

Technical Note

Not peer-reviewed version

---

# Uncertainty Quantification in Coastal Aquifers Using the Multilevel Monte Carlo Method

---

[Alexander Litvinenko](#) <sup>\*</sup>, Dmitry Logashenko, [Raul Tempone](#), Ekaterina Vasilyeva, Gabriel Wittum

Posted Date: 21 February 2023

doi: [10.20944/preprints202302.0359.v1](https://doi.org/10.20944/preprints202302.0359.v1)

Keywords: uncertainty quantification; multigrid; multilevel; density-driven flow; salinization; coastal aquifers; groundwater; salt formations



Preprints.org is a free multidiscipline platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

## Article

# Uncertainty Quantification in Coastal Aquifers Using the Multilevel Monte Carlo Method

Alexander Litvinenko <sup>1</sup>, Dmitry Logashenko <sup>2</sup>, Raul Tempone <sup>1,2</sup>, Ekaterina Vasilyeva <sup>3</sup> and Gabriel Wittum <sup>2</sup>

<sup>1</sup> RWTH Aachen, Aachen, Germany; litvinenko@uq.rwth-aachen.de; raul.tempone@kaust.edu.sa

<sup>2</sup> KAUST, Thuwal-Jeddah, Saudi Arabia; dmitry.logashenko@kaust.edu.sa; gabriel.wittum@kaust.edu.sa

<sup>3</sup> Goethe-Universität Frankfurt am Main, Germany; ekaterina.vasilyeva@gcsc.uni-frankfurt.de

**Abstract:** We consider a class of density-driven flow problems. We are particularly interested in the problem of the salinization of coastal aquifers. We consider the Henry saltwater intrusion problem with uncertain porosity, permeability, and recharge parameters as a test case. The reason for the presence of uncertainties is the lack of knowledge, inaccurate measurements, and inability to measure parameters at each spatial or time location. This problem is nonlinear and time-dependent. The solution is the salt mass fraction, which is uncertain and changes in time. Uncertainties in porosity, permeability, recharge, and mass fraction are modeled using random fields. This work investigates the applicability of the well-known multilevel Monte Carlo (MLMC) method for such problems. The MLMC method can reduce the total computational and storage costs. Moreover, the MLMC method runs multiple scenarios on different spatial and time meshes and then estimates the mean value of the mass fraction. The parallelization is performed in both the physical space and stochastic space. To solve every deterministic scenario, we run the parallel multigrid solver ug4 in a black-box fashion. We use the solution obtained from the quasi-Monte Carlo method as a reference solution.

**Keywords:** uncertainty quantification; ug4; multigrid; density-driven flow; reservoir; groundwater; salt formations

## 1. Introduction

Notation	
QoI $g$	quantity of interest $g$
$\mathcal{D}$	computational spatial domain
$\mathcal{D}_0, \mathcal{D}_1, \dots, \mathcal{D}_L$	hierarchy of spatial meshes
$\mathcal{T}_0, \mathcal{T}_1, \dots, \mathcal{T}_L$	hierarchy of temporal meshes
$L$	number of levels
$s$	complexity
$h_\ell$ (or $h$ ), $n_\ell$	spatial step size and number of spatial degrees of freedom on level $\ell$
$\tau_\ell$ (or $\tau$ ), $r_\ell$	time step size and number of time steps on level $\ell$
$m_\ell$	number of samples (scenarios) on level $\ell$
$\mathbb{E}[\cdot], \mathbb{V}[\cdot]$	expectation and variance
$\Theta$	multidimensional domain of integration in parametric space
$\omega, \xi(\omega)$	random event and random vector
$\phi(\mathbf{x}, \omega)$	porosity random field
$\mathbf{K}(\mathbf{x}, \omega)$	permeability random field
$\rho(\mathbf{x}, \omega)$	density random field
$\mathbf{q}(t, \mathbf{x}, \omega)$	volumetric velocity
$\mathbf{D}$	tensor field $\mathbf{D} = \mathbf{D}(\mathbf{q})$ : molecular diffusion and dispersion of salt
$\bar{\kappa}(\mathbf{x})$	expectation of $\kappa(\mathbf{x}, \omega)$
$d$	physical (spatial) dimension
$c = c(t, \mathbf{x}, \omega)$	mass fraction of salt (solution of the problem)

Saltwater intrusion occurs when sea levels rise and saltwater moves onto the land. Usually, this occurs during storms, high tides, droughts, or when saltwater penetrates freshwater aquifers and



raises the groundwater table. Since groundwater is an essential nutrition and irrigation resource, its salinization may lead to catastrophic consequences. Many acres of farmland may be lost because they can become too wet or salty to grow crops. Therefore, accurate modeling of different scenarios of saline flow is essential [1,2] to help farmers and researchers develop strategies to improve the soil quality and decrease saltwater intrusion effects.

Saline flow is density-driven and described by a system of time-dependent nonlinear partial differential equations (PDEs). It features convection dominance and can demonstrate very complicated behavior [3].

As a specific model, we consider a Henry-like problem with uncertain permeability and porosity. These parameters may strongly affect the flow and transport of salt. The original Henry saltwater intrusion problem was introduced by H.R. Henry in the 1960s (cf. [4]). The Henry problem became a benchmark for numerical solvers for groundwater flow (cf. [3,5–7]). In [8], the authors use the generalized polynomial chaos expansion approximation to investigate how incomplete knowledge of the system properties influences the assessment of global quantities. Particularly, they estimated the propagation of input uncertainties into a few dimensionless scalar parameters.

The hydrogeological formations typically have complicated and heterogeneous structures. These formations may consist of a few layers of porous media with various porosity and permeability coefficients (cf. [9,10]). Measurements of the layer positions and their thicknesses are only possible up to some error, and for the materials inside the layers, the average parameters are typically assumed. Thus, these layers are excellent candidates to be modeled by random fields. Further, due to the nonlinearities in the problem, averaging the parameters does not necessarily lead to the correct mathematical expectation of the solution.

To model uncertainties, we use random fields. Uncertainties in the input data propagate through the model and make the solution (e.g., the mass fraction) uncertain. An accurate estimation of the output uncertainties can facilitate a better understanding of the problem, better decisions, and improved control and design of the experiment.

The following questions can be answered:

1. How long can a particular drinking water well be used (i.e., when will the mass fraction of the salt exceed a critical threshold)?
2. What regions have especially high uncertainty?
3. What is the probability that the salt concentration is higher than a threshold at a certain spatial location and time point?
4. What is the average scenario (and its variations)?
5. What are the extreme scenarios?
6. How do the uncertainties change over time?

Many techniques can quantify uncertainties. A classical method is Monte Carlo (MC) sampling. Although it is dimension-independent, it converges very slowly and requires many samples. This method may not be affordable for time-consuming simulations. Nevertheless, even up-to-date techniques, such as surrogate models and stochastic collocation, require a few hundred to a few thousand time-consuming simulations and assume a certain smoothness by the quantity of interest (QoI).

Another class of methods is the class of perturbation methods [11]. The idea is to decompose the QoI with respect to (w.r.t.) random parameters in a Taylor series. The higher-order terms can be neglected for small perturbations, simplifying the analysis and numerics. These methods assume that random perturbations are small (e.g., up to 5% of the mean, depending on the problem). For larger perturbations, these methods usually do not work.

There are quite a few studies where authors model uncertainties in reservoirs (cf. [12,13]). Reconnecting stochastic methods with hydrogeological applications was accomplished in [14], where the authors analyzed a collaboration between academics and water suppliers in Germany and made recommendations regarding optimization and risk assessment. The fundamentals of stochastic

hydrogeology and an overview of stochastic tools and accounting for uncertainty are described in [15].

The review [16] deals with hydrogeologic applications of recent advances in uncertainty quantification, probabilistic risk assessment, and decision-making under uncertainty. The author reviewed probabilistic risk assessment methods in hydrogeology under parametric, geologic, and model uncertainties. Density-driven vertical transport of saltwater through the freshwater lens on the island of Baltrum (Germany) is modeled in [17].

In [18], the authors examined the implications of transgression for a range of seawater intrusion scenarios based on simplified coastal freshwater aquifer settings. They stated that vertical intrusion during transgressions could involve density-driven convective processes, causing substantially greater amounts of seawater to enter the aquifer and create more extensive intrusion than horizontal seawater intrusion in the absence of transgression.

The methods to compute the desired statistics of the QoI are direct integration methods, such as the MC, quasi-MC (QMC) and collocation methods and surrogate-based (generalized polynomial chaos approximation and stochastic Galerkin [19–22]) methods. Direct methods compute statistics directly by sampling uncertain input coefficients and solving the corresponding PDEs, whereas the surrogate-based method computes a cheap functional (polynomial, exponential, or trigonometrical) approximation of the QoI. Examples of the surrogate-based methods are radial basis functions [23–26], sparse polynomials [27–29], and polynomial chaos expansion [30–32]. Sparse grid methods to integrate high-dimensional integrals are considered in [31,33–40]. An idea to generate goal-oriented adaptive spatial grids and use them in the multilevel MC (MLMC) framework was presented in [41,42].

The quantification of uncertainties in stochastic PDEs could be a significant challenge due to a) the large possible number of involved random variables and b) the high cost of each deterministic solution of the governed PDE. The MC quadrature and its variance-reduced variants have a dimension-independent error convergence rate  $\mathcal{O}(N^{-\frac{1}{2}})$ , and the QMC has the worst-case rate  $\mathcal{O}(\log^M(N)N^{-1})$ , where  $N$  is the number of samples, and  $M$  indicates the dimension of the stochastic space [43]. The MC method is not affected by the dimension of the integration domain, such as collocations on sparse or full grid methods [44,45]. A numerical comparison of other QMC sequences is presented in [46].

Construction of a cheap generalized polynomial chaos expansion-based surrogate model [47–49] is an alternative to the MC method. Some well-known functions, such as the multivariate Legendre, Hermite, Chebyshev, or Laguerre functions, have been taken as a basis [47,50]. Surrogate models have pros and cons. The pros are that the model can be easily sampled once constructed, and all samples are almost free (much cheaper than sampling the original stochastic PDE). For some problem settings, sampling is unnecessary because the solution can be computed analytically (e.g., computing an integral of a polynomial). The nontrivial part of surrogate models is to define how many coefficients are needed and how accurately they should be computed. Another difficulty is that not every function can be approximated well by a polynomial. The MLMC methods do not have such limitations.

This work is structured as follows. Section 2 describes the Henry problem and numerical methods to solve it. The well-known MLMC method is reviewed in Section 3. Next, Section 4 details the numerical results, which include the numerical analysis of the Henry problem, computing different statistics, the performance of the MLMC method, and the practical performance of the parallel ug4 solver for the Henry problem [4,5] with uncertain coefficients. Finally, we conclude this work with a discussion in Section 5.

**Our contribution:** We investigate the propagation of uncertainties in the Henry-like problem. Assuming the porosity, permeability, and recharge are uncertain, we estimate the uncertainties in the density-driven flow. To reduce the high computing complexity, we applied the existing MLMC technique. We use the multigrid ug4 software library as a black-box solver, allowing us to solve the Henry problem and others (see more in [2]). We run all MLMC random simulations in parallel. To the

best of our knowledge, we are unaware of any other studies where Henry's problem [4,5] was solved using MLMC methods with uncertain porosity, permeability, and recharge parameters.

## 2. Henry Problem with Uncertain Porosity and Permeability

### 2.1. Problem setting

In coastal aquifers, salty seawater intruding on the formation on one side (the seaside) displaces the pure water due to water recharge from land sources and precipitation from the other side. Due to its higher density, seawater mainly penetrates along the bottom of the aquifer. This process can achieve a steady state but may be time-dependent due to the periodicity of the recharge or controlling the pumping rate from the wells. An accurate simulation of the salinization is vital for the prediction of water resource availability. However, the accuracy of such predictions strongly depends on the hydrogeological parameters of the formation and the geometry of the computational domain, denoted by  $\mathcal{D}$ .

The aquifer  $\mathcal{D} \subset \mathbb{R}^d$ ,  $d \in \{2, 3\}$ , can be modeled as an immobile porous matrix filled with liquid phase—a solution of salt in water. Due to the nonhomogeneous density distribution, gravitation induces the motion of the liquid phase. This motion transports the salt, which is otherwise subject to molecular diffusion.

A straightforward but very demonstrative model of coastal aquifers is the so-called Henry problem, first considered in [4]. In this two-dimensional setting, the aquifer is represented by a rectangular domain  $\mathcal{D} = [0, 2] \times [-1, 0]$  [ $\text{m}^2$ ] entirely saturated with the liquid phase (Figure 1). The salty seawater intrudes from the right side, and pure water is recharged from the left. The top and bottom are considered impermeable. Analogous settings with partially saturated domains are considered in [51].

The mass conservation laws for the entire liquid phase and salt yield the following equations

$$\partial_t(\phi\rho) + \nabla \cdot (\rho\mathbf{q}) = 0, \quad (1)$$

$$\partial_t(\phi\rho c) + \nabla \cdot (\rho c\mathbf{q} - \rho\mathbf{D}\nabla c) = 0, \quad (2)$$

where  $\phi : \mathcal{D} \rightarrow \mathbb{R}$  denotes the porosity,  $\mathbf{K} : \mathcal{D} \rightarrow \mathbb{R}^{d \times d}$  represents the permeability,  $c(t, \mathbf{x}) : [0, +\infty) \times \mathcal{D} \rightarrow [0, 1]$  is the mass fraction of the salt (or of the brine) in the solution,  $\rho = \rho(c)$  indicates the density of the liquid phase, and  $\mathbf{D}(t, \mathbf{x}) : [0, +\infty) \times \mathcal{D} \rightarrow \mathbb{R}^{d \times d}$  denotes the molecular diffusion and mechanical dispersion tensor. For the velocity  $\mathbf{q}(t, \mathbf{x}) : [0, +\infty) \times \mathcal{D} \rightarrow \mathbb{R}^d$ , we assume Darcy's law:

$$\mathbf{q} = -\frac{\mathbf{K}}{\mu}(\nabla p - \rho\mathbf{g}), \quad (3)$$

where  $p = p(t, \mathbf{x}) : [0, +\infty) \times \mathcal{D} \rightarrow \mathbb{R}$  is the hydrostatic pressure,  $\mu = \mu(c)$  denotes the viscosity of the liquid phase, and  $\mathbf{g} = (0, \dots, 0, -g)^T \in \mathbb{R}^d$  represents the gravity vector. Inserting (3) into (1–2) results in a system of two time-dependent PDEs in the unknowns  $c$  and  $p$ . This system should be closed with boundary conditions for  $c$  and  $p$  and an initial condition for  $c$ .

Following the classical setting in [4], for this variant of the Henry problem, we set

$$\rho(c) = \rho_0 + (\rho_1 - \rho_0)c, \quad \mu = \text{const} \quad (4)$$

and

$$\mathbf{D} = \phi\mathbf{D}\mathbf{I} \quad (5)$$

with a constant scalar  $D \in \mathbb{R}$ , and the identity matrix  $\mathbf{I} \in \mathbb{R}^{d \times d}$ . Furthermore, we assume the isotropic permeability

$$\mathbf{K} = K\mathbf{I}, \quad K \in \mathbb{R}.$$

This setting is consistent with the problem setting in [3]. However, we do not assume the Boussinesq approximation and keep the density variable for all terms. For the initial conditions, we set

$$c|_{t=0} = 0. \quad (6)$$

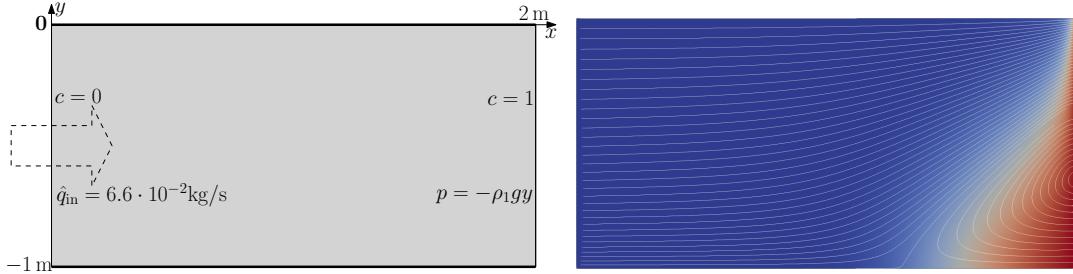
The boundary conditions are presented in Figure 1(a). On the right side of the domain, we impose Dirichlet conditions for the  $c$  and  $p$  variables that represent the adjacent seawater aquifer:

$$c|_{x=2} = 1, \quad p|_{x=2} = -\rho_1 gy. \quad (7)$$

On the left side, we prescribe the inflow of fresh water:

$$c|_{x=0} = 0, \quad \rho \mathbf{q} \cdot \mathbf{e}_x|_{x=0} = \hat{q}_{\text{in}}, \quad (8)$$

where  $\mathbf{e}_x = (1, 0)^\top$ , and  $\hat{q}_{\text{in}}$  is a constant. For the classical formulation of the Henry problem, this value was set to  $\hat{q}_{\text{in}} = 6.6 \cdot 10^{-2} \text{ kg/s}$  in [3] or  $\hat{q}_{\text{in}} = 3.3 \cdot 10^{-2} \text{ kg/s}$  in [5,6]. The Neumann zero boundary conditions are imposed on the upper and lower sides of  $\mathcal{D}$ .



**Figure 1.** (left) Computational domain  $\mathcal{D} := [0, 2] \times [-1, 0]$ . (Right) One realization of the mass fraction  $c(t, \mathbf{x})$  and the streamlines of the velocity field  $\mathbf{q}$  for the undisturbed Henry problem at  $t = 6016$  s.

An example of  $c(t, \mathbf{x})$  and  $\mathbf{q}(t, \mathbf{x})$  for the parameters from Table 1 is presented in Figure 1(right). The dark red color corresponds to  $c = 1$ , and dark blue corresponds to  $c = 0$ . Due to its higher density, the saltwater intrudes into the aquifer in the lower right part. It is pushed back by the lighter pure water coming from the left. This process induces a vortex in the flow in the lower right corner of the domain. The saltwater flows in at the lower part of the right boundary and deviates to the top and right, back to the seaside, forming a salt triangle. This flow does not transport the salt to the left part of the domain. The salt propagates further to the left due to diffusion and dispersion and is washed out by the recharge. In the classical formulation, this salt triangle initially increases over time but achieves a steady state (cf. [3,5,6]). However, the initial nonstationary phase may take significant time. Investigating this phase is especially important to understand the system behavior when changing the recharge. For this, in addition to the mean and variance, we consider the mass fraction at 12 points (listed below) and an integral value—the total amount of pure water (as in Eq. 23). The list of chosen points follows:

$$\{(x, y)_{i=1, \dots, 12}\} = \{(1.10, -0.95), (1.35, -0.95), (1.60, -0.95), (1.85, -0.95), (1.10, -0.75), (1.35, -0.75), (1.60, -0.75), (1.85, -0.75), (1.10, -0.50), (1.35, -0.50), (1.60, -0.50), (1.85, -0.50)\}. \quad (9)$$

The motivation is to consider points where the concentration variation is considerable. In addition, the mass fraction  $c$  at each point  $\mathbf{x}$  is a function of time.

**Table 1.** Parameters of the considered density-driven flow problem

Parameter	Values and Units	Description
$\mathbb{E}[\phi]$	0.35 [-]	mean value of porosity
$D$	$18.8571 \cdot 10^{-6} [\text{m}^2 \cdot \text{s}^{-1}]$	diffusion coefficient in the medium
$\mathbf{K}$	$1.020408 \cdot 10^{-9} [\text{m}^2]$	permeability of the medium
$g$	$9.8 [\text{m} \cdot \text{s}^{-2}]$	gravity
$\rho_0$	$1000 [\text{kg} \cdot \text{m}^{-3}]$	density of pure water
$\rho_1$	$1024.99 [\text{kg} \cdot \text{m}^{-3}]$	density of brine
$\mu$	$10^{-3} [\text{kg} \cdot \text{m}^{-1} \cdot \text{s}^{-1}]$	viscosity

These spatial points may help track salinity changes over time in groundwater wells and understand which areas in the aquifer are most vulnerable. Farmers can use this information to take action, such as decreasing salinity or adapting strategies by planting salt-tolerant crops.

## 2.2. Modeling porosity, permeability, and mass fraction

The primary sources of uncertainty are the hydrogeological properties of the porous medium—porosity ( $\phi$ ) and permeability ( $\mathbf{K}$ ) fields of the solid phase—and the freshwater recharge flux  $\hat{q}_x$  through the left boundary. The QoIs are related to the mass fraction  $c$ , a function of  $\phi$ ,  $\mathbf{K}$ , and the recharge. We model the uncertain  $\phi$  using a random field and assume  $\mathbf{K}$  to be isotropic and dependent on  $\phi$ :

$$\mathbf{K} = \mathbf{K}\mathbf{I}, \quad K = K(\phi) \in \mathbb{R}. \quad (10)$$

The distribution of  $\phi(\mathbf{x}, \xi)$ ,  $\mathbf{x} \in \mathcal{D}$ , is determined by a set of stochastic parameters  $\xi = (\xi_1, \dots, \xi_M, \dots)$ . Each component  $\xi_i$  is a random variable depending on a random event  $\omega$ . For concision, we skip  $\omega$  and write  $\xi := \xi(\omega)$ .

The dependence in Eq. (10) is specific for every material. We refer to [52–54] for a detailed discussion. In the proposed model, we use a Kozeny–Carman-like dependence

$$K(\phi) = \kappa_{KC} \cdot \frac{\phi^3}{1 - \phi^2}, \quad (11)$$

where the scaling factor  $\kappa_{KC}$  is chosen to satisfy the equality  $K(\mathbb{E}[\phi])\mathbf{I} = E(\mathbf{K})$ , resembling the parameters of the standard Henry problem. The inflow flux is kept constant across the left boundary but depends on the stochastic variable  $q_{\text{in}}$ . We also assume that the inflow flux is independent of  $\phi$  and  $\mathbf{K}$ .

## 2.3. Numerical methods for the deterministic problem

The system (1–2) is numerically solved in the domain  $\mathcal{D} \times [0, T]$ , where the symbol  $\times$  denotes the Cartesian product. After the discretization of  $\mathcal{D}$  using quadrilaterals of size  $h$ , we obtain  $\mathcal{D}_h$ . Equations (1–2) are discretized using a vertex-centered finite-volume scheme with a “consistent velocity” for the approximation of Darcy’s law (3), as presented in [55–57]. The degrees of freedom associated with  $\mathcal{D}_h$  are denoted by  $n$ . There are two degrees of freedom per grid vertex: one for the mass fraction and another for the pressure. We use the implicit Euler method with a fixed time step  $\tau$  for time discretization. The number of the computed time steps is  $r = T/\tau$ .

We use partial upwind for the convective terms (cf. [55]). Therefore, the discretization error is of the second order w.r.t. the spatial mesh size  $h$ . However, the diffusion in (2) is minimal compared with the velocity. For the grids in the numerical experiments, the observed reduction of the discretization error after grid refinement corresponds to the first order. Thus, we assume the first-order dependence of the discretization error w.r.t.  $h$ , which is consistent with the numerical experiments. Furthermore, the Euler method provides the first-order dependence of the discretization error w.r.t.  $\tau$ .

The implicit time-stepping scheme provides unconditional stability but requires the solution to an extensive nonlinear algebraic system of the discretized equations with  $n$  unknowns in every time step. The Newton method is used to solve this system. Linear systems inside the Newton iteration are solved using the BiCGStab method (cf. [58]) preconditioned with the geometric multigrid method (V-cycle, cf. [59]). In the multigrid cycle, the  $\text{ILU}_\beta$ -smoothers [60] and Gaussian elimination are used as the coarse grid solver.

To construct the spatial grid hierarchy  $\mathcal{D}_0, \mathcal{D}_1, \dots, \mathcal{D}_L$ , we start with a coarse grid consisting of 512 grid elements (quadrilaterals) and  $n_0 = 1122$  degrees of freedom. This grid is regularly refined to obtain all other grid levels. After every spatial grid refinement, the number of grid elements is multiplied by a factor of four. Consequently, the number of degrees of freedom is increased by a factor of four (i.e.,  $n_\ell \approx n_0 \cdot 2^{d\ell}$ ,  $d = 2$ ; see Table 2). This hierarchy is used in the geometric multigrid preconditioner and MLMC method. We also construct the temporal grid hierarchy  $\mathcal{T}_0, \mathcal{T}_1, \dots, \mathcal{T}_L$ . The time step on each temporal grid is denoted by  $\tau_\ell$  with  $\tau_{\ell+1} = \frac{1}{2}\tau_\ell$ . The number of time steps on the  $\ell$ th grid (level) is  $r_{\ell+1} = 2r_\ell$  and  $r_\ell = r_0 2^\ell$ , where  $r_0$  is the number of grid points on  $\mathcal{T}_0$ . On the  $\ell$ th level, the MLMC uses the grid  $\mathcal{D}_\ell \times \mathcal{T}_\ell$ . Up to six spatial and time grids were used in the numerical experiments.

In the context of this work, it is critical to estimate the numerical complexity of the deterministic solver w.r.t. the grid level  $\ell$ . The most time-consuming part of the simulation is the solution of the discretized nonlinear system. Typically, it is challenging to predict the number of Newton iterations in every time step, but in the numerical experiments, two iterations were sufficient to achieve the prescribed accuracy. Accordingly, the linear solver was called at most two times per time step. Furthermore, the convergence rate of the geometric multigrid method does not depend on the mesh size (cf. [60]). Hence, the computation complexity of one time step is  $\mathcal{O}(n_\ell)$ , where  $n_\ell$  is the number of the degrees of freedom on the grid level  $\ell$ . Therefore, the overall numerical cost of the computation of one scenario on grid level  $\ell$  for  $r_\ell$  time steps is

$$s_\ell = \mathcal{O}(n_\ell r_\ell), \quad s_\ell \propto s_{\ell-1} 2^{(d+1)}, \quad d = 2. \quad (12)$$

### 3. Multilevel Monte Carlo

Various numerical methods can quantify uncertainty, and every method has pros and cons. For example, the classical MC method converges slowly and requires numerous samples. To reduce the total computing cost, we apply the MLMC method, which is a natural idea because the deterministic solver uses a multigrid method (see Section 2.3). The MLMC method efficiently combines samples from various levels. Further, we repeat the main idea of the MLMC method. A more in-depth description of these techniques is found in [61–67].

We let  $\xi(\omega)$  and  $g(\xi) = g(\xi(\omega))$  represent a vector of random variables and the QoI, respectively, where  $\omega$  is a random event. The MLMC method aims to approximate the expected value  $\mathbb{E}[g]$  with an optimal computational cost. In this work,  $g$  could be  $c(t, \mathbf{x}, \xi)$  in the whole domain or at a point  $(t, \mathbf{x})$  or an integral over a subdomain. The MLMC method constructs a telescoping sum, defined over a sequence of spatial and temporal meshes,  $\ell = 0, \dots, L$ , as described next, to achieve this goal. Moreover,  $g$ , numerically evaluated on level  $\ell$ , is denoted by  $g_{h_\ell, \tau_\ell, \ell}$  or, for simplicity, by just  $g_\ell$ , where  $h_\ell$  and  $\tau_\ell$  are the discretization steps in space and time on level  $\ell$ . Further, we assume that  $\mathbb{E}[g_{h, \tau}] \rightarrow \mathbb{E}[g]$  as  $h \rightarrow 0$  and  $\tau \rightarrow 0$ .

Furthermore,  $s_0$  is the computing cost to evaluate one realization of  $g_0$  (the most expensive one from all realizations). Similarly,  $s_\ell$  denotes the computing cost of evaluating  $g_\ell - g_{\ell-1}$ . For simplicity, we assume that  $s_\ell$  for  $g_\ell - g_{\ell-1}$  is almost the same as  $s_\ell$  for  $g_\ell$ . The number of iterations is variable; thus, the cost of computing a sample of  $g_\ell - g_{\ell-1}$  may fluctuate for various realizations.

For a better understanding, we consider a two-level MLMC (cf. [64]) and estimate the optimal number of needed samples on both levels. The two-level MLMC has only two meshes: a coarse one

and a fine one. The QoI  $\mathbb{E}[g]$  can be approximated on the fine mesh by  $\mathbb{E}[g_1]$  and on the coarse mesh by  $\mathbb{E}[g_0]$ . Furthermore,

$$\mathbb{E}[g_1] = \mathbb{E}[g_0] + \mathbb{E}[g_1 - g_0] \approx m_0^{-1} \sum_{i=1}^{m_0} g_0^{(i)} + m_1^{-1} \sum_{j=1}^{m_1} (g_1^{(j)} - g_0^{(j)}), \quad (13)$$

where  $g_1^{(j)} - g_0^{(j)} := g_1(\xi_j) - g_0(\xi_j)$ ,  $\xi_j$  is a random vector, and  $m_0$  and  $m_1$  represent the numbers of quadrature points (numbers of samples/realizations) on the coarse and fine meshes, respectively. The total computational cost of evaluation (13) is  $m_0 s_0 + m_1 s_1$ . The variances of  $g_0$  and  $g_1 - g_0$  are denoted by  $V_0$  and  $V_1$ , and the total variance is obtained by  $V_0/m_0 + V_1/m_1$ , assuming that  $g_0^{(i)}$  and  $g_1^{(j)} - g_0^{(j)}$  use independent samples. By solving an auxiliary minimization problem, the variance is minimal if  $m_1 = m_0 \cdot \frac{\sqrt{V_1/s_1}}{\sqrt{V_0/s_0}}$ . Thus, with the estimates of the variances and  $m_0$ , we can estimate  $m_1$ .

The idea presented above can be extended to a case with multiple levels. Thus, we can find (quasi-) optimal numbers of samples  $m_0, m_1, \dots, m_L$ . The MLMC method calculates  $\mathbb{E}[g_L] \approx \mathbb{E}[g]$  using the following telescopic sum:

$$\mathbb{E}[g_L] = \mathbb{E}[g_0] + \sum_{\ell=1}^L \mathbb{E}[g_\ell - g_{\ell-1}] \quad (14)$$

$$\approx m_0^{-1} \sum_{i=1}^{m_0} g_0^{(0,i)} + \sum_{\ell=1}^L \left( m_\ell^{-1} \sum_{i=1}^{m_\ell} (g_\ell^{(\ell,i)} - g_{\ell-1}^{(\ell,i)}) \right). \quad (15)$$

In the above equation, level  $\ell$  in the superscript  $(\ell, i)$  indicates that independent samples are used at each correction level. As  $\ell$  increases, the variance of  $g_\ell - g_{\ell-1}$  decreases. Thus, the total computational cost can be reduced by taking fewer samples on finer meshes.

We recall that  $h_\ell = h_0 \cdot 2^{-\ell}$  and  $\tau_\ell = \tau_0 \cdot 2^{-\ell}$ . We assume that the average cost of generating one sample of  $g_\ell$  (the cost of one deterministic simulation for one random realization) is

$$s_\ell = \mathcal{O}(n_\ell r_\ell) = \mathcal{O}(h_\ell^{-1} \tau_\ell^{-1}) = \mathcal{O}\left(\frac{1}{h_0 \tau_0} 2^{2\ell} 2^\ell\right) = \mathcal{O}\left(\frac{1}{h_0 \tau_0} 2^{3\ell}\right) = \mathcal{O}\left(\frac{1}{h_0 \tau_0} 2^{(d+1)\ell\gamma}\right), \quad (16)$$

where  $d = 2$  is the spatial dimension, and  $\gamma = 1$  is determined by the computational complexity of the deterministic solver (ug4).

We let  $V_\ell$  be the variance of one sample of  $g_\ell - g_{\ell-1}$ . Then, the total cost and variance of the multilevel estimator in Eq. (14) are  $\sum_{\ell=0}^L m_\ell s_\ell$  and  $\sum_{\ell=0}^L V_\ell/m_\ell$ , respectively. For a fixed variance, the cost is minimized by choosing  $m_\ell$  to minimize the following functional for some value of the Lagrange multiplier  $\mu^2$ :

$$F(m_0, \dots, m_L) := \sum_{\ell=0}^L m_\ell s_\ell + \mu^2 \frac{V_\ell}{m_\ell}. \quad (17)$$

To determine  $m_\ell$ , we take the derivatives w.r.t.  $m_\ell$  and set them equal to zero:

$$\frac{\partial F(m_0, \dots, m_L)}{\partial m_\ell} := s_\ell - \mu^2 \frac{V_\ell}{m_\ell^2} = 0.$$

After solving the obtained equations, we obtain

$$m_\ell^2 = \mu^2 \frac{V_\ell}{s_\ell} \quad \text{and} \quad m_\ell = \mu \sqrt{\frac{V_\ell}{s_\ell}}.$$

To achieve an overall variance of  $\varepsilon^2$ , that is,

$$\sum_{\ell=0}^L V_{\ell}/m_{\ell} = \varepsilon^2,$$

we substitute  $m_{\ell}$  with the computed  $m_{\ell} = \mu \sqrt{\frac{V_{\ell}}{s_{\ell}}}$ , and obtain

$$\sum_{\ell=0}^L \frac{V_{\ell}}{\mu \sqrt{\frac{V_{\ell}}{s_{\ell}}}} = \varepsilon^2.$$

From the last equation, we obtain

$$\begin{aligned} \mu &= \varepsilon^{-2} \sum_{\ell=0}^L \sqrt{V_{\ell} s_{\ell}}, \quad \text{and} \\ m_{\ell} &= \varepsilon^{-2} \sqrt{\frac{V_{\ell}}{s_{\ell}}} \sum_{i=0}^L \sqrt{V_i s_i}. \end{aligned} \quad (18)$$

The total computational cost is  $S := \varepsilon^{-2} \left( \sum_{\ell=0}^L \sqrt{V_{\ell} s_{\ell}} \right)^2$  (for further analysis of this sum, see [64], p.4).

**Definition 1.** We let

$$\mathbb{E}[Y_{\ell}] := \begin{cases} \mathbb{E}[g_0], & \ell = 0 \\ \mathbb{E}[g_{\ell} - g_{\ell-1}], & \ell > 0 \end{cases}. \quad (19)$$

In addition,  $Y := \sum_{\ell=0}^L Y_{\ell}$  denotes a multilevel estimator of  $\mathbb{E}[g]$  based on  $L+1$  levels and  $m_{\ell}$  independent samples on level  $\ell$ , where  $\ell = 0, \dots, L$ . Moreover,  $Y_{\ell} = m_{\ell}^{-1} \sum_{i=1}^{m_{\ell}} (g_{\ell}^{(\ell,i)} - g_{\ell-1}^{(\ell,i)})$ , where  $g_{-1} \equiv 0$ .

The standard theory indicates that  $\mathbb{E}[Y] = \mathbb{E}[g_L]$ ,  $\mathbb{V}[Y_{\ell}] = \sum_{\ell=0}^L m_{\ell}^{-1} V_{\ell}$ , and  $V_{\ell} \equiv \mathbb{V}[g_{\ell} - g_{\ell-1}]$ .

The mean squared error (MSE) is used to measure the quality of the multilevel estimator:

$$\text{MSE} := \mathbb{E}[(Y - \mathbb{E}[g])^2] = \mathbb{V}[Y] + (\mathbb{E}[Y] - \mathbb{E}[g])^2. \quad (20)$$

To obtain an MSE smaller than  $\varepsilon^2$ , we ensure that both  $\mathbb{V}[Y]$  and  $(\mathbb{E}[Y] - \mathbb{E}[g])^2 = (\mathbb{E}[g_L - g])^2$  are smaller than  $\varepsilon^2/2$ . Combining this idea with a geometric sequence of levels in which the cost increases exponentially with the level while the weak error  $\mathbb{E}[g_L - g]$  and multilevel correction variance  $V_{\ell}$  decrease exponentially leads to the following theorem (cf. Theorem 1, p. 6 in [64]):

**Theorem 2.** We let  $d$  denote the problem dimension. Suppose positive constants  $\alpha, \beta, \gamma > 0$  exist such that  $\alpha \geq \frac{1}{2} \min(\beta, \gamma d)$ , and

$$|\mathbb{E}[g_{\ell} - g]| \leq c_1 2^{-\alpha \ell} \quad (21a)$$

$$V_{\ell} \leq c_2 2^{-\beta \ell} \quad (21b)$$

$$S_{\ell} \leq c_3 2^{d\gamma \ell}. \quad (21c)$$

Then, for any accuracy  $\varepsilon < e^{-1}$ , a constant  $c_4 > 0$  and a sequence of realizations  $\{m_{\ell}\}_{\ell=0}^L$  exist, such that

$$\text{MSE} := \mathbb{E}[(Y - \mathbb{E}[g])^2] < \varepsilon^2,$$

and the computational cost is

$$S = \begin{cases} \mathcal{O}(\varepsilon^{-2}), & \beta > d\gamma \\ \mathcal{O}(\varepsilon^{-2}) (\log(\varepsilon))^2, & \beta = d\gamma \\ \mathcal{O}(\varepsilon^{-\left(2+\frac{d\gamma-\beta}{\alpha}\right)}), & \beta < d\gamma. \end{cases} \quad (22)$$

This theorem (see also [61,63,68–70]) indicates that, even in the worst-case scenario, the MLMC algorithm has a lower computational cost than that of the traditional (single-level) MC method, which scales as  $\mathcal{O}(\varepsilon^{-2-d\gamma/\alpha})$ . Furthermore, in the best-case scenario presented above, the computational cost of the MLMC algorithm scales as  $\mathcal{O}(\varepsilon^{-2})$ .

Using preliminary tests, we can estimate the convergence rates  $\alpha$  for the mean (the so-called weak convergence) and  $\beta$  for the variance (the so-called strong convergence). In addition,  $\alpha$  is strongly connected to the order of the discretization error (see Section 2.3), which equals 1, and precise estimates of parameters  $\alpha$  and  $\beta$  are crucial to distribute the computational effort optimally.

#### 4. Numerical Experiments

The goal is to reduce the total computational cost of stochastic simulations. We use the MLMC method to compute the mean value of various QoIs, such as  $c$  in the whole domain,  $c$  at a point, or an integral value (we call it the freshwater integral):

$$Q_{FW}(t, \omega) := \int_{\mathbf{x} \in \mathcal{D}} I(c(t, \mathbf{x}, \omega) \leq 0.012178) d\mathbf{x}, \quad (23)$$

where  $I(\cdot)$  is the indicator function identifying a subdomain  $\{\mathbf{x} : c(t, \mathbf{x}, \omega) \leq 0.012178\}$ , meaning the mass of the fresh water at a time  $t$ . Each simulation may contain up to  $n = 0.5 \cdot 10^6$  spatial mesh points and a few thousand time steps ( $r = 6016$  on the finest mesh).

**Software and parallelization:** The computations presented in this work were performed using the ug4 simulation software toolbox (<https://github.com/ug4/ughub.wiki.git>) [71,72]. This software has been applied for subsurface flow simulations of real-world aquifers (cf. [2]). The toolbox was parallelized using MPI, and the presented results were obtained on the Shaheen II cluster provided by the King Abdullah University of Science and Technology. Every sample was computed on 32 cores of a separate cluster node. Each simulation (scenario) was localized to one node to reduce the communication time between nodes. All scenarios were concurrently computed on different nodes. A similar approach was used in [48,49]. Simulations were performed on different meshes; thus, the computation time of each simulation varied over a wide range (see Table 2).

**Porosity and recharge:** We assume two horizontal layers:  $y \in (-0.75, 0]$  (the upper layer) and  $y \in [-1, -0.75]$  (the lower layer). The porosity inside each layer is uncertain and is modeled as in Eq. (24):

$$\phi(\mathbf{x}, \xi) = 0.35 \cdot (1 + 0.15(\xi_2 \cos(\pi x/2) + \xi_2 \sin(2\pi y) + \xi_1 \cos(2\pi x))) \cdot C_0(\xi_1), \quad (24)$$

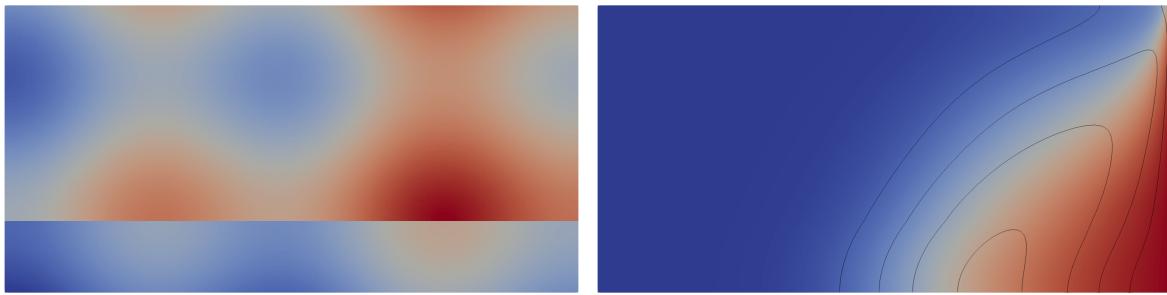
$$\text{where } C_0(\xi_1) = \begin{cases} 1 + 0.2\xi_1 & \text{if } y < -0.75 \\ 1 - 0.2\xi_1 & \text{if } y \geq -0.75, \end{cases} \quad (25)$$

Additionally, the recharge flux is also uncertain and is equal to

$$\hat{q}_{in} = -6.6 \cdot 10^{-2} (1 + 0.5 \cdot \xi_3), \quad (26)$$

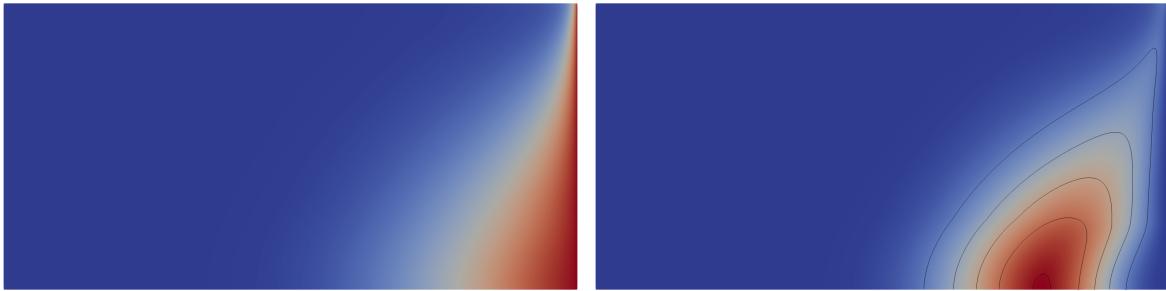
where  $\xi_1$ ,  $\xi_2$ , and  $\xi_3$  are sampled independently and uniformly in  $[-1, 1]$ . Figure 2 depicts a random realization of the porosity random field  $\phi(\xi)$  (left) and the corresponding solution  $c(t, \mathbf{x}, \xi) = c(t, \phi(\xi))$  at  $t = T$  (right). Additionally, four isolines  $\{\mathbf{x} : |c(t, \phi(\xi)) - \bar{c}(t)| = 0.1 \cdot i\}$ ,  $i = 1, 2, 3, 4$ , are presented on the right. The isolines demonstrate the absolute value of the difference between the

computed realization  $c(t, \phi(\xi))$  and the expected value  $\bar{c}(t)$ . These computations were performed for  $\xi = \xi^* = (-0.5898, -0.7257, -0.9616)$  and  $t = T = 6016$  s.



**Figure 2.** (left) Realisation of porosity  $\phi(\xi^*) \in [0.248, 0.499]$ , (right) Corresponding mass fraction  $c(T, \mathbf{x}, \phi(\xi^*)) \in [0, 1]$  with isolines  $\{\mathbf{x} : |c(T, \phi(\xi^*)) - \bar{c}(T)| = 0.1 \cdot i\}, i = 1, 2, 3, 4$ .

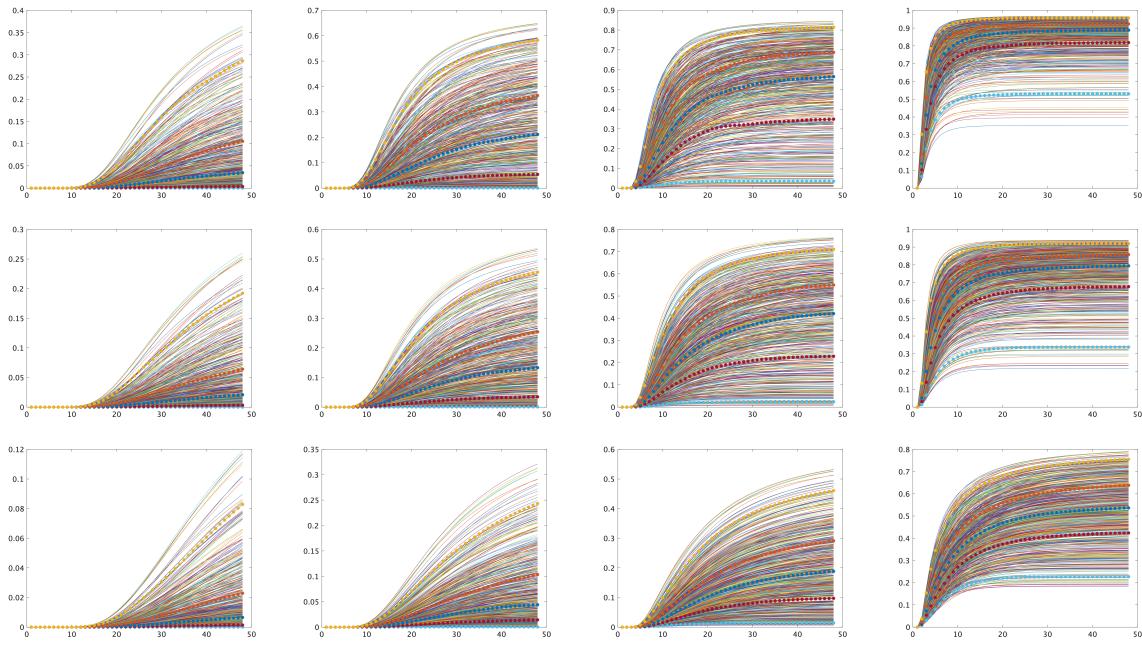
The mean and variance of the mass fraction are provided in Figure 3 on the left and right, respectively. The expectation takes values from  $[0, 1]$ , and the variance range is  $[0, 0.05]$ . The areas with high variance (dark red) indicate regions with high variability/uncertainty. Such regions may need additional attention from specialists (e.g., placement of additional sensors). Additionally, the right image displays five contour lines  $\{\mathbf{x} : \text{Var}[c](t, \mathbf{x}) = 0.01 \cdot i\}, i = 1..5, t = T = 6016$ .



**Figure 3.** (left) Mean value  $\bar{c} \in [0, 1]$  and (right) variance  $\text{Var}[c] \in [0.01, 0.05]$  of the mass fraction, with contour lines  $\{\mathbf{x} : \text{Var}[c] = 0.01 \cdot i\}, i = 1..5, t = T = 6016$ .

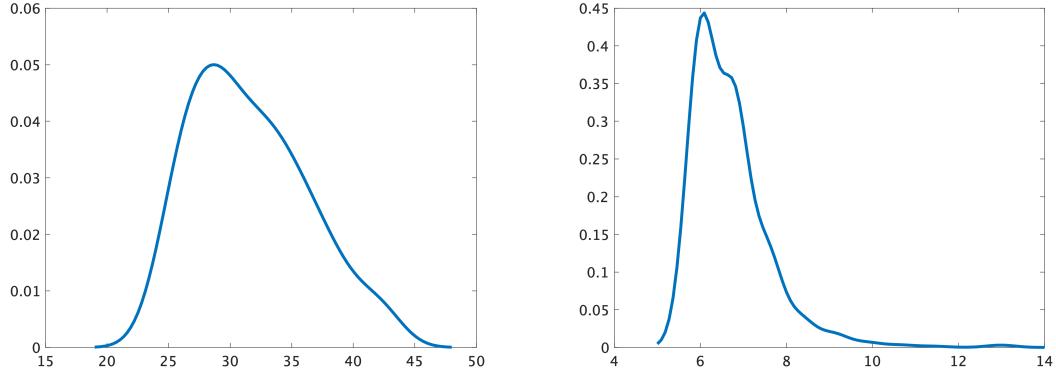
We observed that the variability (uncertainty) of the mass fraction might vary from one grid point to another. At some points (dark blue regions), the solution does not change. At other points (white-yellow regions), the variability is very low or high (dark red regions). In regions with high uncertainty, refining the mesh and applying the MLMC method make sense.

Before we run the MLMC method, we first examine the solution  $c(t, \mathbf{x})$  at 12 preselected points (see Eq. (9)). Figure 4 includes 12 subfigures. Each subfigure presents 600 QMC realizations of  $c(t, \mathbf{x})$  and five quantiles depicted by dotted lines. The dotted line at the bottom indicates the quantile 0.025. The following dotted line is the quantile 0.25, and the dotted line on the top indicates the quantile 0.975. All five quantiles from the bottom to the top are 0.025, 0.25, 0.50, 0.75, and 0.975, respectively. We observe that  $c$  at the final point  $t = T$  varies considerably.



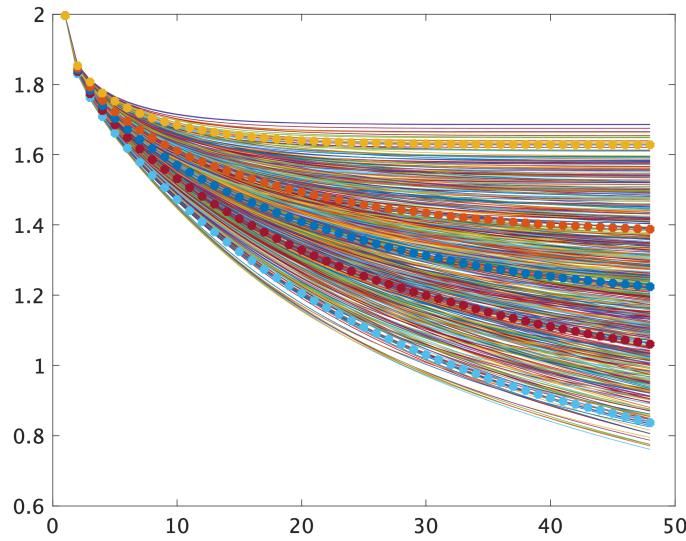
**Figure 4.** Six hundred QMC realisations of  $c(t, x)$  at 12  $x$ -points listed in Eq. (9). First row points:  $\{(1.10, -0.95), (1.35, -0.95), (1.60, -0.95), (1.85, -0.95)\}$ , second row points:  $\{(1.10, -0.75), (1.35, -0.75), (1.60, -0.75), (1.85, -0.75)\}$ , and third row points:  $\{(1.10, -0.50), (1.35, -0.50), (1.60, -0.50), (1.85, -0.50)\}$ . Dotted lines from the bottom to the top indicate the quantiles 0.025, 0.25, 0.50, 0.75, and 0.975, respectively.

**Example.** In Figure 5, we demonstrate the probability density function (pdf) of  $t^*(\omega) = \min_t \{t : Q_{FW}(t, \omega) < 1.2\}$  (left), and the pdf of  $t^*(\omega) = \min_t \{t : Q_{FW}(t, \omega) < 1.7\}$  (right). On average, after approximately 29 time steps (on the left) and six time steps (on the right), the volume of the fresh water becomes less than 1.2 and 1.7, respectively. The initial volume of the fresh water was 2.0.



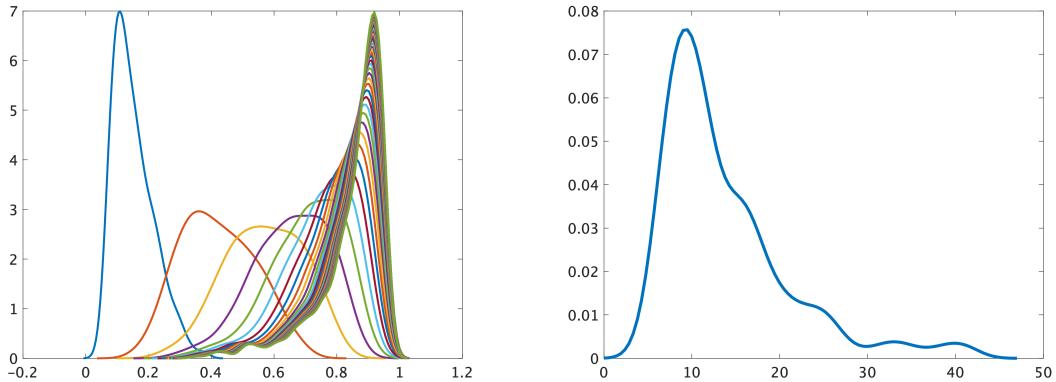
**Figure 5.** The pdf of the earliest time point when the freshwater integral  $Q_{FW}$  becomes smaller than 1.2 (left) and 1.7 (right). The  $x$ -axis represents time points.

All 600 realiations of  $Q_{FW}(t)$  are depicted in Figure 6. The time is along the  $x$ -axis,  $t \in [\tau, 48\tau]$ . Additionally, five quantiles are represented by dotted curves from the bottom to the top and are 0.025, 0.25, 0.50, 0.75, and 0.975, respectively.



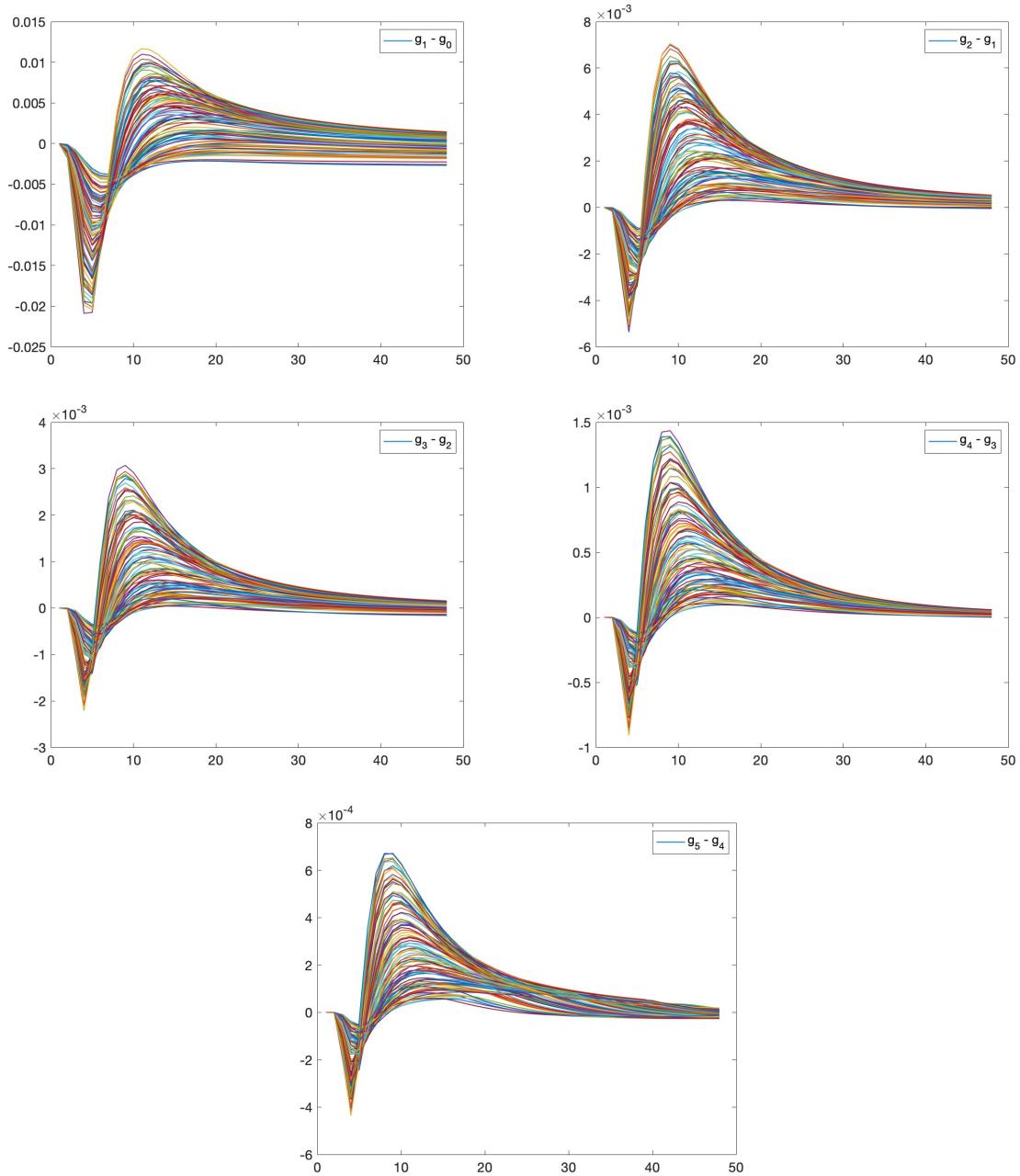
**Figure 6.** Six hundred realizations of  $Q_{FW}(t)$ . The  $x$ -axis represents time  $t = 1\tau, \dots, 48\tau$ ; dotted curves denote five quantiles: 0.025, 0.25, 0.50, 0.75, and 0.975 from the bottom to the top.

**Example.** Figure 7 (left) displays the evolution of the pdf of  $c(t, \mathbf{x}, \omega)$  at a fixed point  $\mathbf{x} = (1.85, -0.95)$  in time  $t = \{3\tau, \dots, 48\tau\}$ . From left to right, the farthest left (blue) pdf corresponds to  $t = 3\tau$ , the second curve from the left (red) corresponds to  $t = 4\tau$ , and so on. In the beginning,  $t = 3\tau$ , and the mass fraction  $c$  is low, about 0.15 on average. Then, with time,  $c$  increases and, at  $t = T = 48\tau$ , is approximately equal to 1. **Example.** The next QoI is the earliest time moment when  $c(t, \mathbf{x})$ , at fixed  $\mathbf{x} = (1.85, -0.95)$ , becomes smaller than the threshold value 0.9 (maximum is 1.0). Figure 7 (right) presents its pdf. On average, after  $t \approx 10$  time steps, the mass fraction becomes smaller than 0.9, but 40 time steps are needed in some scenarios.



**Figure 7.** (left) Evolution of the pdf of  $c(t, \mathbf{x})$  for  $t = \{3\tau, \dots, 48\tau\}$ . (right) The pdf of the earliest time point when  $c(t, \mathbf{x}) < 0.9$  ( $\mathbf{x} = (1.85, -0.95)$  is fixed).

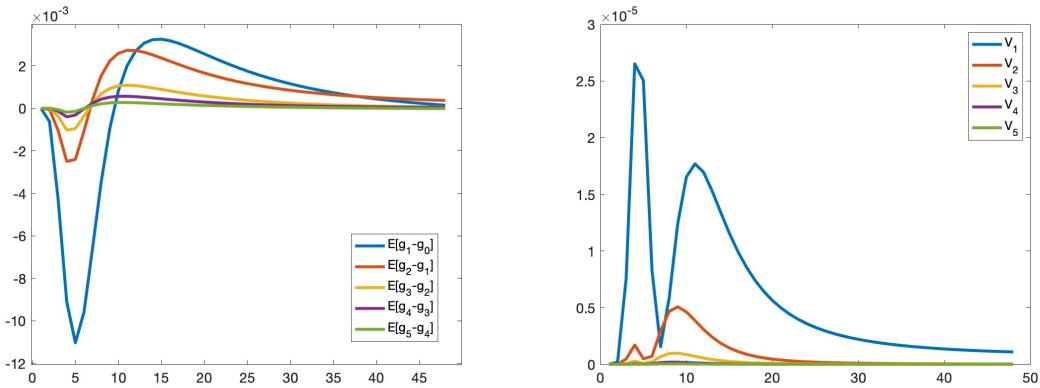
Next, we research how  $g_\ell - g_{\ell-1}$  depends on the time and level. All graphics in Figure 8 display 100 realizations of the differences between solutions computed on two neighbor meshes for every time point  $t_i, i = 1 \dots 48$  (along the  $x$ -axis). The top left graphic indicates the differences between the mass fractions computed on Levels 1 and 0. The other graphics reveal the same, but for Levels 2 and 1, 3 and 2, 4 and 3, and 5 and 4, respectively. The largest value decreases from  $2.5 \cdot 10^{-2}$  (top left) to  $5 \cdot 10^{-4}$ . Considerable variability is observed for  $t \in [3, 7]$  and  $t \in [8, 25]$ . Starting with  $t \approx 30$ , the variability between solutions decreases and stabilizes. From these five graphics, we can estimate that the maximal amplitude decreases by a factor  $\approx 2$ , at 0.015, 0.008, 0.004, 0.0015, and 0.0008. However, it is challenging to make a similar statement about each time point  $t$ . This observation makes it difficult to estimate the weak and strong convergence rates and the optimal number of samples correctly on each mesh level. They are different for each time  $t$  (and for each  $x$ ). For some time points, the solution is smooth and requires only a few levels and a few samples on each level. For other points with substantial dynamics, the numbers of levels and samples are higher.



**Figure 8.** Differences between mass fractions  $c$  computed at the point  $(1.60, -0.95)$  on levels a) 1 and 0, b) 2 and 1 (first row), c) 3 and 2, d) 4 and 3 (second row), and e) 5 and 4 (third row) for 100 realizations ( $x$ -axis represents time).

Because  $g_\ell - g_{\ell-1}$  is random, we visualize its mean and variance. Figure 9 demonstrates the mean (left) and variance (right) of the differences in concentrations  $g_\ell - g_{\ell-1}$ ,  $\ell = 1, \dots, 5$ . On the left, the amplitude decreases when  $\ell$  increases. A slight exception is the blue line for  $t \approx 9, 10, 11$  (right). A possible explanation is that the solutions  $g_0$  or  $g_1$  are insufficiently accurate. The right image presents how the amplitude of the variances decays. This decay is necessary for the successful work of the MLMC method. We also observe a potential issue; the weak and strong convergence rates vary for various time points  $t$ . Thus, determining the optimal number of samples  $m_\ell$  for each level is not possible (only suboptimal).

At the beginning  $t = 0$ , the variability is zero and starts to increase. We observe changes during a specific time interval, and then the process starts to stabilize after  $\approx 45$  time steps. The variability is either unchanging from level to level or decreases.



**Figure 9.** (left) Mean and (right) variance of the differences  $g_\ell - g_{\ell-1}$  vs. time, computed on various levels at the point  $(1.60, -0.95)$ .

Table 2 contains average computing times, which are necessary to estimate the number of samples  $m_\ell$  at each level  $\ell$ . The fourth column contains the average computing time, and the fifth and sixth columns contain the shortest and longest computing times. The computing time for each simulation varies depending on the number of iterations, which depends on the porosity and permeability. We observed that, after  $\approx 6016$  s, the solution is almost unchanging; thus, we restrict this to only  $t \in [0, T]$ , where  $T = 6016$ . For example, if the number of time steps is  $r_\ell = 188$  (Level 0 in Table 2), then the time step  $\tau = \frac{T}{r_\ell} = \frac{6016}{188} = 32$  s.

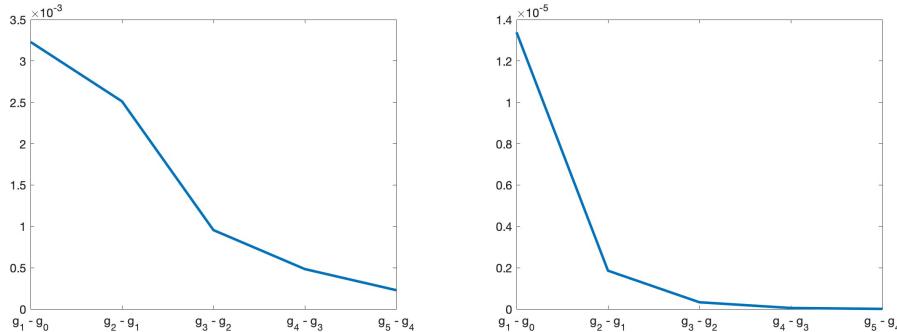
The time step  $\tau$  is adaptive and changing from  $\tau = \frac{6016}{128} = 32$  s (very coarse mesh) to  $\tau = \frac{6016}{6016} = 1$  s (finest mesh). Starting with level  $\ell = 2$ , the average time increases by a factor of eight. These numerical tests confirm the theory in Eq. (12), stating that the numerical solver is linear w.r.t.  $n_\ell$  and  $r_\ell$ .

**Table 2.** Number of degrees of freedom  $n_\ell$ , number of time steps  $r_\ell$ , step size in time  $\tau_\ell$ , average, minimal, and maximal computing times on each level  $\ell$ .

Level $\ell$	$n_\ell$	$r_\ell$	$\tau_\ell = 6016/r_\ell$	Computing times ( $s_\ell$ )		
				average	min.	max.
0	1122	188	32	1.15	0.88	1.33
1	4290	376	16	4.1	3.4	4.87
2	16770	752	8	19.6	17.6	22
3	66306	1504	4	136.0	128	144
4	263682	3008	2	1004.0	891	1032
5	1051650	6016	1	8138.0	6430	8480

With estimates for each level, we can estimate the rates of  $\alpha$  and  $\beta$  (Eqs. (21a)-(21b)) in weak and strong convergences.

The slope in Figure 10 can be used to estimate the rates of the weak (left) and strong (right) convergences. The differences are indicated on the horizontal axis.



**Figure 10.** Weak (left) and strong (right) convergences computed for Levels 1 and 0, 2 and 1, 3 and 2, 4 and 3, and 5 and 4 (horizontal axis) at the fixed point  $(t, x, y) = (14, 1.60, -0.95)$ .

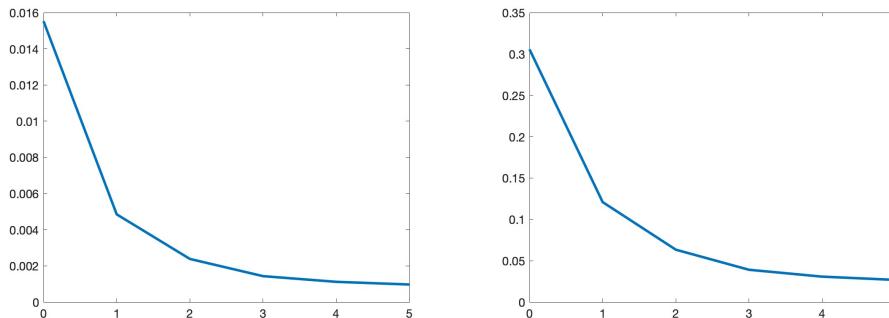
We use computed variances  $V_\ell$  and computing times (work)  $s_\ell$  from Table 2 to estimate the optimal number of samples  $m_\ell$  and compute the telescopic sum from Eq. (15) to approximate the expectation.

Table 3 lists  $m_\ell$  for a given total variance  $\epsilon^2$ :

**Table 3.** Number of samples  $m_\ell$  computed using Eq. (18) as a function of the total variance  $\epsilon^2$ .

level, $\ell$	0	1	2	3	4	5
$s_\ell$	1.156	4.113	20.382	139.0	993.0	8053.0
$V_\ell$	1.4e-5	0.2e-5	0.5e-6	0.1e-6	0.5e-7	1e-7
$m_\ell(\epsilon^2 = 5e-6)$	35	7	2	1	1	1
$m_\ell(\epsilon^2 = 1e-6)$	172	35	8	2	1	1
$m_\ell(\epsilon^2 = 5e-7)$	343	69	16	3	1	1
$m_\ell(\epsilon^2 = 1e-7)$	1714	344	78	14	4	2

After the telescopic sum is computed, we can compare the results with the QMC results. Figure 11 depicts the decay of the absolute (left) and relative (right) errors vs. levels along the  $x$ -axes. The 'true' solution was computed using the QMC method on the finest mesh level  $L = 5$ .



**Figure 11.** Decay of the absolute (left) and relative (right) errors between the mean values computed on a fine mesh via QMC and on a hierarchy of meshes via MLMC at the fixed point  $(t, x, y) = (12, 1.60, -0.95)$ .  $x$ -axis contains the mesh levels.

## 5. Conclusion

We investigated the applicability and efficiency of the MLMC approach for the Henry-like problem with uncertain porosity, permeability, and recharge. These uncertain parameters were modeled by random fields with three independent random variables. The numerical solution for each random realization was obtained using the well-known ug4 parallel multigrid solver. The number of required random samples on each level was estimated by computing the decay of the variances and computational costs for each level. These estimates depend on the minimization function in Eq. (17).

We also computed the expected value and variance of the mass fraction in the whole domain, the evolution of the pdfs, the solutions at a few preselected points  $(t, \mathbf{x})$ , and the time evolution of the freshwater integral value. We have found that some QoIs require only 2-3 of the coarsest mesh levels, and samples from finer meshes would not significantly improve the result. Note that a different type of porosity in Eq. (24) may lead to a different conclusion.

The results show that the MLMC method is faster than the QMC method at the finest mesh. Thus, sampling at different mesh levels makes sense and helps to reduce the overall computational cost.

**Limitations.** 1. It may happen that the QoIs computed on different grid levels are the same (for the given random input parameters). In this case the standard (Q)MC on a coarse mesh will be sufficient. 2. The time dependence is challenging. The optimal number of samples depends on the point  $(t, \mathbf{x})$  and may be small for some points and large for others. 3. Twenty-four hours may not be sufficient to compute the solution at the sixth mesh level.

**Future work.** Our model of porosity in Eq. (24) is quite simple. It would be beneficial to consider a more complicated/multiscale/realistic model with more random variables. A more advanced version of MLMC may give better estimates of the number of levels  $L$  and the number of samples on each level  $m_\ell$ . Another hot topic is data assimilation and the identification of unknown parameters [73–76]. Known experimental data and measurements of porosity, permeability, velocity or mass fraction could be used to minimise uncertainties.

**Acknowledgments:** We thank the KAUST HPC support team for assistance with Shaheen II. This work was supported by the Alexander von Humboldt foundation.

## References

1. Abarca, E.; Carrera, J.; Sánchez-Vila, X.; Dentz, M. Anisotropic dispersive Henry problem. *Advances in Water Resources* **2007**, *30*, 913–926. doi:<https://doi.org/10.1016/j.advwatres.2006.08.005>.
2. Schneider, A.; Zhao, H.; Wolf, J.; Logashenko, D.; Reiter, S.; Howahr, M.; Eley, M.; Gelleszun, M.; Wiederhold, H. Modeling saltwater intrusion scenarios for a coastal aquifer at the German North Sea. *E3S Web Conf.* **2018**, *54*, 00031. doi:[10.1051/e3sconf/20185400031](https://doi.org/10.1051/e3sconf/20185400031).
3. Voss, C.; Souza, W. Variable density flow and solute transport simulation of regional aquifers containing a narrow freshwater-saltwater transition zone. *Water Resources Research* **1987**, *23*, 1851–1866. doi:[10.1029/WR023i010p01851](https://doi.org/10.1029/WR023i010p01851).
4. Henry, H.R. Effects of dispersion on salt encroachment in coastal aquifers, in 'Seawater in Coastal Aquifers'. *US Geological Survey, Water Supply Paper* **1964**, *1613*, C70–C80.
5. Simpson, M.J.; Clement, T.P. Improving the worthiness of the Henry problem as a benchmark for density-dependent groundwater flow models. *Water Resources Research* **2004**, *40*, W01504. doi:[10.1029/2003WR002199](https://doi.org/10.1029/2003WR002199).
6. Simpson, M.J.; Clement, T. Theoretical Analysis of the worthiness of Henry and Elder problems as benchmarks of density-dependent groundwater flow models. *Adv. Water. Resour.* **2003**, *26*, 17–31.
7. Dhal, L.; Swain, S., Understanding and modeling the process of seawater intrusion: a review; 2022; pp. 269–290. doi:[10.1016/B978-0-12-823830-1.00009-2](https://doi.org/10.1016/B978-0-12-823830-1.00009-2).
8. Riva, M.; Guadagnini, A.; Dell’Oca, A. Probabilistic assessment of seawater intrusion under multiple sources of uncertainty. *Advances in Water Resources* **2015**, *75*, 93–104. doi:[10.1016/j.advwatres.2014.11.002](https://doi.org/10.1016/j.advwatres.2014.11.002).
9. Reiter, S.; Logashenko, D.; Vogel, A.; Wittum, G. Mesh generation for thin layered domains and its application to parallel multigrid simulation of groundwater flow. submitted to *Comput. Visual Sci.*

10. Schneider, A.; Kröhn, K.P.; Püschel, A. Developing a modelling tool for density-driven flow in complex hydrogeological structures. *Comput. Visual Sci.* **2012**, *15*, 163–168. doi:10.1007/s00791-013-0207-2.
11. Cremer, C.; ; Graf, T. Generation of dense plume fingers in saturated–unsaturated homogeneous porous media. *Journal of Contaminant Hydrology* **2015**, *173*, 69 – 82. doi:<https://doi.org/10.1016/j.jconhyd.2014.11.008>.
12. Carrera, J. An overview of uncertainties in modelling groundwater solute transport. *Journal of Contaminant Hydrology* **1993**, *13*, 23 – 48. Chemistry and Migration of Actinides and Fission Products, doi:[https://doi.org/10.1016/0169-7722\(93\)90049-X](https://doi.org/10.1016/0169-7722(93)90049-X).
13. Vereecken, H.; Schnepp, A.; Hopmans, J.; Javaux, M.; Or, D.; Roose, T.; Vanderborght, J.; Young, M.; Amelung, W.; Aitkenhead, M.; Allison, S.; Assouline, S.; Baveye, P.; Berli, M.; Brüggemann, N.; Finke, P.; Flury, M.; Gaiser, T.; Govers, G.; Ghezzehei, T.; Hallett, P.; Hendricks Franssen, H.; Heppell, J.; Horn, R.; Huisman, J.; Jacques, D.; Jonard, F.; Kollet, S.; Lafolie, F.; Lamorski, K.; Leitner, D.; McBratney, A.; Minasny, B.; Montzka, C.; Nowak, W.; Pachepsky, Y.; Padarian, J.; Romano, N.; Roth, K.; Rothfuss, Y.; Rowe, E.; Schwen, A.; Šimůnek, J.; Tiktak, A.; Van Dam, J.; van der Zee, S.; Vogel, H.; Vrugt, J.; Wöhling, T.; Young, I. Modeling Soil Processes: Review, Key Challenges, and New Perspectives. *Vadose Zone Journal* **2016**, *15*, vzej2015.09.0131, [\[https://gsweb/content/public/journal/vzj/15/5/10.2136/vzj2015.09.0131/3/vzj2015.09.0131.pdf\]](https://gsweb/content/public/journal/vzj/15/5/10.2136/vzj2015.09.0131/3/vzj2015.09.0131.pdf). doi:10.2136/vzj2015.09.0131.
14. Bode, F.; Ferré, T.; Zigelli, N.; Emmert, M.; Nowak, W. Reconnecting Stochastic Methods With Hydrogeological Applications: A Utilitarian Uncertainty Analysis and Risk Assessment Approach for the Design of Optimal Monitoring Networks. *Water Resources Research* **2018**, *54*, 2270–2287, [\[https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1002/2017WR020919\]](https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1002/2017WR020919). doi:10.1002/2017WR020919.
15. Rubin, Y. *Applied stochastic hydrogeology*; Oxford University Press, 2003.
16. Tartakovsky, D. Assessment and management of risk in subsurface hydrology: A review and perspective. *Advances in Water Resources* **2013**, *51*, 247 – 260. 35th Year Anniversary Issue, doi:<https://doi.org/10.1016/j.advwatres.2012.04.007>.
17. Post, V.; Houben, G. Density-driven vertical transport of saltwater through the freshwater lens on the island of Baltrum (Germany) following the 1962 storm flood. *Journal of Hydrology* **2017**, *551*, 689 – 702. Investigation of Coastal Aquifers, doi:<https://doi.org/10.1016/j.jhydrol.2017.02.007>.
18. Laatooe, T.; Werner, A.; Simmons, C., Seawater Intrusion Under Current Sea-Level Rise: Processes Accompanying Coastline Transgression. In *Groundwater in the Coastal Zones of Asia-Pacific*; Wetzelhuett, C., Ed.; Springer Netherlands: Dordrecht, 2013; pp. 295–313. doi:10.1007/978-94-007-5648-9\_14.
19. Espig, M.; Hackbusch, W.; Litvinenko, A.; Matthies, H.; Wähnert, P. Efficient low-rank approximation of the stochastic Galerkin matrix in tensor formats. *Computers and Mathematics with Applications* **2014**, *67*, 818 – 829. High-order Finite Element Approximation for Partial Differential Equations, doi:<http://dx.doi.org/10.1016/j.camwa.2012.10.008>.
20. Babuška, I.; Tempone, R.; Zouraris, G. Galerkin finite element approximations of stochastic elliptic partial differential equations. *SIAM Journal on Numerical Analysis* **2004**, *42*, 800–825.
21. Giraldi, L.; Litvinenko, A.; Liu, D.; Matthies, H.G.; Nouy, A. To Be or Not to Be Intrusive? The Solution of Parametric and Stochastic Equations—the “Plain Vanilla” Galerkin Case. *SIAM Journal on Scientific Computing* **2014**, *36*, A2720–A2744, [\[https://doi.org/10.1137/130942802\]](https://doi.org/10.1137/130942802). doi:10.1137/130942802.
22. Espig, M.; Hackbusch, W.; Litvinenko, A.; Matthies, H.; Wähnert, P. Efficient low-rank approximation of the stochastic Galerkin matrix in tensor formats. *Computers and Mathematics with Applications* **2014**, *67*, 818–829. doi:10.1016/j.camwa.2012.10.008.
23. Liu, D.; Görtz, S. Efficient Quantification of Aerodynamic Uncertainty due to Random Geometry Perturbations. In *New Results in Numerical and Experimental Fluid Mechanics IX*; Dillmann, A.; others., Eds.; Springer International Publishing, 2014; pp. 65–73.
24. Bompard, M.; Peter, J.; Désidéri, J.A. Surrogate models based on function and derivative values for aerodynamic global optimization. Fifth European Conference on Computational Fluid Dynamics, ECCOMAS CFD 2010;, 2010.
25. Loeven, G.J.A.; Witteveen, J.A.S.; Bijl, H. A probabilistic radial basis function approach for uncertainty quantification. Proceedings of the NATO RTO-MP-AVT-147 Computational Uncertainty in Military Vehicle design symposium, 2007.

26. Giunta, A.A.; Eldred, M.S.; Castro, J.P. Uncertainty quantification using response surface approximation. 9th ASCE Specialty Conference on Probabilistic Mechanics and Structural Reliability; 2004.

27. Chkifa, A.; Cohen, A.; Schwab, C. Breaking the curse of dimensionality in sparse polynomial approximation of parametric PDEs. *Journal de Mathematiques Pures et Appliques* **2015**, *103*, 400 – 428. doi:10.1016/j.matpur.2014.04.009.

28. Blatman, G.; Sudret, B. An adaptive algorithm to build up sparse polynomial chaos expansions for stochastic finite element analysis. *Probabilistic Engineering Mechanics* **2010**, *25*, 183 – 197. doi:<http://dx.doi.org/10.1016/j.probengmech.2009.10.003>.

29. Dolgov, S.; Khoromskij, B.; Litvinenko, A.; Matthies, H. Polynomial Chaos Expansion of Random Coefficients and the Solution of Stochastic Partial Differential Equations in the Tensor Train Format. *SIAM/ASA Journal on Uncertainty Quantification* **2015**, *3*, 1109–1135, [<https://doi.org/10.1137/140972536>]. doi:10.1137/140972536.

30. Najm, H. Uncertainty Quantification and Polynomial Chaos Techniques in Computational Fluid Dynamics. *Annual Review of Fluid Mechanics* **2009**, *41*, 35–52, [<https://doi.org/10.1146/annurev.fluid.010908.165248>]. doi:10.1146/annurev.fluid.010908.165248.

31. Conrad, P.; Marzouk, Y. Adaptive Smolyak Pseudospectral Approximations. *SIAM Journal on Scientific Computing* **2013**, *35*, A2643–A2670, [<https://doi.org/10.1137/120890715>]. doi:10.1137/120890715.

32. Xiu, D. Fast Numerical Methods for Stochastic Computations: A Review. *Commun. Comput. Phys.* **2009**, *5*, No. 2-4, 242–272.

33. Smolyak, S.A. Quadrature and interpolation formulas for tensor products of certain classes of functions. *Sov. Math. Dokl.* **1963**, *4*, 240–243.

34. Bungartz, H.J.; Griebel, M. Sparse grids. *Acta Numer.* **2004**, *13*, 147–269.

35. Griebel, M. Sparse grids and related approximation schemes for higher dimensional problems. In *Foundations of computational mathematics, Santander 2005*; Cambridge Univ. Press: Cambridge, 2006; Vol. 331, *London Math. Soc. Lecture Note Ser.*, pp. 106–161.

36. Klimke, A. Sparse Grid Interpolation Toolbox, [www.ians.uni-stuttgart.de/spinterp/](http://www.ians.uni-stuttgart.de/spinterp/) **2008**.

37. Novak, E.; Ritter, K. The curse of dimension and a universal method for numerical integration. In *Multivariate approximation and splines (Mannheim, 1996)*; Birkhäuser: Basel, 1997; Vol. 125, *Internat. Ser. Numer. Math.*, pp. 177–187.

38. Gerstner, T.; Griebel, M. Numerical integration using sparse grids. *Numer. Algorithms* **1998**, *18*, 209–232.

39. Novak, E.; Ritter, K. Simple cubature formulas with high polynomial exactness. *Constr. Approx.* **1999**, *15*, 499–522.

40. Petras, K. Smolpack—a software for Smolyak quadrature with delayed Clenshaw-Curtis basis-sequence. <http://www-public.tu-bs.de:8080/petras/software.html>.

41. Eigel, M.; Gittelson, C.J.; Schwab, C.; Zander, E. Adaptive stochastic Galerkin FEM. *Computer Methods in Applied Mechanics and Engineering* **2014**, *270*, 247–269. doi:<https://doi.org/10.1016/j.cma.2013.11.015>.

42. Beck, J.; Liu, Y.; von Schwerin, E.; Tempone, R. Goal-oriented adaptive finite element multilevel Monte Carlo with convergence rates. *Computer Methods in Applied Mechanics and Engineering* **2022**, *402*, 115582. A Special Issue in Honor of the Lifetime Achievements of J. Tinsley Oden, doi:<https://doi.org/10.1016/j.cma.2022.115582>.

43. Matthies, H. Uncertainty Quantification with Stochastic Finite Elements. In *Encyclopedia of Computational Mechanics*; Stein, E.; de Borst, R.; Hughes, T.R.J., Eds.; John Wiley & Sons: Chichester, 2007.

44. Babuška, I.; Nobile, F.; Tempone, R. A stochastic collocation method for elliptic partial differential equations with random input data. *SIAM J. Numer. Anal.* **2007**, *45*, 1005–1034 (electronic).

45. Nobile, F.; Tamellini, L.; Tesei, F.; Tempone, R. An adaptive sparse grid algorithm for elliptic PDEs with log-normal diffusion coefficient. MATHICSE Technical Report 04, 2015.

46. Radović, I.; Sobol, I.; Tichy, R. Quasi-Monte Carlo Methods for Numerical Integration: Comparison of Different Low Discrepancy Sequences. *Monte Carlo Methods and Applications* **1996**, *2*, 1–14.

47. Xiu, D.; Karniadakis, G.E. The Wiener-Askey polynomial chaos for stochastic differential equations. *SIAM J. Sci. Comput.* **2002**, *24*, 619–644.

48. Litvinenko, A.; Logashenko, D.; Tempone, R.; Wittum, G.; Keyes, D. Propagation of Uncertainties in Density-Driven Flow. Sparse Grids and Applications — Munich 2018; Bungartz, H.J.; Griebke, J.; Pflüger, D., Eds.; Springer International Publishing: Cham, 2021; pp. 101–126. doi:10.1007/978-3-030-81362-8\_5.

49. Litvinenko, A.; Logashenko, D.; Tempone, R.; Wittum, G.; Keyes, D. Solution of the 3D density-driven groundwater flow problem with uncertain porosity and permeability. *GEM - International Journal on Geomathematics* **2020**, *11*, 10. doi:10.1007/s13137-020-0147-1.

50. Oladyshkin, S.; Nowak, W. Data-driven uncertainty quantification using the arbitrary polynomial chaos expansion. *Reliability Engineering & System Safety* **2012**, *106*, 179–190. doi:<https://doi.org/10.1016/j.ress.2012.05.002>.

51. Stoeckl, L.; Walther, M.; Morgan, L.K. Physical and Numerical Modelling of Post-Pumping Seawater Intrusion. *Geofluids* **2019**, *2019*. doi:10.1155/2019/7191370.

52. Panda, M.; Lake, W. Estimation of single-phase permeability from parameters of particle-size distribution. *AAPG Bull.* **1994**, *78*, 1028–1039.

53. Pape, H.; Clauser, C.; Iffland, J. Permeability prediction based on fractal pore-space geometry. *Geophysics* **1999**, *64*, 1447–1460. doi:10.1190/1.1444649.

54. Costa, A. Permeability-porosity relationship: A reexamination of the Kozeny-Carman equation based on a fractal pore-space geometry assumption. *Geophysical Research Letters* **2006**, *33*. doi:10.1029/2005GL025134.

55. Frolkovič, P.; De Schepper, H. Numerical modelling of convection dominated transport coupled with density driven flow in porous media. *Advances in Water Resources* **2001**, *24*, 63–72. doi:10.1016/S0309-1708(00)00025-7.

56. Frolkovič, P. Consistent velocity approximation for density driven flow and transport. *Advanced Computational Methods in Engineering, Part 2: Contributed papers*; Van Keer, R.; et al., Eds.; Shaker Publishing: Maastricht, 1998; pp. 603–611.

57. Frolkovič, P.; Knabner, P. Consistent Velocity Approximations in Finite Element or Volume Discretizations of Density Driven Flow. *Computational Methods in Water Resources XI*; Aldama, A.A.; et al., Eds.; Computational Mechanics Publication: Southhampton, 1996; pp. 93–100.

58. Barrett, R.; Berry, M.; Chan, T.F.; Demmel, J.; Donato, J.; Dongarra, J.; Eijkhout, V.; Pozo, R.; Romine, C.; van der Vorst, H. *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*; Society for Industrial and Applied Mathematics, 1994; [<https://epubs.siam.org/doi/pdf/10.1137/1.9781611971538>]. doi:10.1137/1.9781611971538.

59. Hackbusch, W. *Multi-Grid Methods and Applications*; Springer, Berlin, 1985.

60. Hackbusch, W. *Iterative Solution of Large Sparse Systems of Equations*; Springer: New-York, 1994.

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

62. Collier, N.; Haji-Ali, A.L.; Nobile, F.; von Schwerin, E.; Tempone, R. A continuation multilevel Monte Carlo algorithm. *BIT Numerical Mathematics* **2015**, *55*, 399–432.

63. Giles, M.B. Multilevel Monte Carlo path simulation. *Operations Research* **2008**, *56*, 607–617.

64. Giles, M.B. Multilevel Monte Carlo methods. *Acta Numerica* **2015**, *24*, 259–328.

65. Haji-Ali, A.L.; Nobile, F.; von Schwerin, E.; Tempone, R. Optimization of mesh hierarchies in multilevel Monte Carlo samplers. *Stoch. Partial Differ. Equ. Anal. Comput.* **2016**, *4*, 76–112. doi:10.1007/s40072-015-0049-7.

66. Teckentrup, A.; Scheichl, R.; Giles, M.; Ullmann, E. Further analysis of multilevel Monte Carlo methods for elliptic PDEs with random coefficients. *Numerische Mathematik* **2013**, *125*, 569–600.

67. Litvinenko, A.; Yucel, A.C.; Bagci, H.; Oppelstrup, J.; Michielssen, E.; Tempone, R. Computation of Electromagnetic Fields Scattered From Objects With Uncertain Shapes Using Multilevel Monte Carlo Method. *IEEE Journal on Multiscale and Multiphysics Computational Techniques* **2019**, *4*, 37–50. doi:10.1109/JMMCT.2019.2897490.

68. Hoel, H.; von Schwerin, E.; Szepessy, A.; Tempone, R. Implementation and analysis of an adaptive multilevel Monte Carlo algorithm. *Monte Carlo Methods and Applications* **2014**, *20*, 1–41. doi:doi:10.1515/mcma-2013-0014.

69. Hoel, H.; Von Schwerin, E.; Szepessy, A.; Tempone, R. Adaptive multilevel Monte Carlo simulation. In *Numerical Analysis of Multiscale Computations*; Springer, 2012; pp. 217–234.

70. Charrier, J.; Scheichl, R.; Teckentrup, A.L. Finite Element Error Analysis of Elliptic PDEs with Random Coefficients and Its Application to Multilevel Monte Carlo Methods **2013**, *51*, 322–352.

71. Reiter, S.; Vogel, A.; Heppner, I.; Rupp, M.; Wittum, G. A massively parallel geometric multigrid solver on hierarchically distributed grids. *Computing and Visualization in Science* **2013**, *16*, 151–164. doi:10.1007/s00791-014-0231-x.
72. Vogel, A.; Reiter, S.; Rupp, M.; Nägel, A.; Wittum, G. UG 4: A novel flexible software system for simulating PDE based models on high performance computers. *Computing and Visualization in Science* **2013**, *16*, 165–179. doi:10.1007/s00791-014-0232-9.
73. Litvinenko, A.; Kriemann, R.; Genton, M.G.; Sun, Y.; Keyes, D.E. HLIBCov: Parallel hierarchical matrix approximation of large covariance matrices and likelihoods with applications in parameter identification. *MethodsX* **2020**, *7*, 100600. doi:<https://doi.org/10.1016/j.mex.2019.07.001>.
74. Litvinenko, A.; Sun, Y.; Genton, M.G.; Keyes, D.E. Likelihood approximation with hierarchical matrices for large spatial datasets. *Computational Statistics & Data Analysis* **2019**, *137*, 115–132. doi:<https://doi.org/10.1016/j.csda.2019.02.002>.
75. Matthies, H.G.; Zander, E.; Rosić, B.V.; Litvinenko, A. Parameter estimation via conditional expectation: a Bayesian inversion. *Advanced Modeling and Simulation in Engineering Sciences* **2016**, *3*, 24. doi:10.1186/s40323-016-0075-7.
76. Rosić, B.; Kučerová, A.; Sýkora, J.; Pajonk, O.; Litvinenko, A.; Matthies, H. Parameter identification in a probabilistic setting. *Engineering Structures* **2013**, *50*, 179 – 196. Engineering Structures: Modelling and Computations (special issue IASS-IACM 2012), doi:<http://dx.doi.org/10.1016/j.engstruct.2012.12.029>.

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.