SPE-163669: Multiscale method for simulating two and three-phase flow in porous media

Mayur Pal, Sadok Lamine, Knut-Andreas Lie, and Stein Krogstad

Abstract

Multiscale methods developed to solve coupled flow equations for reservoir simulation are based on a hierarchical strategy in which the pressure equation is solved on a coarsened grid and transport equation is solved on the fine grid as a decoupled system. The multiscale mixed finite-element (MsMFE) method attempts to capture sub-grid geological heterogeneity directly into the coarse-scale via mathematical basis functions. These basis functions are able to capture important multiscale information and are coupled through a global formulation to provide good approximation of the subsurface flow solution.

In the literature, the general formulation of the MsMFE method for incompressible two-phase and compressible three-phase flow has mainly addressed problems with idealized flow physics. In this paper, we present a new formulation that accounts for compressibility, gravity, and spatially-dependent capillary and relative-permeability effects.

We evaluate the computational efficiency and accuracy of the method by reporting the result of series of representative benchmark tests that have a high degree of realism with respect to flow physics, heterogeneity in petrophysical model, and geometry/topology of the corner-point grids. In particular, the MsMFE method is validated and compared against Shell's in-house simulator MoReS. The fine-scale flux, pressure, and saturation fields computed by the multiscale simulation show a noteworthy improvement in resolution and accuracy compared with coarse-scale models.

Introduction

The flow of hydrocarbons in subsurface rock formations is affected by physical processes occurring on multiple length (and time) scales, from the micrometer scale of individual pores to the kilometer scale of reservoir rock formations. Modern reservoir characterization and geostatistical modeling techniques aim to integrate information from these different scales to build high-resolution models with multi-million cells that describe the heterogeneous reservoir properties in great details. Solving multiphase flow equations on highly detailed geo-cellular models has a very high computational cost, and it is therefore common to perform some kind of data reduction to derive discrete flow equations on a much coarser model. Not only does this upscaling process consume a considerable number of man-hours, but the loss of potentially important fine-scale properties introduces a bias in the simulation that is often hard to estimate and control.

In the past few years, there has been an increasing interest in methods that attempt to reduce or bypass the need for upscaling. So-called multiscale methods (Efendiev and Hou 2009) are designed to accurately and effectively solve problems having multiple scales and offer a systematic framework for incorporating effects from an unresolved scale into the global flow equations in a manner that is consistent with the underlying differential operators. For reservoir simulation, this means that fine-scale petrophysical and geological details are captured directly into the coarse-scale simulation model without the need of explicit computation of effective coarse-scale properties.

The literature contains a wide range of multiscale methods that are applicable to reservoir simulation, including dual-grid methods (Guerillot and Verdiere 1995; Audigane and Blunt 2004; Arbogast 2002; Arbogast and Bryant 2002), (adaptive) local-global methods (Chen et al. 2003; Chen and Durlofsky 2006), finite-element methods (Hou and Wu 1997), mixed finite-element methods (Chen and Hou 2002; Aarnes 2004; Aarnes et al. 2008; Aarnes and Efendiev 2008; Alpak et al. 2011; Pal et al. 2012), and finite-volume multiscale methods (Jenny et al. 2003; Lee et al. 2008; Hajibeygi and Jenny 2009, 2011; Parramore et al. 2012). Although the methods have certain algorithmic differences, they share a common basic concept for incorporating fine-scale into coarse-scale flow equations. The local flow effects of fine-scale petrophysical properties are estimated by solving a local flow problems and represented as a set of multiscale basis functions that have a fine-scale subresolution. Boundary conditions used to localize each basis function can be set using local or global information, or possibly a combination thereof, see e.g., (Alpak et al. 2011). The global flow problem can then be posed in terms of the multiscale basis functions to compute a reduced set of degrees-of-freedom associated with a coarser grid. Altogether, this gives a robust and accurate method for upscaling the flow equations. Alternatively, the basis functions can be used to reconstruct pressure and mass-conservative flux fields that can be fed into an transport solver formulated on the original geo-cellular or some intermediate grid.

Despite their obvious similarities, multiscale methods should not be confused with upscaling. A comprehensive comparison of multiscale method with state-of-the-art upscaling methods for elliptic problems in porous media is presented by Kippe et al. (2008). Whereas the intent of upscaling is to generate approximate coarse-scale solutions, the main objective of the multiscale method is to obtain efficient and accurate approximations on a fine or intermediate scale (Farmer 2002). Moreover, the natural

coupling between local and global scales in multiscale methods avoids the inconsistency and non-physical coarse-scale properties that are often associated with many upscaling techniques.

In the following, we will discuss a particular multiscale method, the multiscale mixed finite-element (MsMFE) method, and discuss how this method can be extended to account for gravity, compressibility, and spatially-dependent relative-permeability and capillary-pressure functions. The resulting method is implemented as a software prototype using the MATLAB Reservoir Simulation Toolbox (MRST) (Lie et al. 2012; SINTEF 2011) and applied to a set of challenging test cases with flow physics, petrophysical heterogeneity, and grid complexity that is representative of real-life models. In particular, the efficiency and accuracy of the MsMFE method is assessed by benchmarking it against Shell's in-house reservoir simulator MoReS.

1 Mathematical model and fine-scale discretization

Two-phase flow in a porous medium can be modeled as a coupled system consisting of an elliptic flow equation and a parabolic transport equation that can be derived from the continuity equation over each phase α and Darcy's law describing the relationship between phase velocity \vec{v}_{α} and phase pressure p_{α} . The phase saturation S_{α} represents the fraction of accessible pore volume filled by a phase α , ρ_{α} is the density of phase α , and q_{α} denotes the sources/sink terms representing injection/production wells. Introducing capillary pressure $p_c = p_o - p_w$ and assuming that the two phases (water and oil) fill the pore volume, $S_o + S_w = 1$, we can derive the following equation for oil pressure and total velocity $\vec{v} = \vec{v}_w + \vec{v}_o$,

$$\nabla \cdot \vec{v} = q, \qquad \vec{v} = -\mathbf{K}\lambda \big[\nabla p_o - \tilde{g}(S_w)\nabla z + h(S_w)\nabla p_c\big],\tag{1}$$

and an evolution equation for the water saturation

$$\phi \frac{\partial S_w}{\partial t} + \nabla \cdot f_w(S_w) \left[\vec{v} + \mathbf{K} \lambda_o(S_w) \left((\rho_w - \rho_o) g \nabla z + \nabla p_c(S_w) \right) \right] = \frac{q_w}{\rho_w}.$$
(2)

Here, $q = q_o + q_w$ is the total flow rate; the permeability tensor **K** is either diagonal or full and will typically be discontinuous across internal boundaries of the domain; ϕ denotes the porosity; and $\lambda_{\alpha} = k_{ri}/\mu_{\alpha}$ the phase mobility, in which k_{ri} and μ_{α} denote the relative permeability and viscosity of phase α , respectively. Moreover, $\lambda = \lambda_w + \lambda_o$ is the total mobility, $f_{\alpha} = \lambda_{\alpha}/\lambda$ is the fractional flow function of phase α , g denotes gravity, and $\tilde{g}(S_w) = (f_w(S_w)\rho_w + f_o(S_w)\rho_o)g$. The form of the function $h(S_w)$ depends upon the choice of primary pressure variable. Here, we will use oil pressure, for which $h(S_w) = f_w(S_w)$. For simplicity, we will henceforth drop the subscripts on the primary variables p and S.

The computational domain Ω is discretized as a set of $\{C_i\}$ of N non-overlapping and matching polyhedral cells. The flow and transport equations Eq. 1–Eq. 2, will be solved sequentially: First, Eq. 1 is solved to provide explicit fluxes at the cell interfaces, which are then fed into Eq. 2 to evolve the saturations a time step Δt .

To simplify the presentation of the multiscale method, we will in the following work with three basic unknowns for the pressure equation: u_i denotes the vector of outward fluxes from cell C_i , p_i denotes the pressure at the cell center, and π_i the pressure at the cell faces. Standard, locally conservative discretization methods can then be in a form that uses Darcy's law to relate the three quantities u, p_i , and π through a matrix T_i of one-sided transmissibilities,

$$\boldsymbol{u}_{i} = \boldsymbol{T}_{i} \big[\boldsymbol{e}_{i} \boldsymbol{p}_{i} - \boldsymbol{\pi}_{i} - \tilde{\boldsymbol{g}}(\boldsymbol{S}_{i}) \Delta \boldsymbol{z}_{i} + h(\boldsymbol{S}_{i}) \big(\boldsymbol{e}_{i} \boldsymbol{p}_{c}(\vec{x}_{i}, \boldsymbol{S}_{i}) - \boldsymbol{p}_{ci} \big) \big], \quad \boldsymbol{e}_{i} = (1, ..., 1)^{T},$$
(3)

where Δz_i denotes the vector of differences in the *z*-coordinate of the cell center \vec{x}_i and the face centroids and the capillary pressure p_{ci} at the cell faces is defined as the linear interpolation of the capillary pressure in the neighboring cells. Alternatively, one can use multipoint methods to define transmissibilities on structured and unstructured grids using multi-point fluxes, see e.g., (Wheeler et al. 2012; Pal et al. 2006; Pal and Edwards 2007; Parramore et al. 2012).

By augmenting Eq. 3 with flux and pressure continuity across cell face, we can derive the following discrete linear system for the global flow problem,

$$\begin{bmatrix} B & C & D \\ C^T & 0 & 0 \\ D^T & 0 & 0 \end{bmatrix} \begin{bmatrix} u \\ -p \\ \pi \end{bmatrix} = \begin{bmatrix} -G(S)\Delta z + H(S)\Delta p_c \\ q \\ 0 \end{bmatrix}.$$
 (4)

Here, u denotes the outward face fluxes ordered cell wise (fluxes over interior faces and faults appear twice with opposite signs), p denotes the cell pressure, and π the face pressures, where each side in the presence of a fault is considered as a separate face.

The first row in the block-matrix equation corresponds to Eq. 3 for all grid cells, for which the right-hand terms $G(S)\Delta z$ and $H(S)\Delta p_c$ are the one-sided face contributions corresponding to gravity and capillary effects. The matrix B is block diagonal, with one block T_i^{-1} defined per cell. The matrix C is also block diagonal and has one block entry e_i per cell that duplicates the cell pressures to one value for each of the cell's faces. Hence, in the second row of Eq. 4, the transposed matrix C^T sums the face fluxes to define one mass-conservation equation per cell. Finally, each column of D corresponds to a unique face and has one (for boundary faces) or two (for interior faces) unit entries corresponding to the index of the face in the cell wise ordering.

In the following, the transport equation Eq. 2 will be solved on the fine-scale using a standard transport solver with upstreamweighted mobilities and two-point discretization of the second-order capillary term. The temporal discretization may be explicit or implicit. If needed, improved numerical accuracy could have been obtained by using higher-order upwind schemes, like the wave-oriented multi-dimensional schemes Lamine and Edwards (2009, 2010).



Fig. 1— The left plot shows the coarse grid defined hierarchically on top the fine grid so that each coarse block consists of a singly-connected set of cells. The diagram to the right shows the workflow for the multiscale simulation using the MsMFE method.

2 Multiscale mixed finite elements

The key idea of the multiscale mixed finite-element method is to construct a special approximation space, consisting of a set of coarse-scale basis functions that satisfy a flow equation locally. The MsMFE method is therefore formulated based on two hierarchically nested grids as illustrated in Figure 1. Rock and rock-fluid properties are given on the fine-scale geo-cellular grid, whereas the basis functions and corresponding degrees-of-freedom are associated with a coarse simulation grid used to solve the global flow problem. The blocks in the coarse grid are defined as a connected set of cells from the fine grid, and can in principle have arbitrary shapes. However, the best numerical resolution is obtained if the blocks are somewhat regular, follow the layered structures of stratigraphic grids (Aarnes et al. 2008), and/or adapt to high-contrast features (Natvig et al. 2012).

Basis functions are computed numerically by solving a local flow problem over each pair of neighboring grid blocks, inside which the flow is driven by source terms rather than boundary conditions, which are normally used in flow-based upscaling. Using these local basis functions, the effects of the fine-scale heterogeneity can be incorporated into the discretized coarse-scale flow problem in a way that is consistent with the local fine-scale properties of the differential operators. Figure 1 summarises our workflow for multiscale simulation.

Multiscale approximation. To formally define the MsMFE method, we start by writing the solution to Eq. 4 as the sum of the basis functions plus a fine-scale residual,

$$\boldsymbol{u} = \boldsymbol{\Psi} \boldsymbol{u}_c + \tilde{\boldsymbol{u}}, \qquad \boldsymbol{p} = \boldsymbol{\Phi} \boldsymbol{p}_c + \tilde{\boldsymbol{p}}, \qquad \boldsymbol{\pi} = \boldsymbol{\Pi} \boldsymbol{\pi}_c + \tilde{\boldsymbol{\pi}}.$$
 (5)

Here, u_c denotes the vector of outward fluxes over the coarse-block interfaces, p_c denotes the vector of coarse-block pressures, and π denotes the vector of coarse-block face pressures. Likewise, \tilde{u} , \tilde{p} , $\tilde{\pi}$ are reminder terms having variations on the fine grid. The matrices Ψ , Φ , and Π represent the fine-scale reconstruction operators for \vec{v} , p, and π . Each column in Ψ corresponds to a multiscale basis function for the flux associated with a unique coarse-grid face and is represented as a $n_f \times 1$ vector of fine-scale fluxes.

For compressible flow, we also need to define fine-scale variations for the pressure basis so that each column of Φ is a basis function associated with unique cell and each column of Π corresponds to a basis function defined over a coarse face. For incompressible flow, on the other hand, pressure is seldom used explicitly except to determine well-rates through the use of appropriate well models. Hence, we define the pressure to be constant within each coarse block and replace Φ by a simple prolongation operator I that maps a constant value from each coarse block onto the cells of the block. Likewise, Π is replaced by a prolongation operator J that maps a constant value from each coarse face onto the individual cell faces of the coarse face. Altogether, this defines a reconstruction operator $R = \text{diag}(\Psi, I, J)$ that enables us to map the degrees-of-freedom $x_c = [u_c, -p_c, \pi_c]$ on the coarse-scale to the corresponding fine-scale quantities $x = [u, -p, \pi]$.

Coarse system. To form a global system on the coarse grid, we need a compression operator that will bring the fine-scale system Eq. 4 to the space spanned by our multiscale basis functions. Here, \mathbf{R}^{T} is a natural choice since the transposed of the prolongation operators I and J correspond to the sum over all fine cells of a coarse block and all fine-cell faces that are part of the faces of the coarse blocks, respectively. Multiplying Eq. 4 from the left by \mathbf{R}^{T} , substituting $\mathbf{x} = \mathbf{R}\mathbf{x}_{c}$, and rearranging terms,

we obtain

$$\begin{bmatrix} \boldsymbol{\Psi}^{\mathsf{T}} \boldsymbol{B} \boldsymbol{\Psi} & \boldsymbol{\Psi}^{\mathsf{T}} \boldsymbol{C} \boldsymbol{I} & \boldsymbol{\Psi}^{\mathsf{T}} \boldsymbol{D} \boldsymbol{J} \\ \boldsymbol{I}^{\mathsf{T}} \boldsymbol{C}^{\mathsf{T}} \boldsymbol{\Psi} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{J}^{\mathsf{T}} \boldsymbol{D}^{\mathsf{T}} \boldsymbol{\Psi} & \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_{c} \\ -\boldsymbol{p}_{c} \\ \boldsymbol{\pi}_{c} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Psi}^{\mathsf{T}} \left(\boldsymbol{H}(\boldsymbol{S}) \Delta \boldsymbol{p}_{c} - \boldsymbol{G}(\boldsymbol{S}) \Delta \boldsymbol{z} \right) - \boldsymbol{\Psi}^{\mathsf{T}} \left(\boldsymbol{B} \tilde{\boldsymbol{u}} - \boldsymbol{C} \tilde{\boldsymbol{p}} + \boldsymbol{D} \tilde{\boldsymbol{\pi}} \right) \\ \boldsymbol{I}^{\mathsf{T}} \boldsymbol{Q} - \boldsymbol{I}^{\mathsf{T}} \boldsymbol{C}^{\mathsf{T}} \tilde{\boldsymbol{u}} \\ -\boldsymbol{J}^{\mathsf{T}} \boldsymbol{D}^{\mathsf{T}} \tilde{\boldsymbol{\pi}} \end{bmatrix} .$$
(6)

The fine-scale reminder terms can be eliminated as follows: \tilde{p} disappears if we interpret the coarse-scale pressure as the *w*-weighted average of the true pressure, $p_c^i = \int_{B_i} wp \, d\vec{x}$, where *w* is the source term used to define basis functions, see Eq. 8. The other two remainder terms \tilde{u} and $\tilde{\pi}$ are simply neglected, and this gives the following coarse-scale system

$$\begin{bmatrix} \boldsymbol{\Psi}^{\mathsf{T}} \boldsymbol{B} \boldsymbol{\Psi} & \boldsymbol{\Psi}^{\mathsf{T}} \boldsymbol{C} \boldsymbol{I} & \boldsymbol{\Psi}^{\mathsf{T}} \boldsymbol{D} \boldsymbol{J} \\ \boldsymbol{I}^{\mathsf{T}} \boldsymbol{C}^{\mathsf{T}} \boldsymbol{\Psi} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{J}^{\mathsf{T}} \boldsymbol{D}^{\mathsf{T}} \boldsymbol{\Psi} & \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_{c} \\ -\boldsymbol{p}_{c} \\ \boldsymbol{\pi}_{c} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Psi}^{\mathsf{T}} \big(\boldsymbol{H}(\boldsymbol{S}) \Delta \boldsymbol{p}_{c} - \boldsymbol{G}(\boldsymbol{S}) \Delta \boldsymbol{z} \big) \\ \boldsymbol{I}^{\mathsf{T}} \boldsymbol{q} \\ \boldsymbol{0} \end{bmatrix}.$$
(7)

Multiscale basis functions. There are two different ways to compute basis functions: in the single-block method, one has to specify fine-scale fluxes over the coarse interface to which the basis function is associated. The method is not very accurate unless the interface flux incorporates some kind of global flow information. Herein, we will therefore use a two-block method that does not impose any condition on the interface between two coarse blocks. The resulting method is not convergent, but will typically give reasonable accuracy on finite grids. To define the method, we consider two blocks B_i and B_j that share a common coarse face $\Gamma_{ij} = \partial B_i \cap \partial B_j$, and let B_{ij} be a singly-connected subset of Ω that contains B_i and B_j . Neglecting the influence of gravity and capillary forces, the two-block multiscale basis function is defined as

$$\vec{\psi}_{ij} = -\mathbf{K}\nabla\phi_{ij}, \qquad \nabla \cdot \vec{\psi}_{ij} = w_{ij}(\vec{x}) = \begin{cases} w_i(\vec{x}), & \text{if } \vec{x} \in B_i, \\ -w_j(\vec{x}), & \text{if } \vec{x} \in B_j, \\ 0, & \text{otherwise}, \end{cases}$$
(8)

in B_{ij} with $\vec{\psi}_{ij} \cdot \vec{n} = 0$ on ∂B_{ij} . If $B_{ij} \neq B_i \cup B_j$, we say that the basis function is computed using overlap or oversampling to lessen the impact of the artificial no-flow condition on the boundary.

The purpose of the weight function $w_{ij}(\vec{x})$ is to distribute the divergence of the velocity, $\nabla \cdot \vec{v}$ over the coarse block. To produce a unit flow across the interface Γ_{ij} , the weight function should be chosen on the form $w_i(x) = \theta(x) / \int_{B_i} \theta(x) dx$. The function $\theta(x)$ can be defined in several ways. For incompressible flow, the simplest choice is to set $\theta(\vec{x}) \equiv 1$ or $\theta(\vec{x}) = \text{trace}(\mathbf{K})$ away from the possible wells and $\theta(\vec{x}) = q(\vec{x})$ in grid blocks penetrated by wells. This will reproduce the lowest-order Raviart–Thomas basis on rectangular blocks with homogeneous, isotropic permeability.

Capillary forces. To account for capillary forces, we introduce an additional set of basis functions defined as

$$\vec{\psi}_{ij}^c = -\mathbf{K} \big(\nabla \phi_{ij}^c - h(S) \nabla p_c(S) \big), \qquad \nabla \cdot \vec{\psi}_{ij}^c = 0.$$
(9)

Using an extra set of basis functions instead of adding capillary effects directly in Eq. 8 has the advantage that we avoid the problem of having to scale the relative contributions of the physical capillary terms and the artificial source term w_{ij} and reduce the saturation dependence in our set of basis functions.

Compressibility. The basic flow model Eq. 1 can be extended to compressible flow as follows

$$\nabla \cdot \vec{v} = q - c_t \frac{\partial p}{\partial t} + \left(\gamma(S, p)\vec{v} + \beta(p)\mathbf{K}g\nabla z\right) \cdot \nabla p, \quad \vec{v} = -\lambda\mathbf{K}\left(\nabla p - \tilde{g}(S)\nabla z\right), \tag{10}$$

where c_t denotes total compressibilities and $\beta(p)$ and $\gamma(S, p)$ are known functions of pressure- and saturation-dependent parameters. (To simplify the presentation, capillary forces have been neglected in Eq. 10).

A discrete system on mixed form can be derived by linearizing Eq. 10 and using a mimetic discretization $v_i = T_i(p_i e_i - \pi_i)$,

$$\begin{bmatrix} \boldsymbol{B}^{n} & \boldsymbol{C} \\ \boldsymbol{C}^{\mathsf{T}} & \boldsymbol{P}^{n}(\boldsymbol{p}_{\nu+1}^{n+1}) \end{bmatrix} \begin{bmatrix} \boldsymbol{v}_{\nu+1}^{n+1} \\ -\boldsymbol{p}_{\nu+1}^{n+1} \end{bmatrix} = \begin{bmatrix} \boldsymbol{f}^{n}(\boldsymbol{p}_{\nu}^{n+1}) \\ \boldsymbol{g}^{n}(\boldsymbol{p}^{n}, \boldsymbol{p}_{\nu}^{n+1}) \end{bmatrix}.$$
(11)

Here, ν indicates iterations in a nonlinear solver. Similarly, the superscript *n* indicates that *B*, *P*, *f*, and *g* are functions of the saturation and pressure at the previous time step and will henceforth be dropped for brevity. For compressible flow, the pressure is no longer immaterial and our basis functions should thus also include subscale pressure variations. From the definition of the basis functions Eq. 8, it follows that the basis functions for pressure and flux should scale similarly. Hence, we use the following saturation-dependent multiscale decomposition for pressure, $p = Ip_c + \Lambda \Phi v_c + \tilde{p}$, where $\Lambda = \text{diag}(\lambda_{\ell}^0 / \lambda_{\ell})$ and λ^0

is the mobility used to calculate basis function ℓ . Introducing the multiscale expansion in Eq. 11, applying compression operator diag(Ψ^{T}, I^{T}), and neglecting residual terms, we obtain the coarse-scale system

$$\begin{bmatrix} \boldsymbol{\Psi}^{\mathsf{T}} \boldsymbol{B} \boldsymbol{\Psi} & \boldsymbol{\Psi}^{\mathsf{T}} \boldsymbol{C} \boldsymbol{I} \\ \boldsymbol{I}^{T} (\boldsymbol{C}^{T} \boldsymbol{\Psi} - \boldsymbol{P}_{\nu} \boldsymbol{\Lambda} \boldsymbol{\Phi}) & \boldsymbol{I}^{T} \boldsymbol{P}_{\nu} \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{v}_{c}^{\nu+1} \\ -\boldsymbol{p}_{c}^{\nu+1} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Psi}^{\mathsf{T}} \boldsymbol{f}_{\nu} \\ \boldsymbol{I}^{T} \boldsymbol{g}_{\nu} \end{bmatrix}$$
(12)

that needs to be solved iteratively to construct a multiscale approximation.

To get a fine-scale approximation that converges to zero fine-scale residual, we need to include an equation for the residual terms that were neglected in Eq. 12

$$\begin{bmatrix} \boldsymbol{B} & \boldsymbol{C} \\ \boldsymbol{C}^{\mathsf{T}} & \boldsymbol{P} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{v}}^{\nu+1} \\ -\hat{\boldsymbol{p}}^{\nu+1} \end{bmatrix} = \begin{bmatrix} \boldsymbol{f}_c - \boldsymbol{\Psi}^{\mathsf{T}} \boldsymbol{B} \boldsymbol{\Psi} \boldsymbol{v}_c + \boldsymbol{\Psi}^{\mathsf{T}} \boldsymbol{C} \boldsymbol{I} \boldsymbol{p}_c \\ \boldsymbol{g}_c - \boldsymbol{I}^{\mathsf{T}} (\boldsymbol{C}^{\mathsf{T}} \boldsymbol{\Psi} - \boldsymbol{P}_{\nu} \boldsymbol{\Lambda} \boldsymbol{\Phi}) \boldsymbol{v}_c + \boldsymbol{I}^{\mathsf{T}} \boldsymbol{P}_{\nu} \boldsymbol{I} \boldsymbol{p}_c \end{bmatrix}.$$
(13)

If the residuals have a localized structure, this equation can be solved efficiently by using a standard iterative, overlapping Schwarz method. Hence, the resulting iterative method, iMsMFE for short, consists of an outer loop, in which we iterate over Eq. 12 and Eq. 13 to reduce the fine-scale residual, and two inner loops that are used to solve Eq. 12 and Eq. 13, respectively.

3 Numerical results

In this section, we will validate the MsMFE method on test cases involving realistic reservoir geometries and properties. The aim of the first test case is assess the computational efficiency of the method on a large-scale geological model with approximately 700,000 cells. The second test case compares the performance of the multiscale method with the Shell standard simulator (MoReS 2010). Case three and four aim to validate the multiscale method for incompressible two-phase flow with gravity and spatially-dependent capillary pressure and relative permeability. Case three involves two regions with different relative permeability and capillary curves, whereas the fourth case corresponds to a sector model with multiple regions with a different relative permeability and capillary curve associated with each region. The final test case demonstrates the use of MsMFE for compressible three-phase flow.

To perform the numerical experiments, the MsMFE methods described above have been implemented as software prototypes in Matlab, using the Matlab Reservoir Simulation Toolbox (MRST) (Lie et al. 2012; SINTEF 2011). The main purpose of MRST is to simplify the prototyping and testing of new computational methods on general unstructured grids, and for this reason, computational efficiency has in several cases been sacrificed for the sake of generality and flexibility of the toolbox. For the MsMFE implementation, in particular, the data structure used to represent the basis functions is far from optimal and significant computational overhead is incurred, e.g., when extracting basis functions to assemble coarse systems. To get a more reliable assessment of the computational efficiency of the MsMFE method, we have developed a C-accelerated version of the incompressible method, in which all steps except for the linear solve of the coarse system are performed in C via a MEX interface to Matlab. However, in both the C-accelerated and the pure Matlab versions, the basis functions are, for the sake of generality, computed using a mimetic discretization with a TPFA inner-product, which is a factor 3–10 less efficient than using a standard cell-centered TPFA implementation.

All runtimes reported in the following refer to simulations performed on a computer with Intel Core2 Duo Processors (6M Cache, 2.80 GHz, 1066 MHz FSB) and 4 GB RAM.

Example 1 (Shell geomodel) In the first test case, we evaluate the efficiency of the MsMFE solver implemented in MRST and, in particular, compare the C-accelerated version to its pure Matlab counterpart. To this end, we consider a realistic large-scale geomodel shown in Figure 2, for which the full simulation grid consists of $253 \times 258 \times 38$ cells. In the simulation, cells with zero porosity or permeability are set to be inactive, giving a total of 721,999 active cells. Furthermore, we set the minimum permeability to 1 micro darcy. We use a two-phase model where water is injected into oil, with a mobility ratio of 10. In our timing experiments, we will focus exclusively on the pressure equation and not consider the transport solves that would have been performed on the fine scale. Our simulation setup consists of first computing basis functions, and then solving the global pressure equation one thousand times. To mimic a worst-case scenario of effects arising from saturation updates, the pressure system is initialized by assigning a random relative permeability value in each cell before each pressure solve. By updating the pressure one thousand times, we get a picture of where the time is spent during a dynamic simulation. For instance, given a pressure step of ten days, our test will mimic a simulation of 27 years of production.

First we compare time consumption on a subset of the Shell geomodel. The results are displayed in Table 1. The C-accelerated code gives a reduction in runtime of over 80% compared to the pure MATLAB implementation even though the MEX interface is not optimal since data must be copied between C and MATLAB. Solving the same system on the fine scale with the AGMG multigrid solver (AGMG 2012; Notay 2010) takes 200 seconds, giving a reduction in runtime compared to the fine-scale solver of 86% for the pure Matlab solver and 97% for the C-accelerated solver.

The results for the C-accelerated code on the full Shell geomodel are shown in Table 2. For large data sets, as in this case, implementing the whole multiscale simulator in a compiled language would be much more efficient, since a significant computational overhead is induced when using the MEX interface to copy data between MATLAB and C in the C-accelerated



Fig. 2— The Shell geomodel. The plot in (a) shows the full $253 \times 258 \times 38$ model with a $15 \times 15 \times 7$ coarse grid imposed. The plot in (b) shows a subset of the full model.

Table 1— Runtimes in seconds for one thousand pressure updates for a $20 \times 20 \times 12$ subset of the Shell geomodel with	th a
$3 \times 3 \times 3$ coarse grid. Fine-scale solution with AGMG: 200.07 seconds.	

Task	C-accelerated		pure Matlab	
Construct coarse grid (x1)	0.01	0.08%	0.02	0.07%
Compute basis functions (x1)	0.74	14.53%	1.68	6.05%
Assemble coarse system (x1000)	0.94	18.37%	20.09	72.20%
Solve coarse system (x1000)	1.81	35.14%	1.92	6.90%
Reconstruct fine flux (x1000)	1.64	31.86%	4.11	14.76%
Total time	5.14	100.00%	27.82	100.00%
Reduction compared to AGMG	97%		8	36%

Table 2— Time consumption in seconds for the full $253 \times 258 \times 38$ Shell geomodel with 1000 pressure steps, for C-accelerated multiscale code without updating the basis functions.

Task/coarse dim	$10 \times 10 \times 5$		$15 \times 15 \times 7$	
Construct coarse grid (x1)	0.09	0.01%	0.01	0.01%
Compute basis functions (x1)	323.42	40.48%	224.24	31.33%
Assemble coarse system (x1000)	46.46	5.82%	54.69	7.64%
Solve coarse system (x1000)	25.35	3.17%	54.12	7.56%
Reconstruct fine flux (x1000)	403.62	50.52%	382.61	53.46%
Total time	798.95	100.00%	715.76	100.00%

MRST code. In particular, for the reconstruction of fine-scale fluxes, which is the most expensive operation reported in Table 2, over 50% of the time is spent copying data. Because of large memory consumption, we were not able to run the AGMG solver on the full model.

We expect that the results presented above extend readily to *incompressible* black-oil models in the absence of gravity and capillary forces: the key to efficiency is to exploit the natural parallelism in computing basis functions and reuse basis functions from one step to the next.

Example 2 (Carbonate reservoir, sector model) The geometrical and physical properties used in this particular sector model are based on a real-field carbonate reservoir. The sector model dimensions are 2 km by 2 km in the areal dimension and about 50 m in the vertical dimension. The fine-scale model has 20×20 cells in the lateral direction and 93 layers in vertical direction. The coarse grid for the multiscale and the upscaled models consists of $5 \times 5 \times 11$ blocks. Figure 3 shows the fine-scale porosity distribution, the hierarchical grid used by the multiscale method, and the upscaled porosity distribution on the coarse model. The reservoir is produced using a five-spot injection pattern, with one injector at each corner of the model and one producer in the center, see Figure 4. The fluid and the reservoir data used for the simulations are presented in Table 3. The model represent a scenario mimicking 2000 days of production.



Fig. 3— Sector model of a carbonate reservoir. The left plot shows the porosity on the Figure showing the multiscale grid overlaying the fine scale model on the left and upscaled model on right



Fig. 4— Five-spot well pattern for the sector model with one injector in each of the four corners and one producer in the middle of the model.

Property	Value	Units
Water viscosity	0.393	сР
Oil viscosity	1.1	cP
Water density	1138	Kg/m3
Oil density	832	Kg/m3
Connate water saturation	0.2	-
Irreducible oil saturation	0.2	-
Initial reservoir pressure	6000	psi
Well injection rate	10000	bbl/day

Table 3— Fluid and reservoir data used for the sector model



Fig. 5— Water cut and oil production curves for the carbonate sector model.

In the following, we will compare four different simulations: using standard MRST solvers on the coarse and fine grids, using the MsMFE method, and using Shell's simulator MoReS (MoReS 2010) on the fine grid. Figure 5 reports the resulting water-cut and oil-production curves. It can be seen from the plots that the MoReS fine-scale, the MRST fine-scale, and the multiscale production curves for oil production and water cut almost coincide, whereas the coarse/upscaled model shows some deviation from the two fine-scale production curves. Moreover, we notice that the multiscale simulation is in close agreement with the fine-scale simulation results obtained by MoReS.

Figure 6 shows the corresponding saturation profiles computed in the four different simulations. The coarse model produces significantly different results than the other three simulations, and would not have been used for simulation in practice. On the other hand, the cost of updating the pressure is almost the same for the coarse-scale and the multiscale simulation, and the coarse-scale saturation profiles have been included to demonstrate what can be gained by exploiting the subresolution that is inherent in the multiscale basis functions to compute saturations. However, the most interesting comparison is between the fine-scale MoReS and the multiscale/fine-scale MRST simulations. Clearly, the volumetric sweeps predicted by the multiscale simulation is in close agreement with the two fine-scale simulations.

The previous example validated the fine-scale and multiscale MRST solvers against Shell's in-house reservoir simulator for a sector model with realistic heterogeneity and geometry, but with simplified flow physics. In the next example, we will consider an example with a simple geometry but more realistic flow physics that includes gravity and spatially-dependent relative-permeability and capillary curves.

Example 3 (Two saturation regions) We consider a simple 2D box that consists of two saturation regions that have different relative-permeability and capillary curve. The permeability of the medium is equal 100 milli darcy throughout the whole domain, and the porosity is homogeneous and equal 0.3. The reservoir is initially fully saturated with oil and is represented on a regular Cartesian grid with 20×20 cells, which we have partitioned uniformly into 5×5 coarse blocks, see Figure 7. Water is injected at the rate of $0.5 \text{ m}^2/\text{day}$ from the bottom of the domain and oil is produced from the top. Gravity is acting in the z-direction. The transport loop runs with a pressure step of 0.1 year. The initial fine-scale and multiscale pressure distributions are shown in Figure 7.

Figure 8 compares production curves and the evolution of the saturation profiles computed by the fine-scale and multiscale simulations. Because of the different capillary curves, the saturation profile of the injected water will be significantly different in the upper and lower parts of the domain. Altogether, the multiscale method is able to capture the (unresolved) fine-scale effects quite accurately, giving an approximate solution that is only slightly different from the fine-scale solution. This is confirmed in the right part of the figure, which shows the percentage discrepancy between the fine-scale and the multiscale simulation along with curves depicting the corresponding saturations in the production well.

Example 4 (Sector model with nine regions) We consider a sector model that covers an area of 3 km by 3 km and has a thickness of approximately 100 m. The model has nine different saturation regions that each corresponds to a rock-type that is represented by its own relative permeability and capillary curves, shown in Figure 9. Petrophysical data and well placement are presented in Figure 10. Initial reservoir pressure is 4728.23 psi and the reservoir is produced by two wells that are located at diagonally opposite corners of the model in a quarter five-spot pattern. The injection well operates at a rate constraint of 3000 STB per day. The production well operates at a bottom-hole pressure constraint of 100 psi.



Fig. 6— Saturation profiles for the carbonate sector model.



Fig. 7— The plot to the left shows two different linear capillary curves corresponding to the two saturation regions shown in the upper-middle plot. The lower-middle plot shows the placement of injection and production well and subdivision into coarse blocks. The plots to the right show the initial pressure distribution computed by the fine-scale and the multiscale solvers.



Fig. 8— The plots to the left show saturation profiles for the fine-scale and multiscale simulation for the box model. Capillary effects are clearly visible in the saturation distribution. The fine-scale and multiscale saturations are very similar. The plots to the right show the percentage discrepancy between fine-scale and multiscale simulation and water saturation in production well.



Fig. 9— The left plot shows the sector model of a reservoir with nine different saturation regions. The middle and left plots show the corresponding relative permeability and capillary pressure curves, respectively.



Fig. 10— Petrophysical data for the $21 \times 21 \times 13$ sector model. The left plot shows the permeability, which spans the interval from 50 mD to 400 mD, and the middle plot shows the porosity, which varies in the interval [0.02, 0.12]. The right plot shows the well placement and a $5 \times 5 \times 3$ coarse grid, outlined on top of the original geo-cellular model.



Fig. 11— Water saturation distribution in the sector model after twenty years computed by the fine-scale solver (left) and the multiscale solver (right).



Fig. 12— The plots show, from left to right: percentage discrepancy between the saturation in the fine-scale and multiscale solutions as a function of time, water saturations in the production well versus time, and oil and water cuts.

Figure 11 shows the water saturation after twenty years computed by the fine-scale solver and the MsMFE solver operating on a $5 \times 5 \times 3$ coarse grid. The computed solutions are not identical, but the solutions exhibit the same qualitative behavior and clearly demonstrate the effect of using different capillary and relative permeability curves in the different regions of the model. Figure 12 gives a more detailed analysis of the difference between the two flow predictions. The discrepancy in saturation rises up to 4.25% until water breakthrough, but then starts to drop. Likewise, the saturation in the production well and the oil and water cuts show that although the breakthrough deviates slightly in the two simulations, the solutions seem to eventually converge. Alltogether, the plots show that the multiscale simulation is able to account for gravity and spatial variations in capillary and relative permeability to provide a reasonably accurate fine-scale flow prediction.

In the last test, we consider a three-phase flow problem described by the compressible, black-oil equations. There are several ways to discretize and solve these equations, and a prerequisite for successful application of the iterative MsMFE methodology is to have a robust numerical formulation for the fine-scale problem that solves the flow and transport in separate steps. To the best of our knowledge, it is not yet clear that there exists such a formulation that is fully robust, efficient, and can be written on the mixed (hybrid) formulation used to derive the MsMFE method. Herein, we will use a sequential method with a standard mixed formulation for the pressure equation and an implicit transport solver with saturation as primary variable. and simply assume that this is a reasonable solution strategy for the fine-scale equations.

Example 5 (Compressible three-phase flow) We consider a sector of a reservoir in the shape of a $500 \times 500 \times 15m$ shoebox,



Fig. 13— Well placement and permeability, plotted on \log_{10} -scale, for the compressible three-phase test case.



Fig. 14— Oil production, gas production, and gas cut computed by the fine-scale and the multiscale solvers for a 3D compressible three-phase test case.

realized on a grid consisting of $10 \times 10 \times 3$ cells. The model is initially filled with oil at 200 bar. The reservoir is produced by two wells that are located in opposite corners of the model which are set to operate at a fixed bottom-hole pressure of 300 and 200 bar, respectively. Gas is the injected fluid. Both fluids are assumed to be compressible, with a compressibility of $5 \cdot 10^{-3}$ bar for the oil and the gas following an ideal gas law. The fluids have linear relative permeabilities and a viscosity of 1 cP for the oil and 0.1 cP for the gas. The heterogeneous permeability distribution and the well pattern are shown in Figure 13.

Our primary interest for this example is to investigate how well the multiscale method is able to predict the global flow responses in this pressure-controlled system; that is, how accurate the method predicts oil and gas rates in the injector and producer, as well as the gas cut in the producer. Figure 14 reports a comparison of these quantities computed by the fine-scale solver, the iterative MsMFE method on $a5 \times 5 \times 1$ coarse grid, and the original MsMFE method without iterations on the same coarse grid. In the simulation, we have used equally spaced time steps, each of length 40 days, to reach the final time of 600 days. The MsMFE method clearly underestimates the oil production, whereas the gas production and gas cut are calculated quite accurately. By adding extra iterations in the iMsMFE method, the multiscale method calculates a correct profile also for the oil production.

4 Conclusions

In this paper we have reviewed a multiscale mixed finite-element method for incompressible two-phase flow and discussed how to extend the method to include more realistic flow physics like gravity and spatially-dependent capillary and relative-permeability curves. The method has been validated and benchmarked on a large number of test cases that focus on on geological and petrophysical models with a high degree of realism, or on realistic flow physics on synthetic grid models designed to exemplify certain behavior. Selected results from four of these test cases were presented above. Altogether, these benchmark cases show that the MsMFE method is efficient, robust, and reasonably accurate compared to the fine-scale simulation and hence has a significant potential for accelerating simulation of two-phase flow applications, particularly for incompressible flow. Compared with coarse-scale models, the multiscale method gives a significant improvement of the accuracy and resolution of the flux, pressure, and saturation fields at a comparable computational cost. Combined with a large degree of robustness, this emphasizes

the importance of the MsMFE method for its ability to capture fine-scale heterogeneity.

The MsMFE method can also be extended to compressible flow and has a certain potential both for weakly and strongly compressible problems, including black-oil methods. Here, however, the formulation of the method hinges on an effective operator-splitting method for the underlying fine-scale problem. Although good results can be obtained in many cases, there is a need for more research to improve the robustness of the methods for application to practical simulation of black-oil models of industry-standard complexity.

5 Acknowledgements

The authors would like to thank the Carbonate Research Team, Shell International Exploration and Production B.V., Rijswijk, The Netherlands for allowing us to publish this work.

References

- Aarnes, J. E. 2004. On the use of a mixed multiscale finite element method for greater flexibility and increased speed or improved accuracy in reservoir simulation. *Multiscale Model. Simul.*, 2(3):421–439 (electronic).
- Aarnes, J. E. and Efendiev, Y. 2008. A multiscale method for modeling transport in porous media on unstructured corner-point grids. *J. Algorithms Comput. Technol.*, 2(2):299–318.
- Aarnes, J. E., Krogstad, S., and Lie, K.-A. 2008. Multiscale mixed/mimetic methods on corner-point grids. Comput. Geosci., 12(3):297–315.
- AGMG 2012. Iterative solution with AGgregation-based algebraic MultiGrid. http://homepages.ulb.ac.be/~ynotay/AGMG/.
- Alpak, F. O., Pal, M., and Lie, K.-A. 2011. A multiscale method for modeling flow in stratigraphically complex reservoirs. In SPE Reservoir Simulation Symposium, The Woodlands, TX, USA, 21–23 February 2011.
- Arbogast, T. 2002. Implementation of a locally conservative numerical subgrid upscaling scheme for two-phase Darcy flow. *Comput. Geosci.*, 6(3-4):453–481.
- Arbogast, T. and Bryant, S. L. 2002. A two-scale numerical subgrid technique for waterflood simulations. SPE J., 7(4):446-457.
- Audigane, P. and Blunt, M. J. 2004. Dual mesh method for upscaling in waterflood simulation. *Transp. Porous Media*, 55:71–89. 10.1023/B:TIPM.0000007309.48913.d2.
- Chen, A. and Hou, T. 2002. A mixed multiscale finite element method for elliptic problems with oscillating coefficients. *Math. Comp.*, 72(242):541–576.
- Chen, Y. and Durlofsky, L. J. 2006. Adaptive local-global upscaling for general flow scenarios in heterogeneous formations. *Transp. Porous Media*, 62(2):157–182.
- Chen, Y., Durlofsky, L. J., Gerritsen, M., and Wen, X.-H. 2003. A coupled local-global upscaling approach for simulating flow in highly heterogeneous formations. *Adv. Water Resour.*, 26(10):1041–1060.
- Efendiev, Y. and Hou, T. Y. 2009. *Multiscale Finite Element Methods*, volume 4 of *Surveys and Tutorials in the Applied Mathematical Sciences*. Springer Verlag.
- Farmer, C. L. 2002. Upscaling: a review. Int. J. Numer. Meth. Fluids, 40(1-2):63-78.
- Guerillot, D. and Verdiere, S. 1995. Different pressure grids for reservoir simulation in heterogeneous reservoirs. In SPE Reservoir Simulation Symposium, 12-15 February 1995, San Antonio, Texas, USA.
- Hajibeygi, H. and Jenny, P. 2009. Multiscale finite-volume method for parabolic problems arising from compressible multiphase flow in porous media. J. Comput. Phys, 228(14):5129 5147.
- Hajibeygi, H. and Jenny, P. 2011. Adaptive iterative multiscale finite volume method. J. Comput. Phys, 230(3):628-643.
- Hou, T. Y. and Wu, X.-H. 1997. A multiscale finite element method for elliptic problems in composite materials and porous media. *J. Comput. Phys.*, 134(1):169–189.
- Jenny, P., Lee, S. H., and Tchelepi, H. A. 2003. Multi-scale finite-volume method for elliptic problems in subsurface flow simulation. *J. Comput. Phys.*, 187:47–67.
- Kippe, V., Aarnes, J. E., and Lie, K.-A. 2008. A comparison of multiscale methods for elliptic problems in porous media flow. *Comput. Geosci.*, 12(3):377–398.
- Lamine, S. and Edwards, M. G. 2009. Higher order multidimensional wave oriented upwind schemes for flow in porous media on unstructured grids. In SPE Reservoir Simulation Symposium, The Woodlands, TX, USA, 2–4 February 2009.
- Lamine, S. and Edwards, M. G. 2010. Higher order multidimensional upwind convection schemes for flow in porous media on structured and unstructured quadrilateral grids. SIAM J. Sci. Comput., 32(3):1119–1139.

- Lee, S. H., Wolfsteiner, C., and Tchelepi, H. 2008. Multiscale finite-volume formulation for multiphase flow in porous media: Black oil formulation of compressible, three phase flow with gravity. *Comput. Geosci.*, 12(3):351–366.
- Lie, K., Krogstad, S., Ligaarden, I., Natvig, J., Nilsen, H., and Skaflestad, B. 2012. Open-source MATLAB implementation of consistent discretisations on complex grids. *Comput. Geosci.*, 16:297–322. 10.1007/s10596-011-9244-4.
- MoReS 2010. Shell's Reservoir Simulation Toolbox, version 2010. http://www.shell.com.
- Natvig, J. R., Lie, K.-A., Krogstad, S., Yang, Y., and Wu, X.-H. 2012. Grid adaption for upscaling and multiscale methods. In *Proceedings of ECMOR XIII–13th European Conference on the Mathematics of Oil Recovery*, Biarritz, France. EAGE.
- Notay, Y. 2010. An aggregation-based algebraic multigrid method. *Electron. Trans. Numer. Anal.*, 37:123–140.
- Pal, M. and Edwards, M. G. 2007. Quasimonotonic continuous darcy-flux approximation for general 3d grids of any element type. In SPE Reservoir Simulation Symposium, The Woodlands, TX, USA, 26–28 February 2007.
- Pal, M., Edwards, M. G., and Lamb, A. R. 2006. Convergence study of a family of flux-continuous, finite-volume schemes for the general tensor pressure equation. *Internat. J. Numer. Methods Fluids*, 51(9-10):1177–1203.
- Pal, M., Lamine, S., Lie, K.-A., and Krogstad, S. 2012. Multiscale methods for two and three-phase flow simulation in subsurface petroleum reservoirs. In ECMOR XIII Conference, Biarritz, France, 10–13 September 2012.
- Parramore, Elliot, T., Edwards, Michael, G., Pal, M., and Lamine, S. 2012. Cvd-mpfa based multiscale formulation on structured and unstructured grids. In ECMOR XIII Conference, Biarritz, France, 10–13 September 2012.
- SINTEF 2011. Matlab Reservoir Simulation Toolbox (MRST), version 2011a. http://www.sintef.no/MRST.

Wheeler, M. F., Xue, G., and Yotov, I. 2012. A multiscale mortar multipoint flux mixed finite element method. ESAIM, 46(4):759–796.