

Improved Monte Carlo methods for computing failure probabilities of porous media flow systems

Fritjof Fagerlund¹, Fredrik Hellman², Axel Målqvist³, Auli Niemi¹

August 31, 2015

Abstract

We study improvements of standard and multilevel Monte Carlo methods for point evaluation of the cumulative distribution function (failure probability) applied to porous media two-phase flow simulations with uncertain permeability. In an injection scenario with sweep efficiency of the injected phase as quantity of interest, we seek the probability that this quantity of interest is smaller than a critical value. In the sampling procedure, we use computable error bounds on the sweep efficiency functional to solve only a subset of all realizations to highest accuracy by means of what we call selective refinement. We quantify the performance gains possible by using selective refinement in combination with both the standard and multilevel Monte Carlo method. We also identify issues in the process of practical implementation of the methods. We conclude that significant savings (one order of magnitude) in computational cost are possible for failure probability estimation in a realistic setting using the selective refinement technique, both in combination with standard and multilevel Monte Carlo.

1 Introduction

Engineering systems are often subject to uncertain conditions that might reduce the performance or function of the system. Monte Carlo methods for quantification of the failure probability of porous media flow systems by numerical simulation is the topic of this report. We focus entirely on an application to two-phase flow where the permeability field is uncertain and modeled as a random field. Given the uncertainty in the permeability, we seek the probability that sweep efficiency is less than a given critical value. In other words, the failure probability is the value of the cumulative distribution function (cdf) for the sweep efficiency functional at the critical value. In this work, we quantify how error bounds on this functional can be used to improve the performance of standard and multilevel Monte Carlo methods. This is a continuation of the works [11, 12] where the selective refinement technique was introduced and asymptotic cost rate results for (multilevel) Monte Carlo methods using error bounds were derived for the point evaluation of a cdf. Other approaches for point evaluation of the cdf in a multilevel Monte Carlo context can

¹Department of Earth Sciences, Uppsala University, Box 337, SE-751 05 Uppsala, Sweden. Supported by EU FP7 project TRUST.

²Department of Information Technology, Uppsala University, Box 337, SE-751 05 Uppsala, Sweden. Supported by the Centre for Interdisciplinary Mathematics, Uppsala University.

³Department of Mathematical Sciences, Chalmers University of Technology and University of Gothenburg, SE-412 96 Göteborg, Sweden. Supported by the Swedish Research Council.

be found in [1, 16]. The considered methods (in total four different Monte Carlo methods) are non-intrusive, i.e. we pick realizations from a distribution of input data for which a partial differential equation (PDE) is solved by numerical simulation and the value of the quantity of interest is computed. Each realization is considered either a success or a failure based on the value of this functional in the numerical solution. It can be solved on meshes of varying resolution depending on desired accuracy. The four Monte Carlo setups are briefly discussed below.

For a sample of independent realizations, the sequence of failures and successes can be used to compute an estimate of the failure probability in a Monte Carlo (MC) estimator by computing the mean of the sequence where failure is counted as 1 and success as 0. An improvement of the MC estimator for application on numerical simulations of controllable numerical accuracy is the multilevel Monte Carlo (MLMC) estimator [19] first introduced in the context of differential equations in [15]. It has since then been applied to elliptic PDEs in [2, 6, 21, 23], to density estimation in [1, 16] and has been further analyzed and extended in [7, 17, 18]. It exploits the convergence of numerical solutions with respect to some discretization parameter h (typically mesh size) and uses a series of corrector estimators of increasing cost but decreasing variance to allow for redistribution of the variance reduction effort to cheap low-accuracy problems. Another (independent) improvement is Monte Carlo with selective refinement (MC-SR, [11]) for estimation of p -quantiles or point evaluation of a cdf. Selective refinement uses error bounds of the quantity of interest to determine which realizations need to be solved on a fine mesh, and which realizations can be solved on a coarser mesh to a smaller cost. Selective refinement can be combined with multilevel Monte Carlo (MLMC-SR, [12]). The computational cost of the four setups (MC, MC-SR, MLMC and MLMC-SR) are estimated for the sweep efficiency in a two-phase flow scenario where the failure probability is of magnitude 5–10% and an absolute accuracy of this probability in the order of a few percent is required. This work is to a large extent experimental and also aims at identifying problems and difficulties in the practical implementation of the mentioned improved Monte Carlo techniques. In particular, the two issues of estimating the variance of the correctors for the multilevel Monte Carlo method, and the establishment of an error bound required for selective refinement are addressed in this work.

The report is structured as follows. The problem setting and the continuous two-phase flow model is described in Section 2. In Section 3 we introduce a mesh hierarchy, the space and time discretization used, and the procedure for generating random permeability fields. Section 4 gives an overview of the four Monte Carlo setups and presents the numerical experiments, their results and a discussion. The conclusion is found in Section 5.

2 Continuous model

The problem is to estimate the failure probability $p = F(y) = \Pr(X \leq y)$, where y is a given value, called critical value, and F is the cdf of the sweep efficiency X . The random sweep efficiency is modeled as a functional of the solution to a nonlinear PDE with random inputs modeling two-phase flow. This section describes the continuous model for the two-phase flow system, introducing the PDE in Section 2.1, random input in Section 2.2 and sweep efficiency in Section 2.3.

2.1 Fractional-flow formulation for two-phase flow

We use the fractional flow equations as model for the two-phase flow in porous media and assume isotropic permeability, immiscibility, incompressibility, no capillary forces and that the flow is perpendicular to the gravitational field. Let the domain be the two-dimensional unit square and denoted by $\mathcal{D} = [0, 1]^2$ and its boundary by Γ . We denote an arbitrary phase by α and the two particular phases by $\alpha = w$ and $\alpha = n$ for the wetting and non-wetting phase, respectively. For each phase, we have a mass conservation equation

$$\rho_\alpha \phi \frac{\partial s_\alpha}{\partial t} + \rho_\alpha \nabla \cdot \mathbf{u}_\alpha = \nu_\alpha, \quad \alpha = w, n, \quad (1)$$

in \mathcal{D} , where ρ_α is density, ϕ is porosity, s_α is saturation, \mathbf{u}_α is volumetric flux and ν_α is a source term. The pore space is occupied only by the two fluids, i.e. $s_w + s_n = 1$. For convenience, we denote the wetting phase saturation by $s = s_w = 1 - s_n$ and refer to it simply by saturation. The flux is coupled with pressure and saturation via the relative permeabilities in Darcy's law,

$$\mathbf{u}_\alpha = -\frac{k_{r,\alpha}(s)K}{\mu_\alpha} \nabla p, \quad \alpha = w, n. \quad (2)$$

Here, K is the isotropic permeability field, $k_{r,\alpha}$ is the relative permeability, p is pressure (assumed equal for both phases), and μ_α is the dynamic viscosity. For the relative permeability, we use

$$k_{r,w} = (s_e)^3 \quad \text{and} \quad k_{r,n} = \zeta(1 - s_e)^3 \quad (3)$$

for the wetting and non-wetting phase, respectively, where s_e is the effective wetting fluid saturation $s_e = (s_w - s_{r,w})(1 - s_{r,w} - s_{r,n})^{-1}$. Here, $s_{r,\alpha}$ is the residual saturation for the two phases and ζ is a parameter, which we set to 1.

We now present the fractional flow formulation. We denote the total fluid flux by $\mathbf{u} = \mathbf{u}_w + \mathbf{u}_n$ and the phase mobilities for the two phases by

$$\lambda_\alpha(s) = \frac{k_{r,\alpha}(s)}{\mu_\alpha}, \quad \alpha = w, n. \quad (4)$$

The total mobility is defined as $\lambda(s) = \lambda_w(s) + \lambda_n(s)$ and the fractional flow function as $f(s) = \lambda(s)^{-1} \lambda_w(s)$. The wetting phase fluxes can be expressed in terms of total flux using the fractional flow function $\mathbf{u}_w = f(s) \mathbf{u}$. Summing the Darcy equations (2) and mass conservation equations (1) (using that $\frac{\partial}{\partial t}(s_w + s_n) = 0$) yields the pressure equation:

$$\begin{aligned} \mathbf{u} + \lambda(s)K \nabla p &= 0, \\ \nabla \cdot \mathbf{u} &= \rho_w^{-1} \nu_w + \rho_n^{-1} \nu_n. \end{aligned} \quad (5)$$

Finally, we use (1) for $\alpha = w$, and obtain the saturation equation:

$$\phi \frac{\partial s}{\partial t} + \nabla \cdot (f(s) \mathbf{u}) = \rho_w^{-1} \nu_w. \quad (6)$$

The pressure and saturation equations form a non-linear system of equations in the unknowns \mathbf{u} , p and s .

The pore space is initially filled with the wetting phase, i.e. we have at $t = 0$,

$$s = 1 \quad \text{in } \mathcal{D}.$$

The pressure and flux depend only on s and need not be assigned initial values. Regarding boundary conditions, we let the upper and lower boundary segments $\Gamma_N \subset \Gamma$ of the square be impermeable; the left boundary $\Gamma_L \subset \Gamma$ be assigned high pressure and zero saturation; and the right boundary $\Gamma_R \subset \Gamma$ be assigned low pressure. The pressure gradient makes it necessary only to pose boundary conditions for the saturation on the left boundary, since inward flux will be present only along the left boundary. More precisely, we have for $t \geq 0$,

$$\begin{aligned} \mathbf{u} &= 0 && \text{on } \Gamma_N, \\ p = 1, s &= 0 && \text{on } \Gamma_L, \\ p &= 0 && \text{on } \Gamma_R. \end{aligned} \tag{7}$$

The flow is driven by the boundary conditions exclusively, and we let the source functions be zero, $\nu_\alpha = 0$.

2.2 Lognormal permeability field with exponential covariance

The permeability field K is considered random input data. It is common in the subsurface hydrology literature to model the random permeability fields as lognormal with exponential covariance, possibly at multiple correlation scales (see e.g. [13]). We use this model, but with a single correlation scale of a tenth of the size of the computational domain. More precisely, we let

$$K(x) = \exp(\kappa(x)),$$

over \mathcal{D} , where $\kappa(x)$ has zero mean and is normal distributed with exponential covariance, i.e. for all $x_1, x_2 \in \mathcal{D}$ we have that

$$\text{Var} [\kappa(x_1)\kappa(x_2)] = \sigma^2 \exp\left(\frac{-\|x_1 - x_2\|_2}{\rho}\right) =: C(\|x_1 - x_2\|_2). \tag{8}$$

For this stationary field, the covariance function $C(r)$ in (8) depends only on the distance r between two points. Two realizations of this field are shown in Figure 1.

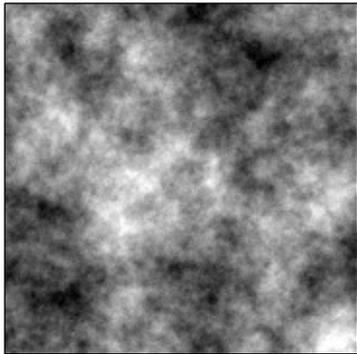
2.3 Sweep efficiency

Our quantity of interest is sweep efficiency. Sweep efficiency is the proportion of the domain covered by the non-wetting fluid at some time after injection, or the proportion of the domain covered at steady-state. Here, since we neglect capillary forces and steady-state is full coverage of the non-wetting fluid, we consider the swept proportion after a fixed time T . The functional is expressed as

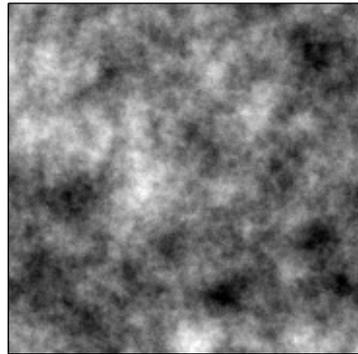
$$X = X(s) = \int_{\mathcal{D}} \chi_{(0,1]}(1 - s(T)) \, dx,$$

where χ_A is the indicator function

$$\chi_A(x) = \begin{cases} 1 & \text{if } x \in A, \\ 0 & \text{otherwise.} \end{cases}$$



(a) Realization 1.



(b) Realization 2.

Figure 1: The 10-logarithm of two realizations of a lognormally distributed random field with exponential covariance. The colormap spans from black to white the range $[-4, 4]$. The parameter values are $\rho = 0.1$ and $\sigma = 3$.

Table 1: Parameter values for the continuous model.

Parameter name (and symbol)	Value
Porosity (ϕ)	1
Relative permeability function parameter (ζ)	1
Residual saturations ($s_{r,w}$ and $s_{r,n}$)	0
Dynamic viscosity (μ_w and μ_n)	1
Source terms (ν_w and ν_n)	0
Standard deviation (σ)	3
Correlation length (ρ)	0.1
Stop time (T)	0.2

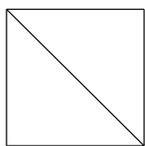
We conclude this section with a table (Table 1) of all parameter values used in this work for the continuous model. Note that the two phases have identical properties.

3 Discretization

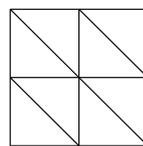
This section is a description of the discretizations used to solve the continuous model numerically. A hierarchy of meshes is introduced in Section 3.1. This is necessary for the multilevel Monte Carlo method and selective refinement, which both require a hierarchy of solutions in a convergent regime to be efficient. A sequential splitting scheme is presented in Section 3.2 describing the simulation procedure, once grid and data are given. Section 3.3 describes how the circulant embedding technique is used to generate permeability data, and how such a field is truncated to the meshes on the lower levels in the mesh hierarchy. An approximation of the sweep efficiency functional is presented in Section 3.4.

3.1 Mesh hierarchy

The domain \mathcal{D} is meshed with a family of uniform and conforming triangulations \mathcal{T}_h depicted in Figure 2. Here h is the vertical and horizontal vertex spacing. The coordinates of the vertices are (ih, jh) for $i, j = 0, \dots, h^{-1}$ and every square in the grid is split into two triangles by connecting the upper-left and lower-right corners. We introduce a mesh hierarchy consisting of all meshes \mathcal{T}_h with $h = 2^{-(L_0+\ell)}$ for hierarchy levels $\ell = 0, 1, \dots, L$. Note that from one level ℓ to the next $\ell + 1$, new vertices are added to the grid, but none are removed. The values of L_0 (coarsest mesh size) and L (the number of levels) are to be determined below. In general, the lower bound L_0 is determined by the coarsest mesh that is part of a convergent regime for the quantity of interest, while the number of levels L depends on the limitations of computational resources and desired accuracy.



(a) The mesh with $h = 1$.



(b) The mesh with $h = 1/2$.

Figure 2: Two members of \mathcal{T}_h .

We choose $L_0 = 4$, which means the coarsest mesh has a mesh size of $h = 1/16$. This is slightly smaller than the correlation length $\sigma = 0.1$ of K . In, for example [4], it has been indicated that a finite element discretization of the pressure equation does not converge in regimes where the correlation length is not resolved by the mesh for a standard finite element method, why we require $h < \sigma$. The number of levels L is chosen to 4, which means the finest mesh size is $h = 1/256$. This was determined based on our available computational resources.

3.2 Sequential splitting

We use a sequential splitting scheme to split the pressure and saturation equation similarly to the improved Implicit Pressure Explicit Saturation (improved IMPES) scheme presented in [5]. This split renders the pressure equation a linear elliptic equation, and the saturation equation a nonlinear hyperbolic transport equation with fixed total flux field. The pressure equation (5) is solved implicitly for a given time step, keeping saturation fixed from the previous time step. The flux solution from the pressure equation is used in the saturation equation (6), from which the saturation for the next time step is computed using an explicit method.

Let $0 = t^0 < t^1 < \dots < t^M = T$ be the M outer time steps for which a pressure iteration is performed (additional inner time steps will be needed for solving the saturation equation). We use a constant outer time step length $\tau = t^{m+1} - t^m$. We denote by \mathbf{u}^m , p^m and s^m the solutions at outer time step t^m , and state the semi-discrete and mixed form of the pressure equation:

$$\begin{aligned} (\lambda(s^m)K)^{-1}\mathbf{u}^{m+1} + \nabla p^{m+1} &= 0, \\ \nabla \cdot \mathbf{u}^{m+1} &= 0, \end{aligned} \tag{9}$$

with the boundary conditions given in (7).

The saturation equation is discretized using the explicit forward Euler method on a finer grid in time to ensure stability. We subdivide each interval $[t^m, t^{m+1}]$ into K_m inner time steps $t^m = t^{m,0} < t^{m,1} < \dots < t^{m,K_m} = t^{m+1}$, with an inner time step length $\tau_m = t^{m,k+1} - t^{m,k}$. Note that $K_m = \tau/\tau_m$. The saturation equation is then discretized in time using forward Euler:

$$\phi s^{m,k+1} = \phi s^{m,k} - \tau_m \nabla \cdot \mathbf{u}_w^{m,k}, \quad k = 0, \dots, K_m - 1. \quad (10)$$

where

$$\mathbf{u}_w^{m,k} = f(s^{m,k}) \mathbf{u}^{m+1}. \quad (11)$$

Equations (9) and (10) are solved in sequence, so that \mathbf{u}^{m+1} is available as data for the saturation equation, and $s^{m,K_m} = s^{m+1}$ is available for the pressure equation. The number of inner time steps K_m is chosen adaptively for each outer time step to match the CFL condition for the full discretization (more details can be found below in this section).

For the spatial discretization of the pressure equation, we use the zeroth order Raviart–Thomas finite elements that yield a conservative flux field \mathbf{u}_h^{m+1} . For the saturation equation, we use a donor cell upwind finite volume scheme (see e.g. [20]) on a triangular mesh and the saturation is approximated by piecewise constants $s_h^{m,k}$.

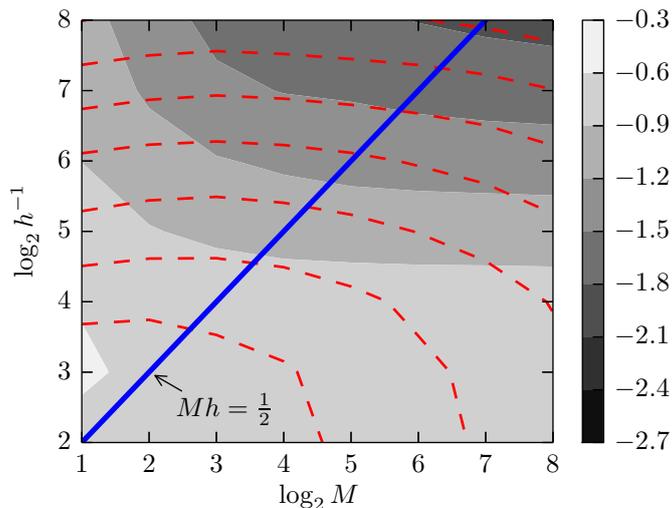


Figure 3: The filled (gray) contour plot shows the 10-logarithm of the estimated relative error. The dashed (red) contour plot shows computational cost iso-lines. The cost increases as h^{-1} and M increases, while the error decreases. The solid (blue) line is selected by hand and is one possible and efficient relation between M and h .

While the description of the discretization above is done for meshes of any size h and any number of time steps M , we use special notation for solutions obtained using the hierarchical meshes introduced in Section 3.1 obeying the relation $Mh = 0.5$. (This relation was determined in an experiment where errors from the spatial and temporal discretizations were balanced to minimize the computational cost, see Figure 3). We define the approximate saturation solution at time T at level ℓ as

$$s_\ell = s_h^M,$$

where $h = 2^{-(L_0+\ell)}$ and $Mh = 0.5$. The remainder of this section is a detailed description of the discretization schemes used.

The set of interior edges in the mesh \mathcal{T}_h are denoted by \mathcal{F}_h^i , i.e. $F \in \mathcal{F}_h^i$ is an edge in the triangulation, but $F \cap \partial\mathcal{D} = \emptyset$. The edge normals are denoted by \mathbf{n} . The edges on the domain boundaries have outgoing normals. The L^2 -scalar product on \mathcal{D} is defined by $(u, v) = \int_{\mathcal{D}} uv \, dx$ for scalar-valued functions u and v , and $(\mathbf{u}, \mathbf{v}) = \int_{\mathcal{D}} \mathbf{u} \cdot \mathbf{v} \, dx$ for vector-valued functions \mathbf{u} and \mathbf{v} . Let $\mathcal{RT}_0(\mathcal{T}_h)$ denote the zeroth order vector-valued Raviart–Thomas finite element space [22] on \mathcal{T}_h and let $\mathcal{P}_0(\mathcal{T}_h)$ denote the space of piecewise constants on \mathcal{T}_h . We then look for a flux solution in $\mathcal{RT}_0(\mathcal{T}_h)$ and for a pressure solution in $\mathcal{P}_0(\mathcal{T}_h)$ (see e.g. [3]). Multiplying (9) with test functions \mathbf{v}_h and q_h from these spaces and integrating over the domain yields the following fully discrete mixed system in \mathbf{u}_h^{m+1} and p_h^{m+1} ,

$$\begin{aligned} ((\lambda(s^m)K)^{-1}\mathbf{u}_h^{m+1}, \mathbf{v}_h) - (p_h^{m+1}, \nabla \cdot \mathbf{v}_h) &= 0 \quad \text{for all } \mathbf{v}_h \in \mathcal{RT}_0(\mathcal{T}_h), \\ (\nabla \cdot \mathbf{u}_h^{m+1}, q_h) &= 0 \quad \text{for all } q_h \in \mathcal{P}_0(\mathcal{T}_h). \end{aligned} \quad (12)$$

The saturation equation (10) is discretized using a donor cell upwind finite volume method or lowest order discontinuous Galerkin method (see e.g. [9]). We seek a saturation solution $s_h^{m,k+1} \in \mathcal{P}_0(\mathcal{T}_h)$. We multiply (10) by a test function r_h and integrate, and do integration by parts for every triangle on the rightmost term to get the following equation in $s_h^{m,k+1}$, for all $k = 0, \dots, K_m - 1$,

$$(\phi s_h^{m,k+1}, r_h) = (\phi s_h^{m,k}, r_h) - \tau \sum_{U \in \mathcal{T}_h} (\mathbf{u}_{h,w}^{m,k} \cdot \mathbf{n}, r_h)_{\partial U} \quad \text{for all } r_h \in \mathcal{P}_0(\mathcal{T}_h). \quad (13)$$

Since $s_h^{m,k}$ is only piecewise constant, $\mathbf{u}_{h,w}^{m,k}(s_h^{m,k}) = f(s_h^{m,k})\mathbf{u}_h^{m+1}$ is undefined on the triangle edges. We define the following upwind numerical flux operator A_h^{upw} :

$$\begin{aligned} (A_h^{\text{upw}} s, r) &= \int_{\partial\Omega} (\mathbf{u}^{m+1} \cdot \mathbf{n})^\ominus sr \, d\gamma - \sum_{F \in \mathcal{F}_h^i} \int_F (\mathbf{u}^{m+1} \cdot \mathbf{n}) \llbracket s \rrbracket \langle r \rangle \, d\gamma \\ &\quad + \sum_{F \in \mathcal{F}_h^i} \int_F \frac{1}{2} |\mathbf{u}^{m+1} \cdot \mathbf{n}| \llbracket s \rrbracket \llbracket r \rrbracket \, d\gamma. \end{aligned}$$

The symbol $(\cdot)^\ominus$ indicates the negative part, i.e. $(a)^\ominus = \frac{1}{2}(|a| - a)$. Further, the symbols $\llbracket \cdot \rrbracket$ and $\langle \cdot \rangle$ are defined on interior edges and denote the jump and average operators respectively. For every edge F , let the normal \mathbf{n} be directed towards triangle U_2 from triangle U_1 . Then we define, for any $r \in \mathcal{P}_0(\mathcal{T}_h)$,

$$\llbracket r \rrbracket = r|_{U_1} - r|_{U_2} \quad \text{and} \quad \langle r \rangle = \frac{r|_{U_1} + r|_{U_2}}{2}.$$

The final form of the discrete saturation equation is

$$(\phi s_h^{m,k+1}, r_h) = (\phi s_h^{m,k}, r_h) - \tau (A_h^{\text{upw}} f(s_h^{m,k}), r_h)$$

for all $r_h \in \mathcal{P}_0(\mathcal{T}_h)$ and $k = 0, \dots, K_m - 1$.

The number of inner time steps K_m is determined by ensuring that the following CFL condition holds, for all $U \in \mathcal{T}_h$,

$$\frac{\tau_m}{|U|} \int_{\partial U} (\mathbf{u}^{m+1} \cdot \mathbf{n}_U)^\ominus L_f \, d\gamma \leq 1. \quad (14)$$

Here $|U|$ is the area of triangle U , \mathbf{n}_U is the outgoing edge normal of ∂U , and $L_f = 3$ is the Lipschitz constant for $f(s)$ for $0 \leq s \leq 1$. Since $K_m = \tau/\tau_m$, a minimum value of K_m can be determined from equation (14). The value of K_m is determined after the pressure equation has been solved. This ensures that the value of $s_h^{m,k+1}$ in triangle U is a convex combination of the values of $s_h^{m,k}$ in the adjacent triangles, U itself and the boundary conditions. A maximum principle and hence stability immediately follows.

3.3 Circulant embedding and spectral truncation

The multilevel Monte Carlo method requires gradual refinement of the permeability field based on the gradual refinement of the mesh in the mesh hierarchy. In this section, we briefly describe how the permeability data is generated and how it is truncated for different levels ℓ in the mesh hierarchy to avoid aliasing effects.

The random field K is realized using the circulant embedding technique [10]. This is an efficient way to realize a stationary random field on a cartesian grid with equispaced points. The computational complexity is $\mathcal{O}(n^2 \log n)$ if n^2 is the number of vertices in the grid. A detailed description of circulant embedding can be found in Appendix A. Here we proceed with a brief description of the truncation of the fields to coarser meshes.

One way to do the truncation is to realize the field on the finest mesh (level L) and then do linear pointwise interpolation for all coarser meshes (levels $\ell < L$). This approach has the disadvantage that aliasing effects become apparent for the coarsest meshes (see e.g. [14]), i.e. non-negligible variance of the omitted high frequencies are folded onto the low frequencies and the low frequency modes get too large variance.

Instead, we perform the truncation in the spectral domain. We use that the circulant embedding permits a Fourier diagonalization of an extended covariance matrix. The field is realized by generating a random eigenvalue-weighted linear combination of the Fourier modes. If to generate a realization on a mesh with n^2 grid points, we include only the n^2 lowest (ℓ^∞ mixed) Fourier modes in the linear combination. This way, the aliasing effect is avoided. Also, this allows for gradual refinement of a mesh by gradually increasing the number of included Fourier modes. We refer to Appendix A and [10] for details.

3.4 Approximate sweep efficiency

The functional X is not suitable for direct application to the discrete approximation s_h^M of $s(T)$ for two reasons: First, $s(T)$ is equal to 1 in large parts of the domain, and so should a good approximation s_h^M . However, the functional is very sensitive for perturbations of saturation values close to 1 due to the step at that point. The approximation s_h^M might deviate from 1 due to rounding or discretization errors and cause large errors in the functional value. Secondly, even if the plume front is sharp in the continuous solution $s(T)$, it is smoothed out in s_h^M . A smoothed out front can cause the sweep efficiency to be

Table 2: Parameter values for the discretization.

Parameter name (and symbol)	Value
Coarsest mesh (L_0)	4
Deepest level (L)	4
Lipschitz constant for f (L_f)	3
Sweep efficiency threshold (c)	0.5

overestimated. Instead, motivated by the appearance of a Buckley–Leverett plume front (see Figure 4), we define the following functional with a threshold $0 < c \leq 1$,

$$X_h^M = \int_{\mathcal{D}} \chi_{(c,1]}(1 - s_h^M) dx.$$

Further, we define

$$X_\ell = X_h^M,$$

when $h = 2^{-(L_0+\ell)}$ and $Mh = 0.5$. Figure 4 shows the wetting saturation in a continuous solution and an upwind discretized solution to the one dimensional Buckley–Leverett equation in a drainage scenario with relative permeability functions given in (3). The figure suggests that a value of $c = 1$ would overestimate the sweep efficiency. Guided by the figure, we choose $c = 0.5$.

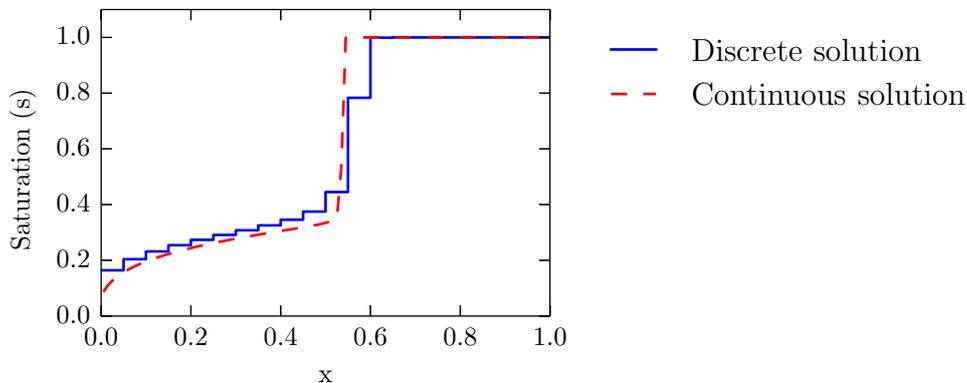


Figure 4: Wetting saturation in continuous and discrete (upwind) solution to Buckley–Leverett solution under drainage.

We conclude this section with a table (Table 2) of all parameter values used for the discretization.

4 Monte Carlo methods for estimating failure probability

We are interested in computing the probability p for the sweep efficiency X to be less than a critical value y , i.e. $p = \Pr(X \leq y)$. We define the failure probability functional

$$Q = \chi_{(-\infty,y]}(X)$$

which is equal to 1 if $X \leq y$, and equal to 0 otherwise. The basis for the Monte Carlo methods in this work is that the failure probability p can be written $p = \mathbb{E}[Q]$, where $\mathbb{E}[\cdot]$ denotes expected value. Similarly to the numerical approximations X_ℓ at level ℓ of X using the numerical scheme presented in Section 3 we define $Q_\ell = \chi_{(-\infty, y]}(X_\ell)$ as the approximate failure probability functional at level ℓ . All estimators \widehat{Q} will be based on discretizations on levels no deeper than L . We use the mean squared error (MSE) (and root mean squared error, RMSE) as a measure of error:

$$e[\widehat{Q}]^2 = \mathbb{E}\left[\left(\widehat{Q} - \mathbb{E}[Q]\right)^2\right] = \text{Var}\left[\widehat{Q}\right] + \left(\mathbb{E}\left[\widehat{Q} - Q\right]\right)^2.$$

The variance $\text{Var}\left[\widehat{Q}\right]$ of the estimator will be referred to as statistical error and the bias $\mathbb{E}\left[\widehat{Q} - Q\right]$ as numerical bias. Note that this latter bias includes error from truncation of the permeability field as well as discretization of space and time.

We review four different Monte Carlo simulation setups. We use either standard Monte Carlo or multilevel Monte Carlo. For each of them, we either use or do not use selective refinement, rendering four possibilities. The asymptotic computational cost rates for (multilevel) Monte Carlo methods in terms of total RMSE tolerance $e[\widehat{Q}] \leq \text{TOL}$ for the failure probability functional with and without selective refinement have been studied in [11, 12], based on the analysis in [15]. Convergence rates for the strong error and expected cost to compute a realization are assumed to satisfy the following assumption.

Assumption 1. *Assume that for some $0 < \gamma < 1$ and $C > 0$ independent of ℓ , it holds*

A1 $\mathbb{E}[|Q_\ell - Q|] \leq C\gamma^\ell,$

A2 $\mathcal{C}[Q_\ell] = \gamma^{-q\ell},$

where $\mathcal{C}[Q_\ell]$ denotes the expected cost to compute a realization of Q_ℓ . Assume additionally that the cdf F is Lipschitz continuous.

A summary of the asymptotic computational cost rates as $\text{TOL} \rightarrow 0$ for each setup is given in Table 3, in a scenario where the two sources of error (statistical error and numerical bias) are balanced, i.e. $\text{Var}\left[\widehat{Q}\right] = \left(\mathbb{E}\left[\widehat{Q} - Q\right]\right)^2 = \frac{1}{2}\text{TOL}^2$ (so that $e[\widehat{Q}] = \text{TOL}$), and the cost for computing a better approximation Q_ℓ increases more than twice as fast as the strong error $\mathbb{E}[|Q_\ell - Q|]$ decreases, i.e. $q > 2$. (It is shown below that this is indeed the case for our application). It is clear from the table that the combination of multilevel Monte Carlo and selective refinement has the lowest asymptotic cost. In fact, the asymptotic cost in this case is proportional to the cost for a single simulation at the deepest level.

In this section, we quantify the time savings possible in a regime where $\text{TOL} \approx 0.01$ and $p \approx 0.07$ for the four setups. The critical value is chosen as $y = 0.08$, i.e. the failure event reads: “the sweep efficiency is less than 8%”.

Table 3: Asymptotic cost rates for the failure probability estimators as $\text{TOL} \rightarrow 0$ for different Monte Carlo method setups when $q > 2$.

	No selective refinement	Selective refinement
Monte Carlo	TOL^{-2-q}	TOL^{-1-q}
Multilevel Monte Carlo	TOL^{-1-q}	TOL^{-q}

4.1 Standard Monte Carlo method

The standard Monte Carlo (MC) estimator $\widehat{Q}_{N,L}^{\text{MC}}$ estimates $\mathbb{E}[Q_L]$ by computing the mean of an i.i.d. N -sample of Q_L ,

$$\widehat{Q}_{N,L}^{\text{MC}} = \frac{1}{N} \sum_{i=0}^N Q_L^i. \quad (15)$$

Each Q_L^i is computed using the procedure described in Section 3. An MC estimator is unbiased $\mathbb{E}[\widehat{Q}_{N,L}^{\text{MC}}] = \mathbb{E}[Q_L]$, so the statistical error and numerical bias are

$$\text{Var}[\widehat{Q}_{N,L}^{\text{MC}}] = \frac{\text{Var}[Q_L]}{N} \quad \text{and} \quad \mathbb{E}[\widehat{Q}_{N,L}^{\text{MC}} - Q] = \mathbb{E}[Q_L - Q],$$

respectively. Note that we already fixed the level L , while N is to be chosen. The asymptotic cost rate in terms of TOL for balanced statistical and numerical error is TOL^{-2-q} for the MC method.

4.2 Multilevel Monte Carlo method

The multilevel Monte Carlo [15] estimator $\widehat{Q}_{\{N_\ell\},L}^{\text{ML}}$ estimates $\mathbb{E}[Q_L]$ by expanding it in a telescoping sum,

$$\mathbb{E}[Q_L] = \mathbb{E}[Q_0] + \sum_{\ell=1}^L \mathbb{E}[Q_\ell - Q_{\ell-1}],$$

and using standard Monte Carlo estimators for the expected values. We introduce correctors $Y_\ell = Q_\ell - Q_{\ell-1}$, and the MLMC estimator reads

$$\widehat{Q}_{\{N_\ell\},L}^{\text{ML}} = \frac{1}{N_0} \sum_{i=1}^{N_0} Q_0^i + \sum_{\ell=1}^L \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} Y_\ell^i, \quad (16)$$

where N_ℓ are the sample sizes for the MC estimators and Q_0^i and Y_ℓ^i are realizations of the lowest level and correctors, respectively. A corrector realization Y_ℓ^i is computed by generating one realization i of the random input data and compute both Q_ℓ^i and $Q_{\ell-1}^i$ using that one realization. Due to the telescoping sum, the MLMC estimator is again an unbiased estimator of Q_L . The variance, however, depends on all N_ℓ . The statistical error and numerical bias are,

$$\text{Var}[\widehat{Q}_{\{N_\ell\},L}^{\text{ML}}] = \frac{\text{Var}[Q_0]}{N_0} + \sum_{\ell=1}^L \frac{\text{Var}[Y_\ell]}{N_\ell} \quad \text{and} \quad \mathbb{E}[\widehat{Q}_{\{N_\ell\},L}^{\text{ML}} - Q] = \mathbb{E}[Q_L - Q],$$

respectively. The number of samples N_ℓ on each level is determined by minimizing the expected cost $\mathcal{C} \left[\widehat{Q}_{\{N_\ell\},L}^{\text{ML}} \right]$ for the estimator constraining the variance, $\text{Var} \left[\widehat{Q}_{\{N_\ell\},L}^{\text{ML}} \right] = \frac{1}{2} \text{TOL}^2$, yielding

$$N_0 = 2C_L \text{TOL}^{-2} \sqrt{\text{Var} [Q_0] / \mathcal{C} [Q_0]} \quad \text{and} \quad N_\ell = 2C_L \text{TOL}^{-2} \sqrt{\text{Var} [Y_\ell] / \mathcal{C} [Y_\ell]}, \quad (17)$$

where $C_L = \sqrt{\text{Var} [Q_0] \mathcal{C} [Q_0] + \sum_{\ell=1}^L \sqrt{\text{Var} [Y_\ell] \mathcal{C} [Y_\ell]}}$. See [15] for details. The variances of Q_0 and Y_ℓ and the expected cost need to be estimated to determine N_ℓ . The asymptotic cost rate in terms of TOL for balanced statistical and numerical error is TOL^{-1-q} (if $q > 1$) for the MC method for the failure probability functional [12].

An additional comment on MLMC for failure probability is that there are difficulties estimating the variance $\text{Var} [Y_\ell]$ for deep levels ℓ , due to the discrete distribution of Y_ℓ . The sample size N_L for the deepest level L in the MLMC estimator is typically too small for estimating $\text{Var} [Y_L]$ reliably using sample variance. (See [7, 12] for elaborate discussions about this). The approach used here is to estimate the variance for the lower levels and extrapolate it to the deeper levels.

4.3 Selective refinement

Selective refinement uses a posteriori error bounds to reduce the expected cost to compute a realization of Q_ℓ . Suppose we are provided with a level j -specific error bound E_j such that

$$|X_\ell - X_j| \leq E_j, \quad (18)$$

for all $\ell > j$. Then if $|y - X_j| > E_j$, the approximate failure probability Q_j is equal to Q_ℓ ,

$$Q_j = \chi_{(-\infty, y]}(X_j) = \chi_{(-\infty, y]}(X_\ell) = Q_\ell,$$

since $|y - X_j| > |X_\ell - X_j|$. The selective refinement idea is to approximate Q_ℓ by $Q_\ell^S := Q_j$ for the smallest $j \geq 0$ such that $|y - X_j| > E_j$ or $j = \ell$. An algorithm for computing Q_ℓ^S is presented in Algorithm 1. Using this algorithm (under Assumption 1) makes the

Algorithm 1 Selective refinement algorithm

- 1: Input arguments: level ℓ , realization i , critical value y
 - 2: Compute X_0^i and E_0^i
 - 3: Let $j = 0$
 - 4: **while** $j < \ell$ and $|y - X_j^i| \leq E_j^i$ **do**
 - 5: Let $j = j + 1$
 - 6: Compute X_j^i and E_j^i
 - 7: **end while**
 - 8: Let $Q_\ell^{S,i} = \chi_{(-\infty, y]}(X_j^i)$.
-

asymptotic cost rate $\mathcal{C} [Q_\ell^S] \leq \tilde{C} \gamma^{-(q-1)\ell}$ for some constant \tilde{C} independent of ℓ (depending however, on the distribution F). Compared with the rate for $\mathcal{C} [Q_\ell]$ in Assumption 1, the cost rate is decreased by 1 (see [12]). Thus, using selective refinement in combination with MC and MLMC, yields the rates TOL^{-1-q} and TOL^{-q} , respectively. See Table 3 for a summary of the asymptotic cost rates.

It is often difficult to provide a guaranteed error bound E_j . However, as will be seen Section 4.3.1, in our case it is possible to give a probabilistic bound that holds with probability at least $1 - \alpha$, i.e. for any $\ell > j$

$$\Pr(|X_\ell - X_j| \leq E_j) \geq 1 - \alpha. \quad (19)$$

Denote the event that the bound is broken for some $j < \ell$ by $A_\ell = \bigcup_{0 \leq j < \ell} \{|X_\ell - X_j| > E_j\}$, so that $\Pr(A_\ell) \leq \sum_{j=0}^{\ell-1} \Pr(|X_\ell - X_j| > E_j) \leq \ell\alpha$. Using a probabilistic bound (19) in place of (18) to compute Q_ℓ^S introduces a bias,

$$\mathbb{E}[|Q_\ell^S - Q_\ell|] = \mathbb{E}[|Q_\ell^S - Q_\ell| \mid A_\ell] \Pr(A_\ell) + \mathbb{E}[|Q_\ell^S - Q_\ell| \mid \bar{A}_\ell] \Pr(\bar{A}_\ell) \leq \ell\alpha,$$

where we denote the complement of A_ℓ by \bar{A}_ℓ . This bias on the last level L should be of comparable size to the numerical bias and stochastic error, i.e. $L\alpha \leq \text{TOL}$.

The Monte Carlo and multilevel Monte Carlo estimator using selective refinement are denoted by

$$\widehat{Q}_{N,L}^{\text{MC,S}} = \frac{1}{N_0} \sum_{i=0}^N Q_L^{S,i} \quad \text{and} \quad \widehat{Q}_{\{N_\ell\},L}^{\text{ML,S}} = \frac{1}{N_0} \sum_{i=1}^{N_0} Q_0^{S,i} + \sum_{\ell=1}^L \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} Y_\ell^{S,i},$$

respectively, where $Y_\ell^S = Q_\ell^S - Q_{\ell-1}^S$.

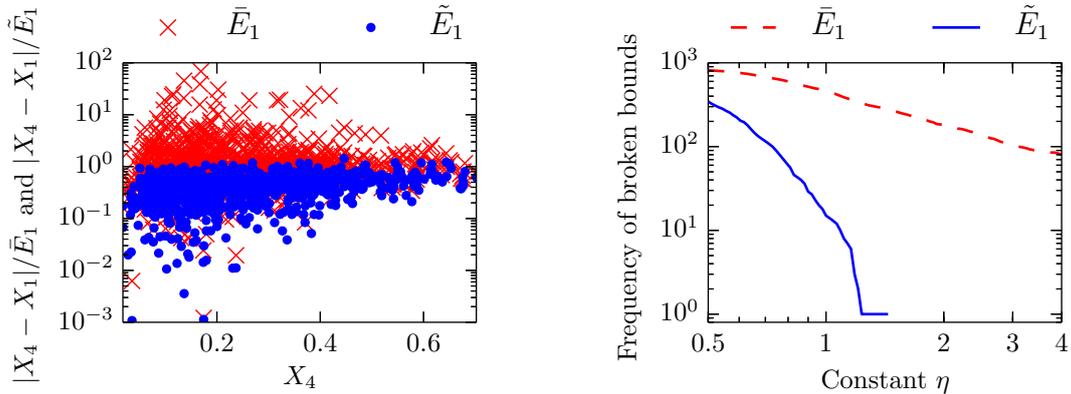
4.3.1 Error bound for sweep efficiency

This section motivates a choice of E_j that satisfies the bound in (19) with probability $1 - \alpha$. Starting with the absolute difference in sweep efficiency between two consecutive levels, we define \bar{E}_j and its upper bound \tilde{E}_j :

$$\begin{aligned} |X_j - X_{j-1}| &= \left| \int_{\mathcal{D}} \chi_{(c,1]}(s_j) - \chi_{(c,1]}(s_{j-1}) \, dx \right| =: \bar{E}_j \\ &\leq \int_{\mathcal{D}} |\chi_{(c,1]}(s_j) - \chi_{(c,1]}(s_{j-1})| \, dx =: \tilde{E}_j. \end{aligned}$$

We now consider $\eta\bar{E}_j$ and $\eta\tilde{E}_j$ as two candidates for a probabilistic bound E_j , where η is a parameter to be determined.

For a sample of size 1000, we computed $X_1, \dots, X_4, \bar{E}_1$, and \tilde{E}_1 . The ratio between the approximate error and the error bounds $|X_4 - X_1|/\bar{E}_1$ and $|X_4 - X_1|/\tilde{E}_1$ were computed and plotted in Figure 5a as functions of X_4 . We also counted number of samples for which the bounds $|X_4 - X_1| \leq \eta\bar{E}_1$ and $|X_4 - X_1| \leq \eta\tilde{E}_1$ are broken for values of η in the range $0.5 \leq \eta \leq 4$. The frequency as function of η is shown in Figure 5b. (The corresponding experiments for $|X_2 - X_1| \leq \eta\bar{E}_1$ and $|X_3 - X_1| \leq \eta\tilde{E}_1$ were performed. The figures only present the results for $\ell = L = 4$, which was the least optimistic case). The means of \bar{E}_1 and \tilde{E}_1 were estimated to 0.028 and 0.056, respectively. It is thus clear from the figures and those mean values that \bar{E}_1 is more suitable as an error bound, since a much larger constant η must be used to get the same error frequency with \bar{E}_1 than with \tilde{E}_1 . This would be worthwhile if \bar{E}_1 was in average much smaller than \tilde{E}_1 , however, the ratio between the mean estimates determines this is not the case.



(a) Scatter plot of ratio between approximate error and error bound.

(b) Frequency of $|X_4 - X_1| \not\leq \eta \bar{E}_1$ and $\not\leq \eta \tilde{E}_1$ among the 1000 realizations for $0.5 \leq \eta \leq 4$.

Figure 5: Results from experiment to determine error bound and η .

Guided by Figure 5b, we see that for $\eta = 2$ the error bound was not broken for a single realization. The behaviour of the tail of the distribution of $|X_4 - X_1|/\tilde{E}_1$ indicates that the probability for breaking the error bound $2\tilde{E}_1$ is less than 10^{-3} . Based on this experiment, we choose

$$E_j = \eta \tilde{E}_j$$

with $\eta = 2$ as our error bound with the estimate $\alpha \approx 10^{-3}$. Note that the computational cost for the error bound is proportional to that of the quantity of interest itself. We extrapolate the results to hold for all $j = 1, 2$ and 3 .

4.4 Evaluation

We are now ready to evaluate the four combinations Monte Carlo (MC), Monte Carlo with selective refinement (MC-SR), multilevel Monte Carlo (MLMC) and multilevel Monte Carlo with selective refinement (MLMC-SR) with respect to computational cost with the aim to obtain estimators of $\mathbb{E}[Q_L]$ with variance $\approx 0.5 \cdot 10^{-4}$.

We summarize the two-phase flow setting. Q_L is the failure probability functional of X_L , the sweep efficiency functional on discretization level L , i.e. $\mathbb{E}[Q_L] = \Pr(X_L \leq y)$ with $y = 0.08$ in our case. All four setups rely on generating realizations of sweep efficiency X_ℓ for different levels ℓ . This procedure starts by generating the lognormal permeability field on a mesh ℓ , where we use the procedure described in Section 3.3, then solving the PDE numerically using the scheme in Section 3.2. The approximation X_ℓ of the sweep efficiency is computed as described in Section 3.4. For the selective refinement procedure, an error bound for X_ℓ is computed using $\eta \tilde{E}_\ell$ in Section 4.3.1.

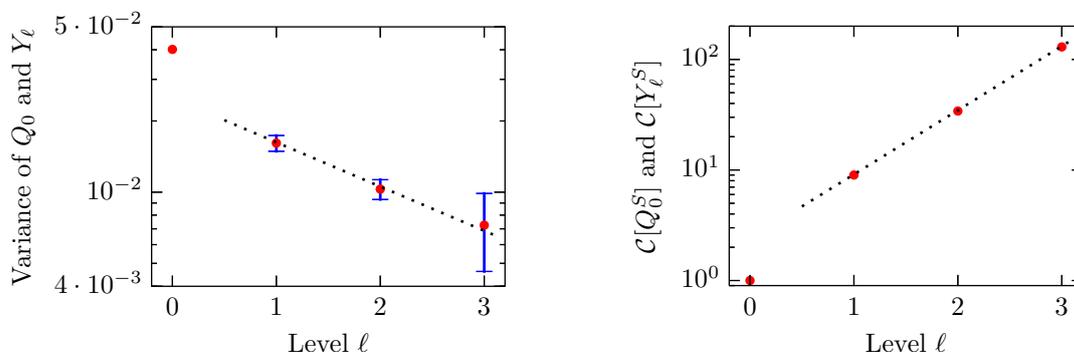
The easiest method to apply is the MC method, which requires knowledge only of $\text{Var}[Q_L]$ (which still is unknown) to determine N . Selective refinement requires an error bound, where we use the error bound developed in Section 4.3.1. MLMC requires a cost model and variance estimates of the correctors $\text{Var}[Y_\ell]$ to determine N_ℓ in (17). As mentioned previously, the variance become increasingly costly to estimate as ℓ increases.

Before evaluating the four setups, we present experiments performed to estimate variances and cost models.

Variance estimates were computed based on a sample of size $4 \cdot 10^4$ for $\ell = 0, 1, 2$ and based on a sample of size $4 \cdot 10^3$ for $\ell = 3$. 95% confidence intervals for the variance were computed for $\ell \geq 1$. The results are presented in Figure 6a. The dotted line is the graph of $\ell \mapsto C\gamma^\ell$ with $C = 0.025$ and $\gamma = 0.65$ and is a convergence rate estimate based on this figure. While the last level ($\ell = 3$) does not verify the rate due to the large confidence interval, it still does not contradict it. Based on this experiment, we extrapolate the rate to hold for all ℓ and base our variance estimate for $\ell \geq 1$ on the function graphed by the dotted line.

As a basis for the cost model we use that the expected computational cost $\mathcal{C}[Q_\ell]$ for a realization follows very closely the relation $\mathcal{C}[Q_\ell] = 2^{3\ell}$ since the mesh size parameter and time step are halved for every level. For the range of meshes used, the computational cost scales linearly with the number of degrees of freedom in the discretization. As is mentioned in Appendix A, the cost to generate random permeability is negligible compared to the cost to do the two-phase flow simulation. The cost for an MLMC-corrector is $\mathcal{C}[Y_\ell] = \mathcal{C}[Q_\ell] + \mathcal{C}[Q_{\ell-1}]$. The costs for a selectively refined realization of Q_ℓ^S and corrector Y_ℓ^S are equal (since $Q_{\ell-1}^S$ comes for free when computing Q_ℓ^S) and are $\mathcal{C}[Y_\ell^S] = \mathcal{C}[Q_\ell^S] = \mathcal{C}[Q_0] + \mathcal{C}[Q_1] + \dots + \mathcal{C}[Q_J]$ for some $J \leq \ell$ where J is random.

The mean computational cost $\mathcal{C}[Q_0^S]$ and $\mathcal{C}[Y_\ell^S]$ at each level $\ell = 1, 2$ and 3 were computed. The results are shown in Figure 6b. Based on this experiment, the cost model for the correctors was chosen as $\mathcal{C}[Y_\ell^S] = C\gamma^{-q\ell}$ with $C = 2.4$ and $q = 3.1$ (where $\gamma = 0.65$ from the variance experiment).



(a) Variance estimates for Q_0 and Y_ℓ as function of level. Red points are variance estimates. Blue bars are confidence intervals. Dotted line is graph $(\ell, C\gamma^\ell)$ with $C = 0.025$ and $\gamma = 0.65$.

(b) Expected computational cost for selective refinement algorithm as function of level. Red points are mean costs. Dotted line is graph $(\ell, C\gamma^{-q\ell})$ with $C = 2.4$, $\gamma = 0.65$ and $q = 3.1$.

Figure 6: Results from experiment in Section 4.4.

In summary, we use the following models for variance and expected cost, with $\gamma = 0.65$

and $\ell \geq 1$:

$$\begin{aligned}
\text{Var} [Q_0] &= \text{Var} [Q_0^S] \approx 0.0401, \\
\text{Var} [Y_\ell] &= \text{Var} [Y_\ell^S] \approx 0.025 \cdot \gamma^\ell, \\
\mathcal{C} [Q_0] &= \mathcal{C} [Q_0^S] = 1 \quad (\text{this defines unit time}), \\
\mathcal{C} [Q_\ell] &\approx 2^{3\ell}, \\
\mathcal{C} [Y_\ell] &\approx (2^3 + 1) \cdot 2^{3(\ell-1)} \approx 1.125 \cdot \gamma^{-4.8\ell}, \\
\mathcal{C} [Q_\ell^S] &= \mathcal{C} [Y_\ell^S] \approx 2.4 \cdot \gamma^{-3.1\ell}.
\end{aligned} \tag{20}$$

To save computational resources, selective refinement was used to do this experiment, whose total cost was $2.32 \cdot 10^6$.

4.4.1 Monte Carlo

We picked a sample of size $N = 2000$ and computed two estimates of $\widehat{Q}_{N,L}^{\text{MC}}$. The sample mean was 0.0755 and sample variance $3.49 \cdot 10^{-5}$ working as estimate for $\text{Var} [\widehat{Q}_{N,L}^{\text{MC}}]$. The expected computational cost for this estimator is $\mathcal{C} [\widehat{Q}_{N,L}^{\text{MC}}] = 2000 \cdot 2^{3 \cdot 4} = 8.19 \cdot 10^6$.

4.4.2 Monte Carlo with selective refinement

Based on the same sample as for the MC method, we computed $\widehat{Q}_{N,L}^{\text{MC,S}}$ using the error bound $\eta \widetilde{E}_k$ presented in Section 4.3.1. Note that \widetilde{E}_0 is not available, since the error bound relies on a simulation on a coarser mesh. Starting with $j = 1$ in Algorithm 1, the MC-SR estimator still uses realizations on level 0 in order to compute \widetilde{E}_1 .

The estimate results (number of failures) were identical to those of MC. The computational cost (including the cost to compute error bounds) for this estimator was estimated using the cost model in (20) to $\mathcal{C} [\widehat{Q}_{N,L}^{\text{MC,S}}] = 2000 \cdot 2.4 \cdot \gamma^{-3.1 \cdot 4} = 1.00 \cdot 10^6$.

4.4.3 Multilevel Monte Carlo

Using the equations for N_ℓ in (17), we choose $\text{TOL} = 10^{-2}$ and compute the sample sizes (rounded up) for each level. The total cost for this estimator was estimated using the cost model in (20) to $\mathcal{C} [\widehat{Q}_{\{N_\ell\},L}^{\text{ML}}] = 1.28 \cdot 10^6$.

4.4.4 Multilevel Monte Carlo with selective refinement

Using (17) with $\text{TOL} = 10^{-2}$ the sample sizes (rounded up) were computed for each level. The total cost (including the cost to compute error bounds) for this estimator was estimated using the cost model in (20) to $\mathcal{C} [\widehat{Q}_{\{N_\ell\},L}^{\text{ML,S}}] = 2.65 \cdot 10^5$.

4.4.5 Variance of MLMC estimators

The performance of the four estimators \widehat{Q} can be compared by examining the product $\mathcal{C} [\widehat{Q}] \cdot \text{Var} [\widehat{Q}]$. Given a deepest level L , this product is constant for all methods under

Table 4: Expected cost estimates, variance estimates and their product for the four setups.

Estimator (\widehat{Q})	$\mathcal{C} [\widehat{Q}]$	$\text{Var} [\widehat{Q}]$	$\mathcal{C} [\widehat{Q}] \cdot \text{Var} [\widehat{Q}]$
MC ($\widehat{Q}_{N,L}^{\text{MC}}$)	$8.19 \cdot 10^6$	$3.49 \cdot 10^{-5}$	286
MC-SR ($\widehat{Q}_{N,L}^{\text{MC,S}}$)	$1.00 \cdot 10^6$	$3.49 \cdot 10^{-5}$	34.9
MLMC ($\widehat{Q}_{\{N_\ell\},L}^{\text{ML}}$)	$1.28 \cdot 10^6$	$3.23 \cdot 10^{-5}$	41.3
MLMC-SR ($\widehat{Q}_{\{N_\ell\},L}^{\text{ML,S}}$)	$2.65 \cdot 10^5$	$3.23 \cdot 10^{-5}$	8.56

Table 5: Cost (relative to the cost of MLMC-SR) to realize the four estimators with equal variance.

	No selective refinement	Selective refinement
Monte Carlo	33.4	4.08
Multilevel Monte Carlo	4.82	1.00

investigation, since they are all Monte Carlo methods where sample size and variance are inversely proportional. Based on the expected cost estimates and variance estimates, this product is presented in Table 4 for each setup. A description of the estimation of the variance of the MLMC (and MLMC-SR) estimator follows.

To determine the variance of the MLMC estimators, a sample of 50 MLMC-SR estimates was computed and the sample variance of these 50 estimators was $\text{Var} [\widehat{Q}_{\{N_\ell\},L}^{\text{ML,S}}] \approx 3.23 \cdot 10^{-5}$. The variance of the MLMC estimator without selective refinement is assumed to be similar, based on the argument that the selective refinement procedure introduces only a bias in the order of 10^{-3} cannot have a significant impact on this variance estimate.

The values of the 50 estimates are plotted at $\ell = L = 4$ in Figure 7 together with a 95% confidence interval for the estimator. Additionally, for all $\ell = 0, \dots, 4$, the mean values of the 50 truncated estimators (i.e. where only the first ℓ correctors in (16) are included) are plotted in the figure together with a 95% confidence interval for these mean values.

In this experiment, we have assumed normality of the MLMC estimator. Normality of the MLMC estimator does hold asymptotically using a generalized central limit theorem, under the condition that the numerical bias decreases as $\text{TOL}^{1-\epsilon}$ for some $0 < \epsilon < 1$ while the statistical error decreases as TOL when $\text{TOL} \rightarrow 0$ (see Lemma 2). For this particular experiment, a Q-Q-plot was used to verify that the estimator was close to normally distributed.

4.4.6 Discussion

The obtained products $\mathcal{C} [\widehat{Q}] \cdot \text{Var} [\widehat{Q}]$ normalized with respect to the cheapest setup for all combinations are presented in Table 5. The results show that there are significant gains in computational cost using multilevel Monte Carlo and selective refinement in addition to standard Monte Carlo. Also, the combination of multilevel Monte Carlo and selective refinement gives further gains, in total a factor 33.

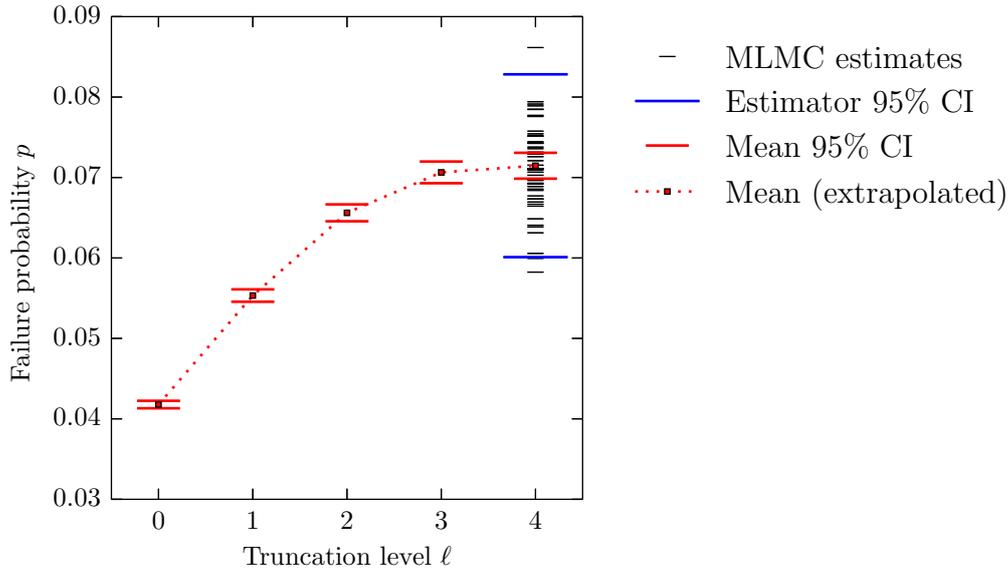


Figure 7: Narrow (black) bars at $\ell = 4$ are 50 MLMC-SR estimates. Wide (blue) bars at $\ell = 4$ cover the 95% confidence interval (CI) for the estimator. Squares (red) at $\ell = 0, \dots, 4$ are the mean values of 50 the level ℓ -truncated estimates. Bars (red) indicate the 95% CI for these mean values.

These gains depend on the choice of L and they will increase as L increases. For the MLMC estimators constructed above, the goal tolerance TOL was chosen as $\text{TOL} = 10^{-2}$ and the number of samples were chosen based on that. It is clear from Figure 7 that there is a large numerical bias for low values of ℓ . However, the bias improvement obtained by going from $\ell = 2$ to $\ell = 3$ or 4 is dominated by the statistical error of the estimator, indicated by the 95% confidence interval for the estimator at $\ell = 4$. A better balance between the two sources of error at this level L would be attained if TOL was chosen smaller.

In addition to the cost to realize an estimator, the cost to construct the estimators differ between the setups. For selective refinement we established the reliability of an error bound by numerical experiments in Section 4.3.1 (which is not necessary if the error bound is already trusted). For multilevel Monte Carlo, we performed an additional experiment to determine the variance of the correctors and to determine the cost model for selective refinement.

5 Conclusion

It is evident from the experiments on the two-phase flow simulations that if a good error bound on the quantity of interest (in this case sweep efficiency) is available, computational savings of one order of magnitude is possible by using selective refinement. Significant savings in computational cost can most likely be expected from other similar applications by using selective refinement in combination with standard and multilevel Monte Carlo estimators.

A Circulant embedding

The circulant embedding technique [10] is a fast method to generate stationary random fields on a cartesian grid with equispaced points for a given covariance function C .

Let n be the number of rows and columns in the mesh \mathcal{T}_h , i.e. $n = h^{-1}$. The total number of vertices is n^2 . Let x_k denote the coordinates of vertex k , where the vertices are ordered lexicographically, for example row-by-row. The covariance matrix $\mathbf{C} = (c_{ij})_{ij}$ for the vertices can be written

$$c_{ij} = C(\|x_i - x_j\|_2),$$

where C was defined in Section 2.2. The lexicographical ordering of the vertices in the cartesian grid gives \mathbf{C} a symmetric Toeplitz block structure of symmetric Toeplitz blocks (each block corresponding to a row in the grid, and each column within a block to a column in the row). We define a $(2n-2) \times n$ extension matrix \mathbf{E}_n such that for any \mathbf{y} , we have $\tilde{\mathbf{y}} = \mathbf{E}_n \mathbf{y}$ where $\tilde{y}_i = y_i$ for $1 \leq i \leq n$ and $\tilde{y}_{n+i} = y_{n-i}$ for $1 \leq i \leq n-2$. We further define $\mathbf{E} = \mathbf{E}_n \otimes \mathbf{E}_n$. Additionally, we define an $n \times (2n-2)$ restriction matrix \mathbf{R}_n such that for any $\tilde{\mathbf{y}}$, it holds $\mathbf{y} = \mathbf{R}_n \tilde{\mathbf{y}}$ where $y_i = \tilde{y}_i$ for $1 \leq i \leq n$. Also, $\mathbf{R} = \mathbf{R}_n \otimes \mathbf{R}_n$. Note that $\mathbf{R}\mathbf{E} = \mathbf{I}$, the identity matrix. Finally, we define the 2D discrete Fourier transform matrix $\mathbf{F} = \mathbf{F}_{2n-2} \otimes \mathbf{F}_{2n-2}$, where \mathbf{F}_{2n-2} is the discrete Fourier transform matrix of size $2n-2$.

The covariance matrix \mathbf{C} has an extension $\tilde{\mathbf{C}} = \mathbf{E}\mathbf{C}\mathbf{E}^T$ that is a block circulant matrix with circulant blocks where each block is of size $2n-2$ and there are $2n-2$ blocks per dimension. (We assume this minimal embedding is nonnegative definite, which it always is for our choice of correlation lengths and grid sizes). This block structure allows $\tilde{\mathbf{C}}$ to be diagonalized by the 2D discrete Fourier transform (see e.g. [8]),

$$\tilde{\mathbf{C}} = \mathbf{F}^H \Lambda \mathbf{F},$$

where \cdot^H denotes conjugate transpose and Λ is a diagonal matrix with the eigenvalues on its diagonal. The 2D discrete Fourier transform of the first row $\tilde{\mathbf{c}}_1$ of $\tilde{\mathbf{C}}$ determines the eigenvalues,

$$\Lambda = \text{diag}((2n-2)\mathbf{F}\tilde{\mathbf{c}}_1).$$

Now, let $\epsilon = \epsilon_1 + i\epsilon_2$, with $\epsilon_1, \epsilon_2 \in \mathcal{N}(\mathbf{0}, \mathbf{I})$ be a random complex vector of length $(2n-2)^2$ and let $\tilde{\mathbf{y}} = \mathbf{F}^H \Lambda^{1/2} \epsilon$. Due to the linear relation with ϵ , both the real and imaginary part of $\tilde{\mathbf{y}}$ are normally distributed with zero mean with covariance $\tilde{\mathbf{C}}$. Let \mathbf{y} be the vector of interest embedded in $\tilde{\mathbf{y}}$, i.e. $\mathbf{y} = \mathbf{R}\tilde{\mathbf{y}}$. The real part of \mathbf{y} can be written $\Re(\mathbf{y}) = (\mathbf{y} + \bar{\mathbf{y}})/2$, and the covariance matrix of the real part is $\mathbb{E}[\Re(\mathbf{y})\Re(\mathbf{y})^T] = \mathbb{E}[\mathbf{y}\mathbf{y}^H + \mathbf{y}\mathbf{y}^T]/2$. We have

$$\mathbb{E}[\mathbf{y}\mathbf{y}^H] = \mathbb{E}[\mathbf{R}\tilde{\mathbf{y}}\tilde{\mathbf{y}}^H\mathbf{R}^T] = \mathbf{R}\mathbf{F}^H\Lambda^{1/2}\mathbb{E}[\epsilon\epsilon^H]\Lambda^{1/2}\mathbf{F}\mathbf{R}^T = 2\mathbf{R}\mathbf{F}^H\Lambda\mathbf{F}\mathbf{R}^T = 2\mathbf{C},$$

and $\mathbb{E}[\mathbf{y}\mathbf{y}^T] = \mathbf{0}$, since $\mathbb{E}[\epsilon\epsilon^H] = \mathbf{I}$ and $\mathbb{E}[\epsilon\epsilon^T] = \mathbf{0}$. That is, the real part of \mathbf{y} has covariance matrix \mathbf{C} . This also holds for the imaginary part, and the two parts are independent. Thus, for each vector \mathbf{y} , two fields are generated. However, the additional data generated (roughly 3/4) due to the extension cannot be used.

In terms of computational cost, computing Λ is an $\mathcal{O}(h^{-2}\log(h^{-2}))$ operation and generating a realization of \mathbf{y} is also an $\mathcal{O}(h^{-2}\log(h^{-2}))$ operation. This is cheaper than

using e.g. Cholesky factorization of \mathbf{C} , which is $\mathcal{O}(h^{-4})$ for factorization and generating a realization.

The mesh on which the fields are generated need to be chosen fine enough to avoid aliasing effects (see e.g. [14]). For the lower levels the meshes are coarse enough for the aliasing effect to be significant for pointwise sampling of the random field. Our approach is as follows. We compute Λ based on the finest mesh with $h = 1/256$ (with 65536 vertices) to make the aliasing effect negligible. When a random field is to be computed for a coarser mesh with $\tilde{n}^2 = \tilde{h}^{-2}$ vertices ($\tilde{n} < n$), we truncate Λ and keep only the lowest \tilde{n}^2 frequencies (the \tilde{n} lowest for each dimension and their combinations). Random numbers are generated only for the kept eigenvalues. Refinement of a field to a finer mesh is done by keeping the coefficients for the low frequency eigenfunctions and generating new random numbers for higher frequency eigenfunctions to match the number of used eigenvalues with the number of vertices. The computational cost to compute Λ is still $\mathcal{O}(h^{-2} \log(h^{-2}))$, however, this only needs to be done once. The computational cost to generate a new realization depends only on the coarse mesh for which it is generated and is $\mathcal{O}(\tilde{h}^{-2} \log(\tilde{h}^{-2}))$. Thus, the computational cost of generating random data for a simulation is asymptotically negligible in comparison to performing the simulation itself. Solving the PDE numerically costs at least $\mathcal{O}(\tilde{h}^{-3})$.

B Asymptotic normality of MLMC estimator

The asymptotic normality holds for $\widehat{Q}_{N,L}^{\text{MC}}$ by the central limit theorem (CLT) and for $\widehat{Q}_{\{N_\ell\},L}^{\text{ML}}$ with fixed L as $\text{TOL} \rightarrow 0$ by applying CLT to each MC estimator in the MLMC estimator individually. However, if numerical bias and statistical error are to be reduced simultaneously, a result from [7] applied to our problem states that asymptotic normality of $\widehat{Q}_{\{N_\ell\},L}^{\text{ML}}$ as $\text{TOL} \rightarrow 0$ holds if the convergence rate of the numerical bias is strictly smaller than the rate of the statistical error, i.e. new levels are added to the estimator slightly slower than necessary for even balance between the two error sources.

Lemma 2. *Assume $\mathbb{E}[|Y_\ell|] \geq C\gamma^\ell$ for some $C > 0$. If letting $L = \log(\text{TOL}^{1-\epsilon})/\log(\gamma)$ for some $0 < \epsilon < 1$ and choosing N_ℓ according to (17), then for $\mathcal{A} = \widehat{Q}_{\{N_\ell\},L}^{\text{ML}}$,*

$$\lim_{\text{TOL} \rightarrow 0} \Pr \left(\frac{\mathcal{A} - \mathbb{E}[\mathcal{A}]}{\sqrt{\text{Var}[\mathcal{A}]}} \leq z \right) = \Phi(z),$$

where Φ is the cdf for the standard normal distribution.

Proof. We apply [7, Lemma 7.1] which is based on the Lindeberg–Feller theorem on asymptotic normality for sums of independent but not necessarily identically distributed random variables.

Considering the assumptions for [7, Lemma 7.1], we first establish $C_1\gamma^\ell \leq \mathbb{E}[(Y_\ell - \mathbb{E}[Y_\ell])^2]$. Since Y_ℓ is discrete with outcomes $-1, 0$ and 1 , we can set $\mathbb{E}[Y_\ell] = p_1 - p_{-1}$ for some $p_1, p_{-1} \geq 0$. Then, using $p_1 + p_{-1} = \mathbb{E}[|Y_\ell|] \geq C\gamma^\ell$, we get

$$\mathbb{E}[(Y_\ell - \mathbb{E}[Y_\ell])^2] = p_1 + p_{-1} - (p_1 - p_{-1})^2 \geq p_1 + p_{-1} - (p_1 + p_{-1})^2 \geq C_1\gamma^\ell$$

for some $C_1 > 0$. Next, since $|Y_\ell - \mathbb{E}[Y_\ell]|$ is bounded by 2, we can bound

$$\mathbb{E}[|Y_\ell - \mathbb{E}[Y_\ell]|^3] = C_2 \mathbb{E}[|Y_\ell - \mathbb{E}[Y_\ell]|^2] \leq C_3 \gamma^\ell$$

for some $C_2, C_3 > 0$. Referring to the notation in [7, Lemma 7.1] we let $\beta = \gamma^{-1}$, $\delta = q_2 = q_3 = \tau = 1$ and apply the lemma to obtain the result. \square

Acknowledgement

We are grateful to Andreas Hellander and Salman Toor for assistance with computational parallelization tools (MOLNs) and resources (SMOG Cloud).

References

- [1] R. Avikainen. On irregular functionals of SDEs and the Euler scheme. *Finance Stoch.*, 13(3):381–401, 2009.
- [2] A. Barth, C. Schwab, and N. Zollinger. Multi-level Monte Carlo finite element method for elliptic PDEs with stochastic coefficients. *Numer. Math.*, 119(1):123–161, 2011.
- [3] D. Boffi, F. Brezzi, and M. Fortin. *Mixed finite element methods and applications*, volume 44 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin Heidelberg, 2nd edition, 2013.
- [4] J. Charrier, R. Scheichl, and A. L. Teckentrup. Finite element error analysis of elliptic PDEs with random coefficients and its application to multilevel Monte Carlo methods. *SIAM J. Numer. Anal.*, 51(1):322–352, 2013.
- [5] Z. Chen, G. Huan, and B. Li. An improved IMPES method for two-phase flow in porous media. *Transp. Porous Media*, 54(3):361–376, 2004.
- [6] K. A. Cliffe, M. B. Giles, R. Scheichl, and A. L. Teckentrup. Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients. *Comput. Vis. Sci.*, 14(1):3–15, 2011.
- [7] N. Collier, A.-L. Haji-Ali, F. Nobile, E. von Schwerin, and R. Tempone. A continuation multilevel Monte Carlo algorithm. *BIT*, 55:399–432, 2015.
- [8] P. J. David. *Circulant matrices*. John Wiley & Sons, Inc., New York, 1979.
- [9] D. A. Di Pietro and A. Ern. *Mathematical Aspects of Discontinuous Galerkin Methods*. Springer-Verlag Berlin Heidelberg, 2012. Mathématiques et Applications, Vol. 69.
- [10] C. Dietrich and G. Newsam. Fast and exact simulation of stationary gaussian processes through circulant embedding of the covariance matrix. *SIAM J. Sci. Comput.*, 18(4):1088–1107, 1997.

- [11] D. Elfverson, D. Estep, F. Hellman, and A. Målqvist. Uncertainty quantification for approximate p-quantiles for physical models with stochastic inputs. *SIAM/ASA J. Uncertain. Quantif.*, 2:826–850, 2014.
- [12] D. Elfverson, F. Hellman, and A. Målqvist. A multilevel Monte Carlo method for computing failure probabilities. *ArXiv e-prints:1408.6856*, 2014.
- [13] L. W. Gelhar. Stochastic subsurface hydrology from theory to applications. *Water Resources Research*, 22(9S):135S–145S, 1986.
- [14] L. W. Gelhar. *Stochastic subsurface hydrology*. Prentice Hall, New Jersey, 1993.
- [15] M. B. Giles. Multilevel Monte Carlo path simulation. *Oper. Res.*, 56(3):607–617, 2008.
- [16] M. B. Giles, T. Nagapetyan, and K. Ritter. Multilevel Monte Carlo approximation of distribution functions and densities. *SIAM/ASA J. Uncertain. Quantif.*, 3(1):267–295, 2015.
- [17] A.-L. Haji-Ali, F. Nobile, and R. Tempone. Multi-index Monte Carlo: when sparsity meets sampling. *Numer. Math.*, pages 1–40, 2015.
- [18] A.-L. Haji-Ali, F. Nobile, E. von Schwerin, and R. Tempone. Optimization of mesh hierarchies in multilevel Monte Carlo samplers. *Stoch. Partial Differ. Equ. Anal. Comput.*, pages 1–37, 2015.
- [19] S. Heinrich. Multilevel monte carlo methods. In *Large-Scale Scientific Computing*, volume 2179 of *Lecture Notes in Computer Science*, pages 58–67. Springer Berlin Heidelberg, 2001.
- [20] R. J. LeVeque. *Finite volume methods for hyperbolic problems*. Cambridge University Press, New York, 2002. Cambridge Texts in Applied Mathematics.
- [21] M. Park and A. Teckentrup. Improved multilevel Monte Carlo methods for finite volume discretisations of darcy flow in randomly layered media. *ArXiv e-prints:1506.04694*, 2015.
- [22] P. A. Raviart and J. M. Thomas. A mixed finite element method for 2-nd order elliptic problems. In Ilio Galligani and Enrico Magenes, editors, *Mathematical Aspects of Finite Element Methods*, volume 606 of *Lecture Notes in Mathematics*, pages 292–315. Springer Berlin Heidelberg, 1977.
- [23] A. L. Teckentrup, R. Scheichl, M. B. Giles, and E. Ullmann. Further analysis of multilevel Monte Carlo methods for elliptic PDEs with random coefficients. *Numer. Math.*, 125(3):569–600, 2013.