



Society of Petroleum Engineers

SPE 182701-PA

Use of Multiple Multiscale Operators to Accelerate Simulation of Complex Geomodels

K.-A. Lie, O. Møyner, SINTEF; J. R. Natvig, Schlumberger Information Solutions

Copyright 2017, Society of Petroleum Engineers

This paper was prepared for presentation at the SPE Reservoir Simulation Conference held in Montgomery, Texas, USA, 20-22 February 2017.

This paper was selected for presentation by an SPE program committee following review of information contained in an abstract submitted by the author(s). Contents of the paper have not been reviewed by the Society of Petroleum Engineers and are subject to correction by the author(s). The material does not necessarily reflect any position of the Society of Petroleum Engineers, its officers, or members. Electronic reproduction, distribution, or storage of any part of this paper without the written consent of the Society of Petroleum Engineers is prohibited. Permission to reproduce in print is restricted to an abstract of not more than 300 words; illustrations may not be copied. The abstract must contain conspicuous acknowledgment of SPE copyright.

Abstract

Multiscale methods have been developed as a robust alternative to upscaling and to accelerate reservoir simulation. In their basic setup, multiscale methods use a restriction operator to construct a reduced system of flow equations that can be solved on a coarser grid, and a prolongation operator to map pressure unknowns from the coarse grid back to the original simulation grid. When combined with a local smoother, this gives an iterative solver that can efficiently compute approximate pressures to within a prescribed accuracy and still provide mass-conservative fluxes. We present an adaptive and flexible framework for combining multiple sets of such multiscale approximations. Each multiscale approximation can target a certain scale; geological features like faults, fractures, facies, or other geobodies; or a particular computational challenge like propagating displacement and chemical fronts, wells being turned on or off, etc. Multiscale methods that fit the framework are characterized by three features. First, the prolongation and restriction operators are constructed by use of a non-overlapping partition of the fine grid. Second, the prolongation operator is composed of a set of basis functions, each of which has compact support within a support region that contains a coarse grid block. Finally, the basis functions form a partition of unity. These assumptions are quite general and encompass almost all existing multiscale (finite-volume) methods that rely on localized basis functions. The novelty of our framework is that it enables multiple pairs of prolongation and restriction operators – computed on different coarse grids and possibly also by different basis-function formulations – to be combined into one iterative procedure.

Through a series of numerical examples consisting of both idealized geology and flow physics as well as a geological model of a real asset, we demonstrate that the new iterative framework increases the accuracy and efficiency of the multiscale technology by improving the rate at which one converges the fine-scale residuals toward machine precision. In particular, we demonstrate how it is possible to combine multiscale prolongation operators having different spatial resolution and that each individual operator can be designed to target, among others, challenging grids including faults, pinchouts and inactive cells; high-contrast fluvial sands; fractured carbonate reservoirs; and complex wells.

Introduction

In reservoir simulation, a system of mass balance equations needs to be solved together with thermodynamics equations and appropriate constraints to determine the reservoir pressure and fluid composition. Each mass balance equation describes the evolution of one fluid species α in a porous medium Ω , in which multiple fluid species exist in M phases. When discretized in time and space, these equations form a system of nonlinear algebraic equations

$$F_\alpha(p, S_1, \dots, S_M, x_{\alpha,1}, \dots, x_{\alpha,M}) = \mathbf{0}. \quad (1)$$

Given a known pressure and fluid distribution at time t , Eq. 1 can be solved to determine the reservoir pressure p and distribution of fluid species (in terms of phase saturations S_ℓ and molar fractions $x_{\alpha,\ell}$) at time $t + \Delta t$. In particular, by manipulating the equation system Eq. 1, it is possible to form a nonlinear system of equations for the reservoir pressure p at time $t + \Delta t$,

$$F_p(p) = 0. \quad (2)$$

Such pressure equations appear in many commonly applied solution procedures within reservoir simulation including constrained pressure-residual (CPR) preconditioners for fully implicit methods or sequential solution methods like the implicit pressure explicit saturation (IMPES) method, the cascade method, and the sequential (fully) implicit method. To compute an approximate solution to Eq. 2, one usually linearizes around an initial guess p_0 and solves

$$A\delta p = -F_0, \quad (3)$$

to obtain a better estimate $p_1 = p_0 + \delta p$. Here, A is the Jacobian matrix of Eq. 2 and $F_0 = F_p(p_0)$. The procedure can be repeated until a sufficiently converged approximation to the pressure at time $t + \Delta t$ is obtained. One then computes volumetric fluxes so that the updated distribution of fluid species can be computed by solving Eq. 1. This usually involves a fractional formulation, in which each phase flux is expressed as a fraction of the total volumetric flux, and the saturation and fluid composition is evolved a time step Δt while keeping the pressure and volumetric fluxes fixed. If necessary, one can introduce an outer iteration over the pressure and the transport solves to guarantee a sufficiently small fine-scale residual of the total discrete system.

Altogether, this sequential solution procedure requires repeated solutions of Eq. 3, which can be challenging to solve directly because it represents an elliptic or near-elliptic equation that may have millions of unknowns for high-resolution geological models and may be quite ill-conditioned for realistic heterogeneities and reservoir geometries. To reduce the computational cost of solving Eq. 3, we will herein consider so-called multiscale methods (Hou and Wu 1997), which are a family of two-level solvers designed to efficiently provide pressures p_f that approximate the fine-scale pressure p to within a prescribed tolerance. The key idea is to define a coarse partition of the simulation grid, and compute a set of locally defined basis functions that map between degrees of freedom associated with cells in the fine grid and blocks in the coarse partition. By use of these basis functions in combination with an appropriate restriction operator that e.g., sums the flow equations over all cells within each coarse bloc, one can derive reduced flow problems on the coarse grid in a systematic manner. The resulting solvers are called 'multiscale' since they were originally developed to solve elliptic problems with variable coefficients having multiscale heterogeneity with no clear scale separation.(The textbook by Efendiev and Hou (2009) gives a thorough introduction to the mathematical concepts and the early literature on these multiscale methods.) As long as the pressure equation is only solved exactly on the coarse partition, one cannot generally expect that the prolongation of this solution onto the underlying fine grid fullfills the flow equations exactly. Usually, it is considered more important to ensure exact mass-balance than fulfilling Darcy's law exactly. Multiscale methods therefore typically sacrifice the exact correspondence between pressure and fluxes on the fine scale to gain computational efficiency, but also contain some iterative mechanism that can reduce the mismatch to within a prescribed tolerance, if this is deemed necessary. Over the past decade, a large variety of multiscale methods have been introduced and extended to accurately account for challenging modeling features such as high-contrast rock heterogeneity, channels and fractures, complex wells, and locally increased grid resolution. Rather than reviewing the large body of literature, we refer the reader to Lie et al. (2017b), who present an extensive literature review focusing on the most industry-relevant research and discuss in detail the state-of-the-art as implemented in a commercial simulator environment.

Existing multiscale methods use a single coarse grid with either a single or a low number of basis functions associated with each coarse block or each interface between two coarse blocks. This may not be sufficient to provide desired accuracy and convergence rate for systems with very strong contrasts in the petrophysical properties. Many different approaches have therefore been suggested to improve the approximation properties of multiscale operators: Efendiev et al. (2011) proposed to use eigenvectors of local spectral problems to systematically enrich the approximation space of the multiscale finite-element method. Dolean et al. (2014) showed that a more compact set of eigenvectors could be obtained by using used a Dirichlet-to-Neumann. Lie et al. (2014) presented an enriched version of the multiscale mixed finite-element method, which enlarges the approximation space by subdividing the faces between coarse blocks to adapt to strong media contrasts. The generalized multiscale finite element method (Efendiev et al. 2013; Bush et al. 2014; Chung et al. 2014) provides a more systematic approach that divides the computation into an offline and an online stage. The offline stage solves a series of representative snapshots and reduces the results to a small-dimensional space through spectral decomposition. This offline space is then used to generate new basis functions for the particular parameter combinations encountered during the dynamic simulation (online stage). Akkutlu et al. (2017) recently proposed a very interesting extension of this method, in which the enriched multiscale space is used to systematically identify fracture networks and thereby avoid many of the limitations of multi-continuum models.

Several approaches have also been suggested to increase the resolution of specially challenging heterogeneities for multiscale finite-volume methods (Jenny et al. 2003). Cortinovis and Jenny (2014) proposed to include extra nodal basis functions and local enrichments combined in a hybrid finite-volume/Galerkin formulation. This method has recently been extended to a so-called *zonal* multiscale method, which introduces extra zonal functions in regions of special interest so that the zonal functions, their extension to the surrounding domain, and the original multiscale basis functions form a partition of unity (Cortinovis and Jenny 2017). In the enriched algebraic multiscale method, Manea et al. (2016, 2017) use an error equation to identify solution modes that are missing from the multiscale operator and then enrich the coarse space with local basis functions that target the largest error components. Finally, Künze et al. (2013) proposed a three-level multiscale finite-volume method to handle models with a large number of cells, whereas Cusini et al. (2016) developed a dynamic multilevel method for fully implicit discretizations.

Our purpose herein is to present an alternative and very general framework. Instead of employing a coarse space associated with a single coarse grid to approximate all relevant features of a simulation model, the framework enables the combination of multiscale operators computed for multiple coarse grids that each either cover the whole domain evenly or has resolution tailored to a particular set of model features. As such, the proposed idea is a truly multiscale approach. The framework can be motivated in different ways. First of all, the framework can be seen as a way to incorporate the ideas of multiple coarse grids from multilevel solvers into a multiscale setting. Secondly, it makes the multiscale methods more robust to user-defined parameters. State-of-the-art multiscale prolongation operators like those computed by the multiscale restriction-smoothed-basis (MsRSB) method (Møyner and Lie 2016b) are not very sensitive to the coarse partitioning and provide accurate approximations across a wide range of different partitions. The degree of coarsening can therefore be chosen by the user to balance the cost of computing

basis functions and the cost of solving the reduced coarse-scale problems. Unfortunately, it is generally difficult to predict the coarsening ratio that will give the most effective error reduction upfront. By combining multiple partitions, which each may be suboptimal in terms of error reduction, one gets a faster reduction in fine-scale residuals.

Previous literature on enrichment methods has primarily focused on methods that detect the need for enrichment automatically. In many cases, however, the informed user may wish to distinguish these features using properties and objects from the underlying geological model. Our new framework is designed to enable expert users to explicitly request better approximation of complex geometry and high-contrast geology with long correlation lengths such as river beds, channels and fractured zones, fault zones with uncertain properties, etc. Likewise, one can introduce extra resolution in the multiscale operator near wells or other types of high-flow zones. A large number of numerical experiments reported in previous literature show that the accuracy of multiscale approximations can be improved if the coarse partition adapts to such geological features. However, determining exactly which features to adapt to and how the resulting partition should be constructed is more an art than a science. Adapted partitions also tend to give highly irregular coarse blocks, which may cause numerical artifacts in various special (corner) cases. Our hypothesis is that working with multiple partitions having different approximation properties will increase both the fidelity and the robustness of the multiscale corrections in an iterative method.

Through a series of numerical examples, we demonstrate that this is indeed the case. Combining sets of multiscale basis functions that each may target specific features in the reservoir model, possibly complemented with (more) regular partitions, gives a multiscale method with better overall approximation and convergence properties. Herein, we primarily focus on static partitions defined *a priori*, but the framework (and our prototype implementation) also allows for dynamic partitions that change to follow distinct features in the solution such as displacement fronts, chemical slugs, etc. Likewise, the solution algorithms can easily be configured to recompute local basis functions to account for dynamic changes in multiphase mobilities.

Background: Multiscale Finite-Volume Methods

In this section, we give a quick introduction to multiscale finite-volume methods. To simplify the discussion, we consider the incompressible pressure equation with gravity neglected,

$$\nabla \cdot \vec{v}_t = q_t, \quad \vec{v}_t = -\mathbf{K} \lambda_t \nabla p, \quad (4)$$

which we assume is discretized by the standard two-point flux approximation method

$$\sum_{f \in \partial \Omega_i} T_f (\mathbf{p}_{\mathcal{N}_2(f)} - \mathbf{p}_{\mathcal{N}_1(f)}) = \mathbf{q}_i \quad (5)$$

on a fine grid $\{\Omega_i\}_{i=1}^n$ holding the petrophysical properties, boundary conditions, well models, etc. Here, f denotes an interface between two cells $\mathcal{N}_1(f)$ and $\mathcal{N}_2(f)$, with a sign convention that results in positive transmissibilities T_f . The resulting fine-scale problem takes the form of a set of linear equations for the pressure in each cell,

$$A\mathbf{p} = \mathbf{q}. \quad (6)$$

Most multiscale methods for Eq. 6 described in the literature start by introducing a non-overlapping coarse partition $\{\bar{\Omega}_j\}_{j=1}^m$ of the fine grid, so that each cell Ω_i in the fine grid belongs to a single block $\bar{\Omega}_j$ in the coarse grid. One then introduces a set of basis functions that map degrees of freedom associated with the coarse grid to degrees of freedom on the fine grid. To ensure that this mapping is locally consistent with the properties of the differential operators in the flow equation, the basis functions are computed numerically by solving localized flow problems. Although the basis functions can be set up to map both pressures and velocities/fluxes, we will herein only consider finite-volume type methods solving for pressure as the primary variable. In this case, the basis functions can be collected into a numerical *prolongation operator* that maps pressure unknowns from the coarse to the fine grid, $P : \{\bar{\Omega}_j\} \rightarrow \{\Omega_i\}$. We also define an analogous *restriction operator* that maps in the opposite direction $R : \{\Omega_i\} \rightarrow \{\bar{\Omega}_j\}$. These operators are represented as matrices P and R of size $n \times m$ and $m \times n$, respectively. Figure 1 illustrates pairs of (P, R) operators resulting from two different partitions of a rectangular grid.

By use of the prolongation operator, one can define an approximate fine-scale pressure $\mathbf{p}_f = P\mathbf{p}_c$ on Ω from any pressure \mathbf{p}_c defined on $\bar{\Omega}$. If we insert the approximate fine-scale pressure into Eq. 3, we get $(AP)\mathbf{p}_c = \mathbf{q}$. This system contains more equations than unknowns, and consequently we left-multiply by the restriction operator R to obtain a square linear system for the approximate pressure,

$$(RAP)\mathbf{p}_c = R\mathbf{q} \longleftrightarrow A_{ms}\mathbf{p}_c = \mathbf{q}_c. \quad (7)$$

The physical interpretation of this coarse system depends on the choices made for P and R . Examples of P include the multiscale finite-volume (MsFV) operator (Jenny et al. 2003), the multiscale two-point (MsTPFA) operator (Møyner and Lie 2014), or the multiscale restriction-smoothed-basis (MsRSB) operator (Møyner and Lie 2016b), which can all be obtained by fully algebraic constructions. Figure 2 shows details of how prolongation operators are computed with the MsRSB method. For the restriction operator, one choice is to set $R = P^T$, which corresponds to a Galerkin coarse-scale discretization, which unfortunately is not

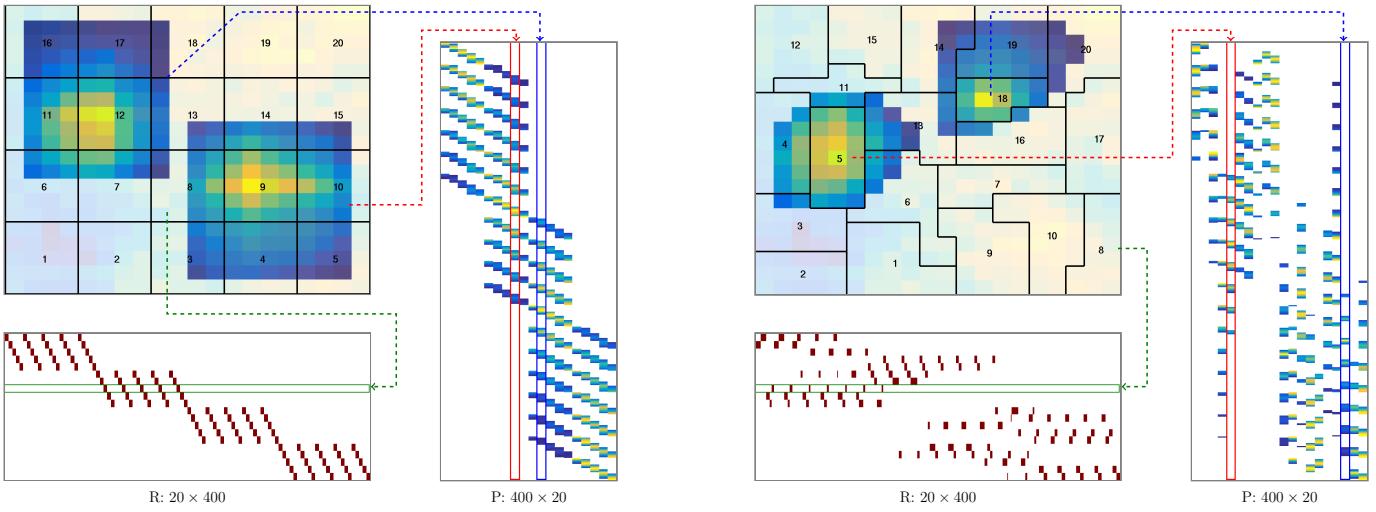


Figure 1—Illustration of two sets of prolongation and restriction operators for two different partitions with 20 blocks each for a simple 20×20 Cartesian grid. The prolongation operators are obtained by solving local flow problems. The restriction operators are defined by setting element i in row j equal to one if cell i belongs to block j .

Coarse grid: 3×3 partition

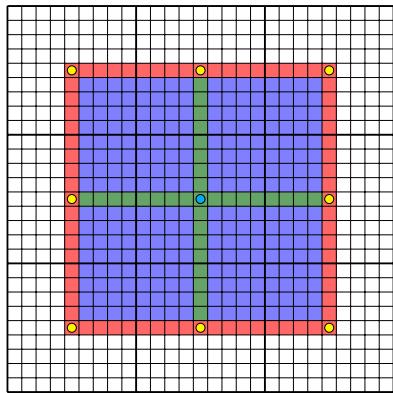


Figure 2—Algebraically smooth basis functions computed by MsRSB. Each basis function is associated with a coarse block (thick lines) and is nonzero inside a support region (blue color), bounded by a perimeter of cells (red). A simple Jacobi iteration to gradually adapt P_j to the underlying heterogeneity. To prevent P_j from growing outside its support region, the increments are set to zero on the perimeter (red cells). Partition of unity is enforced by normalizing in the green cells representing the perimeter of neighboring blocks.

mass conservative. Alternatively, one can define R as the characteristic function of each block, so that it sums all cell quantities inside each block. This corresponds to a finite-volume approximation and is generally mass conservative.

Once an approximate coarse-scale pressure has been computed, we can use P to back an approximate pressure solution defined on the underlying fine-scale grid

$$\mathbf{p}_f = P\mathbf{p}_c = PA_{ms}^{-1}\mathbf{q}_c = P(RAP)^{-1}R\mathbf{q}.$$

Unfortunately, the corresponding fluxes will not be mass conservative, which means, for instance, that $\nabla \cdot \vec{v}_t = q_t$ is not fulfilled in a discrete sense for an incompressible flow. To get mass-conservative fluxes on the fine scale, we must introduce an additional *reconstruction step* that solves a local flow problem with Neumann boundary conditions sampled from the multiscale solution, unless the multiscale solver is only used as a coarse-scale solver (Kippe et al. 2008; Cusini et al. 2016). As explained so far, the multiscale method will act as a robust upscaling method that computes *approximate*, mass-conservative flow solutions on the fine, the coarse, or any intermediate grid, and is able to incorporate effects of sub-scale heterogeneities in a systematic sense. Although these solutions tend to resolve the global couplings in the system quite well, they also contain local errors stemming from the localization introduced to define basis functions. In other words, the low-frequency (long-range) error modes in the solution will be small, but the high-frequency (short-range) error modes will not.

Set P_j to one inside block j

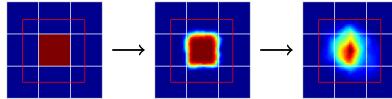
Jacobi increment: $d_j = -\omega D^{-1}AP_j^n$

Localize update:

$$\hat{d}_{ij} = \begin{cases} \frac{d_{ij} - P_{ij}^n \sum_k d_{ik}}{1 + \sum_k d_{ik}} \\ \textcolor{blue}{d_{ij}} \\ 0 \end{cases}$$

Apply increment: $P_{ij}^{n+1} = P_{ij}^n + \hat{d}_{ij}$

Indices: $i = \text{cell}$, $j = \textcolor{blue}{\bullet}$, $k = \textcolor{yellow}{\bullet}$



To get a solver that is able to compute the fine-scale solution of Eq. 6 to within a prescribed residual distance – that is, also reduce the high-frequency error components – we need to cast the multiscale method in an iterative framework. One possibility is to define a Richardson iteration

$$\mathbf{p}^{k+1} = \mathbf{p}^k + \omega^k A_{ms}^{-1}(\mathbf{q} - A\mathbf{p}^k). \quad (8)$$

It is also possible to use A_{ms}^{-1} as a preconditioner for GMRES. Herein, we primarily consider two-stage algebraic multiscale solvers (Wang et al. 2014), in which A_{ms} is employed as a global preconditioner in combination with a local smoother $W(A, \mathbf{b})$, i.e., any kind of inexpensive iterative solver that efficiently removes high frequency errors from the solution. Examples of possible smoothers include incomplete LU-factorization with zero or low degree of fill-in or standard iterative solvers including variants of Gauss-Seidel or Jacobi's method. With this in mind, we can define a two-step preconditioner that first removes local error by use of the smoother and then computes a coarse scale correction,

$$\mathbf{x}^{k+1/2} = \mathbf{x}^k + W(A, \mathbf{q} - A\mathbf{x}^k), \quad (9)$$

$$\mathbf{x}^{k+1} = \mathbf{x}^{k+1/2} + PA_{ms}^{-1}R(\mathbf{q} - A\mathbf{x}^{k+1/2}). \quad (10)$$

The method above can easily be extended to more complex physics including both black-oil and compositional methods (Møyner and Lie 2016a; Hajibeygi and Tchelepi 2014; Møyner and Tchelepi 2017), and is essentially what is implemented as a prototype solver in a commercial environment (Lie et al. 2017b; Kozlova et al. 2016).

The two-stage multiscale preconditioner will typically have a satisfactory initial convergence rate, which eventually deteriorates when the remaining modes of the error are neither captured by the multiscale operator nor the local smoother. These intermediate modes can often be neglected and for many cases it is sufficient to reconstruct a conservative velocity field after a moderate number of iterations. Mathematically, two-stage multiscale solver is equivalent to the typical two-stage methods used in the construction of algebraic multigrid solvers (Vaněk et al. 1996). Philosophically, the multiscale method is different from classical multigrid methods, which use hierarchical grids constructed to accurately resolve all error modes by use of a multistage solver and converge to a strict tolerance.

For cases with complex geology it may not be sufficient to only use the multiscale operator along with a local stage to accurately capture the flow in an efficient manner. The convergence may stagnate before the desired tolerances are reached and a large number of iterations may be required to resolve the fluid displacements in the subsequent transport problem. In the next section, we will therefore extend the multiscale framework with additional basis functions to ensure favorable problem-specific convergence rates, without increasing the size of the coarse systems that are the bottleneck for scaling to parallel systems. By use of operators from different coarse grids, one can hope to reduce the number of intermediate modes for cases when these are not negligible.

Multiscale solver with multiple basis sets

We introduce the notion of multiple basis functions, so that we have N different prolongation operators P^1, \dots, P^N , where each P^ℓ contains an individual set of basis functions corresponding to a distinct coarse grid $\{\bar{\Omega}_j^\ell\}_{j=1}^{m_\ell}$. Likewise, we will also associate a set of restriction operators, R^1, \dots, R^N . To fit to the framework, the individual operators P^ℓ and R^ℓ must fulfill the following three requirements:

1. The prolongation operator P^ℓ for pressure and the restriction operator R^ℓ are constructed over non-overlapping partitions of the fine grid. Each column j in P^ℓ is referred to as a basis function and is associated with a coarse grid block $\bar{\Omega}_j^\ell$.
2. The support of each basis function is compact and must contain the associated coarse block. Hence, for the *support region* S_j^ℓ of the j th basis function, we have $\bar{\Omega}_j^\ell \subset S_j^\ell \subset \cup_{j=1}^{m_\ell} \bar{\Omega}_j^\ell$.
3. The columns of P^ℓ form a partition of unity over the fine grid; that is, each row in P^ℓ has unit row sum.

The N multiscale operators can all have different choices of coarse grids and support regions, and can also come from different types of multiscale methods. For each multiscale operator, the initial coarse blocks, support regions, and basis functions are assumed to be created at the start of the simulation or in a preprocessing (offline) stage. Examples of multiscale methods that fit this description include the MsFV (Jenny et al. 2003), MsTPFA (Møyner and Lie 2014), and MsRSB Møyner and Lie (2016b) methods. Of these, the MsRSB method is most robust and by far the simplest to implement for complex grids and coarse partitions. This will therefore be our method of choice in the subsequent numerical experiments. Notice that the memory cost of introducing additional multiscale operators is modest. Since each basis function only has support within a localized support region, the basis functions can be stored compactly with a memory requirement limited upward by the number of fine-scale cells times the maximum number of support regions overlapping a single cell. For a rectangular partition of a 2D Cartesian grid with n cells, for instance, this would amount to n integers to represent the partition and approximately $4n$ integers and $4n$ doubles to represent the support regions and nonzero elements of the associated basis functions. Feature-specific operators tend to have very limited overlap in support regions and hence give very modest increase in memory consumption.

If we let $B^\ell = P^\ell(R^\ell AP^\ell)^{-1}R^\ell$ denote the multiscale solution operator corresponding to the ℓ -th set of basis functions and let W^ℓ denote the corresponding local stage solver (smoother), then successive application of the different multiscale and local stages can be written as the multiplicative multistep method,

$$\mathbf{x}^{k+\ell*} = \mathbf{x}^{k+\ell-1} + W^\ell(A, \mathbf{d}^{k+\ell-1}) \quad (11)$$

$$\mathbf{x}^{k+\ell} = \mathbf{x}^{k+\ell*} + B^\ell \mathbf{d}^{k+\ell*}, \quad (12)$$

where the defect is defined as $\mathbf{d} = \mathbf{q} - A\mathbf{x}$ and k is an integer multiple of N . Here, we have let the iteration count increase by N if the multistage algorithm cycles through N different multiscale operators to simplify subsequent comparisons with the standard iterative MsFV and MsRSB methods. The presentation can be made more transparent if we consider the homogeneous case with $\mathbf{q} = \mathbf{0}$ and assume that the application of each smoother corresponds to a multiplication with a matrix W^ℓ . Then the multistep method reads,

$$\begin{aligned} \mathbf{x}^{k+N} &= B^N W^N \mathbf{x}^{k+N-1} = [B^N W^N] [B^{N-1} W^{N-1}] \mathbf{x}^{k+N-2} = \dots = [B^N W^N] \cdots [B^1 W^1] \mathbf{x}^k \\ &= [P^N (R^N AP^N)^{-1} R^N] W^N \cdots [P^1 (R^1 AP^1)^{-1} R^1] W^1 \mathbf{x}^k. \end{aligned} \quad (13)$$

The multiscale preconditioners can also be combined in other ways, e.g., in an additive manner or in sequence without the application of a smoother between each multiscale operator. In the following, we will always include at least one set of *general* basis functions that cover the domain relatively evenly and constitute a reasonably accurate multiscale solver on their own. In addition, we will include *specific* basis functions that, through coarsening strategies or a specialized basis construction, aim to capture specific features of the problem not captured by the general basis functions. The benefit of this is that key features in the reservoir model affecting reservoir pressure – such as faults, fractures, wells that gets turned on or exhibit large changes in well controls – can all be addressed by distinct fit-for-purpose multiscale operators in a very flexible manner. If necessary, the basis functions and the shape and extent of coarse grid blocks and support regions can also be updated between the application of each multiscale preconditioner B^ℓ to reflect changes in driving forces and in reservoir and fluid properties. The fact that each prolongation operator forms a partition of unity makes it very easy to enable or disable individual multiscale preconditioners during the course of a simulation run. This would be hard to achieve if all the different and specialized basis functions were lumped together in a single multiscale operator that would need to be recalibrated to ensure partition of unity whenever a particular basis function is added or removed.

The coarse partitions can be formed in many different ways. Partition methods include, but are not restricted to:

- Partitions formed by rectilinear or structured subdivision in physical space or in index space.
- (Un)structured mesh partitions generated by graph partitioning algorithms like in Metis (Karypis and Kumar 1998) that optionally can use transmissibilities or other measures of connection strength to ensure that each coarse block has as homogeneous rock properties as possible, see e.g., (Møyner and Lie 2016b).
- Partitions adapting to geological features such as facies, rock types, saturation regions, geological layers, faults, etc, or to other meta-information coming from the geological modeling tool, see e.g., (Hauge et al. 2012; Hauge 2010).
- Partitions arising from block-structured gridding, local grid refinement (LGR), or local coarsening.
- Partitions resulting from agglomeration of grid cells based on indicator functions set by the user, derived from the geological model, or from a preexisting simulation to give flow-adapted coarse grids, see e.g., (Hauge et al. 2012; Hauge 2010; Lie et al. 2017a)
- Partitions based on error measures/indicators, adjoint sensitivities, etc.
- Partitions adapting dynamically to evolving flow fields, saturations and compositions.

A few examples of such partitions are shown in Figure 3. For brevity, the different partitioning methods will not be discussed further. However, we will discuss how to improve the approximation of wells in some more detail.

Well basis functions By default, the global viscous forcing from wells are only accounted for on the coarse grid in a multiscale method, which may introduce quite inaccurate approximations, especially in the important near-well region on the fine-scale grid. This can be mitigated either by refining the coarse grid near the wells (Møyner and Lie 2016b), or by use of correction functions that are constructed analogously to the basis functions themselves, with source terms included (Wolfsteiner et al. 2006; Skaflestad and Krogstad 2008; Jenny and Lunati 2009). Both these approaches are somewhat limiting in that the choice of coarse grid for the basis functions significantly influences the quality of the pressure approximation near the source terms.

To treat well terms and other singularities in a more independent manner, we introduce a set of well basis functions, with a corresponding coarse grid determined *a posteriori* from the near-well pressure profile. For a problem with n_w different wells,

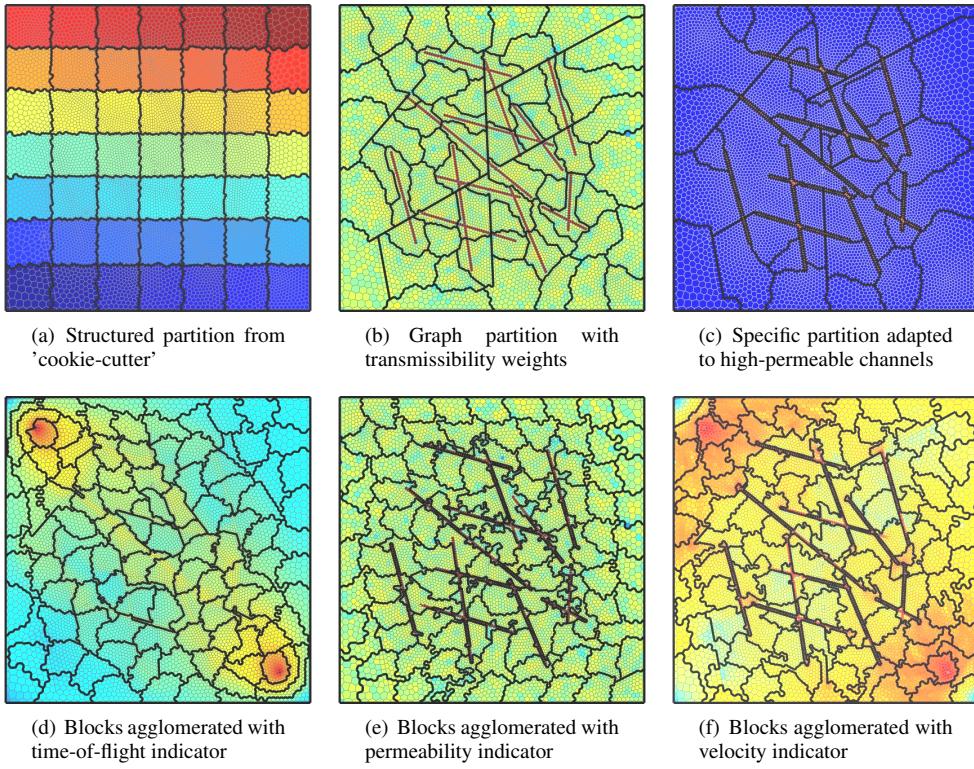


Figure 3—Illustration of various types of coarse partitions for an unstructured Voronoi grid with sealing faults and high-permeable channels.

we define a local support region for each well W_ℓ consisting of all cells within a prescribed distance from the well. The distance can either be the edge distance in a discrete cell graph, physical distance between well and cell centroids, or a more sophisticated measure such as time-of-flight, if available. We proceed to solve local problems in each subdomain with zero boundary conditions and each well control set to unitary bottom-hole-pressure. This gives a set of discrete pressure responses p^ℓ for each well ℓ , which are defined such that p_i^ℓ is zero for cells Ω_i outside the support region and nonzero for cells inside. The pressure response, limited to the support region, is then rescaled so that the maximum pressure is equal to unity. To assemble a prolongation operator from these local solutions, we apply a scaling factor such that $\sum_{\ell=1}^{n_w} p_i^\ell \leq 1$ for all cells Ω_i if different wells are in close proximity of each other. We also add another basis function that has global support throughout the whole reservoir. We can then define the well basis

$$P^w = \left[P_1^w, P_2^w, \dots, P_{n_w}^w, 1 - \sum_{\ell=1}^{n_w} P_\ell^w \right]. \quad (14)$$

Since each individual solution is limited to $[0, 1]$ and their sum is always less than or equal unity, we have a partition-of-unity with $n_w + 1$ basis functions. To define a coarse grid for this prolongation operator, we use the same trick as in (Møyner and Lie 2016b) and assign each fine cell to the coarse block whose basis function has the largest value in this cell.

Efficient flux reconstruction The flux-reconstruction step of a finite-volume multiscale solver relies on the physical interpretation of the coarse scale system A_{ms} when constructed with the control-volume restriction operator,

$$(R)_{ji} = \begin{cases} 1, & \text{if } \Omega_i \subset \bar{\Omega}_j, \\ 0, & \text{otherwise.} \end{cases}$$

The resulting coarse system corresponds to a coarse-scale finite-volume discretization with fluxes induced by the prolongated pressure field. If we consider the continuous form, this can be stated as

$$\int_{\partial\bar{\Omega}_j} \vec{v} \cdot \vec{n} dA = \int_{\bar{\Omega}_j} q dV, \quad \vec{v} = -\mathbf{K} \nabla p \quad \forall j \in \{1, \dots, m\}. \quad (15)$$

Equivalently, the coarse-scale equation for $\bar{\Omega}_j$ resulting from the application of the finite-volume restriction operator equals the sum of the conservation equations in Eq. 5 for each fine cell Ω_i in $\bar{\Omega}_j$,

$$\sum_{\Omega_i \subset \bar{\Omega}_j} \sum_{f \subset \partial\Omega_i} T_f [(\mathbf{p}_{ms})_{\mathcal{N}_2(f)} - (\mathbf{p}_{ms})_{\mathcal{N}_1(f)}] = \quad (16)$$

$$\sum_{f \subset \partial\bar{\Omega}_j} T_f [(\mathbf{p}_{ms})_{\mathcal{N}_2(f)} - (\mathbf{p}_{ms})_{\mathcal{N}_1(f)}] = \sum_{\Omega_i \subset \bar{\Omega}_j} \mathbf{q}_i \quad \forall j \in \{1, \dots, m\}. \quad (17)$$

Because all fluxes between cells inside the same coarse block cancel, the coarse-scale equations give a conservative finite-volume scheme. Hence, the conservative fluxes on $\bar{\Omega}$ can be used as boundary conditions to compute conservative fine-scale fluxes inside each coarse block. Following Lunati and Lee (2009), we define the matrix for the reconstruction problem from the fine-scale linear system as

$$(D)_{kl} = \begin{cases} (A)_{kl}, & \text{if } \Omega_k \text{ and } \Omega_l \text{ belong to the same coarse block,} \\ 0, & \text{otherwise,} \end{cases} \quad (18)$$

The problem for the reconstructed pressure $\bar{\mathbf{p}}$ is then a linear system with boundary conditions estimated from the multiscale pressure,

$$D\bar{\mathbf{p}} = \mathbf{q} - (A - D)\mathbf{p}_{ms} = \bar{\mathbf{q}}. \quad (19)$$

This linear system is in reality a collection of independent, local subproblems. In practice, however, these local problems may still take up a large fraction of the overall simulation time as they must be inverted at the end of each pressure solve if one subsequently wants to solve a transport problem at the fine scale.

By use of multiple sets of basis functions, it is possible to significantly reduce the cost of reconstructing the fluxes. Assume that our method has two multiscale operators with coarse grids $\bar{\Omega}^1$ and $\bar{\Omega}^2$, restriction operators R^1 and R^2 , and prolongation operators P^1 and P^2 . After the second multiscale operator has been applied, \mathbf{p}_{ms} gives conservative fluxes across the interfaces of $\bar{\Omega}^2$, but *not* across the interfaces of $\bar{\Omega}^1$ (unless the interfaces of these two coarse partitions coincide). To construct mass-conservative fine-scale fluxes, we need to solve a series of local flow problems, one for each block $\bar{\Omega}_j^2$, defined as

$$\tilde{D}_j \tilde{\mathbf{p}}_j = \tilde{\mathbf{q}}_j. \quad (20)$$

However, instead of solving these independent problems directly, we first apply the multiscale operator corresponding to $\bar{\Omega}^1$ inside $\bar{\Omega}_j^2$,

$$\tilde{\mathbf{p}}_1 = \tilde{P}^1(\tilde{R}^1 \tilde{D}_j \tilde{P}^1)^{-1} \tilde{R}^1(\tilde{\mathbf{q}}_j - \tilde{D}_j \tilde{\mathbf{p}}_{ms}) + \tilde{\mathbf{p}}_{ms}.$$

where \tilde{P}^1 signifies the restriction of P^1 to $\bar{\Omega}_j^2$. By the same argument as in Eq. 17, this updated pressure $\tilde{\mathbf{p}}_1$ results in mass-conservative fluxes over the (coarse) interfaces of $\bar{\Omega}^1$ inside $\bar{\Omega}_j^2$. Consequently, after this multiscale approximation, we have obtained mass-conservative fluxes with subscale variation on all interfaces of the intersection of $\bar{\Omega}^1$ and $\bar{\Omega}^2$, which results in much smaller subdomains as seen in Figure 4. For each of these new subdomains, the local matrix is defined analogously to the original reconstruction problem,

$$(G)_{kl} = \begin{cases} (D)_{kl}, & \text{if } \Omega_k \text{ and } \Omega_l \text{ belong to the same coarse block pair } \bar{\Omega}^1, \bar{\Omega}^2, \\ 0, & \text{otherwise,} \end{cases} \quad (21)$$

which has additional boundary conditions applied from the intermediate multiscale approximation, resulting in the final linear system,

$$G\hat{\mathbf{p}} = \hat{\mathbf{q}} = \bar{\mathbf{q}} - (D - G)\mathbf{p}_1 = \mathbf{q} - (A - D)\mathbf{p}_{ms} - (D - G)\mathbf{p}_1. \quad (22)$$

The process described in Eq. 20 to Eq. 22 can be repeated to account for more than two multiscale operators if needed. The fine-scale flux field is recovered by computing fluxes on $\partial\bar{\Omega}^2$ from \mathbf{p}_{ms} , fluxes on $\partial\bar{\Omega}^1$ from \mathbf{p}_1 , and the remaining fine-scale fluxes from $\hat{\mathbf{p}}$. The interested reader should consult Künze et al. (2013) for a discussion of the special case when the two coarse partitions are nested.

For fully unstructured grids, the intersection of several different grids may result in blocks that are not contiguous. When this occurs, we merge the cells having the lowest value for the corresponding basis functions into the largest neighbor. Figure 4(c) shows a case taken from one of the numerical examples discussed in the next section. Here, the intersection of two unstructured partitions result in a block being subdivided into two disconnected components, outlined in black. In this case, the single isolated block will merge with the neighboring large blue block.

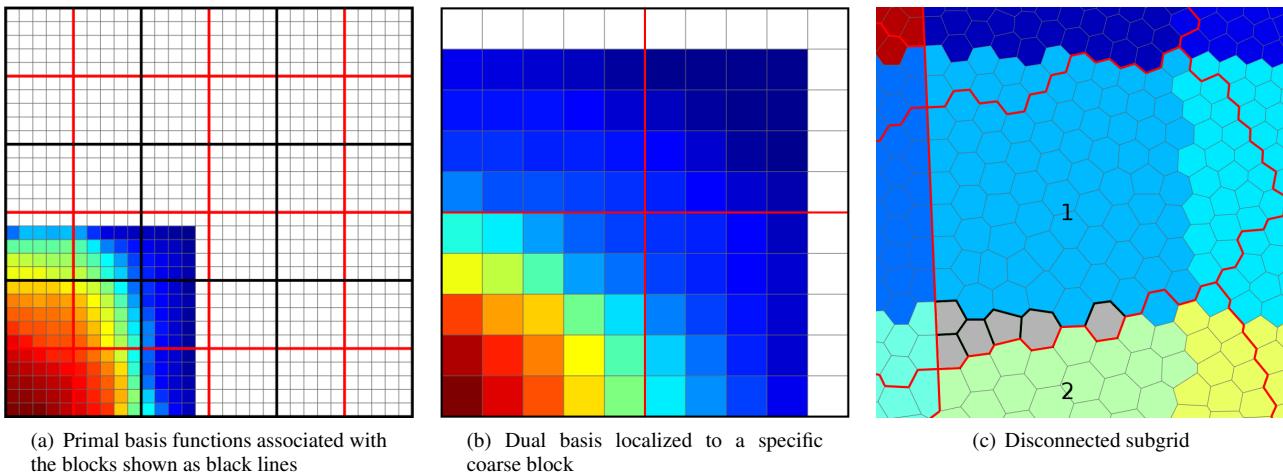


Figure 4—Illustration of how the reconstruction can be subdivided into smaller problems. The left plot shows how the first set of basis functions associated with the primal blocks drawn as black lines is used to find fluxes that are coarse-scale conservative over the primal grid, whereas the middle plot shows how the second set of basis functions associated with dual blocks drawn as red lines are used locally within each block to further reduce the reconstruction to the intersection of the primal and dual blocks. The right plot shows a case with two intersection partitions, one shown by color and one by red lines. The cells marked in gray are part of block 2 in the first partition, but after the intersection they form a new subdomain consisting of two disconnected parts. Here, the single isolated cell is merged into the neighboring block 1 shown in light blue color.

Numerical examples

In the following, we will report a series of experiments set up to illustrate that the framework is flexible with respect to various types of coarse partitions and to examine the accuracy of the resulting multiscale solvers and how efficient they reduce the fine-scale residual toward machine precision. In all experiments, we use prolongation operators constructed by the MsRSB method (Møyner and Lie 2016b) and finite-volume restriction operators. These have been implemented by use of the open-source Matlab Reservoir Simulation Toolbox (MRST), see (Lie et al. 2012; Lie 2016), and our implementation is a generalization of the method incorporated into a commercial simulator environment (Lie et al. 2017b; Kozlova et al. 2016). Although our solvers are capable of running problems with industry-standard flow physics, including both black-oil and compositional models, we will for simplicity only present cases with incompressible single-phase flow or two-phase flow without capillary forces.

Layers of SPE 10 Horizontal layers of Model 2 from the 10th SPE Comparative Solution Project (Christie and Blunt 2001) seem to have become a *de facto* benchmark that is seen in virtually any paper on multiscale methods. The overall SPE 10 model is described by a $60 \times 220 \times 85$ Cartesian grid with cells of uniform size $20 \times 10 \times 2$ ft³ with heterogeneity sampled from a Brent sequence, as seen in the North Sea. The upper 35 layers are from a shallow-marine Tarbert formation, which has a relatively smooth heterogeneity with permeabilities following a lognormal distribution. The fluvial Upper Ness formation found in the bottom 50 layers consists of an intertwined pattern of long and high-permeable sand channels interbedded with low-permeable sandstone. To test the accuracy of our multiscale framework, we pick the bottom layer of the Tarbert formation, which is less smooth than most of the upper layers, as well as the bottom layer of the Upper Ness formation. The flow physics is two-phase flow with linear relative permeabilities and equal viscosities. The domain is initially filled with a nonwetting fluid and a wetting fluid is injected along the north side of the model, driven by a unit pressure drop from the north to the south side.

For the Tarbert layer, we introduce two different rectangular partitions which are set up like the primal and dual partitions typically used in the original MsFV method. The primal grid has 6×11 blocks, with smaller block sizes at the perimeter, whereas the dual grid has 5×10 blocks of uniform size. The upper-left plot in Figure 5 shows the two partitions plotted on top of the permeability field. The next four plots in the upper row show the initial fine-scale pressure and approximate pressures computed by the multiscale method with primal partition, dual partition, and alternating partitions. We observe that the dual grid gives a systematic error as a result of not using smaller blocks near the inflow and outflow boundaries. The primal partition and the alternating combination of the two partitions gives approximate solutions that are visually similar and have comparable discrepancies in the L² norm, see Table 1. Not surprisingly, we observe a somewhat lower pointwise discrepancy when the combined set of basis function from the primal and dual grid are used, as each spans different parts of the fine-scale pressure space. For the Upper Ness layer, we use the 6×11 coarse grid to construct the first prolongation operator. For the second prolongation operator, we first use the permeability to segment the domain into two regions representing the high-permeable sand channels and the low-permeable mudstone, and then use Metis to partition each region separately. This gives an unstructured grid with irregular coarse blocks adapting to strong contrasts in the permeability. Although this may seem to be a good idea, the

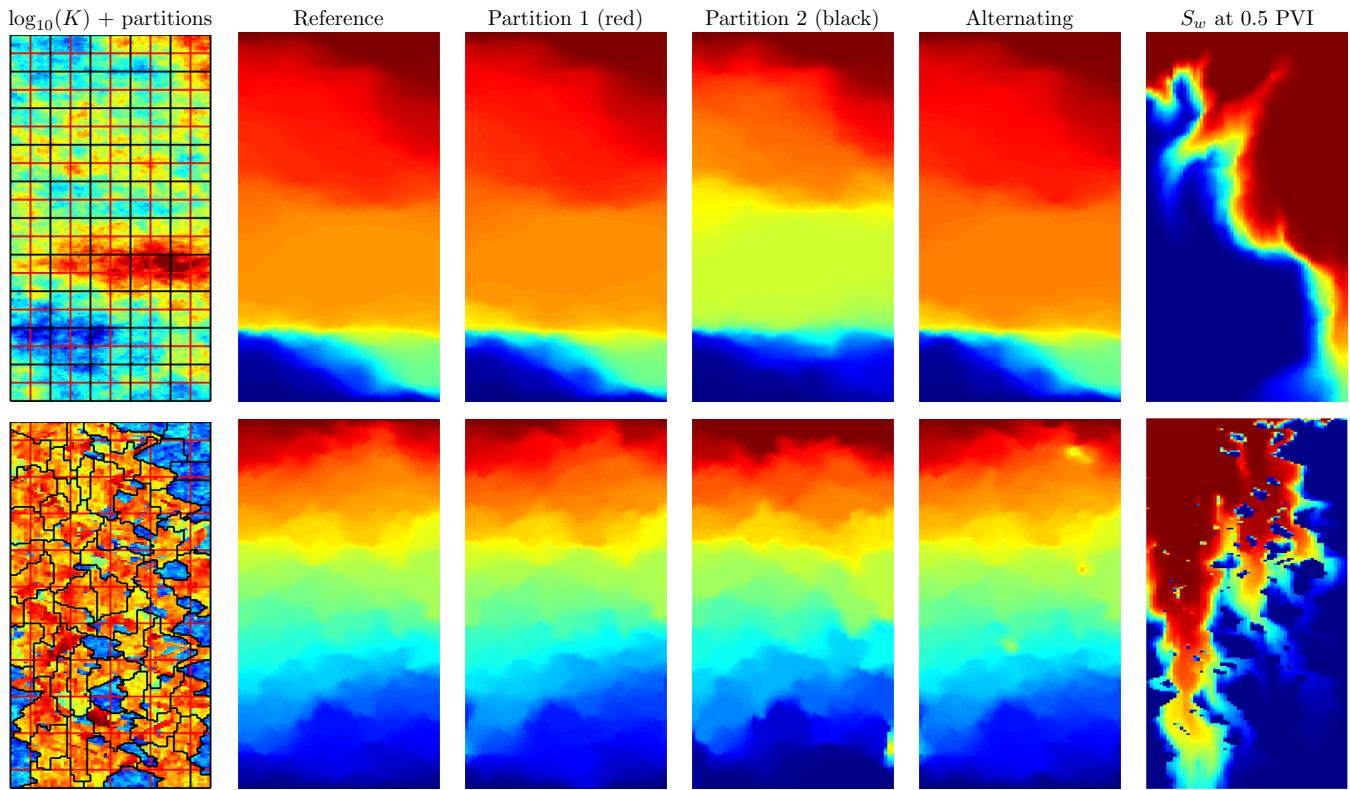


Figure 5—Multiscale pressure solutions computed for 35th and 85th layers of the SPE 10 model. For the Tarbert formation in the upper row, we use two rectangular partitions, a 6×11 and a 5×10 dual. For the Upper Ness formation in the lower row, we use the same 6×11 in combination with a partition that adapts to the permeability. Reference solutions for pressure and saturation are computed by use of a standard fine-scale finite-volume method.

Table 1—Discrepancies in L_2 and L_∞ norm compared with a fine-scale solution for multiscale solutions computed on the 35th (Tarbert) and 85th (Upper Ness) layers of the SPE 10 model.

Coarse grid	Tarbert		Upper Ness	
	L_2	L_∞	L_2	L_∞
Partition 1	0.0155	0.1174	0.0307	0.1782
Partition 2	0.1710	0.3865	0.0791	0.5506
Alternating	0.0198	0.0620	0.0293	0.2929

resulting multiscale approximation is significantly less accurate than with the regular partition in both the L^2 and the pointwise norm. With the exception of MsTPFA, all multiscale finite-volume methods reported in the literature lead to multipoint stencils on the coarse grid. It is well known that one generally cannot guarantee monotone solutions for multipoint stencils (Nordbotten et al. 2007). This deficiency is particularly evident for the MsFV method and can be attributed to inadequate localization assumptions (Wang et al. 2016). Nonmonotone solutions are more rare for MsRSB, but in this particular example, we see a tendency of nonmonotonicity near the south-east corner. Combining the two partitions reduces the L^2 error slightly and gives a somewhat higher pointwise discrepancy compare with the regular partition. Also in this case, we observe a tendency of nonmonotonicity. However, neither of the solutions have cell values outside the interval of the unit pressure drop.

Next, we introduce iterations with a single pass of ILU(0) as local smoother to improve the discrepancies reported in Table 1. In the iterative formulation, the method with combined partitions can be seen as a standard iterative multiscale method that applies different partitions in every second step in an alternating manner. Since different partitions generally will represent different error modes, the convergence of the fine-scale residual reported in Figure 6 is vastly superior when alternating partitions are used both for the smooth Tarbert formation and the channelized Upper Ness formation.

A small fine-scale residual does not necessarily mean that the approximate solution will be able to propagate saturations and components accurately. To also measure the transport properties of the multiscale approximation, we solve the pressure equation once and then simulate the injection of 1 pore volumes of the wetting fluid with a fixed flux field. Figure 7 reports discrepancies in saturations as a function of time compared with using the true fine-scale fluxes. For comparison, we also report discrepancies

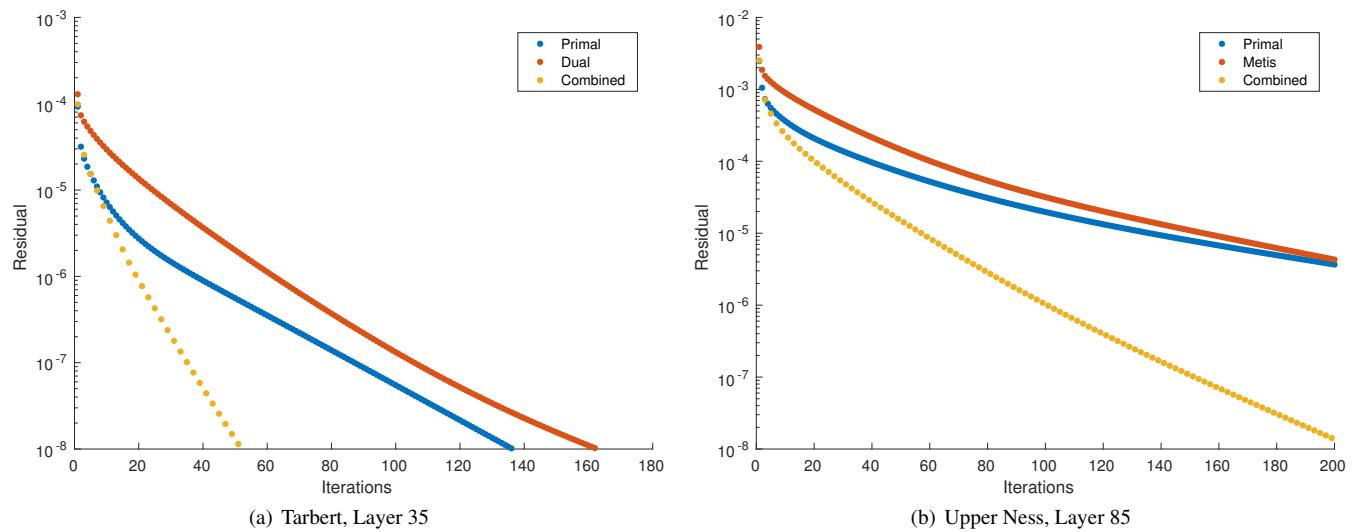


Figure 6—Convergence of the multiscale methods as an iterative solver for two horizontal layers of the SPE 10 model.

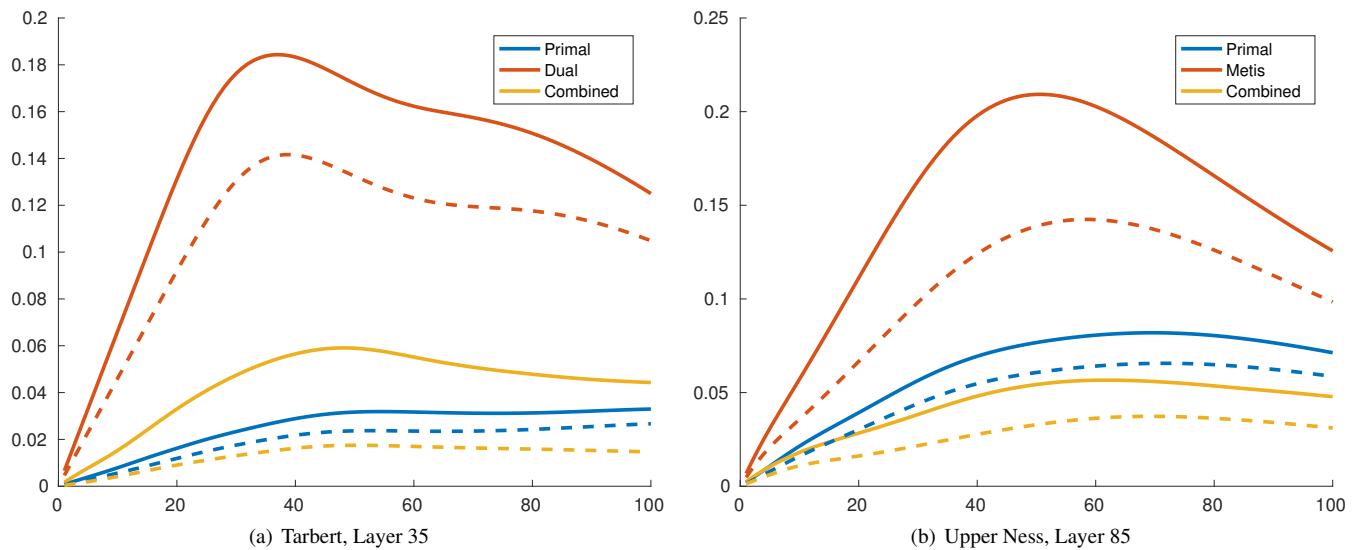


Figure 7—Discrepancies in saturation for a fixed field computed by various multiscale solvers as function of time step. The solid lines is with one multiscale solve and the dashed lines with four multiscale/ILU(0) iterations.

for approximate flow fields computed with four multiscale iterations. Here, we see that we get flux fields with significantly more accurate transport properties when two different coarse partitions are used. Note that we have employed the less inexpensive, local reconstruction for the combined solver.

Well basis To assess the accuracy of well basis functions, we consider two different permeability realizations on a 100×100 rectangular grid, see Figure 8. The first is a Gaussian permeability field with a mean of 125 md, whereas the second case consists of four layers with log-normal distributions in each layer and mean values varying from 50 to 500 md. A modified five-spot well pattern is used to drive flow, consisting of an injector in the middle of the domain and four producers in the corners. All wells are vertical and perforated in a single cell, except for the horizontal producer near the southwest corner, which is perforated in 15 cells. Instead of a fixed injection rate, which is common in papers discussing multiscale methods, we set wells to operate at a fixed bottom-hole pressure. The amount of injected fluids will then depend on the quality of the pressure approximation. The injector operates at a fixed pressure of 500 bar, whereas the producers operate at 75, 100, 110 and 125 bar, respectively.

We consider three different multiscale solvers. The first is the original MsRSB method with a 6×6 coarse grid, in which the wells are located approximately at the block centroids. This is the ideal situation for a rectangular grid with source terms that are defined on an underlying fine grid. The second uses well basis functions entirely on their own, thereby giving a very coarse multiscale solver. The third solver combines well basis functions with the regular MsRSB partition. For the first part of this example, we have set the well distance to a radius of 15 cells, which, given the underlying structured fine-grid, results in

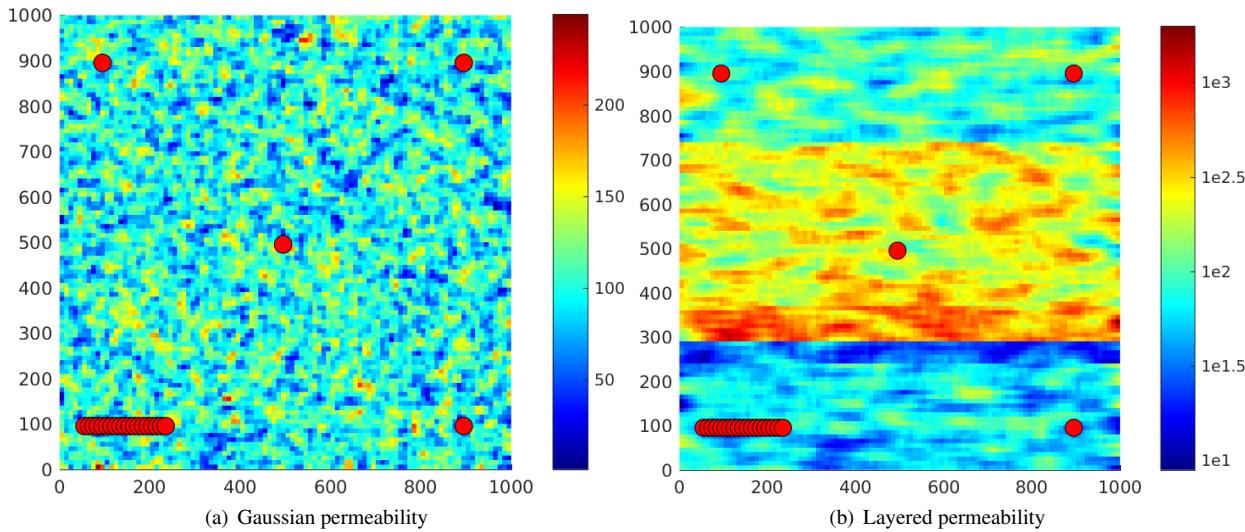


Figure 8—The two 100×100 test problems used to illustrate well basis functions.

Table 2—Discrepancies in L_2 and L_∞ norm compared with a fine-scale solution for the two different permeability distributions used to illustrate use of well basis functions.

Basis	Gaussian		Layered	
	L_2	L_∞	L_2	L_∞
MsRSB	0.0641	0.1679	0.0619	0.1750
Well basis ($d=15$)	0.0760	0.1131	0.1015	0.1215
Well basis + MsRSB	0.0303	0.1136	0.0280	0.0634

circular support regions for the well basis functions. Table 2 reports the discrepancies from the fine-scale solution for the three solvers. We observe that the well basis or a regular partition on their own give larger discrepancies than when the two are used in combination. If we examine the pressure fields in Figure 9, it is clear that basis functions defined in a regular partition will average out the impact of each well over the coarse blocks, giving inadequate detail near the wells. The well basis captures the near-well pressure responses accurately, but because these functions are only supported near the wells, their predictive quality is poor in the middle of the reservoir. The combined solver resolves both the global features and the local flow field accurately. The main purpose of using the well basis functions is to improve the approximation error in the near-well regions. Outside these regions, the basis functions are constant and will hence not contribute significantly to improve convergence when used as iterative preconditioners, which is confirmed in Figure 10.

To investigate how well the three approximate solvers resolve the transport properties of the fine-scale flow field, we solve a passive tracer flow problem in the fixed velocity fields given by each solver over a period normalized to the time it takes to inject one pore volume in the fine-scale, exact solver. Figure 11 reports the corresponding discrepancies in tracer concentration. We observe that the error is larger for the pure multiscale solver than for the well basis, which in turn is outperformed by the combined solver. Computing the flux field is very expensive for the case with only well basis compared to the other two cases, since the subdomains used for reconstruction for the five basis functions are large and consequently expensive to invert.

Finally, we perform a systematic test to investigate how the initial error behaves as a function of the distance d used to determine the radius of the near-well zone. The distance is systematically varied between $d = 0$ (no well basis, only MsRSB) and $d = 50$ (well bases cover the entire domain, and bases for the different wells overlap). Results reported in Figure 12 show that there is a relatively rapid decay in errors initially for both permeabilities, which eventually decays as the near-well effects are completely captured at $d \approx 20$.

Unstructured grid with faults and high-permeable channels For this example, we consider an unstructured Voronoi grid in 2D with 8560 fine cells adapting to faults, high-permeable channels, and wells Klemetsdal et al. (2017). The model seen in Figure 13 contains 5 intersecting faults and 13 high-permeable channels. The faults have a transmissibility multiplier of 0.01 and are thus blocking flow from passing through, but the faults are also intersected by the network high-permeable channels. The high-permeable channels have 5000 md permeability, which is a large contrast to the Gaussian background matrix permeability distribution having a mean of 100 mD. To get an accurate fine-scale solution, the grid has smaller cells near the faults, wells and faults. Flow is driven by an injector-producer pair controlled at 500 and 200 bar bottom-hole pressures. The flow pattern is complex, with large jumps in pressure over the faults and high-permeable channels dominating the saturation distribution (see

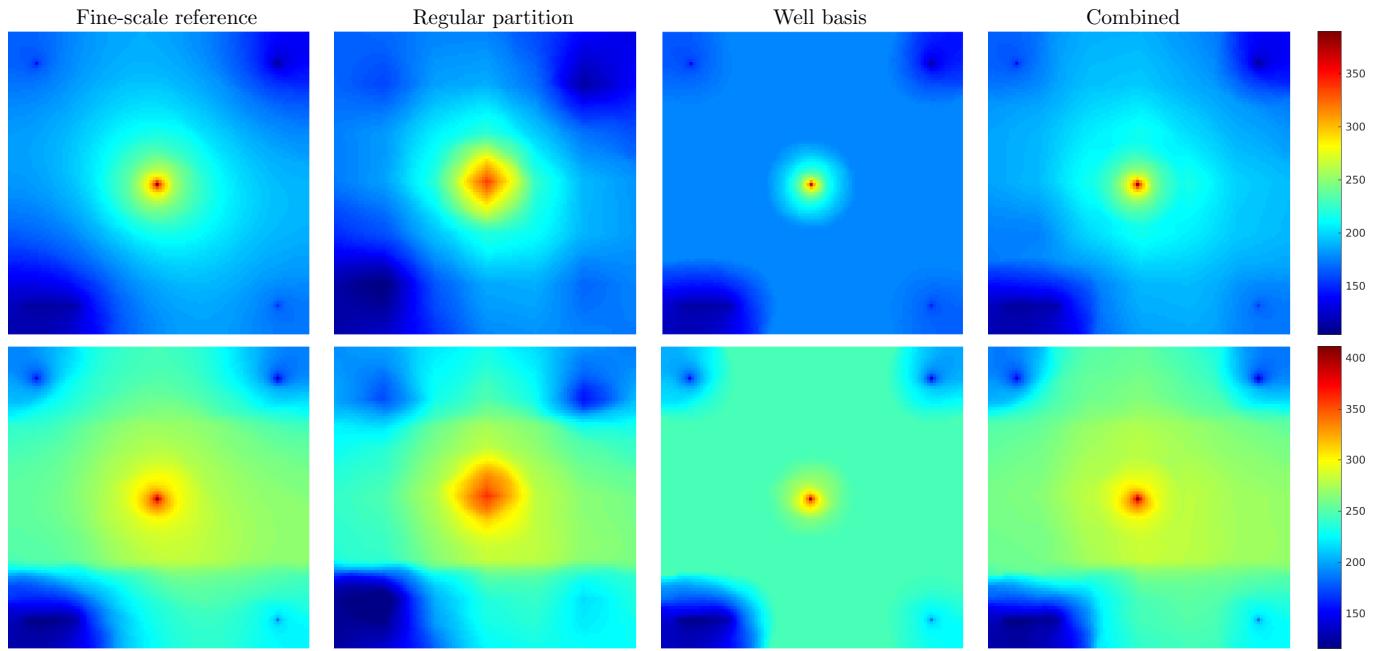


Figure 9—Pressure solutions for the two permeability distributions with different solvers; the upper row is the log-normal case and the bottom row the layered case.

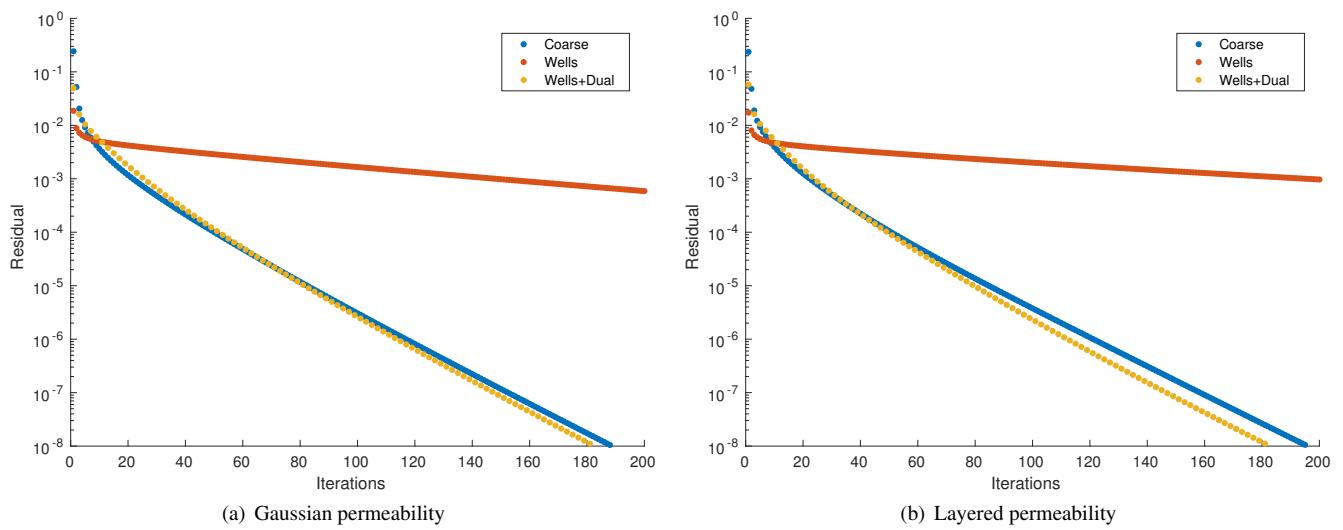


Figure 10—Convergence of the multiscale methods with well basis functions as iterative solvers for the two different test cases from Figure 8.

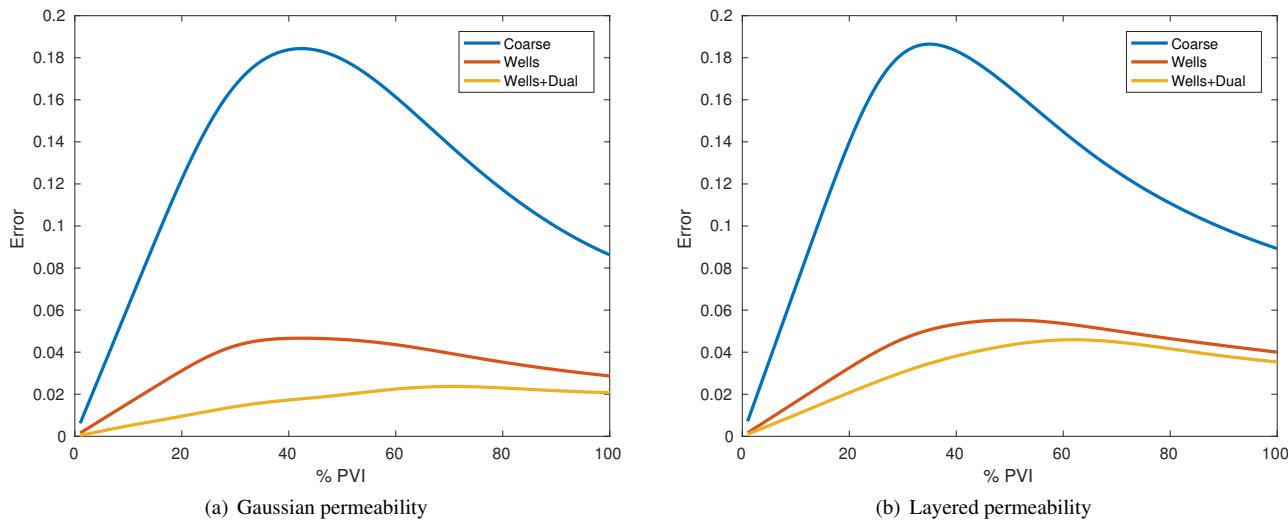


Figure 11—Concentration error for the well basis example for two different permeability realizations.

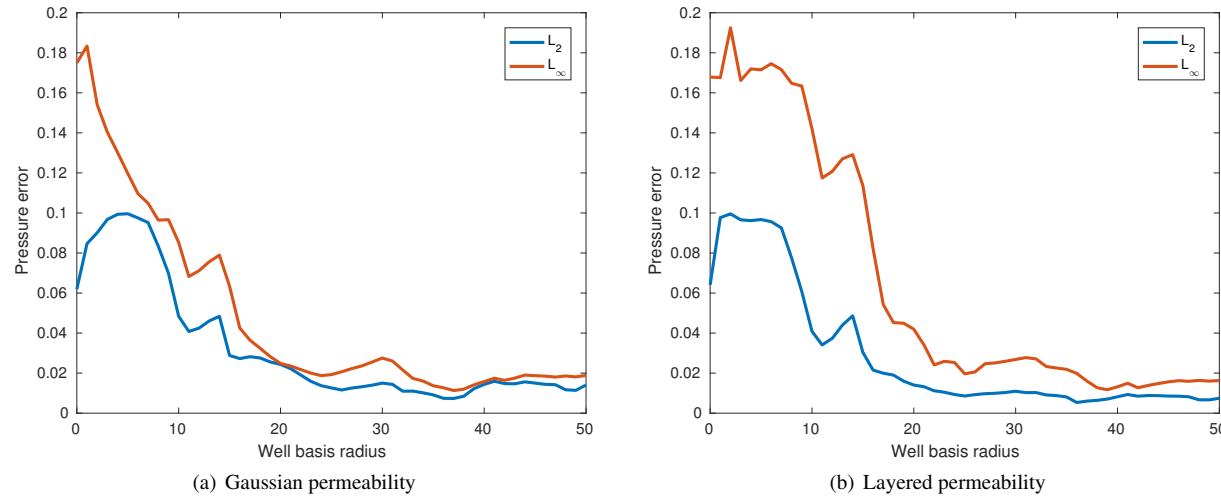


Figure 12—Discrepancy between the fine-scale solution and the multiscale pressure approximation as a function of well basis size.

Figure 14) and constitutes a challenging test for multiscale solvers.

The example is intended to illustrate the situation where a model contains a large number of *features* that affect the flow pattern, and we will investigate to what extent we can improve the accuracy of the multiscale solution by use of partitions that adapt to these features. The baseline multiscale solver relies on a rectangular partition in physical space, consisting of 10×10 coarse blocks that do not account for any spatial features. In addition, we consider a Metis partition with the same number of coarse blocks, for which the graph partitioning strikes a balance between automatically adapting to the geological features as represented in the system matrix and creating coarse blocks that minimize communication volume and discrepancy in block sizes. To get adapt coarse grids to high-permeable channels and faults, we perform an additional Metis partitioning in which connections over faults and between the high-permeable channels and the background sand have been removed. To focus on the local features, we allow up to 500% variation in block sizes and use half the number of degrees-of-freedom compared with the other coarse grids. Finally, we include the well basis functions described in the previous example.

Table 3 reports the initial discrepancy between the multiscale approximation and the fine-scale solution. We observe that the MsRSB method generally gives accurate results regardless of the partition used, but that the Metis grid clearly outperforms the structured coarse grid, which does not take connection strengths of the system into account. Combining different partitions, the best results are obtained when rectangular or Metis partitions are used together with well basis functions. Considering the saturation discrepancies for a linear displacement problem in Figure 15, we observe that there seems to be a steady improvement in accuracy as additional features are added to the approximation. The feature-adapted basis functions, however, do not seem to give any improvement compared to the same solver with only generic basis functions. In the convergence plot for the Richardson iterations, however, we see that including the basis functions for the local features significantly improves the convergence rate.

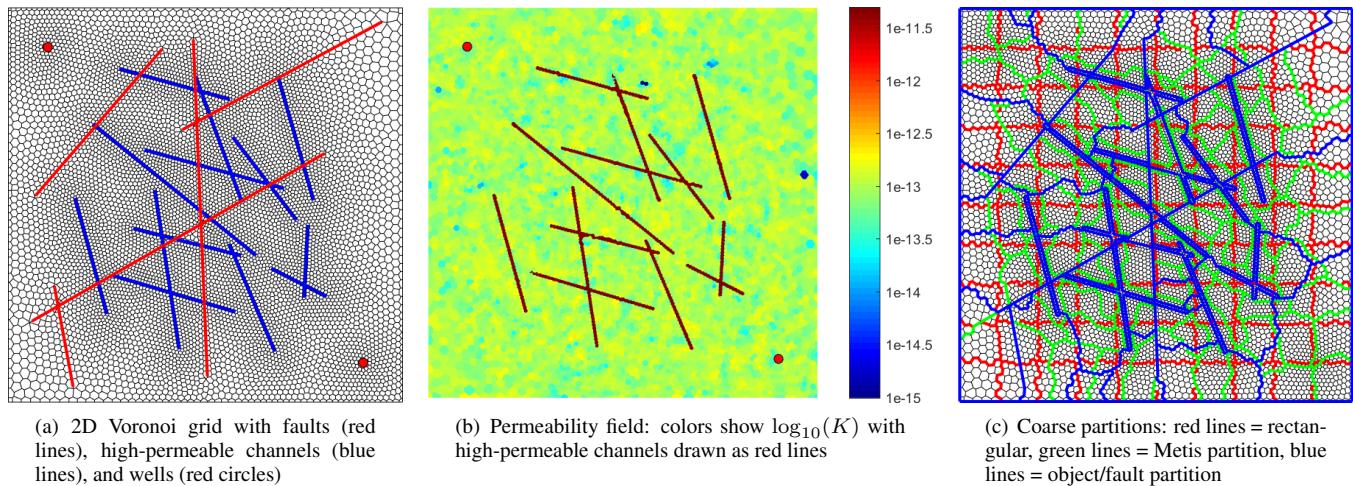


Figure 13—Setup for the test case with an unstructured grid that adapts to faults, high-permeable channels, and wells.

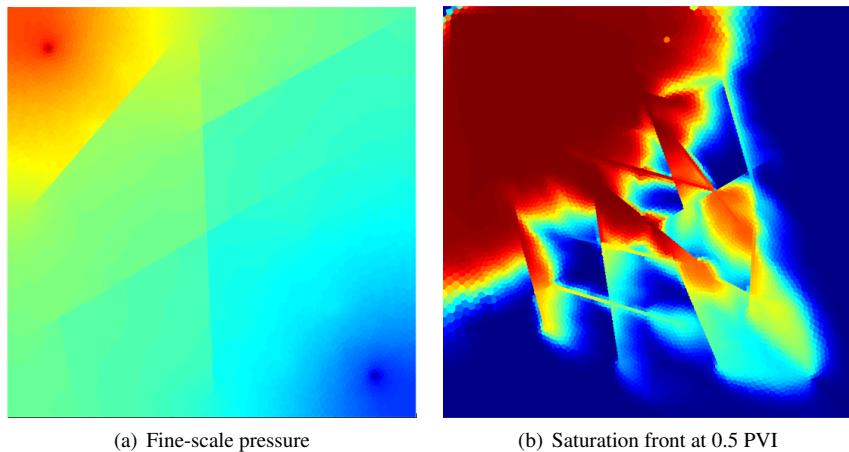


Figure 14—Reference solution for the unstructured test case.

Table 3—Discrepancies between the fine-scale pressure solution and multiscale approximations computed with different combinations of coarse partitions for the 2D unstructured grid example with and without well basis functions.

Solver	Multiscale		With well basis	
	L_2	L_∞	L_2	L_∞
(1) Cartesian(100 dof)	0.0110	0.05232	0.0197	0.0345
(2) Metis (100 dof)	0.0192	0.06431	0.0106	0.0270
(3) Channels separated (50 dof)	0.0215	0.07867	0.0157	0.0471
(1 + 2)	0.0123	0.03351	0.0080	0.0192
(1 + 2 + 3)	0.0119	0.03359	0.0077	0.0145

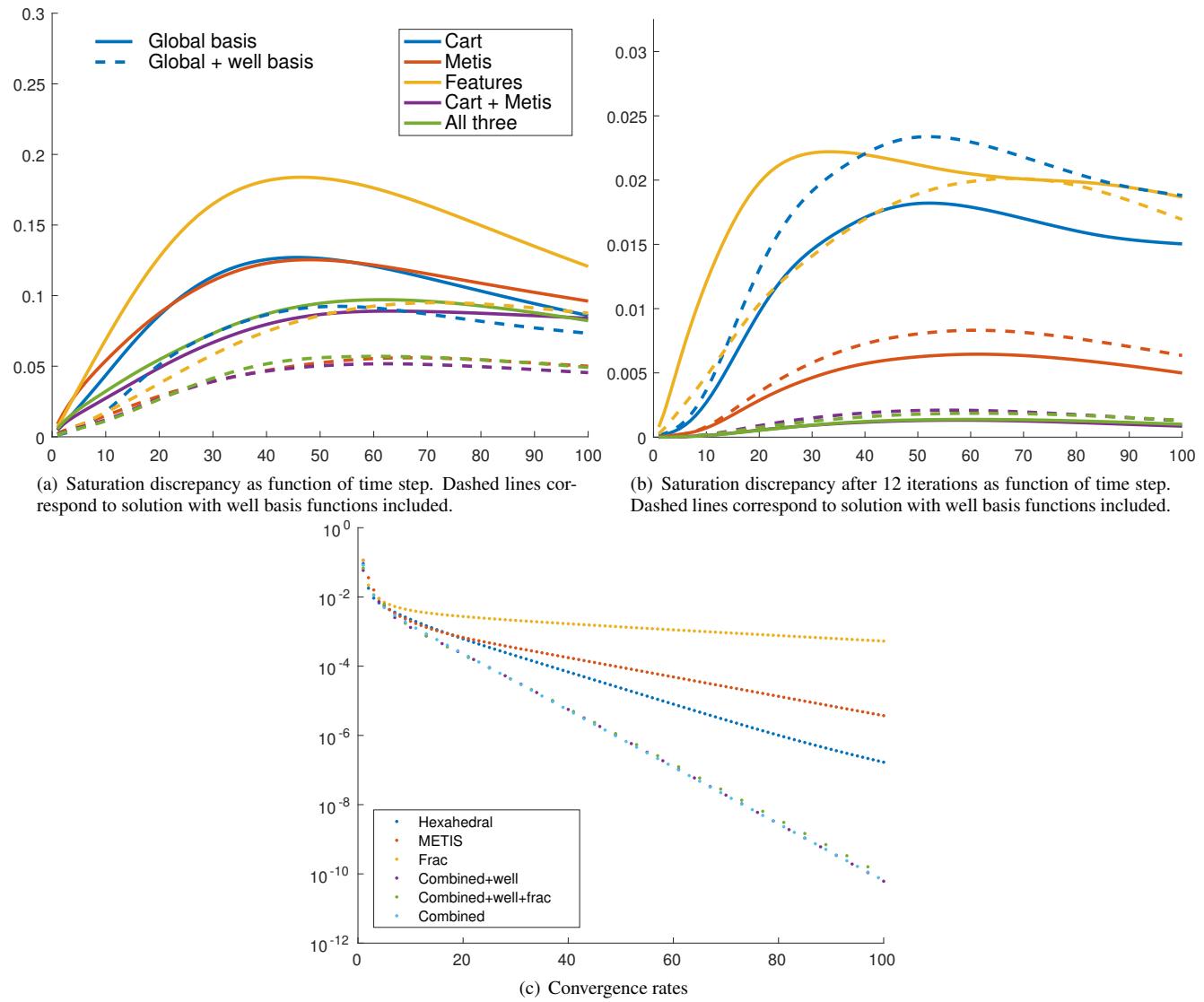


Figure 15—Saturation discrepancies and convergence rates for the unstructured test case.

Gullfaks Our last example revisits a test case from (Møyner and Lie 2016b). The simulation model of the Gullfaks field from the Norwegian sector of the North Sea is a challenging example of a real geological model with strong petrophysical heterogeneity, large anisotropy and aspect ratios, degenerate cell geometries, and unstructured grid topology. The geology consists of several reservoir zones, including delta sandstones, shallow-marine sand, fluvial-channel and delta-plain formations, with the main production coming from Brent sands (i.e., from the same sedimentary environment seen in the SPE 10 model). The structural model has rotated fault blocks in the west and a structural horst in the east, with a highly faulted area in between. The simulation model therefore has a large number of sloping faults, with angles varying from 30 to 80 degrees and throws from zero to 300 m; see (Fossen and Hesthammer 1998) for more details. Out of the 416 000 cells in the $80 \times 100 \times 52$ corner-point grid, 216 344 cells are active and 44% of these have non-neighboring connections. When interpreted as a matching grid, cells will have between four and thirty-one faces. In our experience, it is almost impossible to generate consistent primal–dual partitions necessary to compute MsFV basis functions. On the other hand, the grid has a rich structure that can be utilized to define adapted partitions and MsRSB-type basis functions.

We consider two different partitioning strategies: (i) a rectangular partition in logical space in which we split blocks across intersecting faults, and (ii) a Metis partition with the same number of grid blocks computed with graph weights derived from the fine-scale transmissibilities. We construct two coarse grids of each type: The first rectangular grid is built from logical blocks consisting of $12 \times 12 \times 15$ fine cells, giving a total of 447 coarse blocks after splitting across faults. The high degree of coarsening in the vertical direction is somewhat misleading, as the model contains a large number of inactive layers and eroded cells. The second grid uses a coarsening factor $5 \times 5 \times 8$ and has 2530 blocks. Likewise, we construct two Metis partitions with 447 and 2530 blocks. Figure 16 shows excerpts of the two coarse grids with 447 blocks.

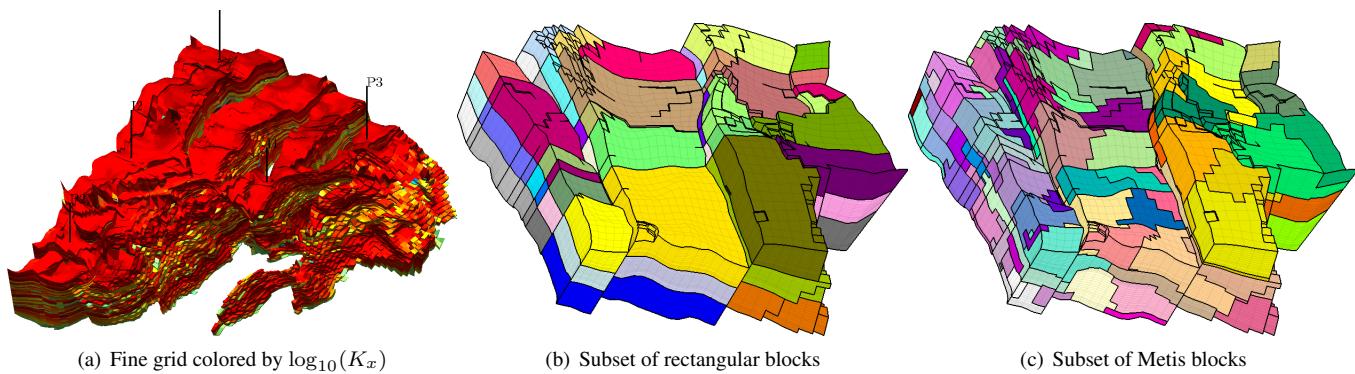


Figure 16—The Gullfaks simulation model partitioned into 447 blocks.

Table 4—The discrepancy between the fine-scale solution and approximate multiscale solutions with different partitions for the Gullfaks test case.

Solver	Multiscale		With well basis	
	L_2	L_∞	L_2	L_∞
(1) Cartesian (447 blocks)	0.0365	0.2299	0.0261	0.2785
(2) Metis (447 blocks)	0.0331	0.1527	0.0190	0.0800
(3) Cartesian (2530 blocks)	0.0344	0.3325	0.0155	0.2895
(4) Metis (2530 blocks)	0.0153	0.1699	0.0143	0.1605
(1 + 2)	0.0270	0.1966	0.0124	0.2863
(3 + 4)	0.0159	0.6544	0.0053	0.2888

To drive flow through the model, we set up a simple well pattern consisting of two vertical injectors and three vertical producers. The discrepancies in the initial multiscale solutions reported in Table 4 are somewhat inconclusive: using only the Metis partition gives the lowest pointwise discrepancy, whereas combining both partitions and well basis functions gives the lowest L^2 error. On the other hand, the discrepancies are very low, given the low number of coarse blocks and the complexity of the geological model. This testifies both to the robustness and the accuracy of the MsRSB method. Moreover, the convergence plots reported in Figure 18 confirm once again that employing (at least) two different partitions will generally reduce more error modes and give significantly faster convergence when the multiscale method is used as an iterative solver.

A word of caution at the end: the Metis partitions and the intersection of the Metis partitions and the rectangular partitions will give blocks with irregular geometry and in some cases high inter-block variation in petrophysical properties. If all the local systems are collected into one global matrix, this matrix will have very poor condition number and local rescaling of the individual subsystems is required if one wants to use a direct solver to invert all the systems in one operation.

Summary

In this paper we have introduced a flexible framework for combining multiple multiscale operators to solve the flow equations in porous media. In a series of test cases we demonstrate how this framework can be used to combine one or more generic multiscale operator defined to have good resolution for the entire reservoir with specific multiscale operators tailored to particular features of the reservoir model, such as well patterns, high-permeable channels, or regions with different rock types. The new framework is easy to implement, provides better approximate solutions and faster convergence rates at a very modest cost. In particular, the framework is applicable to real reservoir models with challenging high contrast geology, complex reservoir geometry with a large number of faults, and complex unstructured grids.

Acknowledgments

This research was funded in part by Schlumberger Information Solutions and by the Research Council of Norway under grant no. 226035.

References

- Akkutlu, I. Y., Efendiev, Y., Vasilyeva, M., and Wang, Y. 2017. Multiscale model reduction for shale gas transport in a coupled discrete fracture and dual-continuum porous media. *J. Nat. Gas Sci. Eng.* doi: 10.1016/j.jngse.2017.02.040.
- Bush, L., Ginting, V., and Presho, M. 2014. Application of a conservative, generalized multiscale finite element method to flow models. *J. Comput. Appl. Math.*, 260:395–409. doi: 10.1016/j.cam.2013.10.006.

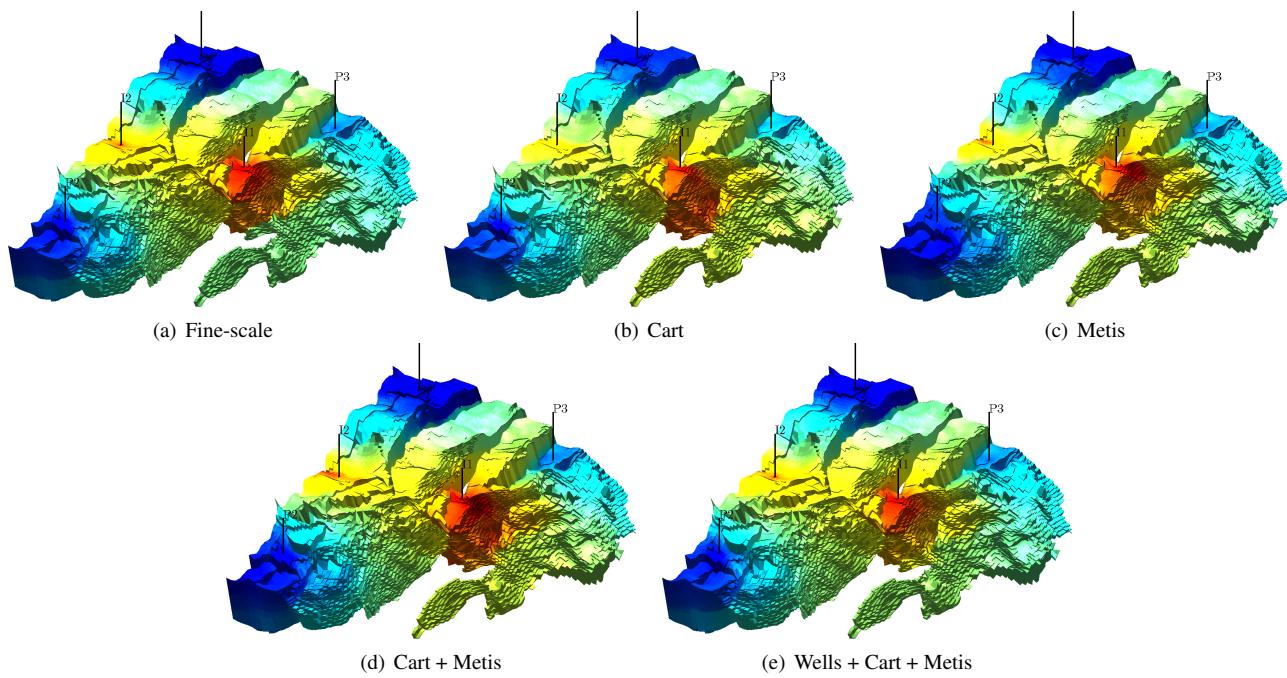


Figure 17—Pressure solutions computed for the Gullfaks model partitioned with 447 blocks.

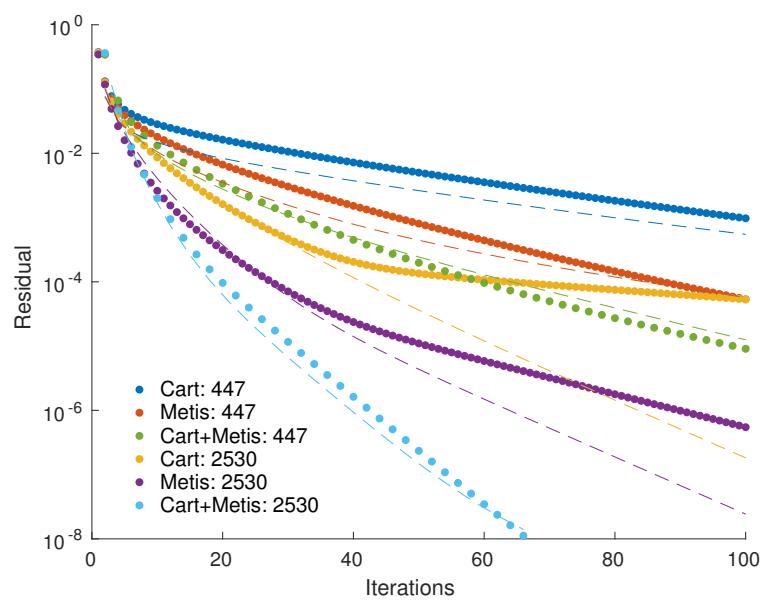


Figure 18—Convergence of the multiscale solver with various partitions for the Gullfaks test case. Dashed lines correspond to solvers with the well basis functions included as a separate stage.

- Christie, M. A. and Blunt, M. J. 2001. Tenth SPE comparative solution project: A comparison of upscaling techniques. *SPE Reservoir Eval. Eng.*, 4:308–317. doi: 10.2118/72469-PA.
- Chung, E. T., Efendiev, Y., and Li, G. 2014. An adaptive GMsFEM for high-contrast flow problems. *J. Comput. Phys.*, 273:54–76. doi: 10.1016/j.jcp.2014.05.007.
- Cortinovis, D. and Jenny, P. 2014. Iterative Galerkin-enriched multiscale finite-volume method. *J. Comput. Phys.*, 277:248–267. doi: 10.1016/j.jcp.2014.08.019.
- Cortinovis, D. and Jenny, P. 2017. Zonal multiscale finite-volume framework. *J. Comput. Phys.*, 337:84–97. doi: 10.1016/j.jcp.2017.01.052.
- Cusini, M., van Kruijsdijk, C., and Hajibeygi, H. 2016. Algebraic dynamic multilevel (ADM) method for fully implicit simulations of multi-phase flow in porous media. *Journal of Computational Physics*, 314:60–79. doi: 10.1016/j.jcp.2016.03.007.
- Dolean, V., Jolivet, P., Nataf, F., Spillane, N., and Xiang, H. 2014. Two-level domain decomposition methods for highly heterogeneous Darcy equations. Connections with multiscale methods. *Oil & Gas Science and Technology – Rev. IFP Energies nouvelles*, 69(4):731–752. doi: 10.2516/ogst/2013206.
- Efendiev, Y., Galvis, J., and Hou, T. Y. 2013. Generalized multiscale finite element methods (GMsFEM). *J. Comput. Phys.*, 251:116–135. doi: 10.1016/j.jcp.2013.04.045.
- Efendiev, Y., Galvis, J., and Wu, X.-H. 2011. Multiscale finite element methods for high-contrast problems using local spectral basis functions. *J. Comput. Phys.*, 230(4):937–955. doi: 10.1016/j.jcp.2010.09.026.
- Efendiev, Y. and Hou, T. Y. 2009. *Multiscale Finite Element Methods*, volume 4 of *Surveys and Tutorials in the Applied Mathematical Sciences*. Springer Verlag, New York.
- Fossen, H. and Hesthammer, J. 1998. Structural geology of the Gullfaks Field. In Coward, M. P., Johnson, H., and Dalaban, T. S., editors, *Structural geology in reservoir characterization*, volume 127, pages 231–261. Geological Society Special Publication.
- Hajibeygi, H. and Tchelepi, H. A. 2014. Compositional multiscale finite-volume formulation. *SPE J.*, 19(2):316–326. doi: 10.2118/163664-PA.
- Hauge, V. L. 2010. *Multiscale Methods and Flow-based Gridding for Flow and Transport in Porous Media*. PhD thesis, Norwegian University of Science and Technology.
- Hauge, V. L., Lie, K.-A., and Natvig, J. R. 2012. Flow-based coarsening for multiscale simulation of transport in porous media. *Comput. Geosci.*, 16(2):391–408. doi: 10.1007/s10596-011-9230-x.
- 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.*, 134:169–189. doi: 10.1006/jcph.1997.5682.
- 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. doi: 10.1016/S0021-9991(03)00075-5.
- Jenny, P. and Lunati, I. 2009. Modeling complex wells with the multi-scale finite-volume method. *J. Comput. Phys.*, 228(3):687 – 702. doi: 10.1016/j.jcp.2008.09.026.
- Karypis, G. and Kumar, V. 1998. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM J. Sci. Comp.*, 20(1):359–392. doi: 10.1137/S1064827595287997.
- 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. doi: 10.1007/s10596-007-9074-6.
- Klemetsdal, Ø. S., Berge, R. L., Lie, K.-A., Nilsen, H. M., and Møyner, O. 2017. Unstructured gridding and consistent discretizations for reservoirs with faults and complex wells. In *SPE Reservoir Simulation Conference, 20-22 February, Montgomery, Texas, USA*. doi: 10.2118/182666-MS.
- Kozlova, A., Walsh, D., Chittireddy, S., Li, Z., Natvig, J., Watanabe, S., and Bratvedt, K. 2016. A hybrid approach to parallel multiscale reservoir simulator. In *ECMOR XV – 15th European Conference on the Mathematics of Oil Recovery*, Amsterdam, The Netherlands. EAGE. doi: 10.3997/2214-4609.201601889.
- Künze, R., Lunati, I., and Lee, S. H. 2013. A multilevel multiscale finite-volume method. *J. Comput. Phys.*, 255:502–520. doi: 10.1016/j.jcp.2013.08.042.
- Lie, K., Krogstad, S., Ligaarden, I. S., Natvig, J. R., and Nilsen, Halvor an d Skaflestad, B. 2012. Open-source MATLAB implementation of consistent discretisations on complex grids. *Comput. Geosci.*, 16:297–322. doi: 10.1007/s10596-011-9244-4.
- Lie, K.-A. 2016. *An Introduction to Reservoir Simulation Using MATLAB: User guide for the Matlab Reservoir Simulation Toolbox (MRST)*. SINTEF ICT, <http://www.sintef.no/Projectweb/MRST/publications>, 3rd edition.

- Lie, K.-A., Kedia, K., Skaflestad, B., Wang, X., Yang, Y., and Wu, X.-H. 2017a. A general non-uniform coarsening and upscaling framework for reduced-order modelling. In *SPE Reservoir Simulation Conference, Montgomery, TX, USA, 20–22 February 2017*. SPE 182681-MS, doi: 10.2118/182681-MS.
- Lie, K.-A., Møyner, O., Natvig, J. R., Kozlova, A., Bratvedt, K., Watanabe, S., and Li, Z. 2017b. Successful application of multiscale methods in a real reservoir simulator environment. *Comput. Geosci.* doi: 10.1007/s10596-017-9627-2.
- Lie, K.-A., Natvig, J. R., Krogstad, S., Yang, Y., and Wu, X.-H. 2014. Grid adaptation for the Dirichlet–Neumann representation method and the multiscale mixed finite-element method. *Comput. Geosci.*, 18(3):357–372. doi: 10.1007/s10596-013-9397-4.
- Lunati, I. and Lee, S. H. 2009. An operator formulation of the multiscale finite-volume method with correction function. *Multiscale Model. Simul.*, 8(1):96–109. doi: 10.1137/080742117.
- Manea, A., Hajibeygi, H., Vassilevski, P., and Tchelepi, H. A. 2016. Enriched algebraic multiscale linear solver. In *ECMOR XV–15th European Conference on the Mathematics of Oil Recovery*. doi: 10.3997/2214-4609.201601894.
- Manea, A., Hajibeygi, H., Vassilevski, P., and Tchelepi, H. A. 2017. Parallel enriched algebraic multiscale solver. In *SPE Reservoir Simulation Conference, 20–22 February, Montgomery, Texas, USA*. doi: 10.2118/182694-MS.
- Møyner, O. and Lie, K.-A. 2014. A multiscale two-point flux-approximation method. *J. Comput. Phys.*, 275:273–293. doi: 10.1016/j.jcp.2014.07.003.
- Møyner, O. and Lie, K.-A. 2016a. A multiscale restriction-smoothed basis method for compressible black-oil models. *SPE J.*, 21(6):2079–2096. doi: 10.2118/173265-PA.
- Møyner, O. and Lie, K.-A. 2016b. A multiscale restriction-smoothed basis method for high contrast porous media represented on unstructured grids. *J. Comput. Phys.*, 304:46–71. doi: 10.1016/j.jcp.2015.10.010.
- Møyner, O. and Tchelepi, H. A. 2017. A multiscale restriction-smoothed basis method for compositional models. In *SPE Reservoir Simulation Conference, Montgomery, TX, USA, 20–22 February 2017*. SPE 182679-MS, doi: 10.2118/182679-MS.
- Nordbotten, J., Aavatsmark, I., and Eigestad, G. 2007. Monotonicity of control volume methods. *Numer. Math.*, 106(2):255–288. doi: 10.1007/s00211-006-0060-z.
- Skaflestad, B. and Krogstad, S. 2008. Multiscale/mimetic pressure solvers with near-well grid adaption. In *Proceedings of ECMOR XI–11th European Conference on the Mathematics of Oil Recovery*, number A36, Bergen, Norway. EAGE. doi: 10.3997/2214-4609.20146387.
- Vaněk, P., Mandel, J., and Brezina, M. 1996. Algebraic multigrid by smoothed aggregation for second and fourth order elliptic problems. *Computing*, 56(3):179–196. doi: 10.1007/BF02238511.
- Wang, Y., Hajibeygi, H., and Tchelepi, H. A. 2014. Algebraic multiscale solver for flow in heterogeneous porous media. *J. Comput. Phys.*, 259:284–303. doi: 10.1016/j.jcp.2013.11.024.
- Wang, Y., Hajibeygi, H., and Tchelepi, H. A. 2016. Monotone multiscale finite volume method. *Comput. Geosci.*, 20(3):509–524. doi: 10.1007/s10596-015-9506-7.
- Wolfsteiner, C., Lee, S. H., and Tchelepi, H. A. 2006. Well modeling in the multiscale finite volume method for subsurface flow simulation. *Multiscale Model. Simul.*, 5(3):900–917. doi: 10.1137/050640771.