

A Multiscale Two-Point Flux-Approximation Method

Olav Møyner

Knut–Andreas Lie

October 25, 2013

A large number of multiscale finite-volume methods have been developed over the past decade to compute conservative approximations to multiphase flow problems in heterogeneous porous media. In particular, several iterative and algebraic multiscale frameworks that seek to reduce the fine-scale residual towards machine precision have been presented. Common for all such methods is that they rely on a dual-primal coarse partition, which makes it challenging to extend them to stratigraphic and unstructured grids. Herein, we propose a general idea for how one can formulate multiscale finite-volume methods using only a primal coarse partition. To this end, we use two key ingredients that are computed numerically: (i) elementary functions that correspond to flow solutions used in transmissibility upscaling, and (ii) partition-of-unity functions used to combine elementary functions into basis functions. We exemplify the idea by deriving a multiscale two-point flux-approximation (MsTPFA) method, which is robust with regards to strong heterogeneities in the permeability field and can easily handle general grids with unstructured fine- and coarse-scale connections. The method can easily be adapted to arbitrary levels of coarsening, and can be used both as a standalone solver and as a preconditioner. Several numerical experiments are presented to demonstrate that the MsTPFA method can be used to solve elliptic pressure problems on a wide variety of geological models in a robust and efficient manner.

1 Introduction

The oil and gas industry has for decades been a big user of computing and data processing to plan cost effective exploration and production of petroleum resources. Advanced desktop workflow tools have removed many bottlenecks that previously existed between different application domains. However, runtime is still often a limitation on the use of reservoir simulation, despite the fact that the available computational power has been greatly increased in the last decades. In part, this is caused by a continued increase in the complexity of reservoir models, and in part by modern workflows such as model calibration, uncertainty quantification, and optimization of production and recovery, which all require fast, scalable, and accurate flow simulations, possibly for large ensembles of model realizations.

Multiscale simulation technology promises to improve runtime by at least an order of magnitude compared to current simulator performance for traditional reservoir engineering workflows and offers a systematic framework for increasing local model resolution. This makes multiscale simulation a better tool for characterizing volumetric sweep and locating bypassed and immobile oil compared with traditional upscaling approaches which always imply a loss of information when homogenizing fine-scale structures. While there exist a wide variety of multiscale methods tailored to an even wider range of problems, the core idea of most multiscale methods

developed for reservoir simulation is an attempt to separate local and global phenomena, giving fine-scale solutions based on coarse-scale problems with precomputed basis functions that account for localized features [7]. Multiscale formulations also provide a framework that enables variable-fidelity simulation that within orders of magnitude shorter simulation times can provide qualitatively correct predictions of flow patterns. Utilizing this, often less emphasized, aspect of multiscale methods may lead to new and innovative simulation-based workflows for building earth models, which cannot be achieved with current state-of-the-art reservoir simulators.

Although modeling approaches used by the industry today are predominantly structured, they still lead to irregular and unstructured simulation grids. Very complex grids having unstructured connections and degenerate cell geometries arise naturally when representing structural framework like faults, joints, and deformation bands, and/or stratigraphic architectural characteristics like channels, lobes, clinoforms, and shale shale/mud drapes. Similarly, unstructured connections are induced when local grid refinement, structured or unstructured, is used to improve the modeling of (deviated) wells. Providing multiscale simulation capabilities for general grids with unstructured connections may accelerate the general adoption of unstructured and refined grids as a means of improved representation of deviated well paths and complex geological features. Likewise, greater flexibility with respect to coarse grids is needed to develop automated coarsening strategies that perform well for complex models. If possible, coarse partitions should adapt to complex features, such as wells, barriers and channels in a way that ensures optimal accuracy for a chosen level of coarsening.

This paper will focus on multiscale methods for computing pressure and fluxes on models that are realistic with the respect to geometrical description and petrophysical heterogeneity. Our main goal is to develop multiscale methods that are more accurate and simpler to use than state-of-the-art upscaling methods across a wide range of upscaling factors and at least an order-of-magnitude faster than conventional methods when used as a fine-grid solver. To provide value for commercial applications, the methods should work with industry-standard grid formats, be easy to implement within next-generation simulators, and generally be accurate, efficient, intuitive to use, and robust with respect to flow models and parameter choices. A lot of the necessary technological components have been developed over the last decade. The current industry-relevant state-of-the-art mainly exists as complementary technologies that are developed from two different mathematical principles: the multiscale finite-volume (MsFV) method [13] and the multiscale mixed finite-element (MsMFE) method [5, 1].

The original MsFV method [13] was developed for solving flow problems with many scales and a predominantly elliptic nature, and used basis functions computed on a dual grid to define transmissibilities in a multi-point coarse-scale discretization combined with another set of flow problems on the primal grid to reconstruct conservative fine-scale fluxes. The method has later been extended with a large body of research, including adaptivity [19], correction functions [22, 24, 37], iterative variants for error control [9, 11], and operator formulations suitable for pre-existing solvers [25, 26, 37], and has been applied to a wide range of physical problems, including compressible black-oil models [18] and compositional flow [10], all on Cartesian grids. Extending the MsFV method to realistic grids is generally encumbered by the requirement of a dual coarse partition. We have previously shown that it is possible to generate compatible primal-dual partitions for grids with realistic features like pinch-outs, erosions, faults and unstructured connections using a geometrical algorithm with topological post-processing [29, 28]. However, the coarsening process is difficult to automate in a robust manner and will sometimes give partitions that drastically reduce solution quality. In addition, it is well known that highly contrasted media and large anisotropy ratios pose problems for the reduced boundary condition used to construct the MsFV basis functions [23, 14, 12, 36].

The ability to handle fully unstructured grids with almost arbitrary connected coarse-scale partitions is one of the main advantages of the MsMFE method. Whereas the basis functions in the MsFV method are computed using a dual-grid formulation with unitary pressure values

at each vertex of the coarse blocks, the basis functions in the MsMFE method are constructed by prescribing a unit flow across faces in the coarse grid, which ensures that the latter is more robust with respect to the size and shape of the coarse blocks when applied to stratigraphic and unstructured grids [2, 3]. Pathological cases will of course also exist, but the accuracy of the MsMFE method can often be significantly improved by introducing a certain degree of adaption to local structures, particularly for high-contrast media [32, 4]. For two-phase flow, the method has been shown to provide good accuracy on industry-standard geological models [34]. On the other hand, although iterative versions of the MsMFE method have been applied to compressible flow [17, 15] and black-oil models [16, 20], a full extension of the method to realistic flow physics is generally encumbered by the need for a robust splitting of flow and transport, in which the flow equations are written on mixed (hybrid) form.

In this paper, we present what we believe is a general approach that combines the best features of the two methods discussed above: multiscale finite-volume methods that use numerically constructed partition-of-unity functions to glue together elementary flow solutions associated with interfaces between coarse blocks. The partition-of-unity functions are crucial and will enable us to dispense with the dual grid and provide the flexibility necessary to handle complex industry-standard grids with high aspect ratios and unstructured connections without significant impact to the solution quality. Likewise, using a finite-volume formulation enables the methods to efficiently resolve realistic flow physics, including capillarity, compressibility, and gravity, and, in particular, be applicable to industry-standard black-oil type models. The construction is fully algebraic, which means that existing multiscale techniques such as local stages and iterative cycles, followed by conservative fine-scale flux reconstructions [36], can be used without modification. Finally, using elementary flow solutions associated with interfaces instead of vertices means that the new multiscale methods will closely resemble traditional methods for transmissibility upscaling.

Although our approach is general and can be applied to construct multiscale methods with multipoint coarse-scale connections, the details will only be developed for the case in which the interfaces consist of the entire face between two neighboring grid blocks. The result is a two-point flux-approximation method (called MsTPFA) in which each coarse-scale transmissibility captures the local properties of the differential operator in a localized region between the centers of two neighboring coarse blocks. For smooth heterogeneities, the resulting method may be less accurate than previous multiscale methods that feature multipoint coarse-scale connectivity. However, as our requirements for gridding and petrophysical properties are quite demanding, we will be willing to sacrifice some accuracy if it leads to increased robustness and ease of implementation. Finally, the MsTPFA method is easy to integrate in existing simulators either as a preconditioner or as a stand-alone solver. Moreover, although this is not elaborated and exploited herein, the method has a large degree of inherent parallelism that can likely be exploited to obtain an implementation with good scalability.

2 The Multiscale Two-Point Flux-Approximation Method

The starting point of our discussion is a discrete linear system,

$$\mathbf{A}p = q, \tag{1}$$

which defines a fine-scale pressure problem. The system can be produced in any number of ways: from a stationary (single-phase) pressure equation, as a pressure subsystem extracted from a fully implicit system, or as a decoupled pressure equation in a sequential pressure-transport splitting,

$$-\nabla \cdot (\mathbf{K}\lambda_t \nabla p) = q, \tag{2}$$

for some total mobility λ_t and permeability \mathbf{K} . Our numerical examples will be limited to the case when (2) is the discretization of an elliptic pressure equation. Although this is not a prerequisite

for the formulation of the new multiscale method, we will in the following tacitly assume that the fine-scale discretization is based upon a two-point flux-approximation (TPFA). The TPFA method is the industry-standard discretization for stratigraphic grids, both corner-point and PEBI type, and while it may suffer from grid-orientation effects, the method is monotone.

The multiscale two-point flux approximation (MsTPFA) method is, like the name suggests, based on a coarse-scale operator that gives a two-point type stencil for computing global flow patterns on a coarse grid much like in traditional transmissibility upscaling. In addition, the method has a set of prolongation operators that allow reconstructions of conservative solutions on the underlying fine-scale grid, e.g., as in the MsFV and MsMFE methods. In the following, we will describe the construction of basis and correction functions, formulation of coarse-scale problem, and reconstruction of fine-scale solutions in more detail. Our main purpose is to develop a multiscale method that works well on the *discretized* cell-centered flow equations, which introduces several features that are not necessarily needed on the continuous flow equations. However, in an attempt to simplify the presentation, we will switch between a geometric and an algebraic (operator) descriptions of the method, trying to always use the description we believe is most transparent to the reader.

To motivate the method, let us first look a bit in detail on the MsFV method. When constructing coarse-scale systems, the MsFV method uses two coarse grids: The primal grid defines a typical coarse grid as used for upscaling, and the dual of this grid is then used to decompose the domain into local pressure problems. Once these problems have been solved and assembled into a basis, the interactions between the basis functions over the primal coarse grid gives a coarse system with multipoint connections. In addition, it is common to construct correction functions that take care of the particulate part of the solution that is not represented by the homogeneous basis functions. This is the geometrical description of the MsFV method. The method can also be described in algebraic form, using vector partitions and manipulation and elimination of matrix blocks in the fine-scale discretization. This operator formulation is particularly useful when formulating the method on unstructured coarse and fine grids [29]. We will not delve deeper into the MsFV derivation here, but simply refer to [25] as it gives a good introduction to both the underlying mathematical principles and the operator formulation.

To avoid MsFV's dependency on two coarse partitions, MsTPFA is designed using a single partition much in the same manner as in the MsMFE method and traditional transmissibility upscaling. We start by discussing the construction of basis functions for pressure, which consists of two parts. The first step is a preprocessing step that is performed once for a given partition to map local problems to the final basis. The second step is the solution of local pressure problems. We will describe the local pressure problems first, as they motivate the preprocessing step.

2.1 Local flow problems

Multiscale and most upscaling methods use local flow problems to estimate how fine-scale properties contribution to a coarse-scale equation that defines the global pressure field. The MsMFE method uses local problems defined over coarse faces and associates the degrees of freedom with the coarse faces. The MsFV method assigns local problems to the interaction regions around a coarse block, giving a single degree of freedom per coarse block. In the MsTPFA formulation, we solve local problems per coarse face, but we will finally associate the coarse system with a single degree of freedom per coarse block.

Figure 1 illustrates how local flow problems are defined for each coarse face and restricted to the two coarse blocks that share the face. To further localize the problem, we introduce two additional boundaries that consist of all cells intersected by a plane through each block center defined by a normal vector directed towards the center point of the other block. In the left plot of Figure 1, these cells are shown in blue and red for each side of the coarse face, respectively. The local problem has unit pressure imposed on one side and zero on the other and no-flow boundary conditions are specified along the remaining outer boundaries of the local domain.

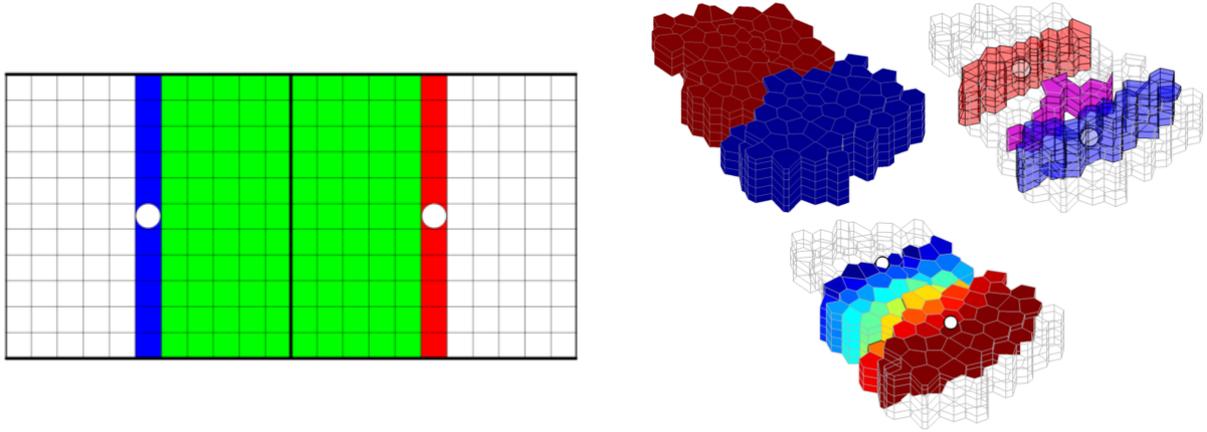


Figure 1: Definition of local flow problems. The left plot shows two coarse blocks and a single face, with unit pressure specified in the blue cells, zero pressure specified in the red cells, and pressure solution computed in the green cells. The three plots to the right show the same construction for a 2.5D PEBI grid: blocks sharing an interface, specification of fixed pressures, and local pressure solution, respectively.

The local flow problem defined over a local domain Ω reads

$$-\nabla \cdot (\mathbf{K}\lambda_t \nabla p) = 0, \quad (3)$$

with boundary conditions

$$p = 1 \text{ on } \Gamma_{in}, \quad p = 0 \text{ on } \Gamma_{out}, \quad (\mathbf{K}\lambda_t \nabla p) \cdot \vec{n} = 0 \text{ on } \partial\Omega \setminus (\Gamma_{in} \cup \Gamma_{out}), \quad (4)$$

where Γ_{in} and Γ_{out} denote the inflow and outflow Dirichlet boundaries, respectively, and \vec{n} denotes the normal vector. In the continuous formulation, this setup is identical to a popular variant of transmissibility upscaling, but in the discrete version it deviates slightly since we have chosen to impose the Dirichlet boundary as set of cells with prescribed pressure values rather than setting the Dirichlet boundary at a fixed location and discretizing the corresponding boundary conditions in a conventional manner. The motivation for doing so, is to simplify the algebraic construction of the method, which is the one we use in practice. To develop this construction from the existing global fine-scale equations, we extract all the rows corresponding to the cells in the local domain (i.e., the red, green and blue part of the left plot in Figure 1) and permute the columns so that all the local unknowns appear in one block \mathbf{p}_i ,

$$[\mathbf{A}_{i1} \quad \dots \quad \mathbf{A}_{ii} \quad \dots \quad \mathbf{A}_{in}] \begin{bmatrix} \mathbf{p}_1 \\ \vdots \\ \mathbf{p}_i \\ \vdots \\ \mathbf{p}_n \end{bmatrix} = \begin{bmatrix} \mathbf{q}_1 \\ \vdots \\ \mathbf{q}_i \\ \vdots \\ \mathbf{q}_n \end{bmatrix}.$$

Then the local matrix is given by

$$(\mathbf{A}_{loc})_{km} = \begin{cases} (\mathbf{A}_{ii})_{km}, & k \neq m \\ (\mathbf{A}_{ii})_{kk} + \sum_{j \neq i} \sum_{\ell} (\mathbf{A}_{ij})_{k\ell}, & m = k \end{cases} \quad (5)$$

Further modifications are made to impose boundary conditions, i.e., replacing the entries of rows corresponding to the boundary (red/blue) with values one or zero. Linear interpolation is used when the centroid of a 'Dirichlet boundary cell' does not lie exactly on the plane that defines the boundary. Finally, the right-hand side of the local problem is set up by forcing correct values (i.e., zero everywhere except on the boundary). This is equivalent to solving a Dirichlet problem with constant pressure boundaries and no-flow on the boundary tangent to the flow direction.

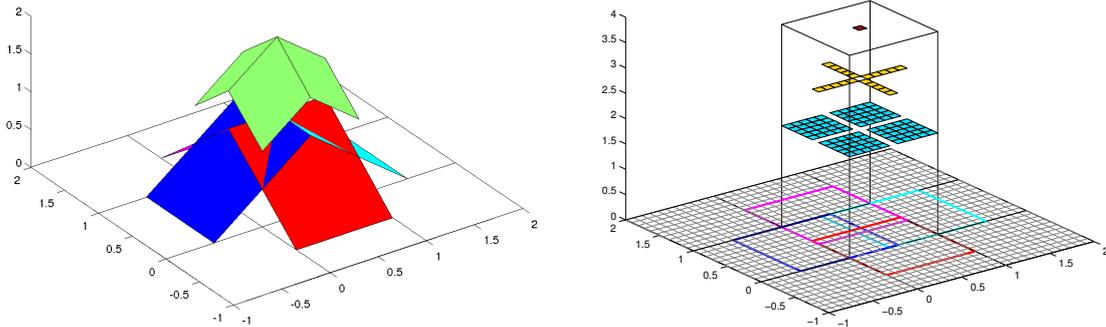


Figure 2: The left plot shows the four elementary solutions p_ℓ ($\ell = E, W, N, S$) defined for a square block along with their superposition within the block. The right plot shows computational domains for the elementary solutions along with the sum, $\sum_\ell(p_\ell + \hat{p}_\ell)$, computed numerically inside the coarse block.

2.2 Partition of unity and definition of basis functions

It is possible to set up a conservative coarse-scale system based on the local flow solutions per face, much in the same manner as for transmissibility upscaling. However, obtaining a fine-scale pressure solution is more difficult. The local solutions can obviously not be used as basis functions directly. To illustrate this, we will look at a simple problem.

Example 1 Assume that the quadratic domain $[-1, 2] \times [-1, 2]$ is divided into quadratic blocks of unit size. For the block $[0, 1] \times [0, 1]$, we will have four different local solutions associated with the east, west, north, and south faces, respectively:

$$\begin{aligned} p_E(x, y) &= 1.5 - x, & (x, y) &\in [0.5, 1.5] \times [0, 1], \\ p_W(x, y) &= 0.5 + x, & (x, y) &\in [-0.5, 0.5] \times [0, 1], \\ p_N(x, y) &= 1.5 - y, & (x, y) &\in [0, 1] \times [0.5, 1.5], \\ p_S(x, y) &= 0.5 + y, & (x, y) &\in [0, 1] \times [-0.5, 0.5] \end{aligned}$$

that describe flow out over each face and into the neighboring block. The basis functions and their superposition restricted to the coarse center block are shown to the left in Figure 2. If we look at the east face, for instance, there will be a similar elementary flow solution $\hat{p}_E = x - 0.5$ that describes flow from the neighbor and into the block. Obviously, these two elementary flow solutions must sum to unity, because if not, they would not be able to describe a constant pressure solution. Each pair of elementary solutions, p_ℓ and \hat{p}_ℓ , computed with the discrete approach described above will by construction sum to unity inside their domain of definition. However, as illustrated in the right plot of Figure 2, the local computational domains will have a non-trivial overlap, which means that it is not possible to represent a constant pressure using a simple superposition of the numerically computed elementary solutions, which is a requirement if such a superposition is to be useful as a multiscale finite-volume prolongation/basis operator.

To construct a useful basis, we will introduce an additional set of functions, one associated with each coarse face, that together will form a partition of unity. The purpose of these partition-of-unity functions is to extract the useful parts of the elementary solutions and discard the parts we are not interested in. That is, because the coarse-scale discretization seeks to mimic a two-point scheme we want to keep the part of the elementary functions that accurately represents the pressure drop between the two block centers on opposite sides of each face. Likewise, we wish to remove as much as possible of the local effects of the no-flow and Dirichlet boundary conditions used to localize the elementary solutions. Figure 3 shows how such a partition of unity w_ℓ ($\ell = E, W, N, S$) can be constructed for the case discussed in Example 1. Here, the

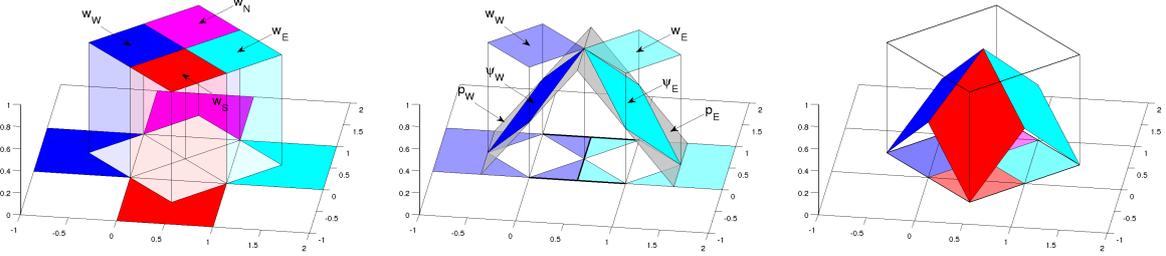


Figure 3: Using partition of unity to define basis functions for the case in Example 1. The left plot shows the four functions w_ℓ ($\ell = E, W, N, S$) defining the partition of unity. The middle plot shows how partition functions w_E and w_W are used to extract parts of the elementary solutions p_E and p_W to define face bases ψ_E and ψ_W ; thick lines indicate where boundary conditions (Dirichlet or no-flow) are specified to localize the problem in the coarse block. The right plot shows the resulting block basis with support on a rotated square.

support of each function w_ℓ covers the corresponding coarse face and narrows in towards the two block centers to which it is associated. In general, if p_j^i denotes the elementary function defined between block i having unit pressure and block j having zero pressure and w_j^i denotes the corresponding partition-of-unity function, one can define a basis for each face,

$$\psi_j^i = w_j^i p_j^i \quad (6)$$

and a block basis that is simply the sum of the basis functions for all coarse faces in the block,

$$\psi^i = \sum_j \psi_j^i = \sum_j w_j^i p_j^i. \quad (7)$$

In Figure 3, the middle plot shows the construction of face bases while the plot to the right shows the a block basis. The basis function associated with the centroid of a coarse block has support on what would be considered the local neighborhood for a coarse block using a TPFA approximation. (The MsFV method, on the other hand, has a support on what could be considered an MPFA-like neighborhood).

The smoothness of the pressure approximation is inherited from the elementary solutions p_j^i and the partition functions w_j^i . In practice, one will obviously not use a discontinuous partition of unity as shown in Figure 3. Indeed, if the partition functions are smooth and zero away from the boundaries used to localize the elementary functions, we will obtain an overall smooth pressure approximation.

Partition functions having the required properties can easily be constructed on almost arbitrarily shaped coarse block, regardless of whether the underlying grid is structured or unstructured. To motivate our approach, we consider a physical analogue. Assume that each coarse face is assigned a unique color and that ink with unit concentration of this color is injected uniformly along the face and a corresponding amount of fluid is extracted from point sources located at the block centers. If we continue injecting, a suitable partition of unity can be obtained from the steady-state ink concentrations inside each coarse block. Computationally, this is achieved by solving an incompressible pressure equation with homogeneous, isotropic permeability for each coarse block with a sink in the cell center and sources equal in magnitude to the face area on each face. Once the corresponding flow field \vec{v} is obtained, the stationary tracer distribution for inflow boundary Γ can be computed from

$$\nabla \cdot (\vec{v}c) = 0, \quad c = 1 \text{ for } \vec{x} \in \Gamma. \quad (8)$$

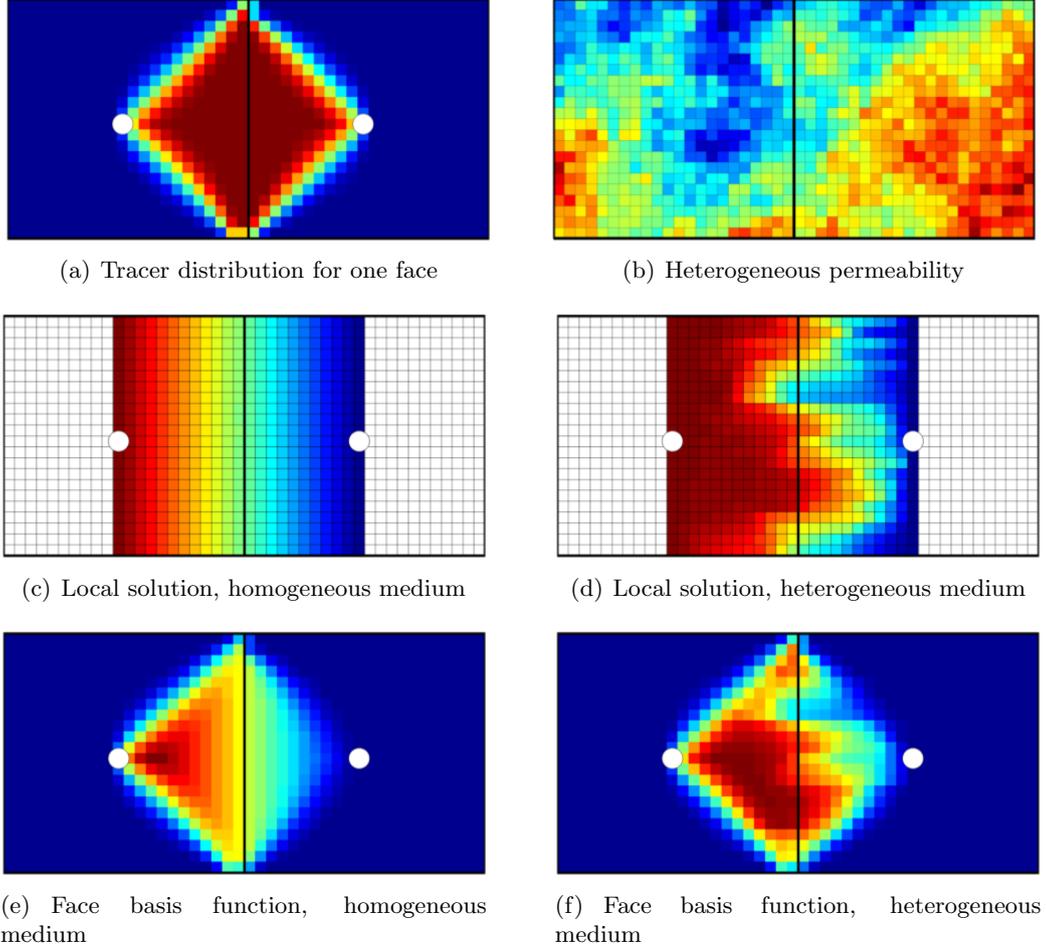


Figure 4: Numerically computation of face basis for a homogeneous and a heterogeneous 2D medium.

The tracer equation (8) is then solved with a unique tracer for each coarse face, which gives values between zero and one for cells inside the coarse block. Figure 4 illustrates the resulting computation of a face basis function for a homogeneous and a heterogeneous case.

To solve the tracer equation, we use a standard first-order upwind discretization, which has an inherent numerical diffusion that will give us the required smoothness of the partition functions. If a two-point method is used to compute the flow field, the resulting fluxes will form a directed graph that can be used to reorder the discrete tracer equations into a triangular form that can be inverted efficiently by an algorithm that visits each cell once [31]. Hence, the cost of constructing partition functions is dominated by the cost of solving one pressure equation per block, an operation that is naturally parallel and only has to be performed once as the grid is partitioned into coarse blocks.

2.3 Correction functions and treatment of wells

While multiscale methods may produce a fine-scale pressure field from a coarse-scale problem, certain fine-scale effects will only be included in the coarse sense. Although the coarse-scale solution may resolve the global pressure trends, many interesting features are important to model on a fine scale. For an illustration of this we will consider a simple example.

Example 2 *Let a rectangular discretized domain of $[0, 2] \times [0, 1]$ be divided into two unit-size blocks. Flow is driven by two injectors with $p_{bhp} = 1$ at $(1/4, 3/4)$ and $(3/4, 1/4)$ and a producer*

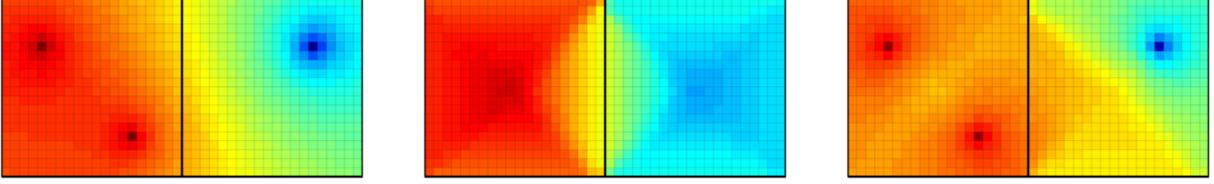


Figure 5: The reference pressure solution (left) contains fine-scale information not present in the prolonged coarse multiscale solution (middle) which can be reintroduced by the use of correction functions (right)

with $p_{bhp} = 0$ at $(3/4, 1/4)$, and no-flow boundary conditions ($\mathbf{K}\lambda_t\nabla p \cdot \vec{n} = 0$) are imposed on the boundary. The resulting solution is shown to the left in Figure 5. If the coarse blocks are used along with regular basis functions to compute a fine-scale pressure, any combination of wells giving the same integral flux over the coarse boundaries will result in the same fine-scale solution as shown in the middle plot. However, by constructing local correction functions that capture the fine-scale well behavior as explained below, the solution becomes unique as shown in the right plot in Figure 5.

The concept of correction functions was originally formulated for the MsFV method [22] and is straightforward to extend to the MsTPFA method. The central idea is to decompose the fine-scale pressure into two parts,

$$\mathbf{p} = \mathbf{B}\mathbf{p}_c + \mathbf{c}, \quad (9)$$

where \mathbf{c} is a *correction term* (or particulate solution) that resolves fine-scale effects not accounted for by the *homogeneous* basis functions. Whereas the basis functions solve a flow problem (3) driven by a pressure differential (4), the correction functions are defined as solutions to an *inhomogeneous* problem driven by a source term

$$-\nabla \cdot (\mathbf{K}\lambda_t\nabla p_c) = q, \quad (10)$$

with zero boundary conditions

$$p_c = 0 \text{ on } \Gamma_i \cup \Gamma_o, \quad (\mathbf{K}\lambda_t\nabla p_c) \cdot \vec{n} = 0 \text{ on } \partial\Omega \setminus (\Gamma_i \cup \Gamma_o). \quad (11)$$

Here, Γ_i and Γ_o denote the inflow and outflow boundaries of the corresponding homogeneous problem (4).

Once the local correction problem (10)–(11) has been solved for each coarse face, the solutions can be combined with the partition of unity to define the correction operator on the whole domain in a similar manner as Equations (6) and (7), producing a local solution for each pair of coarse blocks (i, j)

$$\mathbf{c}_j^i = w_j^i p_c \quad (12)$$

Because the correction term is independent of the coarse-scale problem, one can simply sum over all block-pairs to get the final correction over the entire domain

$$\mathbf{c} = \sum_{(i,j)} \mathbf{c}_j^i = \sum_j w_j^i p_c. \quad (13)$$

The fine-scale information present in correction functions representing wells and boundary conditions should replace and not add to the prolonged solution. It is therefore important that the flow problems (3) used to construct the corresponding homogeneous bases are modified locally to avoid interfering with the correction function. This is done by setting any pressure or flux equivalent to wells and boundary conditions to zero to leave 'gaps' in the solution of

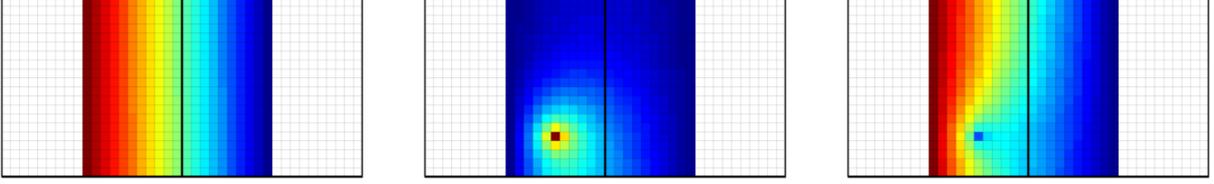


Figure 6: The choice of local problem for a coarse face depends on the use of correction functions. Without correction functions, the local problem is defined by \mathbf{A}_0 and is completely linear (left). Formulated in companionship with a correction function (middle), the local basis problem accounts for the flux out of the well perforation and is constructed using $\mathbf{A}_0 + \mathbf{D}_{bc}$ (right)

the prolonged system that are compatible with the correction functions. Mathematically, one replaces (3) with

$$-\nabla \cdot (\mathbf{K}\lambda_t \nabla p) = \tilde{q}, \quad (14)$$

where \tilde{q} represents the flux from any boundary conditions or wells set to a zero value. In the operator formulation, this is done implicitly. Any cell attached to a boundary with prescribed condition different from no-flow or containing a well perforation will have extra diagonal entries in the linear system corresponding to the boundary transmissibility or well model used. Thus one can extract local linear systems and solve these with zero right hand side. That is, one splits the fine-scale system matrix \mathbf{A}_{tot} into one part \mathbf{A}_0 representing cell-wise flux balances and a diagonal part \mathbf{D}_{bc} representing flux contributions from boundary conditions and wells,

$$\mathbf{A}_{tot} \mathbf{p} = (\mathbf{A}_0 + \mathbf{D}_{bc}) \mathbf{p} = \mathbf{q}. \quad (15)$$

If the basis functions will be used alongside correction functions, we set $\mathbf{A} = \mathbf{A}_0 + \mathbf{D}_{bc}$ in (5). On the other hand, if the solution will be constructed without correction functions, it is natural to set $\mathbf{A} = \mathbf{A}_0$ and impose boundary conditions and wells at the coarse scale. In some cases, including inhomogeneous effects on the fine scale may not be required to obtain an acceptable approximation, while in other cases these effects can be accounted for by iterative techniques [36] or coarse-grid refinement. The effect of omitting inhomogeneous fine-scale effects is illustrated in Figure 6, in which the basis function for a single coarse face from Example 2 is shown with and without correction functions.

2.4 Coarse system

The coarse-scale system for the MsTPFA method is straightforward to formulate, owing to the construction of local basis problems over coarse edges. Applying the divergence theorem to (2) for a coarse block Ω_i ,

$$-\int_{\Omega_i} \nabla \cdot \mathbf{K}\lambda_t \nabla p = -\int_{\partial\Omega_i} (\mathbf{K}\lambda_t \nabla p) \cdot \vec{n} = \int_{\Omega_i} q. \quad (16)$$

Recall that the basis function for each coarse block is denoted ψ^i and let $\mathcal{N}(i)$ be the set of neighbors for coarse block i and \tilde{p}_i be the coarse-scale pressure assigned to block i . This gives

$$\sum_{j \in \mathcal{N}(i)} T_{ij} (\tilde{p}_i - \tilde{p}_j) = -\int_{\partial\Omega_i} (\mathbf{K}\lambda_t \nabla \psi^i) \cdot \vec{n} = -\int_{\partial\Omega_i} (\mathbf{K}\lambda_t \nabla (\sum_j \psi_j^i)) \cdot \vec{n} = \int_{\Omega_i} q. \quad (17)$$

If we let Γ_{ij} denote the interface between blocks i and j , we can write $\partial\Omega_i = \cup_{j \in \mathcal{N}(i)} \Gamma_{ij}$. Moreover, because each local solution is defined to be nonzero for a single interface,

$$\int_{\Gamma_{ik}} (\mathbf{K}\lambda_t \nabla \psi_j^i) \cdot \vec{n} = 0 \quad \text{for } k \neq j. \quad (18)$$

Hence, the transmissibility for each interface depends only on a single local solution, giving

$$T_{ij}(\tilde{p}_i - \tilde{p}_j) = - \int_{\Gamma_{ij}} (\mathbf{K}\lambda_t \nabla \psi_j^i) \cdot \vec{n}, \quad j \in \mathcal{N}(i), \quad (19)$$

Source terms are handled in the integral sense, with the optional correction functions c_j^i being subtracted to avoid accounting for fine-scale effects twice

$$Q_i = \int_{\Omega_i} q + \sum_j \int_{\partial\Omega_i} (\mathbf{K}\lambda_t \nabla c_j^i) \cdot \vec{n} \quad (20)$$

Once these quantities are computed, we can set up the coarse-scale problem

$$\mathbf{A}_c \mathbf{p}_c = \mathbf{q}_c \quad (21)$$

where the entries of the matrix and right hand side are

$$(\mathbf{A}_c)_{ij} = \begin{cases} T_{ij} & i \neq j \\ \sum_{j \in \mathcal{N}(i)} -T_{ij} & i = j \end{cases} \quad \text{and} \quad (\mathbf{q})_i = Q_i. \quad (22)$$

Finally, if we construct a prolongation operator \mathbf{B} of size $N_f \times N_c$, in which entry i, j corresponds to the value of basis function (7) for coarse block j in fine cell i , the fine-scale pressure is approximated as

$$\mathbf{p} \approx \mathbf{B} \mathbf{p}_c + \mathbf{c}. \quad (23)$$

2.5 Conservative reconstruction

The multiscale fine-scale solution will not be conservative everywhere because, like most other multiscale methods, the initial pressure produced by the MsTPFA method only fulfills (1) approximately. If the resulting flux field is to be passed on to a fine-scale transport solver, it must be mass conservative on the fine scale to avoid unphysical values in the transport solver. For the MsFV method, this is typically handled by partially disconnecting the fine-scale flux field from the fine-scale pressure field, and solving local flow problems to reconstruct mass-conservative fluxes corresponding to the inexact multiscale pressure field. Herein, we will employ the same strategy as used in the MsFV method, which we for completeness will outline briefly below.

Conventional solvers let the fluxes be defined by the pressure, i.e., by applying Darcy's law directly,

$$\vec{v} = -\mathbf{K}\lambda_t \nabla p. \quad (24)$$

As the coarse-scale pressure is resolved by a flux conservative system (19), the multiscale fine-scale pressure gives a conservative flux field over the coarse-block interfaces. We can exploit this by solving local flow problems

$$-\nabla \cdot (\mathbf{K}\lambda_t \nabla \bar{p}) = q, \quad (25)$$

in the local domain Ω with Neumann boundary conditions sampled from the block interfaces where the fine-scale pressure is conservative,

$$(\mathbf{K}\lambda_t \nabla \bar{p}) \cdot \vec{n} = (\mathbf{K}\lambda_t \nabla p) \cdot \vec{n} \text{ on } \partial\Omega. \quad (26)$$

We proceed to apply (24) to the reconstructed pressure \bar{p} , giving a new flux field that is conservative everywhere. For further discussion of these local problems as well as the operator analogue thereof the reader is referred to [25].

2.6 Iterative multiscale formulation

The pressure extrapolation described above gives an initial approximation to the fine-scale flow equations that may or may not be sufficiently accurate. To increase the accuracy of the multiscale approximation, the traditional MsFV method has been extended to an iterative formulation, the i-MsFV method [8, 11], in which the mass-conservative multiscale operator is combined with an inexpensive iterative solver to systematically drive the fine-scale residual towards zero. In this multigrid-like approach, the coarse-scale multiscale operator will resolve global features in the pressure field, while the smoother is used to construct correction terms that account for errors in the prolonged pressure field.

If we let \mathbf{p}_c^n denote the coarse pressure at step n and let \mathbf{c}^n be some correction term defined such that

$$\|\mathbf{A}(\mathbf{B}\mathbf{p}_c^n + \mathbf{c}^n) - \mathbf{q}\| \leq \|\mathbf{A}\mathbf{B}\mathbf{p}_c^n - \mathbf{q}\|, \quad (27)$$

a more accurate fine-scale pressure approximation is obtained by setting $\mathbf{p}_s^{n+1} = \mathbf{B}\mathbf{p}_c^n + \mathbf{c}^n$. While this pressure reduces the residual error of (1), it is no longer guaranteed to be conservative on the coarse scale. Hence, instead of using \mathbf{p}_s^{n+1} directly, we will construct a new coarse-scale solution \mathbf{p}_c^{n+1} that is mass-conservative and accounts for the residual effects of \mathbf{c} so that

$$\mathbf{p}^{n+1} = \mathbf{B}\mathbf{p}_c^{n+1} + \mathbf{c}^n. \quad (28)$$

Inserting (28) into (1) and applying the discrete restriction operator

$$\chi_{ij} = \begin{cases} 1, & \text{if cell } j \text{ is in coarse block } i, \\ 0, & \text{otherwise,} \end{cases} \quad (29)$$

gives the coarse-scale problem

$$\chi\mathbf{A}(\mathbf{B}\mathbf{p}_c^{n+1} + \mathbf{c}_n) = \chi\mathbf{q}. \quad (30)$$

Defining $\mathbf{A}_c = \chi\mathbf{A}\mathbf{B}$ and moving the actions of the correction function to the right-hand side, the new coarse-scale systems reads,

$$\mathbf{A}_c\mathbf{p}_c^{n+1} = \chi(\mathbf{q} - \mathbf{A}\mathbf{c}^n). \quad (31)$$

Substituting the solution of (31) into (28) gives a new fine-scale pressure approximation that is conservative on the coarse scale since the coarse interface fluxes from \mathbf{c} have been accounted for via the modified right-hand side. This is referred to as a *multiscale cycle*. It should be noted that writing $\mathbf{A}_c = \chi\mathbf{A}\mathbf{B}$ gives a coarse-scale discretization that is not strictly two-point because the basis functions overlap at the vertices of the coarse blocks, which will result in additional weak multipoint couplings.

So far, no assumptions have been made about exactly *how* \mathbf{c} is obtained, other than it should fulfill (27). In general, if we assume that the coarse-scale operator is able to resolve the global features of the system, any significant errors will be on the sub-scale. Hence, the local solver can be chosen in the same way as in traditional multigrid methods. A local solver for multiscale methods should thus be

- able to *reduce localized errors* present in the initial prolonged coarse-scale solution;
- *inexpensive* to construct and apply;
- *local* so that the domain-of-dependence for each node in the local solver should be small.

There are many methods that fulfill these requirements. For our purposes, the most important distinction is between solvers that require a setup phase and those who do not. Methods without a setup phase include iterative solvers such as Jacobi/Gauss–Seidel and block variants thereof, whereas methods with a setup phase are exemplified by various multigrid methods as well as

LU-based preconditioners in which an incomplete factorization is used to approximate \mathbf{A} . Local solvers with a setup cost are likely useful when several multiscale cycles are required, while iterative solvers without setup cost are more useful when only a modest number of iterations are required, for instance when a single multiscale cycle gives satisfactory results. Herein, we will only utilize what is likely the simplest local solvers in each category to simplify the treatment and implementation. For a method without a setup phase, we will use Jacobi iterations and when more iterations are used, we will use incomplete LU factorization with zero fill-in (ILU-0) along with the multiscale operator stabilized with GMRES.

Although it is possible to use the approach described above to iterate to any desired accuracy, it may not be computationally feasible compared with other methods such as algebraic multigrid, which may achieve better performance if the target is to reduce the fine-scale residual to machine precision. In most workflows, however, one will typically seek computational savings by aborting the iteration of the multiscale pressure solution before it reaches machine precision accuracy or even just use the initial MsTPFA approximation, followed by a mass-conservative reconstruction. In the next section, we will therefore look at both the initial solution quality, the quality after a single cycle with a few Jacobi iterations, as well as convergence for Ms-ILU0-GMRES.

2.7 Connection to upscaling techniques

The MsTPFA method described herein can be seen as a multiscale realization of the standard local interface based transmissibility upscaling. It should be noted that while this paper is focused strictly on this specific coarse scale operator, it is possible to use the same framework to create multiscale-like methods from conventional upscaling techniques. Further avenues of research would include incorporating generic, global solutions in the creation of the basis functions or by creating multipoint-like stencils by changing the local problems. Even though we have specifically chosen the coarse faces for the partition of unity and the local solutions to get a TPFA-like operator, another choice would be to include the corners to get a MPFA-like stencil.

3 Numerical results

The MsTPFA method has been implemented as an open-source prototype as part of the `msfvm` module in MRST [30, 21]. To validate the method, we will consider several different test cases with both varying permeability and geometry. In all examples, we will compare the approximate multiscale solutions to the fine-scale reference solutions using scaled L_∞ and L_2 norms,

$$\|\mathbf{p}\|_\infty = \frac{\|\mathbf{p}_{fs} - \mathbf{p}_{ms}\|_\infty}{\|\mathbf{p}_{fs}\|_\infty}, \quad \|\mathbf{p}\|_2 = \frac{\|\mathbf{p}_{fs} - \mathbf{p}_{ms}\|_2}{\|\mathbf{p}_{fs}\|_2}. \quad (32)$$

As the proposed multiscale formulation has the advantage of being very flexible with regards to grids, coarse grids that adapt to permeability features will be used when appropriate. To produce adapted grids, we will use Metis [27] configured to use the transmissibilities of the fine-scale system as weights for the edge-cut minimization algorithm. We also allow the coarse blocks to vary by 75% in cell number to give the partitioning algorithm some freedom to capture fine-scale details. To give fairly regular blocks in regions of low contrast, the 'minconn' option was specified. Although it would be possible to also use information of the boundary conditions in the construction of the coarse grid, we will for simplicity not consider such coarse grids here.

3.1 SPE10

Model 2 from the 10th Comparative Solution Project [6] was originally designed as a challenging benchmark for upscaling method, and computing flow on subsets of this model has over the years been established as the de facto for multiscale methods. The model is part of a Brent

sequence and contains two formations that are qualitatively different: the Tarbert formation represents a prograding near-shore environment in which the permeabilities follow a lognormal distribution, giving smoothly varying heterogeneities that most multiscale methods are capable of resolving quite well. The Upper Ness formation is fluvial with long high-permeable sand channels appearing in the west-east direction interbedded with low-permeable mudstone. The combination of long correlation lengths and strong abrupt changes makes the Upper Ness very challenging to resolve correctly. The MsFV method, for instance, will typically exhibit strong unphysical pressure oscillations [33, 38, 35, 36, 29].

Horizontal layers. As a first example, we will consider two horizontal 60×220 sub-samples: the bottom layer of the Tarbert formation (logical index $k = 35$ in the full model) and the middle layer of the Ness formation ($k = 60$). We consider two different coarse grids: a uniform 12×22 partition and a 264-block partition produced by Metis, giving an upscaling factor of 50. Figure 7 shows the permeability fields, the coarse grids, and the MsTPFA approximate solutions before and after five iterations with a block-Jacobi smoother; the corresponding errors are reported in Table 1 and Figure 8 reports the reduction in residual for a full multiscale iteration.

Table 1: The error in the pressure solution for two horizontal subsets of SPE10 computed for two different coarse grids by a single multiscale cycle without smoothing and with five Jacobi iterations for both the MsTPFA and the MsFV methods

Setup of model			Multiscale		Multiscale+Jacobi	
permeability	Solver	coarse grid	L_2	L_∞	L_2	L_∞
Tarbert	MsFV	uniform	0.0069	0.0586	0.0035	0.0239
Tarbert	MsTPFA	uniform	0.0735	0.2641	0.0450	0.1389
Tarbert	MsTPFA	adapted	0.0262	0.1741	0.0158	0.0798
Upper Ness	MsFV	uniform	0.9525	19.353	1.3696	11.227
Upper Ness	MsTPFA	uniform	0.3331	3.1365	0.2395	2.2114
Upper Ness	MsTPFA	adapted	0.0427	0.1415	0.0243	0.0577

For the Tarbert layer, the solution quality is comparable to what has been observed for multiscale methods previously, and adapting the coarse grid to structures in the fine-scale transmissibilities only improves the initial MsTPFA solution in the L_2 norm. For the Upper Ness layer, we observe several regions with non-monotone pressure caused by negative transmissibilities introduced in the coarse system because the coarse blocks are arbitrarily placed without regard to the highly structured channelized features of the permeability field. (Compared with the MsFV method, however, these unphysical regions are much less pronounced.) Adapting the coarse grid to the transmissibilities in the fine grid greatly reduces the number of negative coarse-scale transmissibilities. This means that the accuracy of the multiscale coarse-grid operator is significantly improved, giving both error and convergence that is comparable to the Tarbert layer.

Comparison with MsFV. We have run a large number of tests on other 2D and 3D sub-samples of the SPE10 model, including cases with a single fault added in the middle of the domain. The results are qualitatively similar to what was observed for the two horizontal layers discussed above: uniform partitions may introduce certain non-monotonicities, and these are slightly stronger in 3D for non-unit aspect and anisotropy ratios, but can to a large extent be mitigated by using a partition that adapts to fine-scale transmissibilities. Results are not included for brevity. Instead, we will in this example compare the MsTPFA method with the MsFV method on 3D box models with and without a single fault, as discussed in [29]. Our current method for generating the wire-basket ordering needed by MsFV is not yet sufficiently robust to always produce a sensible dual partition for arbitrary transmissibility-adapted primal grids generated

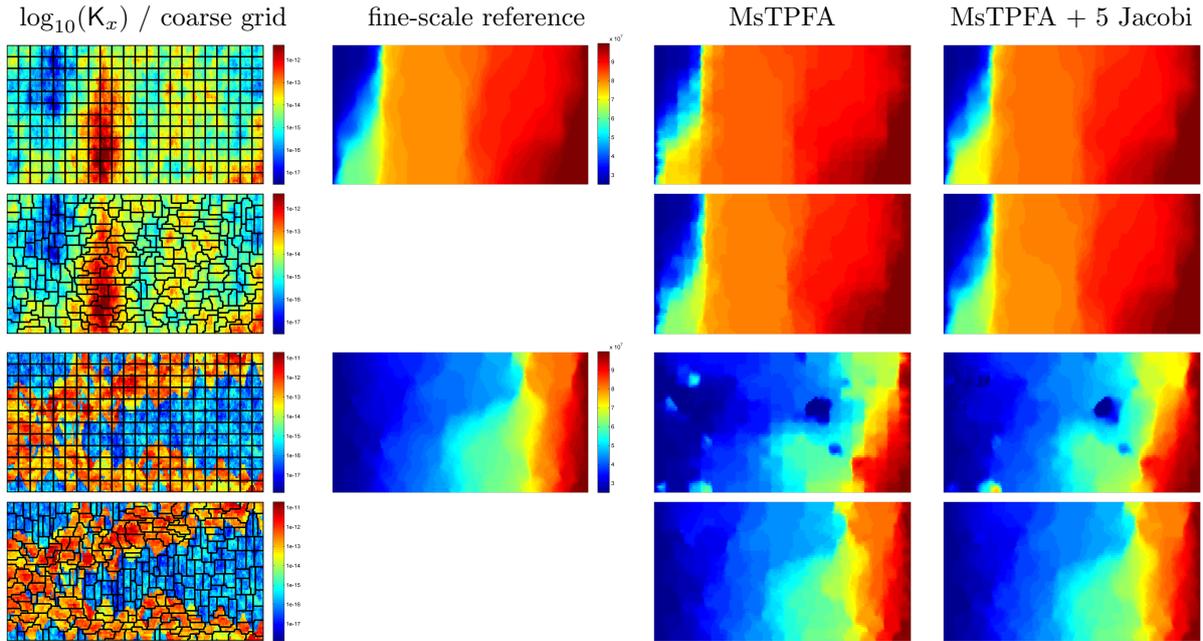


Figure 7: Multiscale solutions for two horizontal sub-samples of the SPE10 data set.

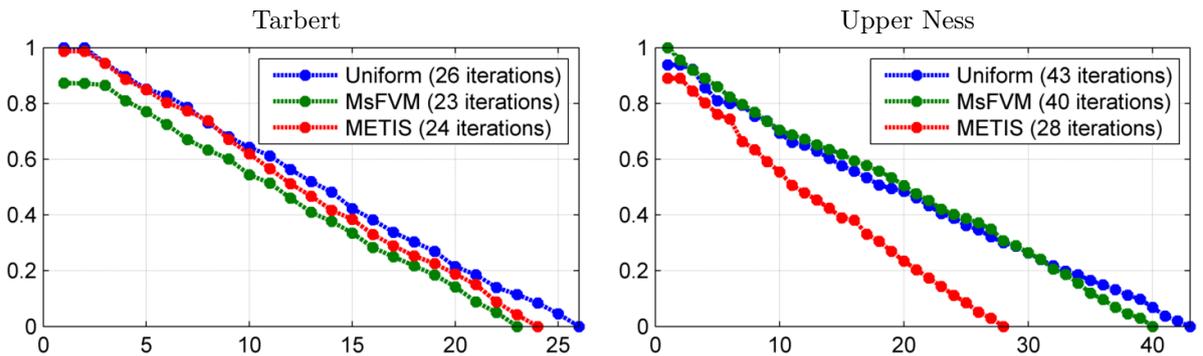


Figure 8: Convergence of multiscale cycles to a tolerance of 10^{-8} on two horizontal sub-samples of the SPE10 data set. The y -axis is logarithmic and normalized to the largest initial residual error and the convergence of the final solution.

by Metis, and hence we only consider uniform primal partitions. Figure 9 reports a comparison of the MsFV and MsTPFA solutions before and after smoothing for four different 3D models. For the relatively smooth Tarbert formation, the MsFV method computes solutions that are qualitatively more accurate than those of the MsTPFA method, except for the small unphysical solutions introduced near the fault by the MsFV method, which are difficult to spot in the plots. For the strongly heterogeneous Upper Ness formation, on the other hand, the MsFV method introduces large unphysical oscillations that are amplified by smoothing iterations, whereas the MsTPFA method is able to capture a qualitatively correct picture of the fine-scale solution. Altogether, this example is a good illustration of the purpose of the MsTPFA method: by sacrificing a certain accuracy for problems with smooth heterogeneity, one gains significantly in robustness for strongly heterogeneous problems.

3.2 Realistic bed model

One of the main sources of unstructured connections in reservoir modeling is pinch-outs, for which erosion or other geological features lead to inactive or highly degenerate cells. To get

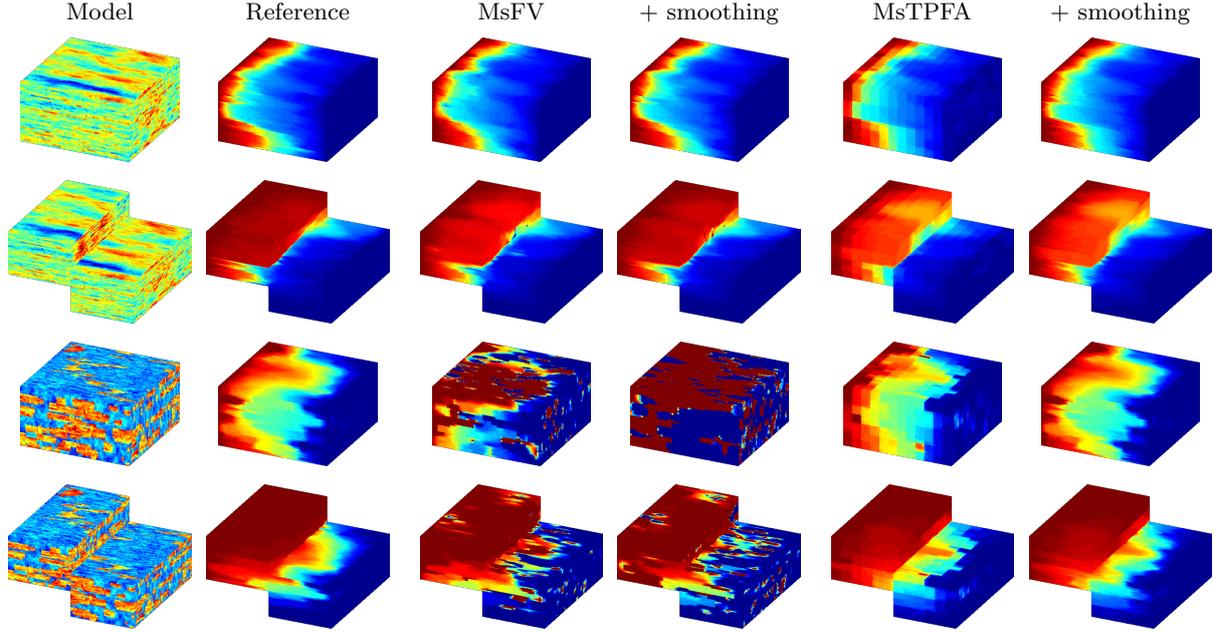


Figure 9: Multiscale solutions computed by the MsFV and MsTPFA methods before and after smoothing for the 3D models using permeability sampled from the SPE10 data set.

Table 2: Error in the pressure solution for a SBED model containing large amount of pinched and inactive cells for some coarse blocks. Here, Metis* refers to an alternate partition in which shales and sand are enforced as separate partitions.

Coarse grid	Blocks	MsTPFA		MsTPFA + Jacobi	
		L_2	L_∞	L_2	L_∞
Uniform	54	0.1048	0.9507	0.0722	0.4019
Metis	54	0.0967	0.9461	0.0744	0.5111
Metis*	54	0.1166	0.9859	0.0909	0.3849
Metis*	200	0.0899	0.9867	0.0665	0.2108
Metis*	1000	0.0448	1.0113	0.0273	0.2275
Metis*	5000	0.0202	0.6534	0.0109	0.0594

a test case with a high number of pinch-outs, we consider a highly detailed model designed to reproduce realistic, small-scale bedding structures on a scale much smaller than a single full reservoir simulation block. The model contains approximately 90 000 fine cells and has previously been used as test case for the MsMFE method [3] and the MsFV method [29]. Even though the domain is rectangular in appearance, almost every cell contains non-neighboring connections and degenerate features. The variance of the permeability distribution is for the most part modest, but intersecting layers of shale make the model especially challenging for the localization assumption in the MsFV method. This type of model is typically built as part of an upscaling workflow to determine effective directional permeability for a given lithofacies and to identify net pay below petrophysical log resolution. Hence, we use Dirichlet boundary conditions to impose a constant pressure drop in the horizontal direction.

We consider three different partition methods that all generate a coarse model with 54 blocks: (i) a simple uniform partition in index space as used for the MsFV method in [29]; (ii) a Metis partition as discussed above; and (iii) a Metis partition in which we do not allow coarse blocks that mix the low-permeable shale and the high-permeable background. For partitions with many blocks, Metis has a tendency to create strongly convex blocks (e.g., consisting of a set of thin shale cells plus a column of sand cells), which will typically produce negative transmissibilities

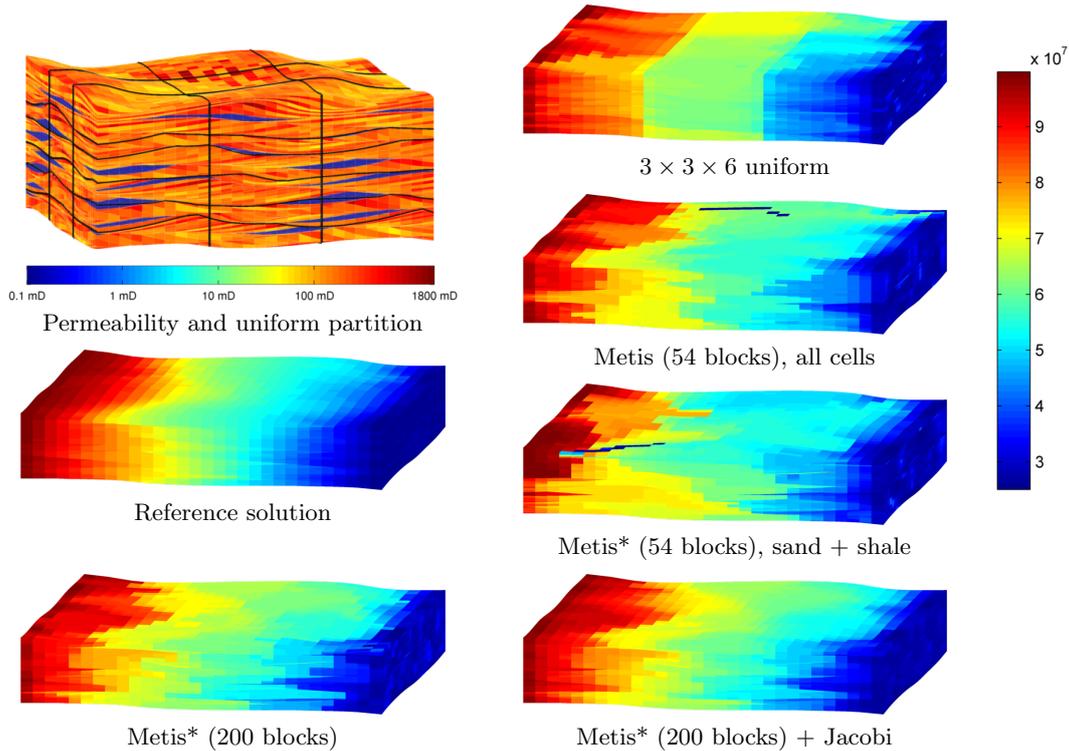


Figure 10: Multiscale solutions computed on different partitions for a high-resolution, core-scale bedding model.

for the coarse system. The third approach is introduced to avoid creating such blocks and uses a straightforward modification of the transmissibility graph so that sand and shale cells are partitioned separately into smaller regions based on transmissibilities. Figure 10 compares the initial multiscale solutions obtained for the three different partition strategies. Errors in L_2 and L_∞ norm before and after smoothing iterations are reported in Table 2. For comparison, the figure and table also include results obtained on finer resolutions for the third partition method. With uniform partition, the multiscale solution has a strong patchy behavior that cannot be rectified by smoothing iterations. Using Metis to partition the cells gives a qualitatively correct solution, except for a small region of unphysical pressure values which are efficiently removed by smoothing cycles. When partitioning sand and shale separately, almost one third of the 54 blocks are required to represent disconnected shale units, which leaves Metis less freedom to create a good partition of the sand volume in which most of the flow occurs. As a result, the corresponding multiscale solution has more grid artifacts than for the second strategy, but after smoothing the L_∞ error is lower than for the other methods.

The third strategy is more effective for a higher number of coarse blocks. To demonstrate this, we consider a wide range of coarsening degrees ranging from 54 to 5000 coarse blocks. The histogram in Figure 11 confirms that the initial L_2 error decays with the size of the coarse blocks. The right-hand plot in Figure 11 reports reduction in residual by an iterative Ms-ILU-GMRES solver. With only 54 blocks, the reduction in residual decays after the first few iterations, indicating that the multiscale coarse operator only contributes significantly to the initial reduction of the residual. With 5000 blocks on the other hand, the multiscale coarse operator contributes significantly to reduce the residual all the way to machine precision.

3.3 The Norne Field

For this example, we will use the corner-point grid from the simulation model of the Norne reservoir from the Norwegian Sea. The simulation model has a layered permeability distribution

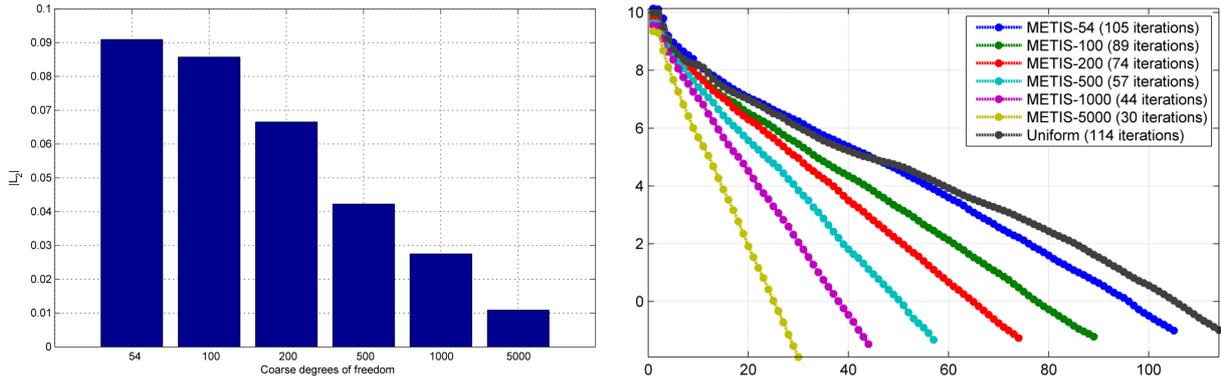


Figure 11: The L_2 error for the bed model as a function of varying number of coarse blocks. The histogram shows the error of the initial multiscale solution, and the line plots show the reduction in the logarithm of the preconditioned residual for the corresponding Ms-ILU-GMRES solver.

Table 3: Error in the pressure solution for a realistic simulation model with geometry and permeability data from the Norne Field. Approximate solutions are compute for three different coarse grids by a single multiscale cycle without smoothing and with ten Jacobi iterations.

Coarse grid	MsTPFA		MsTPFA + Jacobi	
	L_2	L_∞	L_2	L_∞
Uniform	0.0782	0.7771	0.0554	0.3338
Fault-adapted	0.0599	0.9039	0.0387	0.1259
Metis	0.0662	0.9132	0.0402	0.0971

and contains faults, eroded cells and pinch-out leading to non-neighboring connections as well as jumps in permeability. The grid contains approximately 45 000 fine cells, out of which 25% have either more or less than six faces, with the number of faces ranging from four to nineteen. The input data uses a description based on a structured topology, but once processed, the resulting grid has a non-structured topology and irregular geometry. To give flow across the entire domain, two vertical producers operating at fixed bottom-hole pressure $p_{bhp} = 100$ bar are added to the reservoir along with a 500 bar pressure boundary condition at one of the outer boundaries at the opposite side of the wells, see Figure 12.

We will simultaneously consider three different coarse partitions that give approximately 200 coarse degrees of freedom: (i) a simple uniform partition in index space that takes neither geometry nor permeability into account, but gives an unstructured coarse grid because of inactive cells in the fine grid; (ii) a uniform partition but using geometry information to split blocks across faults; and (iii) a purely topological partition produced by Metis. Table 3 reports errors for the three partitions. The last two partitions introduce large errors near the boundaries of the coarse blocks and thus have large initial L_∞ errors. However, these errors can be efficiently removed by a few inexpensive Jacobi iterations. The uniform partition has lower initial L_∞ error, but since the coarse-scale operator is less accurate, the L_∞ error is significantly higher compared with the other two partitions after a single multiscale cycle. Similar results are reported for the MsFV method with the fault-adapted partition in [29]; for the uniform partition, the MsFV method failed to produce a solution, and for the Metis partition, generating a dual partition was not feasible.

To investigate the robustness of the multiscale operator, we use Metis to generate ten different coarse partitions, with number of blocks ranging from 10 to 5 000, giving upscaling factors from 9 to 4 500. The left plot in Figure 13 reports the L_2 error of the initial multiscale solutions, whereas

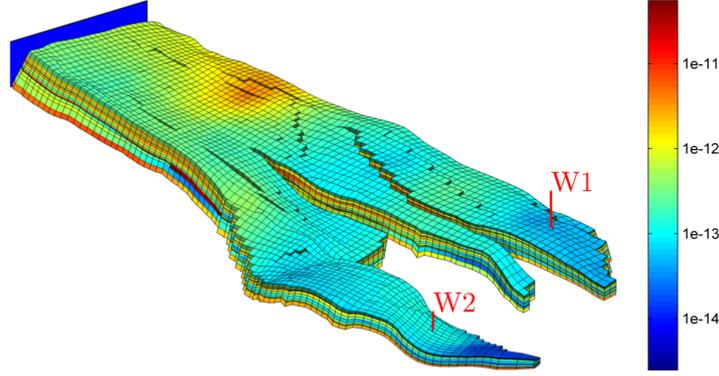


Figure 12: The grid and permeability for the Norne Field case along with producer wells and boundary conditions in red and blue respectively.

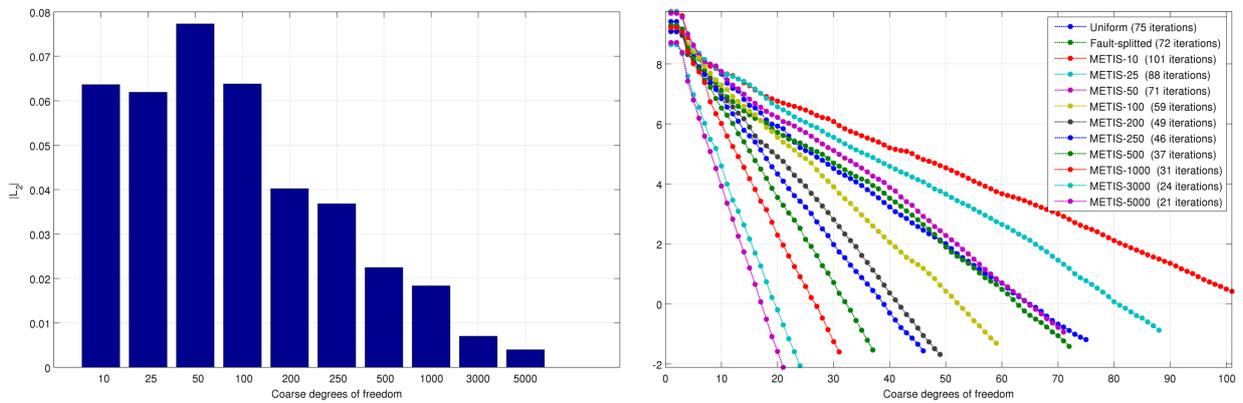


Figure 13: The L_2 error for the Norne Field for varying number of coarse blocks (left) and the convergence of the Ms-ILU-GMRES solver (right) are similar to the results for the SBED example, demonstrating the same robustness and convergence even in the presence of faults and realistic geometry.

the right plot shows convergence the full Ms-ILU-GMRES iterative method. As was observed in the previous example, the convergence of the method is very slow for coarse partitions. From the kink in the residual plot one can infer that the multiscale coarse operator contributes little to the convergence after a few cycles when using less than one hundred blocks and that reduction of the residual is left to the relatively inefficient ILU and GMRES iterations. However, each cycle has a very low computational cost and the first few cycles reduce the error significantly. Hence, one can use MsTPFA on a very coarse grid as an alternative to proxy models in workflows that do not require high simulation accuracy. On the opposite end, using more than one thousand blocks gives a very low initial error and rapid convergence, but the cost of each multiscale cycle is now significantly higher. Optimal efficiency is expected to lie somewhere in between, corresponding to an upscaling factor of a few hundred. Our prototype is not yet parallelized and fully optimized and this reason we refrain from a thorough study of efficiency.

The figure also reports the convergence history for the uniform partition and the uniform partition with splitting of blocks containing faults. For both, the reduction in the residual is comparable to that of the Metis partitions with approximately the same number of blocks for the first 5–10 iterations, but then decays significantly and follows the trend of the 50-block Metis partition. For the 200-block and 250-block Metis partitions, on the other hand, we observe no decay in the reduction rate, which demonstrates that using a permeability-adapted coarse partition gives a significantly better multiscale operator.

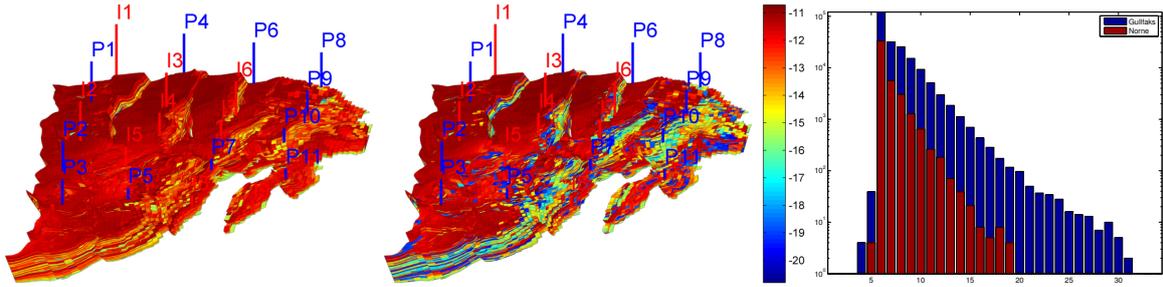


Figure 14: An example of a structurally complex reservoir model: the Gullfaks Field. The two left-most plots show the reservoir geometry with horizontal and vertical permeability (log-scale) and wells colored by type. The right plot shows a histogram of the number of faces per cell compared to the Norne model, which is significantly less structurally complex.

Table 4: Errors in the pressure solution for the Gullfaks model.

Coarse grid	No. blocks	MsTPFA		MsTPFA + Jacobi	
		L_2	L_∞	L_2	L_∞
Uniform	605	0.0581	0.3058	0.0518	0.2893
Metis	10	0.1049	0.3379	0.1001	0.2390
Metis	100	0.0519	0.3041	0.0429	0.2935
Metis	500	0.0324	0.2969	0.0262	0.2936
Metis	1000	0.0303	0.3016	0.0230	0.1769
Metis	10000	0.0206	0.1994	0.0107	0.1070

3.4 The Gullfaks Field

In the last example, we will demonstrate that the MsTPFA method is capable of handling models with a high structural complexity. To this end, we consider a simulation model of the Gullfaks Field, an oil and gas field in the Norwegian sector of the North Sea. The simulation model is represented using a $80 \times 100 \times 52$ corner-point grid in which 216 334 cells are active. The geometry of the Gullfaks model is highly irregular because of the large number of faults. Almost 44% of the cells have non-neighboring connections, and after the non-matching grid has been processed into a matching unstructured grid, the number of cell faces range from four to thirty-one as shown in Figure 14. The permeability varies several orders of magnitude and contains impermeable regions in the horizontal permeability, making this the absolutely most challenging test case in this paper.

To drive flow in the model, we consider a simple pattern of vertical wells distributed somewhat unrealistically throughout the model. The wells are controlled by bottom-hole pressure and the average injector pressure is set to 500 bars while the producers operate at an average pressure of 250 bars. The injectors and producers have per well variations in an order of 100 bars to make the pressure distribution more challenging to capture accurately.

As in the earlier models, we consider both a uniform partition as well as a variety of automatically generated partitions of increasing resolution to investigate systematic error reduction. The uniform coarse grid has local dimensions of $7 \times 7 \times 10$, giving an upscaling factor of approximately 350 once eroded and pinched cells have been removed. No special restrictions are added to the unstructured coarse partitions and we do not use correction functions. The results can be seen in Table 4 and Figure 15. As can be seen from the figures and the table, the pressure field is easily resolved by the MsTPFA method, which gives high-quality pressure solutions for arbitrary degrees of coarsening. Using increased resolution of the coarse partition results in higher solution quality and fewer iterations required for the Ms-GMRES-ILU solver.

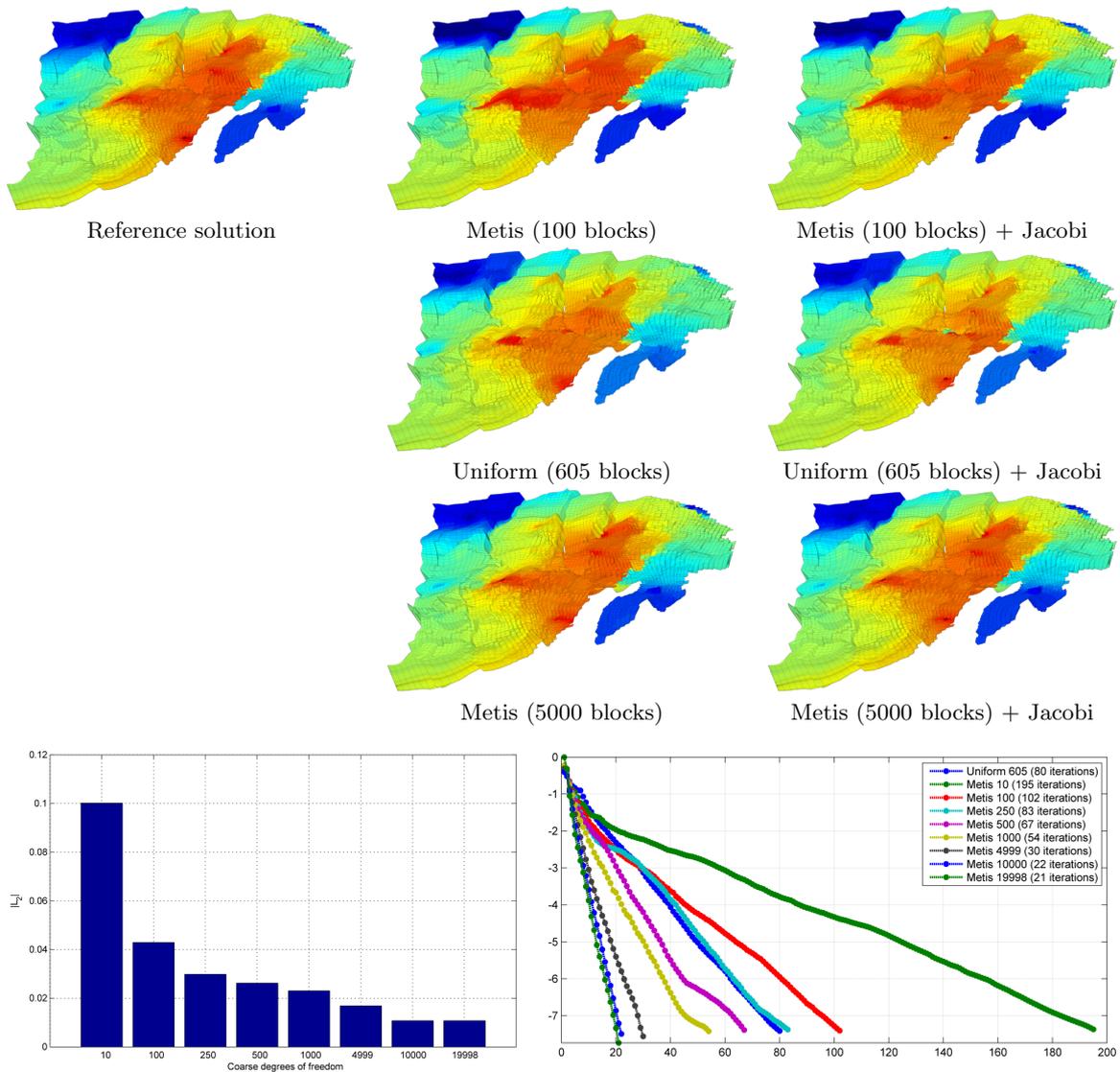


Figure 15: Multiscale solutions computed on different partitions for the Gullfaks model and the corresponding convergence plots.

4 Concluding remarks

In this paper, we have presented a novel multiscale two-point flux-approximation (MsTPA) method that relies on algebraic manipulations of discrete linear systems arising from standard fine-scale, finite-volume discretizations. The method is based on much of the same ideas as the multiscale finite-volume (MsFV) method and can utilize key technologies developed for this method, including correction functions and reconstruction of conservative fine-scale fluxes. Moreover, the method can be cast within iterative frameworks that have been developed for the MsFV method to systematically drive fine-scale residuals towards machine precision. The main advantage of such an iterative framework compared with multigrid methods is that the iteration can be aborted at any step to reconstruct conservative fine-scale fluxes.

Two important aspects, the construction of coarse partitions and the structure of the coarse-scale system, distinguish the MsTPFA from the MsFV method and makes the former more flexible and robust. To localize basis functions, the MsFV method uses reduced, compatible boundary conditions that are vulnerable to strong heterogeneities. As a result, the method forms multipoint coarse-scale systems that may produce strongly non-monotone solutions. The

MsTPFA method, on the other hand, uses a partition of unity derived by solving simple flow problems (tracer partitions) to form a coarse-scale system that is virtually a two-point discretization that computes almost monotone pressure solutions. This means that while the MsTFPA method is not as accurate as the MsFV method on smooth heterogeneities, it is significantly more robust on highly heterogeneous media with non-unit anisotropy and aspect ratios, which is particularly advantageous when the method is used as part of an iterative framework in combination with an inexpensive smoother. If a good starting point is provided for the smoother, high-quality solutions can be obtained within a few multiscale cycles even for problems with strong heterogeneities.

Secondly, while the MsFV method requires a primal-dual partition to compute basis/correction functions and generate coarse-scale systems, the MsTPFA method is based on a single coarse partition and hence shares the flexibility that has previously been reported for the multiscale mixed finite-element method. The use of a single coarse partition makes the MsTPFA method simple to implement on general unstructured grids. Extensive numerical tests on stratigraphic grids with erosion, pinch-outs, faults, and other types of degeneracies and nonconformities show that the method is very robust with regard to the choice of coarse partitions and can produce approximate solutions that resemble the fine-scale reference for almost arbitrary degrees of coarsening. The method may still have grid effects that reduce the quality of the approximate solutions. However, such grid effects, as well as monotonicity problems, can be significantly reduced by using coarse partitions that adapt to the geometry and heterogeneities of the medium. Applying such grids to the multiscale finite-volume method while simultaneously improving the coarse-scale operator is a subject of ongoing research.

Finally, although the details are not yet worked out, we believe that the tracer partition idea can be extended to create other prolongation operators corresponding to multipoint coarse-scale discretizations. For Cartesian grids, the obvious starting point is to subdivide faces into multiple patches and create partition-of-unity functions that reflect the desired multipoint connections. For unstructured coarse and fine grids, the extension is less obvious.

5 Acknowledgments

The authors thank Jostein R. Natvig, Halvor Møll Nilsen, and Stein Krogstad for helpful discussions. Parts of the research has been funded by Schlumberger Information Solutions and the Research Council of Norway under grant no. 226035. The authors would also like to thank Statoil (operator of the Norne field) and its license partners ENI and Petoro for the release of the Norne data.

References

- [1] 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 (electronic), 2004.
- [2] 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 (electronic), 2006.
- [3] J. E. Aarnes, S. Krogstad, and K.-A. Lie. Multiscale mixed/mimetic methods on corner-point grids. *Comput. Geosci.*, 12(3):297–315, 2008.
- [4] 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.
- [5] Z. Chen and T. Hou. A mixed multiscale finite element method for elliptic problems with oscillating coefficients. *Math. Comp.*, 72:541–576, 2003.

- [6] 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. Url: <http://www.spe.org/csp/>.
- [7] Y. Efendiev and T. Y. Hou. *Multiscale Finite Element Methods*, volume 4 of *Surveys and Tutorials in the Applied Mathematical Sciences*. Springer Verlag, 2009.
- [8] H. Hajibeygi. *Iterative multiscale finite volume method for multiphase flow in porous media with complex physics*. PhD thesis, ETH Zurich, 2011.
- [9] H. Hajibeygi, G. Bonfigli, M. A. Hesse, and P. Jenny. Iterative multiscale finite-volume method. *J. Comput. Phys.*, 227(19):8604–8621, 2008.
- [10] H. Hajibeygi and H. A. Jenny. Compositional multiscale finite-volume formulation. In *SPE Reservoir Simulation Symposium, The Woodlands, TX, USA, 18–20 February 2013*, 2013. SPE 163664-MS.
- [11] H. Hajibeygi and P. Jenny. Adaptive iterative multiscale finite volume method. *J. Comput. Phys.*, 230(3):628–643, 2011.
- [12] 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 (electronic), 2008.
- [13] 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.
- [14] 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.
- [15] S. Krogstad. A sparse basis POD for model reduction of multiphase compressible flow. In *2011 SPE Reservoir Simulation Symposium, The Woodlands, Texas, USA, 21–23 February 2011*, 2011.
- [16] 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.
- [17] 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.
- [18] 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.
- [19] S. H. Lee, H. Zhou, and H. A. Tchelepi. Adaptive multiscale finite-volume method for nonlinear multiphase transport in heterogeneous formations. *J. Comput. Phys.*, 228(24):9036 – 9058, 2009.
- [20] K. Lie, S. Krogstad, I. Ligaarden, J. Natvig, H. Nilsen, and B. Skaflestad. Open-source MATLAB implementation of consistent discretisations on complex grids. *Comput. Geosci.*, 16:297–322, 2012.
- [21] K.-A. Lie, S. Krogstad, I. S. Ligaarden, J. R. Natvig, H. M. Nilsen, and B. Skaflestad. Open source MATLAB implementation of consistent discretisations on complex grids. *Comput. Geosci.*, 16:297–322, 2012.
- [22] I. Lunati and P. Jenny. Multiscale finite-volume method for compressible multiphase flow in porous media. *J. Comput. Phys.*, 216(2):616–636, 2006.
- [23] I. Lunati and P. Jenny. Treating highly anisotropic subsurface flow with the multiscale finite-volume method. *Multiscale Model. Simul.*, 6(1):308–318 (electronic), 2007.
- [24] I. Lunati and P. Jenny. Multiscale finite-volume method for density-driven flow in porous media. *Comput. Geosci.*, 12(3):337–350, 2008.
- [25] I. Lunati and S. H. Lee. An operator formulation of the multiscale finite-volume method with correction function. *Multiscale modeling & simulation*, 8(1):96, 2009.
- [26] 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.
- [27] Metis – serial graph partitioning and fill-reducing matrix ordering, 2012. url: <http://glaros.dtc.umn.edu/gkhome/views/metis>.

- [28] O. Møyner. Multiscale finite-volume methods on unstructured grids. Master's thesis, Norwegian University of Science and Technology, 2012.
- [29] 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. SPE 163649-MS.
- [30] The MATLAB Reservoir Simulation Toolbox, version 2013a, Apr. 2013. <http://www.sintef.no/MRST/>.
- [31] J. R. Natvig, K.-A. Lie, B. Eikemo, and I. Berre. An efficient discontinuous Galerkin method for advective transport in porous media. *Adv. Water Resour.*, 30(12):2424–2438, 2007.
- [32] 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), 2011.
- [33] J. M. Nordbotten, E. Keilegavlen, and A. Sandvin. Mass conservative domain decomposition for porous media flow. In R. Petrova, editor, *Finite Volume Method-Powerful Means of Engineering Design*. InTech Europe, Rijeka, Croatia, 2012.
- [34] 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. SPE 163669-MS.
- [35] A. Sandvin, E. Keilegavlen, and J. M. Nordbotten. Auxiliary variables for 3d multiscale simulations in heterogeneous porous media. *J. Comput. Phys.*, 238(0):141–153, 2013.
- [36] Y. Wang, H. Hajibeygi, and H. A. Tchelepi. Algebraic multiscale linear solver for heterogeneous elliptic problems. In *ECMOR XIII - 13th European Conference on the Mathematics of Oil Recovery*, Biarritz, France, 10–13 September 2012. EAGE.
- [37] H. Zhou and H. A. Tchelepi. Operator-based multiscale method for compressible flow. *SPE J.*, 13(2):267–273, 2008.
- [38] H. Zhou and H. A. Tchelepi. Two-stage algebraic multiscale linear solver for highly heterogeneous reservoir models. *SPE J.*, 17(2):523–539, 2012.