

Modeling of two-phase flow in fractured porous media on unstructured non-uniformly coarsened grids

Vera Louise Hauge and Jørg Espen Aarnes

Abstract. In this paper we investigate two strategies for coarsening fractured geological models. The first approach, which generates grids that resolve the fractures, is referred to as explicit fracture-matrix separation (EFMS). The second approach is based on a non-uniform coarsening strategy introduced in (Aarnes et al., 2007a). A series of two-phase flow simulations where the saturation is modeled on the respective coarse grids are performed. The accuracy of the resulting solutions is examined and the robustness of the two strategies is assessed with respect to number of fractures, degree of coarsening, well locations, phase viscosities, and fracture permeability. The numerical results show that saturation solutions obtained on the non-uniform coarse grids are consistently more accurate than the corresponding saturation solutions obtained on the EFMS grids. The numerical results also reveal that it is much easier to tune the upscaling factor with the non-uniform coarsening approach.

Keywords: Two-phase flow; Upscaling; Fractured porous media.

1. Introduction

Many applications in science and engineering involve flow processes in porous media. Porous media flow simulations are for instance used as an integrated part of planning and management of oil and gas reservoirs, to monitor transport of pollutants in the subsurface, and to assess storage capacity and predict leakage rates for potential CO₂ repositories, a key component in a global strategy to mitigate emission of CO₂ into the atmosphere, see e.g., (Schrag, 2007). However, although there is an established framework for modeling subsurface flow, there are still important knowledge gaps, partly due to inadequate data, and partly due to limitations in modeling capabilities. The latter is particularly true for modeling of flow in fractured porous media.

Most natural porous media contain fractures at various scales. At small scales, the magnitude and orientation of fractures strongly influence preferential flow directions. This effect can to some extent be upscaled and embedded into the permeability tensor. Fractures with size comparable to the geological flow domain, as is often observed in carbonate reservoirs, may dominate large scale flow patterns and should ideally be resolved by the simulation grid. Unfortunately, because fractures are thin relative to their surface area and can have complex geometries, it is difficult to resolve fractures with industry-standard grids of suitable size for flow simulation. Hence, although capability to resolve large scale fractures is essential for developing predictive simulation models, this capability is generally not available today.



© 2008 Kluwer Academic Publishers. Printed in the Netherlands.

The traditional approach to model flow in fractured porous media is based on the dual-porosity model (Barenblatt et al., 1960; Warren and Root, 1963; Kazemi, 1969). In a dual-porosity model the fractured porous medium is modeled as two overlapping domains, the fractures and the matrix. The matrix covers the entire domain, whereas the fractures constitute an inter-connected web. The two overlapping continua interact through flow transfer terms between the two systems. The matrix system accounts for most of the pore volume whereas most of the flow takes place in the fractures. In the dual-porosity and single-permeability model there is no flow from matrix to matrix, i.e., all flow into a block flows directly into the fracture web, and then into the matrix in neighboring blocks. In the dual-porosity and dual-permeability model the flow is also allowed to go directly from matrix to matrix.

Dual-porosity models assume only geostatistical information about the fractures, e.g., porosity, volume, and orientation. For small scale fractures deterministic information is not readily available, but when modeling flow in subsurface formations with large scale fractures one will often have some information available on the size, location and geometry of these fractures. To utilize this information, alternatives to dual-porosity models where fractures are treated as explicit lower dimensional entities have been proposed by several authors (Reichenberger et al., 2006; Karimi-Fard and Firoozabadi, 2001; Karimi-Fard et al., 2003). These approaches are based on adapting the simulation grid so that the grid interfaces are aligned with the fractures.

In this paper we also assume explicit knowledge of the fractures, but instead of treating the fractures as lower dimensional entities, we represent them as volumetric elements. That is, we assume that there exists a fine scale geological model (geomodel) where the fractures are represented as connected paths of adjacent cells in the grid. The challenge that we address here is how to model the flow on a coarsened grid. In other words, we assume that performing simulations directly on the geomodel is not feasible.

The main purpose in this work is to develop a strategy for upscaling geomodels with fractures to a coarsened model suitable for simulation in a way capable of preserving the most important flow characteristics. To this end we will investigate two different approaches. In the first approach we introduce a coarse grid that separates the fractures from an underlying coarse grid. In the resulting grid, each grid block will either be a connected part of an original coarse grid block with no fractures, or a connected part of the fracture web. The second approach employs a non-uniform coarsening strategy for structured and unstructured grids introduced in (Aarnes et al., 2007a). This approach aims to resolve high flow regions more accurately than what is generally possible with conventional coarse grids. A major advantage with this approach is that it is very flexible with respect to the fracture geometry so that no special model to handle fractures is needed.

The bulk of the paper deals with pressure driven flow scenarios, i.e., immiscible and incompressible two-phase flow without effects from gravity and capillary pressure. Most field-scale reservoir flow regimes are primarily pressure driven, but gravity may be important during a transition period with gravity segregation, and capillary pressure has a local diffusive effect. As such, the current model problem, chosen primarily for clarity of the presentation, accounts for the main driving force.

A description of how to extend the current simulation approach to flow models with effects from gravity and capillary pressure will be given in Section 6. Inclusion of gravity effects is straightforward, but inclusion of capillary forces require special treatment. In this paper we propose a method based on taking a Galerkin projection with respect to a cell-centered finite difference model on the fine grid. It should be mentioned, however, that flow based grids, e.g., (Durlofsky et al., 1997; He and Durlofsky, 2006), and grids tuned to heterogeneous structures, e.g., (Garcia et al., 1992; Cao and Kitandis, 1999; King et al., 2005), target flow scenarios where the heterogeneity has a dominant impact on flow patterns. In other words, it is assumed that viscous forces dominates gravity and capillary forces. This is also implicitly assumed in this paper, but the methodology is, as is shown in Section 6, also applicable to flows influenced by gravity and capillary pressure.

The model problem will be introduced in Section 2. Next, in Section 3 we present the two grid coarsening strategies. In Section 4 we describe the numerical methods used to discretize the two-phase flow model. In Section 5 we present numerical results that demonstrate the performance of the grid coarsening strategies on models with large scale fractures. Finally, we describe how to include effects from gravity and capillary pressure in Section 6 and summarize the main observations in Section 7.

2. Mathematical model and fractured reservoir

We consider immiscible and incompressible two-phase flow of water and oil, denoted by subscripts w and o , respectively. Gravity and capillary effects are neglected. The mathematical model is given by the conservation of mass for each phase:

$$\phi \frac{\partial S_j(x, t)}{\partial t} + \nabla \cdot v_j = q_j, \quad j = o, w, \quad (1)$$

where ϕ is the porosity (fraction of void space), S_j and v_j denote the saturation and velocity of phase j , respectively, and q_j is a volumetric source term. The phase velocity v_j is related to the phase pressure p_j through Darcy's law:

$$v_j = -\lambda_j \nabla p_j, \quad \lambda_j = \frac{K k_{rj}}{\mu_j}, \quad j = o, w, \quad (2)$$

where K , k_{rj} and μ_j are the absolute permeability, relative permeability and viscosity of phase j , respectively. The relative permeability models the reduced permeability of a phase due to the presence of other phases. We assume that

$$k_{rw} = S_w^2, \quad k_{ro} = S_o^2, \quad 0 \leq S_w, S_o \leq 1.$$

Since we neglect capillary pressure effects so that $\nabla p_o = \nabla p_w$, we assume that $p_o = p_w = p$. Then the Darcy equations (2) combined with conservation of mass (1) yield the pressure equation:

$$v = -\lambda \nabla p, \quad \nabla \cdot v = q, \quad \text{in } \Omega \subset \mathbb{R}^d, \quad (3)$$

where $v = v_o + v_w$, $\lambda = \lambda_o + \lambda_w$ and $q = q_o + q_w$, and d is the spatial dimension. We close the system by imposing no-flow boundary conditions.

Henceforth we assume that $S_o + S_w = 1$, and drop the subscript of S_w so that $S_w = S$ and $S_o = (1 - S)$. The conservation equation for water, henceforth called the saturation equation, is then written as

$$\phi \frac{\partial S}{\partial t} + \nabla \cdot (f_w v) = q_w, \quad (4)$$

where $f_w = \lambda_w / \lambda$. Finally we assume that the computational domain is initially completely oil-saturated, i.e., that $S(x, 0) = 0$.

2.1. FRACTURE MODEL

Assigning permeability to fractures is a non-trivial task, but fractures generally transmit flow more easily than its surroundings. This implies that the flow will seek to follow paths in the high permeable fracture network, rather than in the surrounding matrix. Since the fractures have little storage capacity, the flow will move quickly through the connected parts of the fracture web. But fractures are also sometimes filled with clay, and have significantly lower permeability than the surroundings. In this case the fractures act as flow barriers, and the flow will attempt to take a detour around the fractures.

In this paper we will for the most part consider high permeable fractures, but low permeable fractures will also be considered. The fractures will be represented as a collection of adjacent cells in a high-resolution grid. The Darcy flow model is assumed to be valid in the fractures.

3. Grid coarsening strategy

To generate a coarse grid for solving the saturation equation we will consider two different strategies; An approach where the fractures are identified and separated from an underlying coarse grid by extracting connected components of the fracture web, and a non-uniform coarsening algorithm first

introduced in (Aarnes et al., 2007a). For both algorithms it is assumed that the fractures are represented on the fine grid so that each fracture is defined as a connected “path” of cells in the fine grid. To distinguish between the coarse and fine grid we will use the term *block* to denote a cell in the coarse grid.

3.1. NON-UNIFORM GRID COARSENING

The coarsening strategy presented in (Aarnes et al., 2007a) is essentially based on grouping cells according to flow magnitude. The algorithm involves two parameters which determine the degree of coarsening; A lower bound on volume of blocks, V_{\min} , and an upper bound on total amount of flow through each block, G_{\max} . As a rule of thumb, to generate a coarse grid with approximately N_c blocks, choose $V_{\min} \sim |\Omega|/(4N_c)$ and $G_{\max} \sim 5|\Omega|g(\Omega)/(4N_c)$, where $|\Omega|$ is the total volume of the reservoir and

$$g(E) = \frac{1}{|E|} \int_E \log |v(x)| dx - \min_{x \in \Omega} (\log |v(x)|) + 1, \quad E \subset \Omega.$$

The steps in the coarsening algorithm are as follows:

1. Group cells according to flow magnitude:
 - a) Compute the initial velocity field v on the fine grid.
 - b) Assign an integer from 1 to 10 to each cell c in the fine grid by

$$n(c) = \text{ceil} \left(\frac{10 [g(c) - \min_c g(c)]}{\max_c g(c) - \min_c g(c)} \right).$$

- c) Create an initial coarse grid with one block assigned to each connected collection of cells with the same value $n(c)$.

The coarse grid now consists of a connected collection of cells with similar flow magnitude and corresponds to the top left plot in Figure 1.

2. Merge each block B with less volume than V_{\min} with the block

$$B' = \arg \min_{B'' \in \text{neighbors}} |g(B) - g(B''|),$$

that is, merge with neighboring blocks subject to flow of similar magnitude.

3. Refine each block B with a total flow through the block greater than the given bound, that is with $|B|g(B) > G_{\max}$, as follows
 - a) Pick an arbitrary cell $c_0 \subset B$ and locate the cell $c_i \subset B$ with center furthest away from the center of c_0 .

- b) Define $B' = c_i$ and progressively enlarge B' by adding the layer of cells in B adjacent to cells in B' until $|B'|g(B') > G_{\max}$.
- c) Define $B = B \setminus B'$ and refine B further if $|B|g(B) > G_{\max}$.

4. Repeat step 2 and terminate.

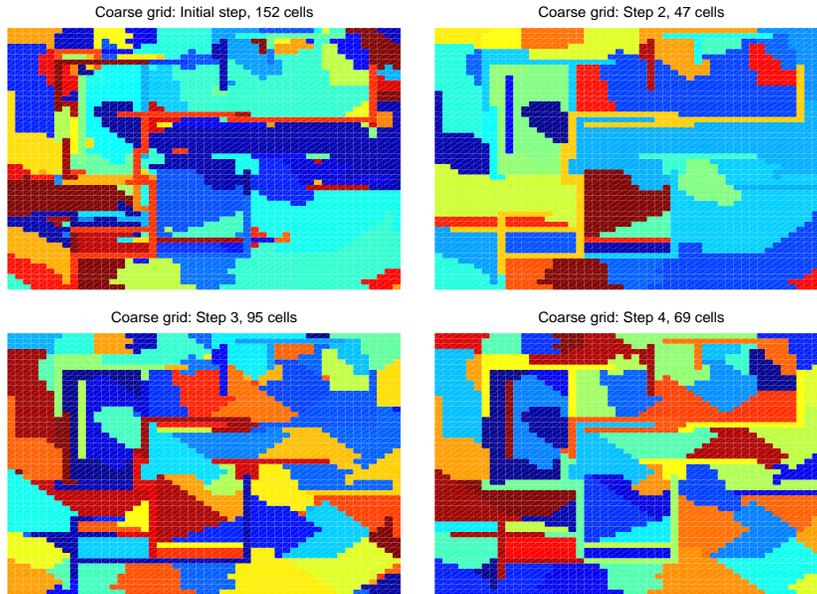


Figure 1. Coarse grid after each step in the non-uniform coarsening algorithm for a case with a 50-by-50 Cartesian grid with homogeneous matrix permeability and 30 high permeable fractures. The coarsening parameters are $V_{\min} = |\Omega|/250$ and $|\Omega|g(\Omega)/50$.

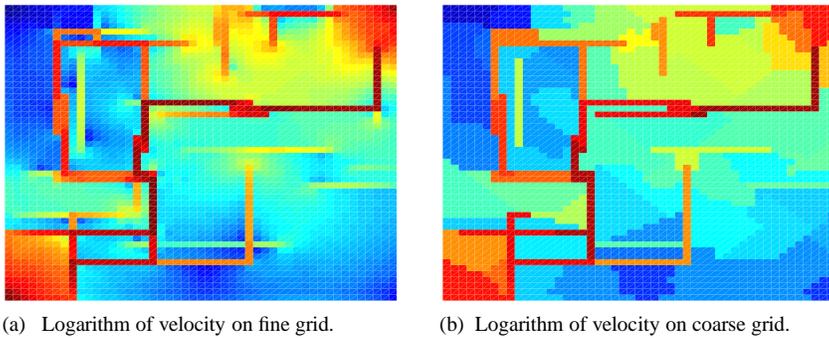


Figure 2. Logarithm of velocity fields for the example case in Figure 1.

Figure 1 illustrates the four steps in the non-uniform coarsening algorithm. We emphasize that the colors do not represent permeability values, but are only used to visualize the sizes and shapes of the blocks in the coarse grid.

Moreover, since the color used to visualize each block is chosen at random each time a grid is plotted, there is no correspondence between the colors in the four subplots.

The top left plot in Figure 1 shows the initial coarse grid. The logarithm of the velocity magnitude in each cell has been used to segment the cells in the fine grid into ten different bins. More specifically, each cell c is assigned a number $n(c) = 1, \dots, 10$ by upper-integer interpolation in the range of $g(c)$. Then an initial coarse grid is created with one block assigned to each connected collection of cells with the same value of $n(c)$. Next, in the top right plot in the figure, blocks with volume less than the given bound V_{\min} has been merged with a neighboring block. Indeed, we observe that the smallest blocks from the initial grid have now disappeared and the number of blocks has been reduced from 152 to 47. In the lower left plot the number of blocks has increased to 95 from the previous step by refining too large blocks with respect to the upper bound on the total flow, G_{\max} . The lower right plot shows the final coarse grid. The number of coarse blocks is slightly reduced again to 69 after removing small blocks that were introduced in the refinement process.

Figure 2(a) plots the magnitude of the velocity field used to generate the initial grid in Figure 1 and Figure 2(b) plots the magnitude of the velocity projected onto the final grid shown in Figure 1. Although the fine grid contains 36 times as many cells as the coarse grid, we observe that there is a good match between Figure 2(a) and Figure 2(b). In particular, we see that the high-flow regions, i.e., the red-colored regions, are matched well.

3.2. EXPLICIT FRACTURE-MATRIX SEPARATION (EFMS)

The EFMS grid generation strategy starts by introducing an initial coarse grid. The next step is to split each block that contains fractures into a matrix part and a fracture part. The final step is to assign grid blocks to the connected pieces from the matrix and fracture web, respectively. Each grid block in the EFMS grid is then a connected collection of cells that either consist solely of fracture cells or solely of matrix cells.

Figure 3 illustrates the steps in this coarsening strategy. The fine grid is 100-by-100 Cartesian grid and the initial coarse grid is a 5-by-5 Cartesian grid. Figure 3(b) shows how the first coarse block in the Cartesian coarse grid is split into a matrix part 1 and a fracture part 2. In Figure 3(c) a further splitting of blocks 1 and 2 is obtained when the non-connected blocks are split into connected components.

In the example in Figure 3 the number of coarse grid blocks is increased from 25 to 130. Thus, the upscaling factor is significantly decreased. The final number of coarse blocks is case specific since it depends both on the initial

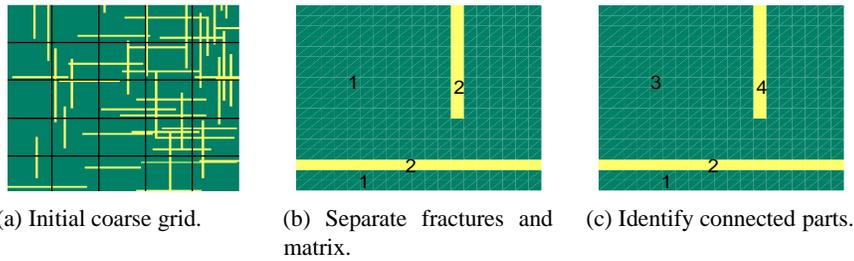


Figure 3. The EFMS coarsening strategy.

coarse grid and inter-connectivity of the fractures. It is therefore difficult to tune the upscaling factor with this algorithm.

4. Numerical discretization

The algorithms presented in the previous section, which generate coarsened grids for solving the saturation equation (4), assume that velocity is computed on the fine grid. Only the non-uniform coarsening algorithm uses the velocity field to generate the coarse grid (Step 1), but both algorithms implicitly demand that one can provide inter-block fluxes, i.e., provide velocity on the interfaces between the blocks in the coarse grid.

4.1. DISCRETIZATION OF THE PRESSURE EQUATION

The geometric complexity of the blocks in the grids generated using the algorithms presented in Section 3 prevents coarse grid discretization of the pressure equation with standard discretization techniques. Hence, the pressure equation must either be solved on the fine grid, or with a multiscale method (e.g., (Arbogast, 2000; Jenny et al., 2003; Aarnes, 2004)) that provides a mechanism for recovering a conservative velocity field on a fine grid from a coarse grid solution.

As the goal of this paper is to present a robust way of modeling flow in fractured porous media as part of a coarse grid simulation framework, we propose solving the pressure equation with a suitable multiscale method. However, since the focus is on modeling the saturation on special coarse grids, we want to eliminate factors that can make it hard to determine where errors stem from. We have therefore chosen to discretize the pressure equation on the fine grid. To this end, we have employed a mimetic finite difference method (F. Brezzi and Shashkov, 2005; F. Brezzi and Simoncini, 2005) that on Cartesian grids (which will be considered in this paper) is equivalent to the Raviart-Thomas mixed finite element method of lowest order (Raviart and Thomas, 1977). Note that if a multiscale method is used, then the coarse

grid for the pressure equation need not coincide with the coarse grid for the saturation equation, although this is an option when using the multiscale mixed finite element method (Aarnes et al., 2007b).

4.2. DISCRETIZATION OF THE SATURATION EQUATION

The saturation equation (4) is discretized on coarse grids where each block B_m consists of a connected collection of cells in the fine grid. Because the blocks typically have highly irregular shapes, it is important that the numerical scheme used to discretize the saturation equation is relatively insensitive to the grid block geometry. One apparent option is the first-order upstream weighted finite volume method

$$S_m^{n+1} = S_m^n + \frac{\Delta t}{\int_{B_m} \phi dx} \left[\int_{B_m} q_w(S^{n+1}) dx - \sum_j \bar{V}_{mj}(S^{n+1}) \right]. \quad (5)$$

Here S_m^n is the net saturation in B_m at timestep n , $\Gamma_{mj} = \partial B_m \cap \partial B_j$ denotes a non-degenerated interface between B_m and B_j , and

$$\bar{V}_{mj}(S) = f_w(S|_{T_i}) \max\{\bar{v}_{mj}, 0\} + f_w(S|_{T_j}) \min\{\bar{v}_{mj}, 0\},$$

where \bar{v}_{mj} is the total Darcy flux from B_m to B_j .

The coarsening algorithms may, however, give rise to grids with blocks that encapsulate other blocks. If an encapsulated block does not contain non-zero source terms, then the velocity is divergence free inside it. This implies that the total Darcy flux across the interface between the encapsulated block and the block surrounding it is zero, which implies that nothing will flow into or out of the encapsulated block when using (5). To avoid this non-physical artifact we can utilize the subgrid resolution in velocity. Indeed, we will generally have that every block is subject to some inflow and some outflow. Hence, by discretizing the saturation equation with a first-order finite volume method where $f_w(S)$ is upstream weighted with respect to the fine grid fluxes on each coarse interface, we avoid the undesirable artifact mentioned above.

Denote the non-degenerate interfaces in the fine grid by $\gamma_{ij} = \partial T_i \cap \partial T_j$. The scheme used to discretize the saturation equation in Section 5 reads

$$S_m^{n+1} = S_m^n + \frac{\Delta t}{\int_{B_m} \phi dx} \left[\int_{B_m} q_w(S^{n+1}) dx - \sum_{\gamma_{ij} \subset \partial B_m} V_{ij}(S^{n+1}) \right], \quad (6)$$

where $V_{ij}(S) = f_w(S|_{T_i}) \max\{v_{ij}, 0\} + f_w(S|_{T_j}) \min\{v_{ij}, 0\}$ and v_{ij} is the flux from T_i to T_j . We point out that although $f_w(S)$ is upstream weighted with respect to fine grid fluxes, and hence evaluated based on saturation in a fine grid cell, we seek a saturation solution that is piecewise constant on the

coarse grid level. The dimension of the discretized system is therefore equal to the number of coarse grid blocks.

5. Numerical simulations

The purpose of the numerical simulations reported in this section is to compare accuracy of solutions obtained by solving the saturation equation on grids generated using the non-uniform coarsening algorithm and the EFMS algorithm, respectively. Robustness will be assessed with respect to number of fractures, degree of coarsening, well configurations, and viscosity ratio. We will for the most part consider high permeable fractures, but low permeable fractures (e.g., fractures filled with clay) will also be considered.

The simulations will be performed using a sequential splitting of the pressure and saturation equation. That is, we compute pressure and velocity at the next time-step with $\lambda(S)$ evaluated using the saturation solution at the current time-step. Next, we keep the velocity v fixed and solve (6) to obtain the saturation at the next time-step. We then reevaluate $\lambda(S)$, compute pressure at the next time-step, and so on.

The alternative to a sequential splitting scheme is either a fully implicit scheme, in which one uses a Newton method to solve for pressure and saturation simultaneously, or an iterated sequential splitting where a fixed-point iterative scheme is used to compute the solution of the fully implicit system. Both of these solution strategies may be applied in conjunction with the discretization methods utilized here, and should give qualitatively similar results for the problems considered in this paper.

We measure accuracy of saturation solutions by comparing the computed water-cut curves – the fraction of water in the produced fluid – to the corresponding water-cut curve obtained by solving the saturation equation on the fine grid. Thus, if $w(t)$ is a water-cut curve obtained by solving the saturation equation on a coarse grid and $w_{\text{ref}}(t)$ is the reference water-cut curve, then the *water-cut error* is defined by

$$e(w) = \|w - w_{\text{ref}}\|_{L^2(0,1)} / \|w_{\text{ref}}\|_{L^2(0,1)}.$$

Time is here measured in PVI (pore volume injected).

5.1. FRACTURE REPRESENTATION

The fractures will be generated stochastically and represented as either a horizontal or vertical strip of cells in a uniform Cartesian grid. The number of fractures in each direction will be proportional to the dimensions of the reservoir, and the length of the fractures is randomly distributed between 20 and 40 percent of the length of the shortest side of the reservoir. Figure 4

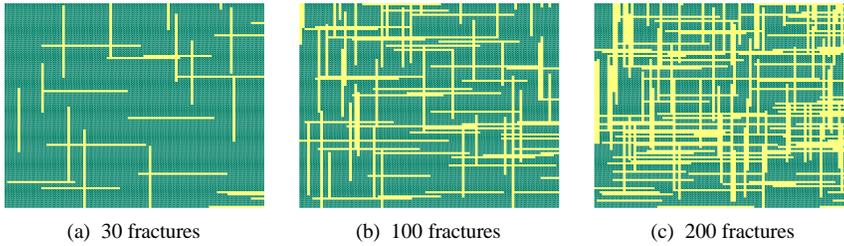


Figure 4. Three fracture realizations in a reservoir with homogeneous background permeability. Each fracture is represented as a horizontal or vertical row of cells.

shows three reservoir models with different number of fractures. Unless stated otherwise the total number of fractures will be 100.

Our fracture distribution model is clearly not realistic. A more realistic model would mainly have fractures of smaller aperture, and perhaps also fractures of preferred orientation. Still our model should be adequate for assessing performance of the proposed approaches for modeling saturation. Indeed, the fractures act as either preferential flow paths or as flow barriers, and flow in reservoirs consisting of only horizontal and vertical fractures does not differ fundamentally from flow in reservoirs with arbitrarily oriented fractures or fractures with a preferred orientation. Moreover, representing arbitrarily oriented fractures as a strip of cells in a fine grid requires an unstructured grid. Although both coarsening algorithms are directly applicable to unstructured grids (this was demonstrated for the non-uniform coarsening algorithm in (Aarnes et al., 2007a)), we here use only Cartesian models for implementational simplicity.

In contrast to what a more realistic model would require, we note that the fractures in our models are relatively thick, i.e., may not have the proper length-scale proportions. The large fracture thickness does not pose a fundamental constraint. The basic constraint in our approach is that the velocity field is computed on a grid where each fracture is a connected path or surface of cells. For the general case with thin arbitrarily oriented fractures we need to modify our fine grid model as follows:

1. Assume that an initial grid that does not resolve the fractures is given.
2. Trace the interface between the fractures and the matrix and employ the EFMS strategy to define a new fine grid that resolves the fractures.
3. Compute velocity on the new grid, e.g., using a mimetic finite difference method (F. Brezzi and Shashkov, 2005; F. Brezzi and Simoncini, 2005) or the multiscale mixed finite element method in (Aarnes et al., 2007b).

This approach avoids local grid refinement in the vicinity of fractures, which would normally give rise to a significant increase in the total number of cells.

This is possible due to the capability of mimetic finite difference methods to handle cells with arbitrary polygonal or polyhedral shapes.

5.2. NUMERICAL EXPERIMENTS

Each numerical experiment will be conducted with both a homogeneous and a heterogeneous background permeability field, both with porosity 0.1. Fractures are cracks in the rock that usually contain void space in which flow may take place. In this case the porosity is 1 inside the fractures, but from a modeling point of view it is more reasonable to try to model a large fracture zone. Due to the void space inside the fractures, the fracture zone will typically have larger porosity than the surrounding rock. Here we have assumed that the fracture zone has a porosity of 0.3 for simplicity. If fractures are filled with clay, then fractures will have very low porosity. Here the porosity of low permeable fractures is set to 0.001. The flow is, apart from in Section 5.6, driven by a so-called quarter-of-a-five-spot, i.e., by injecting water at constant rate in cell at the bottom left corner and producing at constant rate whatever reaches the cell in the opposite corner. Finally, apart from in Section 5.7, we run the simulations with $\mu_w = 0.1$ and $\mu_o = 1$.

The homogeneous model is a 100-by-100 Cartesian grid with equal dimensions in the x - and y -coordinate direction. The background permeability is set to 1 mD, the permeability in the high permeable fractures is 1 D, and the permeability in the low permeable fractures is 10^{-8} mD.

The heterogeneous model is a 60-by-220 Cartesian grid modeling layer 46 in Model 2 from the Tenth SPE Comparative Solution Project (Christie and Blunt, 2001), a model used for comparison and validation of upscaling techniques. The layer is from the lower Upper Ness formation, a fluvial formation characterized by a spaghetti of intertwined high permeable channels on a low permeable background. The dimensions in the x - and y -coordinate direction are 1200 ft and 2200 ft, respectively. Here the permeability of the high permeable fractures is 20 D (the maximum of the background permeability), while the permeability of the low permeable fractures is 10^{-8} mD, i.e., the same as for the homogeneous model. This is about five orders of magnitude less than the minimum of the background permeability.

Apart from in Sections 5.5 and 5.8, the initial coarse grid for the EFMS algorithm will be a 5-by-5 Cartesian grid for the homogeneous model and a 3-by-11 Cartesian grid for the heterogeneous model. The parameters V_{\min} and G_{\max} for the non-uniform coarsening algorithm are chosen such that the number of blocks in the non-uniform coarse grid is slightly less than the number of blocks in the grid generated using EFMS. For brevity we will sometimes use NUC when referring to the non-uniform coarsening algorithm.

Table I. Number of blocks and water-cut error for each of the coarse grids generated for the introductory models depicted in Figure 5(a) and Figure 5(b).

	Homogeneous background:		Heterogeneous background:	
	# of blocks	$e(w)$	# of blocks	$e(w)$
NUC grid	206	0.0245	273	0.0273
EFMS grid	236	0.1027	294	0.1208
Cartesian grid	400	0.1458	330	0.1684

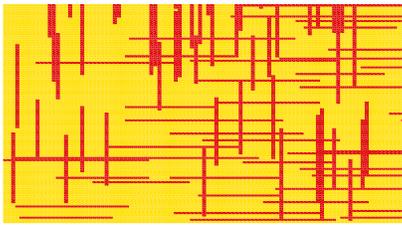
5.3. INTRODUCTORY EXAMPLE

Consider the models depicted in Figure 5(a) and Figure 5(b). In the homogeneous case, the EFMS algorithm transforms the 5-by-5 Cartesian grid into a grid with 236 blocks. If N denotes number of cells in the fine grid model, then using $V_{\min} = 13|\Omega|/N$ and $G_{\max} = 65|\Omega|g(\Omega)/N$ in the non-uniform coarsening algorithm generates a grid with 206 blocks. In the heterogeneous case, the EFMS algorithm transforms the 3-by-11 Cartesian grid into a grid with 294 blocks, and the non-uniform coarsening algorithm, using $V_{\min} = 15|\Omega|/N$ and $G_{\max} = 75|\Omega|g(\Omega)/N$, produces a grid with 273 blocks.

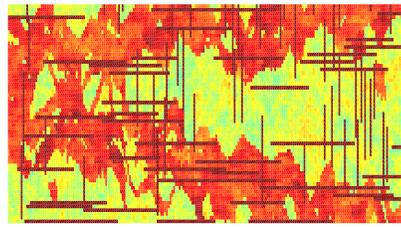
To illustrate capability to provide accurate saturation solutions, we perform a single simulation on each grid and compare the results with the corresponding results obtained on the fine grids and on moderately coarsened Cartesian grids (a 20-by-20 Cartesian grid for the homogeneous model and a 15-by-22 Cartesian grid for the heterogeneous model). Figures 5(c)–(j) show the respective saturation solutions at time $t = 0.48$ PVI and Figure 6 shows the corresponding water-cut curves. Table I displays the number of grid blocks in each coarse grid along with the corresponding water-cut errors.

The results shown in Figure 5, in Figure 6, and in Table I are representative for what type of solution accuracy one gets using the different types of coarse grids. The non-uniform coarse grids consistently give best accuracy, as will be further demonstrated in the following sections. The EFMS grid produces reasonably accurate solutions for the homogeneous model, for which the fracture distribution alone dictates the flow pattern, but less accurate results for the heterogeneous model where the flow pattern is also strongly influenced by the surrounding heterogeneous structures. Finally, the coarse Cartesian grids give lower accuracy than the two other coarse grids. This is due to the fact that the Cartesian grid resolves neither the fractures nor the flow channels in the heterogeneous model, and therefore smears out the saturation profile.

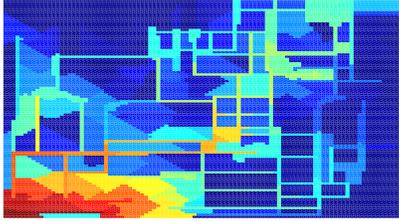
In the following sections we will make more rigorous and qualitative comparisons between solutions obtained on non-uniform coarse grids and



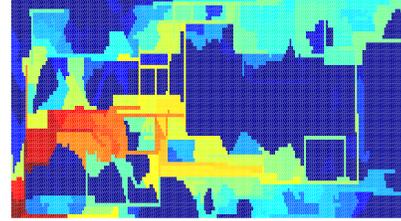
(a) Logarithm of permeability in a homogeneous model with 100 high permeable fractures.



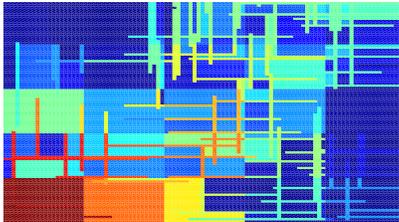
(b) Logarithm of horizontal permeability in the heterogeneous model with 100 fractures.



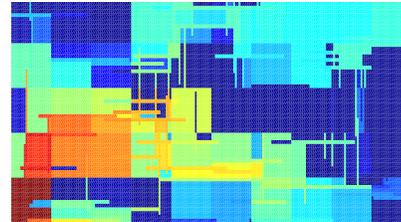
(c) Saturation on NUC grid with 206 blocks.



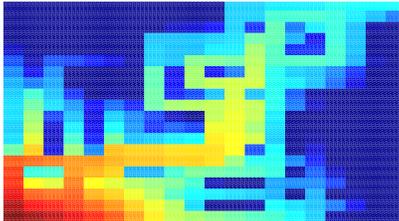
(d) Saturation on NUC grid with 273 blocks.



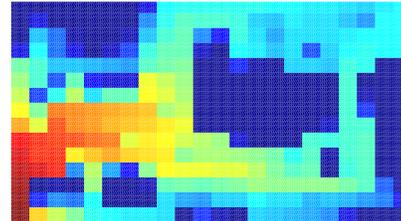
(e) Saturation on EFMS grid with 236 blocks.



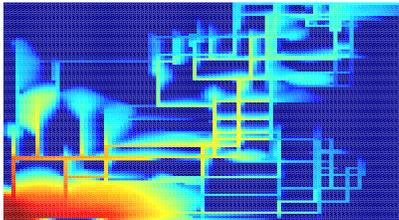
(f) Saturation on EFMS grid with 294 blocks.



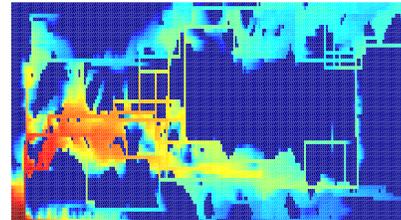
(g) Saturation on 20-by-20 Cartesian grid.



(h) Saturation on 15-by-22 Cartesian grid.



(i) Saturation on fine grid model.



(j) Saturation on fine grid model.

Figure 5. Fractured models and corresponding water saturation profiles at $t = 0.48$ PVI obtained using the numerical scheme (6) on different grids. Left: Saturation profiles for the homogeneous model. Right: Saturation profiles for the heterogeneous model.

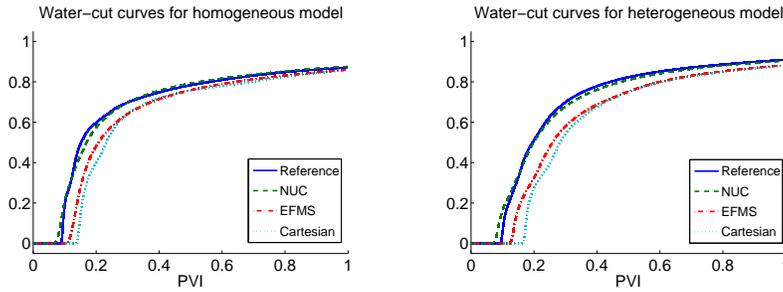


Figure 6. Water-cut curves (as functions of PVI) for the simulations in Figure 5.

Table II. Mean upscaling factors when varying number of fractures.

# fractures	Homogeneous background					Heterogeneous background				
	30	50	100	150	200	30	50	100	150	200
EFMS grid	107	73	42	31	24	104	76	45	33	26
NUC grid	116	74	46	33	26	109	80	50	35	35
% fracture cells	7.4	12.3	22.9	32.2	39.9	6.5	10.5	20.6	29.3	37.6

EFMS grids. To this end, we will for each case generate 25 different fracture distributions and compute the mean error averaged over all 25 realizations.

5.4. ROBUSTNESS WITH RESPECT TO NUMBER OF FRACTURES

In this section we attempt to assess how well the two coarsening algorithms perform when varying the number of fractures. The numerical experiments are conducted on models containing 30, 50, 100, 150, and 200 fractures.

Table II shows how the number of grid blocks in the EFMS grids depends on the number of fractures and the fracture distribution. Models with many fractures give grids with more grid blocks than models with a small number of fractures. Thus as the number of fractures increases, the upscaling factor decreases. As a consequence it is both difficult to predict the upscaling factor, and difficult to retain a high upscaling factor for models with many fractures.

The number of grid blocks in the grids generated using the non-uniform coarsening algorithm, on the other hand, is relatively stable when the parameters V_{\min} and G_{\max} are fixed. This makes it much easier to tune the upscaling factor with the non-uniform coarsening algorithm than with EFMS. In the current section the parameters are tuned to give a slightly higher upscaling factor than the average EFMS upscaling factor. The upscaling factor is therefore moderate for the cases with 150 and 200 fractures. We would like to

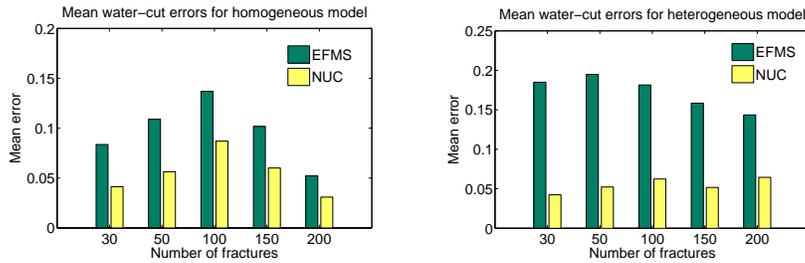


Figure 7. Mean water-cut errors for simulations with different number of fractures.

note, however, that the non-uniform coarsening algorithm is fully capable of retaining a high upscaling factor, and still produce accurate water-cut curves.

Figure 7 shows the mean water-cut error when varying the number of fractures. We see that the non-uniform coarsening algorithm consistently produces more accurate solutions than the EFMS algorithm. In addition, the EFMS grids produce substantially less accurate results for the heterogeneous models than for the homogeneous models, whereas the non-uniform coarse grids give roughly the same level of accuracy for both model types. These results therefore indicate that the non-uniform coarsening algorithm is more robust with respect to the number of fractures, allows easier tuning of the upscaling factor, and is less sensitive to the heterogeneous structures in the underlying model.

5.5. ROBUSTNESS WITH RESPECT TO DEGREE OF COARSENING

In this section we will assess robustness with respect to degree of coarsening while keeping the number of fractures fixed, i.e., we consider here only models with 100 fractures. Since the EFMS algorithm is deterministic, we can only vary the degree of coarsening by changing the initial coarse grid. For the homogeneous models we use three different Cartesian grids: 5-by-5, 10-by-10, and 20-by-20. Similarly, for the heterogeneous models we use a 3-by-11, a 6-by-22, and a 15-by-55 Cartesian grid. The parameters V_{\min} and G_{\max} for the non-uniform coarse grids are chosen accordingly.

Figure 8 shows the mean water-cut error for each degree of coarsening. The mean upscaling factors for both coarsening strategies are shown along the horizontal axes. Again we consistently obtain more accurate water-cut curves using the non-uniform coarse grids than with the EFMS grids. The error decays with both coarsening strategies as the grid is refined. However, by starting with grids with acceptable simulation size in the EFMS algorithm, one obtains limited upscaling factors. This indicates that the EFMS strategy is only useful when a coarse grid with a large upscaling factor can be provided. The non-uniform coarsening approach does not have this constraint.

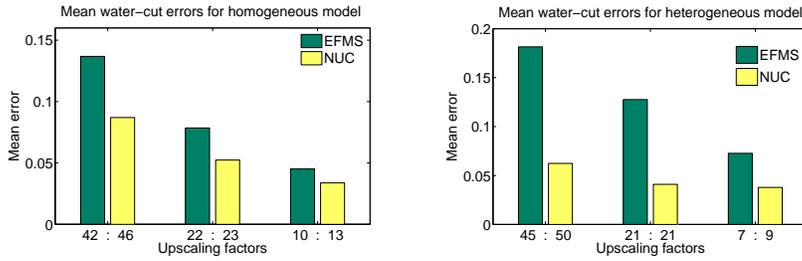


Figure 8. Mean water-cut errors for simulations with different degrees of coarsening.

5.6. ROBUSTNESS WITH RESPECT TO ALTERING WELL LOCATIONS

In this section we perform experiments to assess robustness with respect to choice of source and sink locations. In reservoir simulations, the sources and sinks stem from injection and production wells, respectively. We will therefore refer to each setup of source and sink locations as a well configuration.

During the life-cycle of an oil reservoir, the well configuration changes. Old wells may be shut down when they are no longer profitable, and new wells are drilled to produce from different parts of the reservoir or to improve drainage in flooded areas. In addition, individual chokes (devices that allow petroleum engineers to control inflow and outflow of well perforations) may be closed, opened or partly closed, as part of reservoir management. By modifying well configurations or altering choke settings one also changes the reservoir flow patterns.

The grid generated using the EFMS algorithm is independent of the flow, but the non-uniform coarsened grids depend on the initial velocity field, and therefore implicitly on the well configuration. Thus, it is relevant to ask if one needs to regenerate the grid when well configurations change. (The reservoir flow patterns also change dynamically due to mobility changes and gravity effects, but generally not as “dramatic” as when well configurations change). Here we make an effort to show that this is not the case, i.e., that the non-uniform coarse grid generated with respect to one given well configuration can be used to run simulations on models with other well configurations.

We consider the four well configurations depicted in Figure 9, henceforth referred to as well patterns A, B, C, and D. Injection wells (source locations) are labeled I and production wells (sink locations) are labeled P. All of the non-uniform coarse grid simulations are conducted using the grid generated from a velocity field corresponding to well pattern A.

Figure 10 shows the mean water-cut error for simulations with each well pattern. For all cases, the non-uniform coarsening strategy produces smaller errors than the EFMS strategy. The EFMS gridding strategy gives more or less consistent error for the homogeneous model and the heterogeneous model, respectively. The non-uniform coarsening strategy gives a little larger errors

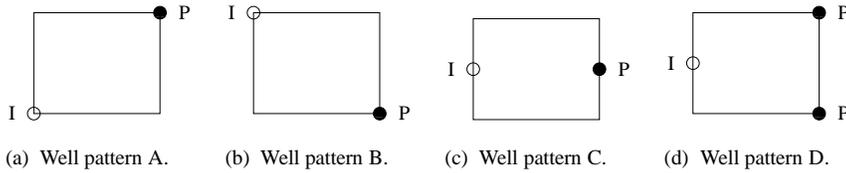


Figure 9. Well configurations used to assess robustness with respect well placement.

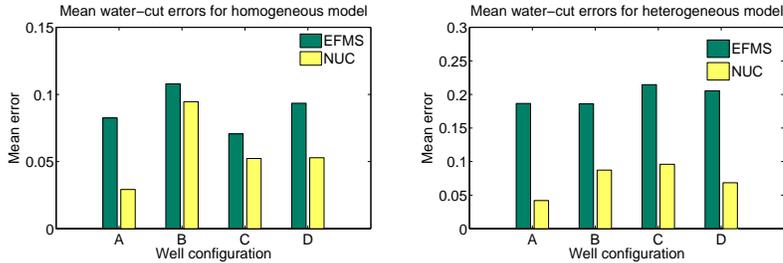


Figure 10. Mean water-cut errors for simulations with different well patterns. The non-uniform coarse grid generated with well pattern A is also used to run the simulations with well patterns B, C and D.

for well patterns B, C, and D than for well pattern A, but the fact that this algorithm still performs better than the EFMS algorithm illustrates that it is not very sensitive to what flow conditions are utilized when computing the velocity field used to generate the coarse grid. This supports the conclusion from (Aarnes et al., 2007a) that it is not necessary to regenerate the coarse grid during simulations with changing well configurations or boundary conditions. This property reflects that for any given well configuration, or set of boundary conditions, the high flow regions correspond to high permeable regions of good connectivity.

For the homogeneous fracture models this is true to a lesser extent than for the heterogeneous fracture models. As a result we observe that when the flow conditions change substantially, e.g., by switching from well pattern A to well pattern B so that the main direction of flow is perpendicular to the main direction of flow in the velocity field used to generate the non-uniform coarse grid, then there is less difference in accuracy between the non-uniform coarsening strategy and the EFMS strategy.

5.7. ROBUSTNESS WITH RESPECT TO VISCOSITY RATIO

The purpose of this section is to investigate robustness with respect to the type of displacement process, here quantified by the ratio between the displacing fluid (water) and the displaced fluid (oil). Thus, we introduce the viscosity ratio, defined by $\mu_r = \mu_w / \mu_o$. Conditions with $\mu_r > 1$ give rise to a so-called stable displacement process where the propagating front of the displacing

fluid is quite sharp and well defined. This is due to the fact that the total mobility is lower ahead of the front than behind the front. In the opposite case with $\mu_r < 1$ the mobility is higher ahead of the front than behind the front. This causes the displacing fluid to enter easily into unflooded areas, and one may observe a phenomenon called viscous fingering. Viscous fingering resembles the way that a wave floods a rocky shore with small “fingers” shooting out in different directions. For unstable displacement flows there is usually a rather slow and smooth incline in saturation from the front and backwards along the flowlines.

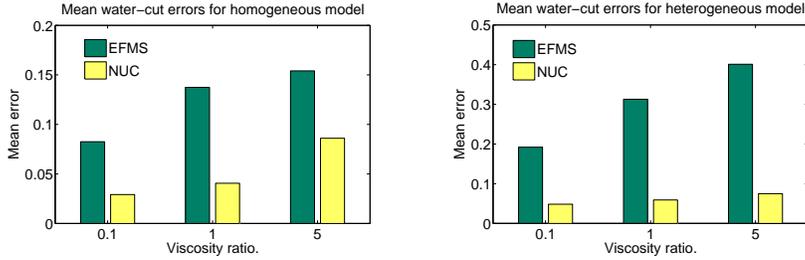
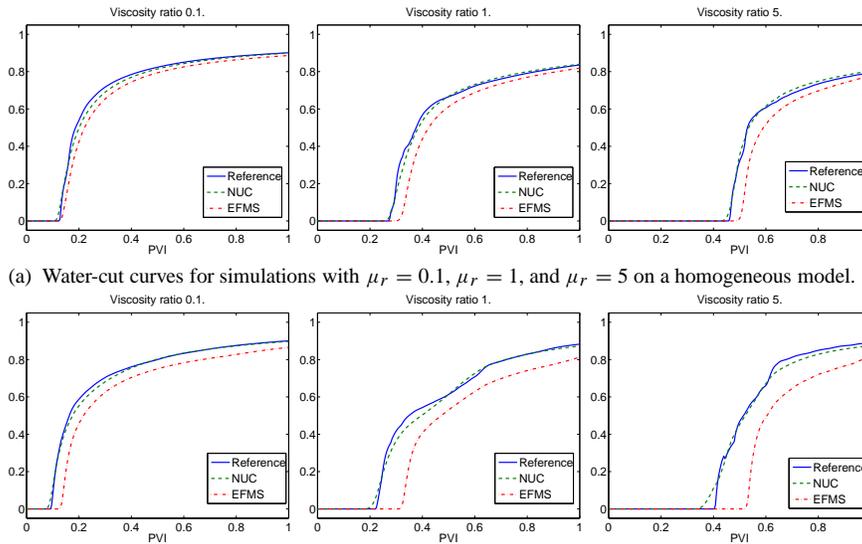


Figure 11. Mean water-cut errors for simulations with different viscosity ratios.



(a) Water-cut curves for simulations with $\mu_r = 0.1$, $\mu_r = 1$, and $\mu_r = 5$ on a homogeneous model.

(b) Water-cut curves for simulations with $\mu_r = 0.1$, $\mu_r = 1$, and $\mu_r = 5$ on a heterogeneous model.

Figure 12. Water-cut curves for simulations with different viscosity ratios.

We perform simulations with three different viscosity ratios: $\mu_r = 0.1$, $\mu_r = 1$, and $\mu_r = 5$, i.e., we consider both stable and unstable displacement flows. The histograms in Figure 11 compare water-cut errors obtained with the two different grid coarsening strategies. Once again we see that the errors

produced using the non-uniform coarse grids are substantially smaller than the errors produced using the EFMS coarsening strategy. For both algorithms we see that the errors increase with increasing viscosity ratio. This is to be expected, since high viscosity ratio flows give sharp fronts that are difficult to capture accurately on a coarse grid. Nevertheless, Figure 12 shows that the water-cut curves corresponding to the non-uniform coarse grids matches the reference water-cut curve very well. In contrast, the water-cut curves for the EFMS simulations are systematically below the reference curve.

5.8. ROBUSTNESS WITH RESPECT TO PRESENCE OF LOW PERMEABLE FRACTURES

The purpose of this section is to study how the accuracy of saturation solutions obtained using the different coarsening strategies is affected by the presence of low permeable fractures. To this end we generate models with 20 low permeable fractures and 100 high permeable fractures. The high permeable fractures provide preferential flow paths whereas the low permeable fractures act as flow barriers.

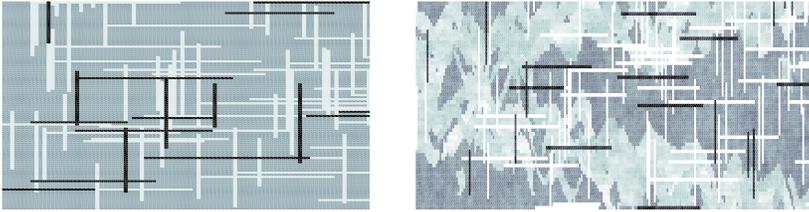


Figure 13. Homogeneous and heterogeneous model with 20 low permeable fractures (dark color) and 100 high permeable fractures (light color).

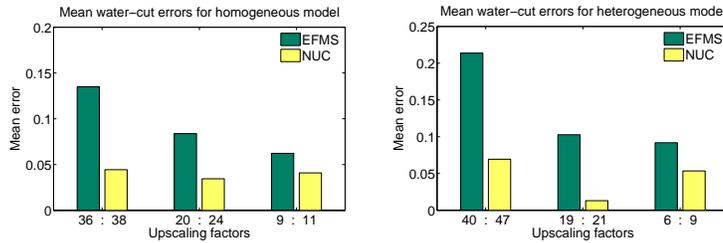


Figure 14. Mean water-cut errors for simulations on models with 20 low permeable and 100 high permeable fractures.

Figure 13 displays a fracture distribution for each model and Figure 14 shows the mean water-cut errors for both coarsening algorithms. The results confirm the trend that we have observed throughout this section, i.e., that the non-uniform coarsening algorithm gives more accurate results for both models, that the results are substantially more accurate for the heterogeneous

model, and that the upscaling factor for the EFMS algorithm is limited if there are many fractures, or if the initial coarse grid is not sufficiently coarse.

6. Inclusion of effects from gravity and capillary pressure

The mathematical model presented in Section 2, the discretization schemes in Section 4, and the corresponding numerical results presented in Section 5 all disregard gravity and capillary pressure. Although gravity and capillary pressure are often “second order effects” for large scale simulations, e.g., most field-scale reservoir flow regimes are primarily pressure driven, gravity and capillary forces should not be neglected. Indeed, gravity may be important during a transition period with gravity segregation and capillary pressure has a local diffusive effect that tends to smooth the saturation front. It is especially important to account for capillary pressure effects in strongly heterogeneous systems for which the capillary forces pull flow into low permeable regions.

The purpose of this section is to describe how to extend the current simulation methodology to account for gravity and capillary forces. The associated pressure equation can still be solved with the same discretization method, e.g., a mimetic finite difference method on the fine grid (F. Brezzi and Shashkov, 2005; F. Brezzi and Simoncini, 2005) or a multiscale method (Arbogast, 2000; Jenny et al., 2003; Aarnes, 2004) on a coarse grid. Hence, we will only focus on the discretization of the saturation equation, i.e., on how to extend the discretization scheme (6) to account for gravity and capillary forces.

6.1. MATHEMATICAL FORMULATION

With gravity included Darcy’s law becomes:

$$v_j = -\lambda_j(\nabla p_j + g\rho_j\nabla z), \quad j = o, w, \quad (7)$$

where g is the magnitude of acceleration of gravity, ρ_j is the density of phase j , and z is the vertical coordinate, i.e., $z = x \cdot n_z$ where n_z is the unit normal in the vertical direction pointing upwards. Upon performing simple algebraic calculations, and introducing the capillary pressure $p_{cow} = p_o - p_w$, we get

$$v_w = f_w(v + \lambda_o\nabla p_{cow} + \lambda_o g(\rho_o - \rho_w)\nabla z). \quad (8)$$

Inserting this expression into (1) we obtain the following saturation equation

$$\phi \frac{\partial S}{\partial t} + \nabla \cdot [f_w(v + \lambda_o\nabla p_{cow} + \lambda_o g(\rho_o - \rho_w)\nabla z)] = q_w. \quad (9)$$

It is common to call $\nabla \cdot (f_w v)$ the viscous term, $\nabla \cdot (f_w \lambda_o g(\rho_o - \rho_w)\nabla z)$ the gravity term, and $\nabla \cdot (f_w \lambda_o \nabla p_{cow})$ the diffusion term. The viscous term usually dominates in field-scale reservoir simulation, whereas the diffusion term

dominates in small scale simulations. The gravity term can be dominant in large scale simulations with very low flow velocity, e.g., when modeling what happens to CO₂ after it has been injected into an aquifer. We target primarily applications where the viscous term is dominant. In particular we assume that capillary pressure effects are adequately modeled through operator splitting, i.e., instead of discretizing (9) we discretize the following system:

$$\phi \frac{\partial S}{\partial t} + \nabla \cdot (f_w v + f_w \lambda_o g(\rho_o - \rho_w) \nabla z) = q_w, \quad (10)$$

$$\phi \frac{\partial S}{\partial t} + \nabla \cdot \left(f_w \lambda_o \frac{\partial p_{cow}}{dS} \nabla S \right) = 0. \quad (11)$$

We have used the capillary pressure function (Espedal and Karlsen, 2000):

$$p_{cow} = \epsilon 0.9 \phi^{-0.9} K^{-0.5} \frac{1-S}{\sqrt{S}}. \quad (12)$$

Here ϵ is a parameter that can be used to adjust the amount of diffusion.

6.2. DISCRETIZATION

To discretize (10) on coarse grids where each block consists of a connected collection of cells in an underlying fine grid with interfaces γ_{ij} we employ the following scheme:

$$S_m^{n+\frac{1}{2}} = S_m^n + \frac{\Delta t}{\int_{B_m} \phi dx} \left[\int_{B_m} q_w(S^{n+\frac{1}{2}}) dx - \sum_{\gamma_{ij} \subset \partial B_m} \left(V_{ij}(S^{n+\frac{1}{2}}) + G_{ij}(S^{n+\frac{1}{2}}) \right) \right].$$

Here $V_{ij}(S)$ is defined as in Section 4 and

$$G_{ij}(S) = g(\rho_o - \rho_w) |\gamma_{ij}| \frac{\lambda_w(S^+) \lambda_o(S^-)}{\lambda_w(S^+) + \lambda_o(S^-)} \nabla z \cdot n_{ij},$$

where n_{ij} is the unit normal on γ_{ij} pointing from T_i to T_j , and

$$S^+ = \max\{S|_{T_i s_{ij}}, -S|_{T_j s_{ij}}\} \quad \text{and} \quad S^- = \max\{S|_{T_j s_{ij}}, -S|_{T_i s_{ij}}\},$$

where $s_{ij} = \text{sign}(\rho_o - \rho_w) \text{sign}(n_{ij} \cdot n_z)$. Hence, λ_o and λ_w are upstream weighted with respect to the gravity driven flow of oil and water, respectively. This is the standard way to discretize the gravity term in reservoir simulators.

The diffusion equation (11) takes the following form:

$$\phi \frac{\partial S}{\partial t} = \nabla \cdot d(S) \nabla S, \quad (13)$$

where $d(S) = -f_w \lambda_o \frac{\partial p_{cow}}{\partial S}$ is a non-negative function. This is a parabolic equation with a degenerate diffusion term. For the time discretization we employ a semi-implicit backward Euler method:

$$\phi S^{n+1} = \phi S^{n+1/2} + \Delta t \nabla \cdot d(S^{n+1/2}) \nabla S^{n+1}.$$

For the spatial discretization we use a Galerkin projection.

6.2.1. Galerkin projection of capillary diffusion

Let \mathbf{D} be a symmetric and semi-positive definite matrix that stems from a cell centered finite-difference discretization of the semi-elliptic operator $L = -\nabla \cdot d \nabla$. Then, to modify $S^{n+1/2}$ to account for capillary diffusion we may solve the following symmetric and positive definite system:

$$(\Phi + \Delta t \mathbf{D}) S^{n+1} = \Phi S^{n+1/2}, \quad (14)$$

where $\Phi = \text{diag}(\phi)$. However, since the viscous and gravity terms are discretized on the coarse grid it is undesirable to have to solve a fine grid system for the diffusion term. Moreover, for grids with complex grid geometries it is not possible to apply standard methods for discretizing L . Devising a numerical method tailored for discretizing L on coarse grids with complex block geometries is not within the scope here. Instead we demonstrate that the capillary diffusion can be modeled on coarse grids with a Galerkin projection.

To this end, let $\mathbf{R} = r_{ij}$ where

$$r_{ij} = \begin{cases} 1 & \text{if cell } i \text{ in the fine grid is contained in block } j \text{ in the coarse grid,} \\ 0 & \text{otherwise.} \end{cases}$$

Hence, if S_c represents saturation on the coarse grid, then $S_f = \mathbf{R} S_c$ is the corresponding interpolated saturation on the fine grid. Moreover, if S^{n+1} is the solution of (14) with $S^{n+1/2} = \mathbf{R} S_c^{n+1/2}$, then the Galerkin projection of S^{n+1} onto the space of piecewise constant functions on the coarse grid is the solution S_c^{n+1} of the following system:

$$[\Phi_c + \Delta t \mathbf{D}_c] S_c^{n+1} = \Phi_c S_c^{n+1/2}, \quad (15)$$

where $\mathbf{D}_c = \mathbf{R}' \mathbf{D} \mathbf{R}$ and $\Phi_c = \mathbf{R}' \Phi \mathbf{R}$.

The current Galerkin projection is an orthogonal projection with respect to the norm $\|S\|_{\mathbf{A}} = (S, \mathbf{A} S)^{1/2}$, where $\mathbf{A} = \Phi + \Delta t \mathbf{D}$. In other words,

$$\|\mathbf{R} S_c^{n+1} - S^{n+1}\|_{\mathbf{A}} = \arg \min_{S_c} \|\mathbf{R} S_c - S^{n+1}\|_{\mathbf{A}},$$

where S^{n+1} is the solution of (14) with $S^{n+1/2} = \mathbf{R} S_c^{n+1/2}$.

To discretize the semi-elliptic operator L on the fine grid to obtain $\mathbf{D} = d_{ij}(S)$ we employ a two-point flux-approximation finite volume scheme. Thus,

since we consider Cartesian grids, we have

$$d_{ij}(S) = - \int_{\gamma_{ij}} d(S) \nabla S \cdot n_{ij} ds \approx |\gamma_{ij}| \tilde{d}(S_i, S_j) \frac{S_i - S_j}{|x_i - x_j|}, \quad (16)$$

where x_i and x_j are the cell centers in T_i and T_j , respectively, and $\tilde{d}(S_i, S_j)$ is a suitable average of $d(S_i)$ and $d(S_j)$.

Note that if S is saturation on a coarse grid, then modeling capillary diffusion using (14) or (15) with \mathbf{D} defined by (16) will in general overestimate the diffusion. This is because the saturation gradient in (16) is computed at the fine grid level, whereas the saturation values represent net saturations within the coarse grid blocks. The capillary diffusion therefore scales with the ratio of the size of the coarse grid blocks relative to the size of the fine grid cells. Thus, ‘‘on average’’ the diffusion should be damped by a factor $(N_b/N_c)^{1/d}$, where N_b denotes number of blocks, N_c denotes number of cells, and d is the spatial dimension. This suggests that one can model capillary diffusion more correctly by using the following scaled Galerkin projection approach:

$$\left[\Phi_c + \Delta t \left(\frac{N_b}{N_c} \right)^{1/d} \mathbf{D}_c \right] S_c^{n+1} = \Phi_c S_c^{n+1/2}. \quad (17)$$

More accurate ways of damping the capillary diffusion, e.g., by estimating saturation gradients across each coarse grid interface, will be pursued in further research. The main idea here will be to exploit some information about the coarse grid geometry. Essentially the current approach approximates the saturation gradient between grid block i and grid block j by

$$\nabla S = \frac{S_j - S_i}{\Delta x},$$

where Δx is a global constant. In further work the saturation gradient will be approximated more accurately by replacing Δx with the distance between the centers of block i and j , respectively.

6.3. NUMERICAL EXAMPLE

The purpose of this section is to illustrate the effect of modeling capillary diffusion using the Galerkin projections (15) and (17) relative to modeling capillary diffusion on the fine grid. To this end we have performed simulations on both a homogeneous fracture model and a heterogeneous fracture model. We consider cases with relatively strong capillary diffusion, i.e., more diffusion than one normally observes in field cases. The source term q , which models a quarter-of-a-five-spot, is chosen so that the balance between the viscous term and the diffusion term is independent of the reservoir dimensions.

Table III. Errors for the water-cut curves depicted in Figures 16 and 18.

	Homogeneous model			Heterogeneous model		
	fine grid	G. proj.	scaled G. proj.	fine grid	G. proj.	scaled G. proj.
EFMS	0.071	0.077	0.028	0.105	0.121	0.065
NUC	0.047	0.055	0.079	0.075	0.121	0.021

Figures 15 and 17 display saturation profiles at 0.2 PVI with capillary diffusion modeled on the fine grid (Figures 15(b)–(d) and 17(b)–(d)), with the Galerkin projection (Figures 15(e)–(f) and 17(e)–(f)), and with the scaled Galerkin projection (Figures 15(g)–(h) and 17(g)–(h)). The corresponding water-cut curves are depicted in Figures 16 and 18, and the associated water-cut errors are listed in Table III. The EFMS grids used in the simulations contain 256 blocks for the homogeneous model and 228 blocks for the heterogeneous model. The corresponding NUC grids contain 230 blocks for the homogeneous model and 205 blocks for the heterogeneous model.

Comparing the saturation plots in Figures 15 and 17 with the saturation plots in Figure 5 (at 0.48 PVI) we clearly see that capillary diffusion has strongly influenced the flow in the simulations in this section. Qualitatively we see that modeling diffusion on the fine grid and with the Galerkin projection gives qualitatively similar results. As expected, the results show evidence of too much diffusion. For instance, unlike in Figure 12, we see that the water-cut curves for the non-uniform coarsening strategy now systematically lie below the reference water-cut curve. This indicates that too much water flows into the low permeable matrix and hence delays its arrival at the producer.

When using the scaled Galerkin projection we see that the water-cut curves are shifted to the left. This stems from the fact that the diffusion effects have been damped, giving rise to earlier arrival at the producer. This gives a significant reduction in the water-cut errors except for the non-uniformly coarsened homogeneous model for which the water-cut error increases. The rise in water-cut error for this case might reflect that a majority of the flow here takes place in the fracture web, giving rise to large gradients between the fracture and the matrix. Hence, if the flow in the fractures is modeled accurately, then it may be better to model the diffusion on the fine grid. For the EFMS grids we know from the results in Section 5 that it generally gives delayed arrival times. Thus, for these grids the shift towards earlier arrival times serves to correct for other modeling errors.

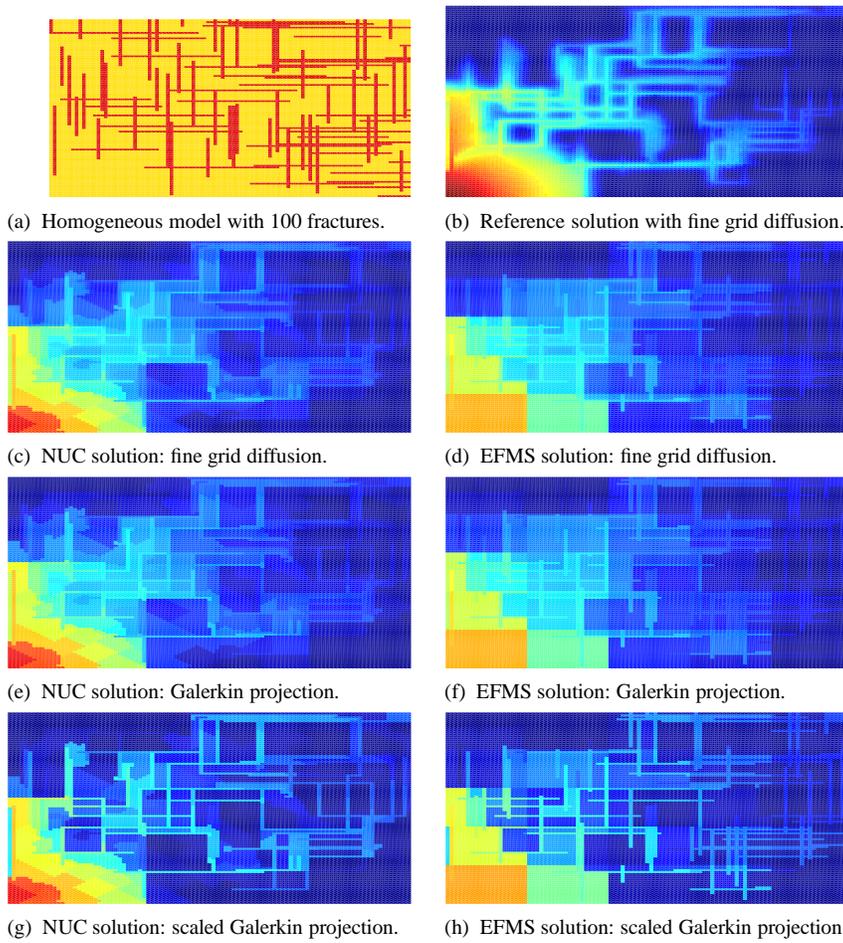


Figure 15. Saturation profiles at 0.2 PVI obtained by modeling capillary diffusion on the fine grid, using the Galerkin projection, and using the scaled Galerkin projection.

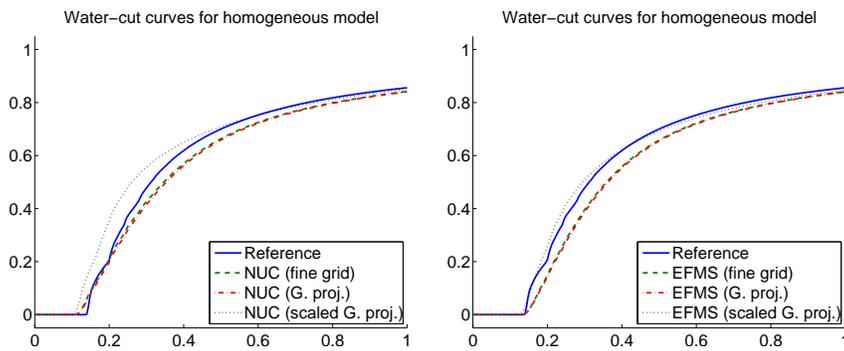


Figure 16. Water-cut curves for the simulations displayed in Figure 15.

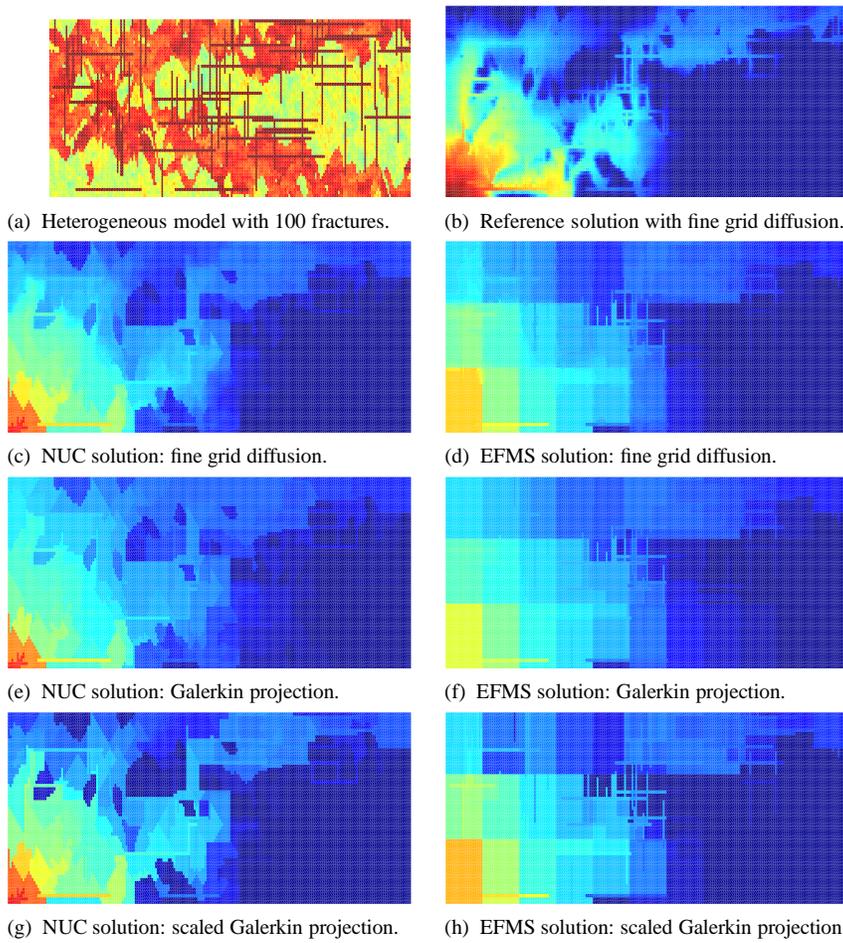


Figure 17. Saturation profiles at 0.2 PVI obtained by modeling capillary diffusion on the fine grid, using the Galerkin projection, and using the scaled Galerkin projection.

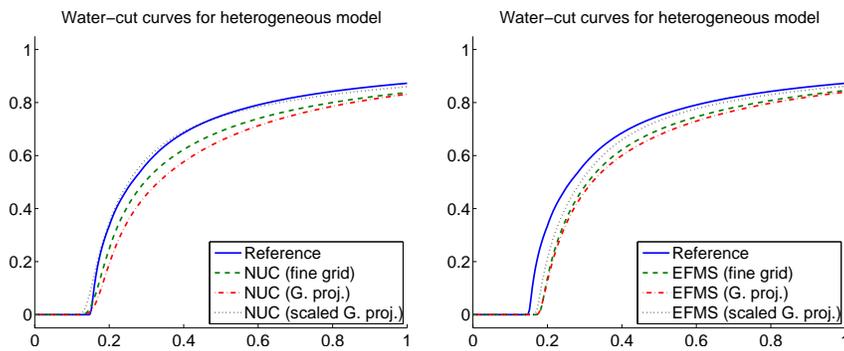


Figure 18. Water-cut curves for the simulations displayed in Figure 17.

7. Concluding remarks

In this paper we have presented two strategies for generating coarse simulation grids. The objective has been to design grids that resolve flow in fractured porous media more accurately than conventional simulation models. The first strategy, called explicit fracture-matrix separation (EFMS), generates coarse grids where the fracture web is separated from an underlying coarse grid. The second strategy is based on the non-uniform coarsening algorithm from (Aarnes et al., 2007a). Both algorithms produce unstructured coarse grids.

A series of two-phase flow simulations where the saturation is modeled on the produced coarse grids are performed. The pressure equation is solved on a fine grid in which the fractures are represented as a path of adjacent cells. Thus, we do not model fractures as lower-dimensional elements, and we do not employ dual-porosity dual-continuum concepts. Robustness of the coarsening algorithms is assessed by comparing the produced water-cut curves with a reference solution for cases with different fracture distributions and flow parameters. The simulations support the following conclusions:

- Both coarsening algorithms give more accurate solutions than one obtains by modeling saturation on conventional coarse grids. The non-uniform coarsening approach consistently produces the most accurate solutions. The EFMS strategy gives poor accuracy when the flow is strongly influenced by underlying heterogeneous structures.
- The non-uniform coarsening strategy is quite robust, e.g., with respect to number of fractures, degree of coarsening, well locations, viscosity ratio, and fracture permeability, and it is easy to tune the upscaling factor. It is difficult to control the upscaling factor with the EFMS strategy. In particular, the upscaling factor for the EFMS algorithm is limited if there are many fractures or if the initial coarse grid is not sufficiently coarse.
- The non-uniform coarsening algorithm assumes no prior knowledge of the fractures provided their presence is reflected in the geological model. The EFMS strategy assumes that the “fracture cells” are prescribed.
- Capillary diffusion can be modeled on unstructured coarse grids with complex block geometries using a damped Galerkin projection.

In summary, the non-uniform coarsening algorithm produces the most accurate results, and hence seems to provide the best tool for modeling flow in fractured reservoirs.

Reservoirs with large scale fractures that strongly influence the flow (e.g., carbonate reservoirs) pose a challenge to reservoir engineers as simulation models often fail to be predictive. This is partly because current simulation models are not sufficiently flexible to allow the grid size or grid flexibility to

get adequate resolution of the fractures. The non-uniform coarsening strategy presented in this paper offers an alternative by producing coarse grids that are tuned to the flow patterns dictated by the fractures and the matrix heterogeneity. As such we believe that it provides a tool that can facilitate modeling and simulation of transport phenomena in fractured porous media.

References

- Aarnes, J. E.: 2004, 'On the use of a mixed multiscale finite element method for greater flexibility and increased speed or improved accuracy in reservoir simulation'. *Multiscale Model. Simul.* **2**(3), 421–439.
- Aarnes, J. E., V. L. Hauge, and Y. Efendiev: 2007a, 'Coarsening of three-dimensional structured and unstructured grids for subsurface flow'. *Adv. Water Resour.* **30**(11), 2177–2193.
- Aarnes, J. E., S. Krogstad, and K.-A. Lie: 2007b, 'Multiscale mixed/mimetic methods on corner-point grids'. *Comput. Geosci.* to appear.
- 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.
- Barenblatt, G. I., I. P. Zheltov, and I. N. Kochina: 1960, 'Basic concepts in the theory of seepage of homogeneous liquids in fissured rocks'. *J. Appl. Math. Mech. (USSR)* **24**(5), 1286–1303.
- Cao, J. and P. K. Kitandis: 1999, 'Adaptive-grid simulation of groundwater flow in heterogeneous aquifers'. *Adv. Water Resour.* **22**(7), 681–696.
- Christie, M. A. and M. J. Blunt: 2001, 'Tenth SPE comparative solution project: A comparison of upscaling techniques'. *SPE Reservoir Eval. Eng.* **4**, 308–317.
- Durlofsky, L. J., R. C. Jones, and W. J. Milliken: 1997, 'A nonuniform coarsening approach for the scale-up of displacement processes in heterogeneous porous media'. *Adv. Water Res.* **20**, 335–347.
- Espedal, M. S. and K. H. Karlsen: 2000, *Numerical solution of reservoir flow models based on large time step operator splitting algorithms*, pp. 9–77, No. 1734 in *Lecture Notes in Mathematics*. Berlin Heidelberg: Springer-Verlag, filtration in porous media and industrial application edition.
- F. Brezzi, K. L. and M. Shashkov: 2005, 'Convergence of mimetic finite difference methods for diffusion problems on polyhedral meshes'. *SIAM J. Num. Anal.* **43**, 1872–1895.
- F. Brezzi, K. L. and V. Simoncini: 2005, 'A family of mimetic finite difference methods on polygonal and polyhedral meshes'. *Math. Models Methods Appl. Sci.* **15**, 1533–1553.
- Garcia, M., A. Journal, and K. Aziz: 1992, 'Automatic grid generation and for modeling reservoir heterogeneities'. *SPE Reservoir Engineering* pp. 278–284.
- He, C. and L. J. Durlofsky: 2006, 'Structured flow-based gridding and upscaling for modeling subsurface flow'. *Adv. Water Resour.* **29**(12), 1876–1892.
- 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.
- Karimi-Fard, M., L. J. Durlofsky, and K. Aziz: 2003, 'An efficient discrete fracture model applicable for general purpose reservoir simulators'. Paper SPE 79699 presented at Reservoir Simulation Symposium.
- Karimi-Fard, M. and A. Firoozabadi: 2001, 'Numerical Simulation of Water Injection in 2D Fractured Media Using Discrete-Fracture Model'. Paper 71615 presented at SPE Annual Technical Conference and Exhibition.

- Kazemi, H.: 1969, 'Pressure transient analysis of naturally fractured reservoirs with uniform fracture distribution'. *Society of Pet. Eng. Journal* **9**(4), 451–462.
- King, M. J., K. S. Burn, P. Wang, V. Muralidharan, F. Alverado, X. Ma, and A. Datta-Gupta: 2005, 'Optimal Coarsening of 3D Reservoir Models for Flow Simulation'. *SPE* **95759**.
- Raviart, P. A. and J. M. 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.
- Reichenberger, V., H. Jakobs, P. Bastian, and R. Helmig: 2006, 'A mixed-dimensional finite volume method for two-phase flow in fractured porous media'. *Adv. Water Res.* **29**(7), 1020–1036.
- Schrag, D. P.: 2007, 'Preparing to capture carbon'. *Science* **315**, 812–813.
- Warren, J. E. and P. J. Root: 1963, 'The behavior of naturally fractured reservoirs'. *Society of Pet. Eng. Journal* **3**, 245–255.