

# Multiscale Mixed/Mimetic Methods on Corner-Point Grids

Jørg E. Aarnes (jaa@sintef.no), Stein Krogstad  
(steink@sintef.no) and Knut–Andreas Lie (knl@sintef.no)  
*SINTEF ICT, Dept. Applied Mathematics, P.O. Box 124 Blindern, N-0314 Oslo,  
Norway.*

**Abstract.** Multiscale simulation is a promising approach to facilitate direct simulation of large and complex grid-models for highly heterogeneous petroleum reservoirs. Unlike traditional simulation approaches based on upscaling/downscaling, multiscale methods seek to solve the full flow problem by incorporating subscale heterogeneities into local discrete approximation spaces. We consider a multiscale formulation based on a hierarchical grid approach, where basis functions with subgrid resolution are computed numerically to correctly and accurately account for subscale variations from an underlying (fine-scale) geomodel when solving the global flow equations on a coarse grid. By using multiscale basis functions to discretise the global flow equations on a (moderately-sized) coarse grid, one can retain the efficiency of an upscaling method, while at the same time produce detailed and conservative velocity fields on the underlying fine grid.

For pressure equations, the multiscale mixed finite-element method (MsMFEM) has shown to be a particularly versatile approach. In this paper we extend the method to corner-point grids, which is the industry standard for modelling complex reservoir geology. To implement MsMFEM, one needs a discretisation method for solving local flow problems on the underlying fine grids. In principle, any stable and conservative method can be used. Here we use a mimetic discretisation, which is a generalisation of mixed finite elements that gives a discrete inner product and allows for curved grid faces and polyhedral elements.

The coarse grid can in principle be any partition of the subgrid, where each coarse block is a connected collection of subgrid cells. However, we argue that when generating coarse grids, one should follow certain simple guidelines to achieve improved accuracy. We discuss partitioning in both index space and physical space, and suggest simple processing techniques.

The versatility and accuracy of the multiscale mixed methodology is demonstrated on two corner-point models: a small Y-shaped sector model and a complex model of a layered sedimentary bed. A variety of coarse grids, both violating and obeying the above mentioned guidelines, are employed. The MsMFEM solutions are compared with a reference solution obtained by direct simulation on the subgrid.

## 1. Introduction

Modern methods for 3D geological modelling and reservoir characterisation is leading industry to routinely build very large and detailed reservoir models; grid models of the subsurface geology currently range in size from 10 to 100 million cells and are growing. Moreover, the industry is moving away from a single “best effort” modelling and towards computations of many plausible realisations to assess uncertainty in geomodels. Due to the highly heterogeneous nature of porous rock formations, geomodels tend to have strongly irregular geometries and very complex hydraulic connectivities. Unfortunately, reservoir simulation technology has not kept pace with the development within geological modelling, and there is a steadily increasing gap between the level of detail seen in industrial geomodels and the capabilities of current flow simulators. In part, industry-standard simulators are incapable of simulating models with multi-million cells, and, in part, the underlying discretisation methods were not designed to handle the challenges posed by current geomodels: high aspect ratios, anisotropic properties, curved faces, degenerate cells, non-conformal grids, etc.

The traditional approach to overcome the gap in resolution between geomodels and simulation models is to use upscaling/downscaling between a detailed geological model and a coarser simulation model. In this paper we will discuss an alternative approach based on a multiscale formulation for computing pressure and flow velocities, where the full flow problem is solved by incorporating subscale heterogeneities into local discrete approximation spaces. This means that the global flow is computed on a coarse grid and fine-scale heterogeneity is accounted for through a set of generalised, heterogeneous basis functions. The basis functions are computed numerically by solving local flow problems (as is done in many flow-based upscaling methods), and when included in the coarse-grid equations, the basis functions ensure that the global equations are consistent with the local properties of the underlying differential operators. By using the multiscale basis functions to discretise the flow equations on a (moderately-sized) coarse grid, one can retain the efficiency of an upscaling method and at the same time produce detailed and conservative velocity fields.

The multiscale mixed finite-element formulation [16, 1, 3] has previously proved to be a particularly versatile approach for flow simulation on Cartesian grids in the sense that it produces more accurate and robust results than what is obtained by traditional upscaling approaches; see [2]. Here we take the important step of extending the methodology to the more complex corner-point grid format used in the oil industry. In particular, we seek to demonstrate that by using a multiscale

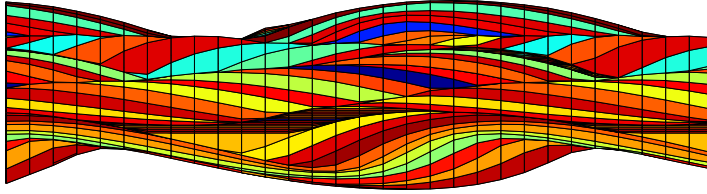


Figure 1. Side view in the  $xz$ -plane of corner-point grid with vertical pillars modelling a stack of sedimentary beds (each layer indicated by a different colour).

formulation one avoids the usual difficulties of resampling grid properties when moving from a fine to a coarse grid, and vice versa. In the multiscale formulation, the blocks in the coarse grid can, at least in principle, be chosen as an almost arbitrary connected collection of cells in the underlying fine grid. However, to obtain an accurate solution, the generation of coarse grid blocks should follow certain simple guidelines as discussed in [3] and in the following.

The flexibility with respect to grids is the major advantage of the multiscale mixed formulation over other multiscale and upscaling-downscaling methods. We refer the reader to [24] for a more thorough discussion of the advantages and disadvantages of the multiscale mixed method compared with other multiscale methods with respect to accuracy, robustness, efficiency, and ease of implementation.

The outline of the paper is as follows. We start by discussing the complexity of geological grid models and some of the challenges they pose for flow simulation in Section 2. In Section 3, we present our basic flow model and its mixed and mimetic formulation. We then introduce the multiscale method in Section 4 and discuss the generation of coarse grids and how the method can be viewed as an attractive alternative to traditional upscaling methods for complex grids. In Section 5, we present the mimetic subgrid discretisation technique that will be used to construct basis functions for the multiscale method. Finally, we present and discuss two numerical examples in Section 6 to highlight some of the properties of the multiscale method. In particular, we compare the multiscale velocity fields obtained on different coarse grids and solve a two-phase flow equation on the underlying fine grid to see how strongly discrepancies in the velocity fields impact flow characteristics.

## 2. Complexity of Reservoir Simulation Models

To model the geological structures of petroleum reservoirs, a standard approach is to introduce what is called a corner-point grid [26]. A corner-point grid consists of a set of hexahedral cells that are aligned

in a logical Cartesian fashion. One horizontal layer in the grid is then assigned to each sedimentary bed to be modelled. In its simplest form, a corner-point grid is specified in terms of a set of vertical or inclined pillars defined over an areal Cartesian 2D mesh in the lateral direction. Each cell in the volumetric corner-point grid is restricted by four pillars and is defined by specifying the eight corner points of the cell, two on each pillar. Figure 1 shows a side-view of such a corner-point grid. Notice the occurrence of degenerate cells with less than eight non-identical corners where the beds are partially eroded away. Some cells also disappear completely and hence introduce connections between cells that are not neighbours in the underlying logical Cartesian grid.

The corner-point format easily allows for degeneracies in the cells and discontinuities (fractures/faults) across faces. Hence, using the corner-point format it is possible to construct very complex geological models that match the geologist's perception of the underlying rock formations. Due to their many appealing features, corner-point grids are now an industry standard and the format is supported in most commercial software for reservoir modelling and simulation. The original corner-point format has been extended in several directions, for instance to allow intersection of two straight pillars in the shape of the letter Y. Similarly, the pillars may be piecewise polynomial curves, resulting in what is sometimes called S-faulted grids.

Using geological models as input to flow simulation introduces several numerical difficulties. First of all, typical reservoirs extend several hundred or thousand metres in the lateral direction, but the zones carrying hydrocarbon may be just a few tens of metres in the vertical direction and consist of several layers with different rock properties. Geological models therefore have grid-cells with very high aspect ratios and often the majority of the flow in and out of a cell occurs across the faces with the smallest area. Similarly, the possible presence of strong heterogeneities and anisotropies in the permeability fields typically introduces large condition numbers in the discretised flow equations. These difficulties are observed even for grid models consisting of regular hexahedral cells.

The flexible cell geometry of the corner-point format introduces additional difficulties. First of all, since each face of a grid cell is specified by four (arbitrary) points, the cell interfaces in the grid will generally be bilinear surfaces and possibly be strongly curved. Secondly, corner-point cells may have zero volume, which introduces coupling between non-neighbouring cells and gives rise to discretisation matrices with complex sparsity patterns. Finally, the presence of degenerate cells, in which the corner-points collapse in pairs, means that the cells will generally be polyhedral and possibly contain both triangular and bi-

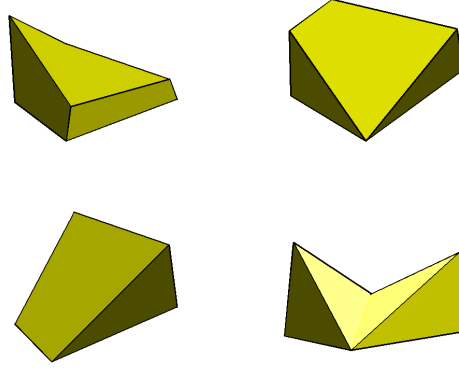


Figure 2. Examples of deformed and degenerate hexahedral cells arising in corner-point grid models.

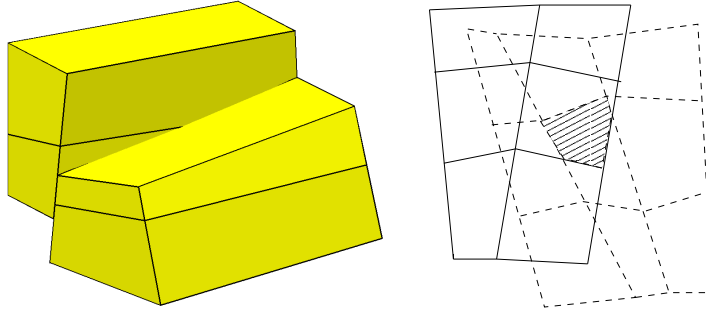


Figure 3. Two examples of fault surface in a three-dimensional model with non-matching interfaces across the faults. (Left) Three-dimensional view. (Right) Two-dimensional view, where the shaded patch illustrates a “sub-interface” on the fault surface.

linear faces (see Figure 2). This calls for a very flexible discretisation that is not sensitive to the geometry of each cell or the number of faces and corner points. Moreover, the discretisation should be capable of handling non-conforming grids. Non-conforming grids arise, using the corner-point format, in fault zones where a displacement along a hyperplane has occurred, see Figure 3. Another consequence is that generation of coarse grids to be used e.g., in upscaling is generally difficult; see the discussion in Sections 4.1 and 4.2.

### 3. Mixed Formulation of Elliptic Model Problem

Let  $\Omega \subset \mathbb{R}^3$  be a polyhedral domain, and let  $n$  be the outward unit normal on  $\partial\Omega$ . As a prototype flow problem, we consider the following

second-order elliptic equation:

$$\begin{aligned} v &= -K \nabla p, & x \in \Omega, \\ \nabla \cdot v &= f, & x \in \Omega, \\ v \cdot n &= 0, & x \in \partial\Omega, \end{aligned} \quad (1)$$

where we require, for compatibility, that  $\int_{\Omega} f \, dx = 0$ . Since this is a pure Neumann boundary-value problem,  $p$  is defined only up to an arbitrary constant. An extra constraint, such as  $\int_{\Omega} p \, dx = 0$ , must therefore be added to close the system. Henceforth we refer to  $p$  as the pressure and  $v$  as the velocity.

To discretise (1), we introduce an arbitrary partition of  $\Omega$  into polyhedral-like cells  $\mathcal{T} = \{T\}$ , and define the following function spaces:

$$\begin{aligned} H^{\text{div}}(T) &= \{v \in L^2(T)^d : \nabla \cdot v \in L^2(T)\}, \\ H_0^{\text{div}}(\mathcal{T}) &= \{v \in H^{\text{div}}(\cup_{T \in \mathcal{T}} T) : v \cdot n = 0 \text{ on } \partial\Omega\}, \\ H_0^{\text{div}}(\Omega) &= H_0^{\text{div}}(\mathcal{T}) \cap H^{\text{div}}(\Omega). \end{aligned}$$

Furthermore, introduce the following bilinear forms:

$$b(\cdot, \cdot) : H_0^{\text{div}}(\mathcal{T}) \times H_0^{\text{div}}(\mathcal{T}) \rightarrow \mathbb{R}, \quad b(u, v) = \sum_{T \in \mathcal{T}} \int_T u \cdot K^{-1} v \, dx \quad (2.1)$$

$$c(\cdot, \cdot) : H_0^{\text{div}}(\mathcal{T}) \times L^2(\Omega) \rightarrow \mathbb{R}, \quad c(v, p) = \sum_{T \in \mathcal{T}} \int_T p \nabla \cdot v \, dx \quad (2.2)$$

$$d(\cdot, \cdot) : H_0^{\text{div}}(\mathcal{T}) \times H^{\frac{1}{2}}(\partial\mathcal{T}) \rightarrow \mathbb{R}, \quad d(v, \pi) = \sum_{T \in \mathcal{T}} \int_{\partial T} \pi v \cdot n_T \, ds \quad (2.3)$$

$$(\cdot, \cdot) : L^2(\Omega) \times L^2(\Omega) \rightarrow \mathbb{R}, \quad (p, q) = \int_{\Omega} pq \, dx. \quad (2.4)$$

Here  $n_T$  is the unit normal on  $\partial T$  pointing outward and  $H^{\frac{1}{2}}(\partial\mathcal{T})$  is the space spanned by the traces on  $\partial\mathcal{T}$  of functions in  $H^1(\mathcal{T})$ . We will now use these bilinear forms to develop a discretisation based on a mixed formulation. The mixed formulation will later be used for two purposes. Primarily, it will be used to formulate the multiscale mixed finite-element method to be introduced in Section 4. Secondly, the mixed formulation will be used in Section 5 to develop the mimetic finite-difference discretisation scheme for the subgrid problems arising in the multiscale formulation.

### 3.1. MIXED FORMULATION

In the general mixed formulation of (1), one seeks a pair of functions  $(p, v) \in L^2(\Omega) \times H_0^{\text{div}}(\Omega)$  such that

$$\begin{aligned} b(u, v) - c(u, p) &= 0, & \forall u \in H_0^{\text{div}}(\Omega), \\ c(v, q) &= (f, q), & \forall q \in L^2(\Omega). \end{aligned} \quad (3)$$

In mixed FEMs, the system (3) is discretised by replacing  $L^2(\Omega)$  and  $H_0^{\text{div}}(\Omega)$  with finite-dimensional subspaces  $U$  and  $V$ . One then seeks approximations  $p \in U$  and  $v \in V$  that satisfy (3) for all test functions  $q \in U$  and  $u \in V$ . A typical example of subspaces is the lowest-order Raviart–Thomas functions [27], for which  $U$  is the space of piecewise constants and  $V$  has piecewise linear components.

A mixed FEM solution  $(p, v)$  of (3) in  $U \times V$  is a saddle-point of the following Lagrange functional:

$$L(v, p) = \frac{1}{2}b(v, v) - c(v, p) + (p, f).$$

This means that  $L(v, q) \leq L(v, p) \leq L(u, p)$  for all  $u$  and  $q$ . Consequently, mixed FEM discretisations lead to indefinite linear systems. Indefinite systems require special linear solvers and are often considered hard to solve. Next, we therefore introduce an alternative formulation of the mixed method that will give a positive definite discrete system and thereby simplify the computation of a discrete solution.

### 3.2. HYBRID FORMULATION

In a hybrid formulation [11], the need to solve a saddle-point problem is avoided by using Lagrange multipliers. The hybrid formulation is therefore sometimes called the Lagrange multiplier technique. For (1), the hybrid formulation is obtained by replacing the mixed formulation (3) with the following problem: find  $(v, p, \pi) \in H_0^{\text{div}}(\mathcal{T}) \times L^2(\Omega) \times H^{\frac{1}{2}}(\partial\mathcal{T} \setminus \partial\Omega)$  such that

$$\begin{aligned} b(u, v) - c(u, p) + d(u, \pi) &= 0, & \forall u \in H_0^{\text{div}}(\mathcal{T}), \\ c(v, q) &= (f, q), & \forall q \in L^2(\Omega), \\ d(v, \mu) &= 0, & \forall \mu \in H^{\frac{1}{2}}(\partial\mathcal{T} \setminus \partial\Omega). \end{aligned} \quad (4)$$

Here  $\partial\mathcal{T} = \cup_{T \in \mathcal{T}} \partial T$  and the new unknowns  $\pi$  correspond to pressures at element faces. To discretise (4), one selects finite-dimensional subspaces  $V \subset H_0^{\text{div}}(\mathcal{T})$ ,  $U \subset L^2(\Omega)$ , and  $\Pi \subset H^{\frac{1}{2}}(\partial\mathcal{T} \setminus \partial\Omega)$ , and seeks  $(v, p, \pi) \in V \times U \times \Pi$  such that (4) holds for all  $(u, q, \mu) \in V \times U \times \Pi$ . Observe that in this approach one departs from the constraint  $V \subset H^{\text{div}}(\Omega)$ . Instead flux continuity is enforced through the Lagrange multipliers. In other words, the idea is to first remove the constraint that the normal velocity must be continuous across element faces and integrate (1) to get a weak form containing jump terms at block boundaries. Continuity of the normal component is then reintroduced by adding an extra set of equations, where the pressure  $\pi$  at the interfaces plays the role of Lagrange multipliers. This procedure does not change  $v$ ,

nor  $p$ , but enables the recovery of pressure values at element faces, in addition to inducing the desired change in structure of the linear system resulting from the discretisation.

### 3.3. MIMETIC FORMULATION

Recent mimetic finite-difference methods (FDMs) [15, 14] can be seen as finite-difference counterparts of mixed FEMs. The discretisation in a standard mixed method is introduced by picking discrete function spaces  $U$  and  $V$  and using numerical quadrature to evaluate the integrals over cell volumes and cell faces in the variational formulation (3). In a mixed method, the discretisation is introduced already in the variational formulation in the form of a discrete inner product. Mathematically, this means that the subspace  $V$  in  $H_0^{\text{div}}(\Omega)$  is replaced by a discrete subspace of  $L^2(\partial\mathcal{T})$ , and the associated bilinear form  $b(\cdot, \cdot)$  is replaced by a bilinear form that acts on  $L^2(\partial\mathcal{T}) \times L^2(\partial\mathcal{T})$ . Hence, instead of seeking an unknown velocity field  $v$  defined over each element  $T$ , one seeks a set of fluxes defined over the cell faces  $\partial T$ .

The most apparent advantage of mimetic FDMs relative to mixed FEMs, is the ability to handle complex polyhedral grids cells (with strongly curved faces [14]) in a straightforward manner. Stability and optimal convergence of mimetic FDMs were established by [12, 13] for very general grids. For flow in porous media, polyhedral cells arise frequently in geological models when conforming hexahedral corner-point grids are deformed to model geological phenomena like erosion and pinch-outs. Similarly, for corner-point grids containing faults and throws, one can subdivide grid faces and turn the non-conforming corner-point grids into conforming polyhedral grids. However, in the current paper we only consider conforming corner-point grids. We will return to the discrete formulation of mimetic FDM in Section 5.1 and introduce a particular variant [15] that closely resembles the lowest-order Raviart–Thomas MFEM for hexahedral cells.

In the hybrid formulation of a mimetic FDM, one only needs to replace  $V$  with a discrete subspace  $M \subset L^2(\partial\mathcal{T})$ , and  $b(\cdot, \cdot)$  with a bilinear form  $m(\cdot, \cdot)$  that acts on  $L^2(\partial\mathcal{T}) \times L^2(\partial\mathcal{T})$ .

### 3.4. DISCRETE FORMULATIONS

The algebraic systems that arise from a mixed finite-element or a mimetic finite-difference discretisation of (1) take the same general form, whether we employ a standard or a hybrid formulation. In this paper we employ a hybrid formulation only. The hybrid formulation



(4) gives rise to linear systems of the following form:

$$\begin{bmatrix} \mathbf{B} & -\mathbf{C}^T & \mathbf{\Pi}^T \\ \mathbf{C} & \mathbf{0} & \mathbf{0} \\ \mathbf{\Pi} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{v} \\ \mathbf{p} \\ \pi \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{f} \\ \mathbf{0} \end{bmatrix}. \quad (5)$$

In the multiscale mixed FEM and the mimetic FDM considered herein, the approximation space  $U$  consists of functions that are cell-wise constant

$$U = \text{span}\{\chi_m : T_m \in \mathcal{T}\}, \quad \chi_m(x) = \begin{cases} 1, & \text{if } x \in T_m, \\ 0, & \text{otherwise.} \end{cases}$$

Consequently, for  $v \in H_0^{\text{div}}(\mathcal{T})$  and  $p \in U$ , we have

$$c(v, p) = \sum_m p_m \int_{T_m} \nabla \cdot v \, dx = \sum_m p_m \int_{\partial T_m} v \cdot n_T \, ds.$$

This shows that  $c(v, p)$  can be evaluated without having an explicit representation of the velocity inside each cell, only values on the cell boundaries are needed. Clearly, this is the case also for  $d(v, \pi)$  for any  $\pi \in \Pi \subset H^{\frac{1}{2}}(\partial\mathcal{T})$ . We therefore assume that  $\Pi$  consists of functions that are constant on each grid face  $\gamma_j^i = \partial T_i \cap \partial T_j$ ; that is,

$$\Pi = \text{span}\{\pi_j^i : |\gamma_j^i| > 0\}, \quad \pi_j^i(x) = \begin{cases} 1, & \text{if } x \in \gamma_j^i, \\ 0, & \text{otherwise.} \end{cases}$$

To derive a discrete formulation, it only remains to define an approximation space  $V$  for velocity in the multiscale mixed FEM, or alternatively a discrete approximation space  $M \subset L^2(\partial\mathcal{T})$  and a corresponding bilinear form  $m(\cdot, \cdot)$  to be used in the mimetic FDM. This will be done in the subsequent sections. For now, we only assume that a “velocity” basis function  $\psi_i^m$  is associated with each face  $\gamma_i^m$  of every grid cell  $T_m$ . Thus, whereas we have one Lagrange multiplier for each “interface”, we have a velocity basis function for each face of every grid cell. Note that  $\psi_i^m$  lives in different spaces, depending on whether a mixed FEM or a mimetic FDM is used.

The matrices  $\mathbf{C}$  and  $\mathbf{\Pi}$  in (5) are now given by

$$\mathbf{C} = [c(\psi_i^m, \chi_n)] \quad \text{and} \quad \mathbf{\Pi} = [d(\psi_k^m, \mu_j^i)],$$

and the matrix  $\mathbf{B}$  is given by

$$\mathbf{B} = [b(\psi_i^m, \psi_j^n)], \quad \text{or} \quad \mathbf{B} = [m(\psi_i^m, \psi_j^n)]$$

for the mixed FEM and the mimetic FDM, respectively.

The hybrid system (5) is indefinite, but  $b(\psi_i^m, \psi_j^n)$  and  $m(\psi_i^m, \psi_j^n)$  are nonzero only if  $n = m$ . Hence, by numbering the basis functions  $\psi_i^m$  on a cell-by-cell basis, the matrix  $\mathbf{B}$  becomes block diagonal, and a Schur-complement reduction with respect to  $\mathbf{B}$  can be performed to obtain the following positive-definite system for  $\mathbf{p}$  and  $\pi$ :

$$\begin{bmatrix} \mathbf{D} & -\mathbf{F}^T \\ \mathbf{F} & -\Pi\mathbf{B}^{-1}\Pi^T \end{bmatrix} \begin{bmatrix} \mathbf{p} \\ \pi \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{0} \end{bmatrix}, \quad (6)$$

where  $\mathbf{D} = \mathbf{CB}^{-1}\mathbf{C}^T$  and  $\mathbf{F} = \Pi\mathbf{B}^{-1}\mathbf{C}^T$ .

Next, we use the fact that  $c(\psi_i^m, \chi_n) = 0$  for  $n \neq m$  to deduce that  $\mathbf{D}$  is a diagonal matrix. A Schur-complement reduction with respect to  $\mathbf{D}$  can therefore be performed to yield the following positive-definite system for  $\pi$  alone:

$$\mathbf{S}\pi = \mathbf{F}\mathbf{D}^{-1}\mathbf{f}, \quad \text{where} \quad \mathbf{S} = \Pi\mathbf{B}^{-1}\Pi^T - \mathbf{F}\mathbf{D}^{-1}\mathbf{F}^T. \quad (7)$$

Once  $\pi$  is computed, one can easily compute  $\mathbf{p}$  and  $\mathbf{v}$  by solving a diagonal and a block-diagonal system, respectively.

#### 4. A Multiscale Mixed Finite-Element Method

Elliptic (or parabolic) equations used to model pressure and velocity in porous media flow applications have velocity solutions that often exhibit a multiscale structure. To solve such equations, Hou and Wu [19] introduced the idea of using a finite-element formulation with special finite-element basis functions that are constructed to be adaptive to local properties of the differential operator. Their method was called the multiscale finite-element method (MsFEM) and was able to generate solutions that reflect important subscale characteristics of the coefficients in the elliptic equation. However, MsFEM did not produce (locally) mass-conservative solutions. By introducing a mixed formulation, Chen and Hou [19] obtained a method that gave mass-conservative velocity fields on the coarse discretisation grid and also on the underlying subgrid in coarse blocks not containing sources. This method will herein be referred to as the multiscale mixed FEM (MsMFEM). In this paper we will use a variant of MsMFEM developed by Aarnes and coworkers [1, 2, 3] that provides a mass-conservative velocity field on the entire underlying subgrid. We note that MsMFEM is related to *subgrid upscaling* introduced by Arbogast [10, 7, 8]. The connection between these two approaches has been addressed in [9]. Other numerical subgrid methods (based on the variational multiscale method [20]) include [23, 25]. Distinctly different, but related approaches include the

multiscale finite-volume method by Jenny, Lee and Tchelepi [21, 22], and the nested-gridding method by Gautier, Blunt and Christie [18].

To formulate MsMFEM, let  $\mathcal{B} = \{B\}$  be a grid where each grid block  $B$  is a connected union of grid cells in an underlying subgrid  $\mathcal{T}$ . The grid  $\mathcal{B}$  will be referred to as the *coarse grid*, and the subgrid  $\mathcal{T}$  will be referred to as the *fine grid*. Moreover, to clearly distinguish between cells in the coarse grid and cells in the fine grid, we will henceforth use block to denote a cell in the coarse grid.

For each interface  $\Gamma_{ij} = \partial B_i \cap \partial B_j$  in the coarse grid, we assign a corresponding basis function  $\Psi_{ij}$  designed to embody the local impact of subgrid variations in the permeability  $K$ . This basis function is related to an unknown function  $\Phi_{ij}$  through the gradient relation  $\Psi_{ij} = -K\nabla\Phi_{ij}$ . The function  $\Phi_{ij}$  is supported in  $\Omega_{ij} = B_i \cup \Gamma_{ij} \cup B_j$  and is obtained by solving (numerically) a local elliptic problem

$$\Psi_{ij} \cdot n_{ij} = 0 \text{ on } \partial\Omega_{ij}, \quad \nabla \cdot \Psi_{ij} = \begin{cases} f_i(x), & \text{for } x \in B_i, \\ -f_j(x), & \text{for } x \in B_j. \end{cases} \quad (8)$$

Here  $n_{ij}$  is the outward-pointing unit normal to  $\Omega_{ij}$ , and the source terms  $f_i$  and  $f_j$  are given by

$$f_i(x) = \frac{w_i(x)}{\int_{B_i} w_i(x) dx}, \quad w_i = \begin{cases} f, & \text{if } \int_{B_i} f dx \neq 0, \\ \text{trace}(K), & \text{otherwise.} \end{cases} \quad (9)$$

The source terms  $\{f_i\}$  are defined as in (9) for the following reasons:

1. With this definition, the basis function  $\Psi_{ij}$  forces unit flux across  $\Gamma_{ij}$ ; that is,  $\int_{\Gamma_{ij}} \Psi_{ij} \cdot n ds = 1$ , where  $n$  is the unit normal of  $\Gamma_{ij}$  pointing into  $B_j$ . This implies that the velocity solution  $\{v_{ij}\}$  gives the fluxes across the respective coarse-grid interfaces.
2. If a conservative method is used to compute basis functions, the velocity  $v = \sum v_{ij} \Psi_{ij}$  conserves mass on the subgrid  $\mathcal{T}$ .
3. Choosing special source terms in blocks containing nonzero source terms allows the method to model radial flow on the subgrid scale around point or line sources, such as wells in oil-reservoirs.
4. By letting  $f_i$  scale according to the trace of  $K$ , as in (9), one can to some extent avoid unnaturally high velocities across flow barriers, see [3].

For regular hexahedral blocks, homogeneous permeability, and no source terms, the multiscale basis functions simplify to the first-order Raviart–Thomas (RT0) basis functions. In addition to capturing the influence

of subgrid heterogeneity, the multiscale basis functions provide a generalisation of the RT0 basis functions to general polyhedral elements.

The main computational effort when solving (1) using MsMFEM is spent on constructing the basis functions  $\{\Psi_{ij}\}$ . The work associated with this task alone is proportional to the amount of work spent on solving the full problem on the fine grid using the best linear solvers available, like state-of-the-art algebraic multigrid methods. This illustrates that MsMFEM does not necessarily reduce the computational cost considerably compared to a fine-grid solver for a single pressure solve. However, MsMFEM offers significant computational savings for multiphase flow problems. Indeed, for two-phase flow simulations, where the pressure equation needs to be solved repeatedly, it has been demonstrated that the basis functions need to be computed only once, or updated infrequently [1, 22, 24]. This means that the main computational task is related to solving the global coarse-grid system, which is significantly less expensive than solving the full fine-grid system. It should also be noted each basis function  $\psi_{ij}$  can be computed independently, which means that the computation of basis functions is a so-called embarrassingly parallel task. Significant speedup of MsMFEM should therefore be expected for parallel implementations.

#### 4.1. GENERATION OF COARSE GRIDS

The above formulation of basis functions is *very* flexible with respect to the geometry and topology of both the coarse grid and the underlying fine grid. This is a major advantage of the multiscale *mixed* formulation. A bit simplified, the grid flexibility can be stated as follows: given an appropriate solver for the local flow problems on a particular type of fine grids, the multiscale method can be formulated and basis functions can be computed on any coarse grid where each grid block consists of an arbitrary collection of connected fine-grid cells.

To illustrate, consider a small model where  $\Omega$  is defined as the union of the three blocks depicted in Figure 4. Although these blocks are stacked on top of each other, each pair of blocks has a common interface. Thus, in the multiscale formulation we construct three basis functions for this set of blocks, one for each pair depicted in Figure 4.

In principle, any conservative numerical method may be used to construct the basis functions. However, one must also be able to evaluate the  $b(\cdot, \cdot)$  inner product between two basis functions (see (2.1)). Hence, if the numerical method used to compute the basis function only provides fluxes over cell faces (i.e., is based on a finite-volume or finite-difference method), one must have means to interpolate velocities internal to the cells. Alternatively, one can use an approximate inner

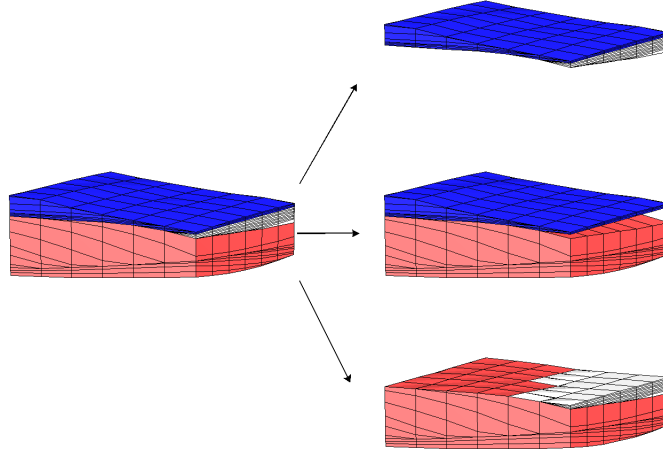


Figure 4. A three-block domain and the corresponding subdomains constituting the support of the resulting MsMFEM basis functions.

product like the one used in the mimetic formulation discussed in Section 3.3, for which  $b(\cdot, \cdot)$  is approximated with an inner product that only involves interface fluxes.

A mass-conservative subscale solver guarantees that the coarse-scale and fine-scale velocities will be conservative. In fact, the fine-scale fluxes can be used to define conservative fluxes on *any* grid with blocks consisting of simply-connected unions of cells from the fine grid. This means that the grid used by an accompanying transport solver does not have to coincide with the coarse grid in the multiscale method or with the underlying fine grid. The multiscale method can therefore easily be combined with, e.g., transport solvers using adaptive mesh refinement.

Extensive tests, some of which are reported in Section 6, show that the accuracy of MsMFEM is generally not very sensitive to the shape of the blocks. In fact, accurate results are obtained for grids containing blocks with rather 'exotic' shapes, see e.g., Figure 8 and [3]. This means that the process of generating a coarse simulation grid from a complex geomodel can be greatly simplified, especially when the fine grid is fully unstructured or has geometrical complications due to faults, throws, and eroded cells. However, MsMFEM does have some limitations, as identified in [3]. Here it was observed that barriers (low-permeable obstacles) may cause inaccurate results unless the coarse grid adapts to the barrier structures. In addition it was demonstrated that MsMFEM in its present form has limited ability to model bidirectional flow across coarse-grid interfaces; fine-grid fluxes at coarse-grid interfaces in the reconstructed flow field will usually go in the same direction.

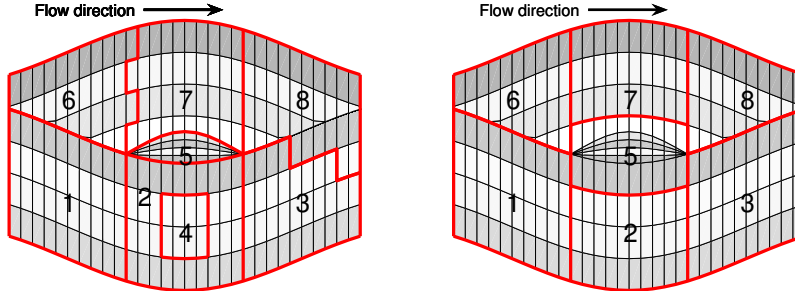


Figure 5. Illustration of some of the guidelines for choosing a good coarse grid. In the left plot, all blocks except for Block 1 violate at least one of the guidelines each. In the right plot, the blocks have been improved at the expense of more couplings in the coarse-grid system.

As a remedy for the limitations identified in [3], it is possible to exploit global information (e.g., from an initial fine-scale pressure solve) when constructing the basis functions [1]. However, with the present definition of the basis functions, our experience indicates that accurate results are obtained if the coarse grid obeys certain guidelines; see the left plot in Figure 5 for illustrations:

1. The coarse grid should preferably minimise the occurrence of bidirectional flow across coarse-grid interfaces. Examples of grid structures that increase the likelihood for bidirectional flow are:
  - Coarse-grid faces with (highly) irregular shapes, like the 'saw-tooth' faces between Blocks 6 and 7 and Blocks 3 and 8.
  - Blocks that do not contain source terms and have only one neighbour, like Block 4. (A simple remedy for this is to split the interface into at least two sub-faces, and define a basis function for each sub-face.)
  - Blocks having interfaces only along and not transverse to the major flow directions, like Block 5. (To represent flow in a certain direction, there must be at least one non-tangential face that defines a basis function in the given flow direction.)
2. Blocks and faces in the coarse grid should follow geological layers whenever possible. This is not fulfilled for Blocks 3 and 8.
3. Blocks in the coarse-grid should adapt to flow obstacles (shale barriers, etc.) whenever possible; see [3].

In addition, to enhance the efficiency of the method, one should try to keep the number of connections between coarse-grid blocks as low

as possible (to minimise the bandwidth of the coarse-scale system), and avoid having too many small blocks (this increases the dimension of the coarse-scale system, but does not necessarily improve accuracy significantly).

In the right plot of Figure 5, we have used the guidelines above to improve the coarse grid from the left plot. In particular, we have increased the size of Block 5 to homogenise the block volumes and introduce basis functions in the major flow direction for this block. In doing so, we increase the number of couplings from nine to twelve (by removing the coupling between Blocks 2 and 4 and introducing extra coupling among Blocks 1, 3, 5, 6, and 8). In general it may be difficult to obtain an 'optimal' coarse grid, since guidelines may be in conflict with each other. On the other hand, this is seldom necessary, since the multiscale formulation is relatively robust with respect to the choice of coarse grids. We will return to some of these issues in Section 6.2, where we discuss the generation of coarse grids for a real industry model.

#### 4.2. ROBUST ALTERNATIVE TO UPSCALING

Multiple direct solutions of the elliptic (or parabolic) pressure equation, as required in reservoir simulation, is generally infeasible on multimillion cell geomodels with current simulators. Hence, rather than using the original geomodel as input, current simulators normally take as input coarsened and simplified grid-models derived through an upscaling process. Modelling the flow on a coarser grid is sufficient only if the upscaled model accounts for the influence of dominant fine-scale features. Unfortunately, because upscaling methods do not offer sufficient resolution, they often fail to capture important subscale features, such as narrow high-flow channels or shale barriers. Multiscale methods, on the other hand, generally model subscale features adequately, even though the global flow equations are solved on a coarse grid only.

Multiscale methods, such as MsMFEM and the multiscale finite-volume method [21], may therefore be viewed as a robust alternative to upscaling. Indeed, if one avoids recomputing basis functions at each time step, the computational cost of a multiscale method is comparable to that of standard flow-based upscaling methods. However, the flexibility to handle grids with very general block geometries is currently a unique feature for MsMFEM. We consider this feature to be a very important advantage. For instance, it allows for automated strategies for generation of coarse grids. In particular, by allowing geomodels to be used as direct input for simulations, MsMFEM opens up for more efficient reservoir modelling and simulation workflows.

As opposed to MsMFEM, most upscaling procedures for the pressure equation are devised for coarse grids having hexahedral blocks. For grids with general polyhedral coarse blocks like in Figures 4 and 8, the choices of upscaling methods are limited, in particular since it is difficult to specify natural flow directions and appropriate boundary conditions. There exist upscaling techniques that are applicable to grids with general polyhedral blocks (e.g., simple averaging approaches), but these usually give less accurate results. Indeed, averaging approaches are in general very crude and create upscaled models where all information about the subscale heterogeneous structures is lost. The main limitation, however, is the lack of a robust numerical method for coarse grids with general polyhedral geometries. Multi-point finite-difference methods, for instance, are not applicable to grids with a wide variety of block geometries. A generalisation of the two-point finite-difference (finite-volume) method may be applied, but will generally give very poor results. Indeed, this method is convergent only for so-called  $K$ -orthogonal grids, meaning that the connections between cell centres are  $K$ -perpendicular to the cell faces.

Hence, in conclusion, upscaling methods for the pressure equation combined with standard numerical methods usually do not provide sufficiently accurate pressure and velocity solutions on coarsened grids having complex block geometries. Thus, to remedy grid limitations associated with upscaling, it is common to spend a significant effort on constructing and improving coarsened simulation grids. In fact, the process of generating simulation grids is generally the most time-consuming part of the upscaling phase. Using MsMFEM, the generation of coarsened grids is greatly simplified. MsMFEM applies and usually gives adequate accuracy for any given partitioning of the fine-grid (e.g., geomodel) into a collection of connected coarse blocks, provided that one has a numerical method for computing the basis functions on the fine grid and means to evaluate the  $b(\cdot, \cdot)$  inner product between two basis functions (approximately). However, improved accuracy is to be expected if the grid generation follows the guidelines from Section 4.1. By allowing grid blocks to consist of a collection of cells in the fine grid, one avoids resampling of geological data, which is a complicating part of standard grid-generation procedures. In summary, we believe that MsMFEM is both a robust and flexible alternative (or even a replacement) of upscaling methods on complex grids. We will make an effort to demonstrate this for corner-point grids in Section 6.



## 5. Subgrid Discretisation Methods

As pointed out in the previous section, MsMFEM relies on a pressure solver on the underlying grid to compute basis functions by solving the local flow problems (8). To this end, there are several options available, as outlined below. However, the choice of a good method is complicated by the fact that real-life corner-point grids may be highly irregular, as illustrated in Figure 2.

**Two-point FDMs:** Most commercial reservoir simulators use traditional finite-difference methods like the two-point flux approximation scheme. These methods were not designed to cope with the type of corner-point grid models that are built today using modern geomodelling tools. For instance, routinely generated corner-point grid models are seldom  $K$ -orthogonal. Hence, if one is interested in *accurate* solutions, two-point schemes should be avoided.

**Multi-point FDMs** [5, 6, 17] are more advanced finite-difference methods that amend the shortcomings of two-point scheme. These methods are accurate, but are unfortunately hard to implement for general corner-point grids, especially if the grid is non-conforming with non-matching faces.

**Mixed FEMs** are more accurate than two-point schemes and generally quite robust. However, since cells with five to seven corners are not diffeomorphic to the unit cube, one cannot use a straightforward mixed FEM relying on a transformation from hexahedral cells back to the unit cube. Instead, one must introduce special reference elements and corresponding Piola transforms for each of the degenerate cases, and this complicates the implementation of a mixed FEM considerably.

To overcome this problem, one can introduce a conforming tetrahedral subdivision of the corner-point grid and use the lowest-order Raviart–Thomas MFEM to discretise (8). To partition the grid, one can simply subdivide each corner-point cell as depicted in Figure 6, and then remove all tetrahedrons with zero volume. This approach does not extend easily (without introducing mortars) to non-conforming grids with non-matching faces.

**Mimetic FDMs** can be seen as finite-difference counterparts of mixed FEMs. The methods are very flexible with respect to cell geometries and easy to formulate for general polyhedral cells. This opens up for a natural way of treating degenerate cells with less than six faces and non-conforming grids with non-matching faces. However,

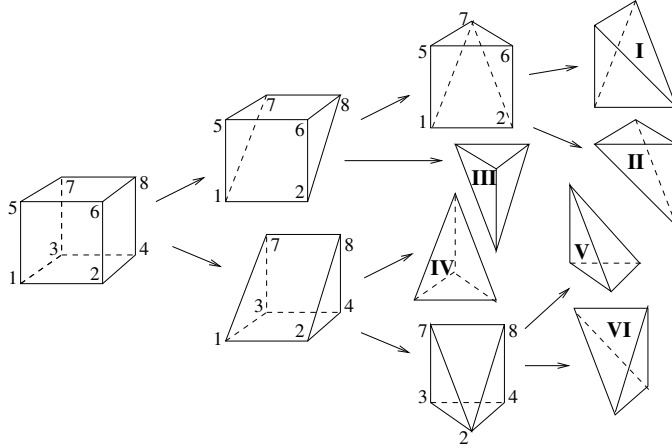


Figure 6. Subdivision of a hexahedral cell into six tetrahedrons.

the mimetic formulation has not yet been widely tested for models with industry-standard geometrical complexity.

In the following we will use a mimetic discretisation. Although mimetic FDMs can handle curved interfaces, we will henceforth make a geometrical simplification. We assume that an underlying subdivision of the corner-point cells into tetrahedrons as illustrated in Figure 6, such that interfaces in the corner-point grid can be described as the union of two planar triangles. The convention in reservoir simulation is to assume that interfaces are bilinear surfaces. Thus, what we consider to be a corner-point cell here is not according to common practice, but we believe that the fact that we define the grid differently is a minor cause of concern. Indeed, geological reservoir models are a result of non-deterministic modelling approaches, and the associated corner-point grid is populated with permeabilities from a stochastic distribution. The assumption that interfaces are bilinear is therefore, in our opinion, invoked only to facilitate numerical reservoir simulation.

The main reason for viewing the corner-point grid cells as a union of tetrahedrons is that it easily allows a consistent treatment of cell geometries. Indeed, the mimetic FDM requires that the following grid information is available for all cells: cell volumes, cell mass centres, face areas, face mass centres, and a net normal vector for each face. In the implementation of the mimetic FDM outlined below, we derive the necessary grid information as follows:

- Cell volumes: sum of tetrahedron volumes.
- Cell centres: volume-weighted average of tetrahedron centres.

- Face areas: sum of sub-triangle areas.
- Face centres: area-weighted average of sub-triangle centres.
- Face normals: area-weighted average of sub-triangle normals.

### 5.1. A MIMETIC FDM FOR CORNER-POINT GRIDS

In this subsection we describe the ideas of the mimetic FDMs in [15]. Since the method is very recent, and not yet in widespread use, we have chosen to present it in some detail. To this end, denote by  $\mathcal{F}_m$  the set of faces of  $T_m$ , and expand  $v$  and  $u$  in the basis  $\{\psi_i^m : F_i \in \mathcal{F}_m, T_m \in \mathcal{T}\}$ :

$$v = \sum_{i,m} v_i^m \psi_i^m \quad \text{and} \quad u = \sum_{i,m} u_i^m \psi_i^m.$$

Since  $b(\psi_i^m, \psi_j^n)$  is nonzero only if  $n = m$ , we may write

$$b(u, v) = \sum_{T_m \in \mathcal{T}} \mathbf{u}_m^T \mathbf{B}_m \mathbf{v}_m, \quad (10)$$

where  $\mathbf{B}_m$  is the block diagonal of  $\mathbf{B}$  associated with  $T_m$ , and  $\mathbf{v}_m, \mathbf{u}_m \in \mathbb{R}^{N_m}$  and  $N_m$  is the number of faces of  $T_m$ . The main idea in mimetic FDMs is to define matrices  $\mathbf{M}_m$  so that  $(\cdot, \cdot)_{\mathbf{M}_m} = (\mathbf{M}_m(\cdot), \cdot)$  defines inner-products that *mimic* the corresponding inner-products  $(\cdot, \cdot)_{\mathbf{B}_m} = (\mathbf{B}_m(\cdot), \cdot)$  associated with the continuous bilinear form  $b(\cdot, \cdot)$ , but which do not require explicit representations of the velocity in each cell.

In the current presentation, we assume that  $v_i^m$  represents the total flux out of  $T_m$  across  $F_i \subset \mathcal{F}_m$ :

$$v_i^m = \int_{F_i} v(s) \cdot n \, ds, \quad F_i \in \mathcal{F}_m. \quad (11)$$

Brezzi et al. [12] established that, in addition to symmetry, the following conditions are sufficient to ensure stability and convergence in pressure and velocity of the mimetic FDM for very general polygons:

1. There exist positive constants  $s_*$  and  $S^*$  such that

$$s_* |T_m| \mathbf{v}^T \mathbf{v} \leq (\mathbf{v}, \mathbf{v})_{\mathbf{M}_m} \leq S^* |T_m| \mathbf{v}^T \mathbf{v}, \quad (12)$$

for all  $T_m \in \mathcal{T}$  and  $\mathbf{v} \in \mathbb{R}^{N_m}$ . Notice that the bounds in this expression differ from the bounds given in [15] in the sense that we use flux unknowns rather than velocity unknowns. Thus, to obtain the constants  $s_*, S^*$  in [15], one should scale the components of  $\mathbf{v}$  by the inverse area of the corresponding faces.

2. For every  $T_m \in \mathcal{T}$ , every  $\mathbf{v}_m$  obtained by inserting  $v = K\nabla p$  into (11) for a linear function  $p$  on  $T_m$ , and every  $\mathbf{u}_m \in \mathbb{R}^{N_m}$ , we have

$$(\mathbf{v}_m, \mathbf{u}_m)_{\mathbf{M}_m} + \left( \sum_{i=1}^{N_m} u_m^{(i)} \right) \int_{T_m} p \, dx = \sum_{i=1}^{N_m} \frac{u_m^{(i)}}{|F_i|} \int_{F_i} p \, ds. \quad (13)$$

These conditions state that there should exist a global bound on the eigenvalues of all  $\mathbf{M}_m$ , and that the inner-products  $(\cdot, \cdot)_{\mathbf{M}_m}$  should obey the Gauss–Green formula for linear pressure. A direct consequence is that the method will be exact for linear pressure; that is, the fluxes obtained by the numerical solution will match the fluxes corresponding to the exact solution.

#### Implementation of $\mathbf{M}_m$

Explicit formulae for computing  $\mathbf{M}_m$ -matrices that obey (12)–(13) in the case where the variables  $v_i^m$  represent net *face velocities* were given in [15]. Below we give the corresponding formulae for the case where the variables represent *fluxes*. To this end, let  $\mathbf{N}_m$  be the matrix whose  $i$ 'th row is defined by

$$\mathbf{n}_{m,i} = \frac{1}{|F_i|} \int_{F_i} n_i^T \, ds, \quad (14)$$

where  $n_i$  is the unit normal to face  $F_i$  pointing out of  $T_m$ ; let  $\mathbf{C}_m$  be the matrix whose  $i$ 'th row is defined by

$$\mathbf{c}_{m,i} = \frac{1}{|F_i|} \int_{F_i} (x - x_m)^T \, ds, \quad (15)$$

where  $x_m$  is the centre of  $T_m$ ; and let  $\mathbf{D}_m$  be the diagonal matrix containing the areas  $|F_i|$  of each face. Note that for cells that are unions of a collection of tetrahedrons, as we assume that corner-point grid cells are, these matrices are easily derived from the corresponding tetrahedral grid properties. Finally, let  $\mathbf{Z}_m$  be a  $N_m \times (N_m - d)$  matrix whose columns form an orthonormal basis for the null space of  $\mathbf{N}_m^T$ , and let  $\mathbf{U}_m$  be a symmetric positive-definite matrix of dimension  $(N_m - d) \times (N_m - d)$ . Then the matrix  $\mathbf{M}_m$  defined by

$$\mathbf{M}_m = \mathbf{M}_{m,1} + \mathbf{M}_{m,2}, \quad (16)$$

where

$$\mathbf{M}_{m,1} = \frac{1}{|T_m|} \mathbf{C}_m K^{-1} \mathbf{C}_m^T, \quad \mathbf{M}_{m,2} = \mathbf{D}_m^{-1} \mathbf{Z}_m \mathbf{U}_m \mathbf{Z}_m^T \mathbf{D}_m^{-1}, \quad (17)$$

satisfies the *discrete* Gauss–Green formula (13). In addition, some mild restrictions must be imposed on the grid and on the eigenvalues of  $\mathbf{U}_m$

in order for  $\mathbf{M}_m$  to also satisfy condition (12). We should also remark that explicit formulae for the inverse of  $\mathbf{M}_m$  (for the case where the variables represent *face velocities*) were also given in [15]. These formulae may be used to assemble  $\mathbf{B}^{-1}$  directly. This is advantageous both with respect to computational efficiency, and with respect to avoiding numerical inversion of nearly singular  $\mathbf{M}_m$ -matrices that arise for elements having one or more faces with almost zero area. The reader should consult [14, 12] for further details.

#### Motivation

To motivate the form of  $\mathbf{M}_m$  given in (16), we consider multiscale basis functions similar to (8), and thus make a twist to the presentation given in [15]. Hence, for each face  $F_i \subset \mathcal{F}_m$ , define  $\psi_i$  by

$$\psi_i = -K\nabla p_i, \quad \nabla \cdot \psi_i = \frac{1}{|T_m|}, \quad \psi_i \cdot n = \begin{cases} 1/|F_i| & \text{for } x \in F_i \\ 0 & \text{for } x \in F_j. \end{cases} \quad (18)$$

One could now, theoretically, define the matrix  $\mathbf{M}_m$  by  $[\mathbf{M}_m]_{ij} = b(\psi_i, \psi_j)$  (where  $b$  is given by (2.1)), but for general polyhedral cells, the computation of  $b(\cdot, \cdot)$  is non-trivial, and thus not practical. Instead one could consider a first order approximation, i.e., find  $\mathbf{M}_m$  such that  $[\mathbf{M}_m]_{ij} = b(\psi_i, \psi_j) + \mathcal{O}(h^2)$ , where  $h$  is the grid-cell diameter. For this to hold true, it is sufficient to require  $\mathbf{u}^T \mathbf{M}_m \mathbf{v} = b(u, v)$  whenever the sum of the polynomial orders of  $u$  and  $v$  are less or equal to one. The mimetic FDMs are based on a somewhat stricter requirement, that is,  $\mathbf{u}^T \mathbf{M}_m \mathbf{v} = b(u, v)$  if one of the functions are constant, and the other is any expansion in the basis functions (18).

In (17), the matrix  $\mathbf{M}_{m,1}$  is defined so that  $b(u, v) = \mathbf{u}^T \mathbf{M}_{m,1} \mathbf{v}$  if  $u$  or  $v$  belongs to the space of constant velocities  $V_0$ , and  $\mathbf{M}_{m,2}$  is defined so that the inner-product  $(\mathbf{u}, \mathbf{v})_{\mathbf{M}_m}$  is not influenced by  $\mathbf{M}_{m,2}$  if either  $\mathbf{u}$  or  $\mathbf{v}$  corresponds to a constant velocity field. This leaves freedom to choose  $\mathbf{U}_m$  only. If we view the mimetic methods as a discrete variant of the mixed finite-element method defined by the basis functions in (18), one should try to define  $\mathbf{U}_m$  so that the following equivalence relation holds:

$$m_*(\mathbf{v}, \mathbf{v})_{\mathbf{B}_m} \leq (\mathbf{v}, \mathbf{v})_{\mathbf{M}_m} \leq M^*(\mathbf{v}, \mathbf{v})_{\mathbf{B}_m}, \quad \forall \mathbf{v} \in \mathbb{R}^{N_m} \quad (19)$$

for some positive constants  $m_*$  and  $M^*$  independent of the mesh and coefficients  $K$  of the problem. Here  $\mathbf{B}_m$  is the local stiffness matrix that stems from the mixed finite-element method defined by (18). Requiring that an equivalence relation of this form should hold might imply that  $\mathbf{U}_m$  should be chosen differently for different cell geometries. However, one generally would like to be able to define  $\mathbf{U}_m$  independent of the grid.

To provide some insight into how  $\mathbf{U}_m$  should be chosen, we consider hexahedral elements (shaped like a shoe-box), and compare  $\mathbf{M}_m$  with the corresponding matrix  $\mathbf{B}_m$  for the RT0 space for hexahedral elements. For a full tensor  $K$ , one can show that  $\mathbf{M}_m = \mathbf{B}_m$  if

$$\mathbf{U}_m = \frac{|T_m|}{6} \text{diag}(K^{-1}), \quad \mathbf{Z}_m^T = \frac{\sqrt{2}}{2} \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}, \quad (20)$$

where  $\text{diag}(K^{-1})$  denotes the diagonal part of  $K^{-1}$ . In particular, if  $K$  is isotropic, then

$$\mathbf{U}_m = \frac{|T_m|}{2\text{trace}(K)} \mathbf{I}. \quad (21)$$

This example indicates that  $\mathbf{U}_m$  should scale proportional to the eigenvalues of  $K^{-1}$  and proportional to the volume of  $|T_m|$ . The latter form is more general than (20) in the sense that it allows cells to have an arbitrary number of faces. Since the number of faces may vary for cells in a corner-point grid, we use  $\mathbf{U}_m$  defined by (21) in our implementation. Results that are reported in [4] reveal that the mimetic FDM outlined above with  $\mathbf{U}_m$  chosen as in (21) generally gives accurate results, also for cases with full tensor permeability. In particular it tends to be more robust than the RT0 method for tetrahedral grids. However, loss of monotonicity is observed for some cases with skew grids, and large anisotropy ratios.

## 5.2. GENERAL REMARKS

Although mimetic methods are less mature than mixed FEMs and multi-point FDMs, they are quite easy to implement on complex grids and allow for a large degree of freedom in using unstructured grids consisting of general polyhedral cells to model complex geology. We believe that using a mimetic method as subgrid solver will be particularly advantageous to model flow in reservoirs with faults. Faults are usually modelled as hyperplanes, i.e., as surfaces. Across fault-faces, the corner-point grids are generally non-conforming, having non-matching interfaces, see Figure 3. A mimetic method handles grids with non-matching faces in a natural way by assigning a “basis function” for velocity to each “sub-face”  $\gamma_{ij} = \partial T_i \cap \partial T_j$  of the faults. In a mixed method, flux continuity across fault-faces can be modelled using the hybrid formulation with a Lagrange multiplier (a mortar) for each “sub-face” of the fault-faces. However, although we feel that the mimetic method handles non-matching faces in a more natural way, we will not demonstrate this flexibility with respect to non-conforming grids arising in grid-models with faults in this paper.

## 6. Numerical Comparisons

In this section we examine the quality of MsMFEM solutions obtained using the mimetic method to compute basis functions. To this end, we consider two examples. In Section 6.1, we consider a small Y-shaped synthetic sector model with three different isotropic permeability structures and discuss several fundamental numerical properties of the multiscale method. Then in Section 6.2, we consider a larger and more complex model of a layered bed on meter-scale.

In the following we will only compare the quality of the MsMFEM velocity solutions. To compare the quality of the multiscale velocity fields we will consider two error measures: (i) errors in fluxes across interfaces in the corner-point grid, and (ii) saturation errors in a two-phase flow scenario. The error in fluxes is measured as follows:

$$e(v) = \|(v_{\text{ref}} - v) \cdot n\|_{L^2(\partial\mathcal{T})} / \|v_{\text{ref}} \cdot n\|_{L^2(\partial\mathcal{T})}. \quad (22)$$

Here  $v_{\text{ref}}$  is obtained by solving (1) using the mimetic solver directly on the subgrid,  $v$  is the multiscale solution,  $\mathcal{T}$  is the corner-point grid, and  $n$  is a unit vector normal to  $\partial\mathcal{T}$ .

The error in fluxes (22) is not always a good indicator for the overall accuracy of a solution, especially when the velocity oscillates rapidly. We therefore also monitor the impact the errors in the velocity have on the saturation field obtained by solving a two-phase transport equation of the form,

$$s_t + \nabla \cdot (F_w(s)v) = \max\{f, 0\} + F_w(s) \min\{f, 0\}, \quad (23)$$

where  $F_w(s) = s^2/(s^2 + (1-s)^2)$ . This nonlinear hyperbolic equation is discretised by the first-order implicit upwind scheme on the subgrid and solved using a Newton–Raphson method. To assess the accuracy of the solutions of (23), we measure how much they differ from the reference solution at 0.5 PVI, i.e., we compute

$$e(s) = \|s_{\text{ref}}(\cdot, T) - s(\cdot, T)\|_{L^2} / \|s_{\text{ref}}(\cdot, T)\|_{L^2},$$

where  $s$  is the saturation induced by the MsMFEM velocity fields,  $s_{\text{ref}}$  is the saturation induced by  $v_{\text{ref}}$ , and  $T$  is defined by

$$\int_{t=0}^{t=T} \max\{f, 0\} dt = 0.5|\Omega|.$$

### 6.1. A Y-SHAPED RESERVOIR

The first example is a synthetic reservoir in the shape of the letter Y represented by a grid with  $32 \times 32 \times 8$  blocks, see Figure 7. (Notice that neither the domain nor the grid blocks are perfectly symmetric about the  $yz$ -plane.) This grid model is much smaller than models to which one usually would apply a multiscale method. However, the size of the model is chosen on purpose to illustrate some fundamental aspects of the multiscale method.

The model is equipped with three different isotropic permeability fields: (i) homogeneous permeability; (ii) a log-normal field where the permeability values vary six orders of magnitude; and (iii) a fluvial type permeability field consisting of high-permeable channels on a low-permeable log-normal background; see Figure 7. The log-normal permeability field has been generated by sampling random numbers independently in each cell, applying a Gaussian smoothing with variance 5.0 in each direction, adjusting the variance by multiplying with a constant, and finally taking the exponential. The mean is 2.58 and the variance is 184.1. The artificial, fluvial-like permeability was generated by adding random sinusoidal curves of high permeability to a low-permeable log-normal background field. The mean of the resulting field is 29.0 and the variance is 2246.

We consider two different well configurations (see Figure 7): (WC1) an injector is placed at the base of the Y and one producer in each of the outer corners of the arms of the Y; and (WC2) an injector and a producer in the two outer corners of the arms of the Y (to the right and left, respectively, in Figure 7). The pressure and velocity distributions are only computed once.

To define the coarse grids for the multiscale method, we use a very simple strategy and partition the grid uniformly in index space. In other words, think of the geomodel as a regular  $32 \times 32 \times 8$  Cartesian grid, and partition it into coarse  $M_x \times M_y \times M_z$  grids, in which each block consists of  $(32/M_x) \times (32/M_y) \times (8/M_z)$  cells. Tables I and II report errors in fluxes and saturations obtained on four different coarse grids for well configurations number one and two, respectively.

Let us first consider well configuration WC1 with homogeneous and log-normal permeability fields, for which the flow tends to take place *almost* independently in the two half domains. Table I shows that the best resolution is obtained with the  $16 \times 16 \times 4$  and the  $2 \times 2 \times 1$  grids. For the  $16 \times 16 \times 4$  coarse grid, the coarse-scale system is able to resolve the majority of the global flow. Similarly, for the  $2 \times 2 \times 1$  coarse grid, the multiscale basis functions are able to capture most of the flow pattern in the arms and the leg of the Y, in particular for



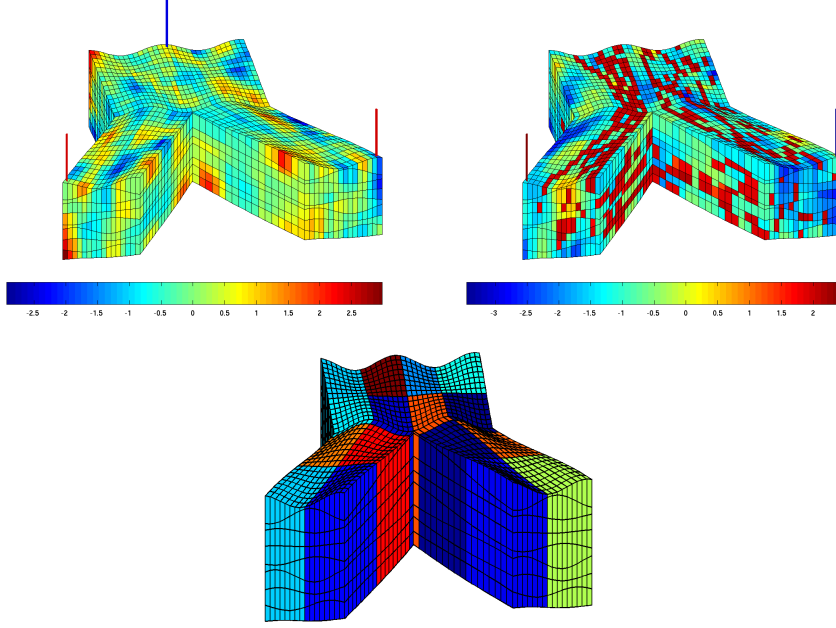


Figure 7. The Y-shaped reservoir: log-normal permeability field and WC1 (left), fluvial permeability field and WC2 (right), and partitioning into  $4 \times 4 \times 1$  coarse blocks (bottom).

Table I. Errors in fluxes and saturations for the Y-shaped reservoir computed using MsMFEM for well configurations number one.

Mimetic FDM Coarse grid	Homogeneous		Log-normal		Fluvial	
	$e(v)$	$e(s)$	$e(v)$	$e(s)$	$e(v)$	$e(s)$
$16 \times 16 \times 4$	0.1152	0.0193	0.1963	0.0532	0.4143	0.2278
$8 \times 8 \times 2$	0.1282	0.0213	0.3174	0.1157	0.4742	0.3607
$4 \times 4 \times 1$	0.1070	0.0249	0.2212	0.1582	0.3119	0.2442
$2 \times 2 \times 1$	0.0111	0.0103	0.1214	0.0751	0.1589	0.0679

the homogeneous case, for which there is a very small flux across the  $yz$ -plane. For the other two coarse grids, the coarse-scale system is too small to accurately capture the global flow and the local fine-scale systems are too small to represent the (global) flow patterns accurately in the multiscale basis functions. Previous experience with MsMFEM on Cartesian grids indicates that typical upscaling factors of 4–16 in each direction are the most computationally feasible, giving local problems with between 100 and 1 000 cells.

Table II. Errors in fluxes and saturations for the Y-shaped reservoir computed using MsMFEM for well configuration number two.

<b>Mimetic FDM</b> Coarse grid	Homogeneous		Log-normal		Fluvial	
	$e(v)$	$e(s)$	$e(v)$	$e(s)$	$e(v)$	$e(s)$
$16 \times 16 \times 4$	0.1409	0.0266	0.2307	0.0478	0.4215	0.2440
$8 \times 8 \times 2$	0.1567	0.0584	0.3472	0.0934	0.4466	0.3331
$4 \times 4 \times 1$	0.1776	0.1527	0.2097	0.1302	0.2974	0.2711
$2 \times 2 \times 1$	0.2537	0.3735	0.2810	0.3720	0.1846	0.1799

The flow pattern for the fluvial permeability field is dominated by high-permeable heterogeneous structures that have the same length scale as the whole reservoir. The flow patterns represented in the multiscale basis functions are restricted locally to each pair of coarse-grid blocks by the assumption of no-flow boundary conditions on the local flow problems. In other words, the local flow problems are not able to accurately resolve and represent the long correlation structures in the high-permeable channels. This is reflected in Tables I and II, where one can observe that the resolution is improved by increasing the size of the coarse blocks. For fluvial, and other heterogeneities with long correlation lengths, a better approach would be to use *global boundary conditions* to define the multiscale basis functions. That is, to use a single initial pressure solution on the fine grid to compute representative boundary conditions for the local flow problems; see e.g., [2]. This way, the multiscale basis functions are able to represent local parts of flow patterns extending beyond a single block in the coarse grid.

For WC2, the velocity profile has a sharp turn at the intersection of the two arms of the Y. The basis functions are formed by driving a unit flow from one grid-block to its immediate neighbour, which here lies along one of the coordinate directions. MsMFEM therefore has no natural representation of flow fields turning ninety degrees. This effect is particularly evident for the homogeneous case, where the multiscale basis functions will correspond roughly to the RT0 basis functions on the coarse grid. The sharp turn at the intersection will therefore be more smeared out as the size of the blocks increases. Hence, the resolution deteriorates when the size of the coarse blocks increases, as observed for the homogeneous and log-normal permeabilities in Table II.

Similar results are reported in [4] for a multiscale method using a mixed FEM with RT0 elements as subgrid solver on a tetrahedral subdivision of the fine grid.

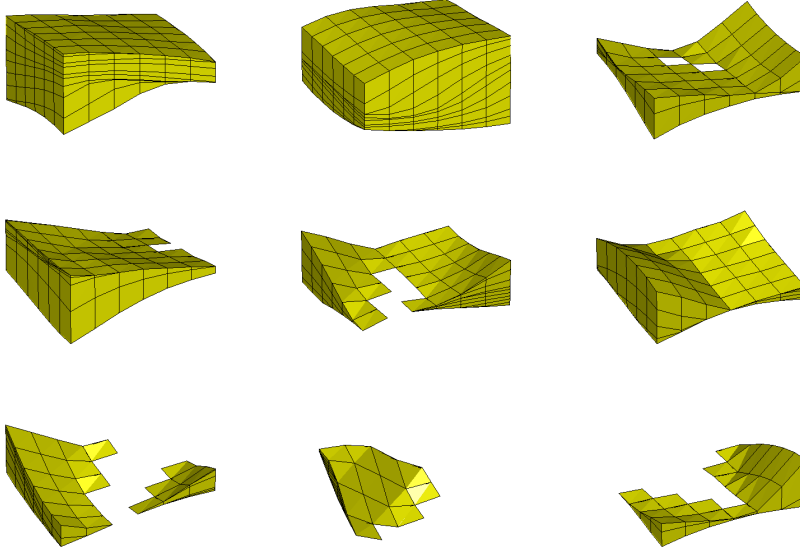


Figure 8. Nine different blocks arising from a uniform partitioning of the wave-bed model in index space.

## 6.2. A WAVY DEPOSITIONAL BED

In the second example we consider a corner-point grid modelling a wavy depositional bed on a meter-scale. The corner-point grid has vertical pillars that form a uniform  $30 \times 30$  grid in the horizontal plane. The model has 100 very thin layers, of which many collapse to a hyper-plane in some regions; see Figure 11. The grid has 29 629 active cells originally. However, thirty-one of the cells have a degeneracy as in the lower-right plot in Figure 2. If we assume that all corner-point cells are unions of tetrahedrons as in Figure 6, the two internal parts of each of these thirty-one cells are only connected through a single line (i.e., through an interior interface of zero area). These cells are therefore split into two cells each, giving a grid that altogether consists of a total of 29 660 cells.

### *Coarse-Grid Partition in Index Space*

We first partition the grid uniformly in index space to obtain  $M_x \times M_y \times M_z$  coarse grids. However, unlike the previous example, we now obtain blocks that have zero volume, blocks that are disconnected, and blocks that contain internal holes; see Figure 8. We therefore split disconnected blocks and remove blocks with zero volume so that each new block consists of a connected collection of fine-grid cells with positive volume.

Table III. Errors in fluxes and saturations for the wavy depositional bed computed using the MsMFEM with subgrid solver MFDM (top) and RT0 MFEM (bottom).

<b>Mimetic FDM</b> Coarse grid	Isotropic $K$		Anisotropic $K$		Heterogeneous $K$	
	$e(v)$	$e(s)$	$e(v)$	$e(s)$	$e(v)$	$e(s)$
$10 \times 10 \times 10$	0.2028	0.0264	0.2485	0.1427	0.2152	0.0941
$6 \times 6 \times 2$	0.1302	0.0420	0.2071	0.1691	0.1547	0.1406
$3 \times 3 \times 1$	0.0624	0.0651	0.0948	0.1271	0.0807	0.1057
$5 \times 5 \times 10$	0.3521	0.0604	0.3058	0.1381	0.3474	0.1415
<b>RT0 MFEM</b>	$e(v)$	$e(s)$	$e(v)$	$e(s)$	$e(v)$	$e(s)$
$10 \times 10 \times 10$	0.1910	0.0223	0.2314	0.1434	0.1967	0.0912
$6 \times 6 \times 2$	0.1256	0.0412	0.2030	0.1734	0.1499	0.1456
$3 \times 3 \times 1$	0.0607	0.0653	0.0931	0.1300	0.0793	0.1077
$5 \times 5 \times 10$	0.3359	0.0541	0.2846	0.1344	0.3294	0.1380

We now consider three different permeability fields; a homogeneous and isotropic field; a homogeneous and anisotropic field for which  $K$  is diagonal with  $k_{xx} = k_{yy} = 1000$  and  $k_{zz} = 1$ ; and a heterogeneous and isotropic field for which the permeability in each layer is log-normally distributed with spatial correlation. The permeability varies between 0.10 mD and 1.69 D with mean 200.9 mD ( $\log(K) = 2.01$ ) and standard deviation 154.1 mD ( $\sigma(\log(K)) = 0.89$ ). Although our grid geometry originally models a depositional bed, we here use it as a sector model and specify a standard quarter-five-spot flow scenario with injection in the first vertical column and production in the last vertical column of the grid. Table III reports the corresponding errors in fluxes and saturations measured relative to the MFDM subgrid solution for four different coarse grids and Figure 9 shows the corresponding watercut curves. For comparison, Table III also reports errors for a multiscale method using MFEM with RT0 elements as a subgrid solver on the tetrahedral subdivision of the fine grid consisting of 147 334 cells. These errors are measured relative to the RT0 fine-grid solution project onto the hexahedral grid.

MsMFEM generally gives good results for both subsolvers on all four coarse grids, both in terms of errors for all three permeability models and watercuts for the heterogeneous model. The worst results for the velocity-error are obtained on the  $10 \times 10 \times 10$  and  $5 \times 5 \times 10$  grids, for which two layers are shown in Figure 10. These grids violate several of the guidelines set forth in Section 4.1, and the blocks depicted in Figure 8 are not among the worst in this sense. Due to pinch-out

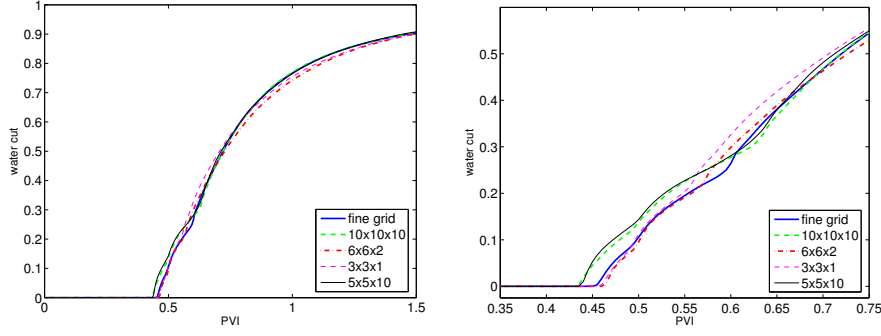


Figure 9. (Left) Watercut curves for the mimetic FDM simulations reported in Table III with a total injection of 1.5 pore volumes of water. (Right) The same watercut curves zoomed in around the breakthrough time.

and erosion, several blocks have an 'ellipsoidal' shape as in Block 5 in the left plot of Figure 5 and block (2,3) in Figure 8. Such blocks have very poor connections in the horizontal direction; in fact, for a few blocks there is no horizontal connection at all. With flow occurring mainly in the horizontal direction, these 'ellipsoidal' blocks tend to be effectively zeroed out in the coarse-scale solution. This is because a horizontal flow will create a bidirectional flow over the top and bottom faces (e.g., enter on the left and leave on the right), which can not be modelled correctly unless special measures are taken to account for bidirectional flows in the MsMFEM basis functions. As an additional complication, several of the layers (number 2, 5–7, and 10 from the bottom) have very thin blocks, and some blocks even consist of only one or two (almost flat) fine-grid cells. Altogether, this gives a higher error in the velocity approximation than what is obtained using one or two coarse layers. However, since the worst blocks typically have small volumes, the enlarged errors in velocity only have a limited effect on the saturation distribution. Finally, the  $10 \times 10 \times 10$  grid has more connections in the horizontal direction than the  $5 \times 5 \times 10$  grid and will therefore give somewhat better results.

#### *Alternative Grid Generation Procedures*

As discussed in Section 4.1, there are several simple guidelines one can use to improve the quality of the coarse grids, and thereby increase the accuracy of the multiscale methods. To shed some light on the general robustness of the method, we give three examples of possible grid generation procedures: (i) uniform partitioning in index space, (ii) partitioning in physical space, and (iii) constrained partitioning in

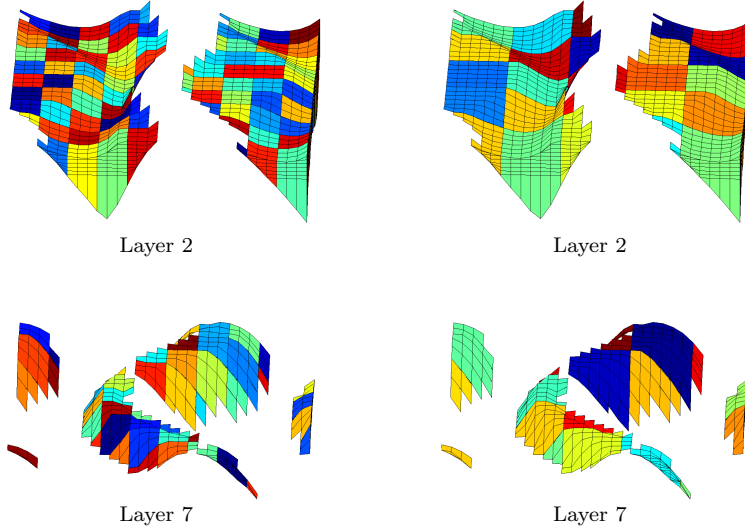


Figure 10. Layers from coarse grid  $10 \times 10 \times 10$  (left) and  $5 \times 5 \times 10$  (right).

index space. The latter may be viewed as a grid-processing routine to improve grids obtained with uniform partitioning in index space.

When partitioning in physical space, we draw a uniform Cartesian  $M_x \times M_y \times M_z$  grid overlaying the fine-grid model, and let a block in the coarse grid be the union of all grid cells with mass centre inside the associated cell in the Cartesian grid. Although more advanced approaches can be applied, such as trying to preserve the shape of the target cells as closely as possible, partitioning in physical space typically gives coarse grids that do not follow the geological layers and have (highly) irregular interfaces.

For the constrained partitioning, we start with a uniform partitioning in index space. Then, starting at the bottom (or top), we stack layers of grid blocks together until the minimum thickness of each stack (except possibly the stack on top or bottom) is larger than  $H_z/M_z$  where  $H_z$  is the total height of the model. This way, we get a coarse grid that follows the geological layers and has no non-neighbouring connections in the horizontal direction. At the same time, we avoid saw-tooth interfaces and blocks with very small volumes.

To evaluate the different grid partitioning procedures, we compare results obtained for the three procedures. First we consider the most difficult grid above, i.e.,  $M_x = 5$ ,  $M_y = 5$ , and  $M_z = 10$ . The corresponding results are presented in Table IV. In the constrained partitioning in index space (starting stacking layers from the top) we reduce the number of layers in the coarse grid to 3. Thus, as a reference,

Table IV. Errors in fluxes and saturations for the wavy depositional bed computed using the MsMFEM with subgrid solver MFDm for coarse grids obtained using uniform partitioning in index space (UPI), uniform partitioning in physical space (UPP), and constrained partitioning in index space (CPI). The dimensions of the (unconstrained) coarse-grid are given in parenthesis.

Grid generation procedure	Isotropic $K$		Anisotropic $K$		Heterogeneous $K$	
	$e(v)$	$e(s)$	$e(v)$	$e(s)$	$e(v)$	$e(s)$
UPI ( $5 \times 5 \times 10$ )	0.3521	0.0604	0.3058	0.1381	0.3474	0.1415
UPI ( $5 \times 5 \times 3$ )	0.1719	0.0566	0.3158	0.1927	0.1790	0.1292
UPP ( $5 \times 5 \times 10$ )	0.5905	0.1339	0.6531	0.2743	0.6011	0.2000
UPP ( $5 \times 5 \times 3$ )	0.3326	0.0535	0.3502	0.1699	0.3777	0.1457
CPI ( $5 \times 5 \times 10$ )	0.1809	0.0573	0.2131	0.1479	0.1962	0.0993

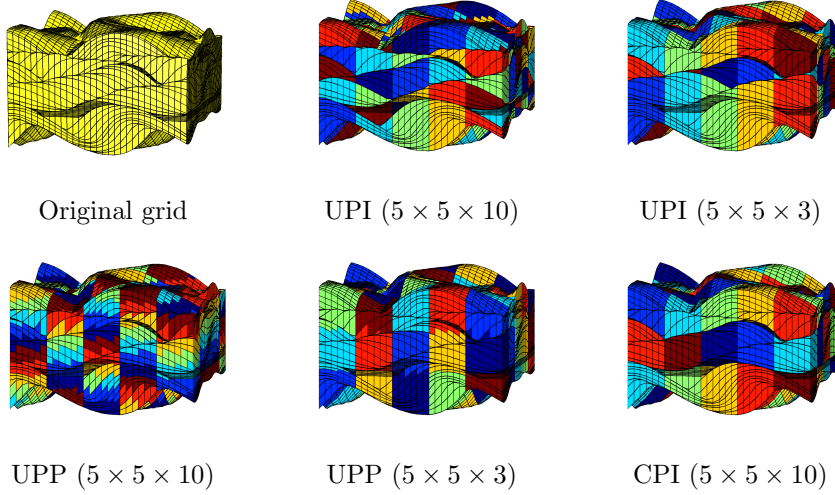


Figure 11. Corner-point grid for the wavy depositional bed: original grid (left) and a coarse grid obtained with different grid processing procedures.

we include also results obtained for  $M_x = 5$ ,  $M_y = 5$ , and  $M_z = 3$  using uniform partitioning in index space, and (uniform) partitioning in physical space. All grids used to obtain the results in Table IV are depicted in Figure 11.

The results clearly illustrate that partitioning in physical space gives significantly less accurate results than partitioning in index space. This is not surprising, considering the very rough surfaces of the blocks obtained. Indeed, as seen in Figure 11, grids obtained by partitioning in

physical space violate many of the guidelines in Section 4.1; in particular, they do not follow the geological layering and have very irregular interfaces. In contrast, we see that partitioning uniformly in index space gives significantly more accurate results. The results indicate also that constraining the thickness of the layers in the coarse grid to prevent non-neighbouring connections may improve accuracy, although the improvement relative to uniform partitioning in index space is not necessarily substantial. Nevertheless, constraining the thickness of each layer is a *very simple* grid-processing procedure, and the small effort it requires pays off by providing consistently accurate results.

## 7. Concluding Remarks

Multiscale methods have become popular among researchers in many engineering disciplines. Nevertheless, multiscale methods are generally regarded as immature, and few, if any, have penetrated into industrial use. In this paper we have tried to promote the use of multiscale methods for flow simulation of geometrically complex and highly heterogeneous reservoir models. To this end, we have extended the multiscale finite-element method (MsMFEM) to corner-point grids, which is the grid format used by most commercial geomodelling software and reservoir simulators to model porous rock formations containing producible hydrocarbon.

In the current paper and in [4], the multiscale mixed formulation was used in combination with two different subgrid solvers that both aim to give accurate and robust discretisations with minimal grid-orientation effects on grids with strong anisotropies and irregular geometries. Although both subgrid solvers give good results, the results obtained with the recent mimetic formulation are particularly promising and encourage further development towards future use within reservoir simulation. Mimetic methods work directly on the corner-point grid, can be formulated for general polyhedral cells with curved faces, and are easy to implement. Moreover, our (limited) experience indicates that they are quite robust with respect to the anisotropies and geometrical complexities encountered in industry-standard corner-point grids.

The numerical results presented in the paper demonstrate that the multiscale mixed method provides detailed and (quite) accurate velocity fields and saturation profiles for two-phase simulations on typical geomodels arising in real-life reservoir engineering. Combined with previous results on Cartesian grids [1, 2, 3], we believe that these results demonstrate that the multiscale mixed formulation is a versatile and robust alternative to the traditional upscaling/downscaling approach



used in the petroleum industry. For a further comparison of various state-of-the-art upscaling and multiscale methods we refer the reader to another paper in this special issue [24].

A particular advantage of the multiscale mixed formulation is its ease of implementation. Given a proper discretisation methodology on the underlying corner-point grid, it is straightforward to implement the multiscale mixed method on top of the existing pressure solver. In fact, our coarse-grid Matlab code used in Section 6—including grid processing, hybrid formulation, and linear algebra—only consists of about 200 code lines. Implementing MsMFEM in a commercial solver should therefore not be a daunting task.

Another advantage of the MsMFEM formulation is the ease with which one handles the coarse grids. In particular, we have demonstrated how one can avoid the difficulties of resampling that are usually encountered in grid coarsening, since the multiscale mixed formulation allows the cells in the coarse grid to be chosen as an (almost) arbitrary connected collection of cells in the underlying geological model. A simple partition of cells in index space often produces a coarse grid that gives sufficient accuracy in the multiscale method. If not, we have presented several simple guidelines that can be used to improve the coarse grids. Using these guidelines, it should not be very difficult to derive a robust and automated algorithm for producing good coarse grids.

We believe that the ease with which one handles coarse grids will prove to be very advantageous when considering more complex geomodells containing fractures and faults, for which the corresponding geological models will generally be non-conforming and therefore have non-matching interfaces. In ongoing research, we try to extend the multiscale formulation to industry-standard geological models with fractures/faults and non-conforming fine grids.

In summary, we believe that the multiscale mixed finite-element method can greatly facilitate flow simulation on complex grid models and potentially be used for reservoir simulation directly on full-scale geological models if one is able to extend the methodology to more complex flow physics like black-oil and compositional models.

### Acknowledgement

The research is funded by the Research Council of Norway under grants no. 158908/I30 and 162606/V30. The corner-point grid used in Section 6.2 is generated with SBED<sup>TM</sup>, and has been used courtesy of Dr. Alf B. Rustad, Statoil Research Centre. We would also like to thank the reviewers for helpful comments that helped improve the quality of the paper.

## References

1. Aarnes, J. E.: 2004, ‘On the use of a mixed multiscale finite element method for greater flexibility and increased speed or improved accuracy in reservoir simulation’. *Multiscale Model. Simul.* **2**(3), 421–439.
2. Aarnes, J. E., V. Kippe, and K.-A. Lie: 2005, ‘Mixed multiscale finite elements and streamline methods for reservoir simulation of large geomodels’. *Adv. Water Resour.* **28**(3), 257–271.
3. Aarnes, J. E., S. Krogstad, and K.-A. Lie: 2006a, ‘A hierarchical multiscale method for two-phase flow based upon mixed finite elements and nonuniform coarse grids’. *Multiscale Model. Simul.* **5**(2), 337–363.
4. Aarnes, J. E., S. Krogstad, K.-A. Lie, and V. Laptev: 2006b, ‘Multiscale mixed methods on corner-point grids: mimetic versus mixed subgrid solvers’. Technical report, SINTEF.
5. Aavatsmark, I., T. Barkve, Ø. Bøe, and T. Mannseth: 1998a, ‘Discretization on unstructured grids for inhomogeneous, anisotropic media. Part I: Derivation of the methods’. *SIAM J. Sci. Comp.* **19**(5), 1700–1716.
6. Aavatsmark, I., T. Barkve, Ø. Bøe, and T. Mannseth: 1998b, ‘Discretization on unstructured grids for inhomogeneous, anisotropic media. Part II: Discussion and numerical results’. *SIAM J. Sci. Comp.* **19**(5), 1717–1736.
7. Arbogast, T.: 2000, ‘Numerical subgrid upscaling of two-phase flow in porous media’. In: Z. Chen, R. Ewing, and Z.-C. Shi (eds.): *Lecture Notes in Phys.* Berlin: Springer-Verlag, pp. 35–49.
8. Arbogast, T.: 2004, ‘Analysis of a two-scale, locally conservative subgrid upscaling for elliptic problems’. *SIAM J. Numer. Anal.* **42**(2), 576–598.
9. Arbogast, T. and K. Boyd: 2006, ‘Subgrid upscaling and mixed multiscale finite elements’. *SIAM J. Numer. Anal.* **44**(3), 1150–1171.
10. Arbogast, T., S. E. Minkoff, and P. T. Keenan: 1998, ‘An operator-based approach to upscaling the pressure equation’. In: B. et al. (ed.): *Computational Methods in Water Resources XII, Vol. 1: Computational Methods in Contamination and Remediation of Water Resources*. Computational Mechanics Publications, pp. 405–412.
11. Arnold, D. N. and F. Brezzi: 1985, ‘Mixed and nonconforming finite element methods: implementation, postprocessing and error estimates’. *RAIRO Modél. Math. Anal. Numér.* **19**(1), 7–32.
12. Brezzi, F., K. Lipnikov, and M. Shashkov: 2005a, ‘Convergence of mimetic finite difference methods for diffusion problems on polyhedral meshes’. *SIAM J. Num. Anal.* **43**, 1872–1895.
13. Brezzi, F., K. Lipnikov, and M. Shashkov: 2006, ‘Convergence of mimetic finite difference methods for diffusion problems on polyhedral meshes with curved faces’. *Math. Models Methods Appl. Sci.* **16**(2), 275–297.
14. Brezzi, F., K. Lipnikov, M. Shashkov, and V. Simoncini: 2005b, ‘A new discretization methodology for diffusion problems on generalized polyhedral meshes’. LAUR-05-8717, Report of Los Alamos National Laboratory, <http://cnls.lanl.gov/~shashkov/>.
15. Brezzi, F., K. Lipnikov, and V. Simoncini: 2005c, ‘A family of mimetic finite difference methods on polygonal and polyhedral meshes’. *Math. Models Methods Appl. Sci.* **15**, 1533–1553.
16. Chen, Z. and T. Hou: 2003, ‘A mixed multiscale finite element method for elliptic problems with oscillating coefficients’. *Math. Comp.* **72**, 541–576.

17. Edwards, M., R. Lazarov, and I. Yotov (eds.): 2002, 'Special issue on locally conservative numerical methods for flows in porous media'. *Comp. Geosci.* **6**(3-4), 225–579.
18. Gautier, Y., M. J. Blunt, and M. A. Christie: 1999, 'Nested gridding and streamline-based simulation for fast reservoir performance prediction'. *Comp. Geosci.* **3**, 295–320.
19. Hou, T. and X.-H. Wu: 1997, 'A multiscale finite element method for elliptic problems in composite materials and porous media'. *J. Comput. Phys.* **134**, 169–189.
20. Hughes, T., G. Feijoo, L. Mazzei, and J.-B. Quincy: 1998, 'The variational multiscale method – a paradigm for computational mechanics'. *Comput. Methods. Appl. Mech. Engrg.* **166**, 3–24.
21. Jenny, P., S. H. Lee, and H. A. Tchelepi: 2003, 'Multi-scale finite-volume method for elliptic problems in subsurface flow simulation'. *J. Comput. Phys.* **187**, 47–67.
22. Jenny, P., S. H. Lee, and H. A. Tchelepi: 2004, 'Multi-scale finite-volume method for elliptic problems in subsurface flow simulation'. *Multiscale Model. Simul.* **3**(1), 50–64.
23. Juanes, R.: 2006, 'A variational multiscale method for the simulation of porous media flow in highly heterogeneous formations'. In: P. J. Binning, P. K. Engesgaard, H. K. Dahle, G. F. Pinder, and W. G. Gray (eds.): *Proceedings of the XVI International Conference on Computational Methods in Water Resources*. <http://proceedings.cmrw-xvi.org>.
24. Kippe, V., J. Aarnes, and K.-A. Lie: 2006, 'A comparison of multiscale methods for elliptic problems in porous media flow'. *Comp. Geosci.* to appear.
25. Målqvist, A.: 2005, 'Adaptive Variational Multiscale Methods'. Ph.D. thesis, Chalmers University of Technology.
26. Ponting, D. K.: 1989, 'Corner point geometry in reservoir simulation'. In: P. King (ed.): *Proceedings of the 1st European Conference on Mathematics of Oil Recovery, Cambridge, 1989*. Oxford, pp. 45–65.
27. Raviart, P. and J. Thomas: 1977, 'A mixed finite element method for second order elliptic equations'. In: I. Galligani and E. Magenes (eds.): *Mathematical Aspects of Finite Element Methods*. Berlin – Heidelberg – New York: Springer-Verlag, pp. 292–315.

