Available online at www.sciencedirect.com
SCIENCE DIRECT
Journal of Computational Physics 217 (2006) 123–142
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

Coarse-gradient Langevin algorithms for dynamic data integration and uncertainty quantification

P. Dostert [a], Y. Efendiev [a], T.Y. Hou [b,*], W. Luo [b]

[a] *Department of Mathematics, Texas A&M University, College Station, TX 77843-3368, United States*
[b] *Applied Mathematics, Caltech, Pasadena, CA 91125, United States*

Received 31 August 2005; received in revised form 9 March 2006; accepted 13 March 2006
Available online 22 May 2006

Abstract

The main goal of this paper is to design an efficient sampling technique for dynamic data integration using the Langevin algorithms. Based on a coarse-scale model of the problem, we compute the proposals of the Langevin algorithms using the coarse-scale gradient of the target distribution. To guarantee a correct and efficient sampling, each proposal is first tested by a Metropolis acceptance–rejection step with a coarse-scale distribution. If the proposal is accepted in the first stage, then a fine-scale simulation is performed at the second stage to determine the acceptance probability. Comparing with the direct Langevin algorithm, the new method generates a modified Markov chain by incorporating the coarse-scale information of the problem. Under some mild technical conditions we prove that the modified Markov chain converges to the correct posterior distribution. We would like to note that the coarse-scale models used in the simulations need to be inexpensive, but not necessarily very accurate, as our analysis and numerical simulations demonstrate. We present numerical examples for sampling permeability fields using two-point geostatistics. Karhunen–Loève expansion is used to represent the realizations of the permeability field conditioned to the dynamic data, such as the production data, as well as the static data. The numerical examples show that the coarse-gradient Langevin algorithms are much faster than the direct Langevin algorithms but have similar acceptance rates.

Keywords: Langevin; MCMC; Multiscale; Uncertainty; Porous media; Two-Phase flow; History matching

1. Introduction

Uncertainties in the detailed description of reservoir lithofacies, porosity, and permeability are major contributors to uncertainty in reservoir performance forecasting. Making decisions in reservoir management requires a method for quantifying uncertainty. Large uncertainties in reservoirs can greatly affect the production and decision making on well drilling. Better decisions can be made by reducing the uncertainty. Thus, quantifying and reducing the uncertainty is an important and challenging problem in subsurface

* Corresponding author. Tel.: +1 626 395 4546; fax: +1 626 578 0124.
E-mail address: hou@ama.caltech.edu (T.Y. Hou).

0021-9991/$ - see front matter © 2006 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2006.03.012

modeling. Additional dynamic data, such as the production data, can be used in achieving more accurate predictions. The previous findings show that dynamic data can be used to improve the predictions and reduce the uncertainty. Therefore, to predict future reservoir performance, the reservoir properties, such as porosity and permeability, need to be conditioned to dynamic data. In general it is difficult to calculate this posterior probability distribution because the process of predicting flow and transport in petroleum reservoirs is nonlinear. Instead, we estimate this probability distribution from the outcomes of flow predictions for a large number of realizations of the reservoir. It is essential that the permeability (and porosity) realizations adequately reflect the uncertainty in the reservoir properties, i.e., we correctly sample this probability distribution.

The prediction of permeability fields based on dynamic data is a challenging problem because permeability fields are typically defined on a large number of grid blocks. The Markov chain Monte Carlo (MCMC) method and its modifications have been used previously to sample the posterior distribution of the permeability field. Oliver et al. [21,22] proposed the randomized maximum likelihood method, which generates unconditional realizations of the production and permeability data and then solves a deterministic gradient-based inverse problem. The solution of this minimization problem is taken as a proposal and accepted with probability one because the rigorous acceptance probability is very difficult to estimate. In addition to the need of solving a gradient-based inverse problem, this method does not guarantee a proper sampling of the posterior distribution. Developing efficient and rigorous MCMC calculations with high acceptance rates remains a challenging problem.

In this paper, we employ the Langevin algorithms within the context of MCMC methods for sampling the permeability field. Langevin algorithms provide efficient sampling techniques because they use the gradient information of the target distributions. However, the direct Langevin algorithm is very expensive because it requires the computation of the gradients with fine-scale simulations. Based on a coarse-scale model of the problem, we propose an approach where the gradients are computed with inexpensive coarse-scale simulation. These coarse-scale gradients may not be very accurate and, for this reason, the computed results are first tested with coarse-scale distributions. If the result is accepted at the first stage, then a fine-scale simulation is performed at the second stage to determine the acceptance probability. The first stage of the method modifies the Markov chain generated by the direct Langevin algorithms. We can show that the modified Markov chain satisfies the detailed balance condition for the correct distribution. Moreover, we point out that the chain is ergodic and converges to the correct posterior distribution under some technical assumptions. The validity of the assumptions for our application is also discussed in the paper.

For sampling the permeability fields in two-phase flows, we use a coarse-scale model based on multiscale finite element methods. The multiscale finite element methods are used to construct coarse-scale velocity fields which are further used to solve the transport equation on the coarse-grid. The multiscale basis functions are not updated throughout the simulation, which provides an inexpensive coarse-scale methodology. In this respect, the multiscale finite element methods are conceptually similar to the single-phase flow upscaling methods (see e.g. [3,5]), where the main idea is to upscale the underlying fine-scale permeability field. These types of upscaling methods are not very accurate because the subgrid effects of transport are neglected. We would like to note that upscaled models are used in MCMC simulations in previous findings. In a pioneering work [12], Glimm and Sharp employed error models between coarse- and fine-scale simulations to quantify the uncertainty.

Numerical results for sampling permeability fields using two-point geostatistics are presented in the paper. Using the Karhunen–Loève expansion, we can represent the high dimensional permeability field by a small number of parameters. Furthermore, the static data (the values of permeability fields at some sparse locations) can be easily incorporated into the Karhunen–Loève expansion to further reduce the dimension of the parameter space. Imposing the values of the permeability at some locations restricts the parameter space to a subspace (hyperplane). Numerical results are presented for both single-phase and two-phase flows. In all the simulations, we show that the gradients of the target distribution computed using coarse-scale simulations provide accurate approximations of the actual fine-scale gradients. Furthermore, we present the uncertainty assessment of the production data based on sampled permeability fields. Our numerical results show that the uncertainty spread is much larger if no dynamic data information is used. However, the uncertainty spread decreases if more information is incorporated into the simulations.

The paper is organized as follows. In the next section, we briefly describe the model equations and their upscaling. Section 3 is devoted to the analysis of the Langevin MCMC method and its relevance to our particular application. Numerical results are presented in Section 4.

2. Fine and coarse models

In this section, we briefly introduce a coarse-scale model used in the simulations. We consider two-phase flows in a reservoir (denoted by Ω) under the assumption that the displacement is dominated by viscous effects; i.e., we neglect the effects of gravity, compressibility, and capillary pressure. Porosity will be considered to be constant. The two phases will be referred to as water and oil, designated by subscripts w and o, respectively. We write Darcy's law for each phase as follows:

$$v_j = -\frac{k_{\mathrm{rj}}(S)}{\mu_{\mathrm{j}}}\boldsymbol{k}\cdot\nabla p, \tag{2.1}$$

where v_{j} is the phase velocity, $\boldsymbol{k}$ is the permeability tensor, k_{rj} is the relative permeability to phase j (j = o, w), S is the water saturation (volume fraction) and p is pressure. Throughout the paper, we will assume that the permeability tensor is diagonal $\boldsymbol{k} = k\boldsymbol{I}$, where k is a scalar and $\boldsymbol{I}$ is the unit tensor. In this work, a single set of relative permeability curves is used. Combining Darcy's law with a statement of conservation of mass allows us to express the governing equations in terms of the so-called pressure and saturation equations:

$$\nabla\cdot(\lambda(S)k\nabla p) = h, \tag{2.2}$$

$$\frac{\partial S}{\partial t} + v\cdot\nabla f(S) = 0, \tag{2.3}$$

where λ is the total mobility, h is the source term, $f(S)$ is the flux function, and v is the total velocity, which are respectively given by:

$$\lambda(S) = \frac{k_{\mathrm{rw}}(S)}{\mu_{\mathrm{w}}} + \frac{k_{\mathrm{ro}}(S)}{\mu_{\mathrm{o}}}, \tag{2.4}$$

$$f(S) = \frac{k_{\mathrm{rw}}(S)/\mu_{\mathrm{w}}}{k_{\mathrm{rw}}(S)/\mu_{\mathrm{w}} + k_{\mathrm{ro}}(S)/\mu_{\mathrm{o}}}, \tag{2.5}$$

$$v = v_{\mathrm{w}} + v_{\mathrm{o}} = -\lambda(S)k\cdot\nabla p. \tag{2.6}$$

The above descriptions are referred to as the fine model of the two-phase flow problem. For the single-phase flow, $k_{\mathrm{rw}}(S) = S$ and $k_{\mathrm{ro}}(S) = 1 - S$.

In most upscaling procedures, the coarse-scale pressure equation is of the same form as the fine-scale Eq. (2.2) but with an equivalent grid block permeability tensor replacing the fine-scale permeability field (see e.g. [5]). In this work, the proposed coarse-scale model consists of the upscaling of the pressure equation (2.2) in order to obtain the velocity field on the coarse-grid and use it in (2.3) to resolve the saturation on the coarse-grid. The pressure equation is scaled up using a multiscale finite volume method. The multiscale finite volume method is similar to the recently introduced multiscale finite element methods. The details of the method are presented in Appendix A. Using the multiscale finite volume method, we obtain the coarse-scale velocity field, which is used in solving the saturation equation on the coarse-grid. Since no subgrid modeling is performed for the saturation equation, this upscaling procedure introduces errors. In Fig. 2.1, we plot a typical fractional flow comparison between fine- and coarse-scale models. Fractional flow $F(t)$ (denoted simply by F in further discussion) is defined as the fraction of oil in the produced fluid and is given by $q_{\mathrm{o}}/q_{\mathrm{t}}$, where $q_{\mathrm{t}} = q_{\mathrm{o}} + q_{\mathrm{w}}$, with q_{o} and q_{w} the flow rates of oil and water at the production edge of the model. More specifically,

$$F(t) = 1 - \frac{\int_{\partial\Omega^{\mathrm{out}}} f(S)v_{\mathrm{n}}\,\mathrm{d}l}{\int_{\partial\Omega^{\mathrm{out}}} v_{\mathrm{n}}\,\mathrm{d}l},$$

where $\partial\Omega^{\mathrm{out}}$ is outflow boundaries and v_{n} is the normal velocity field. In Fig. 2.1 the fractional flows are plotted against the dimensionless time pore volume injected (PVI). The PVI at time T is defined as $\frac{1}{V_{\mathrm{p}}}\int_0^T q_t(\tau)\,\mathrm{d}\tau$, with V_{p} the total pore volume of the system. PVI provides the dimensionless time for the displacement.

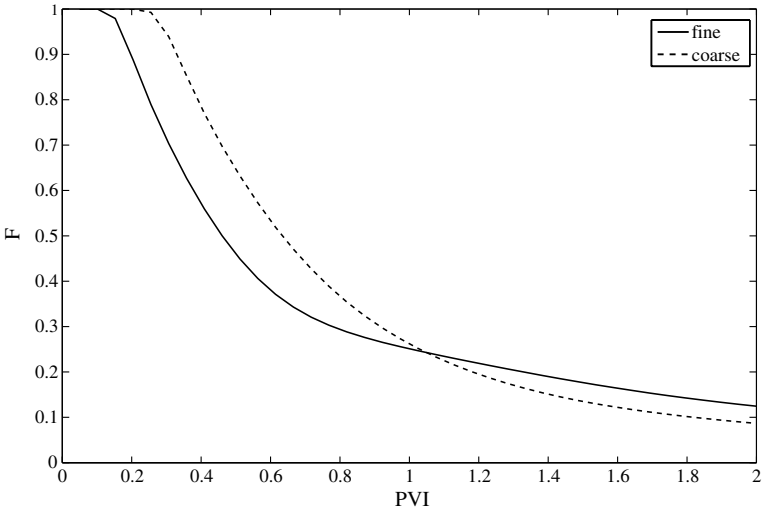

Fig. 2.1. Typical fine and coarse scale fractional flows.

3. Langevin algorithm using coarse-scale models

3.1. Problem setting

The problem under consideration consists of sampling the permeability field given fractional flow measurements. Typically, the prior information about the permeability field consists of its covariance matrix and the values of the permeability at some sparse locations. Since the fractional flow is an integrated response, the map from the permeability field to the fractional flow is not one-to-one. Hence, this problem is ill-posed in the sense that there exist many different permeability realizations for the given production data.

From the probabilistic point of view, this problem can be regarded as conditioning the permeability fields to the fractional flow data with measurement errors. Consequently, our goal is to sample from the conditional distribution $P(k|F)$, where k is the fine-scale permeability field and F is the fractional flow data. Using the Bayes formula we can write

$$P(k|F) \propto P(F|k)P(k). \tag{3.1}$$

In the above formula, $P(k)$ is the unconditioned (prior) distribution of the permeability field. In practice, the measured fractional flow contains measurement errors. In this paper, we assume that the measurement error satisfies a Gaussian distribution, thus, the likelihood function $P(F|k)$ takes the form

$$P(F|k) \propto \exp\left(-\frac{\|F - F_k\|^2}{\sigma_f^2}\right), \tag{3.2}$$

where F is the reference fractional flow, F_k is the fractional flow for the permeability field k, and σ_f is the measurement precision. In practice, F_k is computed by solving the nonlinear PDE system (2.1)–(2.3) for the given k on the fine-grid. Since both F and F_k are functions of time (denoted by t), the norm $\|F - F_k\|^2$ is defined as the L_2 norm

$$\|F - F_k\|^2 = \int_0^T [F(t) - F_k(t)]^2 \, dt.$$

Denote the sampling target distribution as

$$\pi(k) = P(k|F) \propto \exp\left(-\frac{\|F - F_k\|^2}{\sigma_f^2}\right) P(k). \tag{3.3}$$

Since different permeability fields may produce the same fractional flow curve, the distribution $\pi(k)$ is a function of k with multiple local maxima. Sampling from the distribution $\pi(k)$ can be accomplished by the MCMC method. For a given proposal distribution $q(y|x)$, the Metropolis–Hasting MCMC algorithm (see, e.g. [23, p. 233]) consists of the following steps.

Algorithm I. (Metropolis–Hasting MCMC, Robert and Casella [23])

- Step 1. At k_n generate Y from $q(Y|k_n)$.
- Step 2. Accept Y as a sample with probability

$$p(k_n, Y) = \min\left(1, \frac{q(k_n|Y)\pi(Y)}{q(Y|k_n)\pi(k_n)}\right), \tag{3.4}$$

i.e. take $k_{n+1} = Y$ with probability $p(k_n, Y)$, and $k_{n+1} = k_n$ with probability $1 - p(k_n, Y)$.

The MCMC algorithm generates a Markov chain $\{k_n\}$ whose stationary distribution is $\pi(k)$. A remaining question is how to choose an efficient proposal distribution $q(Y|k_n)$.

An important type of proposal distribution can be derived from the Langevin diffusion, as proposed by Grenander and Miller [13]. The Langevin diffusion is defined by the stochastic differential equation

$$\mathrm{d}k(\tau) = \frac{1}{2}\nabla \log \pi(k(\tau))\mathrm{d}\tau + \mathrm{d}W_\tau, \tag{3.5}$$

where W_τ is the standard Brownian motion vector with independent components. It can be shown that the diffusion process $k(\tau)$ has $\pi(k)$ as its stationary distribution. The actual implementation of the Langevin diffusion requires a discretization of Eq. (3.5),

$$k_{n+1} = k_n + \frac{\Delta\tau}{2}\nabla \log \pi(k_n) + \sqrt{\Delta\tau}\epsilon_n,$$

where ϵ_n are independent standard normal distributions. However, the discrete solution k_n can have vastly different asymptotic behavior from the continuous diffusion process $k(\tau)$ [23]. In general, the discrete solution k_n does not necessarily have $\pi(k)$ as its stationary distribution. Instead of taking k_n as samples directly, we use them as test proposals for Algorithm I. The samples will be further tested and corrected by the Metropolis acceptance–rejection step (3.4). Consequently, we choose the proposal generator $q(Y|k_n)$ in Algorithm I as

$$Y = k_n + \frac{\Delta\tau}{2}\nabla \log \pi(k_n) + \sqrt{\Delta\tau}\epsilon_n. \tag{3.6}$$

Since ϵ_n are independent Gaussian vectors, the transition distribution of the proposal generator (3.6) is

$$\begin{aligned} q(Y|k_n) &\propto \exp\left(-\frac{\|Y - k_n - \frac{\Delta\tau}{2}\nabla \log \pi(k_n)\|^2}{2\Delta\tau}\right), \\ q(k_n|Y) &\propto \exp\left(-\frac{\|k_n - Y - \frac{\Delta\tau}{2}\nabla \log \pi(Y)\|^2}{2\Delta\tau}\right). \end{aligned} \tag{3.7}$$

The scheme (3.6) can be regarded as a problem-adapted random walk. The gradient information of the target distribution is included to enforce a biased random walk. The use of the gradient information in inverse problems for subsurface characterization is not new. In their original work, Oliver et al. [21,22] developed the randomized maximum likelihood method, which uses the gradient information of the target distribution. This approach uses unconditional realizations of the production and permeability data and solves a deterministic gradient-based minimization problem. The solution of this minimization problem is taken as a proposal and is accepted with probability one, since the acceptance probability is very difficult to estimate. In addition to the need of solving a gradient-based inverse problem, this method does not guarantee a proper sampling of the posterior distribution. Thus, developing efficient and rigorous MCMC calculations with high acceptance rates remains a challenging problem. Though the Langevin formula (3.6) resembles the randomized maximum

likelihood method, it is more efficient and rigorous, and one can compute the acceptance probability easily. The Langevin algorithms also allow us to achieve high acceptance rates. However, computing the gradients of the target distribution is very expensive. In this paper, we propose to use the coarse-scale solutions in the computation of the gradients to speed up the Langevin algorithms.

3.2. Langevin MCMC method using coarse-scale models

The major computational cost of Algorithm I is to compute the value of the target distribution $\pi(k)$ at different permeabilities. Since the map between the permeability k and the fractional flow F_k is governed by the PDE system (2.1)–(2.3), there is no explicit formula for the target distribution $\pi(k)$. To compute the function $\pi(k)$, we need to solve the nonlinear PDE system (2.1)–(2.3) on the fine scale for the given k. For the same reason, we need to compute the gradient of $\pi(k)$ in (3.6) numerically (by finite differences), which involves solving the nonlinear PDE system (2.1)–(2.3) multiple times. To compute the acceptance probability (3.4), the PDE system (2.1)–(2.3) needs to be solved one more time. As a result, the direct (full) MCMC simulations with Langevin samples are prohibitively expensive.

To bypass the above difficulties, we design a coarse-grid Langevin MCMC algorithm where most of the fine scale computations are replaced by the coarse scale ones. Based on a coarse-grid model of the distribution $\pi(k)$, we first generate samples from (3.6) using the coarse-scale gradient of $\pi(k)$, which only requires solving the PDE system (2.1)–(2.3) on the coarse-grid. Then we further filter the proposals by an additional Metropolis acceptance–rejection test on the coarse-grid. If the sample does not pass the coarse-grid test, the sample is rejected and no further fine-scale test is necessary. The argument for this procedure is that if a proposal is not accepted by the coarse-grid test, then it is unlikely to be accepted by the fine scale test either. By eliminating most of the ''unlikely'' proposals with cheap coarse-scale tests, we can avoid wasting CPU time simulating the rejected samples on the fine-scale.

To model $\pi(k)$ on the coarse-scale, we define a coarse-grid map F_k^* between the permeability field k and the fractional flow F. The map F_k^* is determined by solving the PDE system (2.1)–(2.3) on a coarse-grid. Consequently, the target distribution $\pi(k)$ can be approximated by

$$\pi^*(k) \propto \exp\left(-\frac{\|F - F_k^*\|^2}{\sigma_c^2}\right) P(k), \tag{3.8}$$

where σ_c is the measurement precision on the coarse-grid, and should be slightly larger than σ_f. Then the Langevin samples are generated from (3.6) using the coarse-grid gradient of the target distribution

$$Y = k_n + \frac{\Delta\tau}{2}\nabla \log \pi^*(k_n) + \sqrt{\Delta\tau}\epsilon_n. \tag{3.9}$$

The transition distribution of the coarse-grid proposal (3.9) is

$$\begin{aligned} q^*(Y|k_n) &\propto \exp\left(-\frac{\|Y - k_n - \frac{\Delta\tau}{2}\nabla \log \pi^*(k_n)\|^2}{2\Delta\tau}\right), \\ q^*(k_n|Y) &\propto \exp\left(-\frac{\|k_n - Y - \frac{\Delta\tau}{2}\nabla \log \pi^*(Y)\|^2}{2\Delta\tau}\right). \end{aligned} \tag{3.10}$$

To compute the gradient of $\pi^*(k_n)$ numerically, we only need to solve the PDE system (2.1)–(2.3) on the coarse-grid. The coarse-scale distribution $\pi^*(k)$ serves as a regularization of the original fine-scale distribution $\pi(k)$. By replacing the fine-scale gradient with the coarse-scale gradient, we can reduce the computational cost dramatically but still direct the proposals to regions with larger probabilities.

Because of the high dimension of the problem and the discretization errors, most proposals generated by the Langevin algorithms (both (3.6) and (3.9)) will be rejected by the Metropolis acceptance–rejection test (3.4). To avoid wasting expensive fine-scale computations on unlikely acceptable samples, we further filter the Langevin proposals by the coarse-scale acceptance criteria

$$g(k_n, Y) = \min\left(1, \frac{q^*(k_n|Y)\pi^*(Y)}{q^*(Y|k_n)\pi^*(k_n)}\right),$$

where $\pi^*(k)$ is the coarse-scale target distribution (3.8), $q^*(Y|k_n)$ and $q^*(k_n|Y)$ are the coarse-scale proposal distributions given by (3.10). Combining all the discussion above, we have the following revised MCMC algorithm.

Algorithm II. (Preconditioned coarse-gradient Langevin algorithm)

- Step 1. At k_n, generate a trial proposal Y from the coarse Langevin algorithm (3.9).
- Step 2. Take the proposal k as

$$k = \begin{cases} Y \text{ with probability } g(k_n, Y), \\ k_n \text{ with probability } 1 - g(k_n, Y), \end{cases}$$

where

$$g(k_n, Y) = \min\left(1, \frac{q^*(k_n|Y)\pi^*(Y)}{q^*(Y|k_n)\pi^*(k_n)}\right).$$

Therefore, the proposal k is generated from the effective instrumental distribution

$$Q(k|k_n) = g(k_n, k)q^*(k|k_n) + \left(1 - \int g(k_n, k)q^*(k|k_n)\,\mathrm{d}k\right)\delta_{k_n}(k). \tag{3.11}$$

- Step 3. Accept k as a sample with probability

$$\rho(k_n, k) = \min\left(1, \frac{Q(k_n|k)\pi(k)}{Q(k|k_n)\pi(k_n)}\right), \tag{3.12}$$

i.e., $k_{n+1} = k$ with probability $\rho(k_n, k)$, and $k_{n+1} = k_n$ with probability $1 - \rho(k_n, k)$.

The step 2 screens the trial proposal Y by the coarse-grid distribution before passing it to the fine-scale test. The filtering process changes the proposal distribution of the algorithm from $q^*(Y|k_n)$ to $Q(k|k_n)$ and serves as a preconditioner to the MCMC method. This is why we call it the preconditioned coarse-gradient Langevin algorithm. We note that testing proposals by approximate target distributions is not a very new idea. Similar strategies have been developed previously in [17,2].

Note that there is no need to compute $Q(k|k_n)$ and $Q(k_n|k)$ in (3.12) by formula (3.11). The acceptance probability (3.12) can be simplified as

$$\rho(k_n, k) = \min\left(1, \frac{\pi(k)\pi^*(k_n)}{\pi(k_n)\pi^*(k)}\right). \tag{3.13}$$

In fact, this is obviously true for $k = k_n$ since $\rho(k_n, k_n) \equiv 1$. For $k \neq k_n$,

$$Q(k_n|k) = g(k, k_n)q(k_n|k) = \frac{1}{\pi^*(k)}\min(q(k_n|k)\pi^*(k), q(k|k_n)\pi^*(k_n)) = \frac{q(k|k_n)\pi^*(k_n)}{\pi^*(k)}g(k_n, k) = \frac{\pi^*(k_n)}{\pi^*(k)}Q(k|k_n).$$

Substituting the above formula into (3.12), we immediately get (3.13).

In Algorithm II, the proposals generated by (3.9) are screened by the coarse-scale acceptance–rejection test to reduce the number of unnecessary fine-scale simulations. One can skip that preconditioning step and get the following algorithm.

Algorithm III. (Coarse-gradient Langevin algorithm)

- Step 1. At k_n, generate a trial proposal Y from the coarse Langevin algorithm (3.9).
- Step 2. Accept Y as a sample with probability

$$\rho(k_n, Y) = \min\left(1, \frac{q^*(k_n|Y)\pi(Y)}{q^*(Y|k_n)\pi(k_n)}\right), \tag{3.14}$$

i.e. $k_{n+1} = Y$ with probability $\rho(k_n, Y)$, and $k_{n+1} = k_n$ with probability $1 - \rho(k_n, Y)$.

We will demonstrate numerically that Algorithm II is indeed more efficient than Algorithm III.

In a previous work [10], we studied preconditioning the MCMC algorithms by coarse-scale models, where the independent sampler and random walk sampler are used as the instrumental distribution. In this paper, our goal is to show that one can use coarse-scale models in Langevin algorithms. In particular, we can use coarse-scale gradients instead of fine-scale gradients in these algorithms. Our numerical experiments show that the coarse-scale distribution somewhat regularizes (smooths) the fine-scale distribution, which allows us to take larger time steps in the Langevin algorithm (3.9). In addition, we employ the preconditioning technique from [10] to increase the acceptance rate of the coarse-gradient Langevin algorithms.

3.3. Analysis of the coarse-gradient Langevin algorithms

In this section, we will briefly discuss the convergence property of the preconditioned coarse-grid Langevin algorithm. Denote

$$\begin{aligned} &\mathscr{E} = \{k;\ \pi(k) > 0\},\\ &\mathscr{E}^* = \{k;\ \pi^*(k) > 0\},\\ &\mathscr{D} = \{k;\ q^*(k|k_n) > 0 \text{ for any } k_n \in \mathscr{E}\}. \end{aligned} \tag{3.15}$$

The set $\mathscr{E}$ is the support of the posterior (target) distribution $\pi(k)$. $\mathscr{E}$ contains all the permeability fields k which have a positive probability of being accepted as a sample. Similarly, $\mathscr{E}^*$ is the support of the coarse-scale distribution $\pi^*(k)$, which contains all the k acceptable by the the coarse-scale test. $\mathscr{D}$ is the set of all possible proposals which can be generated by the Langevin distribution $q^*(k|k_n)$. To make the coarse-gradient Langevin MCMC methods sample properly, the conditions $\mathscr{E} \subseteq \mathscr{D}$ and $\mathscr{E} \subseteq \mathscr{E}^*$ must hold (up to a zero measure set) simultaneously. If one of these conditions is violated, say, $\mathscr{E} \not\subseteq \mathscr{E}^*$, then there will exist a subset $A \subset (\mathscr{E} \setminus \mathscr{E}^*)$ such that

$$\pi(A) = \int_A \pi(k)\,\mathrm{d}k > 0 \quad \text{and} \quad \pi^*(A) = \int_A \pi^*(k)\,\mathrm{d}k = 0,$$

which means no element of A can pass the coarse-scale test and A will never be visited by the Markov chain $\{k_n\}$. For Langevin algorithms, $\mathscr{E} \subset \mathscr{D}$ is always satisfied since $\mathscr{D}$ is the whole space. By choosing the parameter σ_c in $\pi^*(k)$ properly, the condition $\mathscr{E} \subset \mathscr{E}^*$ can also be satisfied. A typical choice would be $\sigma_c \approx \sigma_f$. More discussions on the choice of σ_c can be found in [10], where a two-stage MCMC algorithm is discussed.

Denote by K the transition kernel of the Markov chain $\{k_n\}$ generated by Algorithm II. Since its effective instrumental proposal is $Q(k|k_n)$, the transition kernel K has the form

$$\begin{aligned} &K(k_n, k) = \rho(k_n, k)Q(k|k_n),\ \ k \neq k_n,\\ &K(k_n, \{k_n\}) = 1 - \int_{k \neq k_n} \rho(k_n, k)Q(k|k_n)\,\mathrm{d}k. \end{aligned} \tag{3.16}$$

That is, the transition kernel $K(k_n,\cdot)$ is continuous when $k \neq k_n$ and has a positive probability at the point $k = k_n$. First, we show that $K(k_n, k)$ satisfies the detailed balance condition, that is

$$\pi(k_n)K(k_n, k) = \pi(k)K(k, k_n) \tag{3.17}$$

for all k, k_n. The equality is obvious when $k = k_n$. If $k \neq k_n$, then

$$\begin{aligned} \pi(k_n)K(k_n, k) &= \pi(k_n)\rho(k_n, k)Q(k|k_n) = \min(Q(k|k_n)\pi(k_n), Q(k_n|k)\pi(k))\\ &= \min\left(\frac{Q(k|k_n)\pi(k_n)}{Q(k_n|k)\pi(k)}, 1\right)Q(k_n|k)\pi(k) = \rho(k, k_n)Q(k_n|k)\pi(k) = \pi(k)K(k, k_n). \end{aligned}$$

Using the detailed balance condition (3.17), we can easily show that for any measurable set $A \subset \mathscr{E}$ the expression $\pi(A) = \int K(k, A)\,\mathrm{d}k$ holds. So $\pi(k)$ is indeed the stationary distribution of the transition kernel $K(k_n, k)$.

In Algorithm II, the proposal distribution (3.9) satisfies the positivity condition

$$q^*(k|k_n) > 0 \text{ for every} (k_n, k) \in \mathscr{E} \times \mathscr{E}. \tag{3.18}$$

With this property, we can easily prove the following lemma.

Lemma 3.1. *If $\mathscr{E} \subset \mathscr{E}^*$, then the chain $\{k_n\}$ generated by Algorithm II is strongly π-irreducible.*

Proof. According to the definition of strong irreducibility, we only need to show that $K(k_n, A) > 0$ for any $k_n \in \mathscr{E}$ and any measurable set $A \subset \mathscr{E}$ with $\pi(A) > 0$. From the formula (3.16) we have

$$K(k_n, A) \geqslant \int_{A \setminus k_n} K(k_n, k)\,\mathrm{d}k = \int_{A \setminus k_n} \rho(k_n, k) Q(k_n, k)\,\mathrm{d}k = \int_{A \setminus k_n} \rho(k_n, k) g(k_n, k) q(k|k_n)\,\mathrm{d}k.$$

In the above inequality, the equal sign holds when $k_n \notin A$. Since $\pi(A) = \int_A \pi(k)\,\mathrm{d}k > 0$, it follows that $m(A) = m(A \setminus k_n) > 0$, where $m(A)$ is the Lebesgue measure. If $\mathscr{E} \subset \mathscr{E}^*$, then both $\rho(k_n, k)$ and $g(k_n, k)$ are positive in A. Combining the positivity condition (3.18), we can easily conclude that $K(k_n, A) > 0$, which completes the proof. □

For the transition kernel (3.16) of Algorithm II, there always exist certain states $k^* \in \mathscr{E}$ such that $K(k^*, \{k^*\}) > 0$. That is, if the Markov chain is on state k^* at step n, then it has a positive probability to remain on state k^* at step $n + 1$. This condition ensures that the Markov chain generated by Algorithm II is aperiodic. Based on the irreducibility and stability property of Markov chains [23,20], the following convergence result is readily available.

Theorem 3.2 (Robert and Casella [23]). *The Markov chain $\{k_n\}$ generated by the preconditioned coarse-gradient Langevin algorithm is ergodic: for any function $h(k)$,*

$$\lim_{N \to \infty} \frac{1}{N} \sum_{n=1}^{N} h(k_n) = \int h(k)\pi(k)\,\mathrm{d}k. \tag{3.19}$$

Moreover, the distribution of k_n converges to $\pi(k)$ in the total variation norm

$$\lim_{n \to \infty} \sup_{A \in \mathscr{B}(\mathscr{E})} |K^n(k_0, A) - \pi(A)| = 0 \tag{3.20}$$

for any initial state k_0, where $\mathscr{B}(\mathscr{E})$ denote all the measurable subsets of $\mathscr{E}$.

4. Numerical setting and results

In this section we discuss the implementation details of Langevin MCMC method and present some representative numerical results. Suppose the permeability field $k(x)$ is defined on the unit square $\Omega = [0, 1]^2$. We assume that the permeability field k is known at some spatial locations, and the covariance of $\log(k)$ is also known. We discretize the domain Ω by a rectangular mesh, hence the permeability field k is represented by a matrix (thus k is a high dimensional vector). As for the boundary conditions, we have tested various boundary conditions and observed similar performance for the Langevin MCMC method. In our numerical experiments we will assume $p = 1$ and $S = 1$ on $x = 0$ and $p = 0$ on $x = 1$ and no flow boundary conditions on the lateral boundaries $y = 0$ and $y = 1$. We have chosen this type of boundary conditions because they provide a large deviation between coarse- and fine-scale simulations for permeability fields considered in the paper. We will consider both single-phase and two-phase flow displacements.

Using the Karhunen–Loève expansion [19,24], the permeability field can be expanded in terms of an optimal L^2 basis. By truncating the expansion we can represent the permeability matrix by a small number of random parameters. To impose the hard constraints (the values of the permeability at prescribed locations), we will find a linear subspace of our parameter space (a hyperplane) which yields the corresponding values of the permeability field. First, we briefly recall the facts of the Karhunen–Loève expansion. Denote $Y(x, \omega) = \log[k(x, \omega)]$, where the random element ω is included to remind us that k is a random field. For simplicity, we assume that $E[Y(x, \omega)] = 0$. Suppose $Y(x, \omega)$ is a second order stochastic process with $E \int_\Omega Y^2(x, \omega)\,\mathrm{d}x < \infty$, where E is the expectation operator. Given an orthonormal basis $\{\phi_k\}$ in $L^2(\Omega)$, we can expand $Y(x, \omega)$ as a general Fourier series

$$Y(x, \omega) = \sum_{k=1}^{\infty} Y_k(\omega)\phi_k(x), \quad Y_k(\omega) = \int_\Omega Y(x, \omega)\phi_k(x)\,\mathrm{d}x.$$

We are interested in the special L^2 basis $\{\phi_k\}$ which makes the random variables Y_k uncorrelated. That is, $E(Y_iY_j) = 0$ for all $i \neq j$. Denote the covariance function of Y as $R(x,y) = E[Y(x)Y(y)]$. Then such basis functions $\{\phi_k\}$ satisfy

$$E[Y_iY_j] = \int_\Omega \phi_i(x)\,dx \int_\Omega R(x,y)\phi_j(y)\,dy = 0, \quad i \neq j.$$

Since $\{\phi_k\}$ is a complete basis in $L^2(\Omega)$, it follows that $\phi_k(x)$ are eigenfunctions of $R(x,y)$:

$$\int_\Omega R(x,y)\phi_k(y)\,dy = \lambda_k\phi_k(x), \quad k = 1,2,\ldots, \tag{4.1}$$

where $\lambda_k = E[Y_k^2] > 0$. Furthermore, we have

$$R(x,y) = \sum_{k=1}^{\infty} \lambda_k\phi_k(x)\phi_k(y). \tag{4.2}$$

Denote $\theta_k = Y_k/\sqrt{\lambda_k}$, then θ_k satisfy $E(\theta_k) = 0$ and $E(\theta_i\theta_j) = \delta_{ij}$. It follows that

$$Y(x,\omega) = \sum_{k=1}^{\infty} \sqrt{\lambda_k}\theta_k(\omega)\phi_k(x), \tag{4.3}$$

where ϕ_k and λ_k satisfy (4.1). We assume that the eigenvalues λ_k are ordered as $\lambda_1 \geqslant \lambda_2 \geqslant \cdots$ The expansion (4.3) is called the Karhunen–Loève expansion (KLE). In the KLE (4.3), the L^2 basis functions $\phi_k(x)$ are deterministic and resolve the spatial dependence of the permeability field. The randomness is represented by the scalar random variables θ_k. After we discretize the domain Ω by a rectangular mesh, the continuous KLE (4.3) is reduced to finite terms. Generally, we only need to keep the leading order terms (quantified by the magnitude of λ_k) and still capture most of the energy of the stochastic process $Y(x,\omega)$. For an N-term KLE approximation $Y_N = \sum_{k=1}^{N}\sqrt{\lambda_k}\theta_k\phi_k$, define the energy ratio of the approximation as

$$e(N) := \frac{E\|Y_N\|^2}{E\|Y\|^2} = \frac{\sum_{k=1}^{N}\lambda_k}{\sum_{k=1}^{\infty}\lambda_k}.$$

If λ_k, $k = 1,2,\ldots$, decay very fast, then the truncated KLE would be a good approximation of the stochastic process in the L_2 sense.

Suppose the permeability field $k(x,\omega)$ is a log-normal homogeneous stochastic process, then $Y(x,\omega)$ is a Gaussian process and θ_k are independent standard Gaussian random variables. We assume that the covariance function of $Y(x,\omega)$ has the form

$$R(x,y) = \sigma^2 \exp\left(-\frac{|x_1 - y_1|^2}{2L_1^2} - \frac{|x_2 - y_2|^2}{2L_2^2}\right). \tag{4.4}$$

In the above formula, L_1 and L_2 are the correlation lengths in each dimension, and $\sigma^2 = E(Y^2)$ is a constant. We first solve the eigenvalue problem (4.1) numerically on the rectangular mesh and obtain the eigenpairs $\{\lambda_k,\phi_k\}$. Since the eigenvalues decay fast, the truncated KLE approximates the stochastic process $Y(x,\omega)$ fairly well in L^2 sense. Therefore, we can sample $Y(x,\omega)$ from the truncated KLE (4.3) by generating Gaussian random variables θ_k.

In the simulations, we first generate a reference permeability field using the full KLE of $Y(x,\omega)$ and obtain the corresponding fractional flows. To represent the discrete permeability fields from the prior (unconditioned) distribution, we keep 20 terms in the KLE, which captures more than 95% of the energy of $Y(x,\omega)$. We assume that the permeability field is known at nine distinct points. This condition is imposed by setting

$$\sum_{k=1}^{20} \sqrt{\lambda_k}\theta_k\phi_k(x_j) = \alpha_j, \tag{4.5}$$

where α_j ($j = 1,\ldots,9$) are prescribed constants. For simplicity, we set $\alpha_j = 0$ for all $j = 1,\ldots,9$. In the simulations we propose eleven θ_i and calculate the rest of θ_i by solving the linear system (4.5). In all the simulations,

we test 5000 samples. Because the direct Langevin MCMC simulations are very expensive, we only select a 61×61 fine-scale model for single-phase flow and a 37×37 fine-scale model for two-phase flow. Here 61 and 37 refer to the number of nodes in each direction, since we use a finite element based approach. Typically, we consider 6 or 10 times coarsening in each direction. In all the simulations, the gradients of the target distribution are computed using finite-difference differentiation rule. The time step size $\Delta\tau$ of the Langevin algorithm is denoted by δ. Based on the KLE, the parameter space of the target distribution $\pi(k)$ will change from k to θ in the numerical simulations, and the Langevin algorithms can be easily rewritten in terms of θ.

Our first set of numerical results are for single-phase flows. First, we present a comparison between the fine-scale response surfaces π and the coarse-scale response surface π^* defined by (3.3) and (3.8), respectively. Because both π and π^* are scalar functions of 11 parameters, we plot the restriction of them to a 2D hyperplane by fixing the values of 9θ. In Fig. 4.1, π^* (left figure) and π (right figure) are depicted on such a 2D hyperplane. It is clear from these figures that the overall agreement between the fine- and coarse-scale response surfaces is good. This is partly because the fractional flow is an integrated response. However, we notice that the fine-scale response surface π has more local features and varies on smaller scales compared to π^*.

In Fig. 4.2, we compare the acceptance rates of Algorithms I–III with different coarse-scale precision σ_c. The acceptance rate is defined as the ratio between the number of accepted permeability samples and the number of fine-scale acceptance–rejection test. Since Algorithm I does not depend on the coarse-scale precision, its

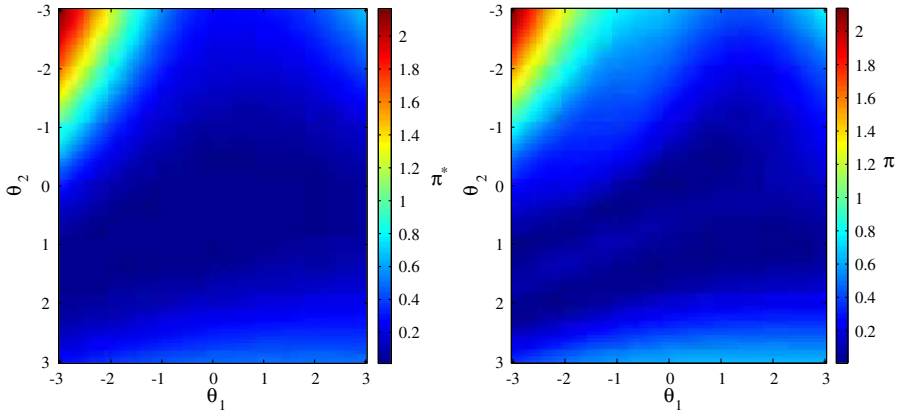

Fig. 4.1. Left: Coarse-scale response surface π^* (defined by (3.8)) restricted to a 2D hyperplane. Right: Fine-scale response surface π (defined by (3.3)) restricted to the same 2D hyperplane.

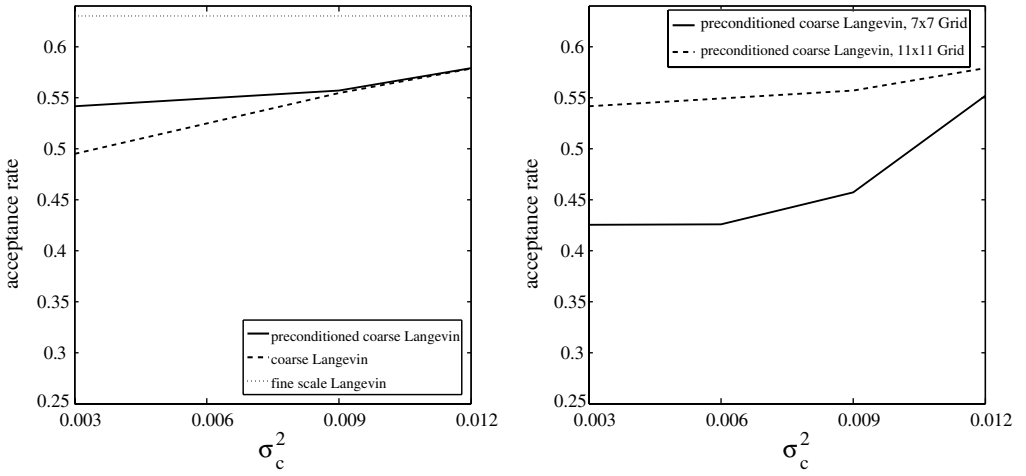

Fig. 4.2. Acceptance rate comparison between the direct fine-scale Langevin, preconditioned coarse-gradient Langevin and coarse-gradient Langevin algorithms for single-phase flow; $\delta = 0.05$, $\sigma_f^2 = 0.003$. In the left plot, the coarse-grid 11×11 is used in the simulation.

acceptance rate is the same for different σ_c. As we can see from the figure, Algorithm II has higher acceptance rates than Algorithm III. The gain in the acceptance rates is due to the Step 2 of Algorithm II, which filters unlikely acceptable proposals. To compare the effect of different degrees of coarsening, we plot in Fig. 4.2 the acceptance rate of Algorithm II using both 7×7 coarse models and 11×11 coarse models. Since 11×11 coarse models are more accurate, its acceptance rate is higher. In Fig. 4.3, we present the numerical results where larger time step δ is used in the Langevin algorithms. Comparing with Fig. 4.2, we find that that the acceptance rates for all the three methods decrease as δ increases. In all the numerical results, the Algorithm I, which uses the fine-scale Langevin method (3.6), gives a slightly higher acceptance rate than both Algorithms II and III. However, Algorithm I is more expensive than Algorithms II and III since it uses the fine-scale gradients in computing the Langevin proposals. In Fig. 4.4, we compare the CPU time for the different

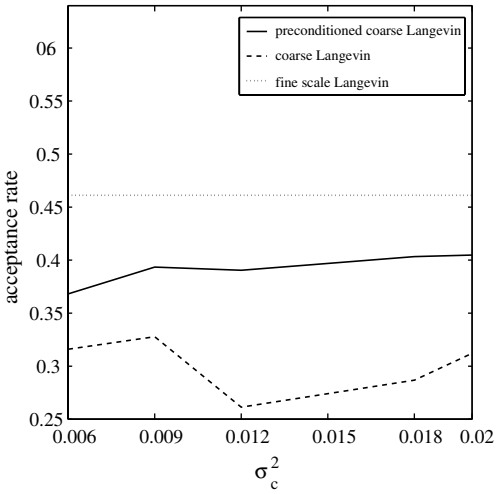

Fig. 4.3. Acceptance rate comparison for the direct fine-scale Langevin, preconditioned coarse-gradient Langevin and coarse-gradient Langevin algorithms for single-phase flow, $\delta = 0.1$, $\sigma_f^2 = 0.003$.

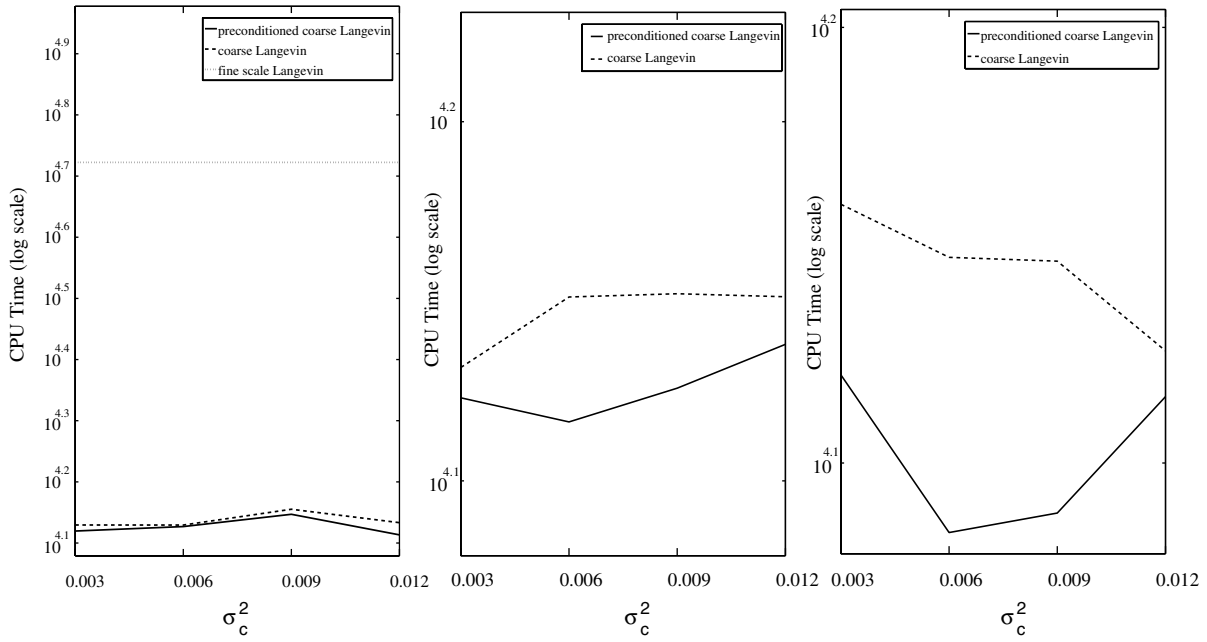

Fig. 4.4. CPU time (seconds) comparison for the different Langevin algorithms. Left: $\sigma_f^2 = 0.003$, $\delta = 0.05$, 11×11 coarse-grid. Middle: $\sigma_f^2 = 0.003$, $\delta = 0.05$, 7×7 coarse-grid. Right: $\sigma_f^2 = 0.003$, $\delta = 0.1$, 11×11 coarse-grid.

Langevin methods. From the left plot we see that Algorithm I is several times more expensive than Algorithms II and III. In the middle and right plots, we compare Algorithms II and III when a different coarse-model and a different time step size δ are used, respectively. We observe that the preconditioned coarse-gradient Langevin algorithm is slightly faster than the coarse Langevin algorithm without preconditioning.

In all the above numerical simulations, we choose the fine-scale error precision $\sigma_f^2 = 0.003$. The scaling of the error precision depends on the norm used in (3.3). If one choses σ_f to be very large, then the precision is very low, and consequently, most proposals will be accepted in the direct Langevin algorithm as well as in the coarse-gradient Langevin algorithms. In this case, the acceptance rates of the coarse-gradient Langevin algorithms are still similar to the acceptance rate of the direct Langevin algorithm. However, for very large σ_f, the preconditioning step in Algorithm II may not help to improve the acceptance rate, since most proposals will pass the preconditioning step.

Next we compare the fractional flow errors for the preconditioned coarse-gradient Langevin method and the direct fine-scale Langevin method in Fig. 4.5. Our objective is twofold. First, we would like to compare the convergence rates of the preconditioned coarse-gradient Langevin algorithm with that of the fine-scale (direct) Langevin algorithm. Second, we would like to show that the sampled permeability fields give nearly the same fractional flow response as the reference fractional flow data. We see from the left plot that both methods converge to the steady state within the same number of iterations. In the right plot, the fractional flows for sampled realizations are plotted (dotted lines). The fractional flows of the sampled realizations are very close to the reference fractional flow. This is because the error precision is taken to be small ($\sigma_f^2 = 0.003$) in the target distribution. In Fig. 4.6, some permeability realizations sampled from the posterior distribution are plotted. In particular, we plot realizations which do not look very similar to each other and represent the uncertainty range observed in our simulations. We observe that the samples capture some features of the reference permeability field. Note that all these permeability fields give nearly the same fractional flows as the reference fractional flow, so they are all eligible samples.

Next we consider two-phase flow simulations. Because two-phase flow simulations are computationally intensive, we restrict our computations to the fine grid 37×37, and the coarse-grid 7×7. In all simulations, we set $\sigma_f^2 = 0.003$ and $\delta = 0.05$. Fig. 4.7 shows the response surfaces π and π^* restricted to a two dimensional hyperplane in θ. As in the case of the single-phase flow, π^* approximates π very well in large scales, though π has more variations on small scales. In Fig. 4.8, the acceptance rates for the Algorithms I–III are compared. As we can see from this figure, the acceptance rates of Algorithm II is very similar to that of Algorithm I. Without preconditioning, Algorithm III has lower acceptance rates than Algorithm II. Comparing the

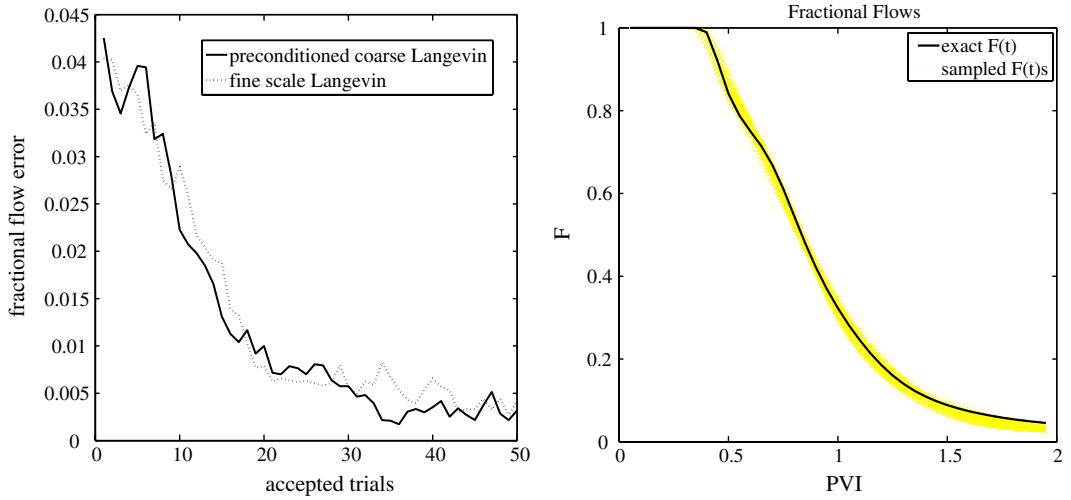

Fig. 4.5. Left: The fractional flow errors for direct fine-scale Langevin algorithm compared with preconditioned coarse-gradient Langevin algorithm. Right: The fractional flows of sampled realizations and the reference fractional flow. In these numerical tests, $\delta = 0.05$, $\sigma_f^2 = 0.003$ and 11×11 coarse-scale model is used.

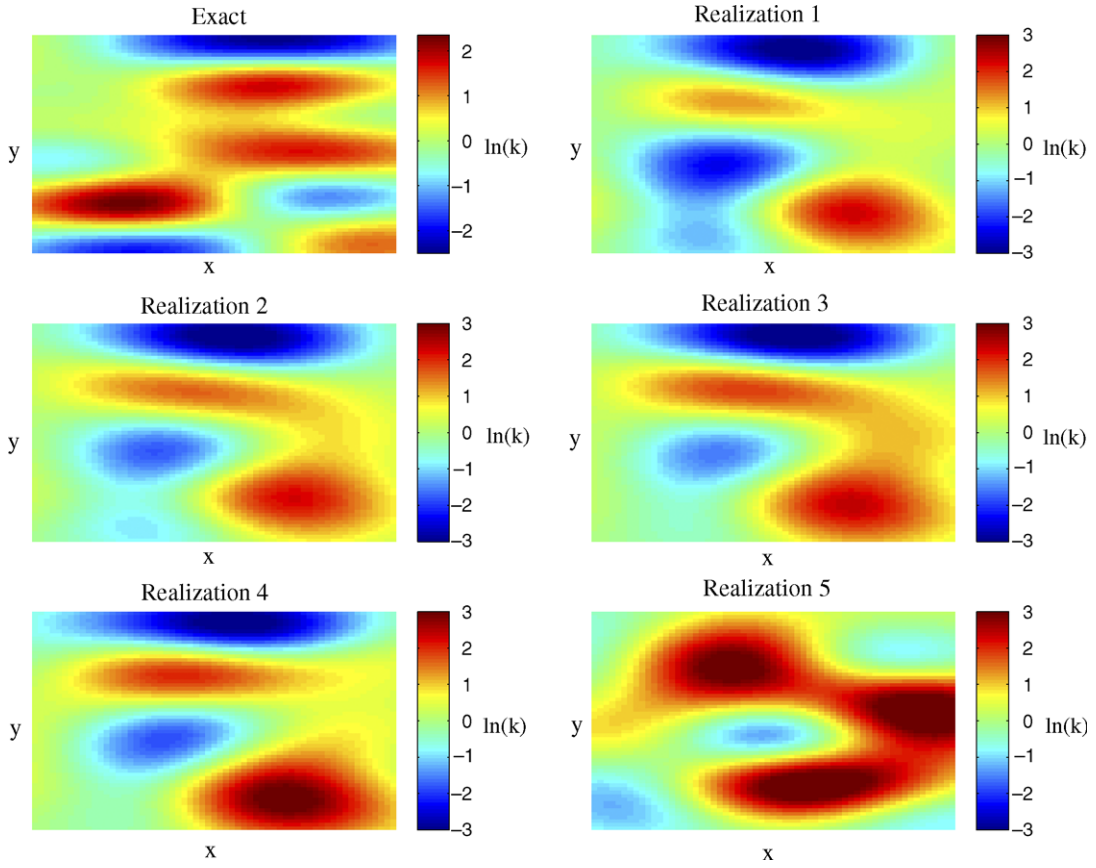

Fig. 4.6. Samples of the permeability realizations. Realizations are selected to represent the uncertainty range in the simulations.

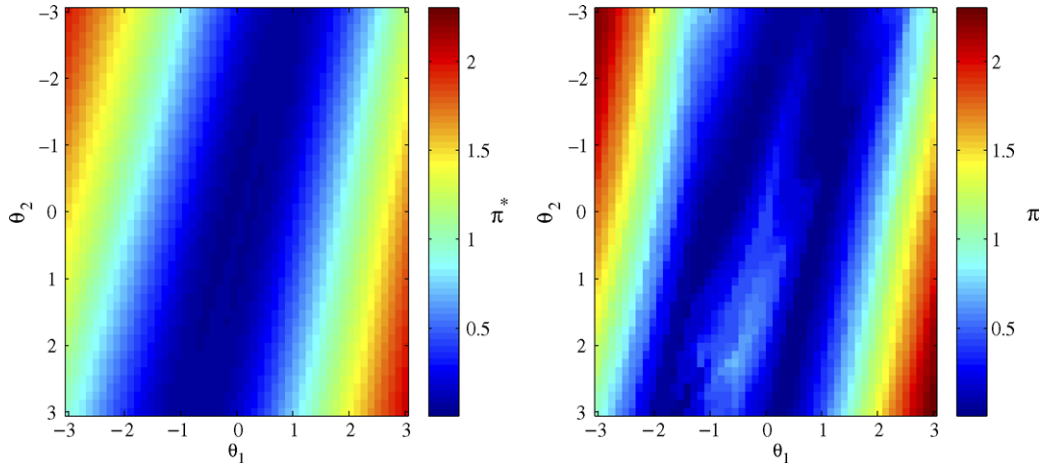

Fig. 4.7. Left: Coarse-scale response surface π^* restricted to 2D hyperplane. Right: Fine-scale response surface π restricted to the same 2D hyperplane.

CPU time in Fig. 4.9, we observe that the preconditioned coarse-gradient Langevin method is an order of magnitude faster than the direct fine-scale Langevin method. If the resolution of the fine-grid is increased, one can expect an even higher acceleration rate by Algorithm II. In Table 4.1, we compare the coarse-gradient and the direct Langevin algorithms for different coarse grid resolutions. In these numerical results, we have

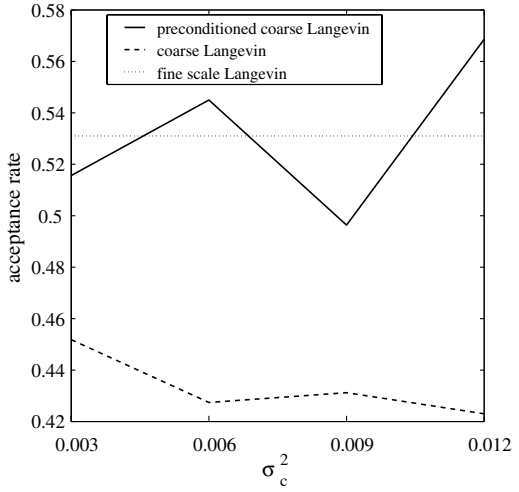

Fig. 4.8. Acceptance rate comparison for direct fine-scale Langevin, preconditioned coarse-gradient Langevin and coarse-gradient Langevin algorithms for two-phase flow, $\delta = 0.05$, $\sigma_f^2 = 0.003$.

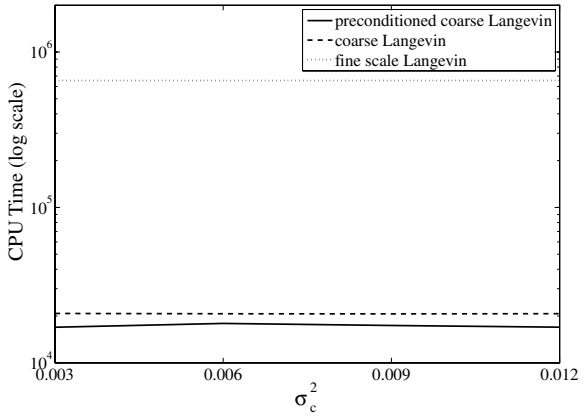

Fig. 4.9. CPU times (s) for Langevin algorithms. $\sigma_f^2 = 0.003$, $\delta = 0.05$, 7×7 coarse-grid.

Table 4.1
Comparison of Algorithms I–III for different coarse-grid resolutions in two-phase flow simulations

Coarse grid	Coarse		Coarse preconditioned		Direct	
	Accept. rate	CPU	Accept. rate	CPU	Accept. rate	CPU
4×4	0.47	8527	0.55	7036	0.53	655895
7×7	0.45	21859	0.52	17051	0.53	655895
10×10	0.46	70964	0.57	48653	0.53	655895

$\sigma_f^2 = \sigma_c^2 = 0.003$, $\delta = 0.05$.

chosen $\sigma_c^2 = 0.003$, though similar results are observed for other values of σ_c^2. We also observe that the preconditioned coarse-gradient Langevin algorithm has higher acceptance rate and lower CPU time compared to the coarse-gradient Langevin algorithm without preconditioning for all coarse grid resolutions.

In Fig. 4.10, the fractional flow errors and fractional flows are plotted. In the two-phase flow case, we observe that the fine-grid Langevin algorithm converges slightly faster than the preconditioned coarse-gradient Langevin method. Finally, in Fig. 4.11, we plot some permeability realizations. We selected the samples which do not look very similar to each other and represent the uncertainty range observed in the simulations.

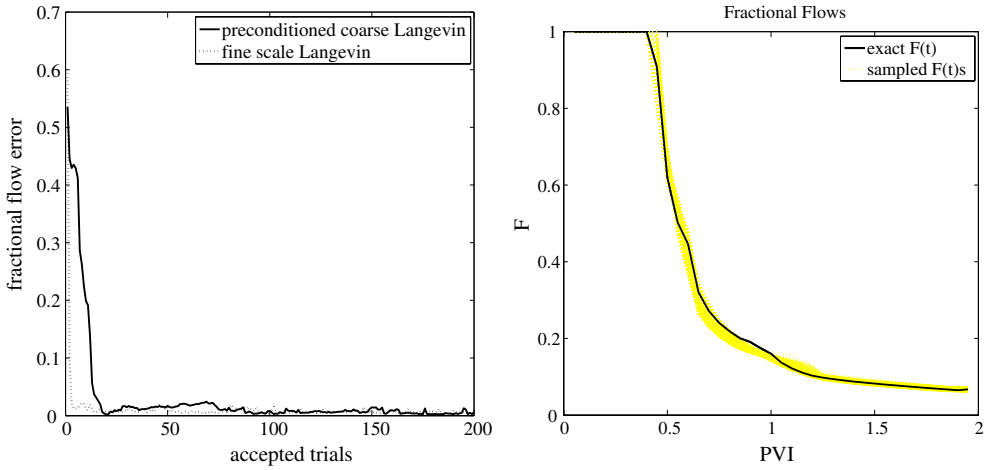

Fig. 4.10. Left: The fractional flow errors for direct fine-scale Langevin algorithm compared with preconditioned coarse-gradient Langevin algorithm. Right: The fractional flows of sampled realizations and the reference fractional flow. In these numerical tests, $\delta = 0.05$, $\sigma_f^2 = 0.003$ and 7×7 coarse-scale model is used.

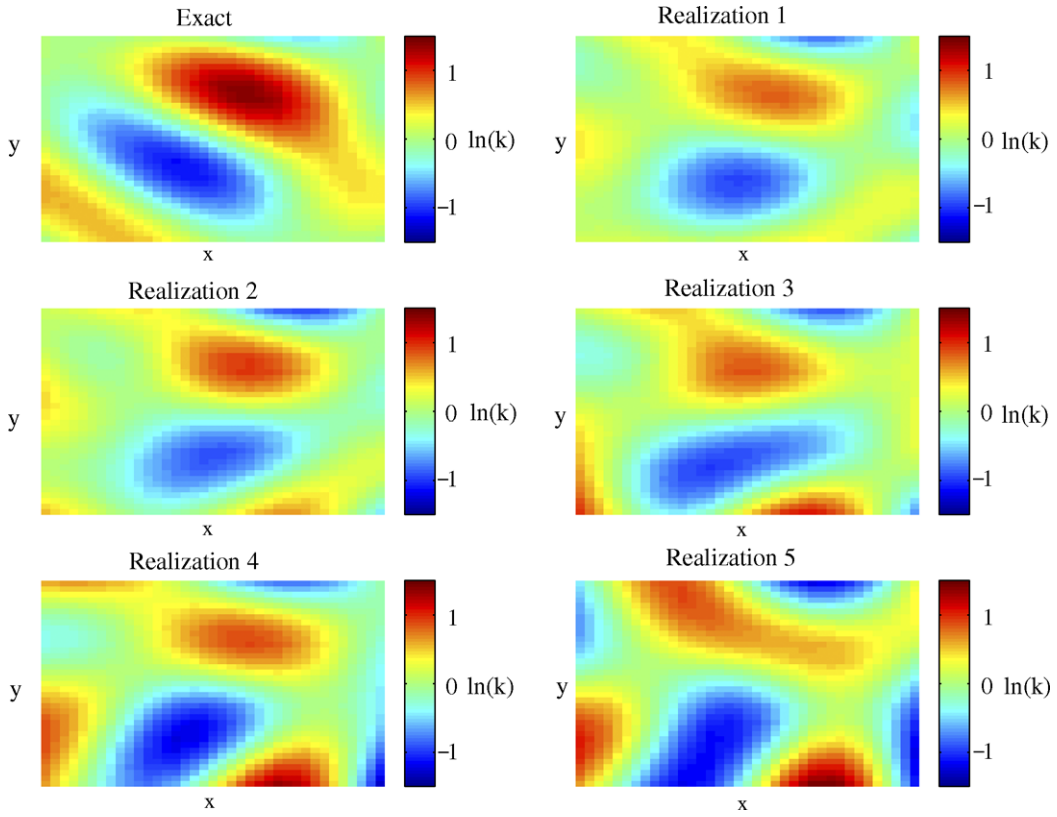

Fig. 4.11. Samples of the permeability realizations. Realizations are selected to represent the uncertainty range in the simulations.

This figure illustrates that the sampled permeability realizations capture the main features of the reference permeability field.

Next, we compare the theoretical computational costs of the three Langevin algorithms for the two-phase flow problem. Denote t_f and t_c as the CPU time to solve the PDE system (2.1)–(2.3) on the fine- and coarse-grid respectively. Suppose D is the dimension of the parameter space of the permeability field k, and N is

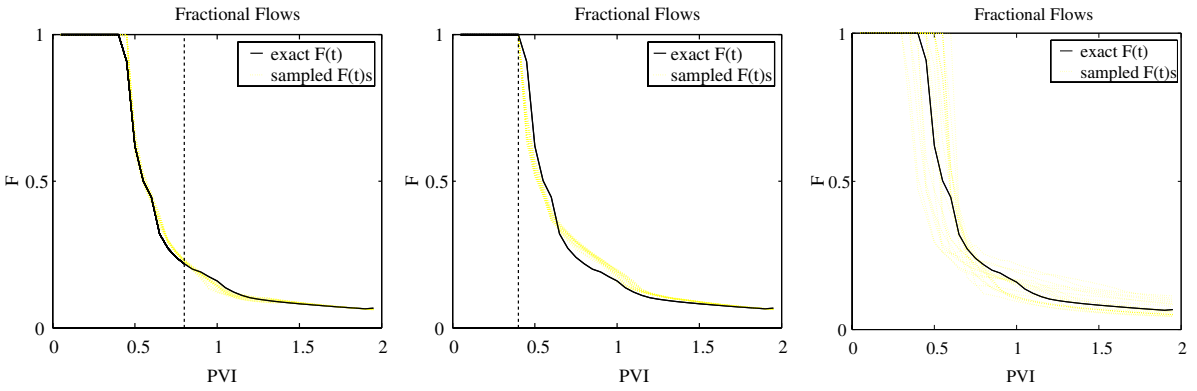

Fig. 4.12. Prediction results using the information about the dynamic data on various time spans. Left: The dynamic data information on [0,0.8] PVI is used; Middle: The dynamic data information in [0,0.4] PVI is used. Right: No dynamic data information is used.

the number of proposals that are tested in all three Langevin algorithms. For each new Y, Algorithm I needs to compute the target distribution $\pi(Y)$ and its gradient $\nabla\pi(Y)$ on the fine-grid. If the gradient is computed by the forward difference scheme, then the PDE system (2.1)–(2.3) needs to be solved on the fine-grid $(D+1)$ times. Therefore, the total computational cost of Algorithm I is $N(D+1)t_f$. For Algorithm III, the gradient of the distribution is computed on the coarse-grid. However, the acceptance test is calculated on the fine-grid for each proposal. Thus, its computational cost is $N(Dt_c + t_f)$. In Algorithm II, the gradient of the distribution is also computed on the coarse-grid. And each new sample is first tested by the coarse-scale distribution. If it passes the coarse-grid acceptance test, then the proposal will be further tested by the fine-scale distribution. Suppose M proposals (out of N) pass the coarse-scale test, then the total computational cost of Algorithm II is $N(D+1)t_c + Mt_f$. Thus, Algorithm II is $\frac{N(D+1)t_f}{N(D+1)t_c+Mt_f}$ times faster than Algorithm I, and $\frac{N(Dt_c+t_f)}{N(D+1)t_c+Mt_f}$ times faster than Algorithm III. In our computations, D is of order of ten because we represent the permeability field by its truncated Karhunen–Loève expansion. If the fine-scale model is scaled up six times in each direction, as we did in the numerical experiment, then the coarse-scale model is approximately 36 times faster than the fine-scale model. Indeed, at each time step solving the pressure equation on the coarse grid is approximately 36 times faster than on the fine grid. The same is true for the saturation equation since it is also solved on the coarse grid and with larger time steps. Moreover, in the preconditioned coarse-gradient Langevin algorithm, only a portion of the N proposals can pass the coarse-scale test, where N is usually two times larger than M. Using these estimates, we expect that the CPU time of the preconditioned coarse-gradient Langevin algorithms should be an order of magnitude lower than that of the fine-scale Langevin algorithm. We indeed observed a similar speedup in our computations, as demonstrated in Fig. 4.9.

Note that one can use simple random walk samplers, instead of the Langevin algorithm, in Algorithm I. We have observed in our numerical experiments that the acceptance rate of the random walk sampler is several times smaller than that of Langevin algorithms. This is not surprising because Langevin algorithms use the gradient information of the target distribution and are problem adapted. One can also use single-phase flow upscaling (as in [5]) in the preconditioning step as it is done in [9]. In general, we have found the multiscale methods to be more accurate for coarse-scale simulations and they can be further used for efficient and robust fine-scale simulations.

Finally, we present the results demonstrating the uncertainties in the predictions. In our simulations, we use the information of the dynamic data in various time spans. In Fig. 4.12, various prediction results are plotted based on information of the dynamic data on [0,0.8] PVI time (left figure), on [0,0.4] PVI time (middle figure), and when no dynamic data information is used (right figure). These results are obtained by sampling 50 realizations from the posterior distribution. As we observe from the figure, the uncertainty spread is the largest if no dynamic data information is used. However, the uncertainty spread decreases, as expected, if more information is incorporated into the simulations. In particular, using the dynamic data information up to 0.8 PVI allows us to obtain accurate predictions and reduce the uncertainties. These results allow us to assess the uncertainties in the predictions.

5. Conclusions

In this paper, we study the coarse-gradient Langevin algorithms for dynamic data integration. The gradients involved in Langevin algorithms are computed by inexpensive coarse-scale simulations using a multiscale finite element framework. Furthermore, the proposals are tested with coarse-scale runs in order to increase the acceptance rate. Numerical results show that the proposed algorithms are efficient and can give very similar performance as the fine-scale Langevin algorithms with less computational cost.

Acknowledgements

The authors thank the referees for valuable comments, and Dr. Victor Ginting for providing the multiscale finite volume code. The research of Y.E. is partially supported by DOE Grant DE-FG02-05ER25669 and NSF Grant DMS-0327713. T.Y.H is supported by the NSF ITR Grant No. ACI-0204932 and the NSF FRG Grant DMS-0353838. The computations are performed using TAMU parallel computers funded by NSF Grant DMS-0216275.

Appendix A. Coarse-scale models using multiscale finite element methods

In this Appendix, we will describe the details of our coarse-scale model. As we noted earlier, this model is similar to single-phase upscaling method. However, instead of coarsening the absolute permeability, we use pre-computed multiscale finite element basis functions. The key idea of the method is the construction of basis functions on the coarse grids such that these basis functions capture the small scale information on each of these coarse grids. The method that we use follows its finite element counterpart presented in [14]. The basis functions are constructed from the solution of the leading order homogeneous elliptic equation on each coarse element with some specified boundary conditions. Thus, if we consider a coarse element K which has d vertices, the local basis functions ϕ^i, $i = 1,\ldots,d$ satisfy the following elliptic problem:

$$\begin{aligned} &-\nabla\cdot(k\cdot\nabla\phi^i) = 0 \quad \text{in } K \\ &\phi^i = g^i \quad \text{on } \partial K, \end{aligned} \tag{A.1}$$

for some function g^i defined on the boundary of the coarse element K. Hou et al. [14] have demonstrated that a careful choice of boundary condition would guarantee the performance of the basis functions to incorporate the local information and hence improve the accuracy of the method. The function g^i for each i varies linearly along ∂K. Thus, for example, in the case of a constant diagonal tensor the solution of (A.1) would be a standard linear/bilinear basis function. We note that, as usual, we require $\phi^i(\xi_j) = \delta_{ij}$. Finally, a nodal basis function associated with the vertex ξ in the domain Ω is constructed from the combination of the local basis functions that share this ξ and zero elsewhere. These nodal basis functions are denoted by $\{\psi_\xi\}_{\xi\in Z_h^0}$.

Having described the basis functions, we denote by V^h the space of our approximate pressure solution which is spanned by the basis functions $\{\psi_\xi\}_{\xi\in Z_h^0}$. Now we may formulate the finite dimensional problem corresponding to the finite volume element formulation of (2.2). A statement of mass conservation on a control volume V_ξ is formed from (2.2), where now the approximate solution is written as a linear combination of the basis functions. Assembly of this conservation statement for all control volumes would give the corresponding linear system of equations that can be solved accordingly. The resulting linear system has incorporated the fine-scale information through the involvement of the nodal basis functions on the approximate solution. To be specific, the problem now is to seek $p^h \in V^h$ with $p^h = \sum_{\xi\in Z_h^0} p_\xi \psi_\xi$ such that

$$\int_{\partial V_\xi} \lambda(S) k\cdot\nabla p^h\cdot\vec{n}\,\mathrm{d}l = \int_{V_\xi} f\,\mathrm{d}A, \tag{A.2}$$

for every control volume $V_\xi \subset \Omega$. Here $\vec{n}$ defines the unit normal vector on the boundary of the control volume, ∂V_ξ, and S is the fine scale saturation field at this point. We note that, concerning the basis functions, a vertex-centered finite volume difference is used to solve (A.1), and a harmonic average is employed to approximate the permeability k at the edges of fine control volumes.

Furthermore, the pressure solution may be used to compute the total velocity field at the coarse-scale level, denoted by $\bar{v} = (\bar{v}_x, \bar{v}_z)$ via (2.6). In general, the following equations are used to compute the velocities in horizontal and vertical directions, respectively:

$$\bar{v}_x = -\frac{1}{h_z}\sum_{\xi\in Z_h^0} p_\xi \left(\int_E \lambda(S) k_x \frac{\partial \psi_\xi}{\partial x} \mathrm{d}z \right), \tag{A.3}$$

$$\bar{v}_z = -\frac{1}{h_x}\sum_{\xi\in Z_h^0} p_\xi \left(\int_E \lambda(S) k_z \frac{\partial \psi_\xi}{\partial z} \mathrm{d}x \right), \tag{A.4}$$

where E is the edge of V_ξ. Furthermore, for the control volumes V_ξ adjacent to the Dirichlet boundary (which are half control volumes), we can derive the velocity approximation using the conservation statement derived from (2.2) on V_ξ. One of the terms involved is the integration along part of the Dirichlet boundary, while the rest of the three terms are known from the adjacent internal control volumes calculations. The analysis of the two-scale finite volume method can be found in [11].

As for the upscaling of the saturation equation, we only use the coarse scale velocity to update the saturation field on the coarse-grid, i.e.,

$$\frac{\partial \overline{S}}{\partial t} + \bar{v} \cdot \nabla f(\overline{S}) = 0, \tag{A.5}$$

where $\overline{S}$ denotes the saturation on the coarse-grid. In this case the upscaling of the saturation equation does not take into account subgrid effects. This kind of upscaling techniques in conjunction with the upscaling of absolute permeability are commonly used in applications (see e.g. [6–8]). The difference of our approach is that the coupling of the small scales is performed through the finite volume element formulation of the pressure equation.

References

[2] A. Christen, C. Fox, MCMC using an approximation. Technical Report, Department of Mathematics, The University of Auckland, New Zealand.

[3] M. Christie, Upscaling for reservoir simulation, J. Pet. Tech. (1996) 1004–1010.

[5] L.J. Durlofsky, Numerical calculation of equivalent grid block permeability tensors for heterogeneous porous media, Water Resour. Res. 27 (1991) 699–708.

[6] Coarse scale models of two phase flow in heterogeneous reservoirs: Volume averaged equations and their relationship to the existing upscaling techniques, Computat. Geosci., 2 (1998) 73–92.

[7] L.J. Durlofsky, R.A. Behrens, R.C. Jones, A. Bernath, Scale up of heterogeneous three dimensional reservoir descriptions, SPE Paper 30709, 1996.

[8] L.J. Durlofsky, R.C. Jones, W.J. Milliken, A nonuniform coarsening approach for the scale up of displacement processes in heterogeneous media, Adv. Water Res. 20 (1997) 335–347.

[9] Y. Efendiev, A. Datta-Gupta, V. Ginting, X. Ma, B. Mallick, An efficient two-stage Markov chain Monte Carlo method for dynamic data integration. Water Resources Research, 41, (in press).

[10] Y. Efendiev, T. Hou, W. Luo, Preconditioning Markov chain Monte Carlo simulations using coarse-scale models. SIAM Science and Computation (in press).

[11] V. Ginting, Analysis of two-scale finite volume element method for elliptic problem, J. Numer. Math. 12 (2) (2004) 119–142.

[12] J. Glimm, D.H. Sharp, Prediction and the quantification of uncertainty, Phys. D 133 (1999) 152–170, Predictability: quantifying uncertainty in models of complex phenomena (Los Alamos, NM, 1998).

[13] U. Grenander, M.I. Miller, Representations of knowledge in complex systems (with discussion), J.R. Statist. Soc. B 56 (1994) 549–603.

[14] T.Y. Hou, X.H. Wu, A multiscale finite element method for elliptic problems in composite materials and porous media, J. Comput. Phys. 134 (1997) 169–189.

[17] J.S. Liu, Monte Carlo Strategies in Scientific Computing, Springer, New York, 2001.

[19] M. Loève, Probability Theory, fourth ed., Springer, Berlin, 1977.

[20] S.P. Meyn, R.L. Tweedie, Markov Chains and Stochastic Stability, Springer-Verlag, London, 1996.

[21] D. Oliver, L. Cunha, A. Reynolds, Markov chain Monte Carlo methods for conditioning a permeability field to pressure data, Math. Geol. 29 (1997).

[22] D. Oliver, N. He, A. Reynolds, Conditioning permeability fields to pressure data, in: 5th European Conference on the Mathematics of Oil Recovery, Leoben, Austria, 3–6 September, 1996.
[23] C. Robert, G. Casella, Monte Carlo Statistical Methods, Springer-Verlag, New York, 1999.
[24] E. Wong, Stochastic Processes in Information and Dynamical Systems, MCGraw-Hill, New York, 1971.