

Coarsening of three-dimensional structured and unstructured grids for subsurface flow*

Jørg Espen Aarnes^{a,*} Vera Louise Hauge^a

^aSINTEF ICT, P.O. Box 124 Blindern, N-0314 Oslo, Norway

Yalchin Efendiev

Department of Mathematics, Texas A & M University, College Station, TX 77843

Abstract

We present a generic, semi-automated algorithm for generating non-uniform coarse grids for modeling subsurface flow. The method is applicable to arbitrary grids and does not impose smoothness constraints on the coarse grid. One therefore avoids conventional smoothing procedures that are commonly used to ensure that the grids obtained with standard coarsening procedures are not too rough. The coarsening algorithm is very simple and essentially involves only two parameters that specify the level of coarsening. Consequently the algorithm allows the user to specify the simulation grid dynamically to fit available computer resources, and, e.g., use the original geomodel as input for flow simulations. This is of great importance since coarse grid-generation is normally the most time-consuming part of an upscaling phase, and therefore the main obstacle that has prevented simulation workflows with user-defined resolution. We apply the coarsening algorithm to a series of two-phase flow problems on both structured (Cartesian) and unstructured grids. The numerical results demonstrate that one consistently obtains significantly more accurate results using the proposed non-uniform coarsening strategy than with corresponding uniform coarse grids with roughly the same number of cells.

Key words: two-phase flow, grid generation, porous media, upscaling

* The research at SINTEF was partially supported by Shell and the Research Council of Norway under grants 158908/420 and 175962/S30. The research of Y. Efendiev was partially supported by NSF grants DMS-062113 and EIA-0540138 and DOE grant DE-FG02-05ER25669.

* Corresponding author.

Email addresses: Jorg.Aarnes@sintef.no (Jørg Espen Aarnes),
Vera.Louise.Hauge@sintef.no (Vera Louise Hauge), Efendiev@math.tamu.edu (Yalchin Efendiev).

1 Introduction

To visualize and quantify fluid flow in heterogeneous subsurface reservoirs (e.g., ground-water and petroleum reservoirs), it is common to perform numerical flow simulations. To this end, statistical descriptions of the subsurface formations are built using geomodeling software tools. Unfortunately the number of cells in the geological grid-model (geomodel) that estimate the spatial distribution of reservoir parameters very often exceed the capabilities of flow simulators. Hence, rather than using the original geomodel as input, current simulators normally take as input coarsened and simplified models derived through an upscaling process. These upscaled models consist of a coarsened grid accompanied by a corresponding set of reservoir parameters. The simulation model then consists of the upscaled geomodel combined with an elliptic (or parabolic) equation modeling pressure and flow velocity, and a set of mass balance equations modeling the transport. In this paper we focus on immiscible flow so that by transport we refer to movement of phases – aqueous, liquid, or vapor – and not dispersion of different components within each phase.

Coarsened grids obtained by upscaling are usually constrained to be on a specific grid format, e.g., corner-point grid format (“logically hexahedral grids”) or PEBI grid format (“orthogonal Voronoi grids”). This is partly due to grid-constraints associated with the numerical methods employed to discretize the governing system of partial differential equations, in particular the pressure equation, and partly because most techniques for upscaling permeability are designed for grids with hexahedral (shoe-box shaped) grid blocks. The widely used two-point finite volume method, for instance, is designed for so-called K -orthogonal grids, meaning that the connections between cell centers are K -perpendicular to the cell faces. In general, grid constraints make it very difficult to build grids that capture the important features in the underlying geomodels in an appropriate way. Indeed, upscaled subsurface flow models often fail to capture important small scale structures that have profound impact on the resulting flow regime, such as narrow high-flow channels and shale barriers (low permeable obstacles).

Grid generation procedures that tune the coarse grid to dominant geological features have been proposed by several authors, see e.g., [13,15,19] and the references therein. The basic goal in these methods is often to generate grids that are more finely gridded in regions of particular importance, e.g., around wells and in high flow regions. However, although these techniques do offer better resolution in regions of interest, and in general provide more robust upscaled models, they tend to suffer from lack of grid flexibility. Flow-based grid generation approaches, for instance, are often limited to two-dimensional applications. Moreover, because flow paths in heterogeneous formations may be highly irregular, flow-based grid generation approaches generally require a grid-smoothing procedure. Finally, because grid lines in the coarsened grid usually do not align with grid lines in the underlying geomodel grid, it is often necessary to perform resampling of geological data.

In this paper we propose a semi-automated grid-coarsening strategy for subsurface flow applications that can be applied to both structured and unstructured geomodel grids.

The basic idea is to develop a generic all-applicable version of the non-uniform coarsening approach introduced by Durlofsky, Jones, and Milliken [10]. Here the coarse grid is generated by selectively grouping cells in the geomodel. We require only that the cells in the coarse grid are connected, have at least some minimum volume, and that the total flow through each coarse cell is bounded above. Because only the transport is computed on this grid, we avoid conventional constraints (e.g., whereas the approach of Durlofsky et al. was based on simply removing grid lines in two-dimensional Cartesian grids, thus obtaining a non-uniform Cartesian grid, we here allow coarse cells with arbitrary shape). The transport is modeled using a two-scale version of a first-order upstream weighted finite volume method. This scheme requires, in principle, only that the grid cells are connected and that the velocity field is mass conservative on this grid.

To model pressure and velocity we employ here a mimetic finite difference method on the geomodel. Hence, we propose that the pressure equation is solved on the geomodel grid, whereas the phase-transport equation is solved on a coarsened grid. This can be justified for oil-water two-phase flow scenarios because the pressure generally changes at a moderate pace so that one can use much larger time-steps for the pressure equation than for the phase-transport equations [8]. However, for flows with strong dynamics, e.g., three phase flow with a separate gas-phase, and for very large geomodels, it may be too computationally expensive to solve the pressure equation directly on the geomodel. For these cases, a more efficient alternative is to use a multiscale method capable of providing mass-conservative velocity fields on the geomodel, e.g., [1,5,14]. The key observation is that a velocity field that is mass conservative on the geomodel is also mass conservative on *any* grid with cells that consist of a connected collection of cells in the geomodel.

We will start by introducing the model problem, which is a model for incompressible and immiscible two-phase flow. Next, in Section 3 we present the non-uniform coarsening algorithm and discuss applicability and limitations and possible implications. We also provide some analysis to give an insight into why the coarsening algorithm provides more accurate simulation results. In Sections 4 and 5 we describe the discretization of the model equations and report the results of the numerical experiments, respectively. The examples range from relatively simple two-dimensional Cartesian models to models with complex channelized heterogeneous structures and unstructured corner-point grid models. Finally, we review the main observations and make some concluding remarks in Section 6.

2 Mathematical model

We consider immiscible and incompressible two-phase flow without gravity and capillary pressure effects. The equations are derived from conservation of mass for each phase:

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

where the phase velocities v_j are given by Darcy's law:

$$v_j = -\lambda_j(S_j)K\nabla p_j. \quad (2)$$

Here ϕ is the porosity, S_j is the j -phase saturation (fraction of the void occupied by phase j) and q_j is a source (or sink) term. In Darcy's law, K is the permeability tensor, p_j is the phase pressure, and $\lambda_j(S_j) = k_{rj}(S_j)/\mu_j$, where k_{rj} and μ_j are the relative permeability and viscosity of phase j respectively. The relative permeability models the reduced conductivity of a phase due to the presence of other phases and is assumed to be function of the saturations only. The porosity is taken to be constant, i.e., the rock is assumed to be rigid and non-deforming.

Let the two phases be oil and water ($j = o, w$). Since we neglect capillary pressure effects so that $\nabla p_o = \nabla p_w$, we assume $p_o = p_w = p$. Then the Darcy equations without gravity effects combined with conservation of mass yields the *pressure equation*:

$$\begin{aligned} v &= -\lambda(S_w, S_o)K\nabla p, \\ \nabla \cdot v &= q, \end{aligned} \quad (3)$$

where $v = v_w + v_o$, $\lambda = \lambda_w + \lambda_o$, and $q = q_w + q_o$.

Assume now that the two phases fill the void space completely, i.e., that $S_w + S_o = 1$, and introduce the water *fractional flow* $f_w(S_w) = \lambda_w(S_w)/\lambda(S_w, 1-S_w)$. Then the conservation equation for water, henceforth called the *saturation equation*, reads as follows:

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

The system of equations (3)–(4) will be solved using a sequential splitting, i.e., the pressure equation is solved at the current time-step using saturation values from the previous time-step. Moreover, for ease of notation, we will henceforth drop the w -subscript of S_w .

3 Grid coarsening strategy

In the current section we seek to develop a grid coarsening strategy based on the following principles (B refers to a cell in the coarsened grid, henceforth called a block):

- The grid should separate high flow regions from low flow regions.
- The volume of a cell in the coarse grid should not be too small.
- The total amount of flow through a single cell in the coarse grid should not be too large.

Hence, after separating high-flow regions from low-flow regions, we will refine blocks that are too large, and merge blocks that are too small with a neighboring block. The refinement strategy is based on imposing an upper bound on the total flow through each block rather than a bound on the total volume. Thus, the coarsening algorithm involves two parameters

that should be tuned to give the intended degree of coarsening: N_L (lower bound on volume of blocks), and N_U (upper bound on total amount of flow in blocks). To quantify the amount of flow through each block we introduce the following monitoring function:

$$g(v) = [\log |v| - \min(\log |v|) + 1].$$

The various steps in the coarsening algorithm now reads as follows:

Algorithm 1 Non-uniform coarsening of heterogeneous geomodels

- 1: Introduce coloring of cells based on the logarithm of the velocity:

$$C_i = \{\text{cells with color } i\} = \{c : m_v + (i-1)D_v < \log |v(c)| < m_v + iD_v\}.$$

Here $m_v = \min(\log |v|)$ and $D_v = (\max(\log |v|) - m_v)/10$. Create an initial grid where each block B is a connected collection of cells in the fine grid with the same color.

- 2: If $|B| < N_L|\Omega|/N$, then merge B with a neighboring block B' for which

$$\left| \frac{1}{|B|} \int_B g(v) dx - \frac{1}{|B'|} \int_{B'} g(v) dx \right| \leq \left| \frac{1}{|B|} \int_B g(v) dx - \frac{1}{|B''|} \int_{B''} g(v) dx \right|$$

for all other neighboring blocks B'' . Here Ω is the computational domain (the reservoir), N is the total number of cells in the original grid, and N_L is a lower bound on the number of average-sized cells that a block B in the coarse grid may consist of. Hence, B is adjoined with a neighboring block subject to flow of similar magnitude as that which B is subject to.

- 3: Refine a block B if

$$\int_B g(v) dx > \frac{N_U}{N} \int_\Omega g(v) dx. \quad (5)$$

Here the right hand side provides an upper bound on the amount of flow allowed to pass through a single block. The refinement strategy consists of the following parts:

- (a) Pick an arbitrary cell T_0 in the fine grid that is contained in B .
- (b) Find the cell $T_i \subset B$ for which the center of T_i is furthest away from the center of T_0 (T_i will be located along the boundary B).
- (c) Define $B' = T_i$ and progressively enlarge B' by successively adding cells surrounding T_i until the upper bound in (5) is reached.
- (d) Define $B = B \setminus B'$ and refine B further if (5) still holds.

- 4: Repeat Step 2 and terminate.
-

Step 4 does not have significant impact on the accuracy of obtained solutions, and may therefore be skipped, but it tends to reduce the number of blocks by 30%–50%. This is because the refinement strategy in Step 3 produces cells with volume less than $N_L|\Omega|/N$. Although the final cells may satisfy (5) we do *not* repeat Step 3.

The numerical results in Section 5 indicate that the algorithm is quite robust with respect to the choice of N_L and N_U . Tuning these parameters is quite easy. For instance, as a rule of thumb, to generate a grid with a scale-up factor of N , choose $N_U \sim 5N/4$ and $N_L \sim N/4$. Moreover, the algorithm is not sensitive to the cells T_0 used in the refinement

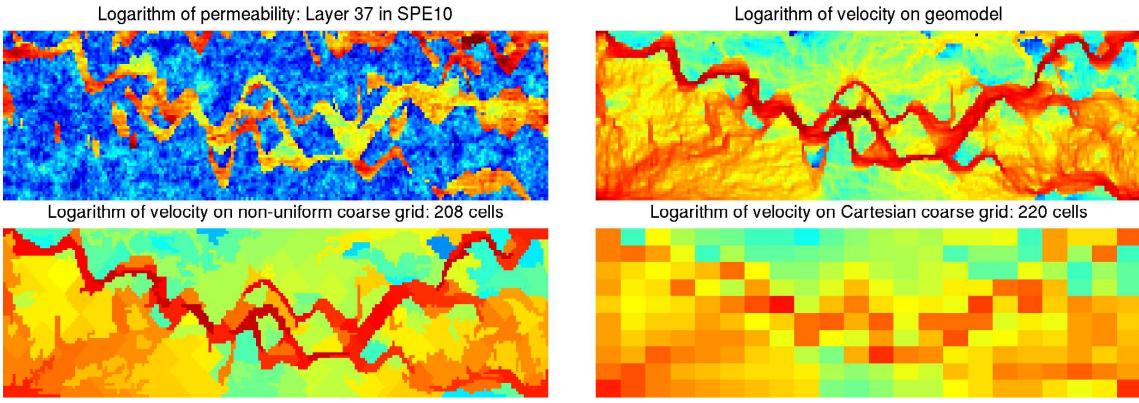


Fig. 1. Top-left: Logarithm of permeability (60×220 Cartesian grid). Top-right: Logarithm of $|v|$ obtained for a five-spot. Bottom-left: Logarithm of $|v|$ mapped onto a non-uniform coarse grid with 208 cells. Bottom-right: Logarithm of $|v|$ mapped onto a Cartesian grid with 220 cells.

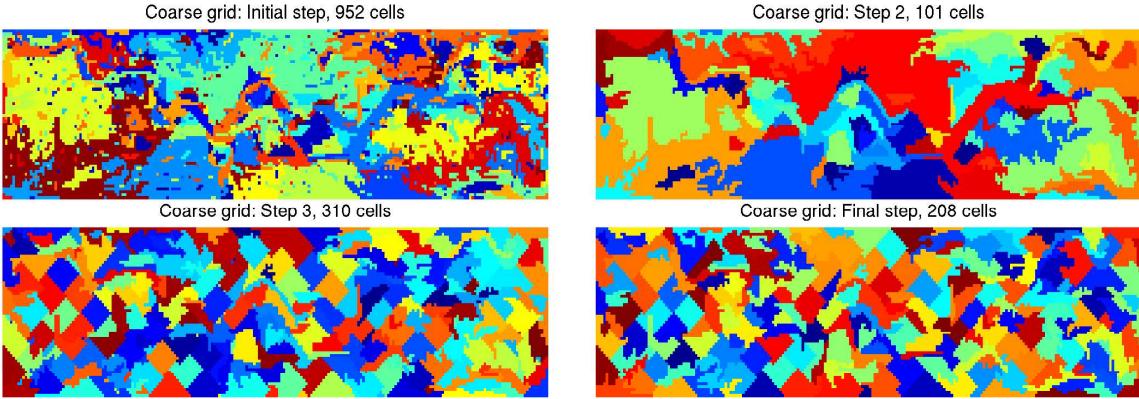


Fig. 2. Grid after each step when the non-uniform coarsening algorithm is applied to layer 37 in the SPE10 model with $N_L = 15$ and $N_U = 80$.

step because it is only used as a starting for locating the cell T_i along ∂B . Hence, in our implementation, we simply take T_0 to be the first cell in the list of cells that belong to B .

To illustrate how the algorithm works we consider a two-dimensional test case representing one of the layers (layer 37) in Model 2 used in the 10th SPE Comparative Solution Project [9], a project used to test and validate upscaling methods. The logarithm of the horizontal permeability is shown in the upper left hand plot of Figure 1. Here we see the trace of high-permeability channels on a low permeable background. Figure 1 also shows plots of the logarithm of the velocity depicted in the upper right plot mapped onto a non-uniform coarse grid with 208 cells and a uniform Cartesian grid with 220 cells.

Figure 2 shows the grids obtained in each step of the non-uniform coarsening algorithm with $N_L = 15$ and $N_U = 80$, giving an upscaling factor of about 65. In each plot each block is assigned a random color. After the initial step we see a myriad of small cells due to oscillations in the velocity, which, in turn, are caused by oscillations in the permeability. These bits and pieces are generally too small to have significant impact on the flow regime. Hence, they are merged with a neighboring block to obtain the next grid depicted in the

upper-right plot. Here we see that some of the blocks stretch across large portions of the domain, hence clearly being too large to model an advancing saturation front. Upon completion of the refinement step we obtain the grid depicted in the lower-left plot. In this grid we may spot some very small blocks consisting of only a few cells. After the blocks consisting of less than 15 cells are merged with a neighboring block, we obtain the final grid depicted in the lower-right plot. Note that the algorithm delivers an unstructured grid even if the fine grid is structured, and that the blocks have very irregular shapes.

The fact that we use flow information to generate the coarse grid may give the impression that the grid is case-specific, and therefore needs to be recomputed when flow conditions change, e.g., when well-rates, boundary conditions, or well-configurations change. But because high flow regions represent high-permeability zones with good large scale connectivity, it is generally *not* necessary to generate a new coarse grid, even if flow conditions change significantly. This will be demonstrated in Section 5.4. Note also that since the coarsened grid is employed only to compute the flow transport, one does not need to solve an additional single-phase flow problem. The grid is generated after the first pressure step in the sequential time-stepping loop, i.e., before the first saturation step.

An alternative coarsening strategy that does not require flow information is to base the initial coloring on the logarithm of permeability. This option has been tested, and found to be less viable. The reason for this is that low permeability cells that occur in high permeability regions may be subject to large amount of flow, and should therefore not be separated from the high permeability region. Similarly, high permeability patches may be isolated inside low permeable regions so that they are not subject to significant flow.

It should also be mentioned that a potential drawback with flow based coarsening strategies is that the grid will in general have to be regenerated for every permeability realization. However, the current algorithm is very fast – the time it takes to generate a new coarse grid is negligible relative to the time it takes to perform a single simulation. We therefore believe that this type of upscaling strategy should be a valuable tool, also for an analyst trying to complete an uncertainty study.

3.1 Discussion

The grid-coarsening approach described above is proposed as a remedy for situations where conventional upscaling is inaccurate or it is prohibitively computationally expensive to perform the entire simulations directly on the geomodel grid. The main objective is to exploit information in high resolution geological models in an optimal way. Here we discuss the applicability and limitations of the proposed approach as well as possible implications and relations to other methods. Since a key to our approach is the ability to compute mass conservative velocity fields with high resolution, e.g., directly on the geomodel, we discuss first the validity of this assumption.

As an intermediate solution between upscaling and direct simulation on geomodels, the

possibility of using multiscale methods [1,5,14] to provide high resolution velocity fields has been proposed. These methods may be used to generate accurate mass conservative velocity fields on high-resolution geomodel grids at the cost of a conventional flow-based upscaling method. They may therefore be regarded as a more robust and flexible alternative to upscaling for the pressure equation (see e.g., [3,16] for a discussion). In particular it has been shown that the multiscale mixed finite element method [7,1] is very flexible with respect to handling very complex grids [3]. This method is currently being extended to the three-phase black-oil model, which is the industry standard for reservoir simulation. Thus, with ongoing efforts, we believe that multiscale methods will, in time, provide a viable tool for computing mass conservative velocity fields directly on real-field geomodels.

Multiscale methods for subsurface flow simulation have for the most part been used to solve the pressure equation and combined with a traditional method for solving the saturation equation(s) on the underlying fine grid. Unfortunately, even with streamline methods, computing the phase transport on the geomodel may create a bottle-neck in the simulations that prevents high speed-up factors relative to sequential time-stepping approaches where the pressure equation is solved with a standard numerical method directly on the geomodel. Thus, whether we solve the pressure equation on the fine grid, or employ a multiscale method to incorporate subgrid effects into a set of coarse scale equations, there is a need to address how the phase transport can be computed in an optimal way with respect to solution accuracy and computational efficiency.

Here we have proposed creating an upscaled model only for the saturation equation, i.e., to generate a coarse grid that resolves more accurately underlying flow patterns than traditional coarse grids used in reservoir simulation, see Figure 1, Figure 4 and the analysis in Section 3.2 below. Although many authors have proposed coarsening strategies for subsurface flow applications, see e.g., [13,15,19] and the references therein, the current approach is primarily motivated by [10]. In particular, it was observed in [10] that introducing a grid which is more finely gridded in high-flow regions allows capturing more accurately flow quantities of interest, such as production characteristics, without resorting to multi-phase upscaling. The main significance of our approach is that it is all-applicable. Hence, whereas the non-uniform coarsening strategy in [10] applies primarily to Cartesian-like geomodels with high flow channels that are aligned with the grid, our approach applies to all types of grids and essentially all types of heterogeneous structures.

In the current paper we have chosen to consider a simplified model that does not include effects from gravity and capillary forces. In general, flow-based grid coarsening approaches based on separating high and low flow regions, or on tuning the grid to resolve high flow regions, are designed for modeling flows where heterogeneity, rather than gravity or capillary forces, dominates the flow patterns. This does not mean that the current approach is not applicable to problems with non-zero gravity or capillary forces. Indeed, most flow scenarios on the scale of an oil reservoir are affected by both gravity and capillary forces, but the flow is usually primarily driven by pressure (viscous forces), meaning that flow patterns are generally dominated by heterogeneity. The current algorithm is therefore expected to work well for this type of flow scenario.

If flow patterns are dominated by gravity, then we do not expect that solutions obtained using flow-based grids will be much more accurate than solutions obtained using standard coarse grids. Flow-based gridding implicitly assumes that the high flow regions remain the same throughout a simulation. For gravity dominated flows the flow patterns may change significantly during simulation so that regions initially subject to significant flow may be subject to little flow at later times, and vice-versa. Thus, tuning the grid to high flow regions at initial time may be of little value. On the other hand, the coarsening strategy proposed in this paper should not be regarded only as a tool to achieve higher accuracy, it may also be used as a generic tool to coarsen complex unstructured geomodels. Moreover, although the algorithm has not been applied to gravity dominated flows, we believe that it will not, on average, produce less accurate results than uniform coarsening strategies.

Upscaling of capillary dominated flows is a problem of a different nature. Indeed, capillary dominated flows are more strongly coupled, and the sequential splitting where one solves the pressure equation and saturation equation sequentially is no longer justified. However, the proposed coarsening algorithm is intended for field scale (10 m–10 km) simulations, whereas flows that are dominated by capillary forces occur on a much finer scale (dm–m).

Finally we would like to remark that combining the proposed coarsening approach with a multiscale method for the pressure equation brings us closer to having an Earth Model shared between reservoir engineers and reservoir geologists [15]. This means that the geological model provided by geologists is used as input for the reservoir simulator. At present geoscientists and reservoir engineers are forced to work with different descriptions of the subsurface. To bridge the gap it is necessary that the reservoir engineer can select semi-automatically numerical technology and grid resolution for the simulations at run-time to fit available computer resources and project requirements. Having a “Shared Earth Model” will, apart from giving the reservoir engineer a better description of the subsurface, offer the geoscientist the ability to validate the geomodel before it is sent to the reservoir engineer. This can simplify and accelerate reservoir simulation workflows considerably, and thereby allow e.g., oil-companies to save many man-hours. Indeed, subsurface geological models are not static, they are analyzed, iterated and rebuilt as project requirements and business decisions change, or when uncertainty studies indicate that the model does not give a representative picture of the reservoir flow regime.

3.2 Analysis

Recall that Algorithm 1 groups high and low flow regions separately. The velocity field inside a single grid block will therefore be of nearly the same magnitude, i.e., the velocity will have small variations in each coarse block. We will now show that this implies also that the saturation variation is small within each coarse block in the non-uniform coarse grid, i.e., that the saturation is well resolved on the coarse grid.

Consider the equation for the time-of-flight function

$$v \cdot \nabla \tau = \phi.$$

The time-of-flight functions describe travel times along flow trajectories (streamlines). The streamlines $\psi = x(\tau)$ are trajectories along which v is a tangential vector (see e.g., [2] for details). Each streamline emanates from an injection well and terminate at a production well. Along streamlines, the corresponding time-of-flight functions are defined by

$$\frac{d\tau}{ds} = \frac{\phi}{|v|},$$

where s is arclength distance. Moreover, between wells we have

$$\frac{\partial S}{\partial t} + \frac{\partial f(S)}{\partial \tau} = 0. \quad (6)$$

Observe now that since we employ a velocity computed on the fine grid for the coarse grid simulations, we do not really alter the streamlines when coarsening the grid. Rather we introduce a coarse grid approximation to the time-of-flight functions. Thus, conceptually we may think of our approach as solving (4) by solving (6) along one-dimensional streamlines where the time-of-flight function τ is replaced with a coarse grid approximation τ_0 . Hence, one should obtain an accurate coarse grid saturation solution S_0 away from shocks if the projection τ_0 of τ onto the coarse grid is close to τ . Moreover, if the time-of-flight functions are close to each other, then the shock locations also remain close to each other. Thus, although the L_∞ norm of $S - S_0$ can be of order $O(1)$ in vicinity of shock regions, the L^p norm of $S - S_0$ is small because of the small volume of this region.

To prove that τ_0 approximates τ , we define $\delta(x) = \tau(x) - \tau_0(x)$, i.e., $\delta(x)$ is the time lag of reaching the point x with coarse-scale velocity. We will show that the path traversed during this time lag is small due to small variations of the velocity field within the coarse block. Thus, if the velocity does not degenerate, the time lag $\delta(x)$ will remain small.

Denote v_0 to be the coarse-scale velocity that determines τ_0 . Or more precise, v_0 is the projection of v onto the coarse grid and τ_0 is the corresponding time-of-flight function. Next, introduce two streamlines ψ and ψ_0 that emanate from the same point, but with velocity v and v_0 , respectively. The corresponding time-of-flight functions satisfy

$$x(\tau) = x, \quad x_0(\tau_0) = x, \quad (7)$$

where $x(t)$ and $x_0(t)$ are functions that trace the streamline trajectories, defined by $dx/dt = v$ and $dx_0/dt = v_0$, respectively. It now follows from (7) that

$$\int_0^\tau v(x(t))dt = \int_0^{\tau_0} v_0(x_0(t))dt,$$

and hence that

$$\int_\tau^{\tau_0} v_0(x_0(t))dt = \int_0^\tau (v(x(t)) - v_0(x_0(t)))dt. \quad (8)$$

To see that the right hand side of (8) is small, note that

$$|\int_0^\tau (v(x(t)) - v_0(x_0(t)))dt| \leq \int_0^\tau |v(x(t)) - v(x_0(t))|dt + \int_0^\tau |v(x_0(t)) - v_0(x_0(t))|dt. \quad (9)$$

If $|v - v_0|$ is uniformly small, then the last integral on the right hand side is small. Moreover, one can also easily show that if $|v - v_0|$ is uniformly small, then flow trajectories that emanate from the same point remain close to each other. Consequently, we have that when $|v - v_0|$ is uniformly small, then also the first term on the right hand side (9) is small, and hence the whole right hand side of (9) is small. On the other hand, the left hand side of (8) is equal to $x_0(\tau_0) - x_0(\tau)$, or the distance traversed from time τ_0 to time τ with the velocity v_0 . Thus, (8) implies that this distance is small. Consequently, if v_0 is non-degenerate, then the time lag $\delta(x) = \tau(x) - \tau_0(x)$ will also remain small.

4 Numerical discretization

4.1 The pressure equation

To discretize (3) we first need to prescribe boundary conditions. In this paper we impose no-flow boundary conditions ($v \cdot n = 0$ on $\partial\Omega$) for simplicity. Now denote the fine grid (the geomodel) by $\mathcal{T} = \{T\}$, and define the following function spaces:

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

Next we introduce the following bilinear forms:

$$\begin{aligned} b(\cdot, \cdot) : H_0^{\text{div}}(\mathcal{T}) \times H_0^{\text{div}}(\mathcal{T}) &\rightarrow \mathbb{R} & b(u, v) &= \sum_{T \in \mathcal{T}} \int_T u \cdot K^{-1} v \, dx \\ c(\cdot, \cdot) : H_0^{\text{div}}(\mathcal{T}) \times L^2(\Omega) &\rightarrow \mathbb{R} & c(v, p) &= \sum_{T \in \mathcal{T}} \int_T p \nabla \cdot v \, dx \\ d(\cdot, \cdot) : H_0^{\text{div}}(\mathcal{T}) \times L^2(\partial\mathcal{T}) &\rightarrow \mathbb{R} & d(v, \pi) &= \sum_{T \in \mathcal{T}} \int_{\partial T} \pi v \cdot n_T \, ds. \end{aligned}$$

Here n_T is the unit normal on ∂T pointing outward.

In a so-called hybrid formulation [6] of (3) with prescribed no-flow boundary conditions

one finds a unique triplet of functions $(v, p, \pi) \in H_0^{\text{div}}(\mathcal{T}) \times L^2(\Omega) \times L^2(\partial\mathcal{T} \setminus \partial\Omega)$ such that

$$\begin{aligned} b(u, v) - c(u, p) + d(u, \pi) &= 0, & \forall u \in H_0^{\text{div}}(\mathcal{T}), \\ c(v, q) &= (f, q), & \forall q \in L^2(\Omega), \\ d(v, \mu) &= 0, & \forall \mu \in L^2(\partial\mathcal{T} \setminus \partial\Omega). \end{aligned} \quad (10)$$

Here p represents pressure, v represents velocity, and π is a Lagrange multiplier used to enforce mass conservation. Moreover, $\partial\mathcal{T} = \cup_{T \in \mathcal{T}} \partial T$ and (\cdot, \cdot) is the L^2 inner product.

4.1.1 Mimetic finite-difference method (FDM)

Mimetic FDMs [12,11] may be regarded as finite-difference or finite-volume versions of mixed finite element methods (FEMs). Since mimetic FDMs are quite new, we describe first the hybrid formulation of corresponding mixed FEMs. In the hybrid formulation one selects finite-dimensional subspaces $V \subset H_0^{\text{div}}(\mathcal{T})$, $U \subset L^2(\Omega)$, and $\Pi \subset L^2(\partial\mathcal{T} \setminus \partial\Omega)$, and seeks $(v, p, \pi) \in V \times U \times \Pi$ such that (10) holds for all $(u, q, \mu) \in V \times U \times \Pi$. Here each approximation space is spanned by a particular set of basis functions. For instance, V is spanned by a set of basis functions $\{\psi_i^m \in H^{\text{div}}(T_m) : T_m \in \mathcal{T}, i = 1, 2, \dots, N_m\}$, where ψ_i^m is supported in T_m only. Thus, since $b(\psi_i^m, \psi_j^n)$ is nonzero only if $n = m$, we have

$$b\left(\sum_{i,m} u_i^m \psi_i^m, \sum_{i,m} v_i^m \psi_i^m\right) = \sum_{m,i,j} u_i^m v_j^m b(\psi_i^m, \psi_j^m) = \sum_m \mathbf{u}_m^T \mathbf{B}_m \mathbf{v}_m, \quad \mathbf{u}_m, \mathbf{v}_m \in \mathcal{R}^{N_m}, \quad (11)$$

where $\mathbf{u}_m = [u_i^m]$, $\mathbf{v}_m = [v_i^m]$, and \mathbf{B}_m is a local matrix associated with T_m . It is important to observe that by enumerating the velocity unknowns (basis-functions) in a cell-wise manner, the mass-matrix that stems from $b(u, v)$ in (10) has a block-diagonal structure, where the local matrices \mathbf{B}_m are the block-diagonals. This allows the hybrid system to be reduced to a symmetric and positive definite linear system, which is in general easier to solve than the original system which is indefinite.

The mimetic FDM formulation is equivalent, except that the subspace V in $H_0^{\text{div}}(\mathcal{T})$ is replaced by a discrete subspace $M \subset L^2(\partial\mathcal{T})$, and the associated bilinear form $b(\cdot, \cdot)$ is replaced by a bilinear form $m(\cdot, \cdot)$ that acts on $L^2(\partial\mathcal{T}) \times L^2(\partial\mathcal{T})$. The basic idea is to introduce means for evaluating $b(\cdot, \cdot)$ in an approximate sense without having explicit representations of the velocity in each cell. In particular, instead of seeking a velocity field defined over each element T , one seeks a set of fluxes defined over the cell faces ∂T .

The bilinear form $m(\cdot, \cdot)$ employed here is defined so that the resulting method reduces to a first order Raviart-Thomas mixed FEM (RT0) [18] when the grid is Cartesian and the permeability is scalar and cell-wise constant. In particular, we associate here (as in RT0) a basis function ψ_i^m with each face F_i^m of every grid cell T_m . The velocity unknown v_i^m corresponding to F_i^m will represent net velocity across F_i^m in the direction of the unit

normal n_i^m to F_i^m pointing out of T_m . If we now expand u and v in the basis $\{\psi_i^m\}$:

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

and assume that $m(\psi_i^m, \psi_j^n)$ is nonzero only if $n = m$, then

$$m(u, v) = \sum_{m,i,j} u_i^m v_j^m m(\psi_i^m, \psi_j^m) = \sum_m \mathbf{u}_m^T \mathbf{M}_m \mathbf{v}_m, \quad \mathbf{u}_m, \mathbf{v}_m \in \mathcal{R}^{N_m}, \quad (12)$$

for a given matrix \mathbf{M}_m associated with T_m . Hence, contrary to (11) where \mathbf{B}_m was defined by the bilinear form $b(\cdot, \cdot)$, the bilinear form $m(\cdot, \cdot)$ is defined by the local matrices \mathbf{M}_m .

In the implementation we employ only the inverse of \mathbf{M}_m . We therefore provide only a formula for computing the inverse \mathbf{W}_m of \mathbf{M}_m [12]. To this end, we define the following auxiliary matrices:

\mathbf{N}_m — matrix whose i 'th row is defined by

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

\mathbf{C}_m — matrix whose i 'th row is defined by (x_m is the center of T_m)

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

\mathbf{D}_m — diagonal matrix containing the areas of each face,

\mathbf{Z}_m — matrix whose columns form an orthonormal basis for the column space of $\mathbf{D}_m \mathbf{C}_m$.

Then the inverse matrix \mathbf{W}_m is defined by

$$\mathbf{W}_m = \frac{1}{|T_m|} \mathbf{N}_m K \mathbf{N}_m^T + \frac{2\text{trace}(K)}{|T_m|} (\mathbf{I} - \mathbf{Z}_m \mathbf{Z}_m^T). \quad (13)$$

This matrix is symmetric and positive definite, and hence ensures that $m(\cdot, \cdot)$ defines an inner-product on $M \times M$.

We would like to note that the current approach to modeling subsurface flow does not require using a mimetic FDM to solve the pressure equation. Any method that provides a mass-conservative velocity field on the geomodel, or at least on some given fine grid, may be used. The primary reason why we have chosen to employ a mimetic FDM is the natural ability to handle complex grid geometries, as well as full-tensor permeabilities [4]. Indeed, geomodels that model real oil or groundwater reservoirs are often significantly more complex than the corresponding upscaled simulation models. Hence, when discretizing the pressure equation directly on a geomodel it is particularly important that a flexible numerical method is employed. For a discussion of the pros and cons of various discretization techniques for (3) on geological models, we refer the reader to [3], in which a multiscale method for corner-point grids, the industry-standard in reservoir simulation, is proposed.

4.2 The saturation equation

To discretize (4) on grids where each block consists of a connected collection of cells in the original fine grid we employ an upstream weighted finite volume method with respect to fluxes obtained on the fine grid. That is, instead of using only the net flux for each interface in the coarse grid and a standard upstream weighted finite volume method with respect to the coarse grid fluxes, we utilize the subgrid resolution in the velocity.

Hence, let each grid block B_m consist of a connected collection of cells in the fine grid and denote non-degenerate interfaces in the fine grid by $\gamma_{ij} = \partial T_i \cap \partial T_j$. The discrete system of equations for the saturation equation now reads:

$$S_m^{n+1} = S_m^{n+1} + \frac{\Delta t}{\int_{B_m} \phi dx} \left[\int_{B_m} q_w(S^{n+1}) dx - \sum_{\gamma_{ij} \subset \partial B_m} V_{ij}(S^{n+1}) \right]. \quad (14)$$

Here S_m is the net saturation in B_m and

$$V_{ij}(S) = \max\{v_{ij} f_w(S|_{T_i}), -v_{ij} f_w(S|_{T_j})\}, \quad (15)$$

where v_{ij} is the Darcy flux across γ_{ij} , i.e., from T_i to T_j . Notice that if there is bi-directional flow across $\Gamma_{kl} = \partial B_k \cap \partial B_l$, i.e., if $\int_{\Gamma_{kl}} \max\{v \cdot n_{kl}, 0\} ds > 0$ and $\int_{\Gamma_{kl}} \min\{v \cdot n_{kl}, 0\} ds < 0$ where n_{kl} is the unit normal to Γ_{kl} pointing from B_k to B_l , then the phase-flux across Γ_{kl} is approximated using both S_k and S_l . Thus, although (14) is based on a one-sided upstream scheme, we may obtain a two-sided upstream scheme on the coarse grid.

5 Numerical results

In this section we seek to demonstrate that by using Algorithm 1 for grid coarsening one consistently obtains more accurate saturation solutions than if uniform coarsening is employed. To this end, solutions obtained using non-uniform coarse grids are compared with corresponding solutions obtained using uniform coarse grids, as well as with a reference solution computed using the original grid to perform the transport simulations.

We first apply the non-uniform coarse gridding strategy to a sequence of Cartesian grid models with permeability data from Model 2 of the Tenth SPE Comparative Solution Project [9], henceforth called the SPE10 model. Next we consider a corner-point model with 30 layers. On this model we generate 20 different permeability fields by populating each layer with values drawn from a spatially correlated log-normal distribution. In terms of grid, this model is more complex to coarsen with traditional coarsening strategies because many layers are partially eroded away, giving rise to degenerate cells, and pinch-outs causing so-called non-neighboring connections. However, the grid is given on a logically Cartesian format, and may therefore be partitioned uniformly in index space. We apply

this strategy as an alternative to the method proposed in this paper. Finally, we employ a model consisting of four layers from the SPE10 model to assess robustness with respect to degree of coarsening, different well-configurations, and varying flow conditions.

5.1 Experimental setup

The numerical experiments below model incompressible and immiscible two-phase flow without gravity and capillary pressure effects. For the time-stepping, we apply a non-iterated sequential splitting. This means that the pressure equation is solved at the current time-step with total mobility computed using saturations from the previous time-step. Next, the saturations are convected forward in time using the current velocities, and the new saturation values are used to compute the pressure at the next time-step, and so on. The total simulation time for all simulations below is one PVI (pore volume injected). This means that at the end of a simulation the total volume of water that has been injected is equal to the total accessible pore volume in the model.

The dynamic nature of an incompressible two-phase flow system is often quantified by the ratio of the end-point values of the total mobility. Here we define the phase mobilities by

$$\lambda_w = \frac{S^2}{\mu_w}, \quad \lambda_o = \frac{(1-S)^2}{\mu_o}, \quad 0 \leq S \leq 1. \quad (16)$$

Without gravity, the simulation results depend on the viscosities only as a function of the ratio μ_w/μ_o , henceforth referred to as the viscosity ratio. Thus, for simplicity we assume that $\mu_w = 1$. If we now choose $\mu_o > 1$ then we will get so-called unstable displacement flows for which small scale “fingers” develop and move rapidly at the saturation front. In contrast, choosing $\mu_o < 1$ gives stable displacement flows for which one typically gets sharp saturation fronts because the total mobility is larger behind the front than ahead of the front. In this paper we use $\mu_o = 0.2$, $\mu_o = 1$ and $\mu_o = 10$. These values give rise to both stable and unstable displacement flows so that the simulation results should give an indication of the versatility of doing simulations using the proposed coarsening strategy.

For two-phase flow with dynamic total mobility, different saturation solutions will give different velocity fields. However, since we here want to assess how much we can improve saturation solutions by using a non-uniformly coarsened grid, as opposed to uniform coarsening, we want to eliminate differences caused by having different velocity fields. We therefore use the same (fine grid) velocity field to perform simulations on both grids. That is, for each time step, the velocity solution used to perform the coarse grid simulations is obtained by solving the pressure equation on the fine grid using a reference saturation solution to update the total mobility. The reference solution S_{ref} is obtained by solving the saturation equation on the fine grid.

We assess the accuracy of a saturation solution S using a measure for the overall accuracy, and also a measure for assessing the accuracy of the predicted oil-production at the

producers. To measure the overall accuracy, we compute how much the solution deviates from the reference solution in the L^1 -norm, divide by the L^1 -norm of the reference solution, and integrate in time, i.e.,

$$e(S) = \int_0^1 e(S, t) dt \quad \text{where} \quad e(S, t) = \frac{\|S(\cdot, t) - S_{\text{ref}}(\cdot, t)\|_{L^1(\Omega)}}{\|S_{\text{ref}}(\cdot, t)\|_{L^1(\Omega)}},$$

with time measured in PVI.

Similarly, to quantify the accuracy of the predicted oil-production, we measure the accuracy of the predicted water-cut curve w , showing the fraction of water in the produced fluid. To this end, we compute how much the water-cut curve deviates from the reference water-cut w_{ref} curve in the L^2 -norm and divide by the L^2 -norm of w_{ref} , i.e.,

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

again with time measured in PVI.

Finally, because well-models are used only to model well-rates and do not appear explicitly in the saturation equation, we will throughout this section assume that the well rates are fixed. Hence, we assume that the source term q in (3) is given explicitly. Moreover, the source term in (4) will be defined by $q_w(S) = \max(q, 0) + f_w(S) \min(q, 0)$. In the figures and tables presented in this section we will frequently use UC as an abbreviation for results obtained with uniformly coarsened grids and NUC as an abbreviation for results obtained with non-uniformly coarsened grids.

5.2 *Cartesian grids*

In this section we consider a series of two- and three-dimensional Cartesian grid models. Each model represents either a single layer in the SPE10 model [9], or a stack of five consecutive layers. The SPE10 model itself consists of a total of 85 layers, where the top 35 layers model a Tarbert formation representing a prograding near-shore environment, and the bottom 50 layers model a fluvial Upper Ness formation with a spaghetti of narrow high-flow channels. The entire model consists of $60 \times 220 \times 85$ cells, each of size $20 \text{ ft} \times 10 \text{ ft} \times 2 \text{ ft}$.

In order to be able to interpret the results in this section, we need to explain some basic characteristics of the heterogeneous structures in the SPE10 model. Both the Tarbert formation and the Upper Ness formation are highly heterogeneous (in each formation the permeability is anisotropic and spans more than 10 orders of magnitude), but the heterogeneous permeability structures are qualitatively different.

- In the Tarbert formation the permeability in each layer is generated stochastically using a spatially correlated log-normal distribution. However, because the mean permeabil-

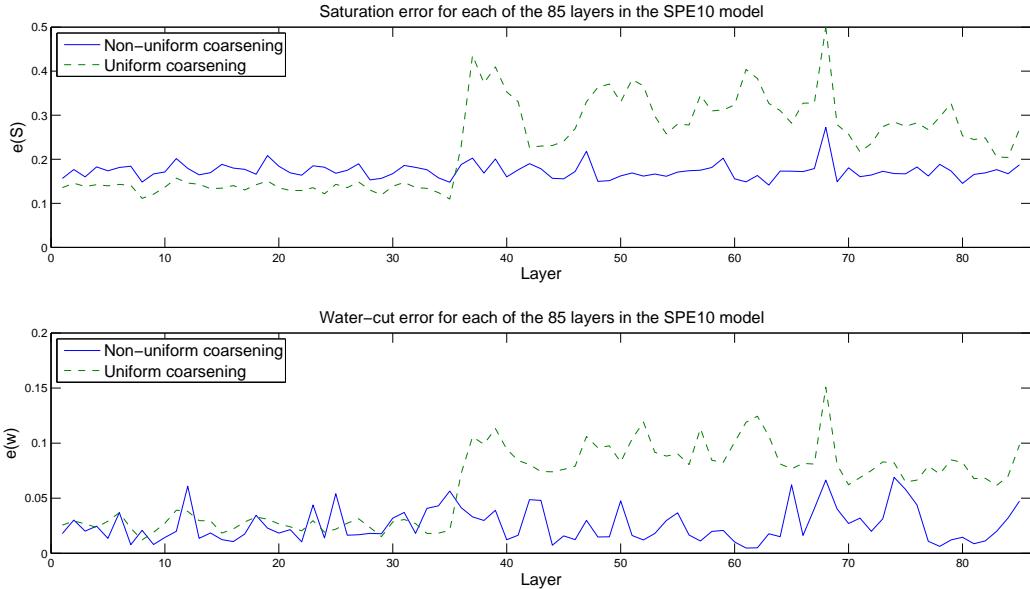


Fig. 3. Saturation and water-cut errors for flow simulations on each of the 85 layers in the SPE10 model. The uniformly coarsened grid is a uniform 15×44 Cartesian grid. The non-uniform grids contain 619–734 cells ($N_L = 6$, $N_U = 20$).

ity in the layers varies significantly, one obtains segregated flow scenarios with sharp saturation contrasts from one layer to the next.

- The Upper Ness formation consists of multiple intertwined high permeable flow-channels through a low permeable background. Hence, in this model the flow channels will carry the majority of the flow, and therefore cause sharp saturation contrasts between the background structure and the channels.

Fluvial formations, such as the Upper Ness model, are generally very hard to upscale. Indeed, whereas one may obtain good results for the Tarbert formation by using grid blocks that do not cut across the layers (i.e., are one layer thick), upscaling the Upper Ness formation adequately requires that the channels are resolved by the coarse grid. This is very difficult, if not impossible, to accomplish with conventional grid-constrained upscaling strategies.

For the flow simulations that we perform here we employ $\mu_w = 1$ and $\mu_o = 10$, i.e., we consider an unstable displacement process with the same viscosity ratio as in [9]. We also use the same well-configuration as in [9], i.e., a five-spot with a vertical injection well in the middle and vertical production wells at each of the four corners. The total production rate is the same for all producers, and q is assumed to be constant along each well trajectory. Finally the simulations are performed with 20 pressure steps and 10 saturation steps between each pressure step.

Table 1

Mean of the errors plotted in Figure 3.

Geomodel	Tarbert formation		Upper Ness formation	
Coarsening strategy	NUC	UC	NUC	UC
Mean saturation error	0.1737	0.1357	0.1730	0.3031
Mean water-cut error	0.0246	0.0258	0.0263	0.0875

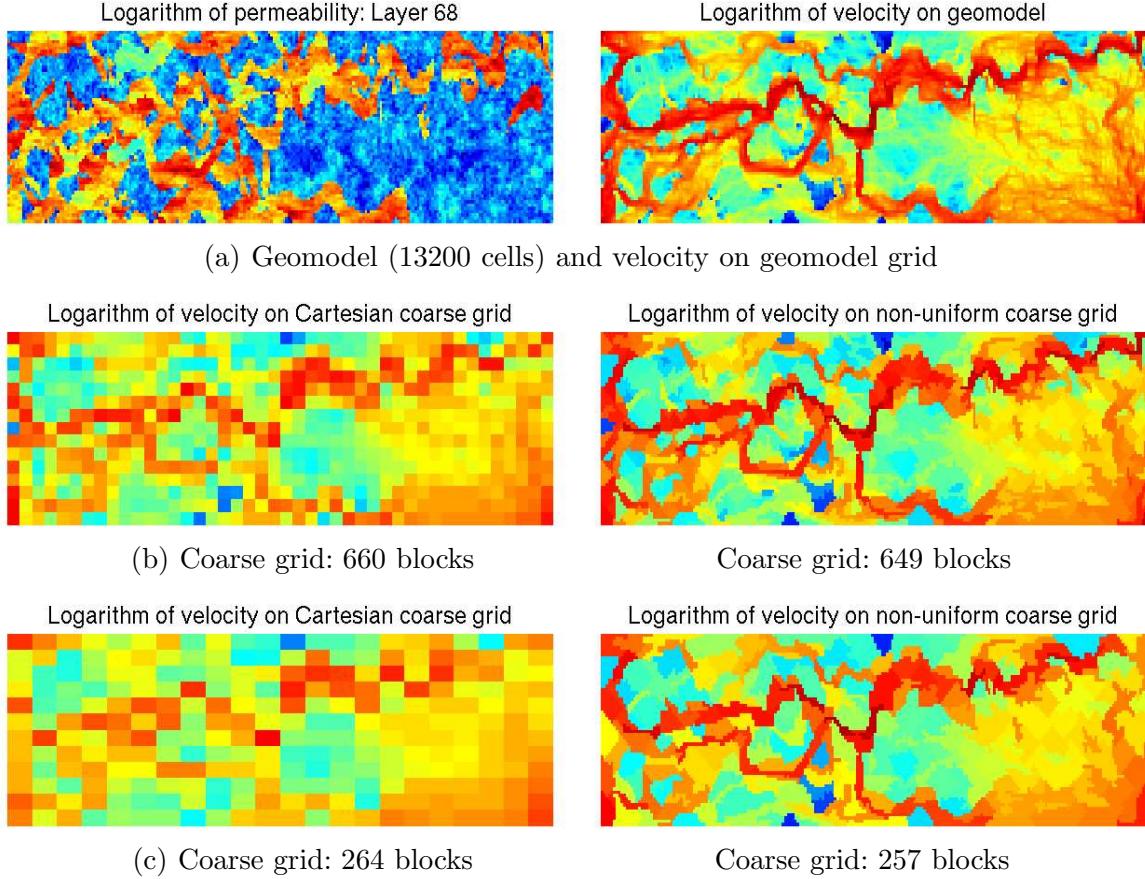
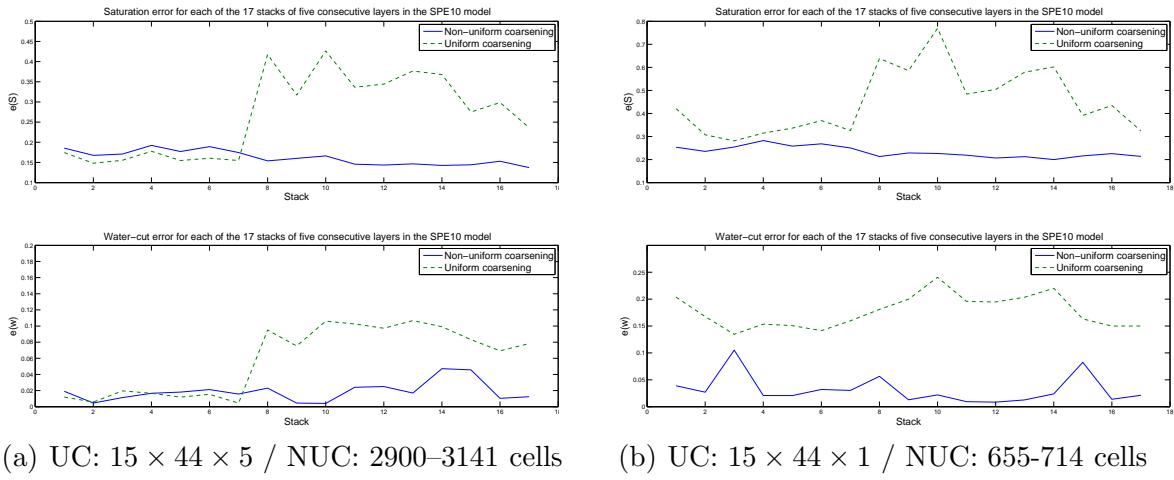


Fig. 4. Logarithm of velocity for a five spot approximated on a sequence of different grids. Notice that the channelized flow pattern is lost in the Cartesian coarse grids, whereas the channels are well resolved on the non-uniform coarse grids, even with only 257 blocks.

5.2.1 Two-dimensional simulations

Figure 3 shows the saturation and water-cut errors for flow simulations on each of the 85 layers in the SPE10 model. Table 1 displays the average saturation and water-cut errors. We see that the quality of the solutions obtained with the non-uniformly coarsened grids is nearly independent of the heterogeneous structures in the model. In contrast, when using uniform coarse grids both the saturation error and water-cut error are much larger for the fluvial Upper Ness formation than for the Tarbert formation.

The varying performance of uniform coarse grids can be explained by the nature of the



(a) UC: $15 \times 44 \times 5$ / NUC: 2900–3141 cells

(b) UC: $15 \times 44 \times 1$ / NUC: 655–714 cells

Fig. 5. Saturation and water-cut errors for flow simulations on 17 models consisting of a stack of five consecutive layers from the SPE10 model.

Table 2
Mean of the errors plotted in Figure 5.

Uniform grid	$15 \times 44 \times 5$ Cartesian grid $N_L = 6$ and $N_U = 20$	$15 \times 44 \times 1$ Cartesian grid $N_L = 25$ and $N_U = 100$
--------------	--	--

Tarbert formation				
Mean saturation error	NUC: 0.1796	UC: 0.1608	NUC: 0.2575	UC: 0.3367
Mean water-cut error	NUC: 0.0152	UC: 0.0123	NUC: 0.0393	UC: 0.1588
Upper Ness formation				
Mean saturation error	NUC: 0.1493	UC: 0.3397	NUC: 0.2162	UC: 0.5315
Mean water-cut error	NUC: 0.0213	UC: 0.0913	NUC: 0.0263	UC: 0.1899

flow. Because the heterogeneity in the layers from the Tarbert formation are relatively smooth, one obtains saturation fields that are well resolved on the uniform coarse grids. In contrast, the heterogeneity in the layers from the Upper Ness formation creates flow scenarios where a majority of the flow occur in narrow flow channels. This gives sharp saturation differences between the channels and the low permeable background. Because the uniform coarse grids do not resolve the channels properly (see e.g., Figure 4), one will generally smooth the saturation profile, and therefore obtain inaccurate results. For the non-uniform coarse grids, on the other hand, significant smoothing of the saturation profile is avoided because the cells are grouped according to magnitude of flow. However, we note that for the layers from the Tarbert formation the errors for the non-uniform grids are slightly larger. This is not really surprising since both methods perform well and the shape of the grid blocks in the non-uniform grids are more irregular in size and shape.

5.2.2 Three-dimensional simulations

Figure 5 shows the saturation and water-cut errors for flow simulations on 17 models consisting of a stack of five consecutive layers from the SPE10 model. Two sets of simulations are done. In the first the non-uniform grid is generated by the parameters $N_L = 6$ and $N_U = 20$ and the uniform grid is a Cartesian grid of $15 \times 44 \times 5$ cells, thus resolving the layers also in the vertical direction. The second set consists of a non-uniform grid generated by the parameters $N_L = 25$ and $N_U = 100$ and the uniform grid is a Cartesian grid of $15 \times 44 \times 1$ cells, i.e., the layers are not resolved in the vertical direction.

We notice first that with the non-uniform coarse grids we consistently obtain accurate results. In particular we see that, even though the number of cells coarse grids used to obtain the results in Figure 5 (b) is decreased by a factor four–five relative to the results in Figure 5 (a), we see only a slight decrease in accuracy.

With uniform coarsening we obtain accurate results only when using the $15 \times 44 \times 5$ Cartesian grid on stacks from the Tarbert formation. The reason why accurate results are obtained for these cases is that each grid block in the uniform Cartesian grid is only one layer thick so that we do not have sharp permeability contrasts within each grid block. Consequently we expect large saturation differences inside individual grid blocks only near the saturation front. For the stacks from the Upper Ness formation, on the other hand, one expects large saturation differences within individual grid blocks in large regions. When using the $15 \times 44 \times 1$ Cartesian grid one obtains poor results for all stacks. This is due to the fact that in the Tarbert formation there are large permeability differences between the respective layers, giving rise to segregated flow scenarios. By approximating the saturation in all five layers with a constant value, as is done in the uniform coarse grid simulations, one obtains large errors. In particular we see that this has great impact on the water-cut errors. Indeed, the errors obtained with the $15 \times 44 \times 1$ Cartesian grid are significantly larger than the errors obtained with the $15 \times 44 \times 5$ Cartesian grid.

5.3 Corner-point grid

The corner-point grid (or pillar grid) format [17] is a very flexible grid format that is used in many commercial geomodeling softwares. Essentially a corner-point grid consists of a set of hexahedral cells that are aligned in a logical Cartesian fashion where one horizontal layer in the logical grid is assigned to each sedimentary bed to be modeled. 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 6 shows the corner-point grid that we employ in this section populated with two different permeability fields. This grid is given on a 30-by-30-by-30 logical Cartesian format. However, several layers are partially eroded away so that many cells disappear and

introduce connections between cells that are not neighbors in the logical grid. Whereas the logical grid has 27000 cells, the physical grid contains only 15206 cells.

We consider here a quarter-of-a-five-spot case, i.e., we inject at constant rate along a vertical column located at one of the corners and produce at constant rate along a vertical well at the opposite corner. The corner-point grid has been populated with 20 different permeability fields. Each field is a layered scalar permeability field where the permeability in each layer is spatially correlated and drawn from a log-normal distribution. For each realization the permeability distribution of layer z in the logical grid is generated using the following Matlab code:

```
k = exp(10*rand(1)+smooth3(randn(30,30,3), 'gaussian', [3,3,3], 0.55));
K(:,:,z) = k(:,:,2);
```

In the simulations we have taken 40 pressure steps and 10 saturation steps between each pressure step. The non-uniform coarse grids are generated using $N_L = 6$ and $N_U = 20$. These parameters resulted in 20 coarse grids with 647–704 cells. For the uniform coarsening approach we subdivide the 30-by-30-by-30 logical Cartesian grid uniformly into a 10-by-10-by-10 grid. However, because some cells disappear, the resulting coarse grid consists of a total of 838 cells. Hence, the non-uniformly coarsened grids contain 20–30 percent less cells than the uniformly coarsened grid that we use for comparison.

Figure 7 shows saturation errors $e(S)$ and the water-cut errors $e(w)$ for all permeability fields and three viscosity ratios. Table 3 shows the mean error over the 20 realizations. The results clearly demonstrate that the solutions obtained with non-uniform coarse grids are, on average, significantly more accurate than the solutions obtained for the uniformly coarsened grid. For instance, the saturation errors obtained on the non-uniform coarse grids are in all cases, except for realization 19 with $\mu_o = 10\mu_w$, about 0.25, whereas the saturation errors obtained using the uniformly coarsened grid is often much larger.

The water-cut errors increase for both methods when μ_o decreases relative to μ_w . This is primarily due to the fact that high viscosity ratios have a sharpening effect on the saturation front. In particular, for $\mu_o = 0.2\mu_w$ we obtain shock-type saturation fronts. Simply due to less spatial resolution, shock-type saturation fronts will be smoothed on coarsened grids. Hence, because the reference water-cut curve is a function of fine-grid saturations, and coarse grid water-cut curves are functions of coarse grid saturations, we obtain larger water-cut errors for high and low viscosity ratio flows. We notice also that for a couple of realizations for the case $\mu_o = 0.2\mu_w$ the water-cut error obtained with the non-uniform grid is larger than for the uniform grid. A reason for this may be that the algorithm sometimes groups many of the cells along the production well in one block, and therefore do not account for different breakthrough time in the different layers. This could of course be avoided by adding an extra feature in the coarsening algorithm that ensures that blocks do not contain too many cells with a source.

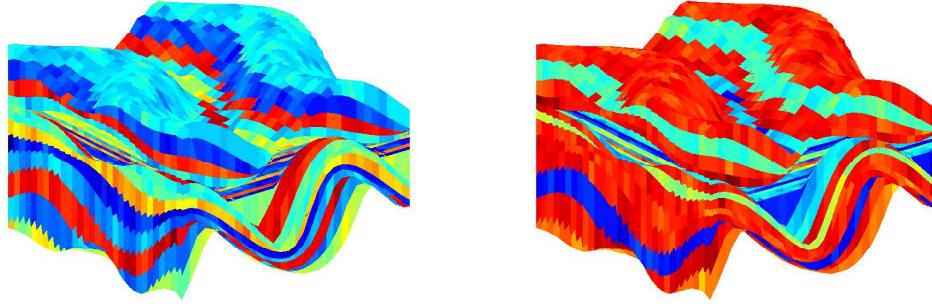


Fig. 6. Permeability field realizations 1 (left) and 2 (right). Realization 1 is particularly difficult to upscale because there are very few high permeability layers. Consequently a lot of flow is forced into only a few of the layers. Realization 2 gives flow scenarios that are much easier to upscale. This model has thick high-permeable zones that transmit majority of the flow.

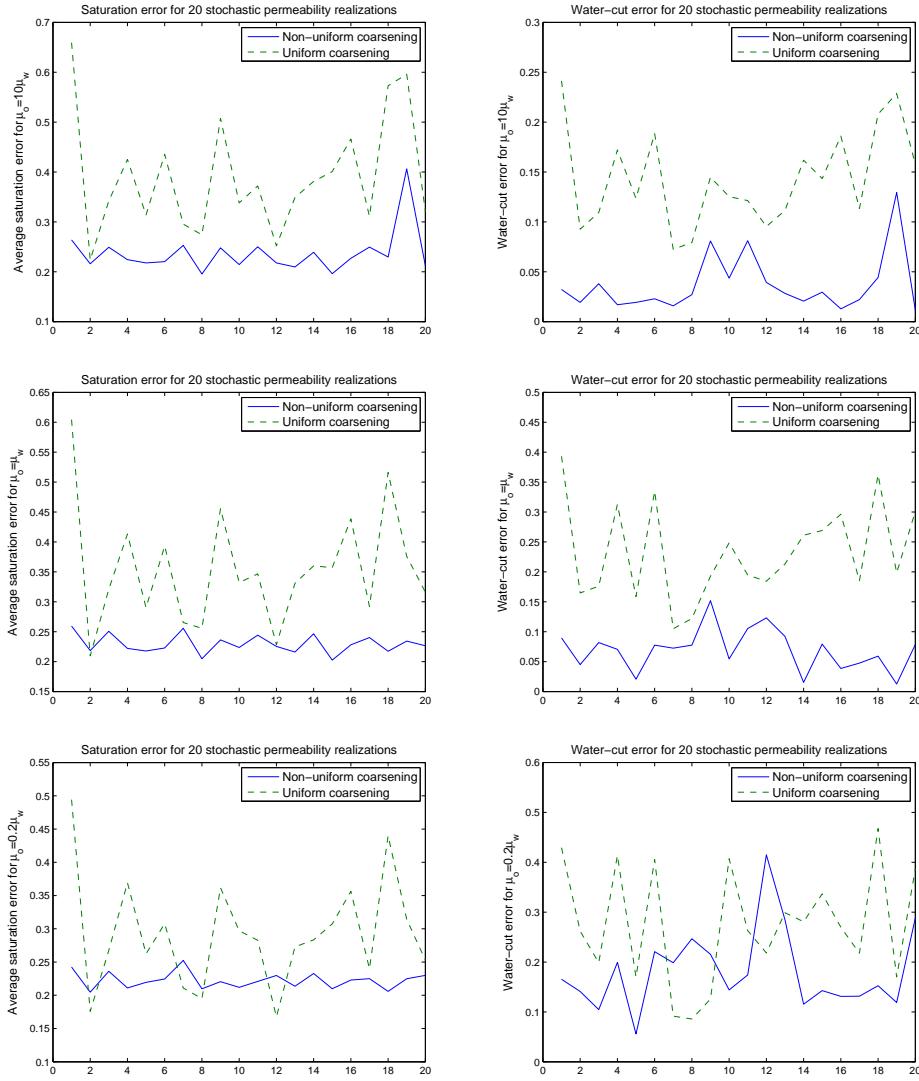


Fig. 7. Saturation and water-cut errors for 20 different permeability field realizations and viscosity ratio 0.1 (top), 1 (middle), and 5 (bottom).

Table 3

Mean of the errors plotted in Figure 7.

Viscosity ratio	$\mu_o = 10\mu_w$		$\mu_o = \mu_w$		$\mu_o = 0.2\mu_w$	
Coarsening strategy	NUC	UC	NUC	UC	NUC	UC
Mean saturation error	0.2370	0.3919	0.2296	0.3551	0.2224	0.2930
Mean water-cut error	0.0368	0.1439	0.0697	0.2337	0.1823	0.2750

5.4 Robustness

In the current section we seek to demonstrate the robustness of the proposed grid coarsening strategy. In particular we make an effort to show that the method provides accurate results for various degrees of coarsening and for various flow scenarios. To demonstrate the latter, we first consider flows imposed by different well-configurations. Next we pick one of the well-configurations and show that one obtains more accurate results with the non-uniform coarsening approach than with uniform coarsening, also when well-rates change during simulation. Finally we explore a flow scenario where the well-configuration changes rapidly during simulations. This case corresponds to a situation where new wells are drilled and old wells are shut down or reopened during simulation.

We would like to emphasize that we never regenerate the non-uniform coarse grid even though flow conditions change significantly so that the simulation velocity differs a lot from the velocity that is used to generate the coarse grid (cf. Algorithm 1).

In this section we revisit the SPE10 model. In particular, we consider the four bottom layers (from the fluvial Upper Ness formation) and use the same definition of the phase mobilities, i.e., λ_w and λ_o are defined by (16) with $\mu_o = 10\mu_w = 0.003$ cp. In each simulation we take 20 pressure steps and 200 saturation steps.

5.4.1 Robustness with respect to degree of coarsening

To assess robustness with respect to the degree of coarsening, we have selected five different Cartesian coarse grids, and chosen the parameters N_L and N_U correspondingly to create grids with comparable resolution. Table 4 shows the Cartesian grid dimensions along with the number of cells in the Cartesian grid and in the non-uniformly coarsened grid.

Here we use the same well-configuration as in [9]; one injector in the middle and a producer at each corner (well-configuration A in Figure 9). The corresponding saturation and water-cut errors are shown in Figure 8. The results show that the non-uniform coarse grids consistently allow more accurate results than one obtains with the corresponding uniformly coarsened Cartesian grids. In particular it is worth noticing that for the four finest grids we obtain water-cut curves that nearly match perfectly the reference curve (see also the water-cut curves corresponding to well-configuration A in Figure 11).

Table 4

Number of cells in each of the uniformly coarsened (Cartesian) and non-uniformly coarsened grids, respectively.

Cartesian grid	$30 \times 110 \times 4$	$20 \times 55 \times 4$	$15 \times 44 \times 2$	$10 \times 22 \times 2$	$6 \times 22 \times 1$
Number of cells	13200	4400	1320	440	132
NUC: N_L/N_U	2/6	4/16	10/40	25/150	50/500
Number of cells	7516	3251	1333	419	150

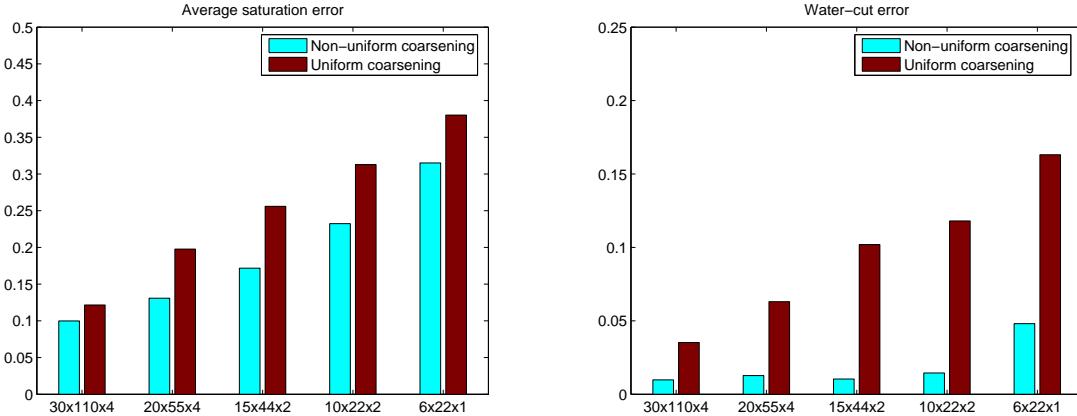


Fig. 8. Saturation and water-cut errors for five degrees of coarsening. For each case, the grid dimensions for the uniformly coarsened Cartesian grid is given.

The saturation errors shown in Figure 8 are computed on the fine grid. Hence since the coarse grid saturation solutions provide estimates for the average saturation in each block, we should indeed expect that the saturation error increases when the blocks become larger. However, the fact that we continue to produce accurate water-cut curves demonstrates that we are able to capture the qualitative behavior of the flow accurately, also on highly coarsened grids. The main reason for this is that even very coarse non-uniform coarse grids capture high flow channels quite well, as is illustrated in Figure 4.

5.4.2 Robustness with respect to well-placement

In this section we investigate the robustness with respect to well-locations. To this end we consider five test-cases with the five well-configurations shown in Figure 9. To create the coarse grid we use $N_L = 10$ and $N_U = 40$, i.e., the same parameters that were used to generate the third non-uniform coarse grid above. Figure 10 compares the saturation and water-cut errors obtained for each well-configuration with the corresponding errors obtained using a uniform $15 \times 44 \times 2$ Cartesian grid. The number of cells in the non-uniform coarse grids are shown in parenthesis along the lower edge of the plots. These plots show the same trend as Figure 8; with the non-uniform coarsening approach we consistently get significantly less saturation and water-cut errors than we obtain using the corresponding Cartesian coarse grid. This is further illustrated in Figure 11 which shows the respective water-cut curves for each well-configuration.

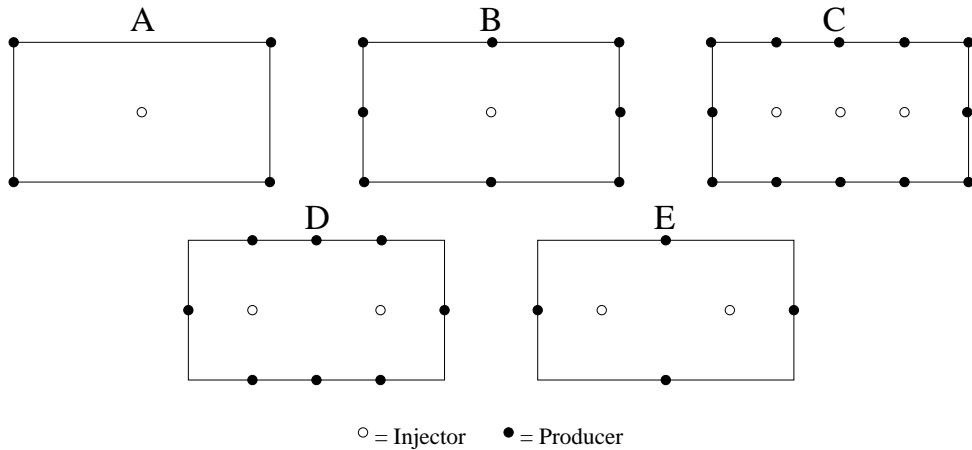


Fig. 9. Five selected well-configurations.

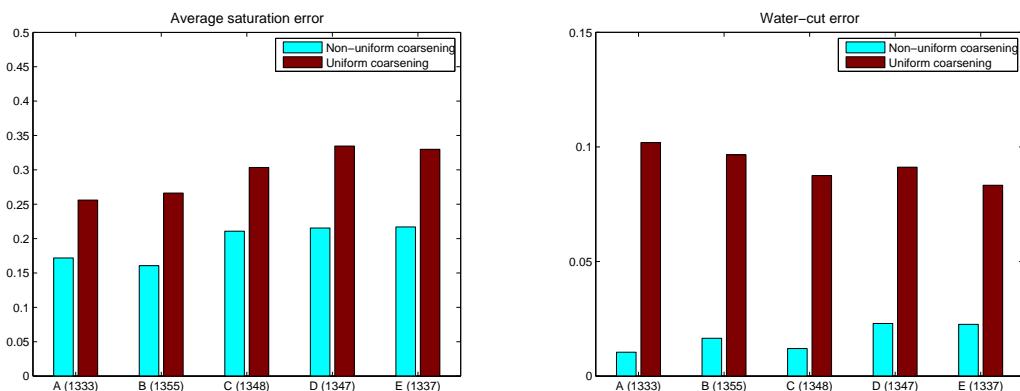


Fig. 10. Saturation and water-cut errors for each of the five well-configurations.

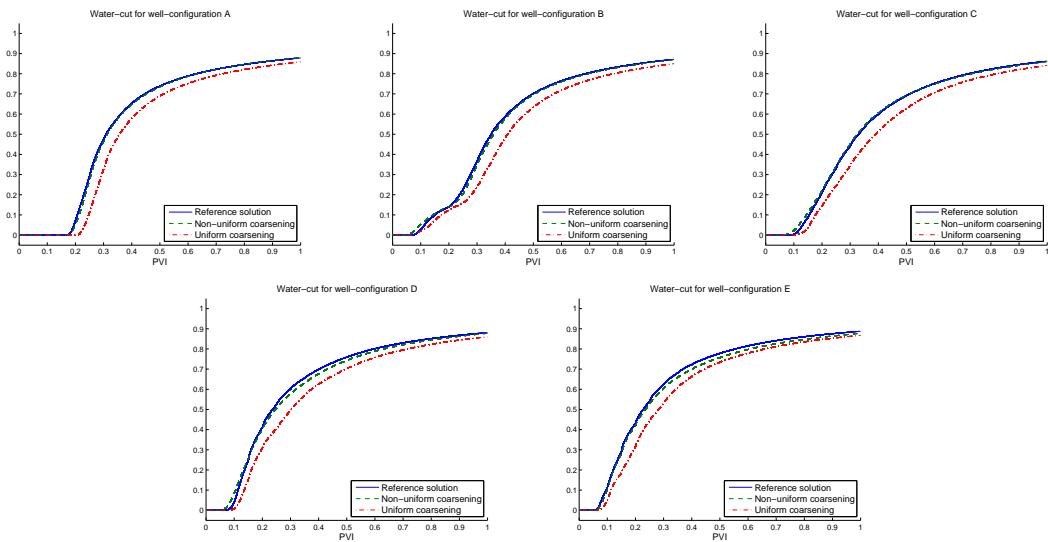


Fig. 11. Water-cut curves for each of the five well-configurations.

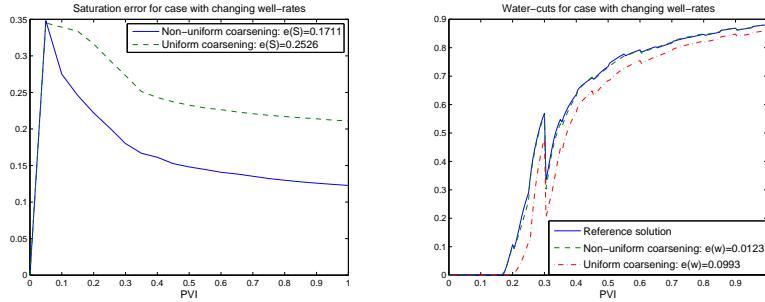


Fig. 12. Saturation and water-cut errors for a case where the well-rates change during simulation.

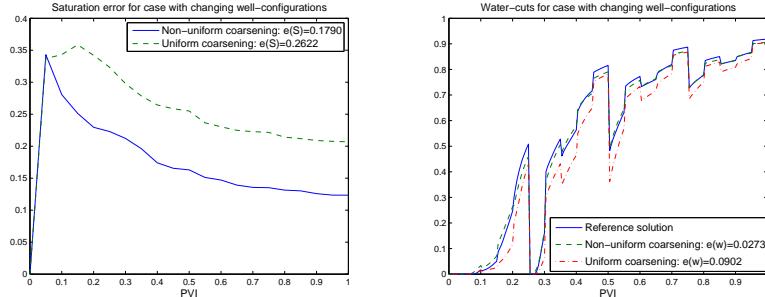


Fig. 13. Saturation and water-cut errors for a simulation where we rotate between the five well-configurations depicted in Figure 9.

5.4.3 Robustness with respect to changing flow conditions

Finally we demonstrate that the proposed gridding strategy is applicable also when flow conditions change significantly, i.e., by generating the non-uniformly coarsened grids at initial time only. To this end we consider first a case with well-configuration A presented above where only the well rates change. For each pressure step we select the total well rates for each producer from a random distribution, ensuring only that they sum up to the injection rate for compatibility. We employ $N_L = 10$, and $N_U = 40$ and compute the initial velocity field for the case where all producers have the same total well rate. Figure 12 shows how the saturation error $e(S, t)$ evolves along with the associated water-cuts. For both the non-uniform grid simulation and the uniform grid simulation we see that the error is about the same as for the case with fixed well-rates (Figure 11: well-configuration A). In particular, the water-cut for the non-uniformly coarsened grid still matches very closely the water cut for the reference solution.

Next we rotate among the well-configurations depicted in Figure 9. That is, for each pressure step we choose a new well-configuration so that at the end of a simulation we have employed each well-configuration four times. Qualitatively the results depicted in Figure 13 are very similar to the results depicted in Figure 12, and thus also to the results for fixed well-rates and fixed well-configurations seen in Figure 10 and Figure 11.

These examples indicate that it is not necessary to regenerate the non-uniform coarse grid when flow conditions change, i.e., when flow patterns during simulation are dominated by heterogeneity, but possibly differ substantially from the pattern used to generate the

grid. More detailed analysis of the effects of changing flow directions on the coarsening algorithm (e.g., for gravity dominated flows) is currently under investigation.

6 Summary and concluding remarks

In this paper we have presented a generic non-uniform coarsening strategy for modeling subsurface flow applications. The main objective has been to exploit information in high resolution geological models in an optimal way. The proposed grid coarsening algorithm is applicable to both structured and unstructured grids and gives consistently more accurate results compared to results obtained with uniformly coarsened grids with roughly the same number of cells. The key to the enhanced accuracy is that the flow velocity is more accurately resolved on the non-uniform coarse grid.

The current work is motivated in particular by previous work of Durlofsky, Jones, and Milliken [10] who proposed a strategy for generating coarse grids that are more finely gridded in high flow regions than in low flow regions. The main significance of our approach is that it is generic. Whereas the coarsening strategy in [10] applies primarily to Cartesian-like geomodels with high flow channels that are aligned with the grid, our approach applies to arbitrary grids and arbitrary heterogeneous structures. The algorithm is also conceptually simple and easy to implement. We therefore believe that the proposed approach should be valuable in an industrial setting, e.g., by allowing users to specify grid-resolution at run-time to fit available computer resources and project requirements.

The ability to generate grids with user-defined resolution at run-time brings us closer to having an Earth Model shared between reservoir engineers and reservoir geologists [15]. This means that the geological model provided by geologists is used as input for the reservoir simulator. Apart from giving the reservoir engineer a better description of the subsurface, this will also offer the geoscientist the ability to validate the geomodel before it is sent to the reservoir engineer. This can simplify and accelerate flow simulation workflows considerably, and thereby allow e.g., oil-companies to save many man-hours.

Acknowledgment: We would like to thank Alf B. Rustad at STATOIL for providing us with the corner-point grid model used for the flow simulations in Section 5.3.

References

- [1] J. E. Aarnes, *On the use of a mixed multiscale finite element method for greater flexibility and increased speed or improved accuracy in reservoir simulation*, Multiscale Model. Simul. **2** (2004), no. 3, 421–439.
- [2] J. E. Aarnes, V. Kippe, and K.-A. Lie, *Mixed multiscale finite elements and streamline methods for reservoir simulation of large geomodels*, Adv. Water Resour. **28** (2005), no. 3,

- [3] J. E. Aarnes, S. Krogstad, and K.-A. Lie, *Multiscale mixed/mimetic methods on corner-point grids*, Comput. Geosci. (to appear).
- [4] J. E. Aarnes, S. Krogstad, K.-A. Lie, and V. Laptev, *Multiscale mixed methods on corner-point grids: mimetic versus mixed subgrid solvers*, Tech. report, SINTEF, 2006.
- [5] T. Arbogast, *Numerical subgrid upscaling of two-phase flow in porous media*, Lecture Notes in Phys. (Z. Chen, R. Ewing, and Z.-C. Shi, eds.), Springer-Verlag, Berlin, 2000, pp. 35–49.
- [6] D. N. Arnold and F. Brezzi, *Mixed and nonconforming finite element methods: implementation, postprocessing and error estimates*, RAIRO Modél. Math. Anal. Numér. **19** (1985), no. 1, 7–32. MR 813687 (87g:65126)
- [7] Z. Chen and T.Y. Hou, *A mixed multiscale finite element method for elliptic problems with oscillating coefficients*, Math. Comp. **72** (2003), 541–576.
- [8] Z. Chen, G. Huan, and B. Li, *An improved impos method for two-phase flow in porous media*, Trans. Porous Media **54** (2004), 361–376.
- [9] M. A. Christie and M. J. Blunt, *Tenth SPE comparative solution project: A comparison of upscaling techniques*, SPE Reservoir Eval. Eng. **4** (2001), 308–317.
- [10] L. J. Durlofsky, R. C. Jones, and W. J. Milliken, *A nonuniform coarsening approach for the scale-up of displacement processes in heterogeneous porous media*, Adv. Water Res. **20** (1997), 335–347.
- [11] K. Lipnikov F. Brezzi, M. Shashkov, and V. Simoncini, *A new discretization methodology for diffusion problems on generalized polyhedral meshes*, LAUR-05-8717, Report, 2005.
- [12] K. Lipnikov F. Brezzi and V. Simoncini, *A family of mimetic finite difference methods on polygonal and polyhedral meshes*, Math. Models Methods Appl. Sci. **15** (2005), 1533–1553.
- [13] C. He and L. J. Durlofsky, *Structured flow-based gridding and upscaling for modeling subsurface flow*, Adv. Water Resour. **29** (2006), no. 12, 1876–1892.
- [14] P. Jenny, S. H. Lee, and H. A. Tchelepi, *Multi-scale finite-volume method for elliptic problems in subsurface flow simulation*, J. Comput. Phys. **187** (2003), 47–67.
- [15] M. J. King, K. S. Burn, P. Wang, V. Muralidharan, F. Alverado, X. Ma, and A. Datta-Gupta, *Optimal coarsening of 3d reservoir models for flow simulation*, SPE **95759** (2005).
- [16] V. Kippe, J. E Aarnes, and K.-A. Lie, *A comparison of multiscale methods for elliptic problems in porous media flow*, Comp. Geosci. (2006, to appear).
- [17] D. K. Ponting, *Corner point geometry in reservoir simulation*, Proceedings of the 1st European Conference on Mathematics of Oil Recovery, Cambridge, 1989 (Oxford) (P.R. King, ed.), Clarendon Press, July 25–27 1989, pp. 45–65.
- [18] P. A. Raviart and J. M. Thomas, *A mixed finite element method for second order elliptic equations*, Mathematical Aspects of Finite Element Methods (I. Galligani and E. Magenes, eds.), Springer-Verlag, Berlin – Heidelberg – New York, 1977, pp. 292–315.
- [19] X.-H. Wen and J. J. Gómez-Hernández, *Upscaling hydraulic conductivities in heterogeneous media: an overview*, J. Hydrol. (1996), no. 183, ix–xxxii.