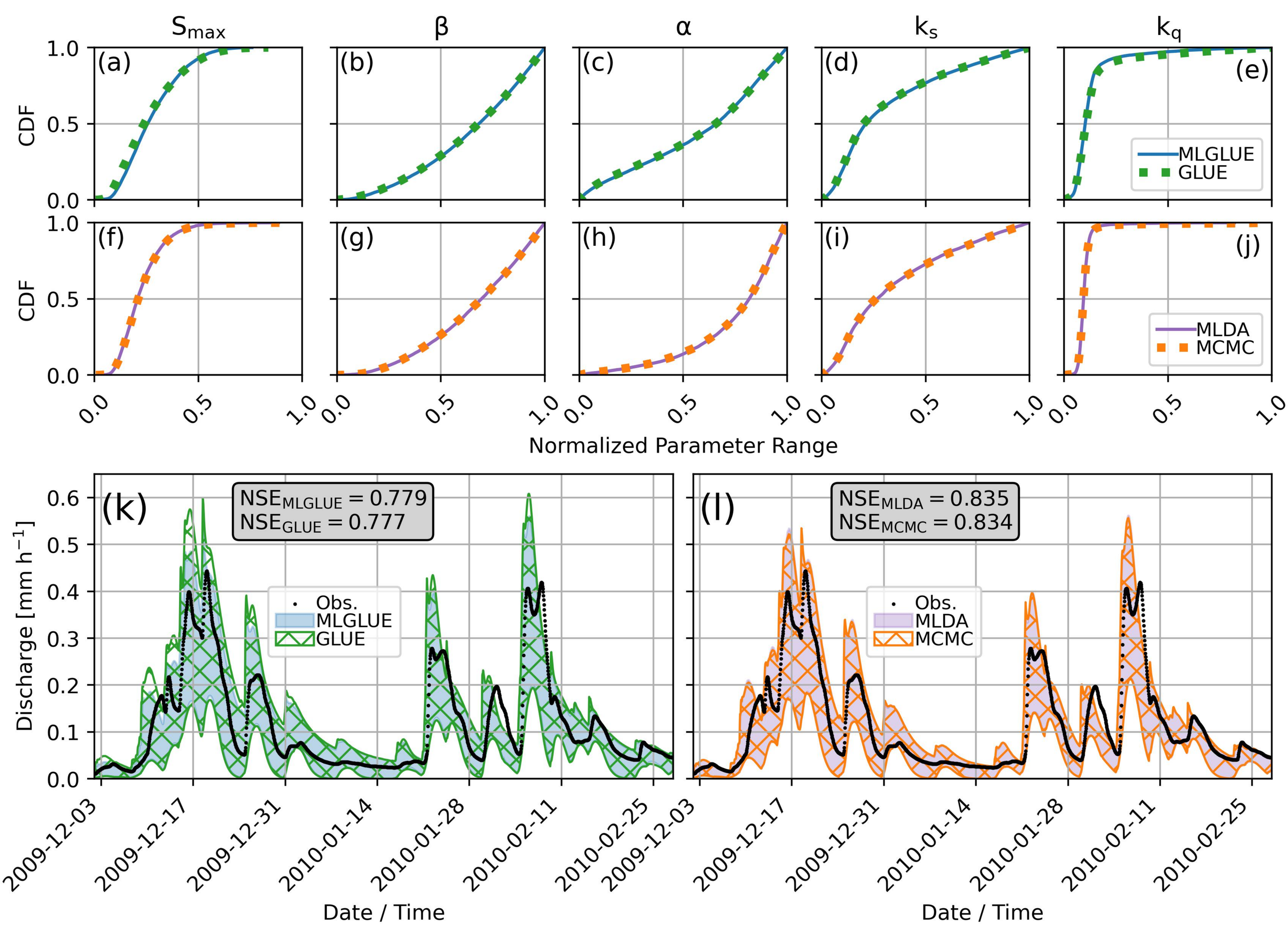

S_{max}
β
α
k_s
k_q
(a)
(b)
(c)
(d)
(e)
(f)
(g)
(h)
(i)
(j)
CDF
1.0
0.5
0.0
MLGLUE
GLUE
MLDA
MCMC
0.0
0.5
1.0
Normalized Parameter Range
(k)
(l)
$NSE_{MLGLUE} = 0.779$
$NSE_{GLUE} = 0.777$
$NSE_{MLDA} = 0.835$
$NSE_{MCMC} = 0.834$
Obs.
MLGLUE
GLUE
MLDA
MCMC
Discharge [mm h^{-1}]
0.6
0.5
0.4
0.3
0.2
0.1
0.0
2009-12-03
2009-12-17
2009-12-31
2010-01-14
2010-01-28
2010-02-11
2010-02-25
Date / Time

Figure 8.

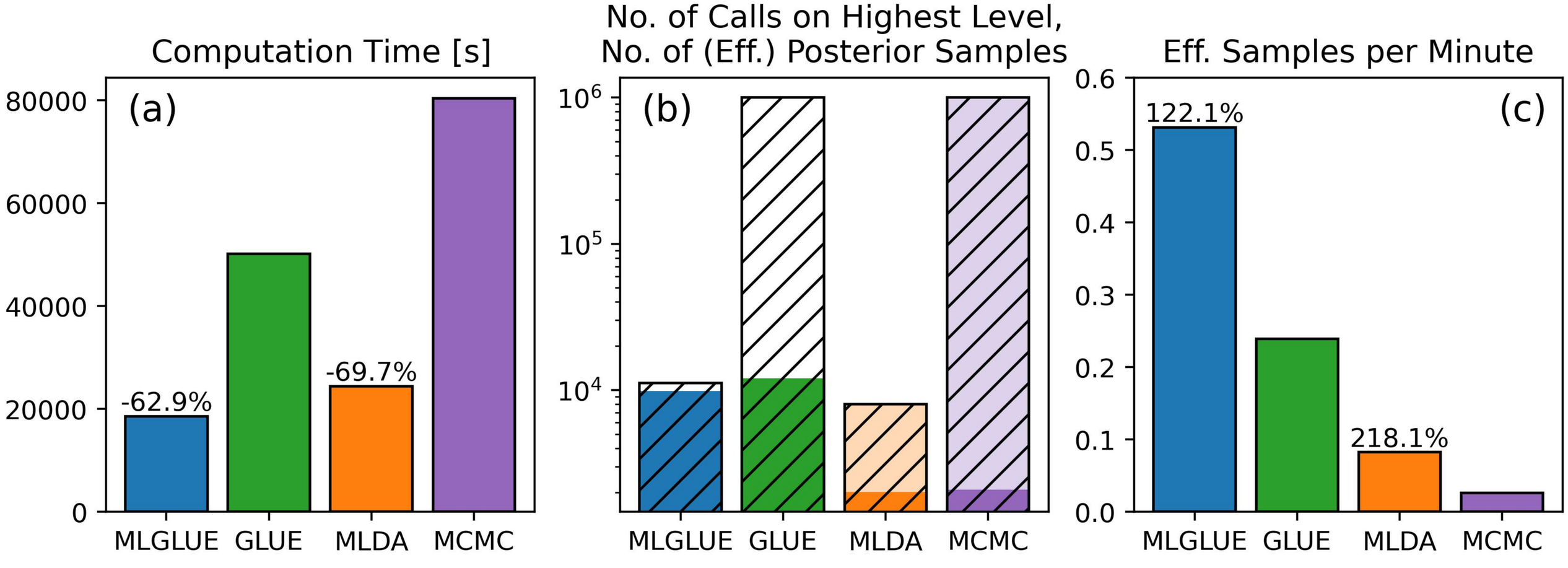

Computation Time [s]
(a)
80000
60000
40000
20000
0
-62.9%
-69.7%
MLGLUE
GLUE
MLDA
MCMC
No. of Calls on Highest Level,
No. of (Eff.) Posterior Samples
(b)
10^6
10^5
10^4
MLGLUE
GLUE
MLDA
MCMC
Eff. Samples per Minute
(c)
0.6
0.5
0.4
0.3
0.2
0.1
0.0
122.1%
218.1%
MLGLUE
GLUE
MLDA
MCMC

Figure 9.

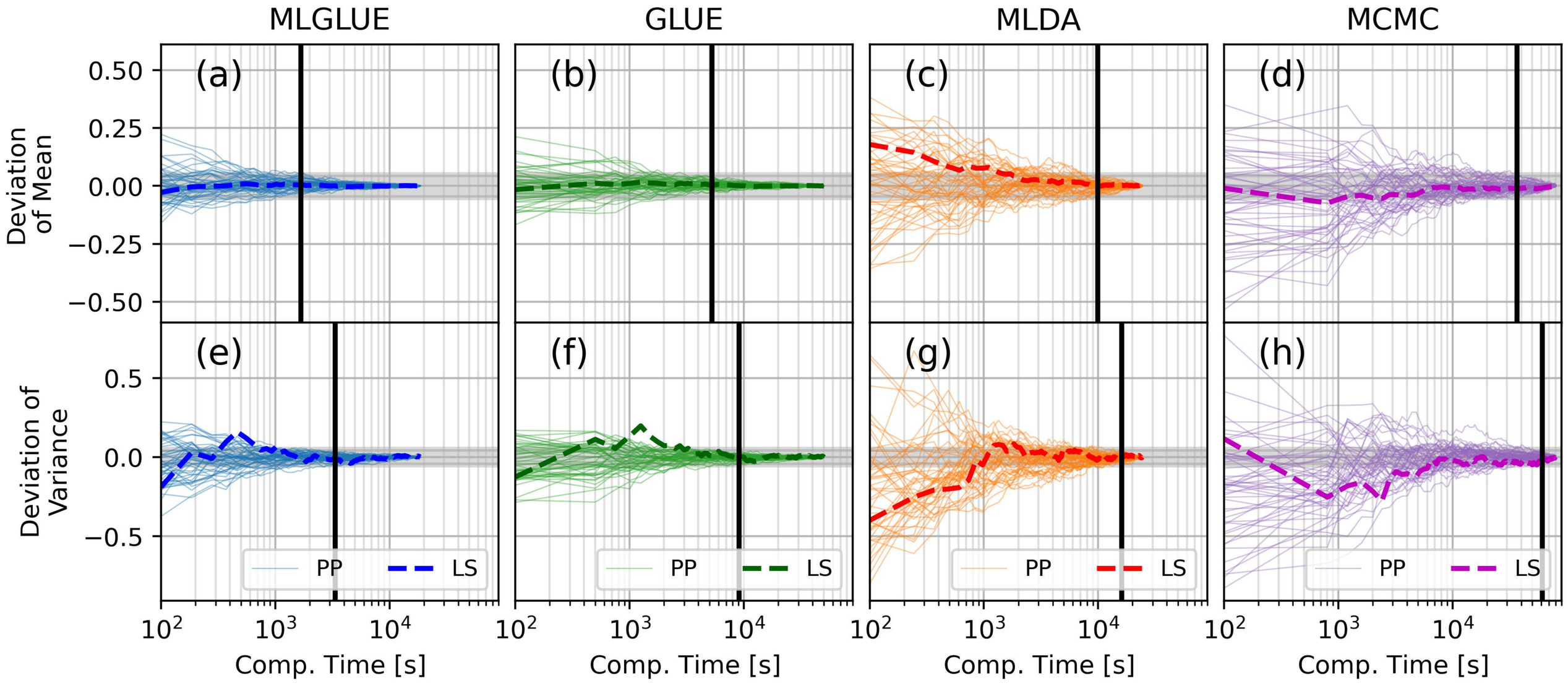

MLGLUE
GLUE
MLDA
MCMC
(a)
(b)
(c)
(d)
(e)
(f)
(g)
(h)
Deviation of Mean
Deviation of Variance
0.50
0.25
0.00
−0.25
−0.50
0.5
0.0
−0.5
PP
LS
10^2
10^3
10^4
Comp. Time [s]

Figure 10.

(a)
MLGLUE
(b)
GLUE
(c)
MLDA
(d)
MCMC
CDF
PP hydr. conductivity
GP length scale
0.0
0.2
0.4
0.6
0.8
1.0
Normalized Parameter Range
(e)
(f)
MLGLUE
GLUE
MLDA
MCMC
True Values
Noisy Values
Head [m]
Observation Point Nr.
$R^2_{MLGLUE} = 0.959$
$R^2_{GLUE} = 0.957$
$R^2_{MLDA} = 0.939$
$R^2_{MCMC} = 0.933$