# A MULTISCALE RESTRICTION-SMOOTHED BASIS METHOD FOR HIGH CONTRAST POROUS MEDIA REPRESENTED ON UNSTRUCTURED GRIDS

OLAV MØYNER AND KNUT-ANDREAS LIE

Abstract. A wide variety of multiscale methods have been proposed in the literature to reduce runtime and provide better scaling for the solution of Poisson-type equations modeling flow in porous media. We present a new multiscale restricted-smoothed basis (MsRSB) method that is designed to be applicable to both rectilinear grids and unstructured grids. Like many other multiscale methods, MsRSB relies on a coarse partition of the underlying fine grid and a set of local prolongation operators (multiscale basis functions) that map unknowns associated with the fine grid cells to unknowns associated with blocks in the coarse partition. These mappings are constructed by restricted smoothing: Starting from a constant, a localized iterative scheme is applied directly to the fine-scale discretization to compute prolongation operators that are consistent with the local properties of the differential operators.

The resulting method has three main advantages: First of all, both the coarse and the fine grid can have general polyhedral geometry and unstructured topology. This means that partitions and good prolongation operators can easily be constructed for complex models involving high media contrasts and unstructured cell connections introduced by faults, pinch-outs, erosion, local grid refinement, etc. In particular, the coarse partition can be adapted to geological or flow-field properties represented on cells or faces to improve accuracy. Secondly, the method is accurate and robust when compared to existing multiscale methods and does not need expensive recomputation of local basis functions to account for transient behavior: Dynamic mobility changes are incorporated by continuing to iterate a few extra steps on existing basis functions. This way, the cost of updating the prolongation operators becomes proportional to the amount of change in fluid mobility and one reduces the need for expensive, tolerance-based updates. Finally, since the MsRSB method is formulated on top of a cell-centered, conservative, finite-volume method, it is applicable to any flow model in which one can isolate a pressure equation. Herein, we only discuss single and two-phase incompressible models. Compressible flow, e.g., as modeled by the black-oil equations, is discussed in a separate paper.

## 1. Introduction

The general movement of fluids in a hydrocarbon reservoir is induced by global forces like gravity and pressure differentials. The micro-scale displacement, however, is determined by small-scale flow paths throughout highly heterogeneous porous rocks. Flow modeling therefore needs to take into account processes taking place on a wide range of spatial and temporal scales. Resolving all these scales using a single high-resolution grid is not computationally tractable. Instead, the traditional approach is to use upscaling or homogenization techniques to develop effective parameters that represent subscale behavior in an averaged sense on a coarser scale. Such methods have proved to be very effective for problems with scale separation. However, porous rocks seldom exhibit clear scale separation and upscaling techniques are therefore not as robust and accurate as one would wish. Effective properties are generally process dependent, and because one needs to assume a specific set of localization conditions to compute effective properties, upscaling techniques tend to only produce reliable results for a limited range of flow scenarios.

In an attempt to overcome some of the limitations of upscaling methods, so-called multiscale discretization methods have been proposed over the past two decades to solve second-order elliptic

SINTEF ICT, Department of Applied Mathematics, P.O. Box 124, Blindern, NO-0314 Oslo, Norway
*E-mail addresses*: olav.moyner@sintef.no, knut-andreas.lie@sintef.no.
*Date*: September 26, 2015.

equations with strongly heterogeneous coefficients [15]. This includes methods such as the generalized finite-element methods [9], finite-element methods [21], numerical-subgrid upscaling [6, 7], multiscale mixed finite-element methods [13, 24], multiscale finite-volume methods [22], mortar mixed finite-element methods [8], and multiscale mimetic methods [34], to name a few. The key idea of all these methods is to construct a set of prolongation operators (or basis functions) that map between unknowns associated with cells of the fine geo-cellular grid and unknowns on a coarser grid used for dynamic simulation. The prolongation operators are computed numerically by solving localized flow problems, much in the same way as for flow-based upscaling methods, but unlike effective parameters, the multiscale basis functions have subscale resolution. The result is that local fine-scale variations can be systematically and correctly accounted for when constructing a reduced coarse-scale problem to study the macro-scale displacement driven by global forces. There is a large body of literature that develops such multiscale methods and studies their mathematical and numerical properties for idealized and simplified problems like the variable-coefficient Poisson equation formulated on Cartesian box geometry. However, to provide value for commercial applications, these methods also need to be developed so that they can handle the complexity in flow physics and geological description seen in real-life simulation models. Over the past decade, there have primarily been two main developments in this direction, focusing on the multiscale finite-volume (MsFV) method [22] and the multiscale mixed finite-element (MsMFE) method [1, 2, 13].

Research on the MsFV method has mainly focused on extending the method from incompressible flow to realistic flow physics [18, 19, 30, 35, 37, 59] and on developing iterative approaches that ensure that the method converges to the correct solution of the underlying fine-scale discretization [39, 58–60]. However, with a few notable exceptions [26, 40, 42, 44, 51], the MsFV method has so far only been studied on grids with a Cartesian topology. (An alternative finite-volume formulation is also discussed in [50]). Such grids are highly desirable in terms of accuracy, efficiency, and robustness of the numerical discretizations and solvers, and modeling approaches used in industry are therefore predominantly structured in a global sense. However, to accurately account for structural features like faults, joints, and deformation bands and stratigraphic characteristics like channels, lobes, clinoforms, and shale/mud drapes, unstructured connections are introduced locally and cell geometries tend to be (highly) skewed or degenerate. Similarly, unstructured connections may be introduced by local grid refinement, e.g., in the near-well zones. The challenge in extending the MsFV method to realistic stratigraphic grids, or in the more general sense to grids with fully unstructured topologies, lies in the underlying primal-dual coarse partition. The MsFV method computes basis functions on a dual partition to define transmissibilities in a multi-point coarse-scale discretization. Approximate solutions computed from these basis functions is then used to define boundary conditions for another set of flow problems on the primal partition to reconstruct conservative fine-scale fluxes. We have previously demonstrated that compatible primal-dual partitions can be generated for grids with degenerate cells and unstructured topologies if these grids are not too irregular [44]. However, the coarsening process is difficult to automate in a robust manner, and so far our most advanced algorithm is only able to provide semi-structured partitions for a limited range of coarsening factors. It is also well known that highly contrasted media and large anisotropy ratios may introduce strong non-monotonicities that are hard to get rid of in the iterative stages of the method [58].

For the MsMFE method, on the other hand, the main focus has been on making the method as geometrically flexible as possible and developing coarsening strategies that semi-automatically adapt to barriers, channels, faults, and wells in a way that ensures good accuracy for a chosen level of coarsening. The resulting method can efficiently predict flow patterns that are qualitatively correct for highly heterogeneous and geologically complex reservoir models under the assumption of incompressible flow [4, 5, 47, 49]. The method has also been extended towards realistic compressible flow physics [27, 28], but this has proved difficult to achieve in a fully robust manner because of the inherent assumption of a pressure equation written on mixed form. In a recent work [43], we

presented a fully algebraic finite-volume framework that combines the best features of the MsFV and MsMFE methods and developed one specific method that mimics a coarse-scale two-point stencil by using numerically generated partition-of-unity functions to glue together elementary flow solutions associated with interfaces between coarse blocks. The resulting MsTPFA method generally produces high-quality approximate solutions for complex industry-standard grids with high aspect ratios and unstructured connections and can easily be extended to incorporate realistic flow physics. However, the method admittedly requires some intricate details to define and compute the partition-of-unity functions. Herein, we present a new and quite different multiscale formulation that offers the same robustness and flexibility as MsTPFA, but is much simpler to implement and gives very accurate interpolation.

The new method constructs mappings based on restricted smoothing: Starting from initial prolongation operators that are defined as the characteristic functions of each coarse block (i.e., equal unity inside the block and zero outside), a localized iterative scheme is applied directly to the fine-scale discretization to modify the prolongation operators so that they become increasingly consistent with the local properties of the differential operators. The use of weighted Jacobi smoothing on interpolation operators have been used with a large degree of success in the algebraic multigrid (AMG) community where fast coarsening is combined with simple operators constructed via one or two smoothing steps [12, 55–57] as an inexpensive alternative to the interpolation operators used in standard AMG [53]. Many high performance multigrid solvers support smoothed aggregation as a strategy for large, complex problems [10, 17] due to the inexpensive coarsening and interpolation strategies. A series of numerical experiments show that the new MsRSB method gives highly accurate prolongation operators for a wide variety of block shapes, e.g., including blocks that adapt to complex geological features in real-world models. Moreover, whereas methods like MsFV, MsMFE, and MsTPFA recompute the prolongation operator locally when faced with mobility changes in the underlying grid, the new method just continues the iteration until the operators are sufficiently smooth. This way, the cost of updating the prolongation operator becomes proportional to the amount of change in fluid mobility, eschewing the typical tolerance based updates. The formulation is algebraic and can be applied directly to linear systems, possibly in combination with existing multiscale techniques such as local stages and iterative cycles [58]. Through a series of numerical experiments, which include the well-known SPE 10 data and grid and petrophysical properties from two Norwegian oil fields, we validate our new MsRSB method and show that it is robust and efficient for single-phase and multiphase incompressible flow models. In a companion paper [45] we discuss how to extend the method to compressible flow problems and demonstrate that it provides one order-of-magnitude speedup compared to a fully-implicit simulator with the constrained pressure residual (CPR) preconditioner and algebraic multiscale preconditioner for compressible water-injection cases. Møyner [41] reports a comparison of the MsFV, MsTPFA, and MsRSB methods used as iterative solvers and shows that the MsRSB either performs equally well or clearly outperforms the other two methods for the studied test cases.

## 2. Model Problems

Multiscale methods, as discussed herein, are designed to efficiently compute the approximate action of second-order elliptic differential operators of the form $\nabla\cdot\mathbf{K}(\boldsymbol{x})\nabla$, where the coefficient $\mathbf{K}(\boldsymbol{x})$ may exhibit orders of magnitude variations over short distances and contain short, intermediate, and long-range correlations. This operator primarily determines the pressure distribution, but may also govern temperature in thermal models.

### 2.1. Single-phase flow.

To introduce the multiscale method and investigate its spatial approximation properties, it is sufficient to consider a incompressible single-phase flow in the absence of gravity, which is modeled by the variable-coefficient Poisson equation,

$$-\nabla\cdot(\mathbf{K}(\boldsymbol{x})\nabla p(\boldsymbol{x})) = q(\boldsymbol{x}), \qquad \boldsymbol{x}\in\mathbb{R}^d, \qquad \mathbf{K}(\boldsymbol{x})\in\mathbb{R}^d\times\mathbb{R}^d, \tag{1}$$

where $p$ is the fluid pressure, $\mathbf{K}$ is the permeability, and $q$ denotes source terms. We discretize this equation using a standard finite-volume scheme,

$$\sum_j v_{ij} = q_i, \qquad v_{ij} = -T_{ij}(p_i - p_j), \tag{2}$$

where the transmissibility $T_{ij}$ is associated with the interface between each pair of two cells $i$ and $j$ and defines a two-point flux approximation to the flux across this interface. For a Cartesian grid in 3D, (2) gives the standard seven-point finite-difference stencil. The resulting linear system

$$\boldsymbol{Ap} = \boldsymbol{q} \tag{3}$$

is weakly diagonally dominant because each equation represents volume conservation over a single cell.

2.2. **Multiphase flow.** The basic model for multiphase flow consists of conservation of mass and Darcy's law for each phase $\alpha$,

$$\partial_t\big(\phi\rho_\alpha S_\alpha\big) + \nabla\big(\rho_\alpha \vec{v}_\alpha\big) = \rho_\alpha q_\alpha, \qquad \vec{v}_\alpha = -\lambda_\alpha \mathbf{K}\big(\nabla p_\alpha - \rho_\alpha g \nabla z\big) \tag{4}$$

Here, $\phi$ denotes porosity, $g$ is the gravity constant, and $z$ the coordinate in the vertical direction, whereas $S_\alpha$ is the saturation (volume fraction) and $\lambda_\alpha = k_{r\alpha}/\mu_\alpha$ the mobility of phase $\alpha$, where $k_{r\alpha}$ is the relative permeability and $\mu_\alpha$ the viscosity of the phase. This model has more unknowns than equations and we must therefore specify an additional closure relationship for the saturations, $\sum_\alpha S_\alpha = 1$, as well as relationships for the phase pressures that express the individual capillary pressures as know functions of fluid saturations.

In the following, we only consider incompressible two-phase flow. There are several ways one can choose primary variables and reorganize the resulting system of equations to express it as an elliptic equation for flow (pressure and fluxes) and a hyperbolic equation for fluid transport. Herein, we use the pressure and saturation of the wetting phase as our primary unknowns, giving the pressure equation

$$-\nabla\cdot\big(\lambda\mathbf{K}\nabla p_w\big) = q_n + q_w - \nabla\big[\lambda_n\mathbf{K}\nabla p_{cnw} + \mathbf{K}(\lambda_n\rho_n + \lambda_w\rho_w)g\nabla z\big], \tag{5}$$

and the transport equation

$$\phi\partial_t S_w + \nabla\cdot\Big(f_w\big[\vec{v} + \lambda_n\mathbf{K}(\rho_w - \rho_n)g\nabla z\big]\Big) - \nabla\cdot\big(\lambda_n\mathbf{K}\nabla p_{cnw}\big) = q_w, \tag{6}$$

where we have introduced the total Darcy velocity $\vec{v} = \vec{v}_w + \vec{v}_n$, the capillary pressure $p_{cnw} = p_n - p_w$, the total mobility $\lambda = \lambda_w + \lambda_n$, and the fractional flow function $f_w = \lambda_w/(\lambda_w + \lambda_n)$. The three latter are known functions of $S_w$.

To solve the system (5)–(6), we use a sequential procedure, in which we first solve (5) to compute $p_w$ and $\vec{v}$ and then hold these constant while advancing (6) one time step. For spatial discretization of (5), we use the same two-point finite-volume scheme as for the single-phase equation, extended with upstream weighting of all terms that depend on saturation. (Note, however, that we could equally well have used a multipoint flux-approximation for the spatial discretization.) For (6), we use potential ordering to determine the upstream weighting [11] for the mobilities on the faces $\lambda_\alpha = k_{r\alpha}/\mu_\alpha$, which in most situations coincides with the upstream weighting used in the pressure solver. To allow for longer time steps, we use an implicit temporal discretization, giving a nonlinear discrete system that is solved with Newton's method. If necessary, one can also iterate the pressure and transport steps to ensure that the residual of the combined discrete system is below a prescribed threshold. However, for the cases considered later in the paper, we have not found this necessary.

## 3. Multiscale Formulation

As explained above, our definition of a multiscale method starts from a fine grid $\{\Omega_i\}_{i=1}^n$ and a coarse partition that defines a coarse grid $\{\bar{\Omega}_j\}_{j=1}^m$ so that each fine cell in $\Omega$ belongs to only one coarse block in $\bar{\Omega}$. We then define a numerical prolongation operator, $P : \{\bar{\Omega}_j\} \to \{\Omega_i\}$, that maps quantities associated with the coarse blocks to quantities associated with the fine cells. Likewise, we define a restriction operator as the analogous map going the other way $R : \{\Omega_i\} \to \{\bar{\Omega}_j\}$. In the implementation, these operators are represented as sparse matrices of size $n \times m$ and $m \times n$, respectively. If we now let $\boldsymbol{p}_c$ denote a pressure computed on the coarse grid, we can find a fine-scale approximate pressure $\boldsymbol{p}_f$ by the use of the prolongation operator,

$$\boldsymbol{p}_f = \boldsymbol{P}\boldsymbol{p}_c. \tag{7}$$

In general, this will not solve (1) exactly no matter how accurate the coarse pressure is; all we can hope for is to compute a good and accurate approximation more efficiently than solving (1) directly on $\Omega$.

### 3.1. Coarse system.

To derive a linear system for $\boldsymbol{p}_c$ on the coarse grid, we insert the fine-scale approximation in (7) into (3) and apply the restriction operator,

$$\boldsymbol{R}\left(\boldsymbol{A}\left(\boldsymbol{P}\boldsymbol{p}_c\right)\right) = (\boldsymbol{RAP})\boldsymbol{p}_c = \boldsymbol{A}_c\boldsymbol{p}_c = \boldsymbol{R}\boldsymbol{q} = \boldsymbol{q}_c. \tag{8}$$

The physical interpretation of this system depends on the restriction operator used. Two variants are reported in the literature, either a control volume summation operator or a Galerkin operator, i.e.,

$$(\boldsymbol{R}_{cv})_{ji} = \begin{cases} 1, & \text{if } \boldsymbol{x}_i \in \bar{\Omega}_j, \\ 0, & \text{otherwise,} \end{cases} \qquad \text{or} \qquad \boldsymbol{R}_G = \boldsymbol{P}^T.$$

Our focus is not on iterative performance, and hence it is natural to consider the control volume operator $\boldsymbol{R}_{cv}$ used in the classical MsFV method [22], which corresponds to setting the connection strength $(\boldsymbol{A}_c)_{ij}$ from coarse block $\bar{\Omega}_i$ into coarse block $\bar{\Omega}_j$ as the sum of the fluxes induced by the prolongation operator defined in block $i$ across the interfaces of $\bar{\Omega}_j$,

$$\int_{\partial\bar{\Omega}_j} \vec{v}_{P_i} \cdot \vec{n}\, dS \approx \sum_{(k,l)\in F_j} -T_{kl}(\boldsymbol{P}_{li} - \boldsymbol{P}_{ki}) = \boldsymbol{A}_{ij}$$

where we have defined $\vec{v}_{P_i}$ as the velocity of the basis function of block $i$, $F_j$ as the set of fine scale interfaces for $\bar{\Omega}_j$, represented as tuples of neighboring cells $(k, l)$ and approximated the flux using(2).

The pressure obtained by prolongating the solution of (8) back to the fine scale is generally not an exact solution of (3), but we can easily compute fluxes that are conservative on the coarse grid since (8) imposes mass balance on this grid. To get fluxes that are conservative also on the subscale, we need to solve an additional local problem with the conservative, coarse-scale fluxes imposed as Neumann boundary conditions. Using these fluxes, it is possible to solve fine-scale transport to a high accuracy without the exact pressure being known. The disadvantage is that we risk producing negative coarse-scale transmissibilities, which may lead to unphysical solutions having non-monotone pressure values that violate the maximum principle, see e.g., [44].

To define a specific multiscale method, we must also describe in detail how to construct the prolongation operator $P$. As in most other multiscale methods, we construct $P$ by piecing together a set of localized functions. However, before we can describe these so-called basis functions, we must provide more details about the coarse grid.

3.2. **Coarse grids and support regions.** Multiscale finite-volume formulations rely on a combination of a primal coarse grid and an auxiliary spatial characterization to obtain localized functions and assemble them together to form the prolongation operator. For the MsFV method, this additional mechanism is a dual coarse grid, whereas for the MsTPFA method [43], the additional mechanism is a set of localized partition-of-unity functions that are computed numerically over the fine grid. Previous research has shown that the choice of the primal grid (and the auxiliary spatial characterization) can have a pronounced impact on the solution quality. We therefore want a formulation that is as flexible as possible and enables basis functions to be computed for fairly general coarse grids. Even for problems posed on a Cartesian mesh without faults or pinched cells, the solution can be greatly improved with coarse grids that adapt to local features in the permeability field or in the flow patterns [41, 43, 44].

The primal coarse grid is defined through a partition vector that has a single unique indicator value per fine cell, so that cells with the same indicator value are agglomerated into coarse blocks. For structured grids, these indicators can simply be based on counting, forming logically hexahedral blocks on the coarse scale, whereas a wide variety of graph-based partitioning algorithms can be used in the unstructured case. Let $F$ be the index set of fine cells and let $C_j$ be the set of fine-scale indices corresponding to coarse block number $j$,

$$C_j \subseteq F, \qquad C_j \cap C_i = \emptyset \quad \forall \quad i \neq j, \qquad i, j \in [1, m], \qquad |F| = n.$$

Once we have defined a coarse grid, we must define the *support regions* that determine the support of the basis functions. If we let $I_j$ denote the set of all points contained in the support region for coarse block $\bar{\Omega}_j$ and $P_j$ the basis function of coarse block $j$, this implies that

$$P_j(\boldsymbol{x}) > 0, \quad \boldsymbol{x} \in I_j \qquad P_j(\boldsymbol{x}) = 0 \text{ otherwise.}$$

Likewise, we define the *support boundary* $B_j$ as the index set of cells that are topological neighbors to the support region $I_j$, but are not themselves contained in it. We also define the *center* of a coarse block $\bar{\Omega}_j$ as $\boldsymbol{x}_j^c$. This is a single point, which in many cases *may* coincide with the centroid of the coarse block. For convenience, we also define the global support boundary $G$ as the index set of all fine cells that are part of the support boundary of one or more coarse blocks,

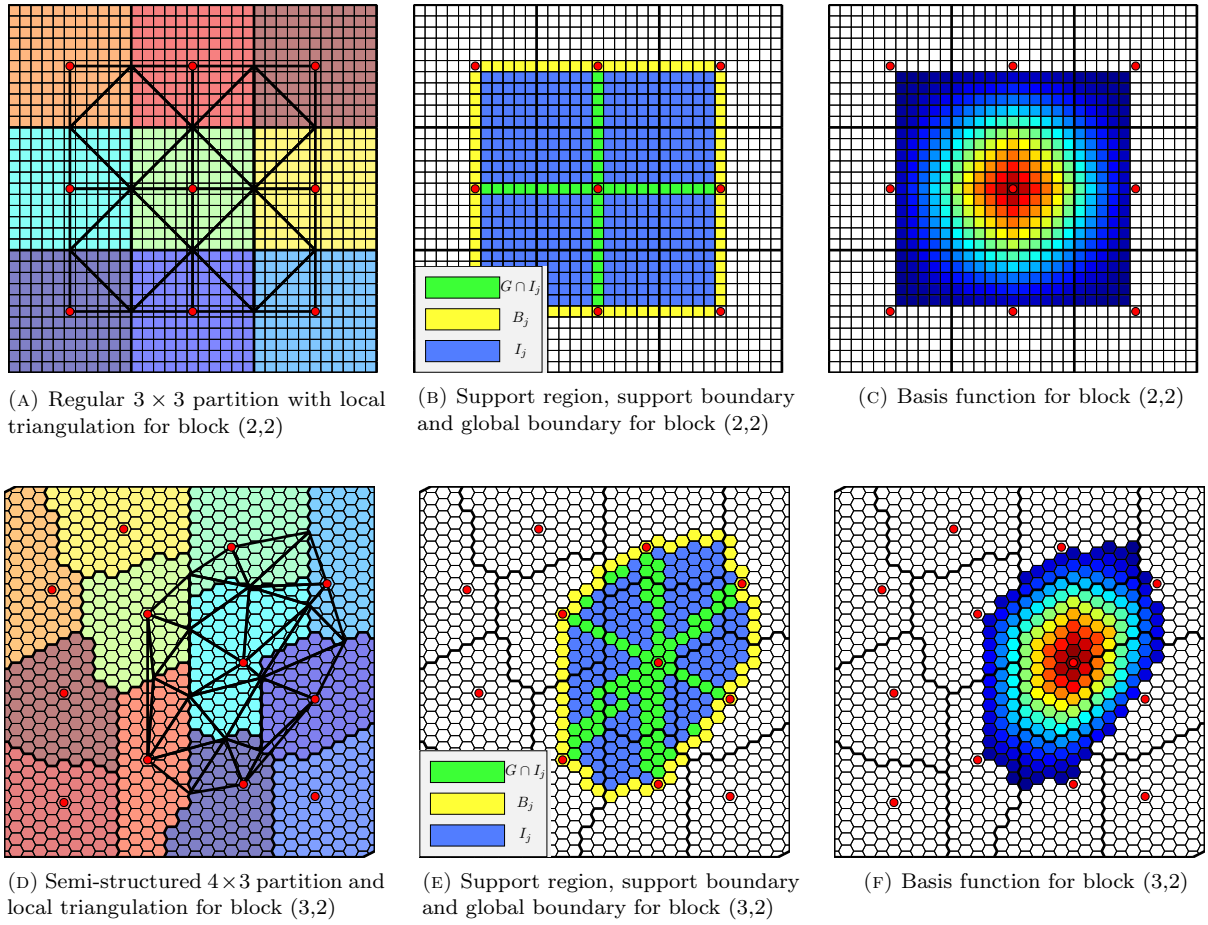
$$G = B_1 \cup B_2 \cup ... \cup B_{m-1} \cup B_m.$$

Finally, for each fine cell that is member of the global support boundary, we define $H_i$ to be the set of indices of the support regions the cell belongs to,

$$H_i = \{j \,|\, i \in I_j, i \in G\}.$$

To make this notation easier to visualize, we refer to Figure 1, which shows a regular partition for a uniform Cartesian grid and a semi-structured partition for a PEBI grid. The figure also shows the construction of support regions, support boundaries, and basis functions.

Careful numerical experiments have shown that rather than setting the block centroid as the block center, we should choose the block centroid as the fine cell whose centroid is closest to the geometric median of the fine-scale faces that bound the block. For regular coarse grids, the two choices coincide, but the geometric median gives basis functions of better quality for coarse partitions with large variation in block sizes and shapes. Figure 2 shows an exaggerated example with a large block neighboring ten small blocks that each consists of a single fine-scale cell. In Figure 2b the block center is chosen as the block centroid. Because of the disparity in size between the block and its neighbors to the west, the support regions are defined so that there is a relatively large region inside the block where only the basis function associated with the block is the only one that has supported. To ensure partition of unity, the basis function will therefore be constant and equal the maximum value of one in this region. Using the geometric mean instead, as in Figure 2c, means that the basis functions in the north, south, and east neighbors are supported in a larger portion of the center block, which reduces the constant region and improves the approximation quality of the




(A) Regular $3 \times 3$ partition with local triangulation for block (2,2)

(B) Support region, support boundary and global boundary for block (2,2)

(C) Basis function for block (2,2)

(D) Semi-structured $4 \times 3$ partition and local triangulation for block (3,2)

(E) Support region, support boundary and global boundary for block (3,2)

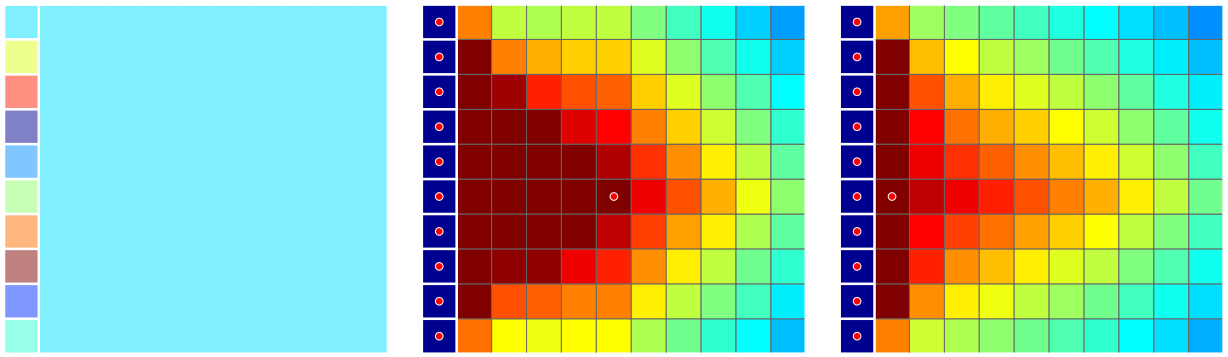
(F) Basis function for block (3,2)

FIGURE 1. Primal coarse grid and construction of a local support region with an associated basis function for a uniform Cartesian grid in the top row and a perpendicular bisector (PEBI) grid in the bottom row.

prolongation operator significantly, so that it, for instance, can better reproduce a linear pressure drop.

To define the support region of block number $i$, we select all blocks that share a coarse node with $\bar{\Omega}_i$ and create a local triangulation based on the block centers and the centroids of all coarse faces that are shared by any two of these blocks, see Figures 1a and 1d. The support region is then defined as all cells within the selected coarse blocks whose centroids lie within the triangulation, and the support boundary is defined as all cells that share at least one face with cells in the support region. Support regions are not allowed to include any center cells aside from their own. Because all these relations can be produced using only topology information, block centers, and face centroids, the implementation is the same regardless of whether we identify support regions in a two-dimensional Cartesian grid or in a complex 3D unstructured grid. If needed, one can include a simple post-processing to ensure that each support region only consists of cells that are connected in the graph defined by the fine-cell faces.

3.3. **Construction of basis functions.** Most multiscale methods rely on numerical solution of localized flow problems to produce the basis functions that form the prolongation operator. These local problems are typically defined as some subset of the global problem with alternate boundary conditions imposed to capture the local features. Herein, we deviate from this and instead construct



(A) Coarse block with refined neighbors (B) Basis function using centroids (C) Basis function using geometric mean

FIGURE 2. The choice made for the definition of the center points affects the quality of the prolongation operator. This example shows a coarse block where the neighboring blocks to the west consist of a single cell, while the neighbors to the north, south, and east are of the same size as the block itself. The prolongation operator takes values in the interval $[0, 1]$ and assumes a more correct, skewed hat shape when the block center is defined as the geometric median of the face centroids rather than as the block centroid.

the basis functions using an iterative process. A similar approach can be found in some multigrid methods that employ a single step of a smoother applied directly to a simple prolongation operator to reduced local error (Jacobi interpolation), see [55].

The basis functions are initiated as the characteristic function of each coarse block,

$$P_{ij}^0 = \begin{cases} 1 & \text{if } i \in C_j \\ 0 & \text{otherwise.} \end{cases}$$

Nothing is preventing us from choosing some more intricate initial guess, but constant functions are convenient because they are trivial to construct and automatically provide partition of unity. We then define a local smoothing iteration,

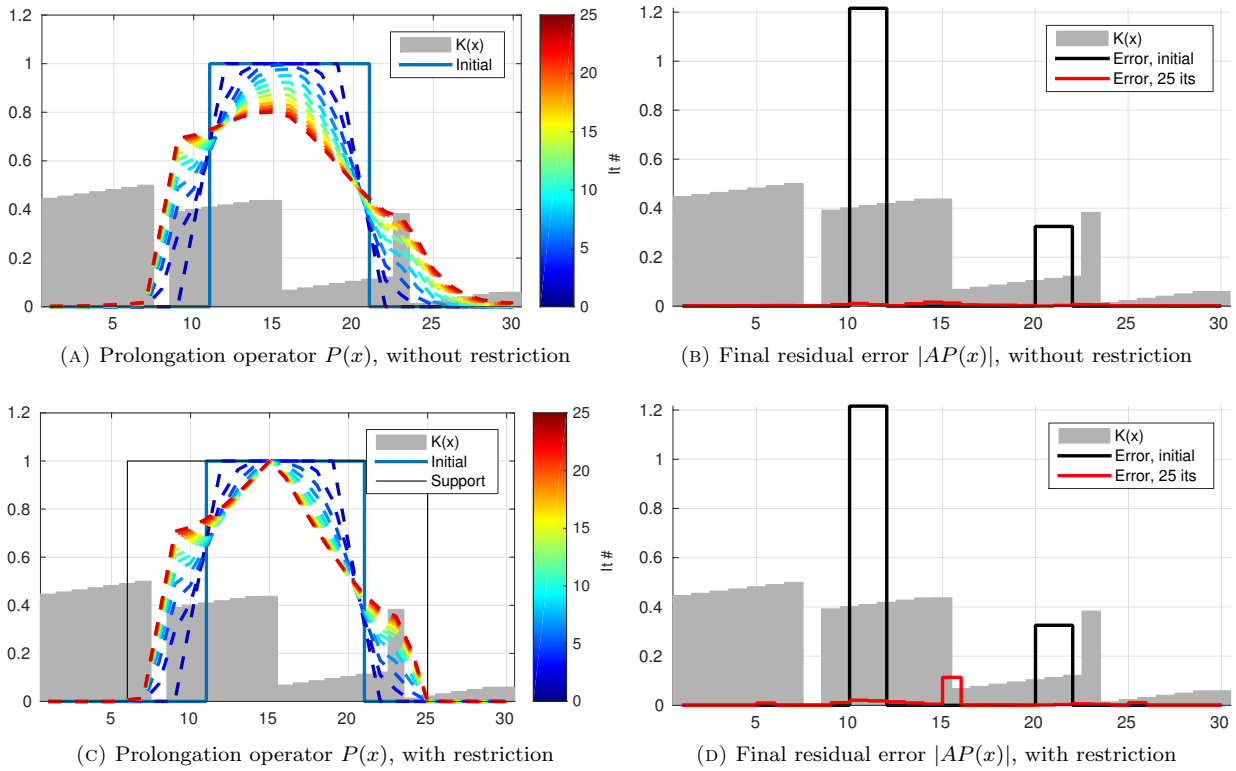
$$P_j^{n+1} = P_j^n - \omega D^{-1} A P_j^n, \tag{9}$$

where $D$ is a diagonal matrix that contains the diagonal entries of our (weakly) diagonally dominant system matrix $A$, and $\omega \in (0, 1]$ is a relaxation factor which we set to $2/3$ for all cases considered in this paper. The value $2/3$ is the optimal choice for Jacobi's method applied to Poisson's equation with constant coefficients. Better choices of $\omega$ will speed up convergence of the basis construction, but are not necessarily obvious for general problems. By iterating on the prolongation operator, we seek to make it algebraically smooth, i.e., reduce $\|AP\|_1$ as much as possible. This means that the residual error in the prolongation operator should be relatively smooth. Figure 3 shows a simple 1D example of how the operator and the associated error change as more iterations are applied. Because each iteration modifies cell values based on the topological neighbors, we can see that the support of the basis functions will eventually grow to cover the entire domain. To avoid this, we use our already defined support regions and support boundaries to localize the updates.

Roughly speaking, the above construction (9) determines an increment for each basis function based on the local error and modifies this increment to limit the support to be within the support regions. This update is also used to determine convergence of the basis construction procedure:

(1) Apply the smoother $D$ to find the increment of the discrete basis function,

$$\widehat{\mathbf{d}}_j = -\omega D^{-1} A P_j^n.$$




(A) Prolongation operator $P(x)$, without restriction

(B) Final residual error $|AP(x)|$, without restriction

(C) Prolongation operator $P(x)$, with restriction

(D) Final residual error $|AP(x)|$, with restriction

FIGURE 3. Illustration of how the iteration (9) and its restricted counterpart gradually smooth a basis function until it has very low residual error.

(2) Modify the update to avoid stencil growth outside of the support region and preserve partition of unity,

$$d_{ij} = \begin{cases} \dfrac{\widehat{d}_{ij} - P_{ij}^n \sum_{k \in H_i} \widehat{d}_{ik}}{1 + \sum_{k \in H_i} \widehat{d}_{ik}}, & i \in I_j, i \in G, \\ \widehat{d}_{ij}, & i \in I_j, i \notin G, \\ 0, & i \notin I_j. \end{cases}$$

(3) Update basis functions

$$P_{ij}^{n+1} = P_{ij}^n + d_{ij}$$

(4) Define local error outside of boundary regions,

$$e_j = \max_i(|\widehat{d}_{ij}|), \qquad i \notin G$$

(5) If $\|\mathbf{e}\|_\infty >$ tol, go to Step 1, otherwise set $P = P^{n+1}$.

In practice, checking for convergence should only be done every tenth iterations or so, since a single iteration has negligible cost. We note that Step 1 of this process is well suited to parallel processing. As each value only depends on the fine neighbors, it can easily be computed using on streaming processors such as GPUs. The modifications in Step 2 at the global boundary depend on the values in several coarse blocks, so each basis function will depend on the previous value of the other basis functions with support in the same cell. While this does make the basis functions use information about each other, the dependence will only extend to the nearest neighbors during each step, and only in a subset of the cells, analogous to the matching boundary conditions used on edges in the classical MsFV.

Because we have explicitly enforced partition of unity in cells belonging to the global boundary, we must now show that the updates $\widehat{d}_{ij}$ preserve the same property in cells not on the boundary. We assume that the row sum of the matrix used for the iterations is zero and that the initial prolongation operator has partition of unity,

$$\sum_j A_{ij} = 0, \qquad \sum_j P_{ij}^0 = 1 \quad \forall \quad i.$$

If needed, one can easily ensure that the iteration matrix has zero row sum by adjusting the diagonal elements. This requirement also applies to the classical MsFV method, so the same adjustment is used if special basis functions for wells, boundary conditions, and compressibility are not constructed. We can then write out the explicit update for a single cell $i$, summed over all coarse blocks,

$$\begin{aligned}\sum_j P_{ij}^{n+1} =& \sum_j P_{ij}^n - \frac{\omega}{A_{ii}} \sum_j \sum_k A_{ik} P_{kj}^n \\ =& 1 - \frac{\omega}{A_{ii}} \sum_k A_{ik} \left( \sum_j P_{kj}^n \right) = 1 - \frac{\omega}{A_{ii}} \sum_k A_{ik} = 1,\end{aligned}$$

showing that the update always preserves partition of unity.

For completeness, we will also verify that the proposed updates for cells in $G$ satisfy partition of unity. If we recall that $\sum_{j \in H_i} P_{ij}^n = 1$ by assumption and that $P_{ij}^n$ is nonzero only in $H_i$, we can explicitly write out the sum over basis functions for a cell $i$ in $G$ at step $n+1$,

$$\begin{aligned}\sum_{j \in \{1,\ldots,m\}} P_{ij}^{n+1} =& \sum_{j \in H_i} \Big( P_{ij}^n + \frac{\widehat{d}_{ij} - P_{ij}^n \sum_{k \in H_i} \widehat{d}_{ik}}{1 + \sum_{k \in H_i} \widehat{d}_{ik}} \Big) = 1 + \sum_{j \in H_i} \frac{\widehat{d}_{ij} - P_{ij}^n \sum_{k \in H_i} \widehat{d}_{ik}}{1 + \sum_{k \in H_i} \widehat{d}_{ik}} \\ =& 1 + \frac{\sum_{k \in H_i} \widehat{d}_{ik}}{1 + \sum_{k \in H_i} \widehat{d}_{ik}} - \frac{\sum_{k \in H_i} \widehat{d}_{ik}}{1 + \sum_{k \in H_i} \widehat{d}_{ik}} \sum_{j \in H_i} P_{ij}^n \; = 1.\end{aligned}$$

Figure 4 illustrates how the prolongation operator changes to adapt to structures in the underlying medium. For comparison, we have also included plots of the corresponding prolongation operators for the MsFV method [22]. Note that while the MsRSB method coincides with MsFV for the case with homogeneous permeability and to a certain extent the lognormal Tarbert layers, there are large differences for the problems with anisotropy and channelized permeability sampled from Upper Ness.

To accurately represent Dirichlet boundary conditions, we use a similar approach as shown for the center block in Figure 2c. That is, we move the block center to a fine cell that is adjacent to the boundary; preferably to the cell that lies closes to the centroid of the block face. This way, we ensure that the corresponding basis function decays smoothly out from the Dirichlet boundary. Using this approach, we can reproduce a linear pressure drop in a homogeneous domain. No other special treatment is required for no-flow boundaries.

3.4. **Iterative multiscale formulation.** Multiscale finite-volume methods have a link to multigrid methods in the sense that they can be used as two-level methods in combination with a *smoother* step that takes care of localized errors. This can be used for error control, to treat compressibility and nonlinear behavior, or to systematically drive the fine-scale residual towards zero. In other words, the MsRSB method can be used in three different ways: as a linear solver for the fine-scale system (see [41]); as an approximate solver that only reduces the fine-scale residual below a prescribed, relaxed tolerance and still guarantees a mass-conservative approximation, or as a one-step approximate solver that is mass-conservative on the fine scale, but has no guarantee on the size of the fine-scale residual.

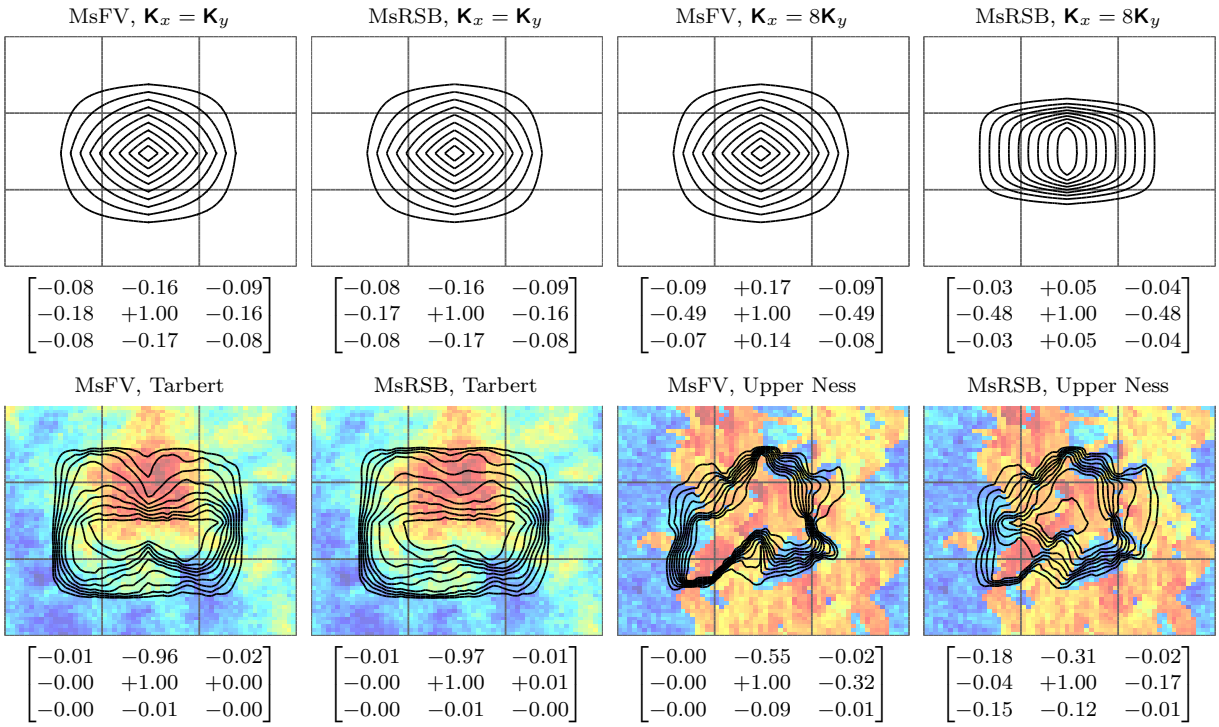



FIGURE 4. Matrix-dependent interpolation operators for a single coarse block with different types of permeability types. The lognormal and the channelized permeabilities in the lower row are both sampled from the SPE10 dataset [14]. The matrices report the net fluxes into or out of the neighboring coarse blocks induced by a unit pressure differential.

Let $P$ be the converged prolongation operator for a given coefficient matrix $\boldsymbol{A}$. To define an iterative scheme, we let the solution at step $k$ be denoted $\boldsymbol{x}^k$ and introduce the defect

$$\boldsymbol{d}^k = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^k,$$

If we let $\boldsymbol{y}^k = S(\boldsymbol{d}^k)$ denote the smoother applied to the defect with initial guess zero, we can then write the next update as the previous solution with the smoothed update added in, along with a *coarse correction* that ensures that the update does not remove the coarse-scale conservative property of the solution,

$$\boldsymbol{x}^{k+1} = \boldsymbol{x}^k + P\big(A_c^{-1} R(\boldsymbol{d}^k - A\boldsymbol{y}^k)\big) + \boldsymbol{y}^k.$$

This iterative scheme relies on an inexpensive smoother for the updates. In the following, we use a incomplete LU-factorization with zero fill in (ILU0) for systematic iteration tests and a few Jacobi iterations for problems where we only want to reduce some local error. One pass of the smoother plus the coarse correction is termed a multiscale cycle. For the problems considered herein, we will let $\boldsymbol{x}^0 = 0$.

3.5. **Flux reconstruction.** The multiscale solution $\boldsymbol{p}_c$ is mass conservative on the coarse scale by construction. However, if we use the prolongated pressures to construct fine-scale fluxes from Darcy's law (see (2)),

$$v_{ij}^{ms} = -T_{ij}\big((\boldsymbol{P}\boldsymbol{p}_c)_i - (\boldsymbol{P}\boldsymbol{p}_c)_j\big), \tag{10}$$

these fine-scale fluxes will not be mass-conservative since $\sum_j v_{ij}^{ms} \neq q_i$. To get conservative fluxes on the fine grid (and hence also on any grid of intermediate resolution), we need to compute a new *reconstructed* pressure $\bar{p}$ to reconcile errors in the pressure gradient with the flux field as formulated

for the classical MsFV method in [22]. We define the reconstructed pressure $\bar{p}$ by solving (1) locally for each coarse block $C_i$ with flux boundary conditions obtained from the multiscale pressure over the coarse edges.

$$-\nabla \cdot (\mathbf{K}\nabla \bar{p}(\boldsymbol{x})) = q(\boldsymbol{x}), \boldsymbol{x} \in \bar{\Omega}_i \qquad \nabla \bar{p}(\boldsymbol{x}) \cdot \vec{n} = v^{ms} \text{ on } \partial\bar{\Omega}_i.$$

Once the reconstructed pressure is found, the velocity field inside each coarse block is found using Darcy's law. The fluxes over the coarse edges are the same as were used for boundary conditions. For the unstructured implementation, we use the operator form for posing these problems, see [38, 59].

3.6. **Treatment of wells.** Herein, we consider wells that are either controlled by rate or by bottom-hole pressure. Wells perforated in a single fine cell are not much different from source terms or boundary conditions. For wells with multiple completions, potentially in different coarse blocks, we need to be a bit more careful. The standard way to treat a well is to use a set of source terms resulting from the pressure drop along the bore, i.e., if $p_i$ is the bottom-hole pressure in the well and $p_j$ the cell pressure, the well model gives us,

$$q_{ij} = \lambda_j^T W_I (p_j - p_i - \rho \vec{g} \nabla z).$$

where $W_I$ is the productivity/injectivity index and $\lambda^T$ the total mobility in the cells. In addition to this, a closure equation is defined per control: $\sum_j q_{ij} = \bar{q}_i$ for rate controls and $p_i = \bar{p}_i$ for pressure controls. In our multiscale framework we keep the control equations at the coarse scale to correctly account for inter-block flow on the coarse scale. For the bottom-hole controls, we take the well fluxes from the reconstructed pressure, as the pressure drop makes the reconstruction problems well posed and consistent with the outgoing block fluxes. For rate controlled wells, we use the fluxes defined by the prolongated pressure, as the well-to-cell flow is analogous to the flow between two coarse blocks.

## 4. Numerical Experiments

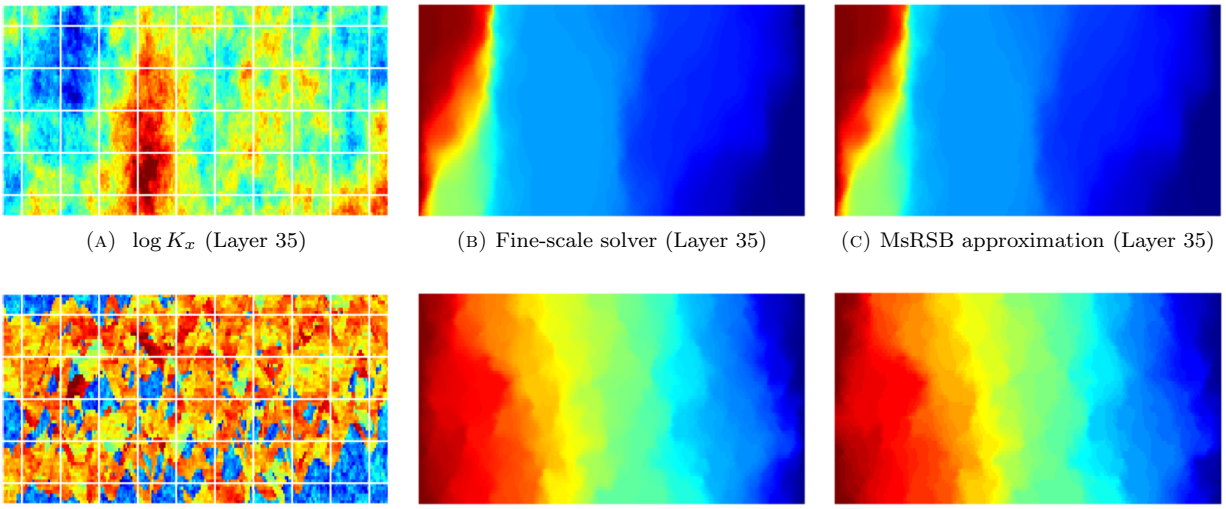
The MsRSB method introduced above has been implemented using the Matlab Reservoir Simulation Toolbox (MRST), see [29, 31, 32, 46], and is released as a part of the 2015a release. In the following we report the result of a series of numerical experiments we have run to validate the multiscale formulation, verify our implementation, and demonstrate the utility of the resulting solver. To this end, we consider a variety of test problems, from simple 2D Cartesian geometries to geological models representing petroleum reservoirs on the Norwegian Continental Shelf.

4.1. **Spatial accuracy.** To assess the spatial accuracy of the MsRSB method, we consider the single-phase model (1) applied to two different test cases: (i) the SPE 10 data set, which seem to be a de facto benchmark for new multiscale methods, and (ii) a model that uses the grid geometry and petrophysical properties from a simulation model of the Gullfaks field. For both models, we investigate the discrepancy between the multiscale approximation and the fine-scale reference solution measured by the scaled $L^\infty$ and $L^2$ norms,

$$\|\boldsymbol{p}^{fs} - \boldsymbol{p}^{ms}\|_\infty = \frac{\max_{i\in F} |\boldsymbol{p}_i^{fs} - \boldsymbol{p}_i^{ms}|}{\max_{i\in F} |\boldsymbol{p}_i^{fs}|}, \qquad \|\boldsymbol{p}^{fs} - \boldsymbol{p}^{ms}\|_2 = \sqrt{\frac{\sum_{i\in F} |\boldsymbol{p}_i^{fs} - \boldsymbol{p}_i^{ms}|^2 |\Omega_i|}{\sum_{i\in F} |\boldsymbol{p}_i^{fs}|^2 |\Omega_i|}}, \tag{11}$$

where $\boldsymbol{p}_i^{fs}$ and $\boldsymbol{p}_i^{ms}$ denote the pressure values computed in cell $\Omega_i$ by the fine-scale and the multiscale methods, respectively. Discrepancies in reconstructed fluxes are defined analogously.

4.1.1. *SPE 10 data set.* Model 2 from the 10th SPE Comparative Solution Project [14] was originally designed as a challenging benchmark for upscaling methods. The model is described on a $60\times220\times85$ Cartesian grid with cells of uniform size $20 \times 10 \times 2$ ft$^3$. The reservoir contains two sands sampled from a Brent sequence with very different heterogeneity. In the Tarbert formation found in the top 35 layers, the permeability follows a lognormal distribution, giving smoothly varying heterogeneities that are resolved quite well by most multiscale methods. The Upper Ness formation found in the bottom 50 layers is fluvial and consists of an intertwined pattern of long and high-permeable sand channels interbedded with low-permeable mudstone. The combination of very long correlation lengths and many orders-of-magnitude difference in permeabilities between neighboring cells makes Upper Ness very challenging to resolve accurately.
Horizontal layers. First, we consider flow in two horizontal $60 \times 220$ slices with isotropic permeabilities sampled from the top and bottom layers of the model, subject to fixed pressure of one hundred bar on the left and zero bar on the right boundary. The domain is partitioned into coarse blocks made up of $10 \times 20$ fine cells so that the coarse grid blocks are square in the interior of the domain. Near the edges of the domain, we add coarse blocks that are half as wide as the other blocks in the $x$ or $y$ direction, respectively, see Figure 5. In each of these blocks, we move the block center to the fine cell that lies closest to the face centroid of the block. This gives a total of $6 \times 11$ coarse blocks, which corresponds to an upscaling factor of 200 in the interior and 40 near the boundary.



(A) $\log K_x$ (Layer 35) (B) Fine-scale solver (Layer 35) (C) MsRSB approximation (Layer 35)

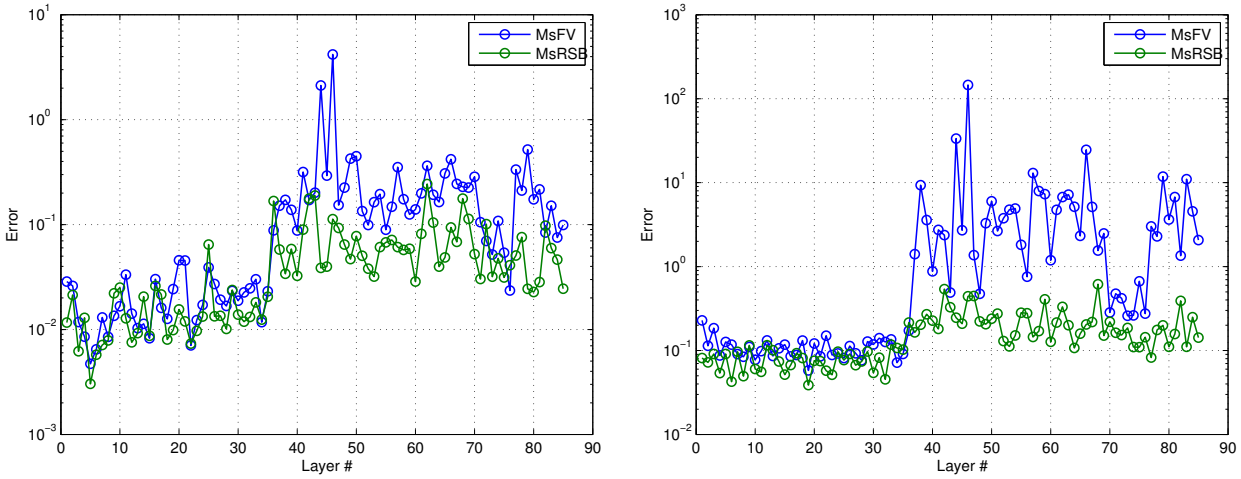
(D) $\log K_x$ (Layer 85) (E) Fine-scale solver (Layer 85) (F) MsRSB approximation (Layer 85)

FIGURE 5. Permeability and pressure solutions for the top and bottom layers of the second SPE 10 dataset. Flow is driven by a difference in the fixed pressures specified at the left and right boundaries.

Figure 5 shows the permeability for both layers and compares the pressure fields computed by the fine-scale solver and by MsRSB using a single multiscale solve without subsequent iteration cycles. Table 1 reports the corresponding discrepancies measured in the relative $L^2$ and $L^\infty$ norms defined in (11). For comparison, we also report discrepancies for the original MsFV method as implemented in the `msfvm` module of MRST, see [44] for details. Whereas the solution quality is generally very good for both solvers on the Tarbert subsample, MsRSB clearly outperforms MsFV on Upper Ness. Figure 6 reports discrepancies for similar experiments performed on all horizontal layers in the model. Several authors have independently shown that the MsFV method has issues with coarse-scale stability in the presence of channelized, high-contrast formations and will suffer from strong unphysical oscillations that may prevent iterative versions of the method from converging properly,

TABLE 1. Discrepancy between the fine-scale solution and approximate solutions computed by the MsFV and MsRSB methods for the 2D test problems shown in Figure 5. Total flux is the flux over the outflow edge normalized by the corresponding flux in the fine-scale reference solution.

| Setup of simulation | | | Pressure | | Flux | | |
|---|---|---|---|---|---|---|---|
| Formation | layer | solver | $L^2$ | $L^\infty$ | $L^2$ | $L^\infty$ | Total |
| Tarbert | 35 | MsFV | 0.0313 | 0.0910 | 0.1138 | 0.4151 | 0.9696 |
| | | MsRSB | 0.0204 | 0.0766 | 0.0880 | 0.4071 | 1.0121 |
| Upper Ness | 85 | MsFV | 0.2299 | 2.0725 | 0.4913 | 0.7124 | 0.8087 |
| | | MsRSB | 0.0232 | 0.0801 | 0.1658 | 0.3240 | 1.0936 |




(A) Pressure discrepancy, $L^1$ norm

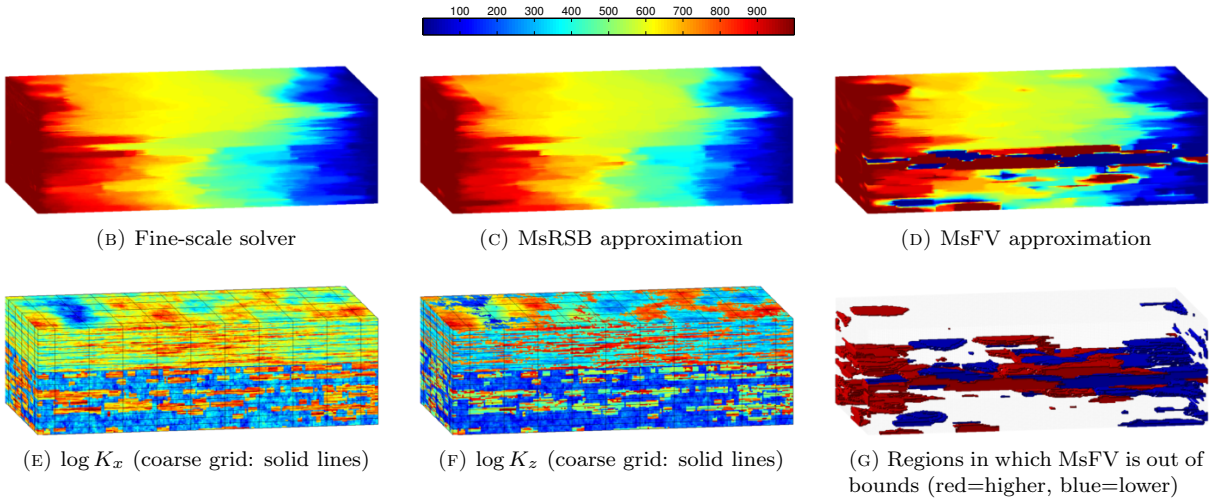
(B) Pressure discrepancy, $L^\infty$ norm

FIGURE 6. Discrepancy between the fine-scale solution and approximate solution computed by the MsFV and MsRSB methods for all horizontal layers of the SPE 10 model.

see e.g., [44, 52, 58] and references therein. MsRSB is much more robust and does not suffer from such problems and therefore has approximately the same level of accuracy for the smooth and the channelized layers.

Full 3D model. The flow patterns in the 3D model are more complex, and the combination of strong anisotropy and higher aspect ratios poses additional challenges for multiscale methods. To coarsen the 1.1 million-cell fine-scale model, we use the same strategy as in the previous example with $10 \times 20 \times 5$ fine cells per coarse block, giving coarse blocks of size $200 \times 200 \times 10$ ft$^3$ in the interior of the domain. Fixed pressures of one hundred and zero bar are prescribed on the east and west boundaries, respectively.

Pressure solutions and discrepancies are reported in Figure 7 and Table 2 and are in line with what we observed in 2D: Whereas MsFV and MsRSB both perform reasonably well in the upper part of the model, the original MsFV method becomes unstable in the lower channelized formation. For MsRSB, it is difficult to distinguish qualitative differences from the reference solution in Figure 7, which is also confirmed by the quantitative comparison in Table 2. SPE 10 is a challenging benchmark in terms of heterogeneity, and the good accuracy obtained with the MsRSB method without any kind of grid adaption or smoothing iterations is quite remarkable.




(B) Fine-scale solver (C) MsRSB approximation (D) MsFV approximation

(E) $\log K_x$ (coarse grid: solid lines) (F) $\log K_z$ (coarse grid: solid lines) (G) Regions in which MsFV is out of bounds (red=higher, blue=lower)

FIGURE 7. Comparison of pressure solutions computed by the fine-scale solver and the MsFV and MsRSB methods on the full SPE 10 model using a $6 \times 11 \times 17$ coarse grid. Flow is driven by a difference in fixed pressures specified at the west and east boundaries.

TABLE 2. Discrepancy between the fine-scale solution and approximate solutions computed by the MsFV and the MsRSB methods on two different coarse grids for the full SPE 10 model, see Figure 7. Total flux is the flux over the outflow face normalized by the corresponding flux in the fine-scale reference solution.

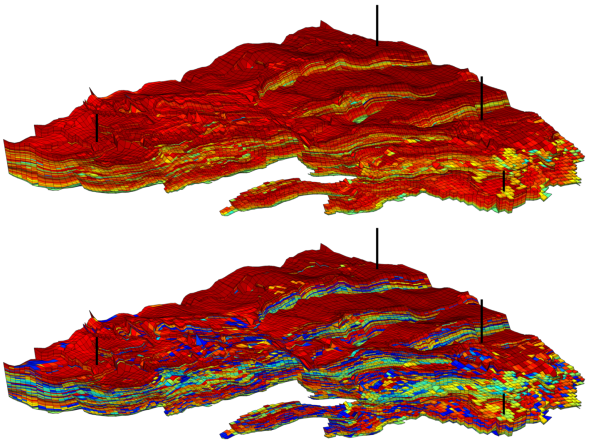
| Solver | Grid | Pressure | | Flux | | |
|---|---|---|---|---|---|---|
| | | $L^2$ | $L^\infty$ | $L^2$ | $L^\infty$ | Total |
| MsFV | $6 \times 11 \times 17$ | 3.580 | 128.461 | 2.288 | 11.957 | 0.9110 |
| MsRSB | $6 \times 11 \times 17$ | 0.039 | 0.309 | 0.397 | 0.487 | 1.2214 |
| MsFV | $P$-adapted | 0.054 | 0.259 | 1.418 | 1.397 | 1.8434 |
| MsRSB | $P$-adapted | 0.036 | 0.209 | 0.426 | 0.453 | 1.2179 |

To mitigate the unstable behavior of the MsFV method, we can modify the coarse grid so that the control volumes adapt to the local structures in the prolongation operator. The adapted grid is defined by computing a new index set

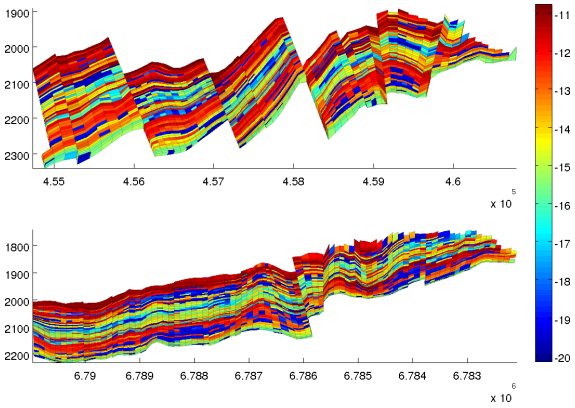
$$\hat{C}_k = \{i \,|k = \text{argmax}_j \, P_{ij}\}. \tag{12}$$

In other words, for each cell $i$ we find the local prolongation operator that has the largest nonzero cell value and assign cell $i$ to the corresponding coarse block. This gives a nonuniform grid that can be used as control volumes when formulating the coarse system. Table 2 shows that this gives a significant reduction in the discrepancy for the MsFV method, in particular in the $L^\infty$ norm since the approximate solution now is kept within the bounds of the boundary conditions. The pressure discrepancy is also slightly reduced for MsRSB, but the reconstructed flux is significantly less accurate because of the local irregularity of the coarse blocks.

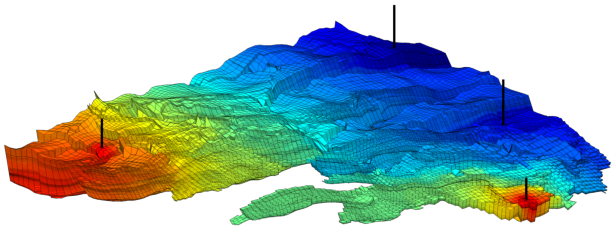
4.1.2. *Gullfaks field model.* Gullfaks is an oil and gas field located in the Norwegian sector of the North Sea that produces primarily from Brent sands, i.e., the same type of sedimentary environments as seen in the SPE 10 model. Unlike SPE 10, the Gullfaks field has a very complex structure and contains a large number of sloping faults, with angles varying from 30 to 80 degrees and throws



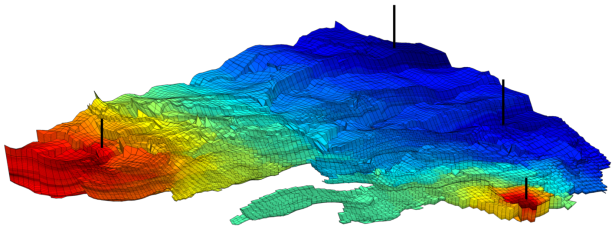
(A) $\log K_x$ (top) and $\log K_z$ (bottom)




(B) Vertical permeability in SI units for a west-east and a north-south crossection of the model



(C) Pressure computed by the fine-scale solver



(D) MsRSB approximation (637 blocks, Metis)

FIGURE 8. Computation of incompressible, single-phase pressure distribution using the grid geometry and the petrophysical data from a simulation model of the Gullfaks field. Pressure is set to 500 bar in the two injectors and 250 bar in the two producers.

from zero and up to three hundred meters; see [16] for a discussion of the structural geology. The simulation model is represented on a $80 \times 100 \times 52$ corner-point grid in which 216 334 cells are active. Almost 44% of the cells have non-neighboring connections, and when the corner-point grid is turned into a matching polyhedral grid, the number of cell faces range from four to thirty-one. The combination of strong heterogeneity, large anisotropy and aspect ratios, degenerate cell geometries, and unstructured grid topology makes the Gullfaks model very challenging for any multiscale solver. Rather than considering the wells pattern that has been drilled in the actual formation, we create a significant pressure drop of 250 bar across the majority of the field using four wells, two producers and two injectors, placed quite arbitrarily near the perimeter and perforated through all layers of the model.

We consider three different partitions. The first is a coarse grid initially partitioned with $15 \times 15 \times 20$ fine cells per coarse block. The large coarsening factor in the vertical direction is chosen because of the many inactive layers and cells. Any coarse block intersected by a fault or divided into disconnected components by inactive cells is then split into two before coarse blocks with very few cells are merged into their neighbors, resulting in a semi-structured coarse grid with 416 blocks. The second coarse grid is constructed by the use of Metis [23] configured with the logarithm of the transmissibilities of the fine-scale system as weights for the edge-cut minimization algorithm and a target set to 416 blocks to match the number of blocks in the structured partition. We also consider another set of finer partitions constructed by the same methodology with approximately twice as many coarse blocks (1028 after processing) to demonstrate how the accuracy of the MsRSB method

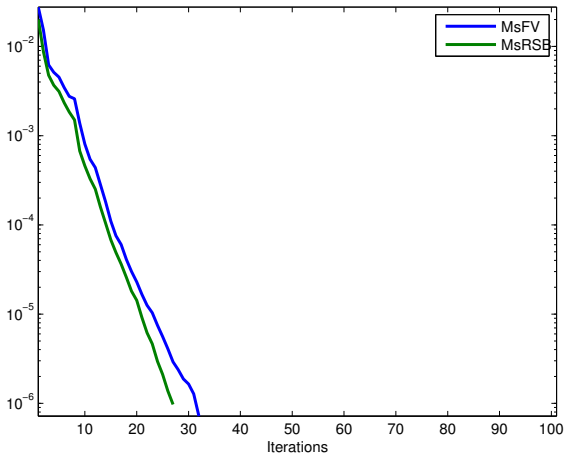
TABLE 3. Discrepancy between the fine-scale solution and approximate MsRSB solutions computed for the Gullfaks model shown in Figure 8. Total flux is the net flow out of the injectors normalized by the corresponding net flow from the fine-scale reference solution.

| Solver | Iter. | Coarse grid | | Pressure | | Flux | | |
|---|---|---|---|---|---|---|---|---|
| | | Block type | DoF | $L^2$ | $L^\infty$ | $L^2$ | $L^\infty$ | Total |
| MsRSB | — | $15 \times 15 \times 20$ | 416 | 0.032 | 0.102 | 1.256 | 1.329 | 1.639 |
| MsRSB | 5 | $15 \times 15 \times 20$ | 416 | 0.033 | 0.100 | 0.606 | 0.865 | 1.283 |
| MsRSB | — | Metis | 416 | 0.032 | 0.086 | 1.399 | 2.008 | 1.374 |
| MsRSB | 5 | Metis | 416 | 0.018 | 0.067 | 0.538 | 0.931 | 1.099 |
| MsRSB | — | $10 \times 10 \times 10$ | 1028 | 0.028 | 0.597 | 2.165 | 4.143 | 1.502 |
| MsRSB | 5 | $10 \times 10 \times 10$ | 1028 | 0.024 | 0.089 | 0.545 | 0.430 | 1.202 |
| MsRSB | — | Metis | 1028 | 0.015 | 0.112 | 1.347 | 1.399 | 1.310 |
| MsRSB | 5 | Metis | 1028 | 0.011 | 0.027 | 0.476 | 0.675 | 1.085 |

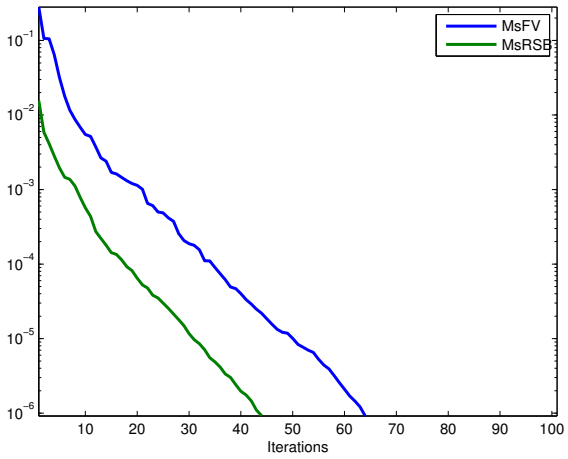
can be improved by coarse mesh refinement. We emphasize that no manual effort was required to create the three coarse partitions.

The approximate solution computed on the coarsest Metis-based grid can be seen in Figure 8, while Table 3 reports the discrepancies from the fine-scale solution for all the three coarse grids. In all cases the approximate pressure values stayed inside the global bounds, i.e., in the interval from 250 to 500 bar. We note that the multiscale approximations are quite accurate and that refining the coarse partition improves the accuracy at the cost of a larger coarse system. The table also reports discrepancies after applying five multiscale cycles of ten Jacobi iterations each, which confirms that local errors are quickly removed by the inexpensive smoother. Altogether, the results are very promising in view of the combined challenge posed by the partially degenerate cell geometries, very complex grid topology, and the large permeability contrasts. Being able to handle models of this level of structural and stratigraphic complexity in a robust and automated fashion is essential if the goal is to bring multiscale methods closer to practical usage. The interested reader can also consult [41] for a discussion of MsRSB used as an iterative linear solver for a case with seven injection and eleven production wells.

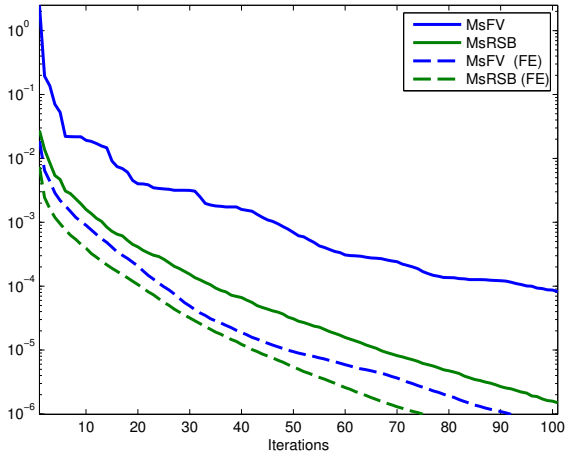
4.2. **Multiscale methods as iterative solvers.** The examples presented so far have only used multiscale methods as approximate solvers that are guaranteed to produce a conservative flux field regardless of the accuracy of the approximation. However, as explained in Section 3.4, multiscale methods can also be used as iterative solvers for the fine-scale system, e.g., in combination with GMRES. To demonstrate this capability for the MsRSB method, and compare its performance to that of the iterative MsFV method, we return to the SPE 10 test cases from Section 4.1.1. Figure 9 reports the convergence of the two multiscale methods used as iterative solvers for two horizontal layer as well as on the full 3D model. On the smooth Tarbert layer, both methods are able to reduce the fine-scale residual six orders of magnitude in approximately thirty iterations (27 iterations for MsRSB and 32 for MsFV). On the channelized Upper Ness layer, both methods require more iterations, but because of the improved stability of the MsRSB prolongation operator, the residual for MsRSB is significantly lower than for MsFV and decays faster. For the full 3D model, the MsFV prolongation operator tends to produce solutions that are outside of the bounds, as shown in Figure 7. Reduction of the residual is therefore mainly left to GMRES, which explains the slow convergence observed in Figure 9c. If the original mass-conservative control-volume restriction operator $\boldsymbol{R}_{cv}$ is replaced by the Galerkin operator $\boldsymbol{R}_G$, as suggested e.g., by Wang et al. [58], the MsFV method exhibits an acceptable convergence that is somewhat faster than MsRSB with finite-volume restriction and slightly slower than MsRSB with Galerkin restriction.




(A) Tarbert: Layer 35




(B) Upper Ness: Layer 85




(C) Full 3D model

FIGURE 9. Convergence history for GMRES-MS with the MsFV or MsRSB prolongation operators on 2D subsets and the full SPE 10 model.

We also ran a set of tests on the Gullfaks model used in Section 4.1.2 where we varied the number of coarse blocks to estimate the impact coarsening has on convergence for an essentially unstructured model. We used Metis to produce coarse grids with 10, 200, 250, 500 and 1000 blocks. The convergence is shown in Figure 10 where we see that adding more degrees of freedom appears to increase the convergence rate. Rapid convergence with a larger coarse system is to be expected, as more error modes will be in the null space of the coarse system. The choice of restriction operator does not have a large impact on the convergence rate, as the MsRSB operator produces a stable coarse system in either case.

Møyner [41] reports a more thorough assessment and comparison of the MsRSB, MsFV, and MsTPFA [43] prolongation operators used as pure multiscale solvers or as part of a GMRES iterative solver. The results confirm what we observed above: The MsFV method gives accurate multiscale solutions and converges rapidly on the smooth Tarbert formation irregardless of the restriction operator used. On Upper Ness, the method is accurate and efficient if one uses Galerkin reconstruction. The MsTPFA method is more robust but less accurate and efficient than MsFV on Cartesian partitions. However, if we use Metis to compute a coarse partition that adapts to

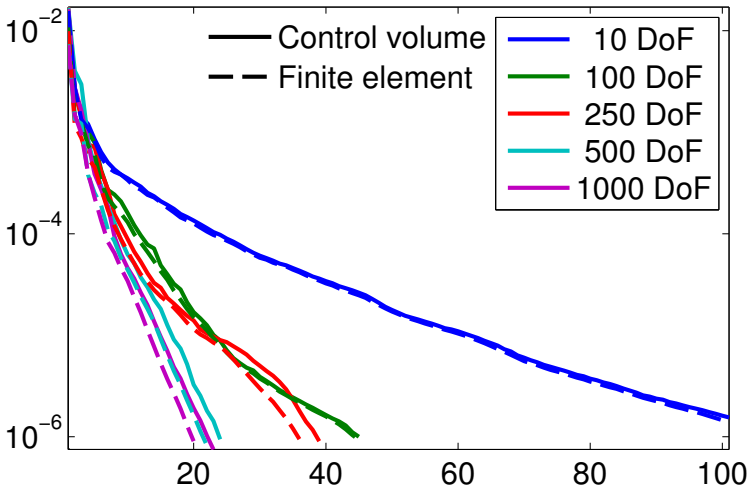



FIGURE 10. Convergence history for GMRES-MS for the Gullfaks field model with different coarsening ratios.

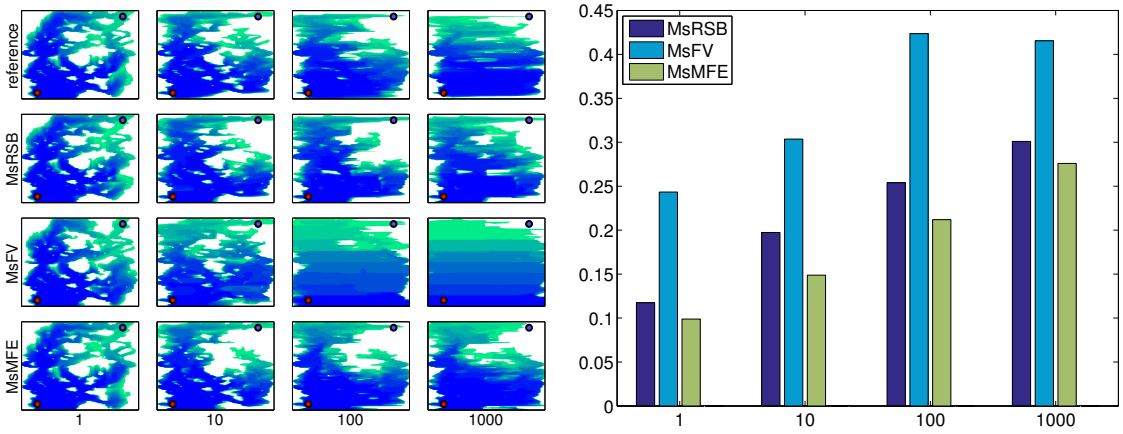
contrasts in the fine-scale transmissibilities, the MsTPFA method becomes as accurate and efficient as MsFV (which generally cannot be applied to such adapted grids). In all tests, however, the MsRSB method performs equally well or better than the other two methods and is less affected by the choice of restriction operator and type of coarse partition.

4.3. **Multiphase flow.** So far, we have mainly investigated how accurate the multiscale method is able to resolve the pressure in a given L-norm. In practical simulations, it is more important that the multiscale method gives fluxes that transport saturations/compositions correctly. For incompressible flow simulation, in particular, the pressure does not appear explicitly in the transport equation (6) and only influences the fluid displacement implicitly. Pressure fields are usually also much smoother than the associated flux fields and hence primarily reflects large-scale heterogeneous structures. To assess how well the multiscale methods resolve the influence of small-scale heterogeneous structures, we will investigate how the multiscale approximation affects the accuracy of the transport equation (6), measured using a relative $L^1$ norm

$$\|\phi S^{fs} - \phi S^{ms}\|_1 = \frac{\sum_{i\in F} \phi_i |S_i^{fs} - S_i^{ms}|\,|\Omega_i|}{\sum_{i\in F} \phi_i |S_i^{fs}|\,|\Omega_i|}. \tag{13}$$

Weighting the error by pore volumes makes this the error in spatial mass distribution for incompressible flow. To this end, we will, as in the previous sections, use a variety of test problems, from simple slices of the SPE 10 data set to a real field model.

4.3.1. *Sensitivity to aspect/anisotropy ratios.* Grids used in field and sector models predominantly have much larger cells in the horizontal than in the vertical direction, or likewise have much smaller vertical than horizontal permeability. Hence, it is important that multiscale methods are robust with regards to large aspect and anisotropy ratios. It is well known that the classical MsFV method is sensitive to high anisotropy and large grid aspect ratios and may produce approximate solutions having strong localized circulations in the reconstructed fine-scale flux field, see e.g., [25]. This will, in turn, give smeared saturation fronts in which the subscale resolution is completely lost because of the strong circulations. In [20], it is shown that deficiencies appear even for homogeneous cases because of the bilinear nature of the basis functions. The problem can be mitigated to a certain degree by a careful choice of boundary conditions for localization of basis functions [20, 36].




(A) Saturation profiles (B) Discrepancy in $L^1$ norm

FIGURE 11. Aspect ratio test for solutions computed by the MsRSB, MsFV, and MsMFE methods on a uniform $4 \times 8$ coarse grid covering a $60 \times 120$ subset of Layer 85 of the SPE 10 data set that has been stretched a factor 1, 10, 100, and 1000 in the $y$-direction.

For the coefficients in the discretized pressure equation, stretching the grid is equivalent to increasing the anisotropy ratio. We consider a $60 \times 120$ subset of Layer 85 of the SPE 10 data set, giving a square domain with $L_x = L_y = 1200$ ft in which water is injected from a well operating at fixed rate near the south-east corner and fluid is produced from a well operating at constant pressure near the north-east corner. We will compare the MsRSB, MsFV, and MsMFE multiscale methods with the same uniform $4 \times 8$ coarse grid for all three methods. To assess how well the multiscale methods resolve the transport properties of the flux field, we simulate the injection of one half pore volume of water using linear relative permeabilities with unit mobility ratio (i.e., $\lambda_w(S) = S$ and $\lambda_n(S) = 1 - S$), for which the equation system (5)–(6) becomes fully decoupled and can be solved using a single pressure step.

Figure 11 reports saturation profiles and discrepancy from the fine-scale solution on four different grids with $L_y = nL_x$, for $n = 1$, 10, 100, and 1000. The MsFV method has significantly larger errors than the other two methods and for aspect ratios 100 and 1000, all fine-scale details in the saturation field are lost. MsRSB is slightly less accurate than MsMFE, but although both methods become gradually less accurate in a pointwise sense as the aspect ratio increases, they manage to maintain a reasonable prediction of the qualitative behavior of the flow pattern.

4.3.2. *Temporal accuracy of multiscale approximation.* There are two different ways of using a multiscale solver for time-dependent problems. The first approach is to keep the same basis functions throughout the whole simulation, and either hope that the iterations can efficiently account for mobility changes or accept the corresponding reduction in local accuracy if the multiscale method is used without iterations. Alternatively, one can update the basis to account for dynamic changes in mobility. The usual way to do this is to recompute basis functions locally whenever the total mobility changes significantly. The key to get enhanced efficiency when using multiscale methods is to perform as few subgrid computations as possible and only where it is necessary. Smoothed basis functions are particularly efficient in this respect since they do not need to be completely regenerated: Because the underlying iterative process can start from any function having partition of unity, we can simply restart the iteration process with changed mobilities and continue until the basis functions are sufficiently smooth again.

The dynamic character of an incompressible two-phase flow system is often quantified by the ratio of the end-point values of the total mobility, usually called the mobility ratio for brevity. To discuss this, we consider a special choice of relative mobilities

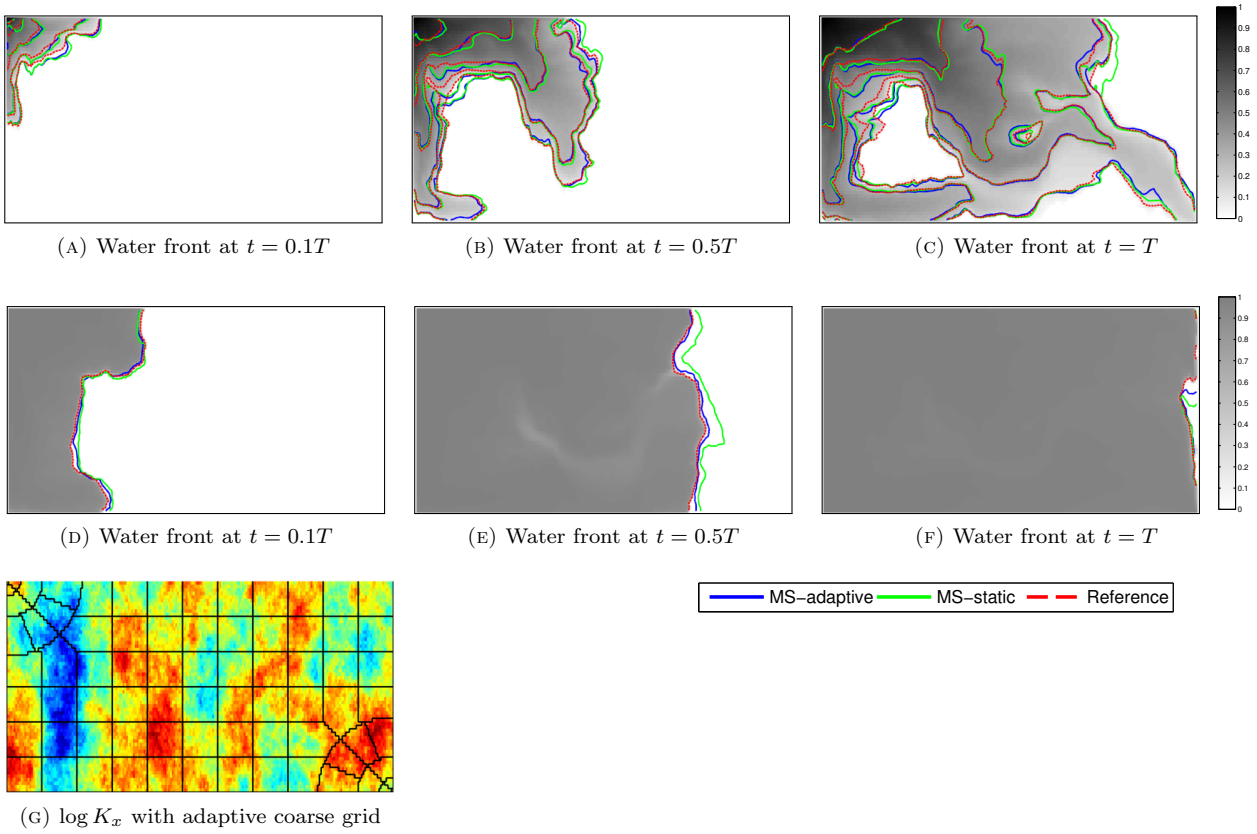
$$\lambda_w(S) = MS^2, \qquad \lambda_n(S) = (1-S)^2, \qquad 0 \le S \le 1. \tag{14}$$

In this case, the total mobility satisfies $\lambda(0) = 1$ and $\lambda(1) = M$, so that the end-point mobility ratio is $M$. Different end-point mobility ratios give rise to very different flow scenarios. With $M > 1$, we have an unstable displacement in which the injected fluid is more mobile than the resident fluid and will therefore tend to develop viscous fingers that penetrate rapidly through the less viscous resident fluid. For $M < 1$, on the other hand, the injected fluid is less viscous than the displaced fluid, giving a stable displacement characterized by a strong, piston-like displacement front. To correctly predict the speed of the displacement front, the multiscale method must accurately account for dynamic changes in the mobility. As noted by Kippe et al. [25], resolving stable displacements is more challenging than resolving unstable displacements, for which the changes in the mobility field will be smooth and relatively small because of the weak displacement front. For a stable displacement, the propagation of the strong displacement front will induce large and abrupt changes in the total mobility field, so that it may deviate significantly from the mobility field used to compute the basis functions that make up the multiscale prolongation operator.

To compare the two multiscale approaches, updating or not updating basis functions, we consider the top layer of the SPE 10 model initially filled with oil and produced by a quarter five-spot well pattern with water injected in one corner and fluid produced in the diagonally opposite corner. The two wells are controlled by pressure, which gives a more challenging test than rate-based injection because the inflow and outflow now depends directly on the pressure distribution. Figure 12 shows the saturation of the injected fluid for $M = 0.1$ and $M = 10$ at three different times in the interval $[0, T]$. To improve the prediction of pressure, we adapted the coarse grid by applying a radial refinement near the wells as shown in Figure 12g. For the weak displacement front there is good agreement between the multiscale and the reference solutions, even if basis functions are not updated throughout the simulation. For the sharp front, however, the multiscale method with static basis functions overestimates the speed of the leading shock. This error accumulates during the simulation and results in earlier breakthrough. A plot of the saturation error over time in Figure 13 shows that the error is larger with static basis functions in both cases, but the difference in errors is more pronounced for the piston-like displacement.

4.3.3. *High-contrast media.* Extensive numerical experiments have shown that contemporary multiscale methods provide approximate solutions of good quality for highly heterogeneous media. However, cases with large permeability contrasts are generally challenging and it is not difficult to construct pathological test cases on which a particular method fails to produce accurate solutions. Diagonal channel. Kippe et al. [25] proposed a simple and illuminating example consisting of a narrow high-permeable channel in a low-permeable background, where the channel is aligned with the diagonal direction of the grid as shown in Figure 14a. For the simulation, we use linear relative permeability, unit viscosity for both fluids, a single pressure step, and twenty time-steps in the implicit transport solver.

It is well known that high permeability contrasts along the edges of the dual grid can give poor localization for multiscale methods. If the channel intersects the faces of the primal coarse grid, the pressure extrapolation used to localize basis functions will start in a low-permeable region at a dual vertex, cross the high-permeable channel along the dual edge, before ending in a low-permeable region at the dual vertex on the opposite side of the channel. The corresponding basis function will overestimate flux between the high and low-permeable regions and cause a saturation front propagating to the high-permeable channel to leak out into the surrounding low-permeable background, as shown in Figure 14, or in the worst case, lead to non-monotone coarse-scale operators, e.g., as discussed in [44]. The same type of problem occurs for the MsMFE method, which relies on degrees

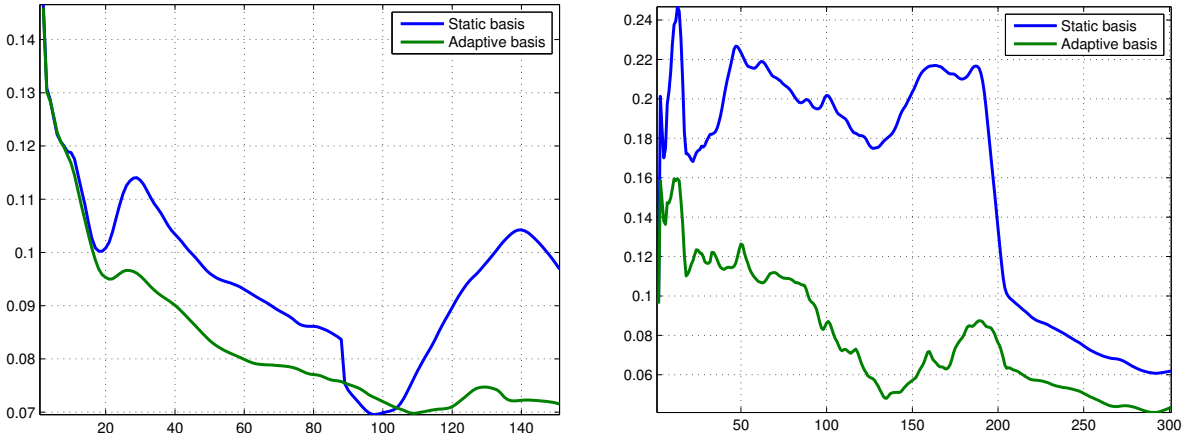



(A) Water front at $t = 0.1T$ (B) Water front at $t = 0.5T$ (C) Water front at $t = T$

(D) Water front at $t = 0.1T$ (E) Water front at $t = 0.5T$ (F) Water front at $t = T$

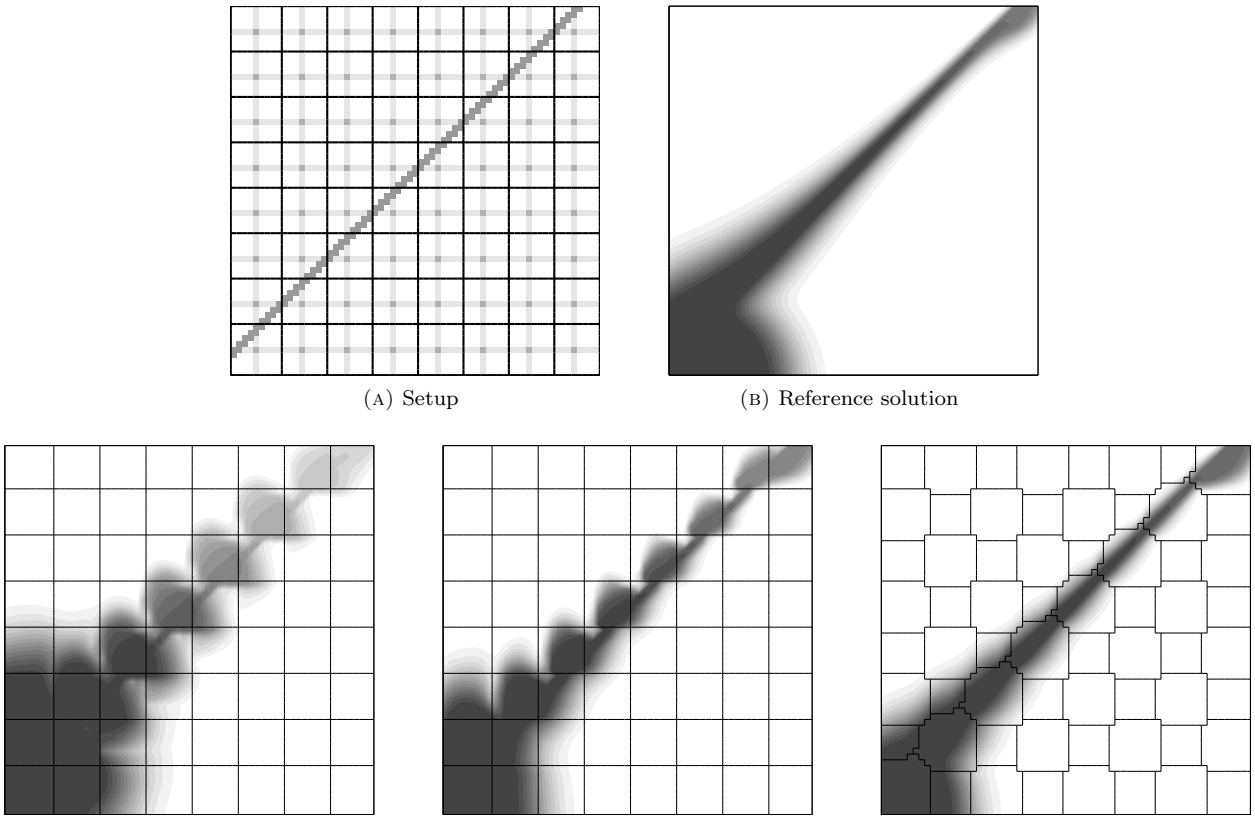
(G) $\log K_x$ with adaptive coarse grid

FIGURE 12. Saturation profiles for the two-dimensional waterflood example. The reference saturation field is shown in graytones, with contour lines of the two multiscale solutions superimposed. The first and second rows correspond to weak and piston-like displacement, respectively.




(A) Weak displacement front (B) Piston-like displacement

FIGURE 13. Saturation error measured in the relative $L^1$ norm (13) as function of time step number for water injection in the top layer of the SPE 10 model.



(A) Setup (B) Reference solution

(C) MsFV approximation (D) MsRSB approximation (E) MsRSB approximation

FIGURE 14. Pathological example of a high-contrast medium. In the setup, dark color indicates the channel having 100 times higher permeability than the background field, solid lines show the coarse blocks, and light colors show cells along the edges of a dual coarse grid. The next four plots show saturation profiles after the injection of 0.25 pore volumes of water from a well placed at the center of the lower-left coarse block.

of freedom associated with the faces of the coarse grid, if the channel is shifted slightly so that it intersects the vertices of the coarse blocks.

The problem can be mitigated by adapting the restriction operator to the local structures in the prolongation operator as discussed in Section 4.1; that is, by assigning cell $i$ to block $j$ if $P_{ij} \geq P_{ik}$ for all $k \neq j$. Figure 14e shows that the saturation profile is significantly improved by using MsRSB on the resulting grid. Results for MsFV are almost identical and are not reported. Because the adaptive procedure relies on comparing floating point numbers, the grid is also perturbed away from the channel. This numerical artifact has no effect on the accuracy for this particular case and could easily have been removed if we had used a slightly more sophisticated implementation.

Multiple channels. Our next example considers a slightly more complex case with a network of thin channels having one thousand times higher permeability than the surrounding rock. Figure 15 reports fine-scale solutions and multiscale approximations computed with the same computational setup as for the diagonal channel, except for changes to the permeability field. Adapting the restriction operator clearly improves the qualitative prediction of the flow patterns for both multiscale methods. However, the approximation errors are quite large for both methods and if the purpose is to accurately predict production profiles or the evolution of the saturation field, extra iterations

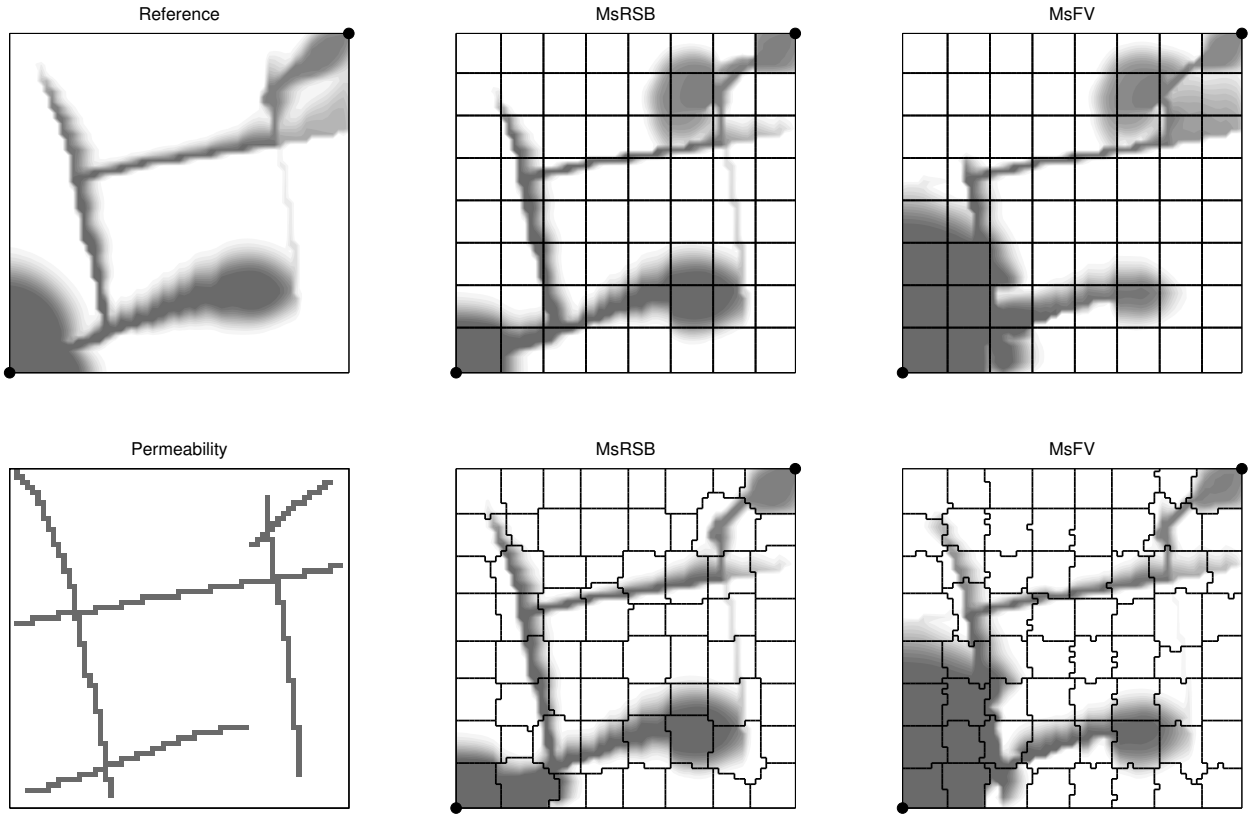



FIGURE 15. Flow in a network of thin channels having one thousand times higher permeability than the surrounding rock. The plots show saturation profiles computed by the MsRSB and MsFV method on a Cartesian coarse grid and a coarse grid adapted to the basis functions.

will be necessary. Plots of production profiles and saturation errors are not included for brevity, but show that MsRSB predicts well production much more accurately than MsFV and also gives much lower $L^1$ errors in the saturation fields. Interestingly, while adapting the restriction operator improves saturations errors significantly for MsRSB, the match in production profile becomes slightly worse. For MsFV, adapting the restriction operator improves both production profiles and the saturation field.

Our experience after having run a number of similar experiments of high-contrast media is that adapting the restriction operator is a simple precaution strategy to improve solution quality near high-permeable channels, at least when these channels are scattered relatively sparsely throughout the domain. Once exception is dense systems of narrow, high-permeable channels as found in fractured media, for which our experiments indicate that is better to describe the high-permeable fractures as lower-dimensional objects in the multiscale method, see [54]. This will be discussed in more detail in a forthcoming paper.

Multiple flow barriers. Previous experience with the MsMFE method has shown that barriers are best represented in a multiscale solution if one adapts the coarse blocks to follow the barriers, see [3, 33]. Figure 16 shows a case containing a network of narrow barriers, simulated with the same setup as in the two previous examples, except that we now inject a total of two volumes. Adapting the restriction operator has no effect on the solution quality up to water breakthrough and only a minor positive effect afterwards. If we instead adapt the prolongation operator by fitting the coarse grid to the low-permeable barriers, the accuracy is significantly improved at the expense of having

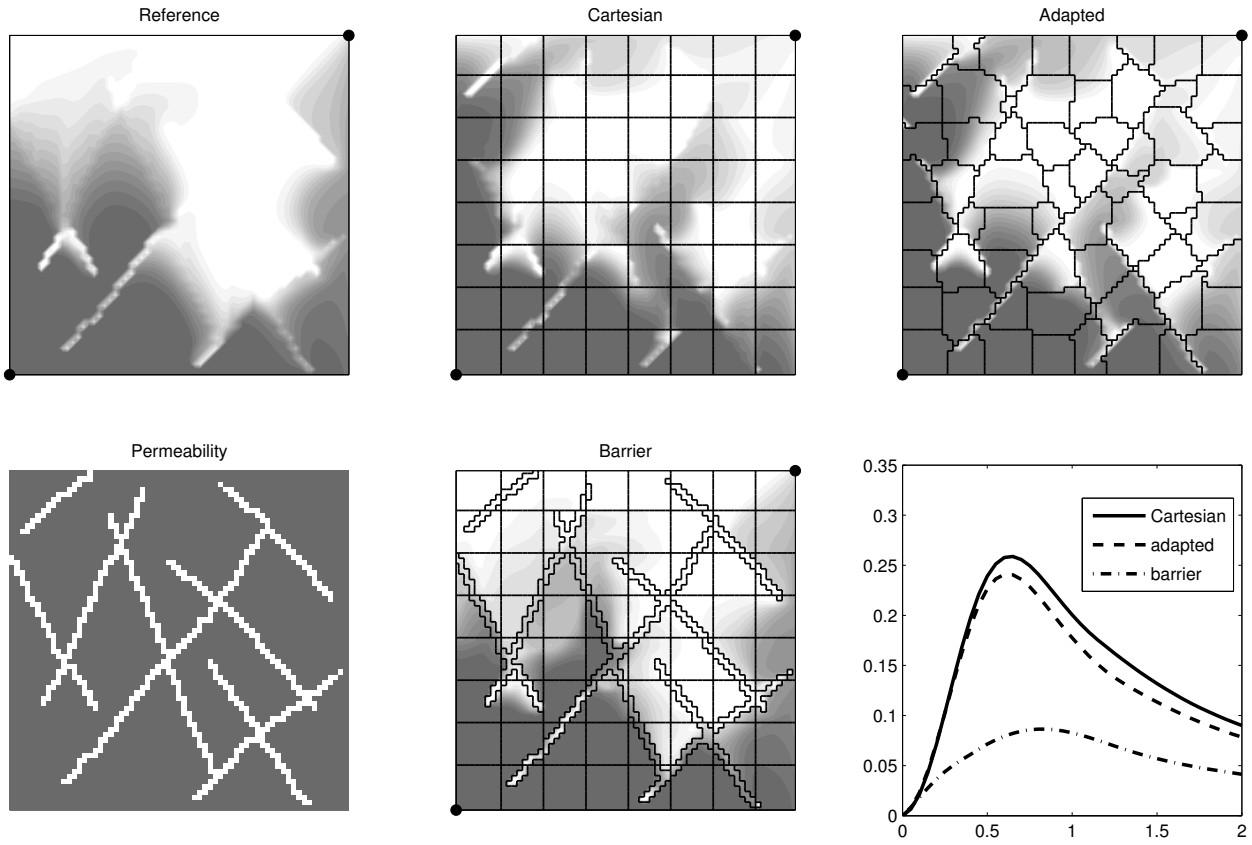
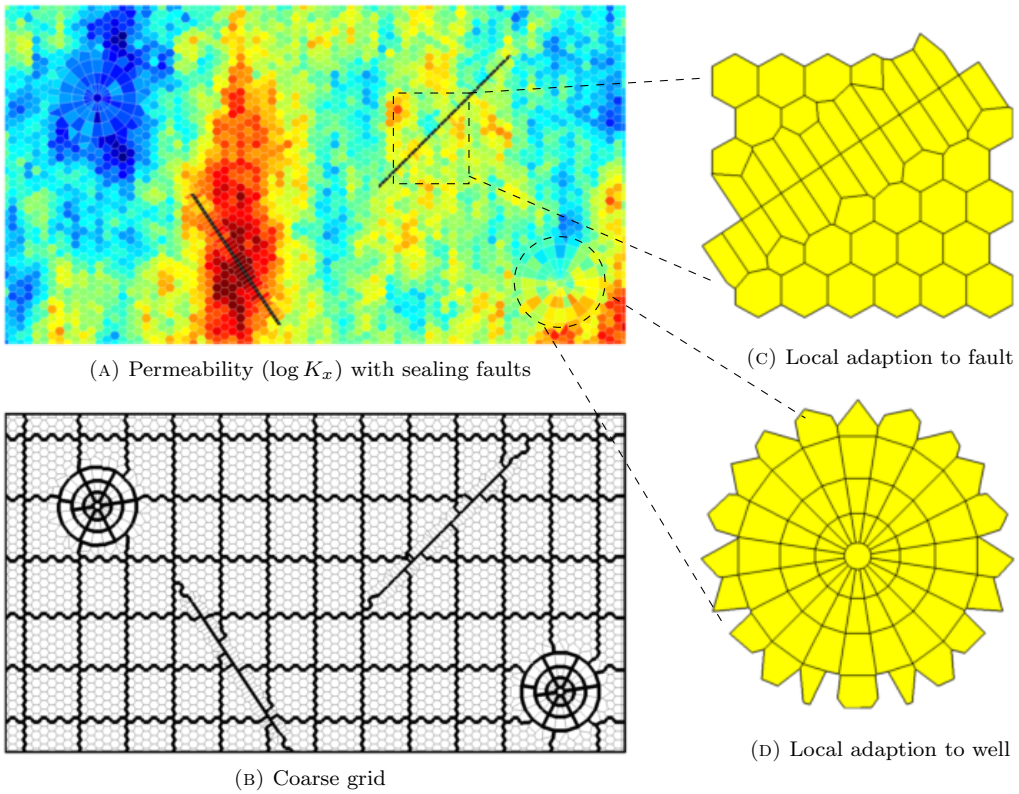



FIGURE 16. Flow in a network of thin flow barriers having one thousand times lower permeability than the surrounding rock. The plots show saturation profiles after 0.5 PVI computed by the MsRSB method on a Cartesian coarse grid, on the same grid but with restriction operator adapted to the prolongation operator, and on a coarse grid that adapts to the barriers.

to work with a much more complicated coarse partition. In principle, one could combine the two approaches and adapt the prolongation operator *a priori* and the restriction operator *a posteriori*. Our experience is that this is difficult in practice as the combination of the two methods tends to produce control volumes that are highly convoluted and hence may cause singular or near-singular coarse problems.

Although we have seen several barrier cases for which adapting the restriction operator gives better results than adapting the prolongation operator, it is our general experience that barriers are best treated by adapting the coarse partition that defines the prolongation operator. However, one should be careful to not create overly complex and irregular grids that may introduce numerical artifacts and instabilities that may impact the approximation properties of the otherwise relatively robust MsRSB prolongation operator.

4.3.4. *Unstructured grid adapted to faults and wells.* The Gullfaks example showed how the structured corner-point format easily can result in unstructured topologies near faults and eroded or inactive cells. However, fully unstructured grids can also be of interest in regions without complex features, for instance to avoid consistency issues with the standard two-point discretization. In this example, we consider a two-dimensional PEBI mesh as shown in Figure 17. The grid is adapted to local features, here exemplified by two sealing faults and two wells. The cells near the faults are



(A) Permeability ($\log K_x$) with sealing faults

(B) Coarse grid

(C) Local adaption to fault

(D) Local adaption to well

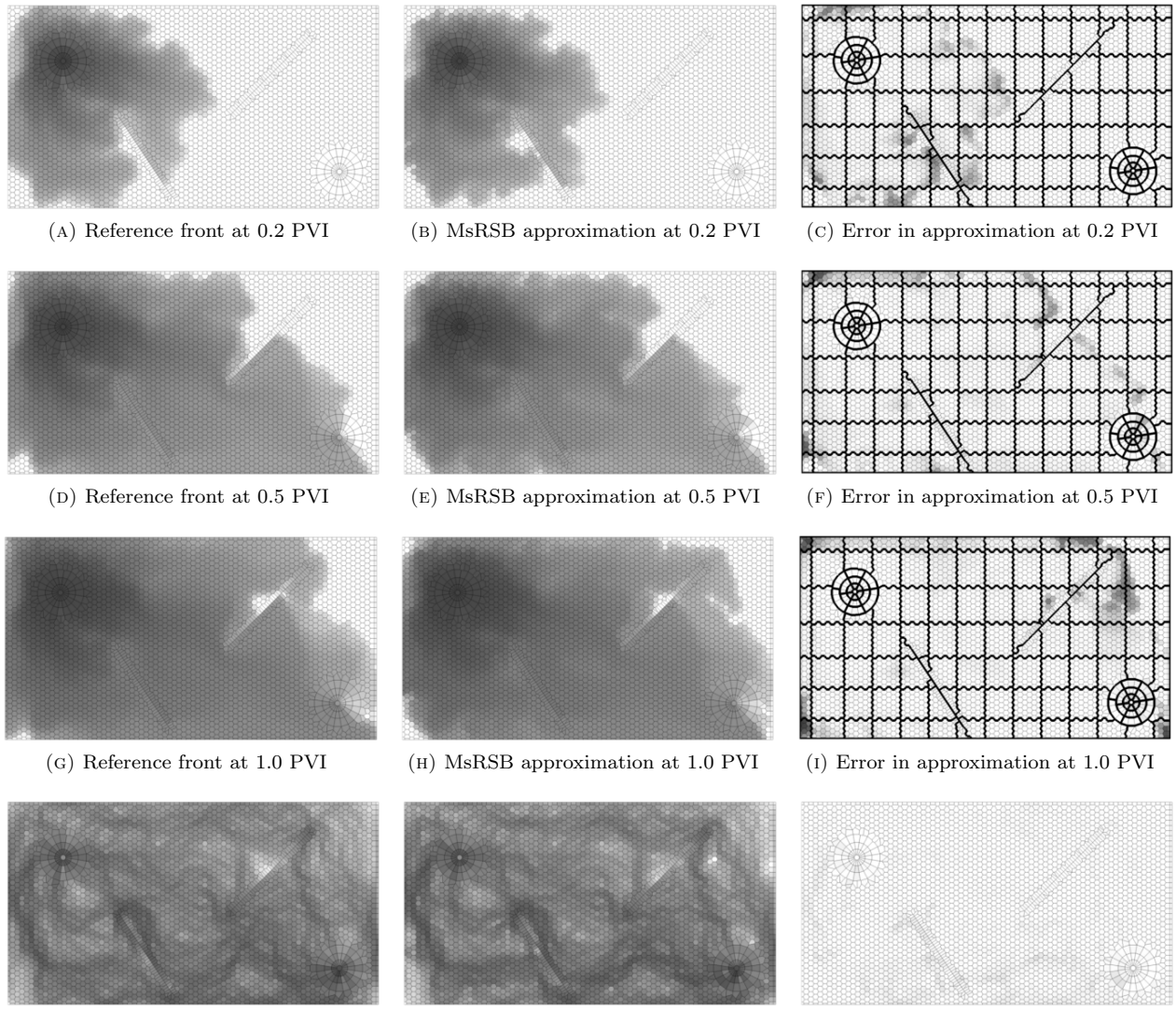
FIGURE 17. Unstructured test case: Fine-scale permeability field, adapted coarse grid, and zoom of local refinements in the fine grid.

oriented to that the cell faces follow the fault and near the well we have introduced a radial grid grid refinement to better approximate the radial flow in the near-well zone.

The fluid model is a simple incompressible fluid with Corey-type relative permeability curves with quadratic exponents. The reservoir is initially filled with oil having a viscosity of 5 centi poise. Water is injected from a well near the upper-left corner whereas fluids are produced at a constant rate from a well near the bottom-right corner, with no-flow conditions at the boundary. Petrophysical data are sampled from Layer 35 of the Tarbert formation from SPE 10 using nearest neighbor interpolation. The resulting grid has 3265 cells. We have selected a slightly smaller grid than the base case to make it easier to see the local refinements. This means that the petrophysical parameters may have somewhat stronger contrasts (more abrupt jumps) than specified in the original SPE 10 data set. The faults are considered to be completely sealing, i.e., the transmissibility over the interfaces is zero.

For the multiscale solver, we create a simple partition using the centroids of the cells. The coarse blocks are then split across faults and the coarse blocks in the radial subsgrids are partitioned using distance from the wellbore. Once all the local features are accounted for, the coarse grid itself is also unstructured and altogether this represents a challenging test case. The coarse grid has 146 blocks. Away from unstructured features the median number of cells in each block is 42.

The saturation profiles for the reference and multiscale simulations presented in Figure 18 show that the multiscale approximation captures the flow pattern accurately. Especially, the local fluid behavior near the faults is correctly represented; such local flow can easily be lost in a typical upscaled model.



(A) Reference front at 0.2 PVI (B) MsRSB approximation at 0.2 PVI (C) Error in approximation at 0.2 PVI

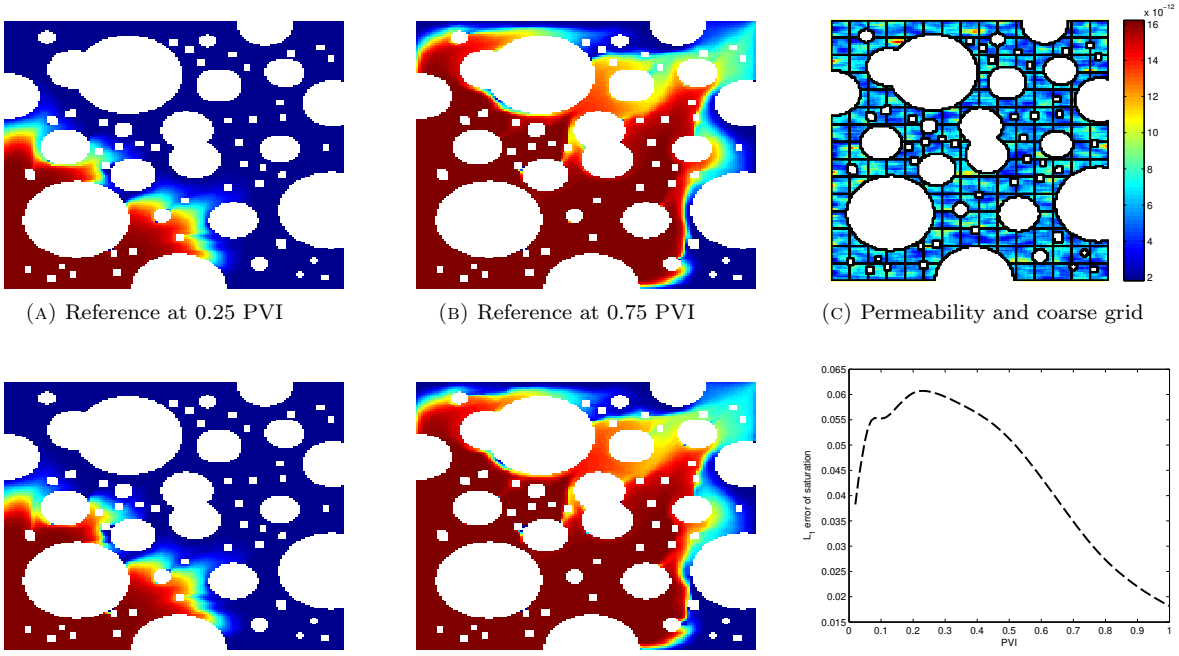
(D) Reference front at 0.5 PVI (E) MsRSB approximation at 0.5 PVI (F) Error in approximation at 0.5 PVI

(G) Reference front at 1.0 PVI (H) MsRSB approximation at 1.0 PVI (I) Error in approximation at 1.0 PVI

(J) $\|v\|$ at first timestep (K) $\|v\|$ at first timestep (MsRSB) (L) Error in $\|v\|$

FIGURE 18. The water saturation at different timesteps for the unstructured testcase with local refinement. The error colormap has a maximum of 0.5, which is higher than the largest error seen.

4.3.5. *Model with a large number of inactive cells.* For many models there will be a large number of cells with insignificant permeability and/or porosity. There are different ways to treat such cells. One approach is to simply let the cells retain their small values and include them in the full system. By including the cells, however, the computational effort will be just as large as if they were highly permeable even though nothing will flow through them. The other option is to remove cells with impermeable rock types, which is what we will consider here. Consider a $1 \times 1$ km domain comprised of $150 \times 150$ fine cells, with lognormal permeability as shown in Figure 19. We remove 43% of the cells, place a quarter five-spot well pattern with injection in the lower-left corner and proceed to inject one pore volume over ten years with unit mobility ratio. The multiscale solver uses a $15 \times 15$ uniform coarse grid and gives an accurate reproduction of the injected fluid, with a flux error of 0.15 in the $L_2$ norm.




(A) Reference at 0.25 PVI (B) Reference at 0.75 PVI (C) Permeability and coarse grid

(D) MsRSB at 0.25 PVI (E) MsRSB at 0.75 PVI (F) Saturation error in $L_1$ norm

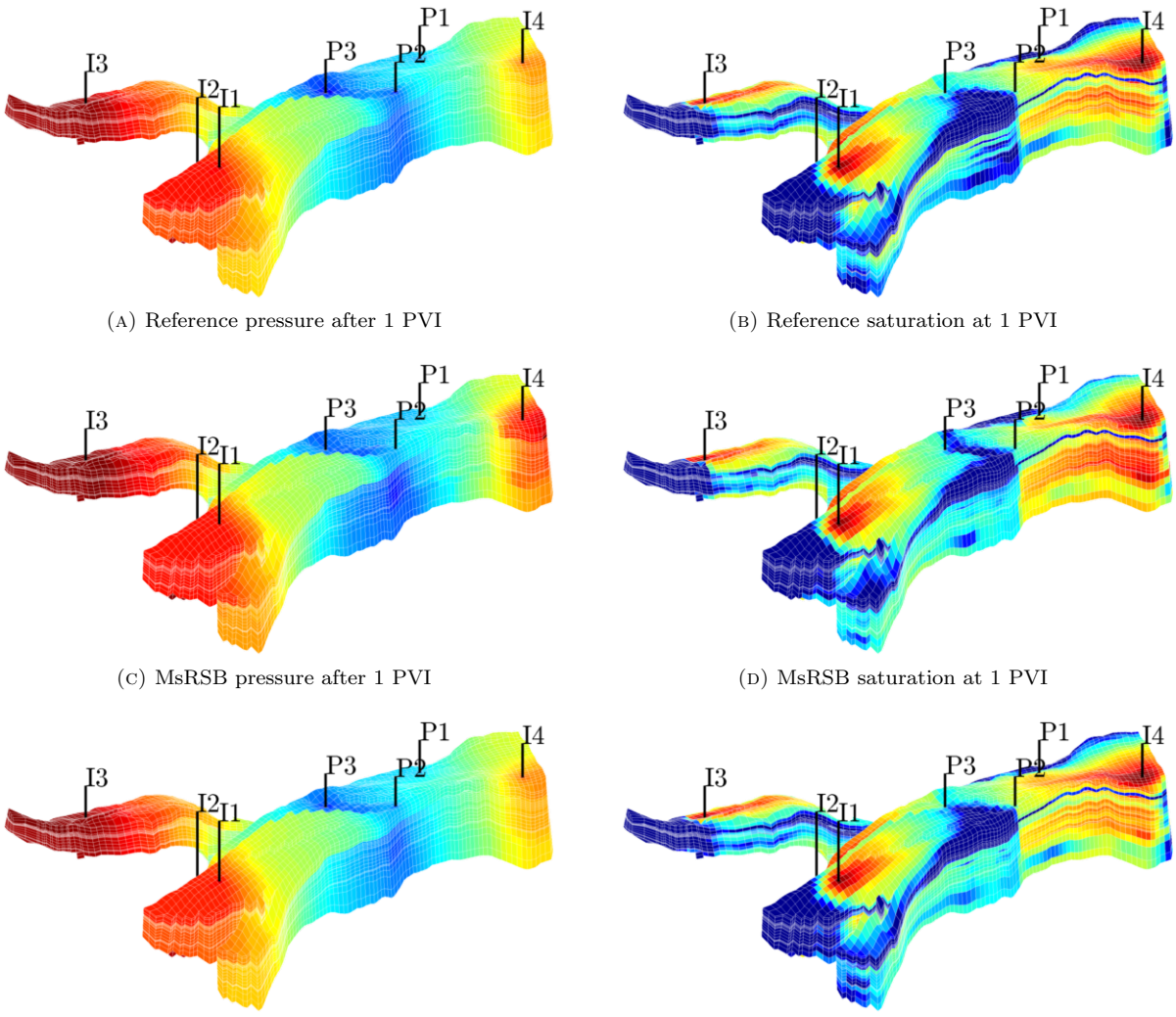
FIGURE 19. Saturation fronts for a model containing a large number of impermeable zones throughout the domain modelled as inactive cells.

4.3.6. *Norne field model.* In our final example, we study a water-injection problem posed on a geological model with grid geometry and petrophysical properties taken from the Norne benchmark case [48]. Initially, the reservoir is completely filled with oil, and the fluid behavior is described using an incompressible, two-phase model with the same fluid properties as in Section 4.3.4. The grid has 44 420 fine cells and layered permeability distribution. The model consists of two disconnected rock formations that are separated by full layer of inactive cells, so that the wells are the primary means of communication between the top three layers and the remaining parts of the model. Several faults and partially eroded layers makes the model effectively unstructured.

The reservoir has seven wells, three producers and four injectors. Each well is completed in all vertical layers, and will hence pass through multiple coarse blocks. The injectors operate at fixed pressure of 500 bar, while the producers have a constant rate corresponding to the drainage of a complete pore volume over the production period of 100 years.

The grid is partitioned into 250 coarse blocks using Metis with transmissibilities as edge weights. Wells are resolved most accurately in a multiscale method if they are placed near the centers of the coarse blocks. Making coarse grids that satisfy this property is straightforward if the grid has a simple structure and each well only perforate a single cell or a small contiguous set of cells. For complex grids with multiple perforations per well, however, it is generally not feasible to make such grids. We therefore consider two different set-ups for the multiscale solver: In the first, we use one pass of the multiscale solver to compute an approximate pressure. In the second, we use five multiscale cycles that each consist of a multiscale solve followed by five Jacobi iterations to remove local error and account for near-well information.

Figure 20 shows pressure and saturation profiles after the injection of one pore-volume of water, as computed by the fine-scale solver and the two multiscale solver. We note that the multiscale pressure approximation is in close agreement with the fine-scale solution, irregardless of whether we use iterations or not. For the saturation field, the multiscale solvers exhibit some smearing near the




(A) Reference pressure after 1 PVI

(B) Reference saturation at 1 PVI

(C) MsRSB pressure after 1 PVI

(D) MsRSB saturation at 1 PVI

(E) MsRSB with 5 cycles pressure after 1 PVI

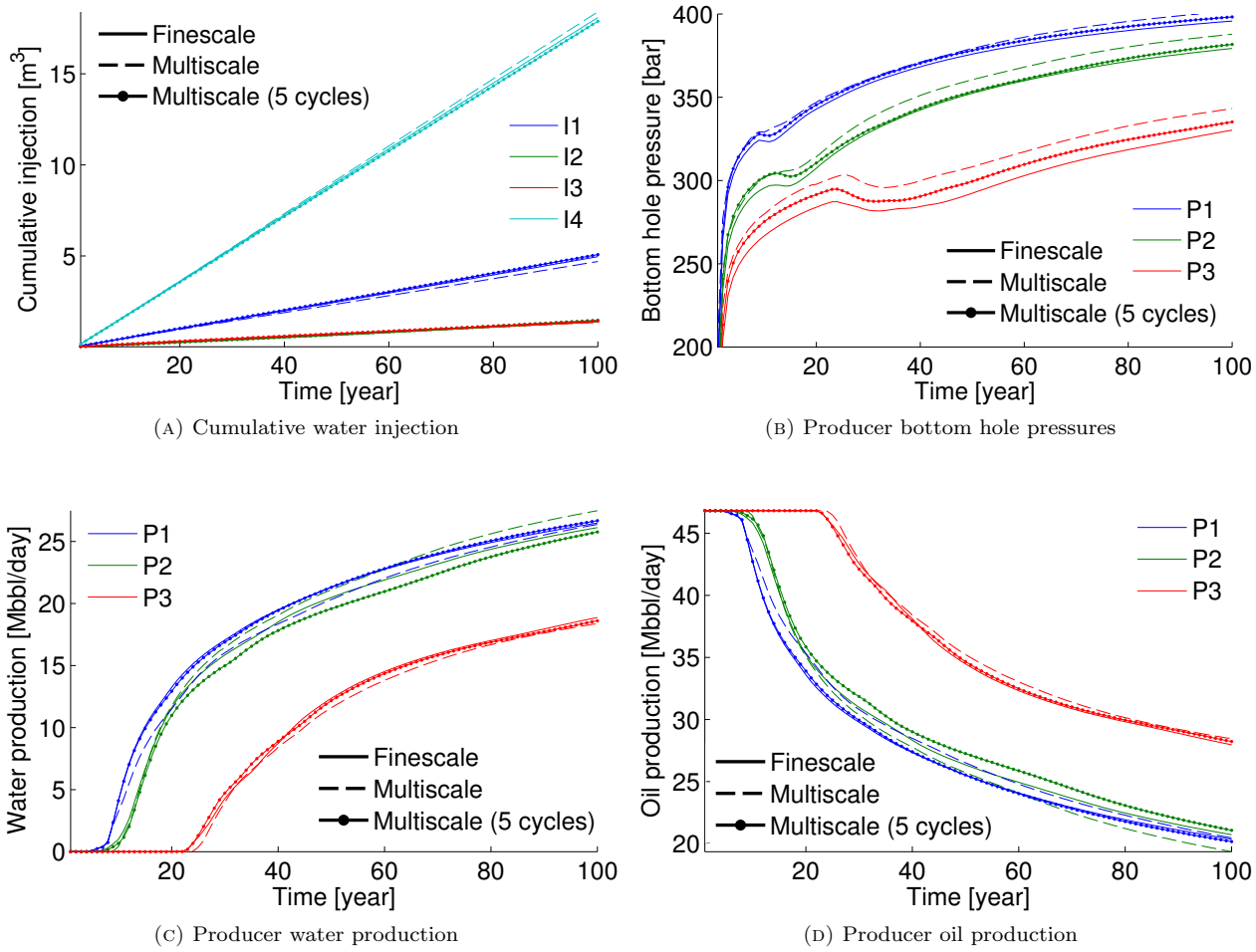
(F) MsRSB saturation with 5 cycles at 1 PVI

FIGURE 20. Pressure and saturation profiles after the injection of one pore-volume of water for the model derived from the Norne benchmark data. All plots of the same type use the same color axes.

boundary of the domain, but otherwise reproduce the water distribution at the end of simulation excellently. This is also confirmed by the well curves reported in Figure 21. Even without iterations, the multiscale solver is able to correctly reproduce the qualitative behavior of the producer, with a discrepancy of less than 5% in the bottom-hole pressure and fluid rates. By adding more iterations, this discrepancy can be reduced to the point where one also has an excellent quantitative match.

## 5. Concluding Remarks

We have presented a novel multiscale formulation that relies on an iterative process to construct basis functions. The iterative process is defined as standard matrix manipulations in combination with simple grid indicators. These indicators can easily be obtained from general coarse partitions without any stringent requirements on the underlying grid topology. Because the formulation only requires knowledge of cell centroids and topological neighborship, the method is applicable to unstructured polyhedral grids in general and stratigraphic grids in particular. The method is




(A) Cumulative water injection

(B) Producer bottom hole pressures

(C) Producer water production

(D) Producer oil production

FIGURE 21. Well curves for the water-injection test case based on the Norne benchmark data.

flexible and very robust with respect to the shape of the coarse blocks, and it is therefore easy to formulate automated strategies to coarsen complex grid models having cells with degenerate geometry and unstructured topology caused by erosion, faults, inactive cells, local grid refinement, and other types of non-neighboring connections. In particular, the coarse partition can be adapted to geological features or to provide extra resolution, e.g., in near-well regions. This can be done in two ways, either by fitting the coarse grid *a priori* to adapt the prolongation operator or by modifying the control volumes *a posteriori* to adapt the restriction operator.

Extensive numerical tests on single-phase problems, some of which are reported herein and in [41], show that the MsRSB method compares favorably with the classical MsFV in terms of accuracy, and is generally much more robust and less affected by the choice of restriction operator and the resolution and type of the coarse partition. Notably, the pressure field for the challenging SPE 10 benchmark is solved to within 5% accuracy without the need for additional iterations or coarse grid adaption. Likewise, a high degree of accuracy is observed on models of the Norne field and the very structurally complex Gullfaks field. For both models, a wide range of coarse partitions can easily be generated using standard graph partitioning algorithms. This indicates that automated coarsening is indeed possible for realistic models with high media contrasts, complex unstructured topology, and cells with very high aspect ratios and degenerate geometries.

Through a set of multiphase test cases we have also demonstrated that the method can easily adapt to changing mobilities and provide prediction of well curves that are qualitatively correct, also when wells crossing different geological layers are completed in multiple coarse blocks. By setting tighter tolerances on the fine-scale residuals, excellent quantitative match can also be obtained at the cost of a few additional iterations. In many aspects, the iterative version of the method resembles an agglomeration-based multigrid method, but unlike these methods, MsRSB can be stopped at any prescribed residual tolerance and still produce a mass-conservative approximation.

## 6. Acknowledgments

This research was funded in part by Schlumberger Information Solutions and by the Research Council of Norway under grant no. 226035. We thank Jostein R. Natvig and Halvor Møll Nilsen for constructive discussions. We also thank Statoil (operator of the Norne field) and its license partners ENI and Petoro for the release of the Norne data and acknowledge the Center for Integrated Operations at NTNU for cooperation and coordination of the Norne cases.

## References

[1] J. Aarnes and K.-A. Lie. Toward reservoir simulation on geological grid models. In *Proceedings of the 9th European Conference on the Mathematics of Oil Recovery*. EAGE, Cannes, France, 2004.

[2] J. E. Aarnes. 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, 2004. ISSN 1540-3459. doi:10.1137/030600655.

[3] J. E. Aarnes, S. Krogstad, and K.-A. Lie. A hierarchical multiscale method for two-phase flow based upon mixed finite elements and nonuniform coarse grids. *Multiscale Model. Simul.*, 5(2):337–363, 2006. ISSN 1540-3459. doi:10.1137/050634566.

[4] J. E. Aarnes, S. Krogstad, and K.-A. Lie. Multiscale mixed/mimetic methods on corner-point grids. *Comput. Geosci.*, 12(3):297–315, 2008. ISSN 1420-0597. doi:10.1007/s10596-007-9072-8.

[5] F. O. Alpak, M. Pal, and K.-A. Lie. A multiscale method for modeling flow in stratigraphically complex reservoirs. *SPE J.*, 17(4):1056–1070, 2012. doi:10.2118/140403-PA.

[6] T. Arbogast. Implementation of a locally conservative numerical subgrid upscaling scheme for two-phase Darcy flow. *Comput. Geosci.*, 6(3-4):453–481, 2002. ISSN 1420-0597. doi:10.1023/A:1021295215383.

[7] T. Arbogast and K. J. Boyd. Subgrid upscaling and mixed multiscale finite elements. *SIAM J. Numer. Anal.*, 44(3):1150–1171, 2004. ISSN 0036-1429.

[8] T. Arbogast, G. Pencheva, M. F. Wheeler, and I. Yotov. A multiscale mortar mixed finite element method. *Multiscale Model. Simul.*, 6(1):319–346, 2007. doi:10.1137/060662587.

[9] I. Babuka, G. Caloz, and J. Osborn. Special finite element methods for a class of second order elliptic problems with rough coefficients. *SIAM J. Numer. Anal.*, 31(4):945–981, 1994. doi:10.1137/0731051.

[10] S. Balay, S. Abhyankar, M. F. Adams, J. Brown, P. Brune, K. Buschelman, L. Dalcin, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, K. Rupp, B. F. Smith, S. Zampini, and H. Zhang. PETSc users manual. Technical Report ANL-95/11 - Revision 3.6, Argonne National Laboratory, 2015. URL `http://www.mcs.anl.gov/petsc`.

[11] Y. Brenier and J. Jaffré. Upstream differencing for multiphase flow in reservoir simulation. *SIAM J. Numer. Anal.*, 28(3):685–696, 1991. doi:10.1137/0728036.

[12] M. Brezina, R. Falgout, S. MacLachlan, T. Manteuffel, S. McCormick, and J. Ruge. Adaptive smoothed aggregation ($\alpha$ sa) multigrid. *SIAM Review*, 47(2):317–346, 2005.

[13] Z. Chen and T. Y. Hou. A mixed multiscale finite element method for elliptic problems with oscillating coefficients. *Math. Comp.*, 72:541–576, 2003. doi:10.1090/S0025-5718-02-01441-2.

[14] M. A. Christie and M. J. Blunt. Tenth SPE comparative solution project: A comparison of upscaling techniques. *SPE Reservoir Eval. Eng.*, 4:308–317, 2001. doi:10.2118/72469-PA. Url: `http://www.spe.org/csp/`.

[15] Y. Efendiev and T. Y. Hou. *Multiscale Finite Element Methods*, volume 4 of *Surveys and Tutorials in the Applied Mathematical Sciences*. Springer Verlag, New York, 2009.

[16] H. Fossen and J. . Hesthammer. Structural geology of the Gullfaks Field. In M. P. Coward, H. Johnson, and T. S. Daltaban, editors, *Structural geology in reservoir characterization*, volume 127, pages 231–261. Geological Society Special Publication, 1998.

[17] M. Gee, C. Siefert, J. Hu, R. Tuminaro, and M. Sala. ML 5.0 smoothed aggregation user's guide. Technical Report SAND2006-2649, Sandia National Laboratories, 2006.

[18] H. Hajibeygi and P. Jenny. Multiscale finite-volume method for parabolic problems arising from compressible multiphase flow in porous media. *J. Comput. Phys*, 228(14):5129–5147, 2009. doi:10.1016/j.jcp.2009.04.017.

[19] H. Hajibeygi and H. A. Tchelepi. Compositional multiscale finite-volume formulation. *SPE J.*, 19(2): 316–326, 2014. doi:10.2118/163664-PA.

[20] M. A. Hesse, B. T. Mallison, and H. A. Tchelepi. Compact multiscale finite volume method for heterogeneous anisotropic elliptic equations. *Multiscale Model. Simul.*, 7(2):934–962, 2008. doi:10.1137/070705015.

[21] T. Hou and X.-H. Wu. A multiscale finite element method for elliptic problems in composite materials and porous media. *J. Comput. Phys.*, 134:169–189, 1997. doi:10.1006/jcph.1997.5682.

[22] P. Jenny, S. H. Lee, and H. A. Tchelepi. Multi-scale finite-volume method for elliptic problems in subsurface flow simulation. *J. Comput. Phys.*, 187:47–67, 2003. doi:10.1016/S0021-9991(03)00075-5.

[23] G. Karypis and V. Kumar. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM J. Sci. Comp.*, 20(1):359–392, 1998. doi:10.1137/S1064827595287997.

[24] V. Kippe, J. E. Aarnes, and K.-A. Lie. Multiscale finite-element methods for elliptic problems in porous media flow. In P. Binning, P. Engesgaard, H. Dahle, G. Pinder, and W. Gray, editors, *Proceedings of the XVI International Conference on Computational Methods in Water Resources*, Copenhagen, Denmark, 18–22 June 2006. URL `http://proceedings.cmwr-xvi.org/`.

[25] V. Kippe, J. E. Aarnes, and K.-A. Lie. A comparison of multiscale methods for elliptic problems in porous media flow. *Comput. Geosci.*, 12(3):377–398, 2008. ISSN 1420-0597. doi:10.1007/s10596-007-9074-6.

[26] A. Kozlova, Z. Li, J. R. Natvig, S. Watanabe, Y. Zhou, K. Bratvedt, and S. H. Lee. A real-field multiscale black-oil reservoir simulator. In *SPE Reservoir Simulation Symposium, 23-25 February, Houston, Texas, USA*, 2015. doi:10.2118/173226-MS. SPE 173226-MS.

[27] S. Krogstad, K.-A. Lie, H. M. Nilsen, J. R. Natvig, B. Skaflestad, and J. E. Aarnes. A multiscale mixed finite-element solver for three-phase black-oil flow. In *SPE Reservoir Simulation Symposium, The Woodlands, TX, USA, 2–4 February 2009*, 2009. doi:10.2118/118993-MS.

[28] S. Krogstad, K.-A. Lie, and B. Skaflestad. Mixed multiscale methods for compressible flow. In *Proceedings of ECMOR XIII–13th European Conference on the Mathematics of Oil Recovery*, Biarritz, France, 10–13 September 2012. EAGE. doi:10.3997/2214-4609.20143240.

[29] S. Krogstad, K.-A. Lie, O. Møyner, H. M. Nilsen, X. Raynaud, and B. Skaflestad. MRST-AD – an open-source framework for rapid prototyping and evaluation of reservoir simulation problems. In *SPE Reservoir Simulation Symposium, 23–25 February, Houston, Texas*, 2015. doi:10.2118/173317-MS.

[30] S. H. Lee, C. Wolfsteiner, and H. Tchelepi. 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, 2008. doi:10.1007/s10596-007-9069-3.

[31] K.-A. Lie. *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`, 1st edition, May 2014.

[32] K.-A. Lie, S. Krogstad, I. S. Ligaarden, J. R. Natvig, H. Nilsen, and B. Skaflestad. Open-source MATLAB implementation of consistent discretisations on complex grids. *Comput. Geosci.*, 16:297–322, 2012. ISSN 1420-0597. doi:10.1007/s10596-011-9244-4. URL `http://dx.doi.org/10.1007/s10596-011-9244-4`.

[33] K.-A. Lie, J. R. Natvig, S. Krogstad, Y. Yang, and X.-H. Wu. Grid adaptation for the Dirichlet–Neumann representation method and the multiscale mixed finite-element method. *Comput. Geosci.*, 18 (3):357–372, 2014. doi:10.1007/s10596-013-9397-4.

[34] K. Lipnikov, J. D. Moulton, and D. Svyatskiy. A multilevel multiscale mimetic ($m^3$) method for two-phase flows in porous media. *J. Comput. Phys.*, 227(14):6727–6753, 2008. doi:10.1016/j.jcp.2008.03.029.

[35] I. Lunati and P. Jenny. Multiscale finite-volume method for compressible multiphase flow in porous media. *J. Comput. Phys.*, 216(2):616–636, 2006. doi:10.1016/j.jcp.2006.01.001.

[36] I. Lunati and P. Jenny. Treating highly anisotropic subsurface flow with the multiscale finite-volume method. *Multiscale Model. Simul.*, 6(1):308–318, 2007. doi:10.1137/050638928.

[37] I. Lunati and P. Jenny. Multiscale finite-volume method for density-driven flow in porous media. *Comput. Geosci.*, 12(3):337–350, 2008. doi:10.1007/s10596-007-9071-9.

[38] I. Lunati and S. H. Lee. An operator formulation of the multiscale finite-volume method with correction function. *Multiscale Model. Simul.*, 8(1):96–109, 2009. doi:10.1137/080742117.

[39] I. Lunati, M. Tyagi, and S. H. Lee. An iterative multiscale finite volume algorithm converging to the exact solution. *J. Comput. Phys.*, 230(5):1849–1864, 2011. doi:10.1016/j.jcp.2010.11.036.

[40] O. Møyner. Multiscale finite-volume methods on unstructured grids. Master's thesis, Norwegian University of Science and Technology, Trondheim, 2012. URL `http://daim.idi.ntnu.no/masteroppgave?id=7377`.

[41] O. Møyner. Construction of multiscale preconditioners on stratigraphic grids. In *ECMOR XIV – $14^{th}$ European Conference on the Mathematics of Oil Recovery, Catania, Sicily, Italy, 8-11 September 2014*. EAGE, 2014. doi:10.3997/2214-4609.20141775.

[42] O. Møyner and K.-A. Lie. The multiscale finite volume method on unstructured grids. In *SPE Reservoir Simulation Symposium, The Woodlands, TX, USA, 18–20 February 2013*, 2013. doi:10.2118/163649-MS. SPE 163649-MS.

[43] O. Møyner and K.-A. Lie. A multiscale two-point flux-approximation method. *J. Comput. Phys.*, 275: 273–293, 2014. doi:10.1016/j.jcp.2014.07.003.

[44] O. Møyner and K.-A. Lie. The multiscale finite-volume method on stratigraphic grids. *SPE J.*, 19(5): 816–831, 2014. doi:10.2118/163649-PA.

[45] O. Møyner and K.-A. Lie. A multiscale restriction-smoothed basis method for compressible black-oil models. *SPE J.*, 2015. submitted.

[46] MRST. The MATLAB Reservoir Simulation Toolbox, version 2014b, Nov. 2014. `http://www.sintef.no/MRST/`.

[47] J. R. Natvig, B. Skaflestad, F. Bratvedt, K. Bratvedt, K.-A. Lie, V. Laptev, and S. K. Khataniar. Multiscale mimetic solvers for efficient streamline simulation of fractured reservoirs. *SPE J.*, 16(4): 880–888, 2011. doi:10.2018/119132-PA.

[48] Norne. IO Center – Norne Benchmark Case, June 2012. URL `http://www.ipt.ntnu.no/~norne/wiki/doku.php`.

[49] M. Pal, S. Lamine, K.-A. Lie, and S. Krogstad. Multiscale method for simulating two and three-phase flow in porous media. In *SPE Reservoir Simulation Symposium, The Woodlands, Texas, USA, 18-20 February 2013*, 2013. doi:10.2118/163669-MS. SPE 163669-MS.

[50] E. Parramore, M. G. Edwards, S. Lamine, and M. Pal. Multiscale formulations with CVD-MPFA schemes on structured and unstructured grids. In *SPE Reservoir Simulation Symposium, The Woodlands, TX, USA, 18–20 February 2013*, 2013. doi:10.2118/163626-MS. SPE 163626-MS.

[51] T. H. Sandve, I. Berre, E. Keilegavlen, and J. M. Nordbotten. Multiscale simulation of flow and heat transport in fractured geothermal reservoirs: inexact solvers and improved transport upscaling. In *Thirty-Eighth Workshop on Geothermal Reservoir Engineering Stanford University, February 11-13*, Stanford, California, USA, 2013.

[52] A. Sandvin, E. Keilegavlen, and J. M. Nordbotten. Auxiliary variables for 3d multiscale simulations in heterogeneous porous media. *J. Comput. Phys*, 238:141–153, 2013. doi:10.1016/j.jcp.2012.12.016.

[53] K. Stüben. A review of algebraic multigrid. *Journal of Computational and Applied Mathematics*, 128 (1):281–309, 2001.

[54] M. Tene, M. S. Al Kobaisi, and H. Hajibeygi. Algebraic multiscale solver for flow in heterogeneous fractured porous media. In *SPE Reservoir Simulation Symposium held in Houston, Texas, USA, 23–25 February 2015*, 2015. doi:10.2118/173200-MS. SPE 173200-MS.

[55] P. Vanek, J. Mandel, and M. Brezina. Algebraic multigrid on unstructured meshes. Technical Report 34, University of Colorado at Denver, Denver, CO, USA, 1994.

[56] P. Vanek, J. Mandel, and M. Brezina. Algebraic multigrid by smoothed aggregation for second and fourth order elliptic problems. *Computing*, 56(3):179–196, 1996. doi:10.1007/BF02238511.

[57] P. Vanek, M. Brezina, and J. Mandel. Convergence of algebraic multigrid based on smoothed aggregation. *Numerische Mathematik*, 88(3):559–579, 2001. doi:10.1007/s211-001-8015-y.

[58] Y. Wang, H. Hajibeygi, and H. A. Tchelepi. Algebraic multiscale solver for flow in heterogeneous porous media. *J. Comput. Phys.*, 259:284–303, 2014. doi:10.1016/j.jcp.2013.11.024.

[59] H. Zhou and H. A. Tchelepi. Operator-based multiscale method for compressible flow. *SPE J.*, 13(2): 267–273, 2008. doi:10.2118/106254-PA.

[60] H. Zhou and H. A. Tchelepi. Two-stage algebraic multiscale linear solver for highly heterogeneous reservoir models. *SPE J.*, 17(2):523–539, 2012. doi:10.2118/141473-PA.