Multilevel Monte Carlo methods for computing failure probability of porous media flow systems

F. Fagerlund[a], F. Hellman[b,*], A. Målqvist[c], A. Niemi[a]

[a]*Department of Earth Sciences, Uppsala University, Box 337, SE-751 05 Uppsala, Sweden.*
[b]*Department of Information Technology, Uppsala University, Box 337, SE-751 05 Uppsala, Sweden.*
[c]*Department of Mathematical Sciences, Chalmers University of Technology and University of Gothenburg, SE-412 96 Göteborg, Sweden.*

Abstract

We study improvements of the standard and multilevel Monte Carlo method for point evaluation of the cumulative distribution function (failure probability) applied to porous media two-phase flow simulations with uncertain permeability. To illustrate the methods, we study an injection scenario where we consider sweep efficiency of the injected phase as quantity of interest and 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 identify small subsets of realizations to solve 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 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.

Keywords: CDF estimation, Failure probability, Porous media flow simulation, Multilevel Monte Carlo, Selective refinement

1. Introduction

Simulation of fluid flow in geological formations of porous and fractured media is a tool used to help understand the processes governing groundwater flow, enhanced oil recovery and underground carbon storage. Simulations can help assess the suitability for enhanced oil recovery or carbon storage at potential injection sites. There are, however, typically large uncertainties in the physical properties (e.g. permeability, porosity, caprock topology) of the porous medium. In this work, we consider forward propagation of uncertainties in input data through the governing flow model to a scalar quantity of interest. The deterministic problems are typically computationally expensive to solve. Adding a high-dimensional stochastic input space poses an even greater challenge. It is not always necessary to compute the full distribution of the quantity of interest, but it is common to seek, for example, the mean value or variance of the quantity of interest. Here, we seek the probability for the quantity of interest to be less than a fixed critical value, an event that we call a failure. We assume that the random input data follows a known distribution.

Although the methods presented in this paper are applicable to more general situations, this work focuses on a specific, but common problem for two-phase fluid flow in porous media. We consider a scenario where a non-wetting fluid is injected into a reservoir initially filled with a wetting fluid. We compute the reservoir volume swept by the injected fluid after a certain time. This is a measure of the sweep efficiency. One situation where this problem is of interest is when estimating the capacity of a geological formation to store carbon dioxide. In this context we consider it a failure of the carbon dioxide storage system if the estimated storage capacity of a target formation is lower than some critical value needed for the planned storage operations. Sweep efficiency is one important indicator for storage capacity [21, 27, 28]. We consider sweep efficiency as our quantity of interest and the permeability field as a random input data. We then seek to estimate the probability that the sweep efficiency is smaller than

*Corresponding author

some critical value. We call this the failure probability. We use a macroscopic incompressible two-phase fractional flow model based on Darcy's law and conservation of mass, which we solve using standard finite element and finite volume techniques.

The general setting is that we have a quantity of interest X defined on random input sample space Ω and that we seek $p = \Pr(X \leq y)$ for some critical value y. For our problem, X is the sweep efficiency of the injected phase. The sample space Ω contains the possible permeability fields. The failure probability p is the value of the cumulative distribution function (cdf) for the sweep efficiency functional X at the critical value y. In this work, we study 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 papers [11, 12] where the selective refinement technique was introduced. While those two papers focus entirely on asymptotic cost rate results (for large sample sizes), this work applies the methods presented therein to more realistic problems in order to investigate to what extent the predicted gains in computational efficiency can be realized in practice. In addition, special care is taken to describe the construction of the error bounds necessary to use the selective refinement approach. In total four methods of Monte Carlo type are studied: Standard Monte Carlo, multilevel Monte Carlo [15] and the two previous Monte Carlo methods in combination with selective refinement [12]. We briefly review the methods and related works below.

An estimate of the failure probability in a Monte Carlo (MC) estimator is obtained by computing the mean of a set of zeros and ones, from independent simulations yielding successful or failing scenarios, respectively. An improvement of the MC estimator for application on numerical simulations of controllable numerical accuracy is the multilevel Monte Carlo (MLMC) estimator [20] first introduced in the context of differential equations in [15]. It has since then been applied to elliptic PDEs in [3, 7, 25, 29] and has been further analyzed and extended in [8, 18, 19]. 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. MLMC methods for point evaluation of a cdf is also studied in [1, 17]. In [1] they investigate how the convergence rate of the error in irregular functionals of the quantity of interest approximation (in particular the failure probability functional, or binary option payoff) depends on the convergence rate of the error in the quantify of interest approximation itself. This motivates that the the irregular failure probability functional can be used in a multilevel Monte Carlo method. In [17], the failure probability functional is smoothed to improve the convergence rates. In contrast, the methods considered in the present work are all based on the irregular functional, i.e. no smoothing is used. 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.

The paper 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 their asymptotic computational cost rates. Two preparatory experiments (specific to the sweep efficiency problem) where a computable error bound, and variances and cost models are estimated are presented in Section 5. In Section 6 we describe the experiment that quantifies the computational cost for the four methods. We conclude with a summary and discussion in Section 7.

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 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. We emphasize that the model presented here should be considered an example. The methods presented in Section 4 are applicable in a much more general setting.

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

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

in $\mathcal{D}$, where ρ_α is density, ϕ is porosity, s_α is saturation, $\mathbf{u}_\alpha$ 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,

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

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 the following power law (a modification of the Brooks–Corey relation [4])

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

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

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

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 equations (or total fluid flux equations):

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

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

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

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

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

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

No initial values need to be explicitly stated for the pressure and flux, since they are completely determined by the initial values of s by (5). 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 that boundary. More precisely, we have for $t \geq 0$,

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

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

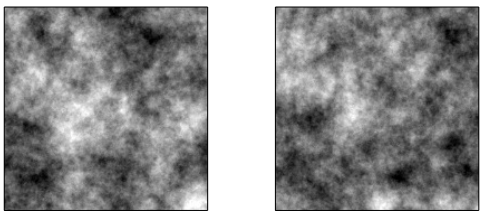
$$k(x) = \exp(\kappa(x)),$$

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,

$$\text{Cov}\left[\kappa(x_1), \kappa(x_2)\right] = \sigma^2 \exp\left(\frac{-\|x_1 - x_2\|_2}{d}\right), \tag{8}$$

where σ is the standard deviation and $d = 0.1$ is the correlation length. (Note that κ is the natural logarithm of k.) For this stationary field the covariance depends only on the distance between two points. Two realizations of this field are shown in Figure 1.

We study two values of the standard deviation, $\sigma = 1$ (low variance case) and $\sigma = 3$ (high variance case). All experiments are performed for both cases, but for space considerations we present the details for the high variance case only. Although the lower variance is more frequently encountered when modeling heterogeneous permeability fields, higher variance can be justified in a number of modeling contexts. For example, for two sand layers at the Heletz CO_2 pilot injection site, the standard deviation of the (natural logarithm of the) permeability in the layers was estimated within a range $1.7 \leq \sigma \leq 2.5$ [23]. In a multimodal model for the permeability at Southwest Scania site including three materials (sandstone, siltstone and clay), the total standard deviation was as high as $\sigma = 3.7$ (Case 1 in [33]). High standard deviation occurs also when modeling fractures as 2D-planes for simulating transport in fractured rocks, for example in [31] where $\sigma = 3.1$. In a modeling study addressing the effect of heterogeneity on CO_2 storage [30] used the range of values $0.46 \leq \sigma \leq 2.3$. To show that the methodology is useful for both high and low variances, we address both. The results for the low variance case are presented in Section 6.6. Until then, we demonstrate the methods by showing the results for the high variance case only.



(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 $d = 0.1$ and $\sigma = 3$.

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,

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 (d)	0.1
Stop time (T)	0.2

we consider the swept proportion after a fixed time T. The functional is formally expressed as

$$X = X(s) = \int_{\mathcal{D}} \chi_{[0,1)}(s(T))\,\mathrm{d}x,$$

where χ_A is the indicator function

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

In other words, $\chi_{[0,1)}(s(T))$ is equal to 0 only where $s(T) = 1$, and hence the integral makes $X(s)$ equal to the area of the domain where $s(T) < 1$. This is the area swept by the injected non-wetting fluid at time T. Since the total area is 1, it is also the proportion of the total area swept.

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.

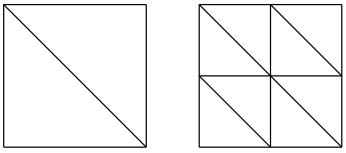
3. Discretization

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. That is, both methods rely on that the numerical error of the quantity of interest is smaller when a fine mesh in the hierarchy is used instead of a coarse mesh. A sequential splitting scheme is presented in Section 3.2 describing the simulation procedure, once grid and data are given. Section 3.3 briefly 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. We define a mesh hierarchy of all meshes $\mathcal{T}_h$ with $h = 2^{-(L_0+\ell)}$ for levels $\ell = 0, 1, \ldots, L$. Here, h is the vertical and horizontal vertex spacing. The coordinates of the vertices are (ih, jh) for $i, j = 0, \ldots, h^{-1} = 2^{L_0+\ell}$ and every square in the grid is split into two triangles by connecting the upper-left and lower-right corners. 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.

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 $d = 0.1$ of k. In, for example [5], it has been indicated that a finite element discretization of the pressure equations does not converge in regimes where the correlation length is not resolved by the mesh for a standard finite element method. Convergence is necessary in order to estimate the numerical error, thus we require $h < d$. In particular, we depend on convergence for computing the error bounds in the selective refinement algorithm presented below. The number of levels L is chosen as 4, which means the finest mesh size is $h = 1/256$.



(a) $h = 1$ (b) $h = 1/2$

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

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 [6]. This split renders the pressure equations linear elliptic equations, and the saturation equation a nonlinear hyperbolic transport equation with fixed total flux field. The pressure equations (5) are solved implicitly for a given time step, keeping saturation fixed from the previous time step. The flux solution from the pressure equations is used in the saturation equation (6), from which the saturation for the next time step is computed using an explicit method.

Let $0 = t^0 < t^1 < \cdots < t^M = T$ 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 equations:

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

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 I_m inner time steps $t^m = t^{m,0} < t^{m,1} < \cdots < t^{m,I_m} = t^{m+1}$, with an inner time step length $\tau_m = t^{m,i+1} - t^{m,i}$. Note that $I_m = \tau/\tau_m$. The saturation equation is then discretized in time using forward Euler:

$$\phi s^{m,i+1} = \phi s^{m,i} - \tau_m \nabla \cdot \mathbf{u}_w^{m,i}, \quad i = 0, \ldots, I_m - 1. \tag{10}$$

where

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

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

For the spatial discretization of the pressure equations, 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. [22]) on a triangular mesh and the saturation is approximated by piecewise constants $s_h^{m,i}$.

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. Given a constant computational cost (a dashed contour line), the optimal choice of Mh is found by minimizing the error (filled contour plot) along that line. Thus, the optimal choice of Mh follows a graph where the gradients of the two functions (cost and error) with respect to M and h are parallel. In our case, the value $Mh = 0.5$ was picked by manual inspection of several figures of the same kind as Figure 3 for different realizations of the permeability field.

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 $Mh = 0.5$.

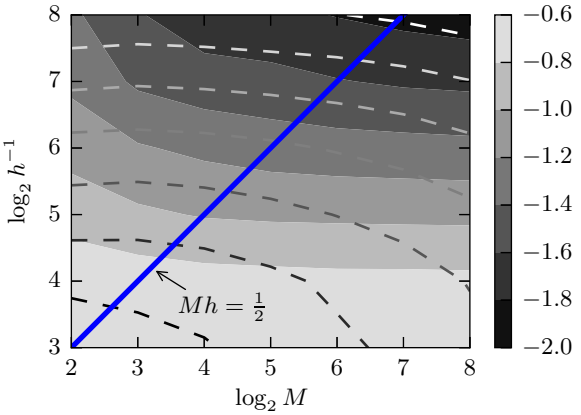


Figure 3: The filled contour plot shows the 10-logarithm of the estimated relative error of s_h^M. The dashed contour plot shows the logarithm of the computational cost. The cost is monotonically increasing from the lower left to the upper right in the plot, with the unit of time chosen to match the colorbar. The solid (blue) line is selected by hand and is one possible and efficient relation between M and h.

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.

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 prominent 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 realize the random field k using the circulant embedding technique [9]. We perform the truncation in the spectral domain and omit high-frequency modes when generating fields for the coarser meshes. This way, the aliasing effect is avoided. See [9] for details on the circulant embedding technique.

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 overestimated, since all values strictly below 1 are considered swept. Instead, we should have a lower threshold than 1 for the functional. Thus, we define the following functional with a threshold $0 < c \leq 1$,

$$X_h^M = \int_{\mathcal{D}} \chi_{[0,c)}(s_h^M)\,\mathrm{d}x,$$

where we choose $c = 0.5$. That is, the part of the domain with saturation between 0 and 0.5 is considered swept. The appearance of a Buckley–Leverett shock front for the current setup of fluids was investigated and we chose the threshold value c to be within the range of the shock. Further, we define

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

when $h = 2^{-(L_0+\ell)}$ and $Mh = 0.5$.

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

Table 2: Parameter values for the discretization.

Parameter name (and symbol)	Value
Coarsest mesh (L_0)	4
Deepest level (L)	4
Sweep efficiency threshold (c)	0.5
Mesh size (h)	$2^{-4}, 2^{-5}, \ldots, 2^{-8}$
Outer time steps (M)	$0.5h^{-1}$

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. For our problem (with $\sigma = 3$) we choose the critical value as $y = 0.08$, i.e. the failure event reads: "the sweep efficiency is less than 8%". We seek $p = \Pr(X \leq 0.08)$.

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. Similar 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 $\widehat{Q}$ are based on discretizations on levels smaller than or equal to L. In the literature of multilevel Monte Carlo methods, most error analysis is based on the root mean square error (RMSE) as a measure of error. We denote RMSE by $e[\cdot]$, and define it for any estimator $\widehat{Q}$ as

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

This definition allows for a convenient split between statistical and numerical error. The variance $\mathrm{Var}\left[\widehat{Q}\right]$ 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 define the error tolerance TOL as an upper limit (to be chosen) of the total RMSE: $e\left[\widehat{Q}\right] \leq \mathrm{TOL}$.

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 combinations. The large sample asymptotic computational cost rates for the standard and multilevel Monte Carlo methods in terms of TOL for the failure probability functional with and without selective refinement have been studied in [11, 12], based on the analysis in [15]. In order to derive asymptotic computational cost rates as is done in the referred papers, convergence rates for the weak error of Q_ℓ and expected cost to compute a realization are assumed to satisfy the following assumption.

Assumption 1. *Assume that for some (not random) constants $0 < \gamma < 1$, $C > 0$, and $q > 0$ 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.

In Section 5.2 we verify these assumptions experimentally for our particular problem and determine the constants C, γ and q. (We will use the same symbol C to represent different constants related to cost and variance models throughout the paper). We make use of two different methods to realize Q_ℓ: standard refinement and selective refinement. They have different costs and consequently different

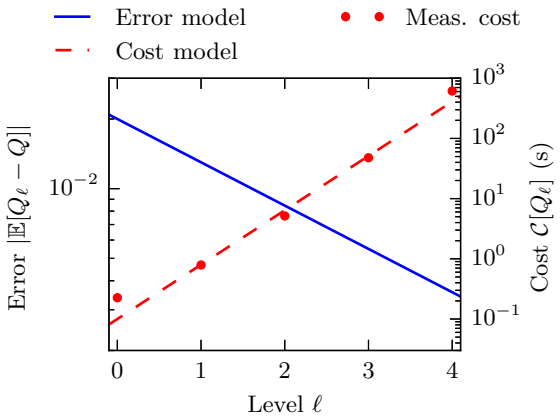


Figure 4: The relation between error and cost in Assumption 1 for $\gamma = 0.65$, and the cost model $q = \log(8)/\log(\gamma) \approx 4.8$ (so that $\gamma^{-q\ell} = 8^\ell$). Along with the cost model, measurements of the expected computational costs for the standard MC are plotted.

values of q above. The computational cost is random in our case, since different permeability fields give rise to linear systems with varying condition number. However, the assumption concerns the expected computational cost, which can be estimated. Figure 4 illustrates the computational cost model and error model used in the numerical experiments. The measured expected computational cost ranges from 0.2 s (level 0) to 600 s (level 4) on a typical computational resource.

In practice, the actual computational cost might not follow exactly a relationship as in A2. Figure 4 illustrates that the actual expected cost seems to obey $q = q(\ell)$, with q increasing with ℓ. However, despite using a simplified model A2 for the expected computational cost, we will note that the multilevel Monte Carlo method will still deliver substantial performance improvements. An empirical model for the computational cost can also be used, and would, if more accurate than the cost model, yield greater savings.

A summary of the asymptotic computational cost rates for all four setups are given in Table 3. The derivation of these rates can be found in [11, 12, 15]. They are valid as TOL $\to 0$ in the case where the statistical and numerical errors are balanced (i.e. decrease with the same rate in terms of TOL). Some way of balancing the error sources is necessary to make the error asymptotically vanish. One might want the errors to converge with different rates, for example to make the multilevel Monte Carlo estimator asymptotically a normal distribution, which requires the stochastic error to decrease with any strictly greater rate than the numerical error (see [8, Lemma 7.1]). However, in practice, an even balance between the errors is justified since it prevents one source of error to dominate the other. 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 [12].

Table 3: Asymptotic cost rates for the failure probability estimators as TOL $\to 0$ 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 $\widehat{Q}^{\text{MC}}_{N,L}$ estimates $\mathbb{E}[Q_L]$ by computing the mean of an i.i.d. N-sample of Q_L,

$$\widehat{Q}^{\text{MC}}_{N,L} = \frac{1}{N}\sum_{i=0}^{N} Q_L^i. \tag{12}$$

Here (and in the remainder of this text) the superscript i in Z^i for arbitrary random variable Z denotes an independent random variable with the same distribution as Z. The realization of these superscripted random variables is what constitute the major computational cost of the the Monte Carlo methods presented in this work. Regarding the realization of Q_L^i, for each i, a random permeability field is generated and a realization of Q_L^i is computed using the procedure described in Section 3.

An MC estimator is unbiased, i.e. $\mathbb{E}\left[\widehat{Q}_{N,L}^{\mathrm{MC}}\right] = \mathbb{E}\left[Q_L\right]$, so the statistical error and numerical bias are

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

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

We here briefly summarize the multilevel Monte Carlo method [15] and the results necessary for implementing it. We refer to the initial works [15, 20] and the survey [16] for more detailed and thorough expositions. In the MLMC method, we introduce the estimator $\widehat{Q}_{\{N_\ell\},L}^{\mathrm{ML}}$ for estimating $\mathbb{E}\left[Q_L\right]$ by expanding it in a telescoping sum,

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

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

$$\widehat{Q}_{\{N_\ell\},L}^{\mathrm{ML}} = \frac{1}{N_0}\sum_{i=1}^{N_0} Q_0^i + \sum_{\ell=1}^{L} \frac{1}{N_\ell}\sum_{i=1}^{N_\ell} Y_\ell^i, \tag{13}$$

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 realization of a corrector $Y_\ell^i = Q_\ell^i - Q_{\ell-1}^i$ is computed by generating one realization of the permeability field k^i and computing both Q_ℓ^i and $Q_{\ell-1}^i$ using that one field. Due to the telescoping sum, the MLMC estimator is again an unbiased estimator of $\mathbb{E}\left[Q_L\right]$. The variance, however, depends on all N_ℓ. The statistical error and numerical bias are,

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

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

$$N_0 = 2C_L\mathrm{TOL}^{-2}\sqrt{\mathrm{Var}\left[Q_0\right]/\mathcal{C}\left[Q_0\right]} \quad \text{and} \quad N_\ell = 2C_L\mathrm{TOL}^{-2}\sqrt{\mathrm{Var}\left[Y_\ell\right]/\mathcal{C}\left[Y_\ell\right]}, \tag{14}$$

where $C_L = \sqrt{\mathrm{Var}\left[Q_0\right]\mathcal{C}\left[Q_0\right]} + \sum_{\ell=1}^{L}\sqrt{\mathrm{Var}\left[Y_\ell\right]\mathcal{C}\left[Y_\ell\right]}$. 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 MLMC method for the failure probability functional [12].

An additional comment on MLMC for failure probability is that there are difficulties estimating the variance $\mathrm{Var}\left[Y_\ell\right]$ 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 $\mathrm{Var}\left[Y_L\right]$ reliably using sample variance. See [8, 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 computable error bounds to reduce the expected cost to compute a realization of Q_ℓ. The idea is simple. Since the value of Q_ℓ for a specific realization is either 0 or 1 only depending on whether X_ℓ is greater than the critical value y or not, and independent of the distance $|y - X_\ell|$, we can allow large errors in X_ℓ when this distance is large, without affecting the value of Q_ℓ. The selective refinement idea is to initially solve all realizations to low accuracy in X and iteratively select those realizations with small distance to y for further refinement. This requires a method for computing an error bound of the quantity of interest.

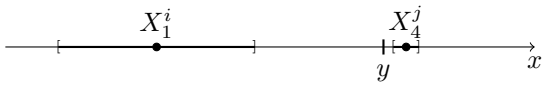


Figure 5: Selective refinement illustrated. Two realizations, one of X_1 and one of X_4 are depicted (dots) with error bounds (thick lines). Realization i is far from y and needs only be solved on mesh level 1 to yield an accurate value of Q^i, while realization j is close to y and requires a mesh on level 4 to yield an accurate value of Q^j.

To describe the method in detail, suppose we are provided with a level j-specific error bound E_j such that

$$|X_\ell - X_j| \leq E_j, \tag{15}$$

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 asymptotic cost rate $\mathcal{C}\left[Q_\ell^S\right] \leq C\gamma^{-(q-1)\ell}$ for

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 $Q_\ell^{S,i} = \chi_{(-\infty,y]}(X_j^i)$.

some constant C independent of ℓ (depending however, on the distribution F). Compared with the rate for $\mathcal{C}\left[Q_\ell\right]$ 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. Exactly how the error bounds E_j^i are computed is presented in Section 5.1.

It is often difficult to provide a guaranteed error bound E_j. However, as will be seen Section 5.1, in our case it is possible to give a probabilistic bound that holds with probability at least $1 - \alpha$, i.e. for any $\ell > j$

$$\Pr(|X_\ell - X_j| \leq E_j) \geq 1 - \alpha. \tag{16}$$

Here, α is the probability that the bound is broken and is preferably very small. The following lemma gives a bound on the error (a bias in this case) from using a probabilistic bound (16) instead of a guaranteed bound (15).

Lemma 2 (Bound of error introduced by replacing guaranteed error bound by probabilistic bound)**.** *Suppose E_j satisfies* (16) *for some $0 < \alpha < 1$ and Algorithm 1 defines Q_ℓ^S. Then,*

$$\mathbb{E}\left[|Q_\ell^S - Q_\ell|\right] \leq \ell\alpha.$$

Proof. Denote the event that the bound is broken for some $j < \ell$ by $A_\ell = \bigcup_{0 \le j < \ell} \{|X_\ell - X_j| > E_j\}$, so that $\Pr(A_\ell) \le \sum_{j=0}^{\ell-1} \Pr(|X_\ell - X_j| > E_j) \le \ell\alpha$. Using a probabilistic bound (16) in place of (15) to compute Q_ℓ^S introduces the bias,

$$\mathbb{E}\left[|Q_\ell^S - Q_\ell|\right] = \mathbb{E}\left[|Q_\ell^S - Q_\ell| \mid A_\ell\right] \Pr(A_\ell) + \mathbb{E}\left[|Q_\ell^S - Q_\ell| \mid \overline{A_\ell}\right] \Pr(\overline{A_\ell}) \le \ell\alpha,$$

where we denote the complement of A_ℓ by $\overline{A_\ell}$. □

We consider it acceptable that the error introduced by using the probabilistic bound is of the same size TOL as the other two error sources (statistical and numerical error). Then the bias on the last level L should be of comparable size to the numerical bias and stochastic error, i.e. $L\alpha \le \text{TOL}$. This means we don't necessarily need a guaranteed error bound, as long as the probability α for breaking the bound is less than $L^{-1}\text{TOL}$. In the experiment in Section 5.1, we determine α for a choice of a probabilistic error bound specific to our problem.

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

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

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

5. Preparatory experiments

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 a computable error bound E_j. MLMC requires a cost model and variance estimates of the correctors $\text{Var}[Y_\ell]$ to determine N_ℓ in (14). Before evaluating the four setups, we here present experiments performed to determine an approximation of E_j for selective refinement and estimate variances and cost models for the MLMC method. Note, all experiments presented in the section apply to the high variance case ($\sigma = 3$).

5.1. Error bound for sweep efficiency

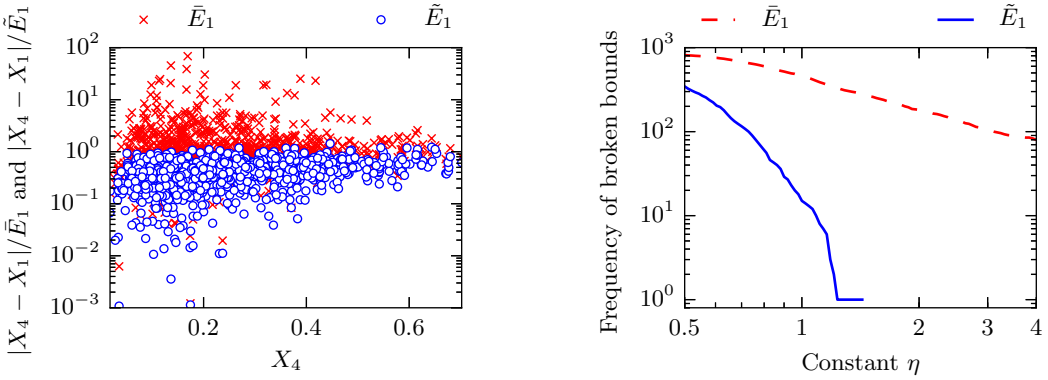
This section motivates a choice of E_j that satisfies the bound in (15) with probability $1 - \alpha$ (i.e. inequality (16)). Starting with the absolute difference in sweep efficiency between two consecutive levels, we define $\bar{E}_j$ and its upper bound $\tilde{E}_j$:

$$\begin{aligned} |X_j - X_{j-1}| &= \left| \int_{\mathcal{D}} \chi_{[0,c)}(s_j) - \chi_{[0,c)}(s_{j-1}) \, \mathrm{d}x \right| = \bar{E}_j \\ &\le \int_{\mathcal{D}} \left| \chi_{[0,c)}(s_j) - \chi_{[0,c)}(s_{j-1}) \right| \, \mathrm{d}x = \tilde{E}_j. \end{aligned}$$

In other words, $\bar{E}_j$ is the (absolute) difference in sweep efficiency for two consecutive levels $j-1$ and j. This error indicator allows for cancellation of errors from different parts of the domain; an erroneously swept area (negative contribution) can be cancelled by an erroneously non-swept area (positive contribution). Moving the absolute value inside the integral does not permit this kind of cancellation and gives the more pessimistic bound $\tilde{E}_j$, where any erroneously swept area, regardless of sign, is contributes to the error indicator. 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 scalar parameter to be determined.

For a sample of size 1000, we computed $X_1, \ldots, X_4, \tilde{E}_1$, and $\bar{E}_1$. The ratio between the approximate error and the error bounds $|X_4 - X_1|/\tilde{E}_1$ and $|X_4 - X_1|/\bar{E}_1$ were computed and plotted in Figure 6a as functions of X_4. This figure shows as how well the error bounds $\tilde{E}_1$ and $\bar{E}_1$ bound the actual error. If a point appears above 1 on the y-axis, the bound was broken. It is apparent that $\bar{E}_1$ is more likely to be broken than $\tilde{E}_1$.

We also counted number of samples for which the bounds $|X_4 - X_1| \le \eta\tilde{E}_1$ and $|X_4 - X_1| \le \eta\bar{E}_1$ are broken for values of η in the range $0.5 \le \eta \le 4$. The occurrence as function of η is shown in Figure 6b. (The corresponding experiments for $|X_2 - X_1| \le \eta\tilde{E}_1$ and $|X_3 - X_1| \le \eta\tilde{E}_1$ 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 $\tilde{E}_1$



(a) Scatter plot of ratio between approximate error and error bound versus the "exact" value X_4.

(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 6: Results from experiment to determine error bound and η.

were estimated to 0.028 and 0.056, respectively. It is thus clear from the figures and those mean values that $\tilde{E}_1$ is more suitable as an error bound, since a much larger constant η must be used to get the same error frequency with $\bar{E}_1$ as with $\tilde{E}_1$.

Guided by Figure 6b, we see that for $\eta = 2$ the error bound was not broken for a single realization. The behavior 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 $\alpha \approx 10^{-3}$. According to Lemma 2 this causes an additional bias of at most $L\alpha = 4 \cdot 10^{-3}$ to the total error. As a side note, the computational cost for the error bound $\tilde{E}_j$ is proportional to that of the quantity of interest X_j, since $\tilde{E}_j$ is computed only using the two consecutive X_{j-1} and X_j.

There are more sophisticated and rigorously motivated methods for constructing error bounds within the theory of a posteriori error estimation for quantities of interests. See e.g. [2, 32] for text books on a posteriori error analysis and [24, 26] for two works focusing on estimating errors of goal functionals.

5.2. Variance estimates and cost models

Variance estimates of Q_0 and Y_ℓ 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[1] for the variance were computed for $\ell \geq 1$. The results are presented in Figure 7a. The dotted line is the graph of $C\gamma^\ell$ 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 number of degrees of freedom increases by a factor 2^3 for every level due to doubling the number of cells in every dimension (two space dimensions and one time dimension). Also, the cost is typically linear in the number of degrees of freedom for approximations yielding sparse matrices. For the range of meshes used, it was verified experimentally that the computational cost scales approximately as $2^{3\ell}$. See Figure 4 for a graph of measurements of the computational cost (using a typical computational resource) together with the cost model. Note that we

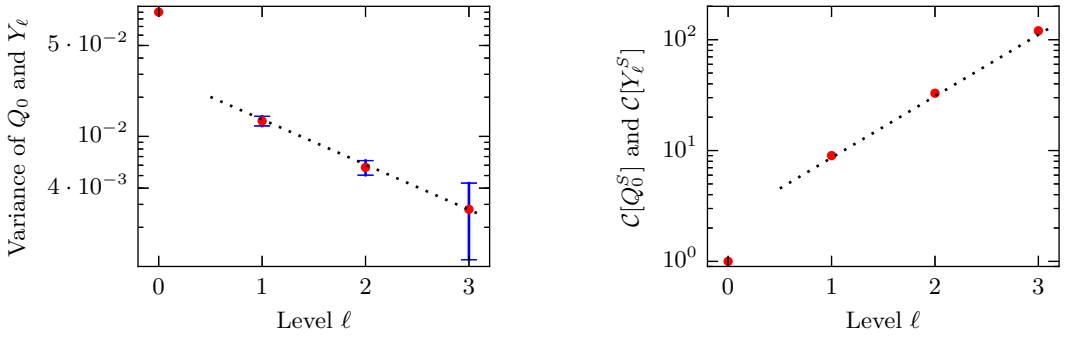
[1] $\mathrm{Var}[Y_\ell]$ is bounded and well-approximated by $\mathbb{E}[|Y_\ell|]$ (with an error less than $\mathbb{E}[|Y_\ell|]^2$). For our experiment, presented in Figure 7a, $\mathbb{E}[|Y_\ell|]$ and $\mathrm{Var}[Y_\ell]$ are small (approx. 10^{-2}), and this approximation is appropriate to use. Thus, we presented the mean estimates and 95% confidence intervals for these means (assuming normality) for the Bernoulli random variables $|Y_\ell|$.

Table 4: Summary of variance estimates and estimated cost models based on experiment in Section 5.2.

Quantity	Estimated value or model
$\mathrm{Var}\left[Q_0\right] = \mathrm{Var}\left[Q_0^S\right]$	0.0401
$\mathrm{Var}\left[Y_\ell\right] = \mathrm{Var}\left[Y_\ell^S\right]$	$0.025 \cdot \gamma^\ell$
$\mathcal{C}\left[Q_0\right] = \mathcal{C}\left[Q_0^S\right]$	1 (this defines unit time)
$\mathcal{C}\left[Q_\ell\right]$	$2^{3\ell}$
$\mathcal{C}\left[Y_\ell\right]$	$(2^3+1)\cdot 2^{3(\ell-1)} \approx 1.125 \cdot \gamma^{-4.8\ell}$
$\mathcal{C}\left[Q_\ell^S\right] = \mathcal{C}\left[Y_\ell^S\right]$	$2.4 \cdot \gamma^{-3.1\ell}$

have defined unit time as the expected cost on level 0, i.e. $\mathcal{C}\left[Q_0\right] = 1$. This time was measured to about 0.2 s for a typical computational resource. We also note that 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 $\mathcal{C}\left[Y_\ell\right] = \mathcal{C}\left[Q_\ell\right] + \mathcal{C}\left[Q_{\ell-1}\right]$. The cost 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 $\mathcal{C}\left[Y_\ell^S\right] = \mathcal{C}\left[Q_\ell^S\right] = \mathcal{C}\left[Q_0\right] + \mathcal{C}\left[Q_1\right] + \ldots + \mathcal{C}\left[Q_J\right]$ for some $J \leq \ell$ where J is random. This comes directly from Algorithm 1, where we always start by computing X_0 (note that $\mathcal{C}\left[X_\ell\right] = \mathcal{C}\left[Q_\ell\right]$), and then X_1, X_2 and so on to level at most ℓ. For realizations with quantity of interest far from the critical value y, we can terminate the loop in the selective refinement algorithm earlier, on some level $J \leq \ell$. The depth J is, however, random since it depends on the specific realization. We determine $\mathcal{C}\left[Y_\ell^S\right]$ by an experiment described below.



(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 of $C\gamma^\ell$ 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 of $C\gamma^{-q\ell}$ with $C = 2.4$, $\gamma = 0.65$ and $q = 3.1$.

Figure 7: Results from experiment in Section 6. Note that the point on level 0 does not correspond to a corrector and is thus not expected to obey the asymptotic relation.

The mean computational cost for selective refinement $\mathcal{C}\left[Q_0^S\right]$ and $\mathcal{C}\left[Y_\ell^S\right]$ at each level $\ell = 1, 2$ and 3 were computed based on the same samples as the variance estimates. The selective refinement algorithm was used and the average distribution of mesh levels was computed. Then, based on the average distribution of mesh levels, the cost model for $\mathcal{C}\left[Q_\ell\right]$ was used to compute a mean computational cost for the selective refinement algorithm. The results are shown in Figure 7b. Based on this experiment, the cost model for the correctors was chosen as $\mathcal{C}\left[Y_\ell^S\right] = C\gamma^{-q\ell}$ with $C = 2.4$ and $q = 3.1$ (where $\gamma = 0.65$ from the variance experiment). All variance estimates and estimated cost models are summarized in Table 4. Selective refinement was used to do this both the variance and the cost model experiment, whose total cost was $2.32 \cdot 10^6$.

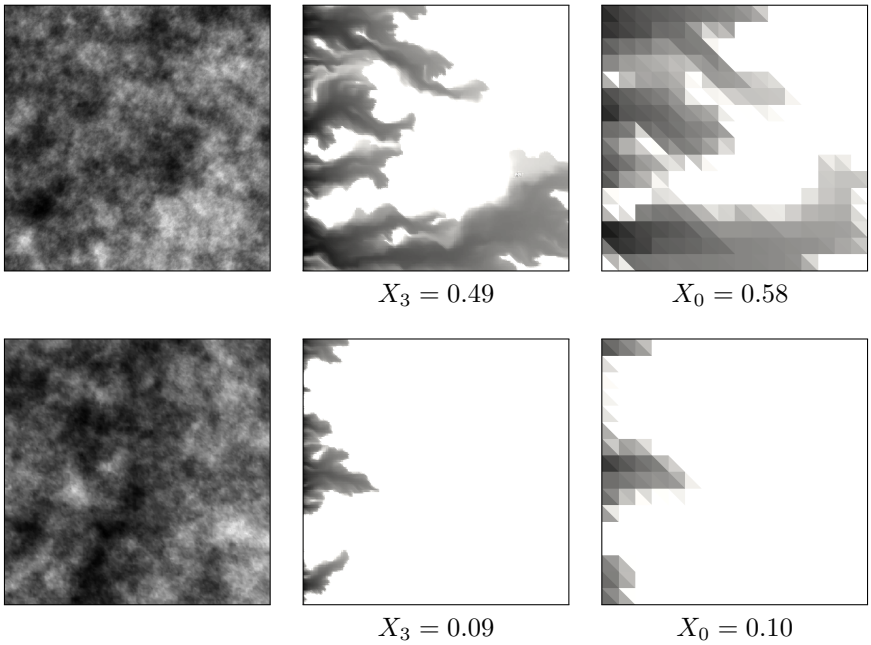


Figure 8: Each row corresponds to one realization of the permeability field k (black is low permeability, white is high). Left: Permeability field. Middle: Saturation at $T = 0.2$ (fine mesh, $h = 2^{-7}$). Right: Saturation at $T = 0.2$ (coarse mesh, $h = 2^{-4}$). The value of the quantity of interest X is given below the saturation solutions.

6. Quantification of computational cost

In this section, we quantify the computational costs required by the four methods reviewed above applied to our particular problem.

For illustrative purposes, Figure 8 shows two examples of a permeability realization and the corresponding non-wetting saturation solution at $T = 0.2$ computed using two different meshes. In the standard MC method, only the finest meshes (level 4, $h = 2^{-8}$) are used, regardless of the value of the quantity of interest. In the selective refinement algorithm, the upper realization in the figure is so far from being a failure ($X_0 = 0.58$ and critical value $y = 0.08$) that it will be solved on a coarse mesh, probably a level 1 mesh. The lower realization is close to the critical value and needs probably to be solved on the finest mesh to classify it accurately as either a failure or a success.

We pick TOL $= 0.01$ and evaluate the four setups: Monte Carlo (MC), Monte Carlo with selective refinement (MC-SR), multilevel Monte Carlo (MLMC), and multilevel Monte Carlo with selective refinement (MLMC-SR). The number of levels $L = 4$ is already fixed, and we aim at obtaining estimators of $\mathbb{E}[Q_L]$ with variance $\frac{1}{2}\text{TOL}^2 = 0.5 \cdot 10^{-4}$.

6.1. Monte Carlo

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

6.2. Monte Carlo with selective refinement

Based on the same sample as for the MC method, we computed $\widehat{Q}^{\text{MC,S}}_{N,L}$ using the error bound $\eta \tilde{E}_k$ presented in Section 5.1. Note that $\tilde{E}_0$ 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 $\tilde{E}_1$.

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

Table 5: Distribution of simulations over mesh hierarchy level. Note that a realization can be solved at several meshes for MLMC and selective refinement, i.e. it can be counted in several ℓ-columns.

Estimator ($\widehat{Q}$)	Number of simulations					
	$\ell = 0$	$\ell = 1$	$\ell = 2$	$\ell = 3$	$\ell = 4$	Realization count
MC ($\widehat{Q}^{\mathrm{MC}}_{N,L}$)	0	0	0	0	2000	2000
MC-SR ($\widehat{Q}^{\mathrm{MC,S}}_{N,L}$)	2000	2000	602	320	181	2000
MLMC ($\widehat{Q}^{\mathrm{ML}}_{\{N_\ell\},L}$)	38 734	8714	2484	709	158	41 376
MLMC-SR ($\widehat{Q}^{\mathrm{ML,S}}_{\{N_\ell\},L}$)	19 654	5084	587	104	18	19 654

6.3. Multilevel Monte Carlo

With TOL $= 0.01$ we used the cost model for standard refinement (i.e. $\mathcal{C}\left[Q_0\right]$ and $\mathcal{C}\left[Y_\ell\right]$) to compute the sample sizes N_ℓ in (14) (rounded up) for each level. This resulted in $(N_\ell)_{\ell=0}^4 = (31\,953, 6781, 1933, 551, 158)$. We realized $\widehat{Q}^{\mathrm{ML}}_{\{N_\ell\},L}$ using the computed sample sizes for the different levels. The total cost for this estimator was estimated using the cost model for standard refinement in Table 4 to $\mathcal{C}\left[\widehat{Q}^{\mathrm{ML}}_{\{N_\ell\},L}\right] = 1.28 \cdot 10^6$.

6.4. Multilevel Monte Carlo with selective refinement

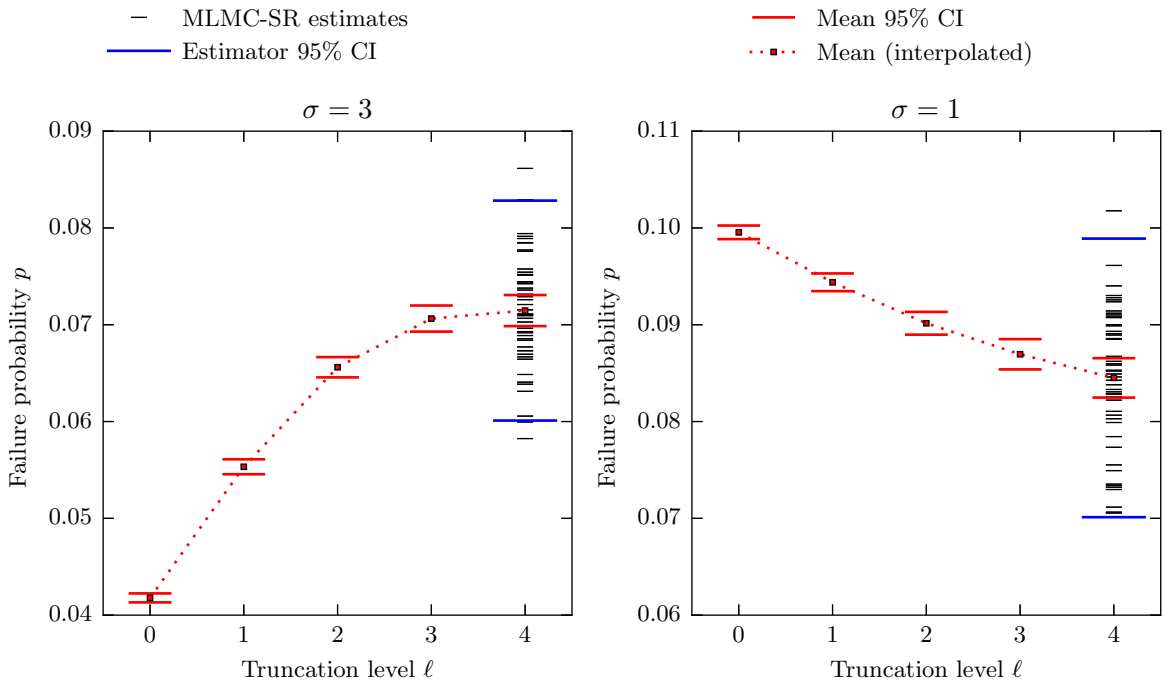
With TOL $= 0.01$ we used the cost model for selective refinement (i.e. $\mathcal{C}\left[Q_0^S\right]$ and $\mathcal{C}\left[Y_\ell^S\right]$) to compute the sample sizes N_ℓ in (14) (rounded up) for each level. This resulted in $(N_\ell)_{\ell=0}^4 = (14\,570, 3071, 1270, 525, 218)$. We realized $\widehat{Q}^{\mathrm{ML,S}}_{\{N_\ell\},L}$ using the error bound $\eta\tilde{E}_k$ presented in Section 5.1. The total cost (including the cost to compute error bounds) for this estimator was estimated using the cost model in Table 4 to $\mathcal{C}\left[\widehat{Q}^{\mathrm{ML,S}}_{\{N_\ell\},L}\right] = 2.65 \cdot 10^5$.

6.5. Summary of simulation results

The performance of the four estimators $\widehat{Q}$ can be compared by examining the product $\mathcal{C}\left[\widehat{Q}\right] \cdot \mathrm{Var}\left[\widehat{Q}\right]$. Given a deepest level L, this product is constant for all methods under 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 in the four previous sections, this product is presented in Table 6 for each setup.

Table 5 shows how many simulations were performed for each mesh hierarchy level. We comment each row. For MC, all realizations are solved at the finest mesh. For MC-SR, all realizations are initially solved at level 0 and 1 to construct the first error bound. It is then refined to a sufficient level according to the selective refinement algorithm. (Note that the exact distribution depends on the sample and is hence random. The row shows the distribution for one particular sample.) For MLMC, the distribution is determined by (14). For MLMC-SR, the distribution is determined by (14), but all realizations are initially solved at level 0 and are gradually refined. (Note that the exact distribution depends on the sample and is hence random. The row shows the distribution for one particular sample.) Common to all methods except MC is that the computational effort is shifted (to different extent) towards simulating on coarse instead of fine meshes.

To determine the variance of the MLMC and MLMC-SR estimators, a sample of 50 estimates was computed, and the sample variance of these 50 estimates was computed. The values of the 50 estimates for the MLMC-SR estimator are plotted at $\ell = L = 4$ in Figure 9 (for both the high and low variance case) together with a 95% confidence interval for the estimator. Additionally, for all $\ell = 0, \ldots, 4$, the mean values of the 50 truncated estimators (i.e. where only the first ℓ correctors in (13) are included) are plotted in the figure together with a 95% confidence interval for these mean values. In order to compute confidence intervals, 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 $\mathrm{TOL}^{1-\epsilon}$ for some $0 < \epsilon < 1$ while the statistical error decreases as TOL when TOL $\to 0$ (see [8, Lemma 7.1]). For this particular experiment, a Q–Q-plot was used to verify that the estimator was close to normally distributed.



(a) High variance case.

(b) Low variance case.

Figure 9: Statistics of truncated MLMC-SR estimator versus truncation level. 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 $\ell = 0, \ldots, 4$ are the mean values of the 50 level ℓ-truncated estimates. Bars (red) indicate the 95% CI for these mean values.

Table 6: Expected cost estimates, variance estimates and their product for the four setups in the high variance case ($\sigma = 3$).

Estimator ($\widehat{Q}$)	$\mathcal{C}\left[\widehat{Q}\right]$	$\mathrm{Var}\left[\widehat{Q}\right]$	$\mathcal{C}\left[\widehat{Q}\right] \cdot \mathrm{Var}\left[\widehat{Q}\right]$	Normalized cost
MC ($\widehat{Q}^{\mathrm{MC}}_{N,L}$)	$8.19 \cdot 10^6$	$3.49 \cdot 10^{-5}$	286	33.4
MC-SR ($\widehat{Q}^{\mathrm{MC,S}}_{N,L}$)	$1.00 \cdot 10^6$	$3.49 \cdot 10^{-5}$	34.9	4.08
MLMC ($\widehat{Q}^{\mathrm{ML}}_{\{N_\ell\},L}$)	$1.28 \cdot 10^6$	$2.62 \cdot 10^{-5}$	33.5	3.91
MLMC-SR ($\widehat{Q}^{\mathrm{ML,S}}_{\{N_\ell\},L}$)	$2.65 \cdot 10^5$	$3.23 \cdot 10^{-5}$	8.56	1.00

6.6. Summary of simulation results for low variance

All experiments were also performed for a random permeability field with $\sigma = 1$. In this case, the critical value was chosen as $y = 0.175$. This makes the sought failure probability similar (around 5–10%) to the high variance case. The procedure for this experiment is the same as for the high variance case: a computable error bound E_k was established ($E_k = \eta \tilde{E}_k$ with $\eta = 2$ was used), variance estimates and cost models for the correctors were constructed and used to compute N_ℓ for the MLMC methods. The final results are presented in Table 7 and we can see that they are comparable with the results for the high variance case in Table 6. Figure 9 shows statistics of the truncated MLMC-SR estimator to illustrate how the statistical and numerical errors are balanced.

7. Summary and discussion

The obtained products $\mathcal{C}\left[\widehat{Q}\right] \cdot \mathrm{Var}\left[\widehat{Q}\right]$ normalized with respect to the cheapest setup for all combinations are presented in the last column of Table 6 and 7 for both the high and low variance cases. 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

Table 7: Expected cost estimates, variance estimates and their product for the four setups in the low variance case ($\sigma = 1$).

Estimator ($\widehat{Q}$)	$\mathcal{C}\left[\widehat{Q}\right]$	$\mathrm{Var}\left[\widehat{Q}\right]$	$\mathcal{C}\left[\widehat{Q}\right] \cdot \mathrm{Var}\left[\widehat{Q}\right]$	Normalized cost
MC ($\widehat{Q}^{\mathrm{MC}}_{N,L}$)	$8.19 \cdot 10^6$	$4.20 \cdot 10^{-5}$	344	61.5
MC-SR ($\widehat{Q}^{\mathrm{MC,S}}_{N,L}$)	$7.96 \cdot 10^5$	$4.20 \cdot 10^{-5}$	33.4	5.98
MLMC ($\widehat{Q}^{\mathrm{ML}}_{\{N_\ell\},L}$)	$4.94 \cdot 10^5$	$6.10 \cdot 10^{-5}$	30.1	5.38
MLMC-SR ($\widehat{Q}^{\mathrm{ML,S}}_{\{N_\ell\},L}$)	$1.08 \cdot 10^5$	$5.18 \cdot 10^{-5}$	5.59	1.00

Carlo and selective refinement gives further gains, in total a factor of one to two orders of magnitude, for both the low and high variance cases.

It is interesting to note that the savings for MLMC-SR compared to MC in the low variance case is twice that of the savings in the high variance case. The savings depend on two factors that might cancel or reinforce each other depending on the problem: (i) the probability mass "close to" the critical value y and (ii) the size of the computable error bounds E_j. The more probability mass there is "close to" y, the more realizations need to be solved on a fine mesh. However, it is the size of the computable error bounds that defines what is considered "close to" y: small error bounds reduce the probability for realizations to be refined to the finest mesh. Exactly how these two factors affect the cost of the method can be seen in the proof of Theorem 5.2 in [12], where the cost rate for the selective refinement method is derived. We stress that the computational savings cannot in general be expressed as a function of the variance of the random permeability data alone.

The obtained 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 TOL $= 0.01$ and the number of samples were chosen based on that. Figure 9 suggests that the numerical bias decreases and that the expected value converges as more correctors (or levels) are added. The statistical error (as a 95% confidence interval) is given by the blue bars for level $\ell = 4$. It is clear from Figure 9 that there is a large numerical bias for low values of ℓ. However, the bias improvement obtained by adding additional refinement levels is dominated by the statistical error for both the high and low variance case. A better balance between the two sources of error at this level L would be attained if TOL was chosen smaller, i.e. if a more accurate solution was desired. Since for a given L, the relative costs of the methods are independent of TOL, the relative cost savings for a more accurate solution remain as presented in Table 6 and 7 also for smaller values of TOL, if L remains constant.

In addition to the cost to realize an estimator, the cost for preparatory work (see Section 5) to construct the estimators differs between the setups. For selective refinement we established the reliability of an error bound for sweep efficiency by numerical experiments in Section 5.1. The exact method for computing the error bound is problem specific and needs to be reconsidered for other problems. As was mentioned previously, error bounds can be constructed by more sophisticated a posteriori error estimation methods. Then this additional preparatory computational work to establish reliability might not be necessary. For multilevel Monte Carlo, we performed a preparatory experiment in Section 5.2 to determine the variance of the correctors and the cost model for selective refinement.

Although this work was performed for a specific problem, where we sought the probability for the sweep efficiency to be less than 8%, the methods presented are applicable to a wide variety of problems. For the multilevel Monte Carlo method to be applicable, we need a hierarchy of at least two approximation levels in a regime where the error of the quantity of interest converges. For selective refinement to be applicable, we additionally need computable error bounds of the quantity of interest. 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.

Acknowledgment

The authors are grateful to Andreas Hellander and Salman Toor for assistance with computational parallelization tools (MOLNs [10]) and resources (SNIC Smog Cloud). The authors thank the sponsors of this research: EU FP7 project TRUST, Centre for Interdisciplinary Mathematics, Uppsala University and the Swedish Research Council.

References

[1] R. Avikainen. On irregular functionals of SDEs and the Euler scheme. *Finance Stoch.*, 13(3):381–401, 2009.

[2] W. Bangerth and R. Rannacher. *Adaptive finite element methods for differential equations*. Lectures in Mathematics. ETH Zürich. Birkhäuser, Basel, 2003.

[3] 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.

[4] R. H. Brooks and A. T. Corey. *Hydraulic Properties of Porous Media*. Colorado State University Hydrology Papers. Colorado State University, Fort Collins, 1964.

[5] 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.

[6] 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.

[7] 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.

[8] 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.

[9] 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.

[10] B. Drawert, M. Trogdon, S. Toor, L. Petzold, and A. Hellander. MOLNs: A cloud appliance for interactive, reproducible and scalable spatial stochastic computational experiments. *SIAM J. Sci. Comput.*, to appear.

[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(1):826–850, 2014.

[12] D. Elfverson, F. Hellman, and A. Målqvist. A multilevel Monte Carlo method for computing failure probabilities. *SIAM/ASA J. Uncertain. Quantif.*, 4(1):312–330, 2016.

[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. Multilevel monte carlo methods. In J. Dick, F. Y. Kuo, G. W. Peters, and I. H. Sloan, editors, *Monte Carlo and Quasi-Monte Carlo Methods 2012*, volume 65 of *Springer Proceedings in Mathematics & Statistics*, pages 83–103. Springer Berlin Heidelberg, 2013.

[17] 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.

[18] A.-L. Haji-Ali, F. Nobile, and R. Tempone. Multi-index Monte Carlo: when sparsity meets sampling. *Numer. Math*, pages 1–40, 2015.

[19] 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.

[20] 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.

[21] L. W. Lake. *Enhanced oil recovery*. Prentice Hall, New Jersey, 1989.

[22] R. J. LeVeque. *Finite volume methods for hyperbolic problems*. Cambridge University Press, New York, 2002. Cambridge Texts in Applied Mathematics.

[23] A. Niemi, J. Bensabat, V. Shtivelman, K. Edlmann, P. Gouze, L. Luquot, F. Hingerl, S. M. Benson, P. A. Pezard, K. Rasmusson, T. Liang, F. Fagerlund, M. Gendler, I. Goldberg, A. Tatomir, T. Lange, M. Sauter, and B. Freifeld. Heletz experimental site overview, characterization and data analysis for CO2 injection and geological storage. *International Journal of Greenhouse Gas Control*, 2016.

[24] J. T. Oden and S. Prudhomme. Goal-oriented error estimation and adaptivity for the finite element method. *Comput. Math. Appl.*, 41(5-6):735–756, 2001.

[25] 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.

[26] S. Prudhomme, J. T. Oden, T. Westermann, J. Bass, and M. E. Botkin. Practical methods for a posteriori error estimation in engineering applications. *Int. J. Numer. Meth. Engng*, 56:1193–1224, 2003.

[27] A. Shafeen, E. Croiset, P. L. Douglas, and I. Chatzis. CO2 sequestration in Ontario, Canada. Part I: storage evaluation of potential reservoirs. *Energy Conversion and Management*, 45(17):2645–2659, 2004.

[28] S. Tanaka, H. Koide, and A. Sasagawa. Possibility of underground CO2 sequestration in Japan. *Energy Conversion and Management*, 36(6-9):527–530, 1995.

[29] 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.

[30] L. Tian, Z. Yang, F. Fagerlund, and A. Niemi. Effects of permeability heterogeneity on CO2 injectivity and storage efficiency coefficient. *Greenhouse Gases: Science and Technology*, 6(1), 2016.

[31] C.-F. Tsang, C. Doughty, and M. Uchida. Simple model representations of transport in a complex fracture and their effects on long-term predictions. *Water Resources Research*, 44(8), 2008.

[32] R. Verfürth. *A review of a posteriori error estimation and adaptive mesh-refinement techniques*. Advances in numerical mathematics. Wiley-Teubner, 1996.

[33] Z. Yang, L. Tian, A. Niemi, and F. Fagerlund. Upscaling of the constitutive relationships for CO2 migration in multimodal heterogeneous formations. *International Journal of Greenhouse Gas Control*, 19:743–755, 2013.