AGU ADVANCING EARTH AND SPACE SCIENCE

Water Resources Research

RESEARCH ARTICLE
10.1029/2019WR026777

Key Points:
- We use iterated batch importance sampling, a blend between a particle filter and MCMC, to optimize groundwater model parameters
- Consistency with a prescribed geology is maintained through multipoint geostatistics, adjusting uncertain boundaries to reduce bias
- We test the resulting quasi-online optimizer with a field site in Northern Italy

Supporting Information:
- Supporting Information S1

Correspondence to:
M. Ramgraber,
max.ramgraber@eawag.ch

Citation:
Ramgraber, M., Camporese, M., Renard, P., Salandin, P., & Schirmer, M. (2020). Quasi-online groundwater model optimization under constraints of geological consistency based on iterative importance sampling. *Water Resources Research*, *56*, e2019WR026777. https://doi.org/10.1029/2019WR026777

Received 19 NOV 2019
Accepted 8 APR 2020
Accepted article online 14 APR 2020

Quasi-Online Groundwater Model Optimization Under Constraints of Geological Consistency Based on Iterative Importance Sampling

Maximilian Ramgraber[1,2], Matteo Camporese[3], Philippe Renard[2], Paolo Salandin[3], and Mario Schirmer[1,2]

[1]Department of Water Resources and Drinking Water, Swiss Federal Institute of Aquatic Science and Technology (Eawag), Dübendorf, Switzerland, [2]Centre for Hydrogeology and Geothermics (CHYN), University of Neuchâtel, Neuchâtel, Switzerland, [3]Department of Civil, Environmental and Architectural Engineering, University of Padova, Padua, Italy

Abstract The increasing use of wireless sensor networks and remote sensing permits real-time access to environmental observations. Data assimilation frameworks tap into such data streams to autonomously update and gradually improve numerical models. In hydrogeology, such methods are relevant in areas of long-term interest in water quality and quantity, for example, in drinking water production. Unfortunately, accurate hydrogeological predictions often demand a degree of geological realism, which is difficult to reconcile with the operational limitations of many data assimilation frameworks. Alluvial aquifers, for example, are sometimes characterized by paleo-channels of unknown extent and properties, which may act as preferential flow paths. Gradually optimizing such fields in real-time or quasi-real-time settings is a formidable task. Besides subsurface properties, ill-specified model forcings are a further source of predictive bias, which an optimizer could learn to compensate. In this study, we explore the use of a quasi-online optimizer based on the iterative batch importance sampling framework for a groundwater model of a field site near Valdobbiadene, Italy. This site is characterized by the presence of paleo-channels and heavily exploited for drinking water production and irrigation. We use Markov chain Monte Carlo steps to explore new parameterizations while maintaining consistency between states and parameters as well as conformance to a multipoint statistics training image. We also optimize a preprocessor designed to compensate for potential bias in the model forcing. We achieve promising and geologically consistent quasi-real-time optimization, albeit at the loss of parameter uncertainty.

1. Introduction

Groundwater is a critical resource for sustainable human and ecological development (Burri et al., 2019; Schirmer et al., 2013), constituting the dominant source of drinking water and irrigation in many countries across the globe (e.g., Gorelick & Zheng, 2015). As such, safeguarding this resource is a long-term endeavor across multiple time scales, ranging from months/years (e.g., dewatering of construction sites; Powers et al., 2007) to decades (e.g., drinking water production and aquifer remediation; Kresic & Stevanovic, 2010; UNICEF, 2016) or even centuries (e.g., prevention of saltwater intrusion; Oude Essink, 2001; Werner et al., 2013).

Numerical modeling plays a crucial role in informing such hydrogeological practices (Reilly & Harbaugh, 2004). The parameterization of groundwater models demands a full characterization of subsurface properties, information which can only partially be obtained from direct measurements. Consequently, modelers often find themselves tasked with the synthesis of plausible parameter fields from different sources of information (McLaughlin & Townley, 1996; Yeh, 1986) such as geological field characterizations (e.g., Linde et al., 2015; Zovi et al., 2017), geophysical measurements (e.g., Zovi et al., 2017), or parameter-dependent state observations (e.g., Doherty et al., 2010; Sun, 1994).

As the most accessible data type in inverse modeling, state observations can provide a wealth of information. Unfortunately, depending on climatic conditions, they might reveal only certain aspects of the subsurface at any given time. As such, the process of assembling a sufficiently informative data set may be slow. This motivates the use of online model optimization routines based on data assimilation (e.g., Hendricks Franssen & Kinzelbach, 2008). While the adoption of these methods into practice is still in its infancy, such algorithms

could connect to online sensor networks and assimilate data as it becomes available, sequentially optimizing a model's state and parameter estimates. In hydrogeology, certainly the most popular among such techniques is the *ensemble Kalman filter* (*EnKF*; Evensen, 1994, 2003). This method has seen a great rise in popularity over the past decades (e.g., Aanonsen et al., 2009; Hendricks Franssen & Kinzelbach, 2008; Reichle et al., 2002; Tang et al., 2015, 2017; Zhou et al., 2011) owed to its simplicity, relative ease of implementation, great computational efficiency, and remarkable robustness to both small ensemble sizes and violations of its implicit assumptions of *linearity* and *Gaussianity*.

However, the price for the EnKF's elegance is that it updates its variables *as if* its fundamental assumptions were met (e.g., Katzfuss et al., 2016), which is rarely—if ever—the case in hydrogeology. Particularly complex geological priors often deviate substantially from Gaussianity (Aanonsen et al., 2009; Sun et al., 2009). As such, the common practice of optimizing log-conductivity fields sampled from such priors with the EnKF (e.g., Jafarpour & McLaughlin, 2009; Tang et al., 2015, 2017; Zhou et al., 2011; Zovi et al., 2017) risks leaving the support of the prior. This, in turn, means that the EnKF eventually erases geological features present in the initial ensemble (Ramgraber et al., 2019; Zovi et al., 2017) and yields posterior samples incompatible with the prior. Attempts to enforce conformance by construction (e.g., Hu et al., 2013) circumvent this issue, but instead often suffer from a weakened linear relation between parameter changes and state response, exploited by the EnKF's parameter update (e.g., Crestani et al., 2013).

In order to achieve a good fit to hydraulic heads, practitioners often neglect geological realism or structural uncertainty in favor of simpler formulations such as a priori zonation with homogeneous properties or interpolation from a set of pilot points (Cirpka & Valocchi, 2016). While this may prove adequate for the prediction of flow only, using a model for transport-related quantities (e.g., flow paths, travel times, and reactive transport) demands a more faithful representation of the geology (Alcolea & Renard, 2010; Cirpka & Valocchi, 2016; Fogg & Zhang, 2016; Sanchez-Vila & Fernàndez-Garcia, 2016).

To reconcile the challenges of geological realism with the operational limitations of sequential optimization frameworks, it seems auspicious to return to more general sequential Monte Carlo (SMC) techniques like particle filters (e.g., Doucet & Johansen, 2009; Doucet & Tadić, 2003; Peter Jan van Leeuwen, 2009; van Leeuwen et al., 2019). These methods promise greater freedom in exploring complex probability distributions with nonlinear relations between parameters and states. The price for this flexibility is often drastically lower efficiency: classic particle filters demand an ensemble size exponential with regard to the dimensionality of the system (e.g., Snyder et al., 2015), a restriction known as the *curse of dimensionality*. Failing to provide a sufficiently large ensemble—a virtual inevitability given the high dimensionality and computational cost of most subsurface models—results in *sample degeneracy* (e.g., Doucet & Johansen, 2009; Li et al., 2015) and eventually the collapse of the particle approximation. While this degeneration of the particle approximation often cannot be avoided in practice, it can still provide a powerful basis for model optimization.

In this study, we construct a quasi-online optimizer based on *iterated batch importance sampling* (*IBIS*; Chopin, 2002; Chopin et al., 2013), a particle filter that uses Markov chain Monte Carlo (MCMC) steps to counteract sample degeneracy (*rejuvenation*) and optimizes the ensemble in the process. While MCMC steps are a common rejuvenation mechanism in hydrological particle filters (e.g., Moradkhani et al., 2012; Noh et al., 2011; Vrugt et al., 2013), many methods simplify the point-wise evaluations of the posterior probability density it requires by using intermediate density estimates from the particle approximation. In the IBIS algorithm—similar to the Restart EnKF (Gu & Oliver, 2007)—the full observation history is resimulated instead and the posterior density is computed directly. This guarantees that states and parameters are always internally physically consistent and renders the fidelity of the rejuvenation mechanism largely independent of the (possibly degenerate) particle approximation, at the cost of steadily increasing computational demand. In our study, we compensate this effect by dynamically adjusting the ensemble size and use the flexibility of the MCMC framework to sequentially optimize a model under a complex geological prior, maintaining conformance by construction through a combination of hyperparameterization and multipoint statistics (*MPS*; Caers et al., 2003; Journel & Zhang, 2006). The optimizer is tested at a field site in northern Italy characterized by paleo-channels, the object of a previous study employing the EnKF (Zovi et al., 2017). We implement the algorithm in three different scenarios and compare the results obtained to the previous study.

2. Theory

2.1. Nomenclature

In probabilistic systems model variables are separated into two classes. The *parameters* θ are usually static model variables, such as hydraulic conductivities or specific yield, and generally independent from other variables. *States*, denoted by x, are typically time-varying quantities, which depend on parameters or model forcing. Hydraulic heads, temperatures, or concentrations all are common examples. The *observations* y—generally measurements of states—are treated as a third, separate variable type.

All system variables of the same type are combined into a vector and interpreted as coordinates of a point in high-dimensional variable space (*parameter space*, *state space*, and *observation space*, respectively). *Particles* occupy one such point and thus represent a full set of the respective variable type required by the model. We assign a superscript index in brackets to individual particles and their associated variables, for example, $x^{(\text{index})}$. Time-dependent variables are designated by subscripts x_{time} for specific time points, and time spans between a start and end point are represented by $x_{\text{start}:\text{end}}$. A "~" should be read as "sampled from," and a semicolon ";" denotes "parameterized by." Figures and tables numbered with a leading "S" refer to material in the supporting information. A list of all variables is provided in Table S1 in the supporting information.

2.2. Sequential Bayesian Inference

At the heart of Bayesian parameter estimation lies the inference of the posterior *probability density function* (pdf) $p(\theta|y_{1:t})$, which can be determined to proportionality through the prior $p(\theta)$ and the likelihood of the observation time series conditional on the parameters $p(y_{1:t}|\theta)$:

$$p(\theta|y_{1:t}) \propto p(\theta)p(y_{1:t}|\theta) \tag{1}$$

Sequential data assimilation frameworks incorporate data in increments. Assuming time-independent likelihoods, we can reformulate equation 1 to obtain

$$p(\theta|y_{1:t}) \propto p(\theta)\prod_{s=1}^{t} p(y_s|\theta) \tag{2}$$

While $p(\theta)$ is generally user-prescribed, the likelihood $p(y_s|\theta)$ is not always straightforward to obtain. Most EnKF variations and particle filters based on state-vector augmentation (e.g., Moradkhani et al., 2005) instead use the observational likelihood $p(y_s|x_s)$, then extend the Bayesian update to the parameters via the parameter-dependent states. Nested particle filters (e.g., SMC^2; Chopin et al., 2013) also use the observational likelihood, but then integrate over the state space to obtain $p(y_s|\theta)$.

An alternative approach is to omit the model states entirely from the probabilistic part of the inference process. Since most numerical groundwater models $M(x_0, u_{1:t}, \theta)$ are deterministic to begin with, one may interpret the model states $x_{1:t}$ as the output of a deterministic mapping from parameter space, the initial states x_0, and the model forcing $u_{1:t}$. If we further assume that x_0 and $u_{1:t}$ depend only on θ and constants, this dependency effectively reduces to θ, and we can consider x_0 and $u_{1:t}$ intermediate results of the deterministic map from θ to $x_{1:t}$. The states $x_{1:t}$, in turn, map deterministically to observation space, providing us with a deterministic map of parameters to observations by "bridging across" state space. This process is often called a *forward operator* (also *forward solver* and *forward map*; Linde et al., 2015; McLaughlin & Townley, 1996):

$$\theta \overset{\text{det.}}{\rightarrow} x_{1:t} \overset{\text{det.}}{\rightarrow} y_{1:t}^{\text{sim}} \tag{3}$$

The map from state space to observation space ($x_{1:t} \overset{\text{det.}}{\rightarrow} y_{1:t}^{\text{sim}}$) can be a nonlinear function (e.g., Peter Jan van Leeuwen, 2015), but in hydrogeology—where the observed quantities are generally simulated directly—it can often be simplified to a dot product with a matrix H extracting the relevant entries from the state vector:

$$y_s^{\text{sim}} = Hx_s \tag{4}$$

where H is a matrix of zeros and ones. Combining equations 3 and 4 yields

$$\theta \overset{\text{det.}}{\rightarrow} x_{1:t} = M(x_0, u_{1:t}, \theta) \overset{\text{det.}}{\rightarrow} y_{1:t}^{\text{sim}} = \begin{bmatrix} y_0^{\text{sim}} \\ \vdots \\ y_t^{\text{sim}} \end{bmatrix} = \begin{bmatrix} Hx_0 \\ \vdots \\ Hx_t \end{bmatrix} \tag{5}$$

Given this deterministic projection into observation space, we require an error model to permit a probabilistic analysis of the observed data. One possible way to do so is through the addition of a lumped, additive, multivariate Gaussian error centered around the forward operator's output $y_{1:t}^{\text{sim}}$ with a specified covariance matrix Σ. Since we assume spatially and temporally uncorrelated errors in our study (equation 2), the error model could theoretically be broken down into a product of univariate Gaussians. To better reflect the incremental structure in which new observations become available, we instead define it as a multivariate Gaussian over all observation points available at a given time step s centered on the predicted observations y_s^{sim} with a diagonal covariance matrix Σ:

$$p(y_s|\theta) = \mathcal{N}(y_s; \mu = y_s^{\text{sim}}, \Sigma) = \frac{1}{\sqrt{(2\pi)^{N_s^{\text{obs}}} \det\Sigma}} \exp\left(-\frac{1}{2}(y_s - y_s^{\text{sim}})\Sigma^{-1}(y_s - y_s^{\text{sim}})\right) \tag{6}$$

where N_s^{obs} denotes the number of elements in the observation vector y_s. With the likelihood $p(y_s|\theta)$ defined, we can proceed to the specific algorithm used in this study.

2.3. Iterated Batch Importance Sampling (IBIS)

The IBIS algorithm was introduced by Chopin (2002) for the sequential filtering of static parameters, provided that the likelihood $p(y_s|\theta)$ can be evaluated. Since Bayes theorem (equation 1) is all but impossible to solve analytically in the general case, it often becomes necessary to resort to Monte Carlo approximations. These methods assume that a set of Monte Carlo samples (an *ensemble* of *particles*) may act as a surrogate for the distribution from which they were drawn. *SMC* methods, then, try to retain this surrogate property along otherwise intractable Bayesian update operations through gradual updates to the particle ensemble.

The IBIS algorithm is closely related to the particle filter. It is initialized by drawing an ensemble of N *independent*, *identically distributed* (*i.i.d.*) parameter particles $\theta^{(n)}$, $n = 1,...,N$ from the prior $p(\theta)$:

$$\theta^{(n)} \sim p(\theta) \forall n \in \{1, ..., N\} \tag{7}$$

which yields a particle approximation of the prior

$$\widehat{p}(\theta) = \sum_{n=1}^{N} w_0^{(n)} \delta\left(\theta - \theta^{(n)}\right) \overset{N\rightarrow\infty}{\rightarrow} p(\theta) \tag{8}$$

where δ is the Dirac delta function centered on $\theta^{(n)}$, $w_0^{(n)}$ is the particle's individual retrieval weight (initially a uniform $\frac{1}{N}$ starting from i.i.d. samples), and $\widehat{p}(\theta)$ denotes the particle approximation of $p(\theta)$, converging toward the true prior in the limit of an infinite number of particles. Particle filters implement the Bayesian update in two steps: first, by adjusting the previous weights $w_{t-1}^{(n)}$ according to the likelihood increments $l_t^{(n)}$:

$$l_t^{(n)} = p\left(y_t|\theta^{(n)}\right) \forall n \in \{1, ..., N\} \tag{9}$$

$$w_t^{(n)} = w_{t-1}^{(n)} l_t^{(n)} \forall n \in \{1, ..., N\}. \tag{10}$$

In preparation for a later step, we also carry along an estimate of the total likelihood accumulated so far (initially $L_0^{(n)} = 1$):

$$L_t^{(n)} = L_{t-1}^{(n)} l_t^{(n)} = p\left(y_{1:t}|\theta^{(n)}\right) \forall n \in \{1, ..., N\} \tag{11}$$

Second, since the updated weights of equation 10 no longer sum to unity, they must be renormalized (equation 12) to obtain the updated particle approximation of the posterior (equation 13):

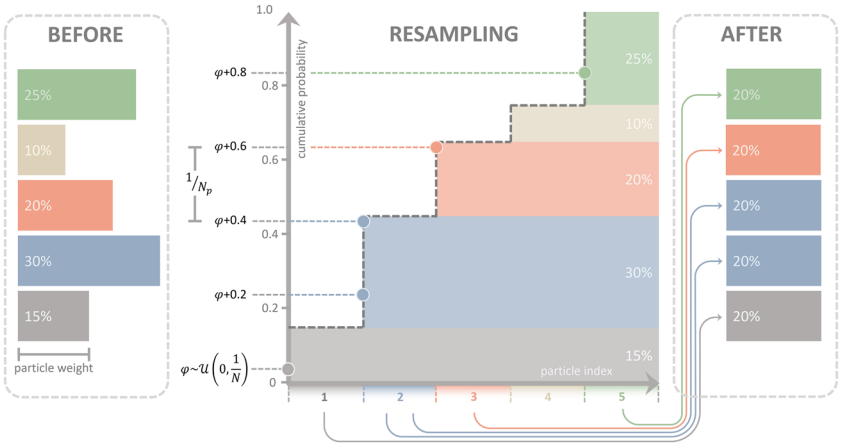

Figure 1. Schematic illustration of stochastic universal resampling for $\boldsymbol{N = 5}$. Starting with an ensemble of nonuniformly weighted particles (left), we construct a cumulative probability function (dashed gray line, center). After drawing a random offset $\boldsymbol{\varphi}$ from a uniform distribution $\mathcal{U}(\boldsymbol{min}=\mathbf{0}, \boldsymbol{max}=\mathbf{1}/\boldsymbol{N})$, we obtain resampled particle indices by sampling this function in additive increments of $\mathbf{1}/\boldsymbol{N}$. Finally, each particle slot inherits the variables of its respective resampled particle index ($\mathbf{1} \rightarrow \mathbf{1,2} \rightarrow \mathbf{2,3} \rightarrow \mathbf{2,4} \rightarrow \mathbf{3,5} \rightarrow \mathbf{5}$) and the weights are reset (right, equation 15).

$$W_t^{(n)} = \frac{w_t^{(n)}}{\sum_{m=1}^{N} w_t^{(m)}} \forall n \in \{1, ..., N\} \tag{12}$$

$$\widehat{p}(\theta|y_{1:t}) = \sum_{n=1}^{N} W_t^{(n)} \delta\left(\theta - \theta^{(n)}\right) \overset{N\to\infty}{\rightarrow} p(\theta|y_{1:t}) \tag{13}$$

Theoretically, the steps outlined in equations 9 to 13 may be repeated indefinitely. Eventually, however, sample degeneracy will cause only one single particle to retain any significant weight. This issue is generally addressed through a *resampling* step, making use of the surrogate property in equation 13 to randomly draw a set of new, equally weighted particles. From a practical perspective, resampling duplicates well-performing particles while discarding poorly performing ones.

The original IBIS algorithm only resamples once a certain degeneracy criterion is fulfilled. With a quasi-online implementation in mind, however, we want to control the computation time and trigger a resampling and rejuvenation step after every assimilation. Resampling can be implemented in several ways, each with their own advantages and drawbacks (Li et al., 2015). In this study we employ *stochastic universal resampling* (SUR; Baker, 1987; Townsend, 2003), a method resilient to random loss of diversity during resampling (Figure 1). Its output is a set of indices $(a^{(1)}, ..., a^{(N)})$ defining which particle values each particle slots $(1, ..., N)$ inherits (equation 1):

$$a^{(1)}, ..., a^{(N)} = \mathrm{SUR}\left(W_t^{(1)}, ..., W_t^{(N)}\right) \in \{1, ..., N\} \tag{14}$$

Mind that the number of resampled particles does not necessarily have to be equal to the previous ensemble size. In fact, we will make use of this property later to stifle growing computational demand by dynamically reducing the ensemble size. Meanwhile, however, the ancestral indices allow us to obtain a new, equally weighted ensemble of parameter particles and their associated variables:

$$\theta^{(n)} \leftarrow \theta^{(a^{(n)})}, \; L_t^{(n)} \leftarrow L_t^{(a^{(n)})}, \; w_t^{(n)} \leftarrow 1 \qquad \forall n \in \{1, ..., N\} \tag{15}$$

The process of resampling replaces low-weighted, unique samples with copies of high-weighted particles, reducing weight degeneracy at the cost of diversity (*sample impoverishment*; e.g., Doucet & Johansen, 2009; Li et al., 2015). While this process does not in itself solve the fundamental issue—replacing weight-based degeneracy with position-based degeneracy—it forms a more efficient basis for *rejuvenation*,

the reintroduction of diversity. Ideally, this rejuvenation mechanism should be invariant with respect to the underlying probability distribution or we risk compromising the ensemble's surrogate property. The IBIS algorithm achieves this through a *Metropolis-Hastings Markov Chain Monte Carlo* (*MH-MCMC*) jump for each particle $\theta^{(n)}$, employing a proposal density $p\left({}^{*}\theta \mid \theta^{(n)}\right)$:

$$ {}^{*}\theta^{(n)} \sim p\left({}^{*}\theta \mid \theta^{(n)}\right) \qquad \forall n \in \{1, \ldots, N\} \tag{16} $$

where the asterisk denotes the proposal, so ${}^{*}\theta^{(n)}$ is the proposal for an original particle $\theta^{(n)}$. MH-MCMC jumps are randomly accepted with a probability defined by the ratio between the transition densities $\left(p\left(\theta^{(n)} {}^{*}\theta^{(n)}\right)\right.$ and $\left.p\left({}^{*}\theta^{(n)} \mid \theta^{(n)}\right)\right)$, the prior densities $\left(p\left(\theta^{(n)}\right)\right.$ and $\left.p\left({}^{*}\theta^{(n)}\right)\right)$, and the total likelihoods $\left(L_t^{(n)}\right.$ and $\left.{}^{*}L_t^{(n)}\right)$, capped at one:

$$ p_{\text{accept}}^{(n)} = \min\left(1, \frac{p\left({}^{*}\theta^{(n)}\right) {}^{*}L_t^{(n)}\, p\left(\theta^{(n)} {}^{*}\theta^{(n)}\right)}{p\left(\theta^{(n)}\right) L_t^{(n)}\, p\left({}^{*}\theta^{(n)} \mid \theta^{(n)}\right)}\right) \forall n \in \{1, \ldots, N\} \tag{17} $$

If the proposal is accepted ($v^{(n)} \sim \mathcal{U}(0,\ 1) < p_{\text{accept}}^{(n)}$), the new particle (${}^{*}\theta^{(n)}$) and its associated variables (${}^{*}L_t^{(n)}$, ${}^{*}w_t^{(n)}$) replace the original:

$$ \text{if } v^{(n)} \sim \mathcal{U}(0,\ 1) < p_{\text{accept}}\text{:} \quad \theta^{(n)} \leftarrow {}^{*}\theta^{(n)},\ L_t^{(n)} \leftarrow {}^{*}L_t^{(n)},\ x_t^{(n)} \leftarrow {}^{*}x_t^{(n)} \quad \forall n \in \{1, \ldots, N\} \tag{18} $$

If the proposal is rejected, the algorithm continues to the next time step using the original particle and its associated variables. The evaluation of $p_{\text{accept}}^{(n)}$ is also why we carried along the total likelihood $L_t^{(n)}$ (equations 11 and 15). The most computationally expensive term in equation 17 to evaluate is ${}^{*}L_t^{(n)}$, which requires resimulating the entire observation history up to time step t. The cost of this evaluation increases as the assimilated time series grows, thus precluding a true online implementation. Moradkhani et al. (2012) propose a workaround by evaluating only the latest likelihood increment ${}^{*}l_t^{(n)}$ and estimating $p\left({}^{*}\theta^{(n)} \mid y_{1:t-1}\right)$ by evaluating a Gaussian distribution fitted to the original particles. This approach has the advantage of keeping the computational demand constant but becomes problematic if the ensemble collapses.

In this study, we instead opt to resimulate the full observation history to obtain ${}^{*}L_t^{(n)}$ and compensate for the growing computational demand by resampling a reduced number of particles if the simulation time exceeds a user-specified threshold. Since this trick will not work indefinitely—only until the resimulation takes longer than the time between assimilation intervals—our IBIS-based optimizer is only quasi-online. If the proposal is accepted, we replace the particle and its associated variables with its proposal's equivalents similarly to resampling (equation 15):

where $v^{(n)}$ is a random value drawn from the uniform distribution $\mathcal{U}(\min = 0, \max = 1)$. After rejuvenation, the algorithm resumes from equation 9. With the theoretical foundations laid out, we can proceed to the algorithm's implementation.

3. Data and Implementation

3.1. Study Area

Figure 2 illustrates various features of Settolo, the studied field site. The site is located on the eastern bank of the river Piave, near the city of Valdobbiadene in the province of Treviso, Northern Italy. Its unconfined aquifer is recharged by the nearby river and exploited for drinking water production (Zovi, 2014; Zovi et al., 2017). The surface elevation of the model domain slopes from 165 m above sea level in the north-west with an aquifer depth of about 30 m to about 155 m above sea level with an aquifer depth of about 50 m in the south-east. Assimilated water table measurements are available from 18 observation wells and two production wells from 1 December 2010 to 31 January 2012. The main assimilation period is from 1

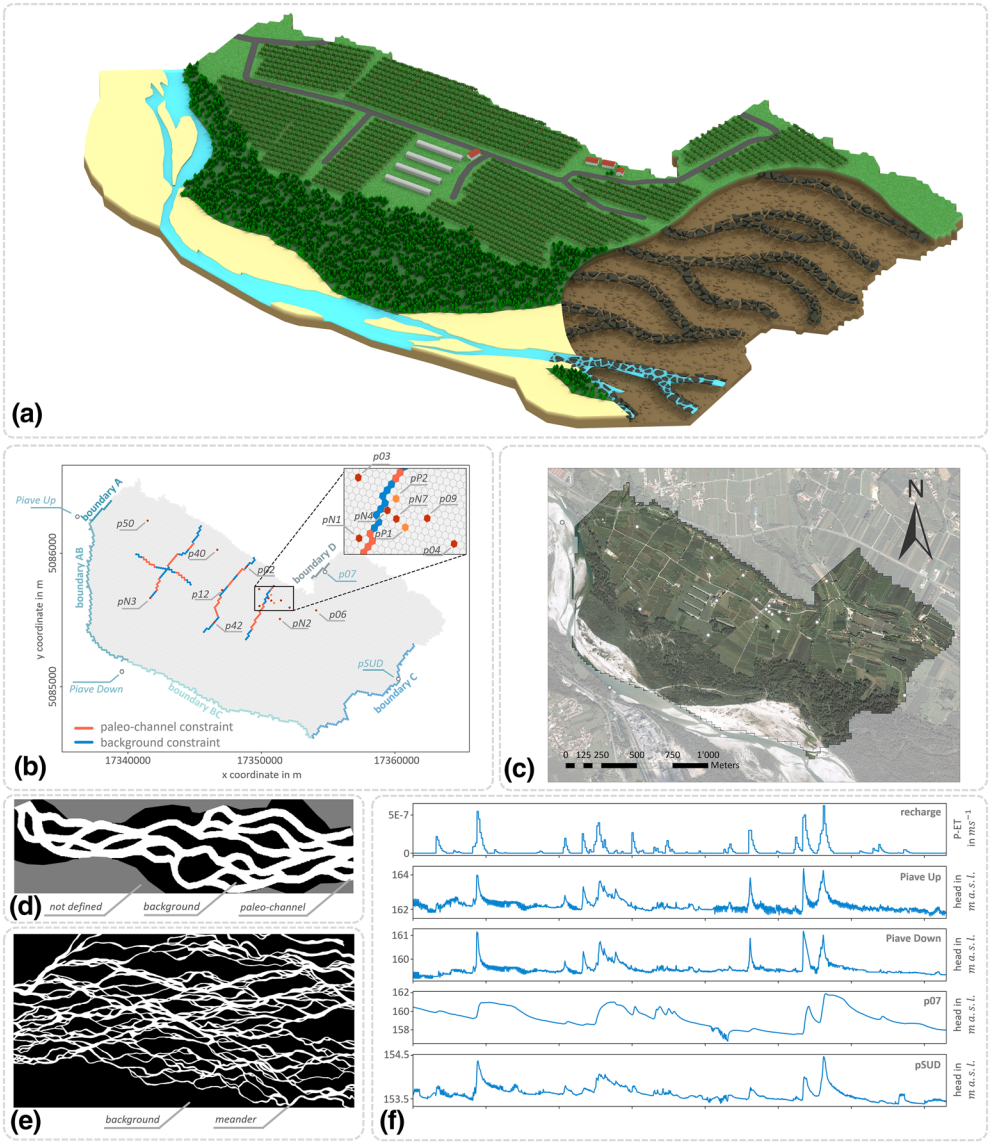

Figure 2. Overview of the field site in Settolo, Italy: (a) schematic render of the geological concept (not to scale); (b) grid-based map with geophysical constraints (orange and blue hexagons), observation (crimson hexagons), and pumping (yellow hexagons) wells; the gray circles mark the positions of the observation wells informing the boundaries; (c) model domain overlay on a satellite image; (d) site-based training image for multipoint statistics simulations; (e) training image based on Skeidararsandur river, Iceland, for multipoint statistics simulations; (f) P-ET and hydraulic head in the four boundary wells over the data assimilation period.

February 2011 to 31 January 2012; the remaining data are used for validation. Of these 18 observation wells, four (*Piave Up*, *Piave Down*, *p07*, and *pSUD*) serve to inform five time-variable head boundaries (Figure 2b). Boundaries AB and BC are linearly interpolated from *Piave Up* and *Piave Down*, and *Piave Down* and *pSUD*, respectively, whereas A, C, and D are uniformly assigned the heads of their neighboring wells. All other boundaries are assumed to be no-flow. An irrigation water pipeline crosses the Piave riverbed between *Piave Down* and *pSUD*, acting as a physical obstacle which may cause a river level discontinuity between 0.5 and 1 m, depending on discharge. This obstacle was not represented explicitly in the model and partially motivates the forcing model introduced in the next section. Figure 2f illustrates the raw forcing data over the assimilation period. Raw recharge estimates were obtained by taking the difference between precipitation and evapotranspiration estimates after the Penman-Monteith equation using meteorological data from a nearby station in Valdobbiadene.

The (hydro)geological characterization of the site assumes the presence of highly conductive paleo-channels of the nearby river and is supported by electric resistivity tomography (ERT) measurements along a number of transects (Figure 2b, see also Zovi et al., 2017). To maintain consistency with this nontrivial geological prior as defined by a training image (Figure 2d), we use a MPS framework (*DeeSse*; Mariethoz et al., 2010, 2015) and constrain its realizations with the ERT transects. These constraints force the MPS framework to reproduce the prescribed facies characterization (Figure 2b). The training image (Figure 2d) was derived from satellite imagery based on the current river morphology. We note that the process of deriving a training image in such a manner can be contentious, as there is no guarantee that contemporary hydraulics are representative of the geomorphology at the time of deposition of the paleo-channels (Zovi et al., 2017). To gauge the impact of the training image, we also test an alternative image based on an unrelated fluvial system in Iceland (Figure 2e). Further details and information on the field site are available in Zovi et al. (2017), Zovi (2014), and at http://settolo.dicea.unipd.it/index.php.

3.2. Assembling the Forward Operator

In the following, we outline the components of the forward operator introduced in section 2.2. This operator is composed of several auxiliary, deterministic modules applied in sequence. For the moment, we restrict ourselves to the overall structure and reserve greater detail for the following sections.

The first module is the *field generator* $G(\Theta)$. When pursuing conformance with a complex geological prior, it is rarely helpful to optimize the grid parameters θ directly. Instead, we optimize a set of *hyperparameters* Θ, which broadly serve as the forward operator's input and thus also parameterize the field generator. This module then creates parameter fields consistent with the geological prior, thereby ensuring conformance by construction:

$$G(\Theta) \rightarrow \theta \tag{19}$$

The second, perhaps more unconventional module is the *forcing model* $F(\Theta, U_{0:t})$. Groundwater dynamics are usually controlled by external driving forces such as recharge or time-variable flow across boundaries. In general, however, we cannot observe these forcings directly and must instead approximate them from related quantities. This approximation demands assumptions (e.g., instantaneous recharge, well in contact to aquifer, and extrapolated head boundaries), which risk introducing systematic bias. To permit compensation for such effects, we propose a hyperparameterized preprocessor that transforms the raw forcing data $U_{0:t}$ into an updated form $u_{0:t}$:

$$F(\Theta, U_{0:t}) \rightarrow u_0, \ldots, u_t \tag{20}$$

The third module is an interpolation and extrapolation of the initial states x_0 through inverse distance weighting (*IDW*; Shepard, 1968) based on variable-head boundaries included in u_0 (which depend on Θ; see equation 20) as well as an initial set of head observations y_0:

$$\mathrm{IDW}(u_0, y_0) \rightarrow x_0 \tag{21}$$

The fourth deterministic module is the numerical model, which requires the previously generated grid parameters θ, the model forcing $u_{1:t}$, and initial conditions x_0 to predict the states $x_{1:t}$:

$$M(x_0, u_{1:t}, \theta) \rightarrow x_1, \ldots, x_t \tag{22}$$

Finally, the fifth module simply extracts the predicted observations $y^{\mathrm{sim}}_{1:t}$ from the simulated state trajectory $x_{1:t}$:

$$H x_{1:t} \rightarrow y^{\mathrm{sim}}_{1:t} \tag{23}$$

With the introduction of hyperparameters Θ as the variables of interest, we note that the theory outlined in section 2 should be read in terms of the hyperparameters Θ, not the grid parameters θ. These variables completely define the error model, by parameterizing both its deterministic (the forward operator) and

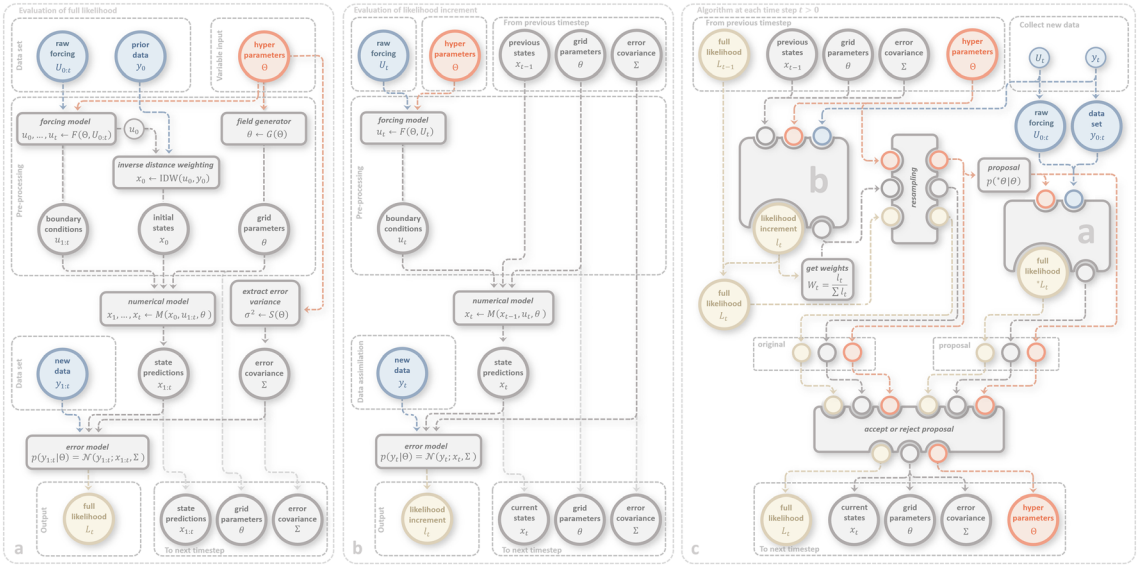

Figure 3. (a) Schema for the evaluation of the full likelihood $\boldsymbol{L_t}$, (b) the likelihood increment $\boldsymbol{l_t}$, and (c) the algorithm at time $\boldsymbol{t} > 0$. For $\boldsymbol{L_t}$ (a), we generate the model input files $\boldsymbol{x_0}$, $\boldsymbol{u_{1:t}}$, and θ from the hyperparameters $\boldsymbol{\Theta}$ and the raw forcing data $\boldsymbol{U_{0:t}}$. After obtaining $\boldsymbol{x_{1:t}}$, we use the assimilated data $\boldsymbol{y_{1:t}}$ and the error covariance matrix $\boldsymbol{\Sigma = I\sigma^2}$ to evaluate $\boldsymbol{L_t}$. For $\boldsymbol{l_t}$ (b), $\boldsymbol{x_{t-1}}$, θ, and $\boldsymbol{\Sigma}$ are already available from the previous time step after initialization. We must only calculate $\boldsymbol{u_t}$ and can directly obtain $\boldsymbol{x_t}$ by deterministic simulation. Afterward, we can evaluate $\boldsymbol{l_t}$ and pass $\boldsymbol{x_t}$ along as initial condition for the next time step. The implementation of the algorithm (c) uses both a and b.

probabilistic (the additive Gaussian error) part. The algorithm's procedure during an assimilation increment is schematically illustrated in Figure 3. In the following, we will explore the modules in greater detail.

3.2.1. Field Generator $G(\Theta)$

The field generator is built around a facies distribution map, the output of the MPS framework DeeSse (Mariethoz et al., 2010, 2015; Straubhaar, 2019). This facies distribution map is a set of hyperparameters, which defines whether a cell belongs to the paleo-channel or background sediment, then assigns grid parameters accordingly. Additional hyperparameters define the hydraulic properties of each facies: the mean hydraulic conductivity K, an internal conductivity heterogeneity map with range K, and a specific yield S_y, which was assumed to be homogeneous, following Zovi et al. (2017). Figure 4 illustrates how we assemble the conductivity maps and create the full grid parameters θ from the two complementary parts. The heterogeneity map was generated through convolution of a white noise field with an isotropic Gaussian filter, then normalized and centered around zero, and finally scaled by K.

3.2.2. Forcing Model $F(\Theta, U_{0:t})$

The forcing considered in this study are time-variable prescribed head boundaries (Figure 2b) and transient, uniform recharge (Figure 2f). The pumping rates in the production wells were considered sufficiently well quantified to warrant exclusion from the forcing model.

The forcing model is a relatively simple preprocessor illustrated in Figure 5. The raw forcing data $U_{0:t}$ (hydraulic heads or recharge flux) are first normalized, then transformed according to the spline defined by three hyperparameters—the spline control points—then reverted into their canonical range (Figure 5a). These control points are defined independently for each boundary well and recharge.

For recharge, this process is further extended by distributing the transformed recharge $\widetilde{U}_s$ at each time s among the next λ time steps according to an exponential distribution, where λ is an additional hyperparameter to be optimized (Figure 5b). Finally, the full recharge time series is assembled by summarizing the resulting recharge components for each time step (Figure 5c). The transformed hydraulic heads and the transformed and redistributed recharge constitute $u_{0:t}$.

The motivation for a spline transformation are (i) possible nonlinearities during rainfall or high-flow events, which might cause the forcing to fit better during some meteorological regimes than during others, and (ii)

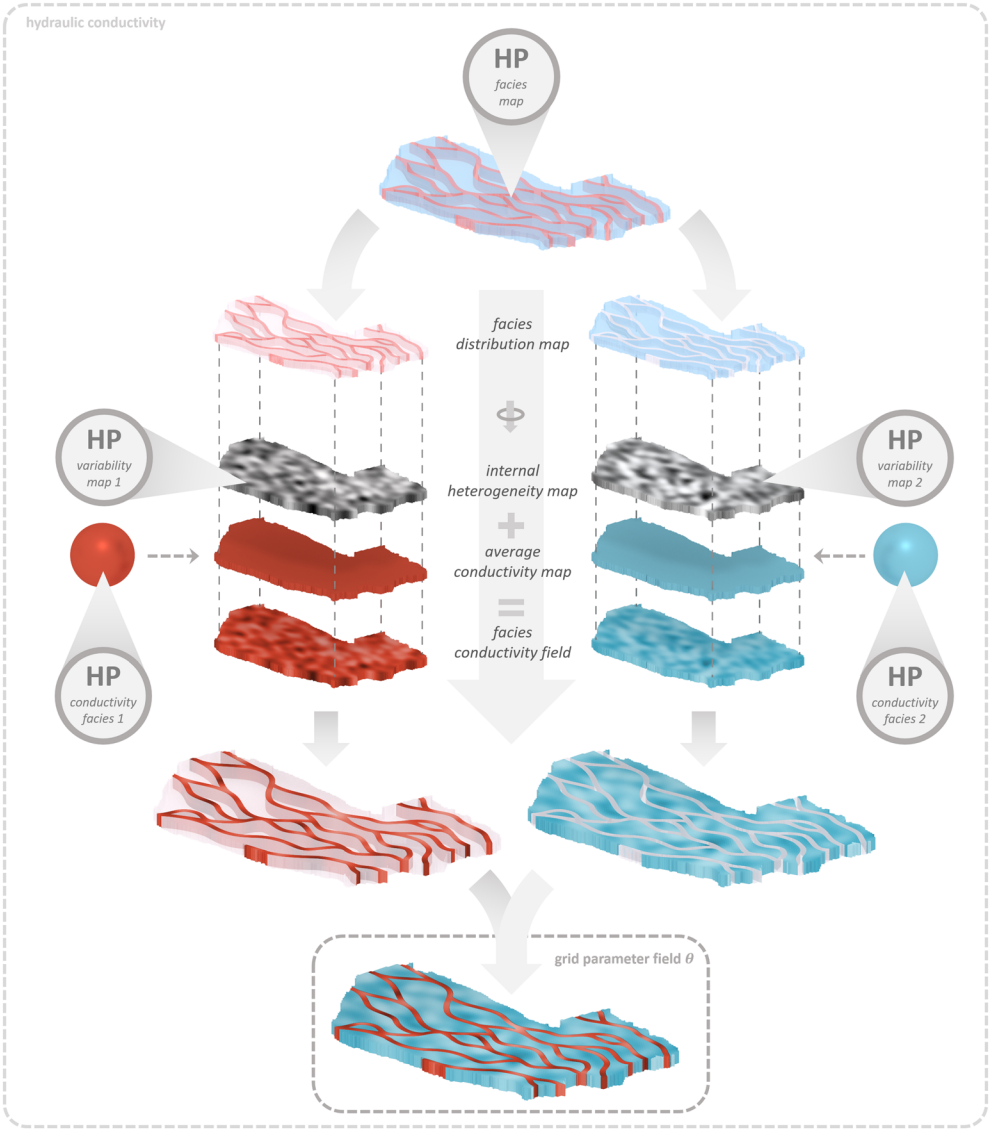

Figure 4. Schematic render of the field generator. We generate facies conductivity fields for the background sediment and paleo-channels by offsetting a uniform, average conductivity field with internal variability. The facies distribution map then extracts grid parameter values from each map according to the facies assignment and assembles the parameter field θ. Assignment of specific yield proceeds equivalently, barring the addition of internal heterogeneity.

the possibility of bias from the interpolation and extrapolation of the hydraulic heads along their respective boundaries. The transformation and redistribution of recharge is justified by large uncertainties in P and ET estimates, omission of overland flow, and vadose zone dynamics.

We note that for a real application, the full time series is not available from the start. As such, $U_{\min}$ and $U_{\max}$ used during normalization could change as new extrema are recorded. This affects the transformation between canonical and normalized space and as such the value ranges affected by the spline control points, which are defined in normalized space. In real data assimilation scenarios, we thus recommend specifying a sufficiently broad $U_{\min}$ and $U_{\max}$ a-priori.

3.2.3. Inverse Distance Weighting IDW(u_0, y_0)

The initial states are generated with IDW according to Shepard (1968). This approach extrapolates from a collection of N_v known points $v_0 = \{y_0 \cup u_0\}$ to any of the N_{cell} grid points based on spatial distance $d(i,j)$ between cells i and known points j as well as a power factor p (in our case $p = 3$).

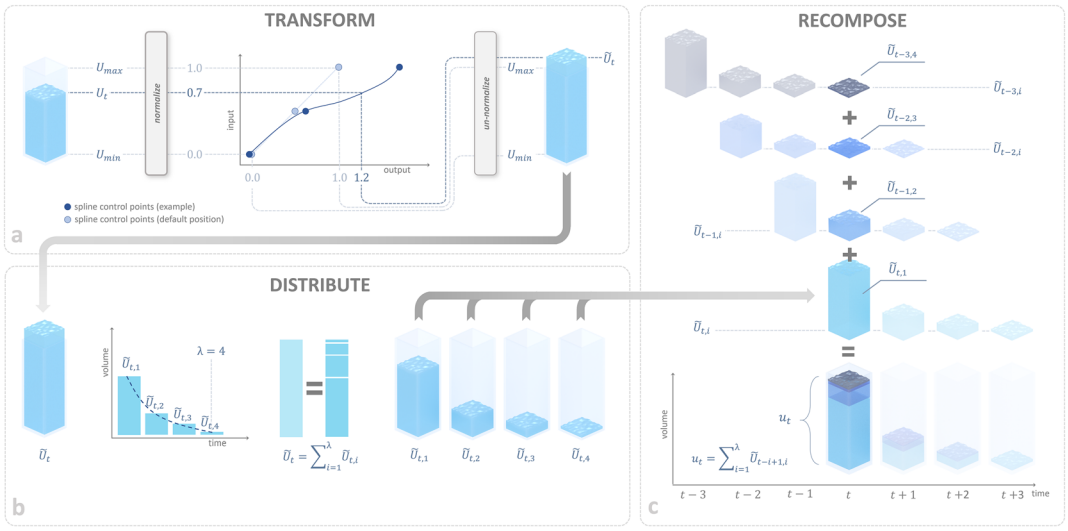

Figure 5. Schematic mechanism of the forcing model, with water column heights representing recharge or hydraulic head at the boundaries. The first step (a) applies to both the variable-head boundaries and the recharge: The raw quantity $\boldsymbol{U_t}$ (volume for recharge and head for boundaries) is normalized, then transformed according to a function defined by three spline control points, then reverted back to obtain the transformed forcing $\widetilde{\boldsymbol{U}}_{\boldsymbol{t}}$, which may lie outside the interval of $\boldsymbol{U_{min}}$ and $\boldsymbol{U_{max}}$. For groundwater recharge, $\widetilde{\boldsymbol{U}}_{\boldsymbol{t}}$ is then distributed between the next $\boldsymbol{\lambda}$ time steps according to an exponential distribution (b). Finally, the composite recharge for a certain time step $\boldsymbol{u_t}$ is assembled out of the recharge components of earlier time steps (c).

3.2.4. Numerical Model $M(x_0, u_{1:t}, \theta)$

The field site is simulated as a two-dimensional, unconfined, transient groundwater model using MODFLOW-USG (Panday et al., 2013) with the Python interface FloPy (Bakker et al., 2016). Its full governing equation describes water balance conservation on an infinitesimally small control volume:

$$\frac{\partial}{\partial x}\left(K_x \frac{\partial h}{\partial x}\right) + \frac{\partial}{\partial y}\left(K_y \frac{\partial h}{\partial y}\right) + \frac{\partial}{\partial z}\left(K_z \frac{\partial h}{\partial z}\right) = S_s \frac{\partial h}{\partial t} + Q_s \tag{24}$$

where $K_{x/y/z}$ are the (possibly heterogeneous and anisotropic) hydraulic conductivities in each spatial dimension, Q_s is a source-sink term, and S_s is the specific storage of the porous medium. MODFLOW solves these equations through a finite volume approach (Langevin et al., 2017). The model domain is discretized with uniform, hexagonal cells. Since DeeSse demands a regular grid, we further define a regular but anisotropic support grid, which contains the hexagonal cell centers for the MPS simulation. Cells with extraction wells are further subdivided to increase grid resolution around the cones of depression (Figure S1 in the supporting information). The subdivided cells are not resolved for the purpose of optimization and inherit their host cell's parameterization. The cell count without the subdivision is 12,856.

3.3. Probabilistic Setup

3.3.1. Hyperparameter Priors

Since we seek to optimize the hyperparameters Θ, we also define the model prior in terms of these variables. Summing over section 3.2, we have a total of $3N_{\text{cells}}$+23 hyperparameters—three hyperparameter fields (the facies distribution and two independent internal variability maps) and 23 scalar hyperparameters. The hyperparameter limits and priors are listed in Table 1.

3.3.2. Rejuvenation Mechanism

As described in section 2.3, we rejuvenate the ensemble with MH-MCMC jumps. This procedure requires a proposal distribution $p\left({}^{*}\Theta \,|\Theta^{(n)}\right)$ to suggest new values ${}^{*}\Theta$ for an original particle $\Theta^{(n)}$. While we could update all hyperparameters at once, doing so is not always useful—to achieve a good acceptance rate, it is important to keep the magnitude of the mutation sufficiently low. We achieve this by defining $p\left({}^{*}\Theta \,|\Theta^{(n)}\right)$ as a mixture distribution of random multivariate normal mutations, which only update a subset of

10.1029/2019WR026777

Table 1
Hyperparameter Limits, Priors, and Proposal Distributions for the Scalar Hyperparameters

Hyperparameters [unit]	Limits		Prior pdf			Proposal distribution								
	min	max	uniform [limits]	Gaussian [σ]	exponential [α]	standard deviation [σ]	correlation matrix							
K_1 [$\log_{10}$m/s]	−2.3	−1.9	yes	N/A	N/A	0.05	1		−0.5			−0.5		1
	−3.3	−2.7	yes	N/A	N/A									
K_1 [$\log_{10}$m/s]	0.1	0.2	yes	N/A	N/A	0.005	1		−0.5			−0.5		1
	0.2	0.5	yes	N/A	N/A									
$S_{y,1}$ [$\log_{10}$m/s]	0.1	0.35	yes	N/A	N/A	0.025	1		−0.5			−0.5		1
	0.1	0.35	yes	N/A	N/A									
SCP_{well}(0.0) [−]	N/A	N/A	N/A	0.01	N/A	0.001	1	0.5	0.5	0.5	1	0.5	0.5	0.5	1
	N/A	N/A	N/A	0.01	N/A	0.05									
SCP_{well}(1.0) [−]	N/A	N/A	N/A	0.01	N/A	0.1									
SCP_{rech}(0.0) [−]	N/A	N/A	N/A	0.001	N/A	0.001	1	0.5	0.5	0.5	1	0.5	0.5	0.5	1
	N/A	N/A	N/A	0.05	N/A	0.05									
SCP_{rech}(1.0) [−]	N/A	N/A	N/A	0.1	N/A	0.1									
λ [3 hr]	1	N/A	N/A	N/A	1	1						N/A			
σ [m]	0	N/A	N/A	N/A	0.025	0.01						N/A			

Note. Limits are assigned only to the nonnegative recharge delay **λ** and model error **σ**, as well as the hydrogeological hyperparameters **K**,**K**, and $\mathbf{S_y}$, where subscript "1" denotes the paleo-channel and subscript "2" denotes the background sediment. The priors are either uniform (with the assigned limits), Gaussian (with mean in brackets, and listed standard deviation), or exponential (with listed factor **α**). For the proposal distribution, some hyperparameters are updated jointly with a multivariate Gaussian of listed standard deviation and correlation structure. The limits for the hydraulic parameters were adopted from Zovi et al. (2017).
Abbreviations: N/A: not available; pdf: probability density function.

hyperparameters at a time. Independently, for each particle, we either update the hydraulic hyperparameters (*K*, *K*,S_y: 65%), the spline control points for recharge (SCP_{rech}: 7.5%) or one of the boundary wells (SCP_{well}: 3.75 % * 4 = 15%), the recharge delay λ (7.5%), or the error standard deviation σ (5%). The proposal standard deviations and correlation structures between hyperparameters are listed in Table 1.

Furthermore, we must account for possible asymmetries in the proposal $\left(\text{i.e., } p\left({}^{*}\Theta^{(n)}\,|\Theta^{(n)}\right) \neq p\left(\Theta^{(n)*}\Theta^{(n)}\right)\right)$, which arise for example in the presence of hyperparameter limits. Quantifying these asymmetries would require evaluating truncated, correlated, multivariate Gaussians pdfs, which is not a trivial task. Instead, we simplify this issue by interpreting hyperparameter limits as reflective boundaries, then summing the probability densities of all different jumps, which would result in equivalent proposals (see Figure S2). Since this would require evaluating probability densities of an infinite series of mirrored positions along dimensions with both a lower and an upper limit, we restrict our evaluation only to the probability densities of the first reflection across each parameter limit. For practical application, we reflect any potential proposal falling outside the hyperparameter limits across its closest limit until it falls within the hyperparameter constraints.

The hyperparameter maps (facies distribution and internal variability) are updated together with the hydraulic hyperparameters. The facies distribution is updated with a blocking-moving window procedure similar to Alcolea and Renard (2010): first, we select a random cell, then remove all entries in a random radius between 75 and 500 m around this cell. After this, the facies distribution within the masked circle is regenerated with MPS, with the remaining facies assignments outside acting as constraints to ensure continuity of the features. The internal variability maps are updated through small, gradual changes: first, each map is normalized and recentered around zero. Then a new variability map is generated (see section 3.2.1), scaled by 0.25, and added to the previous map. The result is then renormalized, recentered, and scaled with its respective (updated) *K*.

3.4. Scenarios and Computational Setup

In this study we consider three different scenarios: Scenario 1 uses the training image based on the local field site (Figure 2d) and the forcing model. Scenario 2 is the same as Scenario 1 but replaces the field-based training image with the alternative training image (Figure 2e). Scenario 3 differs from Scenario 1 only in omission of the forcing model. We initialized each scenario with 350 particles, then allowed the algorithm to dynamically adjust the ensemble size to meet the prescribed computational limits.

The optimization algorithm is implemented in Python 3.3.2 in a parallelized framework with eight workers on three workstations. Simulations for each scenario are repeated in triplicate with different random seeds to test reproducibility: the scenarios with suffix "a" were simulated on a Lenovo ThinkPad X1 Carbon with an Intel® Core™ i7-6600 CPU with two cores (four logical processors) at 2.60 GHz and 16 GB of RAM, the simulations with suffix "b" on a workstation with an Intel® Core™ i7-3770 CPU with four cores (eight logical processors) and 8 GB of RAM, and the simulations with suffix "c" on a workstation with an Intel® Core™ i7-2600 CPU with four cores (eight logical processors) at 3.40 GHz and 8 GB of RAM.

The MPS framework we used was DeeSse (Mariethoz et al., 2010, 2015; Straubhaar, 2019), a commercial direct sampling MPS software freely available for scientific use. The communication between DeeSse and our optimizer was established with a self-designed Python interface. Data were assimilated every 3 hr, with resampling and rejuvenation steps being triggered every 24 hr of data. The allocated simulation time for 24 hr worth of data was bounded between 600 and 900 s—considering 365 days of data, on average a little over 3 days—well below the time available in a real application. Before proceeding to the presentation of the results, we will introduce the performance metrics used.

3.5. Performance Metrics

The *root-mean-square error* (*RMSE*) is calculated for each observation well $o = 1,...,N_{\text{obs}}$ according to

$$\text{RMSE}^o = \sqrt{\frac{1}{N}\frac{1}{t}\sum_{n=1}^{N}\sum_{s=1}^{t}\left(y_s^{\text{sim},o,(n)} - y_s^o\right)^2} \tag{25}$$

and subsequently averaged across all observation wells

$$\overline{\text{RMSE}} = \frac{1}{N_{\text{obs}}}\sum_{o=1}^{N_{\text{obs}}}\text{RMSE}^o \tag{26}$$

while the *bias* is calculated according to

$$\overline{\text{bias}} = \frac{1}{N}\frac{1}{N_{\text{obs}}}\frac{1}{t}\sum_{n=1}^{N}\sum_{o=1}^{N_{\text{obs}}}\sum_{s=1}^{t}\left(y_s^{\text{sim},o,(n)} - y_s^o\right) \tag{27}$$

The *percentage bias* and the *Kling-Gupta efficiency* (Gupta et al., 2009) are two performance metrics popular in surface hydrology. Both metrics require a reference level, which is not clearly defined for groundwater tables. In this study, we use the aquifer bottom at each observation well z^o as the reference level:

$$\overline{\text{pbias}} = \frac{1}{N}\frac{1}{N_{\text{obs}}}\frac{1}{t}\sum_{n=1}^{N}\sum_{o=1}^{N_{\text{obs}}}\sum_{s=1}^{t}\left(\frac{y_s^{\text{sim},o,(n)} - y_s^o}{y_s^o - z^o}\right)*100 \tag{28}$$

$$\overline{\text{KttGE}} = \frac{1}{N_o}\sum_{o=1}^{N_o}\left(1 - \sqrt{(r^o-1)^2 - (\alpha^o-1)^2 - (\beta^o-1)^2}\right) \tag{29}$$

where r^o is the Pearson correlation coefficient, α^o the ratio of standard deviations, and β^o the ratio of means (against the reference level) between the simulated and observed time series at observation well o. The mean Spearman correlation coefficient (Spearman, 1987) was averaged across all observation wells:

$$\overline{r_{sp}} = \frac{1}{N_o}\sum_{o=1}^{N_o} r_{sp}^o \tag{30}$$

where r_{sp}^o was obtained with the python library scipy.stats.

4. Results

The simulation results are evaluated at the end of the data assimilation period (day 365) using the final hyperparameter ensemble and the performance metrics described in section 3.5. We report these metrics

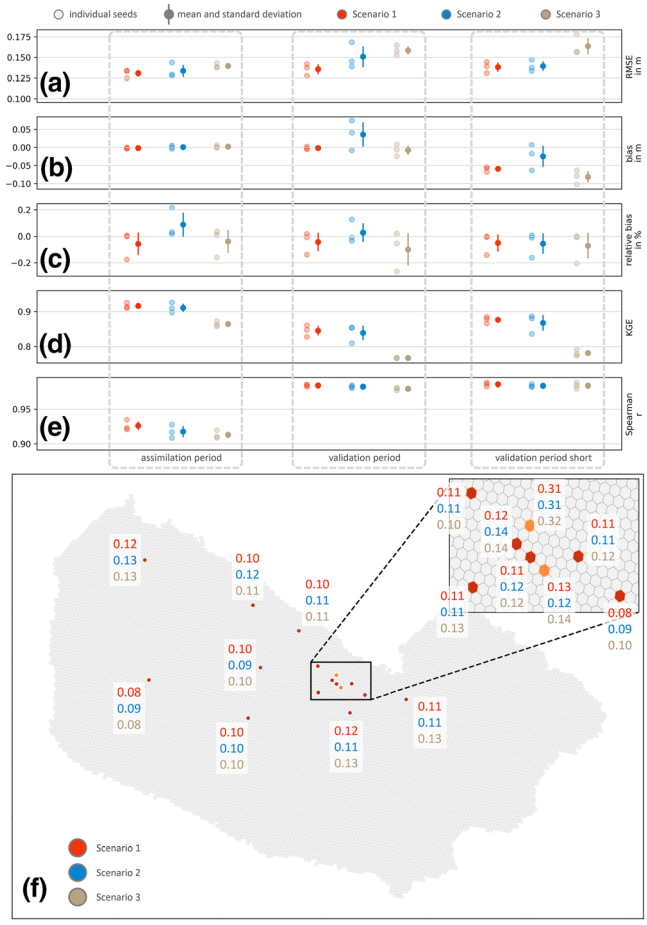

Figure 6. Performance metrics for the posterior ensemble of hyperparameters at the end of the 1-year assimilation period, evaluated over the assimilation period, the 90-day validation period, and the shortened 60-day validation period. Individual seeds, mean, and standard deviation are plotted for (a) root-mean-square error, (b) bias, (c) relative bias, (d) Kling-Gupta efficiency, and (e) Spearman *r*. (f) The lowermost subplot depicts average root-mean-square errors over the assimilation period for all scenarios and observation (red) and pumping (orange) wells. Further detail is provided in tabular form in Table S1.

over the 365-day assimilation period (from 1 February 2011 to 31 January 2012) and the 90-day validation period (from 1 December 2010 to 28 February 2011) used in Zovi et al. (2017). Since this validation period overlaps for the last month with the assimilation period—a decision made since the validation set's observation time series otherwise would not extend to all wells—we also evaluated a shortened, nonoverlapping validation period spanning only 60 days (from 1 December 2010 to 29 January 2011) for reference. The resulting metrics for all three scenarios and evaluation periods are depicted in Figure 6 and listed in Table S2.

4.1. Performance Metrics

In general, most scenarios display relatively similar performance metrics (Figure 6 and Table S2). The overall model fit is very satisfying across all scenarios (Figure 7, Figures S3 to S11). On average, Scenario 1 yields the best $\overline{\text{RMSE}}$ over the assimilation (0.131 m), validation (0.136 m), and shortened validation (0.138 m) periods, closely followed by Scenario 2 (0.134, 0.151, and 0.139 m). Scenario 3 performs slightly worse (0.140, 0.158, and 0.164 m) than the other two scenarios. The absolute values of $\overline{\text{bias}}$ and $\overline{\text{pbias}}$ are very low across the assimilation period in all scenarios. Over the validation periods, biases are slightly larger, likely owed to the increased prominence of the rain event and its subsequent recession (Figure S3). Mean $\overline{\text{KGE}}$s over the assimilation period are also favorable, ranging from 0.864 (Scenario 3) to 0.916 (Scenario 1), but are somewhat lower over the validation (0.767 in Scenario 3 to 0.845 in Scenario 1) and shortened validation (0.781 in Scenario 3 and 0.877 in Scenario 1) periods. We note that $\overline{\text{KGE}}$s are lower in Scenario 3, owed primarily to a lower α value in this scenario (Tables S2 to S4). The mean Spearman correlation coefficients $\overline{r_{sp}}$ are generally very high (>0.90), indicating strong control by the boundaries, and, curiously, are somewhat higher in the validation periods than in the assimilation period.

Analyzing the well-specific RMSE^{o}s over the assimilation period reveals a significant error in the second pumping well (*pP2*; Figure 6f, northern pumping well). A look at Figure 7b reveals that the hydraulic head in this well is systematically overpredicted and indicates that this error is exacerbated during periods of intense pumping. This could suggest that in this well, the connection of the water table to the surrounding aquifer is weakened, possibly due to a local region of low conductivity, a clogged well screen, or skin effects during pumping (e.g., Barrash et al., 2006). It is furthermore possible that the vicinity of the geophysical constraints (see Figure 2b) might not have left the MPS algorithm sufficient flexibility to possibly generate a more promising facies constellation.

Finally, we observe that the error residuals seem to be significantly autocorrelated across time and space (Figure 7b), with a pronounced trough—indicating underprediction—from roughly April to November, or (conversely) an overprediction from November to April. Similar start and end points over the assimilation period suggest that this could be due to seasonal phenomena not accounted for.

4.2. Scalar Hyperparameters

Figure 8 illustrates the hyperparameters (and their uncertainty, when applicable) for all scenarios along the optimization process. The first and third subplots (Figures 8a and 8c) visualize the ensemble size adjustments to maintain a bounded computational effort. The effective ensemble size in terms of unique i.i.d. samples (Figure 8b, semitransparent) is generally about an order of magnitude lower than the raw ensemble size (Figure 8b, full). The second subplot (Figure 8b) illustrates the time-averaged acceptance chance for rejuvenation proposals, which is markedly lower for Scenario 3—the scenario without access to the forcing model. The error standard deviation (Figure 8d) reveals a similar trend across all scenarios and random seeds,

10.1029/2019WR026777

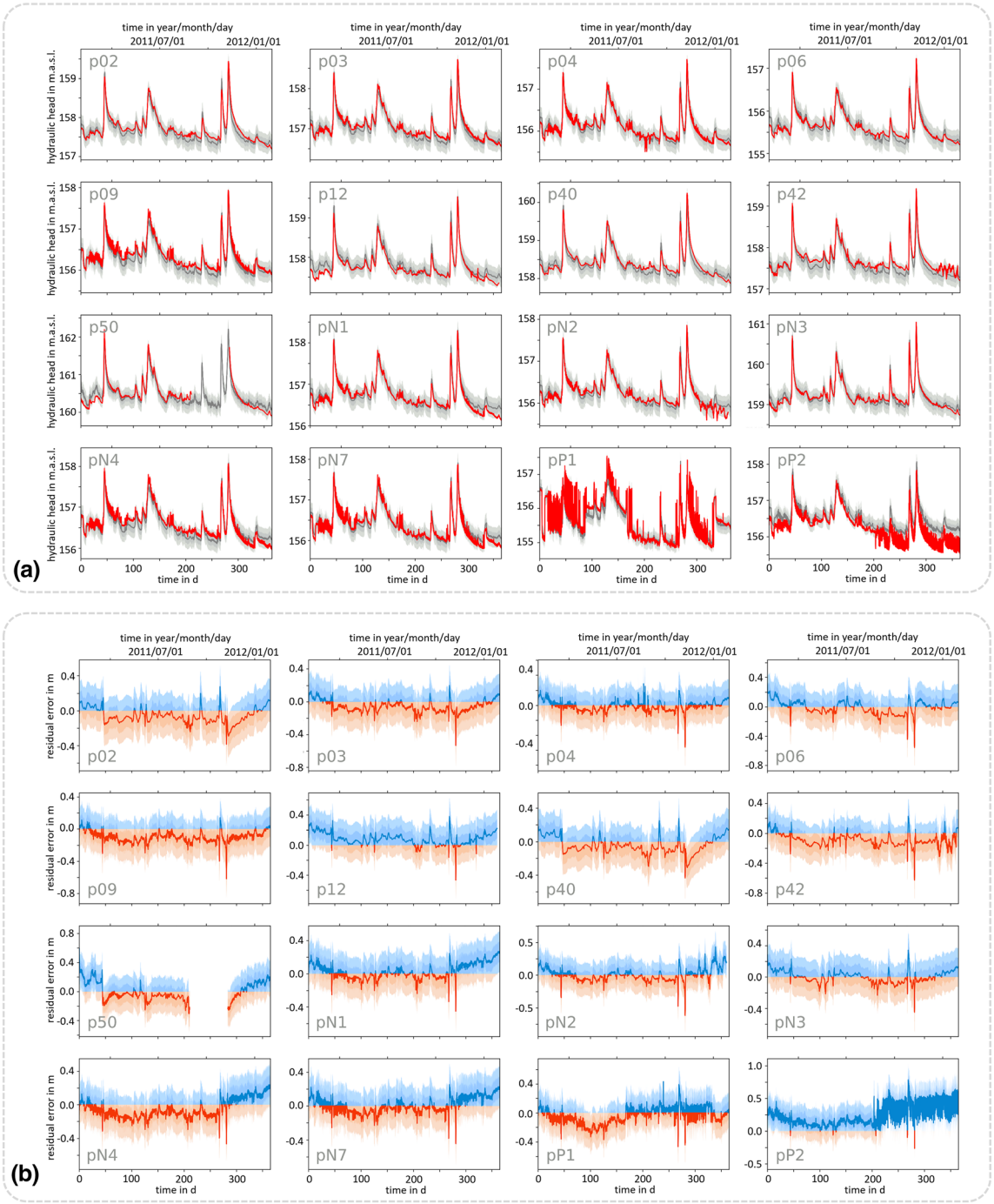

Figure 7. Evaluation of the model predictions of Scenario 1c over the assimilation period for all observation and pumping wells. (a) The observations (red line) and simulated mean (dark gray line) with single (gray area) and double (light gray area) lumped error standard deviation. (b) The corresponding residual errors, colored according to positive (blue) and negative (red) residual components. Equivalent figures for the other scenarios and periods, as well as videos illustrating the evolution of the model prediction during assimilation, are available in Figure S3 to S11.

matching the similar performance metrics obtained in Section 4.1, yielding final values between 0.13 and 0.15 m. The average hydraulic conductivity of the paleo-channel facies (Figure 8e) is strongly constrained by its limits, but Scenarios 1 and 3 converge, on average, toward a slightly lower value than Scenario 2. This might be rooted in the lower channel-to-background ratio of the alternative MPS training image in Scenario 2. The average conductivity of the background (Figure 8f) is similar across all scenarios,

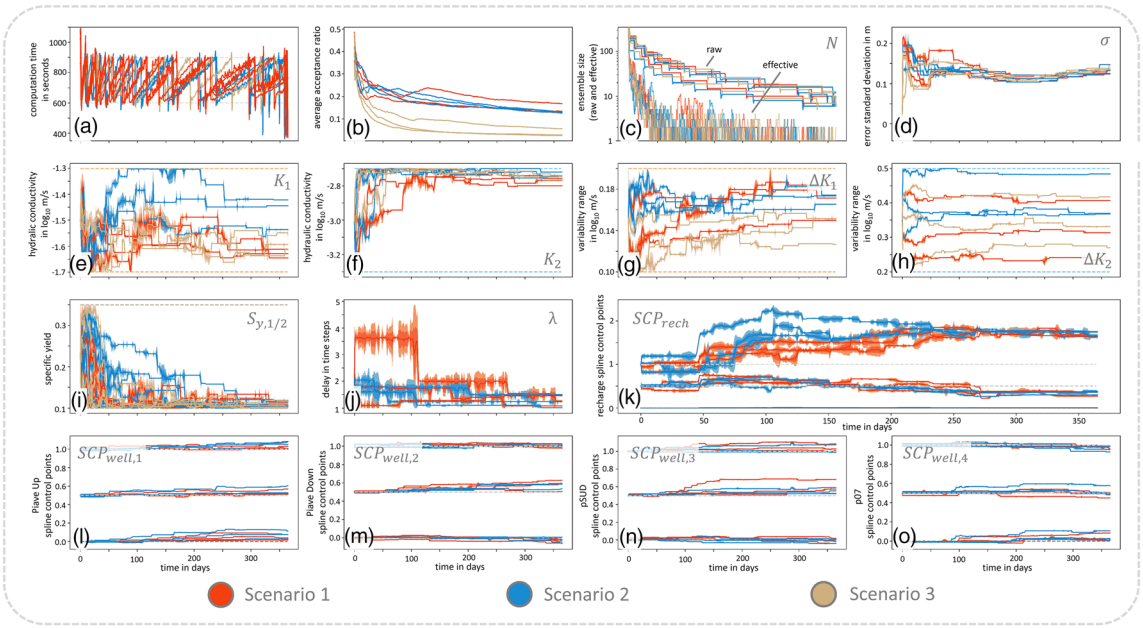

Figure 8. Metaparameter and hyperparameter for all scenarios over time: (a) computation time per day, (b) time-averaged proposal acceptance ratio, (c) raw and effective particle count, (d) error standard deviation, (e) mean hydraulic conductivity for the paleo-channels and (f) the background sediment, (g and h) their respective internal variability ranges, (i) specific yield for both facies, (j) recharge delay, and (k) spline control points for recharge and the boundary wells (l) Piave Up, (m) Piave Down, (n) pSUD, and (o) p07. Single and double standard deviations of hyperparameter uncertainty are plotted in lighter shades, where applicable. Individual corresponding hyperparameter plots are available in the Figure S12 to S20. Subplots (a) through (c) are reproduced in greater detail in Appendix S21.

converging against the upper bound. The internal variability ranges of the paleo-channel (Figure 8g) and background facies (Figure 8h) show no significant trend across the different scenarios. Specific yield (Figure 8i) for both facies quickly converges toward the lower bound across all scenarios, likely to achieve the simulated drawdown. The forcing-related hyperparameters (Figure 8j to Figure 8o) are only used in Scenarios 1 and 2. The parameter for recharge delay seems to be unnecessary, as it converges toward the lower bound of 1 (i.e., no delay) for both scenarios. The recharge spline control points (Figure 8k) seemingly feature a common optimum: the upper control point converges to values around 1.65 for both scenarios, whereas the central control point drops to a value of 0.35. Physically interpreted, these values increase the flashiness of the recharge. For the boundary wells, the spline control points display no clear, significant optimum but seem to faintly increase the hydraulic head in the river-based wells. Combining this finding with the increased proposal acceptance rate in Scenarios 1 and 2 (Figure 8b) and the overall greater degree of uncertainty of recharge-related parameters (Figures 8j and 8k) suggests that updates to the recharge control parameters have only moderate effect on the predictions. Updates to the hydraulic parameters, the model error, or the boundary well control points seem to induce larger changes, resulting in lower acceptance rates.

4.3. Hyperparameter Fields

Since conformance to a prescribed geology was a major objective of this study, investigating the hydraulic conductivity and facies maps at the end of the assimilation period (Figure 9) can reveal interesting insights into the optimization process.

First off, the different characteristics of the two training images become clear already when investigating the prior conductivity maps (Figures 9a, 9d, and 9g): the smaller, local training image (Scenarios 1 and 3) generates fewer distinct variations of its patterns and thus displays clear preferences for prior channel placement, whereas the larger, alternative training image (Scenario 2) contains a wider range of possible channel constellations, resulting in much less preferential initial facies assignments.

10.1029/2019WR026777

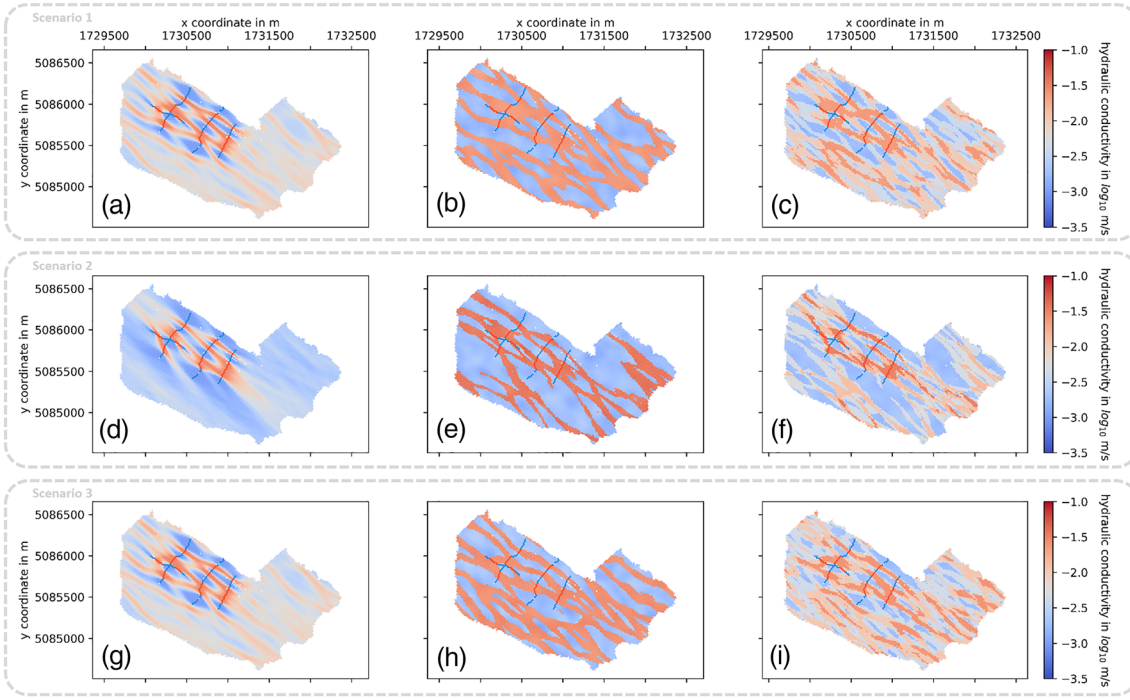

Figure 9. Representative ensemble mean parameter fields of (a–c) Scenarios 1, (d–f) 2, and (g–i) 3 at the start (a, d, and g) and end (b, e, and h) of the data assimilation period, for the third random seed (suffix c) each. The right-most column (c, f, and i) depicts the mean conductivity field across all three random seeds for each scenario.

Investigating the expected conductivity fields at the end of the assimilation period (Figures 9b, 9e, and 9h) reveals that the realizations retained connected features through the optimization process. We further note differences in the mean hydraulic conductivities: as suspected in section 4.1, Scenario 2 indeed identifies higher facies conductivities to compensate for its lower channel-to-background facies ratio.

The final column (Figures 9c, 9f, and 9i) serves to reveal any potential common facies assignments across the different random seeds, scenarios, and training images. Evidently, it seems that there is no clear preference aside from the prescribed ERT constraints. This can partially be explained in the relatively low information content that water tables alone provide for the placement of preferential flow paths.

However, we noted that changes to the facies distribution map have significant impact on the magnitude of the likelihood ratios (Figures S22 to S24), suggesting that the facies distribution is far from insensitive. Combining this observation with the lack of structural uncertainty in each individual scenario, and the different constellations identified across all scenarios, suggests the presence of multiple, isolated optima at least in the facies distribution hyperparameter-subspace.

5. Discussion

Summarizing the results, we found that in all scenarios considered, the proposed quasi-online optimizer proves capable of identifying promising hyperparameter sets, while honoring a complex geological prior as prescribed by a training image. Simulations in triplicate suggest that the algorithm reliably identifies global optima for most of the scalar hyperparameters, barring the tightly constrained internal variability ranges *K*s. The boundary spline control points deviate slightly but do not display a clear preference toward a global optimum either. This might be owed to interactions with the facies distribution map, which seems characterized by multiple, isolated optima.

Despite restricting the optimization to the support of the geological prior, we achieved results similar or even favorable to Zovi et al. (2017). Over the validation period, they obtained $\overline{\text{RMSE}}$s of 0.155 m without and

0.302 m with normal-score transform using the ensemble mean. However, their best-performing scenario could maintain only limited conformance to the geological structure, with no clear identification of the paleo-channels. Our most similar setup—Scenario 3, without the forcing model—yielded $\overline{\text{RMSE}}$s between 0.153 and 0.165 m, with a plausible paleo-channel pattern that honors the available geophysical data. We found that further improvements could be made by introducing a (hyper)parameterized forcing model, which reduced $\overline{\text{RMSE}}$s down to values between 0.128 and 0.142 m (Scenario 1). Its introduction permitted an increase of recharge intensity during precipitation events (Figure 8j), which brought simulated heads closer to the observations (Table S3). We note, however, that we could not use the exact same prior as Zovi et al. (2017) and acknowledge that their EnKF implementation was more computationally efficient (requiring only 30 wall-clock hours on a four-core machine) than our algorithm.

Combining the forcing model with a different training image (Scenario 2) slightly deteriorates the performance metrics relative to Scenario 1, yielding $\overline{\text{RMSE}}$s between 0.139 and 0.168 m. An analysis of the residuals (Figures 7 and S3 to S11) suggests that a more flexible forcing model—capable of transforming not only the raw input data but adding seasonal effects—might be necessary to attain further improvements. A possible explanation for this apparent seasonality might be recharge from irrigation, a practice known to be used in this region (D'Agata, 2019; Zovi, 2014) but for which no data are available. D'Agata (2019) mentions compacted sands with "low water reserves" as the cause for the need for irrigation, which might match the low S_y identified across all scenarios. Further simulation inadequacies are the large RMSE^{o}s in pumping well pP2, which might be addressed by permitting the model to consider well skin effects.

A limitation of this study is ensemble collapse. Across all scenarios, the effective ensemble size is about 1 order of magnitude smaller than the raw ensemble size (Figure 8c). This results in an underestimation of parameter uncertainty, the potential loss of separate posterior modes, and exacerbates the risk of entrapment in a local optimum. A quick back-of-the-envelope calculation reveals that this is a consequence of highly tapered posterior modes, in turn a result of our error model definition:

Assume two hyperparameter sets creating predictions of different fidelity: the first one's predictions are always 0.01 m off the observations in each well and the other's always 0.02 m. Further assume that $\sigma = 0.15$ m and that we assimilate data in 16 wells every 3 hr for a year ($365 \cdot 8 \cdot 16 = 46720$ observations). The log-likelihood difference between the two parameter sets would be -311.5. This effect becomes more pronounced for greater deviations from the observations.

This means that even for small differences in performance, the rejuvenation mechanism accepts only strict improvements. Combining this finding with the presence of isolated optima indicated by the results across different random seeds suggests that the posterior might be both highly tapered and multimodal, which renders the identification of multiple modes at once with a particle filter highly challenging.

We argue that ensemble collapse—while clearly undesirable—has less irreversible repercussions for the IBIS algorithm than for classic particle filters. Without relying on the fidelity of the ensemble approximation, the rejuvenation mechanism samples the posterior directly. This critical property permits the algorithm to recover samples from the true posterior even after ensemble collapse. As possible remedies, the steep divergence of the likelihoods (see Footnote 1) could be slowed through adjustments to the error model, such as the consideration of temporal and spatial correlations or the use of a likelihood function with heavier tails (e.g., a Cauchy distribution: P. J. van Leeuwen, 2003). From an algorithmic perspective, adjustments to the MH-MCMC proposal—for example, the introduction of elements from evolutionary algorithms (e.g., Abbaszadeh et al., 2018; Zhu et al., 2018)—might permit the proposal to capitalize on information of other ensemble members to better identify multiple optima. We could also adapt the proposal distribution dynamically during assimilation in order to adjust acceptance rates. Such efforts, however, would be complicated by the conflicting findings of sections 4.1 and 4.2: particularly for the hydraulic parameters, larger proposals may be required to escape local minima, and smaller proposals may be required to increase the proposal acceptance rate. This contradiction is a well-known challenge in MCMC literature (e.g., Foreman-Mackey et al., 2013; Holmes et al., 2017; Tjelmeland & Hegstad, 2001).

Alternatively, ad-hoc solutions based on artificial variance inflation (e.g., Moradkhani et al., 2005) remain the most efficient remedy for the negative repercussions of ensemble collapse. Such methods might prevent or even reverse the uncontrolled tapering of probability densities. Unfortunately, despite their pragmatic

allure, such approaches invariably corrupt the posterior (e.g., Vrugt et al., 2013) and should thus be used with caution.

However, it may be an interesting direction for future research to explore the interaction of variance inflation through artificial random parameter dynamics (e.g., Moradkhani et al., 2005; Ramgraber et al., 2019) with the rejuvenation mechanism used in this study. While the indiscriminate addition of random components will corrupt the posterior, we expect that the inclusion of MCMC steps might limit posterior drift.

We believe that results of this study demonstrate the ability of the IBIS algorithm to optimize complex hydrogeological models under field conditions, although the loss of parameter uncertainty remains a major concern. The inclusion of a forcing model furthermore extended the potential for bias correction from the grid parameters to boundary conditions. This improved results slightly and permitted a more detailed diagnosis of residual model inadequacies. Considering the prevailing uncertainties in meteorological and hydrological forcing, we advocate the careful use of such preprocessors as a valuable extension to the conventional scope of hydrogeological parameter inference, if designed with physical processes or error-diagnostic capabilities in mind.

6. Conclusions

Reconciling the challenges posed by complex geological priors with the operational limitations of online optimization frameworks is not a trivial task, but a necessary endeavor if such methods should ever find use outside of simple settings. In pursuit of such a framework, we presented a quasi-online optimizer based on the IBIS algorithm. Instead of optimizing grid parameters directly, we updated a set of hyperparameters, which parameterized a number of preprocessors. We used a field generator built around MPS facies maps and the flexibility of the MH-MCMC rejuvenation mechanism to maintain conformance with a non-trivial geological prior by construction. We further introduced another preprocessor to reduce forcing-related biases and inadequacies. We demonstrated the performance of the algorithm with data from a field site in northern Italy.

Optimization results were promising, identifying similar hyperparameter optima across all scenarios. Despite remaining confined to the support of the geological prior, the performance metrics we obtained revealed equivalent or even superior performance to a previous study using the EnKF (Zovi et al., 2017). The use of a local training image as well as the forcing model both improved the predictions. Analysis of the error residuals further suggests forcing-related inadequacies, particularly omitted seasonal effects possibly linked to irrigation, which our preprocessor was not equipped to compensate. Allowing the forcing model to correct for unaccounted seasonality might be required to achieve further improvements.

To conclude, we believe that the ability to sequentially optimize parameter fields with nontrivial priors while providing estimates of predictive uncertainty could prove a valuable asset to practitioners in the future. Considering the larger time window available for an assimilation step in practice—24 hr as opposed to 15 min—we believe that there is sufficient computational space for larger ensembles or more complex models. A possible limitation of this study is the lack of structural uncertainty in the MPS facies distribution map, despite the identification of separate, functionally similar optima across different random seeds. This suggests the presence of multiple, isolated, highly tapered optima in the posterior pdf, which prove a highly challenging task for most Bayesian inference algorithms. We suggested adjustments to the error model, the proposal function, or variance inflation as possible remedies. Continuing research could investigate some of these avenues in order to provide better structural uncertainty estimates. Combining the forcing model with information related to vegetation activity might furthermore permit estimates of irrigation activity, a factor which is otherwise difficult to quantify.

Acknowledgments
We express our gratitude to Julien Straubhaar, University of Neuchâtel, for providing support during the implementation of DeeSse. We furthermore thank Peter Reichert, Carlo Albert, and Andreas Scheidegger, Swiss Federal Institute of Aquatic Science and Technology (Eawag), for providing valuable advice during the conceptualization phase of this study. Finally, the first author would like to thank the Department of Civil and Environmental Engineering, University of Padova, for hosting him during a secondment. The research leading to these results has received funding from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement 675120. The data and output files for this study are deposited under the following DOI: 10.25678/0001e9 (Note that this DOI is reserved but will only link to the data once the review process is concluded; for reviewers, the data can be accessed here: https://drive.switch.ch/index.php/s/HnO6EiMzIIarqJl).

References

Aanonsen, S. I., Nævdal, G., Oliver, D. S., Reynolds, A. C., & Vallès, B. (2009). The ensemble Kalman filter in reservoir engineering—A review. *SPE Journal*, *14*(03), 393–412. https://doi.org/10.2118/117274-PA

Abbaszadeh, P., Moradkhani, H., & Yan, H. (2018). Enhancing hydrologic data assimilation by evolutionary particle filter and Markov chain Monte Carlo. *Advances in Water Resources*, *111*, 192–204. https://doi.org/10.1016/j.advwatres.2017.11.011

Alcolea, A., & Renard, P. (2010). Blocking moving window algorithm: Conditioning multiple-point simulations to hydrogeological data. *Water Resources Research*, *46*, W08511. https://doi.org/10.1029/2009WR007943

Baker, J. E. (1987). Reducing bias and inefficiency in the selection algorithm. Second International Conference on Genetic Algorithms and their Application, 206, 260. https://doi.org/10.1136/bmj.39184.617049.80

Bakker, M., Post, V., Langevin, C. D., Hughes, J. D., White, J. T., Starn, J. J., & Fienen, M. N. (2016). Scripting MODFLOW model development using Python and FloPy. *Groundwater*, *54*(5), 733–739. https://doi.org/10.1111/gwat.12413

Barrash, W., Clemo, T., Fox, J. J., & Johnson, T. C. (2006). Field, laboratory, and modeling investigation of the skin effect at wells with slotted casing, Boise Hydrogeophysical Research Site. *Journal of Hydrology*, *326*(1-4), 181–198. https://doi.org/10.1016/j.jhydrol.2005.10.029

Burri, N. M., Weatherl, R., Moeck, C., & Schirmer, M. (2019). A review of threats to groundwater quality in the anthropocene. *Science of the Total Environment*, *684*, 136–154. https://doi.org/10.1016/j.scitotenv.2019.05.236

Caers, J., Strebelle, S., & Payrazyan, K. (2003). Stochastic integration of seismic data and geologic scenarios: A West Africa submarine channel saga. *The Leading Edge*, *22*(3), 192–196. https://doi.org/10.1190/1.1564521

Chopin, N. (2002). A sequential particle filter method for static models. *Biometrika*, *89*(3), 539–552. https://doi.org/10.1093/biomet/89.3.539

Chopin, N., Jacob, P. E., & Papaspiliopoulos, O. (2013). SMC2: An efficient algorithm for sequential analysis of state space models. *Journal of the Royal Statistical Society, Series B: Statistical Methodology*, *75*(3), 397–426. https://doi.org/10.1111/j.1467-9868.2012.01046.x

Cirpka, O. A., & Valocchi, A. J. (2016). Debates—Stochastic subsurface hydrology from theory to practice: Does stochastic subsurface hydrology help solving practical problems of contaminant hydrogeology? *Water Resources Research*, *52*, 9218–9227. https://doi.org/10.1002/2016WR019087

Crestani, E., Camporese, M., Baú, D., & Salandin, P. (2013). Ensemble Kalman filter versus ensemble smoother for assessing hydraulic conductivity via tracer test data assimilation. *Hydrology and Earth System Sciences*, *17*(4), 1517–1531. https://doi.org/10.5194/hess-17-1517-2013

D'Agata, I. (2019). *Italy's Native Wine Grape Terroirs*. California: University of California Press.

Doherty, J. E., Fienen, M. N., & Hunt, R. J. (2010). Approaches to highly parameterized inversion: Pilot-point theory, guidelines, and research directions. Consume hasta Morir. US Geological Survey. https://doi.org/10.5194/2010-5168

Doucet, A., & Johansen, A. M. (2009). A tutorial on particle filtering and smoothing: Fifteen years later. Handbook of Nonlinear Filtering.

Doucet, A., & Tadić, V. B. (2003). Parameter estimation in general state-space models using particle methods. In Annals of the Institute of Statistical Mathematics. https://doi.org/10.1023/A:1026390323638

Evensen, G. (1994). Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *Journal of Geophysical Research*, *99*(C5), 10,110–13,162. https://doi.org/10.1029/94JC00572

Evensen, G. (2003). The ensemble Kalman filter: Theoretical formulation and practical implementation. *Ocean Dynamics*, *53*(4), 343–367. https://doi.org/10.1007/s10236-003-0036-9

Fogg, G. E., & Zhang, Y. (2016). Debates—Stochastic subsurface hydrology from theory to practice: A geologic perspective. *Water Resources Research*, *52*, 9235–9245. https://doi.org/10.1002/2016WR019699

Foreman-Mackey, D., Hogg, D. W., Lang, D., & Goodman, J. (2013). emcee: The MCMC Hammer. *Publications of the Astronomical Society of the Pacific*. https://doi.org/10.1086/670067

Gorelick, S. M., & Zheng, C. (2015). Global change and the groundwater management challenge. *Water Resources Research*, *51*, 3031–3051. https://doi.org/10.1002/2014WR016825

Gu, Y., & Oliver, D. S. (2007). An iterative ensemble Kalman filter for multiphase fluid flow data assimilation. *SPE Journal*, *12*(4), 438–446. https://doi.org/10.2118/108438-pa

Gupta, H. V., Kling, H., Yilmaz, K. K., & Martinez, G. F. (2009). Decomposition of the mean squared error and NSE performance criteria: Implications for improving hydrological modelling. *Journal of Hydrology*, *377*(1-2), 80–91. https://doi.org/10.1016/j.jhydrol.2009.08.003

Hendricks Franssen, H. J., & Kinzelbach, W. (2008). Real-time groundwater flow modeling with the ensemble Kalman filter: Joint estimation of states and parameters and the filter inbreeding problem. *Water Resources Research*, *44*, W09408. https://doi.org/10.1029/2007WR006505

Holmes, C., Krzystof, L., & Pompe, E. (2017). Adaptive MCMC for multimodal distributions. Retrieved from https://pdfs.semanticscholar.org/c75d/f035c23e3c0425409e70d457cd43b174076f.pdf

Hu, L. Y., Zhao, Y., Liu, Y., Scheepens, C., & Bouchard, A. (2013). Updating multipoint simulations using the ensemble Kalman filter. *Computers & Geosciences*, *51*, 7–15. https://doi.org/10.1016/j.cageo.2012.08.020

Jafarpour, B., & McLaughlin, D. B. (2009). Estimating channelized-reservoir permeabilities with the ensemble Kalman filter: The importance of ensemble design. *SPE Journal*, *14*. https://doi.org/10.2118/108941-pa

Journel, A., & Zhang, T. (2006). The necessity of a multiple-point prior model. *Mathematical Geology*, *38*(5), 591–610. https://doi.org/10.1007/s11004-006-9031-2

Katzfuss, M., Stroud, J. R., & Wikle, C. K. (2016). Understanding the ensemble Kalman filter. *The American Statistician*, *70*(4), 350–357. https://doi.org/10.1080/00031305.2016.1141709

Kresic, N., & Stevanovic, Z. (2010). Groundwater hydrology of springs. https://doi.org/10.1016/C2009-0-19145-6

Langevin, C. D., Hughes, J. D., Banta, E. R., Niswonger, R. G., Panday, S., & Provost, A. M. (2017). Documentation for the MODFLOW 6 groundwater flow model. U.S. Geological Survey. https://doi.org/10.3133/tm6A55

Li, T., Bolić, M., & Djurić, P. M. (2015). Resampling methods for particle filtering: Classification, implementation, and strategies. IEEE Signal Processing Magazine. https://doi.org/10.1109/MSP.2014.2330626

Linde, N., Renard, P., Mukerji, T., & Caers, J. (2015). Geological realism in hydrogeological and geophysical inverse modeling: A review. *Advances in Water Resources*, *86*, 86–101. https://doi.org/10.1016/j.advwatres.2015.09.019

Mariethoz, G., Renard, P., & Straubhaar, J. (2010). The direct sampling method to perform multiple-point geostatistical simulations. *Water Resources Research*, *46*, W11536. https://doi.org/10.1029/2008WR007621

Mariethoz, G., Straubhaar, J., Renard, P., Chugunova, T., & Biver, P. (2015). Constraining distance-based multipoint simulations to proportions and trends. *Environmental Modelling and Software*, *72*, 184–197. https://doi.org/10.1016/j.envsoft.2015.07.007

McLaughlin, D., & Townley, L. R. (1996). A reassessment of the groundwater inverse problem. *Water Resources Research*, *32*(5), 1131–1161. https://doi.org/10.1029/96WR00160

Moradkhani, H., Dechant, C. M., & Sorooshian, S. (2012). Evolution of ensemble data assimilation for uncertainty quantification using the particle filter-Markov chain Monte Carlo method. *Water Resources Research*, *48*, W12520. https://doi.org/10.1029/2012WR012144

Moradkhani, H., Hsu, K.-L., Gupta, H., & Sorooshian, S. (2005). Uncertainty assessment of hydrologic model states and parameters: Sequential data assimilation using the particle filter. *Water Resources Research*, *41*, W05012. https://doi.org/10.1029/2004WR003604

Noh, S. J., Tachikawa, Y., Shiiba, M., & Kim, S. (2011). Applying sequential Monte Carlo methods into a distributed hydrologic model: Lagged particle filtering approach with regularization. *Hydrology and Earth System Sciences*, *15*(10), 3237–3251. https://doi.org/10.5194/hess-15-3237-2011

Oude Essink, G. H. P. (2001). Salt water intrusion in a three-dimensional groundwater system in the Netherlands: A numerical study. Transport in Porous Media. https://doi.org/10.1023/A:1010625913251

Panday, S., Langevin, C. D., Niswonger, R. G., Ibaraki, M., & Hughes, J. D. (2013). MODFLOW – USG Version 1: An unstructured grid version of MODFLOW for simulating groundwater flow and tightly coupled processes using a control volume finite-difference formulation. *U.S. Geological Survey*, (Techniques and Methods 6-A45).

Powers, J. P., Corwin, A. B., Schmall, P. C., & Kaeck, W. E. (2007). Construction dewatering and groundwater control: Bew methods and applications. Construction Dewatering and Groundwater Control: New Methods and Applications. https://doi.org/10.1002/9780470168103

Ramgraber, M., Albert, C., & Schirmer, M. (2019). Data assimilation and online parameter optimization in groundwater modeling using nested particle filters. *Water Resources Research*, *55*, 9724–9747. https://doi.org/10.1029/2018WR024408

Reichle, R. H., McLaughlin, D. B., & Entekhabi, D. (2002). Hydrologic data assimilation with the ensemble Kalman filter. *Monthly Weather Review*, *130*, 103–114. https://doi.org/10.1175/1520-0493(2002)130<0103:hdawte>2.0.co;2

Reilly, T. E., & Harbaugh, A. W. (2004). Guidelines for evaluating ground-water flow models. Scientific Investigations Report 2004–5038. https://doi.org/10.1017/CBO9781107415324.004

Sanchez-Vila, X., & Fernàndez-Garcia, D. (2016). Debates—Stochastic subsurface hydrology from theory to practice: Why stochastic modeling has not yet permeated into practitioners? *Water Resources Research*, *52*, 9246–9258. https://doi.org/10.1002/2016WR019302

Schirmer, M., Leschik, S., & Musolff, A. (2013). Current research in urban hydrogeology—A review. *Advances in Water Resources*, *51*, 280–291. https://doi.org/10.1016/j.advwatres.2012.06.015

Shepard, D. (1968). A two-dimensional interpolation function for irregularly-spaced data. In Proceedings of the 1968 23rd ACM national conference on -. https://doi.org/10.1145/800186.810616

Snyder, C., Bengtsson, T., & Morzfeld, M. (2015). Performance bounds for particle filters using the optimal proposal. *Monthly Weather Review*, *143*(11), 4750–4761. https://doi.org/10.1175/mwr-d-15-0144.1

Spearman, C. (1987). The proof and measurement of association between two things. *The American Journal of Psychology*, *100*(3/4), 441. https://doi.org/10.2307/1422689

Straubhaar, J. (2019). DeeSse user's guide.

Sun, A. Y., Morris, A. P., & Mohanty, S. (2009). Sequential updating of multimodal hydrogeologic parameter fields using localization and clustering techniques. *Water Resources Research*, *45*, W07424. https://doi.org/10.1029/2008WR007443

Sun, N.-Z. (1994). Inverse problems in groundwater modeling. Theory and applications of transport in porous media.

Tang, Q., Kurtz, W., Brunner, P., Vereecken, H., & Hendricks Franssen, H. J. (2015). Characterisation of river-aquifer exchange fluxes: The role of spatial patterns of riverbed hydraulic conductivities. *Journal of Hydrology*, *531*, 111–123. https://doi.org/10.1016/j.jhydrol.2015.08.019

Tang, Q., Kurtz, W., Schilling, O. S., Brunner, P., Vereecken, H., & Hendricks Franssen, H. J. (2017). The influence of riverbed heterogeneity patterns on river-aquifer exchange fluxes under different connection regimes. *Journal of Hydrology*, *554*, 383–396. https://doi.org/10.1016/j.jhydrol.2017.09.031

Tjelmeland, H., & Hegstad, B. K. (2001). Mode jumping proposals in MCMC. *Scandinavian Journal of Statistics*, *28*(1), 205–223. https://doi.org/10.1111/1467-9469.00232

Townsend, A. A. R. (2003). Genetic algorithms—A tutorial. Retrieved from https://pdfs.semanticscholar.org/eccb/f6523d2d29a5f6dbed9-d7a0210e5ded49b96.pdf

UNICEF. (2016). Professional water well drilling: A UNICEF guidance note. Retrieved from http://skat.ch/wp-content/uploads/2017/04/UNICEF_GuidanceNote_ProfessionalWellDrilling.pdf

van Leeuwen, P. J. (2003). A variance-minimizing filter for large-scale applications. *Monthly Weather Review*, *131*(9), 2071–2084. https://doi.org/10.1175/1520-0493(2003)131<2071:AVFFLA>2.0.CO;2

van Leeuwen, P. J. (2009). Particle filtering in geophysical systems. *Monthly Weather Review*, *137*(12), 4089–4114. https://doi.org/10.1175/2009MWR2835.1

van Leeuwen, P. J. (2015). Representation errors and retrievals in linear and nonlinear data assimilation. *Quarterly Journal of the Royal Meteorological Society*, *141*(690), 1612–1623. https://doi.org/10.1002/qj.2464

van Leeuwen, P. J., Künsch, H. R., Nerger, L., Potthast, R., & Reich, S. (2019). Particle filters for high-dimensional geoscience applications: A review. *Quarterly Journal of the Royal Meteorological Society*, *145*(723), 2335–2365. https://doi.org/10.1002/qj.3551

Vrugt, J. A., ter Braak, C. J. F., Diks, C. G. H., & Schoups, G. (2013). Hydrologic data assimilation using particle Markov chain Monte Carlo simulation: Theory, concepts and applications. *Advances in Water Resources*, *51*, 457–478. https://doi.org/10.1016/j.advwatres.2012.04.002

Werner, A. D., Bakker, M., Post, V. E. A., Vandenbohede, A., Lu, C., Ataie-Ashtiani, B., et al. (2013). Seawater intrusion processes, investigation and management: Recent advances and future challenges. *Advances in Water Resources*, *51*, 3–26. https://doi.org/10.1016/j.advwatres.2012.03.004

Yeh, W. W. (1986). Review of parameter identification procedures in groundwater hydrology: The inverse problem. *Water Resources Research*, *22*(2), 95–108. https://doi.org/10.1029/WR022i002p00095

Zhou, H., Gómez-Hernández, J. J., Hendricks Franssen, H. J., & Li, L. (2011). An approach to handling non-Gaussianity of parameters and state variables in ensemble Kalman filtering. *Advances in Water Resources*, *34*(7), 844–864. https://doi.org/10.1016/j.advwatres.2011.04.014

Zhu, G., Li, X., Ma, J., Wang, Y., Liu, S., Huang, C., et al. (2018). A new moving strategy for the sequential Monte Carlo approach in optimizing the hydrological model parameters. *Advances in Water Resources*, *114*, 164–179. https://doi.org/10.1016/j.advwatres.2018.02.007

Zovi, F. (2014). Assessment of heterogeneous hydraulic properties in natural aquifers at the intermediate scale. *PhD thesis*. University of Padova. Retrieved from http://paduaresearch.cab.unipd.it/6708/

Zovi, F., Camporese, M., Hendricks Franssen, H. J., Huisman, J. A., & Salandin, P. (2017). Identification of high-permeability subsurface structures with multiple point geostatistics and normal score ensemble Kalman filter. *Journal of Hydrology*, *548*, 208–224. https://doi.org/10.1016/j.jhydrol.2017.02.056