

Solution of the 3D density-driven groundwater flow problem with uncertain porosity and permeability

Alexander Litvinenko¹, Dmitry Logashenko², Raul Tempone^{1,2}, Gabriel Wittum^{2,3}, and David Keyes²

Abstract As groundwater is an essential nutrition and irrigation resource, its pollution may lead to catastrophic consequences. Therefore, accurate modeling of the pollution of the soil and groundwater aquifer is highly important. As a model, we consider a density-driven groundwater flow problem with uncertain porosity and permeability. This problem may arise in geothermal reservoir simulation, natural saline-disposal basins, modeling of contaminant plumes, and subsurface flow. This strongly nonlinear time-dependent problem describes the convection of the two-phase flow. This liquid streams under the gravity force, building so-called “fingers”. The accurate numerical solution requires fine spatial resolution with an unstructured mesh and, therefore, high computational resources. Consequently, we run the parallelized simulation toolbox ug4 with the geometric multigrid solver on Shaheen II supercomputer. The parallelization is done in physical and stochastic spaces. Additionally, we demonstrate how the ug4 toolbox can be run in a black-box fashion for testing different scenarios in the density-driven flow. As a benchmark, we solve the Elder-like problem in a 3D domain. For approximations in the stochastic space, we use the generalized polynomial chaos expansion. We compute the mean, variance, and exceedance probabilities of the mass fraction. As a reference solution, we use the solution, obtained from the quasi-Monte Carlo method.

Keywords: uncertainty quantification, ug4, multigrid, density driven flow, reservoir, groundwater, salt formations

1 Motivation

This work is a 3D extension of our previous 2D work [47]. The problem setup is the same; the difference is only in the aquifer. In this work, we consider two 3D aquifers. These new 3D geometries contain a much larger number of degrees of freedom (DoFs), and, therefore, numerical experiments require significantly more computational resources.

Accurate prediction of the contamination of the groundwater is highly essential. Certain pollutants are soluble in water and can leak into groundwater systems, such as seawater into coastal aquifers or wastewater leaks. Indeed, some pollutants can change the density of a fluid and induce density-driven flows within the aquifer. This causes faster propagation of the contamination due to convection. Thus, simulation and analysis of this density-driven flow play an important role in predicting how pollution can migrate through an aquifer [28, 43].

In contrast to the transport of pollution by molecular diffusion, convection due to density-driven flow is an unstable process that can undergo quite complicated patterns of distribution. The Elder problem is a simplified but comprehensive model that describes the intrusion of salt water from a top boundary into an aquifer [23, 24, 25, 79]. The evolution of the concentration profile is typically referred to as *fingering*. Note that due to the nonlinear nature of the model, the distribution of the contamination strongly depends on the model parameters, so that the system may have several stationary solutions [40].

¹RWTH Aachen, Kackertstr. 9C, Aachen, Germany, Phone: +492418099203, e-mail: litvinenko, tempone@uq.rwth-aachen.de

²KAUST, Thuwal/Jeddah, Saudi Arabia, e-mail: dmitry.logashenko, gabriel.wittum, david.keyes@kaust.edu.sa

³G-CSC, Frankfurt University, Kettenhofweg 139, Frankfurt, Germany

Hydrogeological formations typically have a complicated and heterogeneous structure, and geological media may consist of layers of porous media with different porosities and permeability coefficients [68, 73]. Difficulty in specifying hydrogeological parameters of the media, as well as measuring the position and configuration of the layers, gives rise to certain errors. Typically, the averaged values of these quantities are used. However, due to the nonlinearities of the flow model, the averaging of the parameters does not necessarily lead to the correct mathematical solution. Uncertainties arise from various factors, such as inaccurate measurements of the different parameters and the inability to measure parameters at each spatial location at any given time. To model the uncertainties, we can use random variables, random fields, and random processes. However, uncertainties in the input data spread through the model and make the solution uncertain too. Thus, an accurate estimation of the output uncertainties is crucial.

Many techniques are used to investigate the propagation of uncertainties associated with porosity and permeability into the solution, such as classical Monte Carlo sampling. Although Monte Carlo sampling is dimension-independent, it converges slowly and therefore requires a lot of samples, and these simulations may be time-consuming and costly. Alternative, less costly techniques use collocations, sparse grids, and surrogate methods, each with their advantages and disadvantages. But even advanced techniques may require hundreds to thousands of simulations and assume a certain smoothness in the quantity of interest. Our simulations contain up to 4.5MI spatial mesh points and 1000-3000 time-steps and are therefore run on a massive parallel cluster. Here, we develop methods that require much fewer simulations, from a few dozen to a few hundred.

Perturbation methods, which decompose the quantity of interest (QoI) with respect to random parameters in a Taylor series, were considered and compared in [19].

Here, we use the well-established generalized polynomial chaos expansion (gPC) technique [82], where we compute the gPC coefficients by applying a Clenshaw-Curtis quadrature rule [4]. We use gPC to compute QoIs such as the mean, the variance, and the exceedance probabilities of the mass fraction (in this case, a saltwater solution). We validate our obtained results using the quasi-Monte Carlo approach. Both methods require the computation of multiple simulations (scenarios) for variable porosity and permeability coefficients.

To the best of our knowledge, no other reported works have solved the Elder's problem [23, 79] with uncertain porosity and permeability parameters using gPC.

Fingers in a highly unstable free convective flow have been studied in [84]. Overviews of the uncertainties in modeling groundwater solute transport [13] and modeling soil processes [77] have been performed, as well as reconnecting stochastic methods with hydrogeological applications [9], which included recommendations for optimization and risk assessment. Fundamentals of stochastic hydrogeology, an overview of stochastic tools and accounting for uncertainty have been described in [72].

Recent advances in uncertainty quantification, probabilistic risk assessment, and decision-making under uncertainty in hydrogeologic applications have been reviewed in [76], where 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 has been modeled in [66].

Various methods can be used to compute the desired statistics, such as direct integration methods (Monte Carlo, quasi Monte Carlo, collocation) and surrogate-based methods (generalized polynomial chaos approximation, stochastic Galerkin) [3, 27, 56]. Direct integration methods compute statistics by sampling unknown input coefficients (they are undefined and therefore uncertain) and solving the corresponding PDE, while the surrogate-based method computes a low-cost functional (polynomial, exponential, trigonometrical) approximation of QoI. Examples of the surrogate-based methods include radial basis functions [10, 35, 51, 53], sparse polynomials [8, 14], polynomial chaos expansion [16, 59, 81, 83] and low-rank tensor approximation [21, 22, 41, 48, 49]. A nonintrusive stochastic Galerkin was introduced in [34]. The surrogate-based methods have an additional advantage if the gradients are available [52]. Sparse grids methods to integrate high-dimensional integrals are considered in [11, 16, 17, 32, 36, 42, 62, 63, 64, 75]. The sparse grid software package is available in [65].

The quantification of uncertainties in stochastic PDEs can pose a great challenge due to the possibly large number of random variables involved, and the high cost of each deterministic solution.

For problems with a large number of random variables, the methods based on a regular grid (in stochastic space) are less preferable due to the high computational cost. Instead, methods based on a scattered sampling scheme (MC, qMC) give more freedom in choosing the sample number N and protect against sample failures. The MC quadrature and its variance-reduced variants have dimension-independent convergence $\mathcal{O}(N^{-\frac{1}{2}})$, and qMC has the worst case convergence $\mathcal{O}(\log^M(N)N^{-1})$, where M is the dimension of the stochastic space [54]. Collocations on sparse or full grids are affected by the dimension of the integration domain [2, 60, 61]. In this work, we use the Halton rule to generate quasi-Monte Carlo quadrature points [12, 39]. A numerical comparison of other qMC sequences has been described in [67].

The construction of a gPC-based surrogate model [55] is similar to computing the Fourier expansion, where only the Fourier coefficients need to be computed. Some well-known functions, such as multivariate Legendre, Hermite, Chebyshev, and Laguerre functions, are taken as a basis [82]. These functions should possess some useful properties, e.g., orthogonality. Usually, a small number of quadrature points are used to compute these coefficients. Once the surrogate is constructed, its sampling could be performed at a lower cost than a sampling of the original stochastic PDE.

To tackle the high numerical complexity, we implement a two-level parallelization. First, we run all available quadrature or sampling points (also say scenarios) in parallel, with each scenario on a separate computing node. Second, each scenario is computed in parallel on 2 – 32 cores.

This work is structured as follows: In Section 2, we outline the model of the density-driven groundwater flow in porous media and the numerical methods for this type of problem. We describe the stochastic modeling, integration methods, and the generalized polynomial chaos expansion technique in Section 3. We present details of the parallelization of the computations in Section 4. Our multiple numerical results, described in Section 5, demonstrate the practical performance of the parallelized solver for the Elder-type problems with uncertainty coefficients in 3D computational domains. We conclude this work with a discussion in Section 6.

1.1 Our contribution

We applied the gPC expansion to approximate the solution of the density driven flow. This is a time-dependent, nonlinear, second order differential equation. We estimated propagation of input uncertainties in the porosity and permeability into the mass fraction.

2 Problem setup

2.1 Density-driven groundwater flow

The groundwater in the porous medium of the aquifers is subjected to gravitation. For this, any spatial variation of its density induces a flow. In this work, we regard the dependence of the density on the mass fraction of the dissolved salt. Traditional approaches for modelling of this system are described in [5, 6, 7, 20, 66]. In this section, we briefly review the model to introduce the notation.

We consider a domain $\mathcal{D} \subset \mathbb{R}^d$, $d = 3$ consisting of two phases: the immobile solid porous matrix and a mobile liquid in its pores. We denote the porosity of the matrix by $\phi : \mathcal{D} \rightarrow \mathbb{R}$ and its permeability $\mathbf{K} : \mathcal{D} \rightarrow \mathbb{R}^{d \times d}$ and the permeability by $\mathbf{K}$. The liquid is a solution of *NaCl* in water. The mass fraction of the brine (a saturated saltwater solution) in the liquid phase is $c(t, \mathbf{x}) : [0, +\infty) \times \mathcal{D} \rightarrow [0, 1]$. The density liquid phase is $\rho = \rho(c)$ and its viscosity $\mu = \mu(c)$.

The motion of the liquid phase through the solid matrix is characterized by the Darcy velocity $\mathbf{q}(t, \mathbf{x}) : [0, +\infty) \times \mathcal{D} \rightarrow \mathbb{R}^d$. Mass conservation of the liquid phase and the salt can be written as

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

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

where the tensor field $\mathbf{D}$ represents the molecular diffusion and the mechanical dispersion of the salt. For the velocity $\mathbf{q}$, Darcy's law is used:

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

Here, $p = p(t, \mathbf{x}) : [0, +\infty) \times \mathcal{D} \rightarrow \mathbb{R}$ is the hydrostatic pressure and $\mathbf{g} = (0, \dots, -g)^T \in \mathbb{R}^d$ the gravity vector.

In this work, we consider a special case of this general model. We assume that the porous medium is isotropic and characterized by the scalar permeability

$$\mathbf{K} = K\mathbf{I}, \text{ where } K = K(\mathbf{x}) \in \mathbb{R}, \text{ and } \mathbf{I} \in \mathbb{R}^{d \times d} \text{ is the identity matrix.} \quad (4)$$

For the density, we set

$$\rho(c) = \rho_0 + (\rho_1 - \rho_0)c, \quad (5)$$

where ρ_0 and ρ_1 are the densities of pure water and the brine, respectively. Note that $c \in [0, 1]$, where $c = 0$ corresponds to pure water and $c = 1$ to the saturated solution. The viscosity is assumed to be constant. Finally, the dispersion is neglected:

$$\mathbf{D} = \phi D_m \mathbf{I}, \quad (6)$$

where D_m is the molecular diffusion coefficient.

Fields $\phi(\mathbf{x})$ and $K(\mathbf{x})$ will depend on the stochastic variables. Values of the other parameters used in this work are presented in Table 1.

In our numerical experiments, variants of the Elder problem [23, 79] are considered. In it, concentrated solution intrudes the upper boundary into an aquifer filled with pure water. Due to the difference in the density, a complicated, unstable flow arises. This leads to a specific distribution of the mass fraction typically described as “fingering” [40]. The time evolution of the mass fraction and the pressure in (1–3) is determined by the initial conditions for c . Note that the system (1–3) is nonlinear, and the model may have several stationary states [40]. Any changes of the porosity and permeability fields may have an essential effect on the solution: The same initial and boundary data may correspond to principally different asymptotic behaviors. The influence of this factor can be estimated by the application of the stochastic models.

We consider two spatial 3D domains, cf. Fig. 1a: A rectangular parallelepiped $\mathcal{D} = [0, 600] \times [0, 600] \times [0, 150]$ m³ (Fig. 1a(a)) and an elliptic cylinder (Fig. 1a(b)) with $x \in [-300, 300]$ (the long side), $y \in [-150, 150]$ (the short side) and $z \in [-150, 0]$ (the depth). We impose the Dirichlet boundary condition $c = 1$ on the red area on the top of $\mathcal{D}$ ($z = 0$), and $c = 0$ on the green area on the top and on the other boundaries. For equation (1), we impose the no-flux boundary conditions on the bottom and the vertical walls. The boundary conditions for p differ for both geometries: For the parallelepiped, we impose the no-flux boundary conditions on all the boundaries. However, to fix p , we set $p = 0$ on the edges between the green and the blue faces. For the cylinder, we set $p = 0$ on the whole top boundary.

For the spatial discretization of (1)–(3), a vertex-centered FV scheme is used. Degrees of freedom (DoFs) for c and p are located in the grid nodes of the primary mesh. The domain in Fig. 1(a) is covered by a grid of 524,288 hexahedra, so that the total number of the DoFs is 1,098,306. For the domain in Fig. 1(b), we use an unstructured mesh of 241,152 tetrahedra with 89,586 DoFs.

Resolution of the spatial and temporal discretizations play an important role in the accuracy of the simulations. The distribution of the hydrogeological parameters of the porous medium should be represented correctly. Besides that, smoothing the flow on a coarse grid due to the numerical diffusion can lead to a loss of some phenomena and therefore reduce the accuracy of the uncertainty quantification. For this, the grid convergence of the simulations w.r.t. the spatial mesh has been tested. We ran our simulation on the twice regularly refined spatial mesh for geometry from Fig. 1(b). It consists of 15,433,728 cells and has 2,822,552 DoFs. The obtained solution was similar to the solution, computed on a coarser mesh mentioned above. Thus the further refinement does not change the behaviour of the solution essentially. To achieve reasonable computation times, parallelization of the evaluation of every realization is performed.

Table 1: Parameters of the density-driven flow problem

Parameters	Values and Units	Description
$\mathbb{E}(\phi)$	0.1 [-]	mean value of the porosity
D_m	$0.565 \cdot 10^{-6}$ [m ² · s ⁻¹]	molecular diffusion
$\mathbb{E}(\mathbf{K})$	$4.845 \cdot 10^{-13}$ [m ²]	mean value of the permeability
$\mathbf{g}$	$9.81\mathbf{e}_z$ [m · s ⁻²]	gravity
ρ_0	1000 [kg · m ⁻³]	density of pure water
ρ_1	1200 [kg · m ⁻³]	density of brine
μ	10^{-3} [kg · m ⁻¹ · s ⁻¹]	viscosity

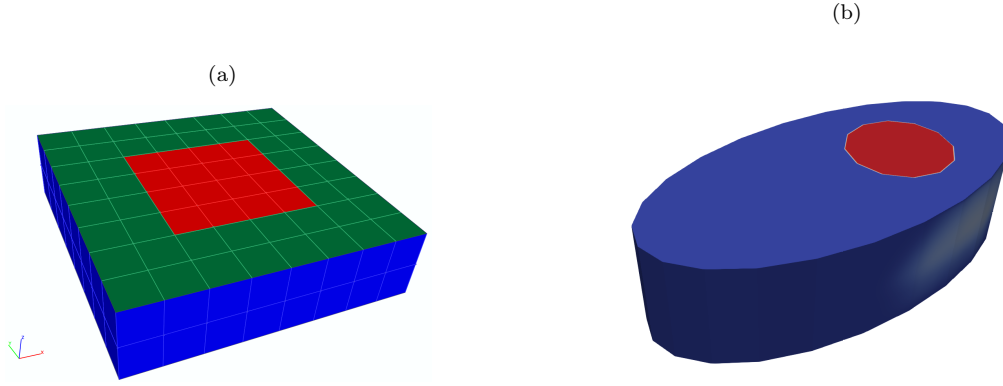


Fig. 1: Geometries of the domains used in the 3d simulations. Boundary conditions for the salt: $c = 1$ in the red area on the top and $c = 0$ otherwise.

2.2 Stochastic modeling of porosity, permeability and mass fraction

Due to lack of knowledge or inaccurate measurements, the two input parameters porosity and permeability are unknown. One of the approaches to deal with this situation is to say that both input parameters are uncertain. For example, we can introduce a prior distribution for these two values. We also can, based on expert knowledge, assume that the mean and the variance values are known.

In the following we model the porosity ($\phi(\mathbf{x})$) by a random field ($\phi(\mathbf{x}, \boldsymbol{\theta})$). We also assume that the permeability ($\mathbf{K}$) is a function of ϕ as shown in (7) and (8):

$$\mathbf{K} = K(\phi)\mathbf{I}. \quad (7)$$

Since parameters are uncertain the solution c is uncertain too, and this solution is a function of ϕ and $\mathbf{K}$. Here we assume $\mathbf{K}$ to be isotropic. Usually the distribution of $\phi(\mathbf{x}, \boldsymbol{\theta})$, $\mathbf{x} \in \mathcal{D}$, is non-trivial or even very complicated. Therefore, it is comfortable from the numerical point of view, to represent this complicated random field in some easy to compute basis, e.g. with Gaussian random variables. Often, many Gaussian random variables are required. We introduce $\boldsymbol{\theta} = (\theta_1, \dots, \theta_M, \dots)$, where θ_i are Gaussian random variables.

The dependence (7) is specific for every material and there is no a general law. In our experiments, we use a Kozeny-Carman-like dependence [18]:

$$K(\phi) = \kappa_{KC} \cdot \frac{\phi^3}{1 - \phi^2}, \quad (8)$$

where the scaling factor κ_{KC} is chosen to satisfy the equality $K(\mathbb{E}(\phi))\mathbf{I} = \mathbb{E}(\mathbf{K})$. Further parameters of the standard Elder problem are listed in Table 1.

2.3 Solution of the stochastic flow model

We start with sampling random variables $\{\boldsymbol{\theta}_i\}$ and computing the realisations of porosity $\phi_i(\mathbf{x}) = \phi(\mathbf{x}, \boldsymbol{\theta}_i)$ and permeability $\mathbf{K}_i(\mathbf{x}) = \mathbf{K}(\mathbf{x}, \boldsymbol{\theta}_i)$, and then solving equations (1–2). To compute the solution we used plugin d³f of the simulation framework ug4, [69, 78]. This framework presents a flexible tool for the numerical solution of non-stationary and nonlinear systems of PDEs, and is parallelized and highly scalable. Equations 1–2 are discretized by a vertex-centered finite-volume method on unstructured grids in the geometric space, as presented in [30]. In particular, the consistent velocity approach is used for the approximation of the Darcy velocity (3), [29, 31]. The implicit Euler scheme is used for the time discretization. The implicit time-stepping scheme is chosen for its unconditional stability, as the velocity depends on the unknowns of the system. This is especially

important for variable coefficients as it is difficult to predict the range of the variation of the velocity in the realizations.

In this discretization, we obtain a large sparse system of nonlinear algebraic equations in every time step, and we solve it using the Newton method with a line search method. The sparse linear system, which appears in the nonlinear iterations, is solved by the BiCGStab method [85] with the multigrid preconditioning (V-cycle, [37]), which proved to be very efficient in this case. In the multigrid cycle, we use the ILU $_{\beta}$ -smoothers [38].

3 Stochastic Modeling and Methods

Let $\boldsymbol{\theta} = (\theta_1, \dots, \theta_M)$ be a vector of independent random variables. Let $S := (\Theta, \mathcal{B}, \mathbb{P})$ be a probability space. Here Θ denotes a sample space, $\mathcal{B}$ a σ -algebra on Θ , and $\mathbb{P}$ a probability measure on $(\Theta, \mathcal{B})$. Random variable $\theta : \Theta \rightarrow \mathbb{R}^M$ with finite variance can be represented in terms of the polynomial chaos expansion, e.g. as a multivariate Hermite polynomial of Gaussian RVs, [80]. Alternatively to Hermite polynomials, other orthogonal polynomials (Legendre, Chebyshev, Laguerre) [1, 82, 83], or splines, for instance, can be used. We assume $\{\Psi_{\beta}\}_{\beta \in \mathcal{J}}$ is a basis of $S = L_2(\Theta, \mathbb{P})$. The cardinality of the multi-index set $\mathcal{J}$ is infinite, therefore, we truncate it and obtain a finite set $\mathcal{J}_{M,p}$ and a subspace $S_{M,p}$. The solution $c(t, \mathbf{x}, \boldsymbol{\theta})$ lies in the tensor product space $L_2(D) \otimes L_2(\Theta) \otimes L_2([0, T])$, where $[0, T]$ is a time interval, where the initial problem is solved. After a discretization, we obtain $\mathbf{c} \in \mathbb{R}^n \otimes S_{M,p} \otimes \mathbb{R}^{n_t}$, where n is the number of basis functions in the physical space, $|\mathcal{J}_{M,p}|$ dimension of the stochastic space, and n_t dimension of the temporal space.

The empirical mean value and the variance of $c(t, \mathbf{x}, \boldsymbol{\theta})$ can be computed as follow

$$\bar{c}(t, \mathbf{x}) = \mathbb{E}(c(t, \mathbf{x}, \boldsymbol{\theta})) = \int_{\Theta} c(t, \mathbf{x}, \boldsymbol{\theta}) \mathbb{P}(d\boldsymbol{\theta}) \approx \sum_{i=1}^{N_q} w_i c(t, \mathbf{x}, \boldsymbol{\theta}_i), \quad (9)$$

with some quadrature points $\boldsymbol{\theta}_i$ and weights w_i . Here $c(t, \mathbf{x}, \boldsymbol{\theta}_i)$ is the solution evaluated at fixed point $\boldsymbol{\theta} = \boldsymbol{\theta}_i \in \mathbb{R}^M$. The sampled variance is computed as follow

$$\text{Var}[c](t, \mathbf{x}) \approx \sum_{i=1}^{N_q} w_i (\bar{c}(t, \mathbf{x}) - c(t, \mathbf{x}, \boldsymbol{\theta}_i))^2. \quad (10)$$

Other well-known methods to compute the mean and the variance are quasi-Monte Carlo, Monte Carlo and multi-level Monte Carlo methods.

3.1 Generalized Polynomial Chaos Expansion

As outlined in [80, 81, 82], the random variable (field) $\mathbf{c}(t, \mathbf{x}, \boldsymbol{\theta})$ can be represented as a series of multivariate Legendre polynomials in uncorrelated and independent uniformly distributed random variables $\boldsymbol{\theta} = (\theta_1, \dots, \theta_j, \dots)$:

$$\mathbf{c}(t, \mathbf{x}, \boldsymbol{\theta}) = \sum_{\beta \in \mathcal{J}} \mathbf{c}_{\beta}(t, \mathbf{x}) \Psi_{\beta}(\boldsymbol{\theta}), \quad (11)$$

where $\{\Psi_{\beta}\}$ is a multivariate Legendre basis, $\beta = (\beta_1, \dots, \beta_j, \dots)$ is a multiindex and $\mathcal{J}$ is a multiindex set.

This expansion is called the *generalized polynomial chaos expansion* (gPC) (see more in the Appendix or in [26, 80, 82, 83]). For numerical purposes, we truncate this infinite gPC series and obtain a finite series approximation, i.e., we keep only M random variables, $\boldsymbol{\theta} = (\theta_1, \dots, \theta_M)$, and limit the maximal order of the multi-variate polynomial $\Psi_{\beta}(\boldsymbol{\theta})$ by p . We denote the new multiindex subset as $\mathcal{J}_{M,p} \subset \mathcal{J}$. Then the unknown random field $\mathbf{c}(t, \mathbf{x}, \boldsymbol{\theta})$ can be approximated as follows:

$$\mathbf{c}(t, \mathbf{x}, \boldsymbol{\theta}) \approx \widehat{\mathbf{c}}(t, \mathbf{x}, \Theta) = \sum_{\beta \in \mathcal{J}_{M,p}} \mathbf{c}_{\beta}(t, \mathbf{x}) \Psi_{\beta}(\boldsymbol{\theta}). \quad (12)$$

Here $\Psi_\beta(\boldsymbol{\theta})$ are multi-variate Legendre polynomials defined as follows

$$\Psi_\beta(\boldsymbol{\theta}) := \prod_{j=1}^M \psi_{\beta_j}(\theta_j); \quad \forall \boldsymbol{\theta} \in \mathbb{R}^M, \beta \in \mathcal{J}_{M,p}, \sum_{j=1}^M \beta_j \leq p,$$

$\psi_{\beta_j}(\cdot)$ are Legendre monomials, defined in (26). The scalar product is

$$\langle \Psi_\alpha, \Psi_\beta \rangle_{L_2(\Theta)} = \mathbb{E}(\Psi_\alpha(\boldsymbol{\theta})\Psi_\beta(\boldsymbol{\theta})) = \int_{\Theta} \Psi_\alpha(\boldsymbol{\theta})\Psi_\beta(\boldsymbol{\theta})d\mathbb{P}\boldsymbol{\theta} = Q_\alpha \delta_{\alpha\beta}, \quad (13)$$

where $Q_\alpha = \mathbb{E}(\Psi_\alpha^2) = q_{\alpha_1} \cdot \dots \cdot q_{\alpha_d}$, $q_{\alpha_j} = \langle \psi_{\alpha_j}, \psi_{\alpha_j} \rangle = 1/(2\alpha_j + 1)$, are the normalization constants and $\delta_{\alpha\beta} = \delta_{\alpha_1\beta_1} \cdot \dots \cdot \delta_{\alpha_M\beta_M}$ is the M -dimensional Kronecker delta function.

Decomposition (11) can be understood as a response surface for $\mathbf{c}(t, \mathbf{x}, \boldsymbol{\theta})$. As soon as the response surface is built, the value $\mathbf{c}(t, \mathbf{x}, \boldsymbol{\theta})$ can be evaluated for any $\boldsymbol{\theta}$ almost for free (only by evaluating the polynomial (11)). It can be very practical if 10^6 samples of $\mathbf{c}(t, \mathbf{x}, \boldsymbol{\theta})$ are needed, for example. Since Legendre polynomials are L_2 -orthogonal, the coefficients $\mathbf{c}_\beta(t, \mathbf{x})$ can be computed by projection:

$$\mathbf{c}_\beta(t, \mathbf{x}) = \frac{1}{\langle \Psi_\beta, \Psi_\beta \rangle} \int_{\Theta} \widehat{\mathbf{c}}(t, \mathbf{x}, \boldsymbol{\theta}) \Psi_\beta(\boldsymbol{\theta}) \mathbb{P}(d\boldsymbol{\theta}) \approx \frac{1}{\langle \Psi_\beta, \Psi_\beta \rangle} \sum_{i=1}^{N_q} \Psi_\beta(t, \boldsymbol{\theta}_i) \widehat{\mathbf{c}}(t, \mathbf{x}, \boldsymbol{\theta}_i) w_i, \quad (14)$$

where w_i are weights and $\boldsymbol{\theta}_i$ quadrature points, defined, for instance, by a Gauss-Legendre integration rule. Once all gPC coefficients are computed we substitute them into (11). For estimates of truncation and aliasing errors, see [16]. There are different strategies to truncate the multi-index set $\mathcal{J}$ to $\mathcal{J}_{M,p}$. In the multi-index set $\mathcal{J}_{M,p}$ we keep only M RVs, i.e., $\boldsymbol{\theta} = (\theta_1, \dots, \theta_M)$ and p_i is the monomial degree w.r.t. the variable θ_i . For the total multi-variate polynomial degree p we pose one of the following condition a) $\prod_{i=1}^M p_i \leq p$ or b) $\sum_{i=1}^M p_i \leq p$ or c) $p_i \leq p, \forall i = 1..M$.

Remark 1. If only the variance and the mean values are of interest, we recommend to use the collocation method with some sparse or full-tensor grid. If some additional statistics are required, such as sensitivity analysis, parameter identification, or computing cdf or pdf, we recommend to compute the gPC expansion.

Quadrature grids for computing multi-dimensional integrals can be constructed from products of 1D integration rules. To keep computational costs to a minimum, a sparse grid technique can be used for certain class of smooth functions [11]. To compute (14), we apply the well-known Clenshaw Curtis quadrature rules [15].

Since $\Psi_0(\boldsymbol{\theta}) = 1$, it becomes apparent that the mean value is the first gPC coefficient

$$\bar{\mathbf{c}} := \mathbb{E}(\mathbf{c}(t, \mathbf{x}, \boldsymbol{\theta})) = \mathbf{c}_0(t, \mathbf{x}) = \frac{1}{\langle \Psi_0, \Psi_0 \rangle} \int_{\Theta} \mathbf{c}(t, \mathbf{x}, \boldsymbol{\theta}) \Psi_0(\boldsymbol{\theta}) \mathbb{P}(d\boldsymbol{\theta}) \approx \sum_{i=1}^{N_q} \mathbf{c}(t, \mathbf{x}, \boldsymbol{\theta}_i) w_i, \quad (15)$$

$\mathbf{0} = (0, \dots, 0)$. The variance is the sum of squared gPC coefficients [33, 44, 82]:

$$\text{Var}[\mathbf{c}(t, \mathbf{x}, \boldsymbol{\theta})] = \mathbb{E}((\mathbf{c}(t, \mathbf{x}, \boldsymbol{\theta}) - \bar{\mathbf{c}}) \otimes (\mathbf{c}(t, \mathbf{x}, \boldsymbol{\theta}) - \bar{\mathbf{c}})) \quad (16)$$

$$= \sum_{\gamma > 0} \|\Psi_\gamma\|^2 \mathbf{c}_\gamma(t, \mathbf{x}) \otimes \mathbf{c}_\gamma(t, \mathbf{x}) = \sum_{\gamma > 0} \prod_{k=1}^M \frac{1}{2\gamma_k + 1} \mathbf{c}_\gamma(t, \mathbf{x}) \otimes \mathbf{c}_\gamma(t, \mathbf{x}), \quad (17)$$

where $\otimes$ is the Kronecker product.

3.2 Computing probability density functions

After we calculated the gPC surrogate $\widehat{\mathbf{c}}(t, \mathbf{x}, \boldsymbol{\theta})$ in (12), we can compute the probability density functions, exceedance probabilities, and quantiles in the selected points. For this, one should evaluate the multi-variate polynomial $\widehat{\mathbf{c}}(t, \mathbf{x}, \boldsymbol{\theta})$ in a sufficiently large number of points $\boldsymbol{\theta}$

$$\widehat{c}(\boldsymbol{\theta}) := \widehat{c}(t^*, \mathbf{x}^*, \boldsymbol{\theta}) = \sum_{\beta \in \mathcal{J}_{M,p}} c_\beta(t^*, \mathbf{x}^*) \Psi_\beta(\boldsymbol{\theta}). \quad (18)$$

The random variable $\widehat{c}(\boldsymbol{\theta})$ can be sampled, e.g., $N_s = 10^6$ times (no additional extensive simulations are required). The obtained sample can be used to evaluate the required statistics, and the probability density function. The exceedance probability for some threshold c^* can be estimated as follows:

$$P(c > c^*) \approx \frac{\#\{\widehat{c}(\boldsymbol{\theta}_i) : \widehat{c}(\boldsymbol{\theta}_i) > c^*, i = 1, \dots, N_s\}}{N_s}. \quad (19)$$

Having a sufficiently large sample set makes it possible to estimate quantiles [47].

3.3 Numerical errors

Applying gPC approximation, we introduce the truncation and approximation errors [16, 17, 26, 74]. Truncating gPC coefficients $\{\mathbf{c}_\beta : \beta \in \mathcal{J}_c\}$, we introduce the truncation error

$$\text{err}_t = \|\mathbf{c}(t, \mathbf{x}, \boldsymbol{\theta}) - \widehat{\mathbf{c}}(t, \mathbf{x}, \boldsymbol{\theta})\| = \left\| \sum_{\beta \in \mathcal{J}_c} \mathbf{c}_\beta(t, \mathbf{x}) \Psi_\beta(\boldsymbol{\theta}) \right\|, \quad \mathcal{J}_{M,p} \cup \mathcal{J}_c = \mathcal{J}. \quad (20)$$

Additionally, we approximate the coefficients $\mathbf{c}_\beta(t, \mathbf{x})$ by $\widehat{\mathbf{c}}_\beta(t, \mathbf{x})$. This introduces the approximation error

$$\text{err}_a = \left\| \sum_{\beta \in \mathcal{J}_{M,p}} \mathbf{c}_\beta(t, \mathbf{x}) \Psi_\beta(\boldsymbol{\theta}) - \sum_{\beta \in \mathcal{J}_{M,p}} \widehat{\mathbf{c}}_\beta(t, \mathbf{x}) \Psi_\beta(\boldsymbol{\theta}) \right\| = \left\| \sum_{\beta \in \mathcal{J}_{M,p}} (\widehat{\mathbf{c}}_\beta(t, \mathbf{x}) - \mathbf{c}_\beta(t, \mathbf{x})) \Psi_\beta(\boldsymbol{\theta}) \right\| \quad (21)$$

The sum of both errors is

$$\text{err}_t + \text{err}_a = \underbrace{\left\| \sum_{\beta \in \mathcal{J}_c} \mathbf{c}_\beta(t, \mathbf{x}) \Psi_\beta(\boldsymbol{\theta}) \right\|}_{\text{truncation error}} + \underbrace{\left\| \sum_{\beta \in \mathcal{J}_{M,p}} (\mathbf{c}_\beta(t, \mathbf{x}) - \widehat{\mathbf{c}}_\beta(t, \mathbf{x})) \Psi_\beta(\boldsymbol{\theta}) \right\|}_{\text{approximation error}}. \quad (22)$$

Remark 2. Spatial resolution plays an important role in the simulations. Thus, the distribution of the parameters of the porous medium should be represented correctly. Furthermore, smoothing the flow on a coarse grid due to numerical diffusion can lead to a loss of some phenomena and therefore reduce the accuracy of the uncertainty quantification. Thus we must cover $\mathcal{D}$ with sufficiently fine grids and use small time steps. To achieve reasonable computation times, this also assumes parallelization of the evaluation of every realization.

4 Parallelization

The computation of the mean value, variance, and other statistics are done in parallel. There are two levels of parallelization. First, we compute the solution in all quadrature or sampling points in parallel (no communication is needed). Second, the solution in each single quadrature point is computed also in parallel (communication is needed).

We performed computations on a Cray XC40 parallel cluster Shaheen II provided by the King Abdullah University of Science and Technology (KAUST) in Saudi Arabia. Shaheen II has 6174 nodes with 2 Intel Haswell 2.3 GHz CPUs per node, and 16 processor cores per CPU. Every node has 128 GB RAM.

On the first level, we allocated, depending on the experiment, up to 600 computing nodes. It allowed us to compute (up to) 600 qMC simulations simultaneously. All these simulations are independent, and, therefore, no communication between nodes was required. The total computing time is equal to the time, required for computing one simulation. On the second level, on each node, we allocated 32 computing cores to resolve each single simulation. Thus, in total, we run our code on up to $600 \times 32 = 19200$ parallel cores. If a simulation is large or requires a lot of memory, we can allocate more than one node for it. `ug4` allows allocating all available

nodes (6174) for solving just one expensive simulation. Theoretically, ug4 allows using all available nodes and cores, i.e., $6174 \times 32 = 197,568$ cores.

Parallelization of the framework ug4 is based on the distribution of the spatial grid between the processes and is implemented using MPI [69]. As the computation of the realizations involves the nonlinear iterations and the solution of the large sparse linear systems (see Section 2.3), quite intensive communication is required. Shaheen II used the SLURM system to manage parallel jobs, and the job arrays were used for the concurrent computations of the realizations. This was the most time-consuming phase, which required hours of computing time.

5 Numerical Experiments

5.1 Solving the deterministic problem

The deterministic solver ug4 uses the multigrid solver with a hierarchy of L nested spatial grids in $\mathcal{D}$. There are N_ℓ nodes on each level $\ell = 1, \dots, L$. Each grid is distributed over all available processors on the node (32 in the tests below).

Equations 1–2 are discretized by a vertex-centered finite-volume method on unstructured grids in space, as presented in [30]. In particular, we use the “consistent velocity” for the approximation of the Darcy velocity (3), cf. [29, 31]. The implicit Euler scheme is used for the time discretization. The choice of the implicit time stepping scheme is motivated by its unconditional stability, as the velocity depends on the unknowns of the system.

After discretization one obtains a large sparse system of nonlinear algebraic equations in every time step. It is solved by the Newton method with a line search method. The sparse linear system, which appears in the nonlinear iterations, is solved by the BiCGStab method (cf. [85]) with the multigrid preconditioning (V-cycle, cf. [37]) which proved to be very efficient in this case. In the multigrid cycle, we use the ILU_β -smoothers [38]. The matrices are assembled as the Jacobians of the discretized nonlinear systems for each grid. On each step of the Newton method the inversion of the Jacobian is computed. Developers of ug4 also efficiently solve the problem of optimal distribution of the hierarchy of grids onto parallel processes.

Software. ug4 is a novel flexible software system for simulating PDE based models on high performance parallel clusters [69, 78]. The software toolbox ug4 has been developed since the early 1990’s. This software framework has successfully been applied to a variety of problems such as density-driven and thermohaline flow in porous media, Navier-Stokes equation, drug diffusion through human skin, level-set methods. The aim of the software framework ug4 is to solve the above mentioned problems on adaptive hybrid grid hierarchies in 2D and 3D dimensions. Special care is taken to parallel performance issues, scalability, linear algebra algorithms, and data structures. In addition, ug4 is specially designed for user-friendly handling and functionality is made available to non-programming users by the usage of scripts and graphical representations. The numerical solution of the systems (1–2) has been performed using the D3F plugin of the simulation software package ug4, cf. [69, 78].

5.2 3D computational domains

The geometry of real-life 3D computational domains can be very complicated. Discretisation of such domains (reservoirs) may require non-trivial 3D meshes. This could be very challenging in the stochastic context, which we have here. Therefore, we start with two relatively trivial reservoir geometries: 1) a parallelepiped $\mathcal{D} = [0, 600] \times [0, 600] \times [0, 150]$ (see Fig. 1a); 2) an elliptic cylinder with the semi-axes 600 and 300 for the base ellipse and height 150. Thus, the longest radius is 300, $x \in [-300, 300]$, the shortest radius is 150, $y \in [-150, 150]$, the height $z \in [-150, 0]$ and the center is in the point $(0, 0, -75)$ (see Fig. 1b). The polluted area is a circle with the center in $(-150, 0, 0)$, and the radius 100.

5.3 Parallelepiped with 3 RVs

We consider an experiment with three random variables $\boldsymbol{\xi} = (\xi_1, \xi_2, \xi_3)$ and with the following porosity coefficient

$$\phi(\mathbf{x}, \boldsymbol{\xi}) := 0.1 + 0.01 \left(\xi_1 \sin \frac{x\pi}{600} + \xi_2 \sin \frac{y\pi}{600} + \xi_3 \sin \frac{z\pi}{150} + \xi_1 \sin \frac{x\pi}{600} \sin \frac{y\pi}{600} + \xi_2 \sin \frac{x\pi}{600} \sin \frac{z\pi}{150} \right), \quad (23)$$

where $\mathbf{x} = (x, y, z) \in [0, 600] \times [0, 600] \times [0, 150]$ (parallelepiped).

In Fig. 2, we visualize isosurfaces $\{\mathbf{x} : \bar{c}(\mathbf{x}) = 0.5\}$ after 1500 time steps ($0.005 \cdot 1500 = 7.5$ years), computed using (a) 200 qMC simulations (Halton sequence) and (b) gPC response surface of order 4. All gPC coefficients were computed via a 3D Gauss-Legendre quadrature rule over the domain $[-1, 1]^3$ with 125 quadrature points.

Note that in the present problem setting, the flow field is very complicated. The main process tends to represent the same behaviour as in the 2D version of the Elder problem. The heavy concentrated solution from the top boundary falls down and replaces the pure water located initially at the bottom. This pure water is pressed up. As the diffusion is very small, the transport of the salt by these opposite flows creates a non-trivial pattern of the concentration field typically described as “fingering”. The particular pattern depends on the profile of the boundary conditions for c and the initial flow. In our case, we see 5 “fingers” (4 at the corners and one at the center). As we see, the isosurfaces $\{\mathbf{x} : \bar{c}(\mathbf{x}) = 0.5\}$ are almost identical for both the methods.

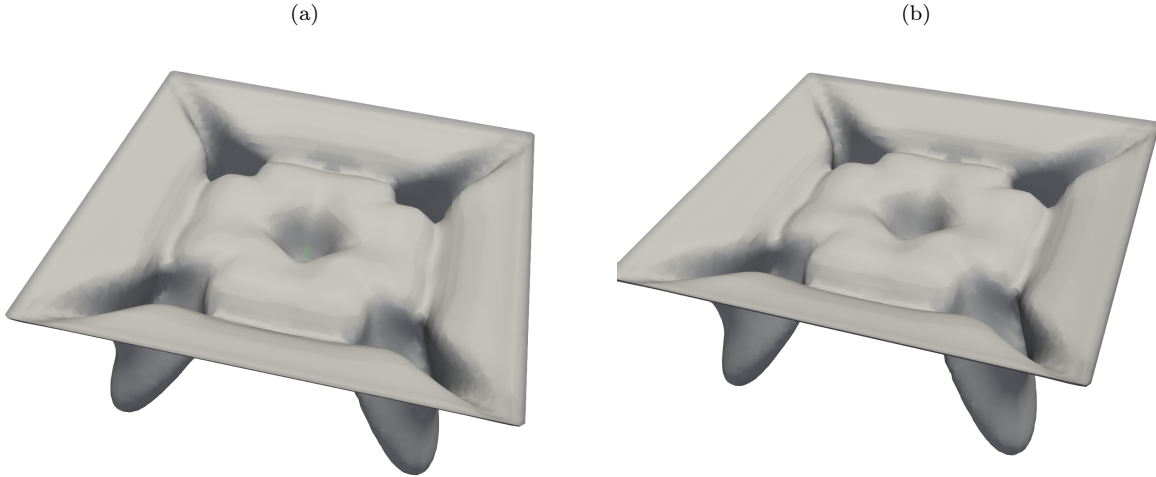


Fig. 2: Isosurface $\{\mathbf{x} : \bar{c}(\mathbf{x}) = 0.5\}$ computed using (a) 200 qMC simulations and (b) gPC response surface of order 4. The upper square edge corresponds to the boundary of the red area in Fig. 1(a)

The same holds for the variance. Fig. 3 presents the isosurface $\{\mathbf{x} : \text{Var}[c](\mathbf{x}) = 0.05\}$ after these 1500 time steps for (a) 200 qMC simulations (Halton sequence), (b) gPC response surface of order 4 and (c) comparison of the isosurfaces from (a) and (b). We see that positions, numbers, and shapes of all fingers are similar. The isosurfaces, computed via qMC, are slightly larger, especially one in the middle (see Fig. 3c).

Five different contour lines (isosurfaces) of the deterministic solution $c(\mathbf{x}, \boldsymbol{\theta} = 0)$ for the porosity, defined in (23), are shown in Figures 4.

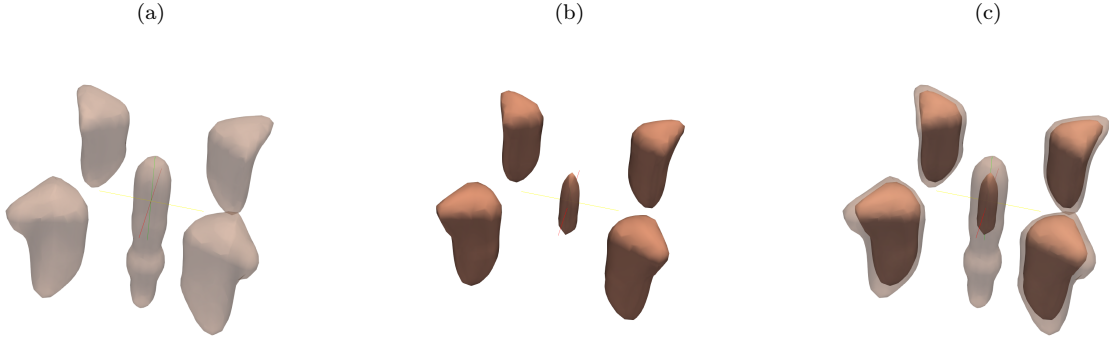


Fig. 3: Isosurface $\{\mathbf{x} : \text{Var}[c] = 0.05\}$, computed via a) qMC and b) gPC response surface of order 4, and c) comparison of both isosurfaces.

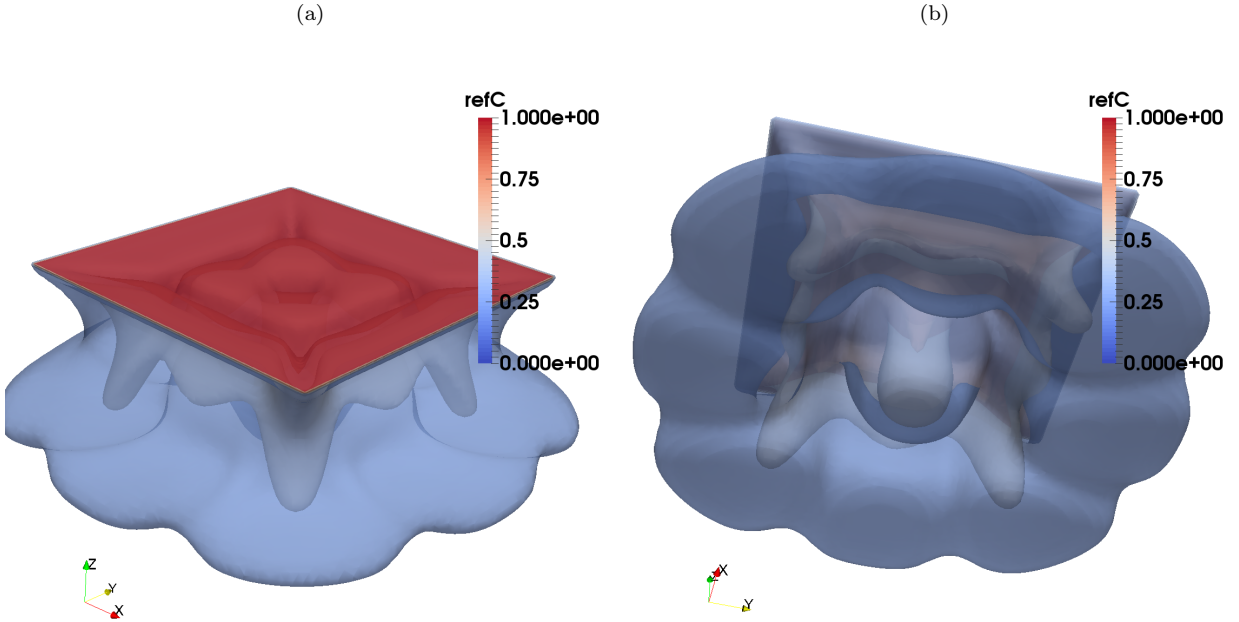


Fig. 4: Five isosurfaces of the deterministic solution $\{\mathbf{x} : c(t, \mathbf{x}, 0)\} = \{1.0, 0.75, 0.5, 0.25, 0.0\}$ after $t = 9.6$ years; (a) view from the top; (b) view from the bottom. The upper square edge corresponds to the boundary of the red area in Fig. 1(a).

5.4 Elliptical cylinder with 3 RVs

In this example we consider a cylindrical domain (see Fig. 1b). The goal is to see the influence of the computing domain on the solution. As we can see, the fingers have different shapes and numbers.

The porosity coefficient has three layers and is defined with three RVs as follows:

$$\phi(t, \mathbf{x}, \boldsymbol{\theta}) = 0.1 + 0.05 \cdot c_0 \cdot \left(\frac{\xi_1 x}{600} \cos \frac{\pi x}{300} + \xi_2 \sin \frac{\pi y}{150} + \xi_3 \cos \frac{\pi x}{300} \sin \frac{\pi y}{150} \right) \quad (24)$$

$$\begin{cases} c_0 = 0.01 & \text{if } z \leq -100 \\ c_0 = 0.10 & \text{if } -100 < z \leq -50 \\ c_0 = 1.0 & \text{if } -50 < z \leq 0 \end{cases} \quad (25)$$

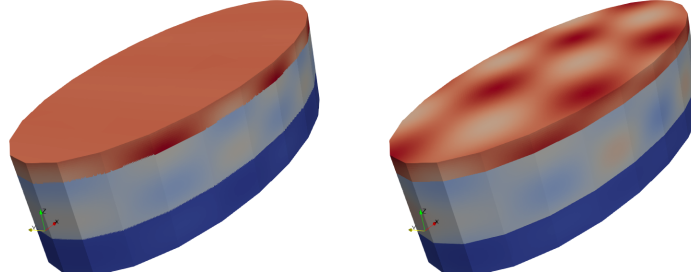


Fig. 5: Two realizations of the porosity field, $\phi(\mathbf{x}) \in [0.079, 0.13]$

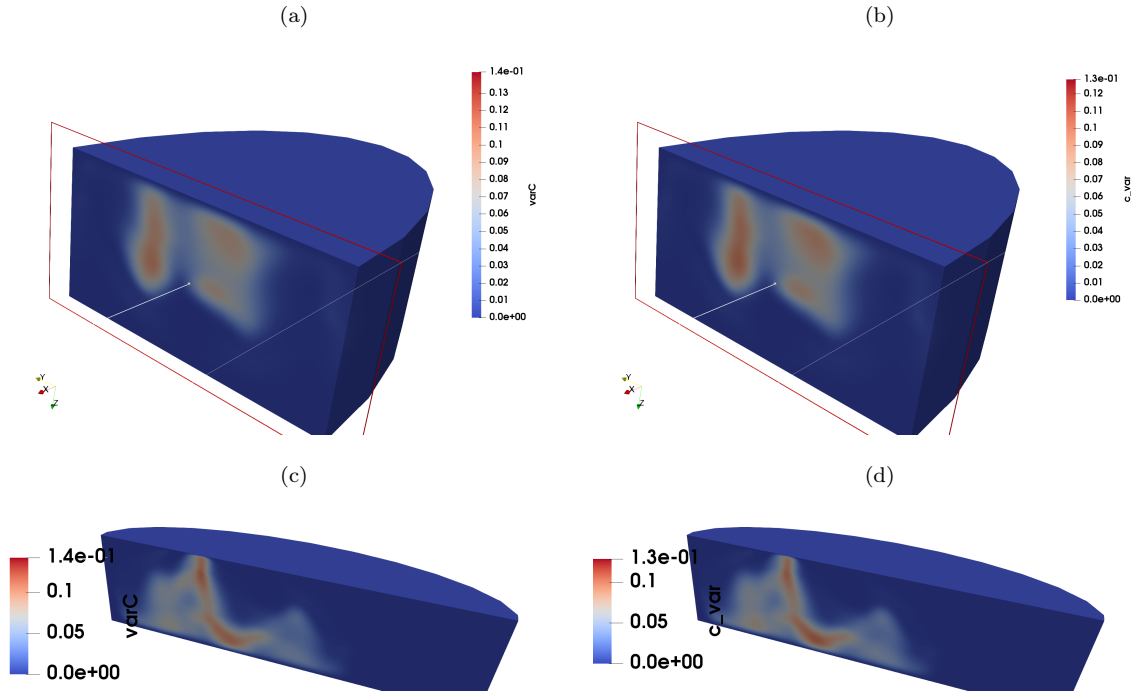


Fig. 6: Variance of the mass fraction in the cylindrical 3D reservoir, computed via 200 qMC simulations (a) and (c) and gPC of order 4 with 35 gPC coefficients (b) and (d). $\text{Var}[c]_{qMC} \in [0, 0.14]$, $\text{Var}[c]_{gPC} \in [0, 0.13]$. The first cutting plane on (a)-(b) has normal vector $(1, 0, 0)$, the second on (c)-(d) $(0, 1, 0)$.

Figure 5 shows two random realizations of the porosity field.

In Fig. 6 we present a comparison of the variance of c , computed via qMC (200 simulations) and gPC of order $p = 4$. All $\frac{(m+p)!}{m!p!} = \frac{7!}{3!4!} = 35$ gPC coefficients are computed with a full Clenshaw-Curtis quadrature, containing 125 quadrature points. In Figures 6a and 6b the cutting plane has the normal vector $(1, 0, 0)$, and in Fig. 6c and 6d vector $(0, 1, 0)$. The variances, computed via qMC and gPC lie in the intervals $\text{Var}[c]_{qMC} \in [0, 0.14]$, and $\text{Var}[c]_{gPC} \in [0, 0.13]$ respectively. The small difference is not really visible in the pictures. Thus, we conclude that the gPC surrogate model, computed on the full Clenshaw-Curtis quadrature grid can be used for approximating the mean and the variance of c .

To demonstrate more similarities between the solutions, computed via qMC and gPC we visualize isosurfaces of the variance of the mass fraction in the cylindrical 3D reservoir (Figures 7 and 8).

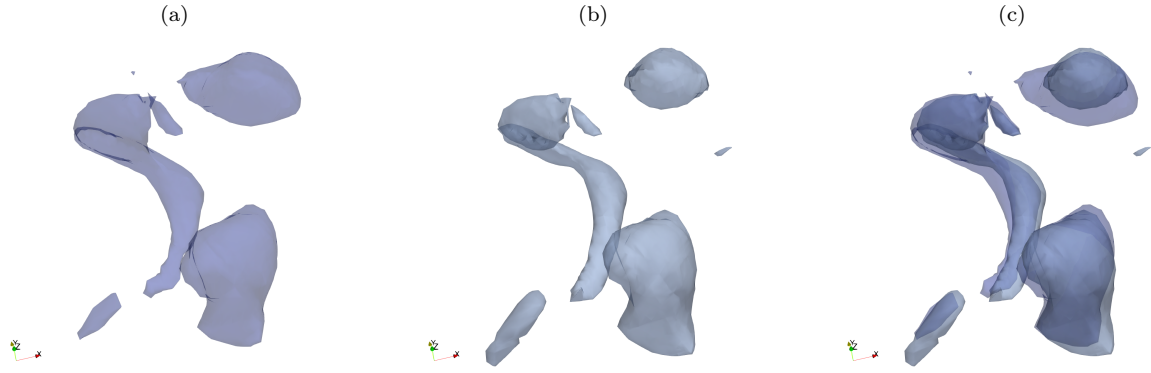


Fig. 7: The contour $\text{Var}[c] = 0.025$, computed by a) qMC , b) gPC and c) comparison of both.

In Figure 7 we demonstrate two isosurfaces $\{\mathbf{x} : \text{Var}[c](\mathbf{x}) = 0.025\}$ computed via a) qMC method, and b) gPC of order 4 surrogate. The Figure 7c shows the comparison of both isosurfaces. As we can see the results are very similar, the isosurface computed via qMC is slightly larger.

In Figure 8 we demonstrate two isosurfaces $\{\mathbf{x} : \text{Var}[c](\mathbf{x}) = 0.07\}$ computed via a) qMC method, Fig. 8a, and b) gPC of order 4 surrogate, Fig. 8b. The Figure 8c shows the comparison of both isosurfaces. As we can see the results are very similar, the isosurface computed via qMC is slightly larger.

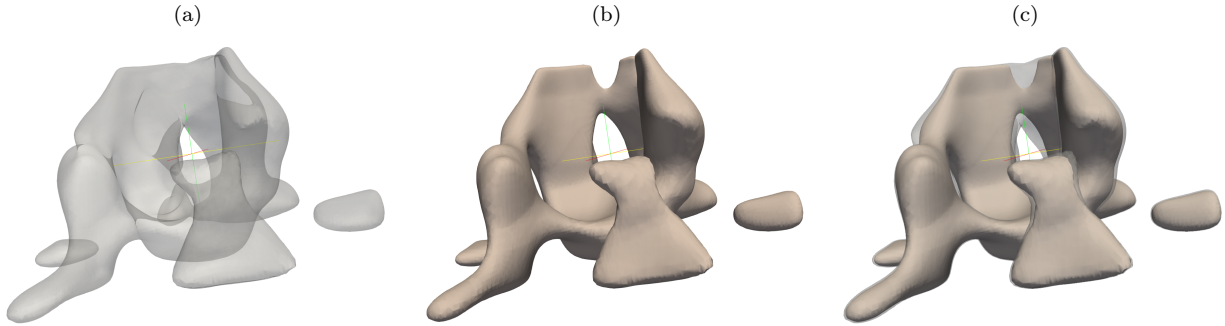


Fig. 8: Isosurface $\{\mathbf{x} : \text{Var}[c](\mathbf{x}) = 0.07\}$, computed via a) qMC and b) gPC response surface of order 4, and c) comparison of both isosurfaces

In Figure 9 we compare two isosurfaces $\{\mathbf{x} : \text{Var}[c](\mathbf{x}) = 0.12\}$ computed via qMC method and gPC of order 4 surrogate. As we can see the results (shape and number) are very similar again. The isosurface computed via qMC is slightly larger.

(a)

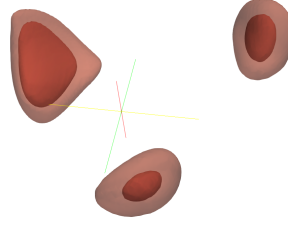
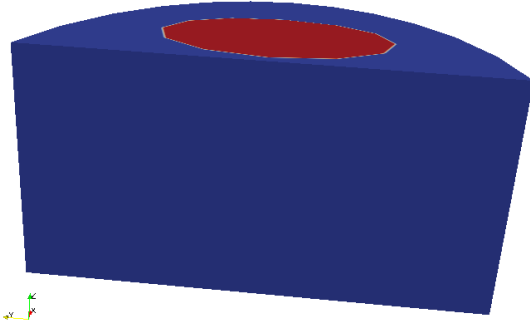
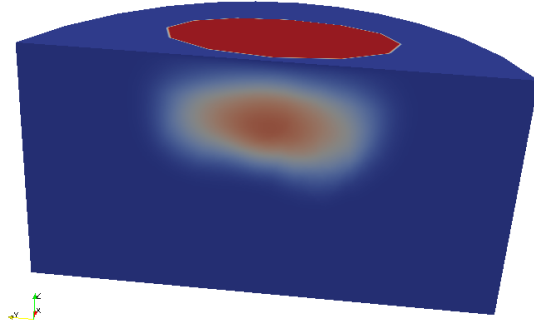


Fig. 9: Comparison of isosurfaces $\{\mathbf{x} : \text{Var}[c](\mathbf{x}) = 0.12\}$, computed via qMC and gPC response surface of order 4. The isosurface computed via qMC is slightly larger.

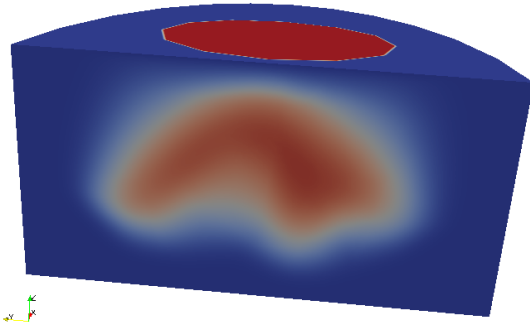
(a)



(b)



(c)



(d)

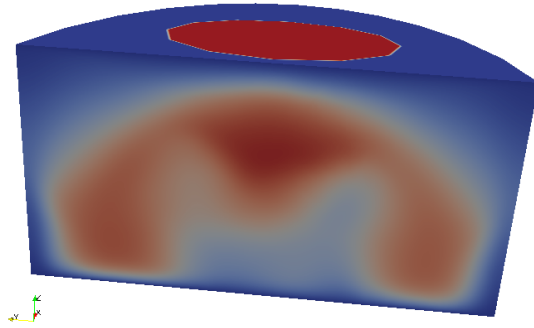


Fig. 10: Propagation of the mean of the mass fraction in the cylindrical 3D reservoir, cutting plane is $(150, y, z)$. Evolution of $\mathbb{E}(c)$ in time after a) 0, b) 0.55, c) 1.1, d) 2.2 years.

5.5 Time evolution of $\bar{c}(t, \mathbf{x})$ and $\text{Var}[c](t, \mathbf{x})$

This is again an experiment with ϕ as in (24)-(25). The gPC coefficients were computed with 125 Gauss-Legendre quadrature points. Figure 10 shows evolution of the mean of the mass fraction in the cylindrical 3D reservoir after a) 0, b) 0.55, c) 1.1, d) 2.2 years. The cutting plane is $(150, y, z)$. One can clearly see different stages of the "finger" building. Figure 11 shows the variance of the mass fraction in the cylindrical 3D reservoir after $\{0.55, 1.1, 2.2\}$ years (after 100, 200, and 400 time steps, respectively).

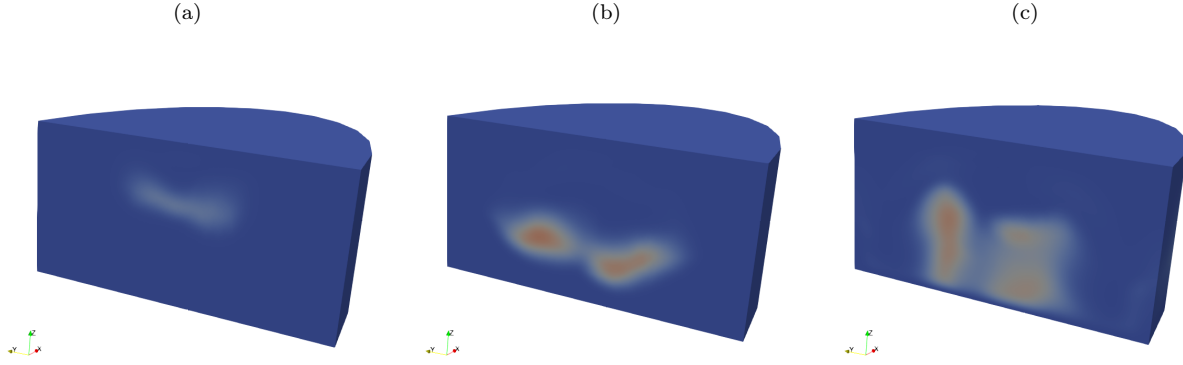


Fig. 11: Propagation of the variance of the mass fraction in the cylindrical 3D reservoir, cutting plane is $(150, y, z)$. Evolution of the mean concentration in time after a) 0.55, b) 1.1, d) 2.2 years (100, 200, 400 time steps respectively). $\text{Var}[c] \in [0, 0.17]$.

5.6 Three layers example with cylindrical domain

The porosity coefficient inside of each level is random and is defined by the following equation

$$\phi(t, \mathbf{x}, \boldsymbol{\theta}) = 0.1 + c_z(\theta_1 \cos(\pi x/600) + \theta_2 \cos(\pi y/300) + \theta_3 \sin(\pi x/600) \cos(\pi z/150)),$$

$$\text{with } c_z = \begin{cases} 0.01, & -150 \leq z < -100 \\ 0.1, & -100 \leq z < -50 \\ 1.0, & -50 \leq z \leq 0. \end{cases}$$

Here, $\boldsymbol{\theta} = (\theta_1, \theta_2, \theta_3)$ is a quadrature point on a full Gaussian-Legendre grid (level 3, dimension 3).

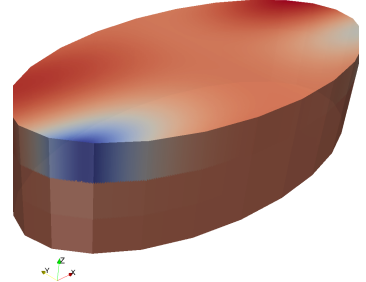


Fig. One realization of the porosity field, $\phi \in [0.05, 0.13]$.

The maximum and the minimum over all porosity samples and over all $\mathbf{x}$ are 0.02 and 0.15, respectively. One random realization of the porosity is shown in Figure on the right. The random porosity has three horizontal layers along z -axes: the first $z \in [-150, -100)$, second $z \in [-100, -50)$, and third $z \in [-50, 0]$.

Figure 12 shows three different isosurfaces of the mean value of the mass fraction: (a) $\{\mathbf{x} : \bar{c}(\mathbf{x}) = 0.65\}$, (b) $\{\mathbf{x} : \bar{c}(\mathbf{x}) = 0.78\}$, and (c) $\{\mathbf{x} : \bar{c}(\mathbf{x}) = 0.95\}$. The large blue area visualizes the reservoir contour and the smaller dark isosurface represents a isosurface of the mean value.

Three isosurfaces of the variance of the mass fraction after 500 time steps (≈ 2.5 years) are shown in Fig. 13. These isosurfaces are $\{\mathbf{x} : \text{Var}[c](\mathbf{x}) = 0.01\}$, (b) $\{\mathbf{x} : \text{Var}[c](\mathbf{x}) = 0.05\}$, and (c) $\{\mathbf{x} : \text{Var}[c](\mathbf{x}) = 0.15\}$, respectively. The gPC surrogate model of order $p = 3$ with $M = 3$ random variables was used to compute the mean and the variance of the mass fraction.

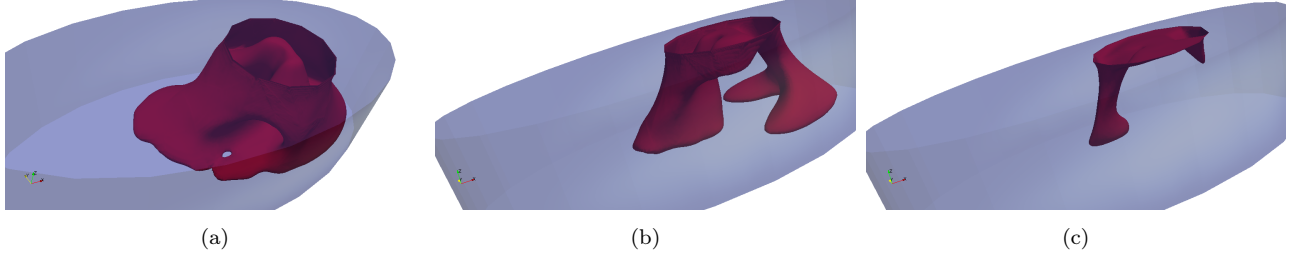


Fig. 12: Isosurfaces of the mean of the mass fraction after 500 time steps (≈ 2.5 years); (a) $\{\mathbf{x} : \bar{c}(\mathbf{x}) = 0.65\}$, (b) $\{\mathbf{x} : \bar{c}(\mathbf{x}) = 0.78\}$, (c) $\{\mathbf{x} : \bar{c}(\mathbf{x}) = 0.98\}$.

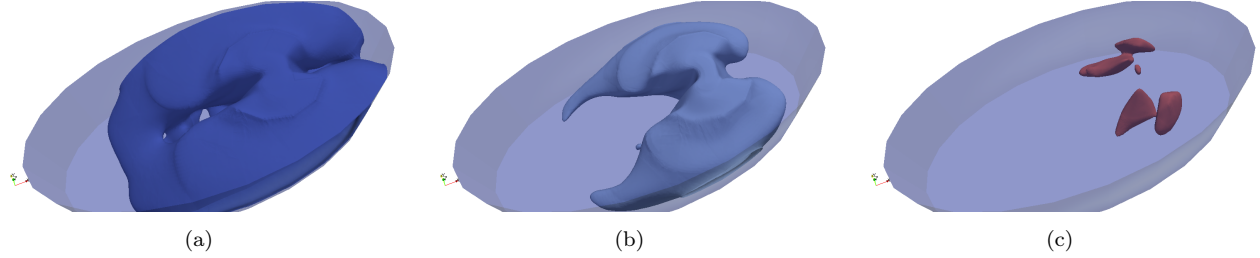


Fig. 13: Isosurfaces of the variance of the mass fraction after 500 time steps, $\{\mathbf{x} : \text{Var}[c](\mathbf{x}) = 0.01\}$, (b) $\{\mathbf{x} : \text{Var}[c](\mathbf{x}) = 0.05\}$, (c) $\{\mathbf{x} : \text{Var}[c](\mathbf{x}) = 0.15\}$.

5.7 Best practices

After numerous numerical tests, we created the best practice guide as below:

1. The number of uncertain variables to describe the porosity coefficient depends on the model and assumptions. We recommend to start with 1-3 variables to understand the phenomena better. We note that a heterogeneous porosity field in 3D may require hundreds of random variables. That means thousands of gPC coefficients. This is computationally un-affordable unless one uses some advance low-rank tensor techniques [21] or multi-scale/homogenization strategy. In this work, we considered three random variables. The more random variables are present, the harder it could be to interpret the results, and additional sensitivity analysis may be required.
2. It is assumed a smooth dependence of the output QoI on the input uncertain parameters. We recommend to check it at least visually before computing gPC .
3. The gPC -based surrogate, for the cases when the variance of the porosity is not so high, showed reliable results. For the validation, one can use a qMC method.
4. We were not able to get a correct variance of c with the gPC method in tests where the variance of the porosity was large.
5. We recommend the maximal polynomial order in gPC to be $p = 2, 3, 4$ or 5 . With our computational budget, we failed to get the correct variance of c for $p > 5$. Probably the quadrature rule was not good enough to catch oscillatory behavior of gPC polynomials. This is a general known drawback of using global polynomials.
6. The coefficients of the gPC surrogate require calculating high-dimensional integrals. For stochastic dimensions 4 and higher, we recommend to use Gauss-Legendre (or Clenshaw-Curtis) sparse grid. For dimensions 1-3, we recommend to use a full grid.
7. The polynomial order in gPC is 2-4. Otherwise, the number of simulations will be too large.
8. The total numerical cost could be reduced by choosing the spatial and time step sizes adaptive, depending on the random perturbation. But in our implementation, it is hard to it do on the fly. Therefore, we recommend “conservative” settings, which are the same for all random simulations. A large time or large spatial step sizes may result in lack of convergence.

6 Conclusion

We solved the density driven groundwater flow problem with uncertain porosity and permeability. An accurate solution of this time-dependent and non-linear problem is impossible due to the presence of natural uncertainties in the porosity and permeability in the reservoir. Therefore, we estimated the mean value and the variance of the solution, as well as the propagation of uncertainties from the random input parameters to the solution. For this, we, first, computed a multi-variate polynomial approximations (gPC) to the solution and then used it to estimate the required statistics. Utilizing the gPC method allows us to reduce the computational cost compare of the classical Monte Carlo method. gPC assumes that the output function $c(t, \mathbf{x}, \boldsymbol{\theta})$ is square-integrable and smooth w.r.t uncertain input variables $\boldsymbol{\theta}$.

We considered two different aquifers - a solid parallelepiped and a solid elliptic cylinder. One of our goals was to see how the domain geometry influences the shape of fingers.

Since the considered problem is nonlinear, a high variance in the porosity may result in totally different solutions, for instance, the number of fingers, their intensity, the shape, the propagation time, and the velocity may strongly variate.

From the implementation point of view, we have demonstrated how to use the parallel multigrid solver ug4 as a black-box solver. An additional novelty of this work is that all algorithms for uncertainty quantification are implemented on a distributed memory system, where all solutions and gPC coefficients are distributed over multiple nodes. We saw great potential in using the ug4 library as a black box solver for quantification of uncertainties. The results are reproducible, and the code is available online on GitHub repository¹.

The number of cells in the presented experiments varied from 241,152 till 15,433,728 for the cylindrical domain and from 524,288 till 4,194,304 for parallelepiped. The maximal number of parallel processing units was 800×32 , where 800 is the number of parallel nodes, and 32 is the number of computing cores on each node. The computing time varied from 2 hours for the coarse mesh to 23 hours for the finest mesh (for one simulation).

Although this work let many questions open, it is an important step towards the investigation of the density driven flow phenomena. Potential further applications are an estimation of the contamination risks, monitoring the quality of drinking water, modeling of the seawater intrusion into coastal aquifers, radioactive waste disposal, and contaminant plumes.

As a next step, we will integrate the ug4 multi-grid solver in the Multi Level Monte Carlo framework with the idea to compute mostly samples on coarse grids and just a few on a fine grid.

Another idea is to infer the (unknown) porosity coefficient. We assume that there are some noisy measurements of c at some locations $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k\}$ are available. Then the prior probability density function of c could be improved via Bayesian update formula [50, 57, 58, 70, 71]. Herewith, the solution $c(\mathbf{x}^*)$ at a point $\mathbf{x}^*$ (or at k points or in a sub-domain) could be computed with a much smaller numerical cost and with the smaller memory requirement as the whole solution in the whole domain [45, 46].

Acknowledgements:

We would like to thank KAUST HPC support team for the assistance with Shaheen II. This work was supported by the Extreme Computing Research Center, by the SRI-UQ Strategic Initiative and by Computational Bayesian group at King Abdullah University of Science and Technology.

References

- [1] R. ASKEY AND J. WILSON, *Some basic hypergeometric orthogonal polynomials that generalize Jacobi polynomials*, Providence, R.I., U.S.A. : American Mathematical Society, 1985.
- [2] I. BABUŠKA, F. NOBILE, AND R. TEMPONE, *A stochastic collocation method for elliptic partial differential equations with random input data*, SIAM J. Numer. Anal., 45 (2007), pp. 1005–1034 (electronic).
- [3] I. BABUŠKA, R. TEMPONE, AND G. ZOURARIS, *Galerkin finite element approximations of stochastic elliptic partial differential equations*, SIAM Journal on Numerical Analysis, 42 (2004), pp. 800–825.
- [4] V. BARTHELMANN, E. NOVAK, AND K. RITTER, *High dimensional polynomial interpolation on sparse grids*, Advances in Computational Mathematics, 12 (2000), pp. 273–288.
- [5] J. BEAR, *Dynamics of fluid in Porous Media*, Dover Publications, INC. New York, 1972.
- [6] ———, *Hydraulics of Groundwater*, Dover Publications. Inc., Mineola, 1979.
- [7] J. BEAR AND Y. BACHMAT, *Introduction to Modeling of Transport Phenomena in Porous Media*, Kluwer Academic Publishers, Dordrecht, Boston, London, 1990.

¹ <https://github.com/UG4>, <https://github.com/UG4/ughub>

- [8] G. BLATMAN AND B. SUDRET, *An adaptive algorithm to build up sparse polynomial chaos expansions for stochastic finite element analysis*, Probabilistic Engineering Mechanics, 25 (2010), pp. 183 – 197.
- [9] F. BODE, T. FERRÉ, N. ZIGELLI, M. EMMERT, AND W. NOWAK, *Reconnecting stochastic methods with hydrogeological applications: A utilitarian uncertainty analysis and risk assessment approach for the design of optimal monitoring networks*, Water Resources Research, 54 (2018), pp. 2270–2287.
- [10] M. BOMPARD, J. PETER, AND J.-A. DÉSIDÉRI, *Surrogate models based on function and derivative values for aerodynamic global optimization*, in Fifth European Conference on Computational Fluid Dynamics, ECCOMAS CFD 2010, Lisbon, Portugal, 2010.
- [11] H.-J. BUNGARTZ AND M. GRIEBEL, *Sparse grids*, Acta Numer., 13 (2004), pp. 147–269.
- [12] R. E. CAFLISCH, *Monte Carlo and quasi-Monte Carlo methods*, Acta Numerica, 7 (1998), pp. 1–49.
- [13] J. CARRERA, *An overview of uncertainties in modelling groundwater solute transport*, Journal of Contaminant Hydrology, 13 (1993), pp. 23 – 48. Chemistry and Migration of Actinides and Fission Products.
- [14] A. CHKIFA, A. COHEN, AND C. SCHWAB, *Breaking the curse of dimensionality in sparse polynomial approximation of parametric PDEs*, Journal de Mathématiques Pures et Appliquées, 103 (2015), pp. 400 – 428.
- [15] C. W. CLENSHAW AND A. R. CURTIS, *A method for numerical integration on an automatic computer*, Numerische Mathematik, 2 (1960), pp. 197–205.
- [16] P. CONRAD AND Y. MARZOUK, *Adaptive smolyak pseudospectral approximations*, SIAM Journal on Scientific Computing, 35 (2013), pp. A2643–A2670.
- [17] P. CONSTANTINE, M. ELDRED, AND E. PHIPPS, *Sparse pseudospectral approximation method*, Computer Methods in Applied Mechanics and Engineering, 229–232 (2012), pp. 1 – 12.
- [18] A. COSTA, *Permeability-porosity relationship: A reexamination of the kozeny-carman equation based on a fractal pore-space geometry assumption*, Geophysical Research Letters, 33 (2006).
- [19] C. CREMER, , AND T. GRAF, *Generation of dense plume fingers in saturated–unsaturated homogeneous porous media*, Journal of Contaminant Hydrology, 173 (2015), pp. 69 – 82.
- [20] H.-J. G. DIERSCH AND O. KOLDITZ, *Variable-density flow and transport in porous media: approaches and challenges*, Advances in Water Resources, 25 (2002), pp. 899–944.
- [21] S. DOLGOV, B. KHOROMSKIJ, A. LITVINENKO, AND H. MATTHIES, *Computation of the response surface in the tensor train data format*, arXiv preprint arXiv:1406.2816, (2014).
- [22] ———, *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, 3 (2015), pp. 1109–1135.
- [23] J. ELDER, C. SIMMONS, H.-J. DIERSCH, P. FROLOVIĆ, E. HOLZBECHER, AND K. JOHANNSEN, *The Elder problem*, 2 (2017), p. 11.
- [24] J. W. ELDER, *Steady free convection in a porous medium heated from below*, Journal of Fluid Mechanics, 27 (1967), pp. 29–48.
- [25] ———, *Transient convection in a porous medium*, Journal of Fluid Mechanics, 27 (1967), p. 609–623.
- [26] O. ERNST, A. MUGLER, H.-J. STARKLOFF, AND E. ULLMANN, *On the convergence of generalized polynomial chaos expansions*, ESAIM: Mathematical Modelling and Numerical Analysis, 46 (2012), p. 317–339.
- [27] M. ESPIG, W. HACKBUSCH, A. LITVINENKO, H. MATTHIES, AND P. WAHNERT, *Efficient low-rank approximation of the stochastic Galerkin matrix in tensor formats*, Computers and Mathematics with Applications, 67 (2014), pp. 818 – 829. High-order Finite Element Approximation for Partial Differential Equations.
- [28] Y. FAN, C. DUFFY, AND D. OLIVER, *Density-driven groundwater flow in closed desert basins: field investigations and numerical experiments*, Journal of Hydrology, 196 (1997), pp. 139 – 184.
- [29] P. FROLOVIĆ, *Consistent velocity approximation for density driven flow and transport*, in Advanced Computational Methods in Engineering, Part 2: Contributed papers, R. Van Keer and et al., eds., Maastricht, 1998, Shaker Publishing, pp. 603–611.
- [30] ———, *Maximum principle and local mass balance for numerical solutions of transport equation coupled with variable density flow*, Acta Mathematica Universitatis Comenianae, 1 (1998), pp. 137–157.
- [31] P. FROLOVIĆ AND P. KNABNER, *Consistent velocity approximations in finite element or volume discretizations of density driven flow*, in Computational Methods in Water Resources XI, A. A. Aldama and et al., eds., Southampton, 1996, Computational Mechanics Publication, pp. 93–100.
- [32] T. GERSTNER AND M. GRIEBEL, *Numerical integration using sparse grids*, Numer. Algorithms, 18 (1998), pp. 209–232.
- [33] R. GHANEM, H. OWHADI, AND D. HIGDON, *Handbook of uncertainty quantification*, Springer, 06 2017.
- [34] L. GIRALDI, A. LITVINENKO, D. LIU, H. MATTHIES, AND A. NOUY, *To be or not to be intrusive? the solution of parametric and stochastic equations—the "plain vanilla" galerkin case*, SIAM Journal on Scientific Computing, 36 (2014), pp. A2720–A2744.
- [35] A. A. GIUNTA, M. S. ELDRED, AND J. P. CASTRO, *Uncertainty quantification using response surface approximation*, in 9th ASCE Specialty Conference on Probabilistic Mechanics and Structural Reliability, Albuquerque, New Mexico, USA, 2004.
- [36] M. GRIEBEL, *Sparse grids and related approximation schemes for higher dimensional problems*, in Foundations of computational mathematics, Santander 2005, vol. 331 of London Math. Soc. Lecture Note Ser., Cambridge Univ. Press, Cambridge, 2006, pp. 106–161.
- [37] W. HACKBUSCH, *Multi-Grid Methods and Applications*, Springer, Berlin, 1985.
- [38] W. HACKBUSCH, *Iterative Solution of Large Sparse Systems of Equations*, Springer, New-York, 1994.
- [39] S. JOE AND F. KUO, *Constructing sobol sequences with better two-dimensional projections*, SIAM Journal on Scientific Computing, 30 (2008), pp. 2635–2654.
- [40] K. JOHANNSEN, *On the validity of the boussinesq approximation for the Elder problem*, Computational Geosciences, 7 (2003), pp. 169–182.
- [41] N. H. KIM, H. WANG, AND N. V. QUEIPO, *Efficient shape optimization under uncertainty using polynomial chaos expansions and local sensitivities*, AIAA journal, 44 (2006), pp. 1112–1116.
- [42] A. KLIMKE, *Sparse grid interpolation toolbox*, www.ians.uni-stuttgart.de/spinterp/, (2008).

- [43] H. KOBUS, *Soil and groundwater contamination and remediation technology in europe*, in Groundwater Updates, K. Sato and Y. Iwasa, eds., Tokyo, 2000, Springer Japan, pp. 3–8.
- [44] O. LE MAÎTRE AND O. KNIO, *Introduction: Uncertainty quantification and propagation*, in Spectral Methods for Uncertainty Quantification, Springer, 2010, pp. 1–13.
- [45] A. LITVINENKO, *Application of hierarchical matrices for solving multiscale problems*, Ph.D. Thesis, University of Leipzig, Germany, (2006).
- [46] A. LITVINENKO, *Partial inversion of elliptic operator to speed up computation of likelihood in Bayesian inference*, arXiv e-prints, (2017), p. arXiv:1708.02207.
- [47] A. LITVINENKO, D. LOGASHENKO, R. TEMPONE, G. WITTUM, AND D. KEYES, *Propagation of uncertainties in density-driven flow*, arXiv preprint arXiv:1905.01770, (2019).
- [48] A. LITVINENKO AND H. MATTHIES, *Uncertainties quantification and data compression in numerical aerodynamics*, PAMM, 11 (2011), pp. 877–878.
- [49] A. LITVINENKO, H. MATTHIES, AND T. A. EL-MOSELHY, *Sampling and low-rank tensor approximation of the response surface*, in Monte Carlo and Quasi Monte Carlo Methods 2012, J. Dick, F. Y. Kuo, G. W. Peters, and I. H. Sloan, eds., vol. 65 of Springer Proceedings in Mathematics & Statistics, Springer Berlin Heidelberg, 2013, pp. 535–551.
- [50] A. LITVINENKO AND H. G. MATTHIES, *Inverse problems and uncertainty quantification*, arXiv preprint arXiv:1312.5048, (2013).
- [51] D. LIU AND S. GÖRTZ, *Efficient quantification of aerodynamic uncertainty due to random geometry perturbations*, in New Results in Numerical and Experimental Fluid Mechanics IX, A. Dillmann et al., eds., Springer International Publishing, 2014, pp. 65–73.
- [52] D. LIU, A. LITVINENKO, C. SCHILLINGS, AND V. SCHULZ, *Quantification of airfoil geometry-induced aerodynamic uncertainties—comparison of approaches*, SIAM/ASA Journal on Uncertainty Quantification, 5 (2017), pp. 334–352.
- [53] G. J. A. LOEVEN, J. A. S. WITTEVEEN, AND H. BIJL, *A probabilistic radial basis function approach for uncertainty quantification*, in Proceedings of the NATO RTO-MP-AVT-147 Computational Uncertainty in Military Vehicle design symposium, 2007.
- [54] H. MATTHIES, *Uncertainty quantification with stochastic finite elements*, in Encyclopedia of Computational Mechanics, E. Stein, R. de Borst, and T. R. J. Hughes, eds., John Wiley & Sons, Chichester, 2007.
- [55] H. MATTHIES, *Uncertainty quantification with stochastic finite elements.*, (2007). Part 1. Fundamentals. Encyclopedia of Computational Mechanics.
- [56] H. MATTHIES AND A. KEESE, *Galerkin methods for linear and nonlinear elliptic stochastic partial differential equations*, Comput. Methods Appl. Mech. Engrg., 194 (2005), pp. 1295–1331.
- [57] H. G. MATTHIES, E. ZANDER, B. V. ROSIĆ, AND A. LITVINENKO, *Parameter estimation via conditional expectation: a Bayesian inversion.*, Advanced Modeling and Simulation in Engineering Sciences, 3 (2016), p. 24.
- [58] H. G. MATTHIES, E. ZANDER, B. V. ROSIĆ, A. LITVINENKO, AND O. PAJONK, *Inverse problems in a Bayesian setting*, in Computational Methods for Solids and Fluids — Multiscale Analysis, Probability Aspects and Model Reduction, A. Ibrahimbegović, ed., vol. 41 of Computational Methods in Applied Sciences, 2016, pp. 245–286.
- [59] H. NAJM, *Uncertainty quantification and polynomial chaos techniques in computational fluid dynamics*, Annual Review of Fluid Mechanics, 41 (2009), pp. 35–52.
- [60] F. NOBILE, L. TAMELLINI, F. TESEI, AND R. TEMPONE, *An adaptive sparse grid algorithm for elliptic PDEs with log-normal diffusion coefficient*, MATHICSE Technical Report 04, 2015.
- [61] F. NOBILE, R. TEMPONE, AND C. G. WEBSTER, *A sparse grid stochastic collocation method for partial differential equations with random input data*, SIAM J. Numerical Analysis, 46 (2008), pp. 2309–2345.
- [62] E. NOVAK AND K. RITTER, *High dimensional integration of smooth functions over cubes*, Numerische Mathematik, 75 (1996), pp. 79–97.
- [63] ———, *The curse of dimension and a universal method for numerical integration*, in Multivariate approximation and splines (Mannheim, 1996), vol. 125 of Internat. Ser. Numer. Math., Birkhäuser, Basel, 1997, pp. 177–187.
- [64] ———, *Simple cubature formulas with high polynomial exactness*, Constr. Approx., 15 (1999), pp. 499–522.
- [65] K. PETRAS, *Smolpack—a software for Smolyak quadrature with delayed Clenshaw-Curtis basis-sequence*. <http://www-public.tu-bs.de:8080/petras/software.html>.
- [66] V. POST AND G. HOUBEN, *Density-driven vertical transport of saltwater through the freshwater lens on the island of baltrum (germany) following the 1962 storm flood*, Journal of Hydrology, 551 (2017), pp. 689 – 702. Investigation of Coastal Aquifers.
- [67] I. RADOVIĆ, I. SOBOL, AND R. TICHY, *Quasi-monte carlo methods for numerical integration: Comparison of different low discrepancy sequences*, Monte Carlo Methods and Applications, 2 (1996), pp. 1–14.
- [68] S. REITER, D. LOGASHENKO, A. VOGEL, AND G. WITTUM, *Mesh generation for thin layered domains and its application to parallel multigrid simulation of groundwater flow*. submitted to Comput. Visual Sci., 2017.
- [69] S. REITER, A. VOGEL, I. HEPPNER, M. RUPP, AND G. WITTUM, *A massively parallel geometric multigrid solver on hierarchically distributed grids*, Computing and Visualization in Science, 16 (2013), pp. 151–164.
- [70] B. ROSIĆ, A. KUČEROVÁ, J. SYKORA, O. PAJONK, A. LITVINENKO, AND H. MATTHIES, *Parameter identification in a probabilistic setting*, Engineering Structures, 50 (2013), pp. 179 – 196. Engineering Structures: Modelling and Computations (special issue IASS-IACM 2012).
- [71] B. V. ROSIĆ, A. LITVINENKO, O. PAJONK, AND H. G. MATTHIES, *Direct Bayesian update of polynomial chaos representations*, Informatikbericht 2011-02, 2011.
- [72] Y. RUBIN, *Applied stochastic hydrogeology*, Oxford University Press, 2003.
- [73] A. SCHNEIDER, K.-P. KRÖHN, AND A. PÜSCHEL, *Developing a modelling tool for density-driven flow in complex hydrogeological structures*, Comput. Visual Sci., 15 (2012), pp. 163–168.
- [74] M. SINSBECK AND W. NOWAK, *An optimal sampling rule for nonintrusive polynomial chaos expansions of expensive models*, International Journal for Uncertainty Quantification, 5 (2015), pp. 275–295.

- [75] S. A. SMOLYAK, *Quadrature and interpolation formulas for tensor products of certain classes of functions.*, Sov. Math. Dokl., 4 (1963), pp. 240–243.
- [76] D. TARTAKOVSKY, *Assessment and management of risk in subsurface hydrology: A review and perspective*, Advances in Water Resources, 51 (2013), pp. 247 – 260. 35th Year Anniversary Issue.
- [77] H. VEREECKEN, A. SCHNEPF, J. HOPMANS, M. JAVAUX, D. OR, T. ROOSE, J. VANDERBORGHT, M. YOUNG, W. AMELUNG, M. AITKENHEAD, S. ALLISON, S. ASSOULINE, P. BAVEYE, M. BERLI, N. BRÜGGEMANN, P. FINKE, M. FLURY, T. GAISER, G. GOVERS, T. GHEZZEHEI, P. HALLETT, H. HENDRICKS FRANSSEN, J. HEPPPELL, R. HORN, J. HUISMAN, D. JACQUES, F. JONARD, S. KOLLET, F. LAFOLIE, K. LAMORSKI, D. LEITNER, A. MCBRATNEY, B. MINASNY, C. MONTZKA, W. NOWAK, Y. PACHEPSKY, J. PADARIAN, N. ROMANO, K. ROTH, Y. ROTHFUSS, E. ROWE, A. SCHWEN, J. ŠIMŮNEK, A. TIKTAK, J. VAN DAM, S. VAN DER ZEE, H. VOGEL, J. VRUGT, T. WÖHLING, AND I. YOUNG, *Modeling soil processes: Review, key challenges, and new perspectives*, Vadose Zone Journal, 15 (2016), p. vzj2015.09.0131.
- [78] A. VOGEL, S. REITER, M. RUPP, A. NÄGEL, AND G. WITTUM, *Ug 4: A novel flexible software system for simulating pde based models on high performance computers*, Computing and Visualization in Science, 16 (2013), pp. 165–179.
- [79] C. VOSS AND W. SOUZA, *Variable density flow and solute transport simulation of regional aquifers containing a narrow freshwater-saltwater transition zone*, Water Resources Research, 23 (1987), pp. 1851–1866.
- [80] N. WIENER, *The homogeneous chaos*, American Journal of Mathematics, 60 (1938), pp. 897–936.
- [81] D. XIU, *Fast numerical methods for stochastic computations: A review*, Commun. Comput. Phys., 5, No. 2-4 (2009), pp. 242–272.
- [82] D. XIU AND G. E. KARNIADAKIS, *The Wiener-Askey polynomial chaos for stochastic differential equations*, SIAM J. Sci. Comput., 24 (2002), pp. 619–644.
- [83] D. XIU AND G. E. KARNIADAKIS, *Modeling uncertainty in flow simulations via generalized polynomial chaos*, Journal of Computational Physics, 187 (2003), pp. 137 – 167.
- [84] X. YUEQING, S. C. T., W. A. D., AND D. H.-J. G., *Prediction and uncertainty of free convection phenomena in porous media*, Water Resources Research, 48 (2012).
- [85] B. ZHAOJUN, J. DEMMEL, J. DONGARRA, A. RUHE, AND H. VAN DER VORST, *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*, SIAM, Philadelphia, 2000.

Appendices

1 Legendre polynomials

Since the uncertain input parameters have Uniform distribution, we employ multivariate Legendre polynomials as the gPC basis. Multivariate Legendre polynomials are defined on $[-1, 1]^M$, where M is the dimensionality of the stochastic space. The first few Legendre polynomials are

$$\psi_0(x) = 1, \quad \psi_1(x) = x, \quad \psi_2(x) = \frac{3}{2}x^2 - \frac{1}{2}, \quad (26)$$

$$\psi_3(x) = \frac{5}{2}x^2 - \frac{3}{2}x, \quad \psi_4(x) = \frac{35}{8}x^4 - \frac{30}{8}x^2 + \frac{3}{8} \quad (27)$$

$$\psi_5(x) = \frac{63}{8}x^5 - \frac{70}{8}x^3 + \frac{15}{8}x \quad (28)$$

The Legendre polynomials are orthogonal with respect to the L_2 norm on the interval $[-1, 1]$. The recursive formula is

$$\psi_0(x) = 1, \quad \psi_1(x) = x, \quad (n+1)\psi_{n+1}(x) = (2n+1)x\psi_n(x) - n\psi_{n-1}(x). \quad (29)$$

The scalar product is

$$\int_{-1}^1 \psi_n(x)\psi_m(x)\mu(x)dx = \frac{1}{2n+1}\delta_{nm}, \quad (30)$$

or

$$\int_{-1}^1 \psi_n(x)\psi_m(x)dx = \frac{2}{2n+1}\delta_{nm}, \quad (31)$$

where the density function $\mu(x)$ is a constant (is equal 1/2 on $[-1, 1]$), and where δ_{nm} denotes the Kronecker delta. The $L^2[-1, 1]$ norm $\|\psi_n\| = \sqrt{\frac{2}{2n+1}}$ and the weighted $L^2_\mu[-1, 1]$ norm is $\|\psi_n\| = \sqrt{\frac{1}{2n+1}}$. The gPC coefficients can be computed as follows

$$c_\alpha(x) := \frac{(c(\boldsymbol{\theta}), \Psi_\alpha(\boldsymbol{\theta}))_{L^2_\mu}}{\|\Psi_\alpha\|_{L^2_\mu}^2} = \frac{\int_{-1}^1 c\Psi_\alpha\mu(\boldsymbol{\theta})d\boldsymbol{\theta}}{\int_{-1}^1 \Psi_\alpha\Psi_\alpha\mu(\boldsymbol{\theta})d\boldsymbol{\theta}} = \frac{\mu \int_{-1}^1 c\Psi_\alpha d\boldsymbol{\theta}}{\mu \int_{-1}^1 \Psi_\alpha\Psi_\alpha d\boldsymbol{\theta}} = \frac{0.5 \cdot \sum_{i=1}^{n_q} c(\boldsymbol{\theta}_i)\Psi_\alpha(\boldsymbol{\theta}_i)w_i}{\frac{1}{2\alpha+1}} \quad (32)$$

and for $d > 1$ we have

$$c_\alpha(x) := \frac{(c(\boldsymbol{\theta}), \Psi_\alpha(\boldsymbol{\theta}))_{L^2_\mu}}{\|\Psi_\alpha\|_{L^2_\mu}^2} = \frac{\int_{[-1,1]^d} c\Psi_\alpha\mu(\boldsymbol{\theta})d\boldsymbol{\theta}}{\int_{[-1,1]^d} \Psi_\alpha\Psi_\alpha\mu(\boldsymbol{\theta})d\boldsymbol{\theta}} = \frac{\mu \int_{[-1,1]^d} c\Psi_\alpha d\boldsymbol{\theta}}{\mu \int_{[-1,1]^d} \Psi_\alpha\Psi_\alpha d\boldsymbol{\theta}} = \frac{0.5^d \cdot \sum_{i=1}^{n_q} c(\boldsymbol{\theta}_i)\Psi_\alpha(\boldsymbol{\theta}_i)w_i}{\prod_{i=1}^d \frac{1}{2\alpha_i+1}}. \quad (33)$$