

SPE 173306-MS

Application of Flow Diagnostics and Multiscale Methods for Reservoir Management

Knut-Andreas Lie, Olav Møyner, Stein Krogstad, SINTEF ICT.

Copyright 2015, Society of Petroleum Engineers

This paper was prepared for presentation at the SPE Reservoir Simulation Symposium held in Houston, Texas, USA, 23–25 February 2015.

This paper was selected for presentation by an SPE program committee following review of information contained in an abstract submitted by the author(s). Contents of the paper have not been reviewed by the Society of Petroleum Engineers and are subject to correction by the author(s). The material does not necessarily reflect any position of the Society of Petroleum Engineers, its officers, or members. Electronic reproduction, distribution, or storage of any part of this paper without the written consent of the Society of Petroleum Engineers is prohibited. Permission to reproduce in print is restricted to an abstract of not more than 300 words; illustrations may not be copied. The abstract must contain conspicuous acknowledgment of SPE copyright.

Abstract

Reservoir simulation workflows contain significant elements of uncertainty, particularly in the geological description of reservoir geometry and petrophysical parameters such as permeability and porosity. To accurately account for uncertainty and span the range of likely outcomes, different equiprobable realizations should be kept as long as possible throughout a modelling workflow. However, working with multiple realizations of the same reservoir for optimization purposes may quickly become prohibitively expensive, particularly since using a full forward simulation can be quite demanding even for a single model realization. Herein, we propose to combine two recent and quite different technologies to enable optimization of multiple realizations. The first is the use of multiscale technology, wherein approximate, but well-behaved pressure solutions can be efficiently computed using precomputed basis functions that capture local flow features. Secondly, the use of single-phase flow diagnostics can serve as an efficient alternative to full physics flow simulations for optimization and characterization purposes. By combining these technologies in a single implementation, we obtain a workflow that makes it possible to quickly evaluate and optimize multiple realizations, while still retaining error control. In particular, it is possible to adjust accuracy dynamically from inexpensive proxy models provided by pure multiscale and flow diagnostics, via more accurate iterated multiscale solutions and incompressible flow, to fully-implicit solvers that incorporate the relevant flow physics.

Introduction

Computational tools for reservoir modeling play a critical role in the development of long-term strategies for optimal recovery of hydrocarbon resources. Reservoir simulation, in particular, can to a large extent realistically describe fluid flow in the reservoir on the time scale associated with reservoir management and offers a means of forecasting recovery based on available data and a set of modeling assumptions about the reservoir. However, to be able to span the range of possible and likely scenarios, the reservoir engineer must be able to efficiently validate and verify alternative hypotheses, systematically explore the parameter space, and assess how recovery forecasts are influenced by uncertainty in assumptions, data, and operating constraints.

Unfortunately, reservoir simulation is computationally demanding and a single simulation on a full reservoir model that includes comprehensive description of geology, reservoir fluids, flow physics, well controls, and coupling to surface facilities may require from a few tens of minutes to hours or even days. This is at odds with modern reservoir characterization techniques, which have shifted focus from traditional variogram-based descriptions towards object and feature-based models, of which hundreds of equiprobable realizations may be generated to quantify uncertainty in the characterization. Dynamic simulation of large ensembles of high-resolution geo-cellular models is not practical with full-featured simulations, and traditionally the computational gap has been bridged by (aggressive) upscaling or by clustering the ensembles based on fast proxies and then selecting an appropriate subset of models for flow simulation.

Herein, we propose a somewhat alternative approach that combines two recent and quite different technologies for model reduction of flow simulations in an attempt to keep multiple realizations of fine-scale geological details as long as possible in various optimization workflows. Our first technology component is so-called flow-diagnostics methods, which briefly can be explained as numerical flow experiments that are run to probe a reservoir model, establish connections and basic volume estimates, and quickly provide a qualitative picture of the flow patterns in the reservoir. The starting point is a computation of time-of-flight and numerical tracer partitions, from which we can compute volumes that are connected to a well or a well pair, well-allocation factors, heterogeneity measures, sweep efficiencies, etc. Time-of-flight and and derived quantities have traditionally been associated with streamline simulation (Datta-Gupta and King 2007) and have been used for ranking and upscaling (Idrobo et al. 2000; Ates et al. 2005; Shook and Mitchell 2009), identifying reservoir compartmentalization (He et al. 2004), rate optimization (Thiele and Batycky 2003; Park and Datta-Gupta 2011; Izgec et al. 2011), and flood surveillance (Batycky et al. 2008). Two

recent papers (Shahvali et al. 2012; Møyner et al. 2014) demonstrate that the same type of information can easily be computed using standard finite-volume discretizations. This may have certain advantages since finite-volume discretizations are already available in standard simulators and better developed than streamline methods for unstructured grids with general polyhedral cells, multi-continuum models, models with lower-dimensional objects, and so on. Regardless of the specific spatial discretization, flow diagnostics methods aim to be so computationally efficient that they provide an answer within seconds when applied to a single model or can process large ensembles of realizations within the computational time currently associated with a single full-featured flow simulation. As such, flow diagnostics provides an efficient approach to ranking, comparing, and validating reservoir models and upscaling procedures. The methods can also potentially assist in the challenging task of integrating data into reservoir models through fast screening of multiple scenarios, identifying regions associated with mismatches, and suggesting appropriate model updates.

Another area where flow diagnostics may prove useful is for field-development planning and execution of production plans. Here, the high computational cost limits the number of multiphase simulations one can afford to run in search for an optimal strategy. During optimization, however, there may be limited need for the complexity of a full simulation and flow diagnostics based on models with reduced flow physics will in many cases provide sufficient quantitative information to steer the optimization procedure in the right direction. In (Møyner et al. 2014), we demonstrate how flow diagnostics can be combined with rigorous mathematical optimization methods to optimize well placements and drilling sequences. The methods can also suggest plausible production plans that optimize approximate functionals for net-present value, operational expenditures, or similar economical measures. These production plans are derived from models with reduced physics and will not necessary satisfy multiphase operational constraints. The production plans are therefore fed as a series of *targets* to a more comprehensive flow simulation, which will follow what the flow diagnostics suggests when the suggestion is feasible, but modify or switch controls if constraints are violated.

One limitation of flow diagnostics is that these computations require at least one known set of fluxes for the region of interest. Sometimes, these fluxes can be backed out from a previous reservoir simulation, but in most cases the only way to obtain a representative flow field is to solve an incompressible pressure equation using information of the fluid distribution, and if this is not available, by assuming single-phase flow. In terms of computational cost, this is generally the most time-consuming part of a flow-diagnostics computation, which motivates our second technology component: Multiscale pressure solvers promise significantly reduced runtimes compared with traditional, fully-implicit simulators, and at the same time offer a systematic framework for increasing local model resolution (Efendiev and Hou 2009). This makes multiscale simulation a better tool for characterizing volumetric sweep and locating bypassed and immobile oil compared with traditional upscaling, which always implies a loss of information when homogenizing fine-scale structures. Previous research has shown that multiscale and streamline methods constitute a powerful and computationally efficient combination (Aarnes et al. 2005; Stenerud et al. 2008; Natvig et al. 2011).

The key idea of multiscale methods is to separate small-scale and large-scale flow effects. The global mechanisms that drive flow are resolved by a coarse discrete system with a reduced number of unknowns, in which each degrees of freedom corresponds to a so-called multiscale basis function that accounts for localized heterogeneity effects and enables the reconstruction of subscale flow fields. The basis functions are computed numerically by solving a flow problem with appropriate boundary conditions that localize the functions' support. Recent research on multiscale methods has primarily focused on making these methods into highly efficient iterative solvers for computing the exact discrete solution of fine-scale, multiphase flow models, see e.g., (Hajibeygi 2011; Wang et al. 2014; Hajibeygi and Tchelepi 2014) and references therein. However, multiscale methods also provide a framework for variable-fidelity simulations that compute approximate, conservative, but qualitatively correct predictions of flow patterns at computational cost that is much lower than for solving the fine-scale problem. We believe that utilizing this less emphasized aspect of multiscale methods in combination with flow diagnostics can lead to new and innovative workflows within reservoir management. The purpose of the paper is to take the first step in this direction and provide a few illustrative examples of potential use cases. The methods presented in the following have all been implemented using the MATLAB Reservoir Simulation Toolbox (Lie et al. 2012; Krogstad et al. 2015; MRST 2014).

Flow Diagnostics

The term flow diagnostics, as we use it herein, refers to methods that can be exploited to reveal information about communication and flow patterns in a reservoir without running a full dynamic simulation. There are several methods that can be characterized as being flow diagnostics. One recent idea is to use fast-marching methods to compute pressure propagation, from which one can determine radius/depth of investigation Datta-Gupta et al. (2011); Zhang et al. (2013), provide dynamic ranking of model ensembles (Sharifi et al. 2014), verify dynamic upscaling (Sharifi and Kelkar 2014), and perform well testing (Lallier et al. 2014). Herein, however, we focus on methods that analyze the properties of a single static flow field. This flow field can be extracted from a dynamic simulation, but in the following we will assume that it is computed by solving an incompressible pressure equation of the form

$$\nabla \cdot \vec{v} = q, \qquad \vec{v} = -\mathsf{K}\lambda\nabla p,\tag{1}$$

where \vec{v} is the Darcy velocity and p the fluid pressure. If the fluid distribution in the reservoir is known, this is accounted for in the relative mobility λ , which otherwise is set equal unity. Apart from this, the primary inputs will be the absolute permeability K, the porosity ϕ , and a description of wells to determine the volumetric fluid sources q.

Time-of-flight. From the static Darcy velocity \vec{v} one can compute the forward and backward *time-of-flight*, which denote the time it takes an imaginary particle released at the inflow to reach a given point in the reservoir, and the travel time from a given point to the nearest outflow point, respectively. Forward time-of-flight is usually associated with streamline methods, but can also be posed in terms of a steady-state transport equation,

$$\vec{v} \cdot \nabla \tau_f = \phi, \qquad \tau_f |_{\text{inflow}} = 0,$$
(2)

and computed from a standard finite-volume or higher-order discontinuous Galerkin discretizations (Natvig et al. 2006, 2007; Eikemo et al. 2009; Shahvali et al. 2012; Rasmussen and Lie 2014). The backward time-of-flight τ_b is computed analogously by setting a zero condition at the outflow and reversing the sign of the flow field. Isocontours of time-of-flight define natural time lines in the reservoir and will indicate how multiphase displacements will evolve under fixed well and boundary conditions. Time-of-flight therefore gives more information than pressure and velocity alone and can be exploited to study how heterogeneity in petrophysical parameters affect flow patterns or identify regions that are likely to remain unswept or undrained, which will show up as having high time-of-flight values.

Tracer partitions. In addition to time lines, we may want to delineate the reservoir into sub-volumes that are in communication with a well or can be associated with a given pair of injection–production wells. These communication patterns and volume partitions can be determined by simulating an imaginary tracer experiment in which a unique, neutral and non-diffusive tracer is injected in each injector. To this end, we freeze the flow pattern, start injecting a unit concentration of each tracer, and compute the overall steady state the tracer concentration approaches as time goes to infinity. A unit concentration of one tracer indicates that this part of the reservoir is only influenced by the corresponding well, while non-unit concentrations indicate parts of the reservoir that are influenced by more than one injection well. Obviously, such an experiment cannot be carried out in real life, but it is easy to compute numerically by solving,

$$\nabla \cdot (\vec{v}c_i) = 0, \qquad c_i|_{\text{inflow}} = 1. \tag{3}$$

By reversing the flow field, the same type of tracer distribution can be computed for the producers.

We can now define partitions that delineate the reservoir volume into units associated with unique injectors or producers by thresholding the tracer distributions. The resulting tracer partitions give the portion of volume that *eventually* will come from a given injector or arrive at a given producer if the driving forces used to compute the snapshot of the flux field prevails. Similar partitions can also easily be computed for individual well completions or segments, or parts of inflow or outflow boundaries.

In Eqs. 2 and 3, we see that the only difference between τ_f and c_i is the boundary conditions and the right-hand sides. Hence, the computation of τ_f and all c_i can be formulated as a single linear system with multiple right-hand sides and computed efficiently, e.g., using a flow-based reordering technique (Natvig et al. 2007; Natvig and Lie 2008) that permutes the linear system into triangular or block-triangular form and also enables us to effectively localize the computation to sub-regions of a large reservoir model if we are only interested in a subset of the wells and well perforations.

Derived quantities. From the tracer partitions it is straightforward to determine quantities such as sweep volumes, drainage volumes, well-pair connections for wells or individual completions. One can also compute production/injection allocation factors that give the amount of flow in/out of a producer/injector that can be attributed to another injector/producer. These quantities are all visually intuitive and can, possibly in combination with time-of-flight, be used to understand injector-producer coupling of flow, investigate areal/vertical sweep and drainage patterns, rank geomodels, verify model upscaling, identify regions that need to be modified in a history match, and so on.

As suggested by Shook and Mitchell (2009) and Shahvali et al. (2012), one can also use time-of-flight to generalize measures of heterogeneity from classical sweep theory (Lake 1989) like flow and storage capacity and the Lorenz coefficient, which is a popular measure of displacement heterogeneity. To define these quantities, we introduce the total travel time, $\hat{\tau} = \tau_f + \tau_b$, defined as the sum of the forward and backward time-of-flight. We can then define the storage capacity $\Phi(\tau)$ as the cumulative pore volume of all cells that have a total travel time $\hat{\tau} \leq \tau$, whereas the flow capacity $F(\tau)$ is the cumulative flux associated with all cells that have $\hat{\tau} \leq \tau$. For incompressible flow, this flux is given as the ratio between the pore volume and the total travel time. If we normalize F and Φ , the displacement can now be interpreted as a 1D process that has $F(\Phi)$ as its fractional flow curve. This curve is a straight line if the displacement is homogeneous. The Lorenz coefficient L_c is defined as the area bounded above by the $F(\Phi)$ curve and below by the line $F = \Phi$, normalized by a factor 0.5. The coefficient measures the difference in flow capacity between an infinitely heterogeneous displacement for which $L_c = 1$ and a planar, piston-like displacement for which $L_c = 0$. In (Møyner et al. 2014), we demonstrate how the Lorenz coefficient can act as a simple flow proxy for recovery factors and be used to optimize well placement and injection rates.

Multiscale Pressure Solvers

A large number of multiscale methods have been proposed over the past two decades to solve second-order elliptic equations of the form as in Eq. 1 with strongly heterogeneous coefficients, see e.g., (Babuka et al. 1994; Hou and Wu 1997; Chen and Hou 2003; Jenny et al. 2003; Aarnes 2004; Arbogast et al. 2007; Hajibeygi 2011). All methods rely on two nested grids: a fine grid



Fig. 1—Examples of flow diagnostics used in reservoir management. *Visualisation:* To the left, a drainage volume associated with a producer is delineated into volumes associated with injector-producer pairs. The middle plots shows sweep volumes and time-of-flight for an artificial well pattern on the Gullfaks field, whereas the right plot shows volumetric flow associated with each injector-producer pair. *Ranking:* the upper plot shows correlation between oil recovered and the Lorenz coefficient for a set of equiprobable permeability realizations for four different fluid models. The lower plots show seven different realizations. *Upscaling:* match in production/injection allocation factors between fine-grid solution (solid lines) and coarse-grid solution (colors) is used to determine the quality of upscaling. *Optimisation:* an approximate measure of net-present value derived from time-of-flight is used as objective function to optimise the injection strategy for the Norne field. See (Møyner et al. 2014) for more details.

that gives the geometry and the petrophysical properties of the reservoir, and a coarse grid on which one discretizes the flow equations to compute the flow driven by global forces such as gravity, injection and production wells, aquifer support, etc. To link the two grids, the multiscale methods define and solve a set of localized flow problems, whose solutions are concatenated to define operators (or basis functions) that correctly account for subscale variation in the formulation of the coarse-grid equations and likewise provide a way of mapping the coarse-scale unknowns to approximate solutions defined on the fine grid. Different methods are distinguished by how they define the localized flow problems, construct the coarse-scale flow problem, and map back solutions back to the fine grid.

By far, the most well-developed method in terms of geomodel complexity is the multiscale mixed finite-element method, which is posed in terms of degrees-of-freedom associated with the fluxes between neighboring coarse blocks. The method only requires a primal coarse partition and can hence easily be formulated on structured, stratigraphic grids as well as fully-unstructured, polyhedral grids (Aarnes et al. 2006, 2008; Krogstad et al. 2009; Natvig et al. 2011; Alpak et al. 2012; Lie et al. 2014). An extended approach based on proper orthogonal decompositions (POD) was suggested by Krogstad (2011) for problems in which one wants to investigate new cases that represent minor changes to scenarios that have already been simulated by a multiphase simulation. On the other hand, mixed formulation are not widely used in reservoir simulators and herein we will therefore instead focus on finite-volume methods that rely on multiscale approximation of the pressure field.

To formalize our description, we start by the discrete version of our flow equation, Eq. 1, defined over a fine grid $\Omega = \{C_i\}_{i=1}^n$,

$$\boldsymbol{A}\boldsymbol{p}=\boldsymbol{q}, \tag{4}$$

Herein, the fine-scale discretization will exclusively come from applying the standard two-point flux-approximation scheme. The vector p has one unknown pressure value per cell C_i , whereas q accounts for the known forcing terms from wells and body



Fig. 2—Illustration of the multiscale restriction-smoothed basis method (Møyner and Lie 2015).

forces. The matrix A is sparse and positive definite. On top of the fine grid, we define a coarse grid $\Omega = \{B_j\}_{j=1}^m$ in which each coarse block $\overline{\Omega}_j$ consists of a collection of cells from the fine grid. We then introduce the operator P that maps the vector p_c of unknown pressures on the coarse grid to the vector p of unknowns on the fine grid,

$$\boldsymbol{p}_f = \boldsymbol{P} \boldsymbol{p}_c. \tag{5}$$

There are many ways to construct this operator, and this is where different multiscale finite-volume methods deviate most from each other. Examples of specific choices are the multiscale finite-volume method (Jenny et al. 2003; Møyner and Lie 2014a), the multiscale two-point method (Møyner and Lie 2014b), and the restriction-smoothed basis method (Møyner and Lie 2015). The latter is particularly simple, flexible, and robust and will be our method of choice herein. The key steps in the method are illustrated in **Fig. 2**. The prolongation operator P is constructed as a sum of local prolongation operators P_j that each have support on an interaction region I_j that contains B_j . Each P_j is constructed iteratively by setting $P_j^{n+1} = P_j^n - \omega D^{-1} A P_j^n$, where D is the diagonal of the local fine-scale discretization matrix A, ω is a relaxation parameter, and P_j^0 is set to one inside B_j and zero outside. This iterative scheme will eventually cause P_j to grow outside of I_j , and to prevent this, the update $-\omega D^{-1}A P_j^n$ is modified so that it is zero outside of I_j and preserves partition of unity inside, see (Møyner and Lie 2015) for details. It is also possible to define prolongation operators that are composed of basis functions having global support; we will get back to one such example later in the paper.

Next, we need to define a restriction operator \mathbf{R} to map quantities associated with the fine grid to the coarse grid. This will either be a control-volume summation operator or a Galerkin operator,

$$(\boldsymbol{R}_{cv})_{ji} = \begin{cases} 1, & \text{if } C_i \in B_j, \\ 0, & \text{otherwise,} \end{cases} \quad \text{or} \quad \boldsymbol{R}_G = \boldsymbol{P}^T.$$
(6)

If only one degree of freedom is associated with each coarse block, the operators P and R are represented as $n \times m$ and $m \times n$ matrices.

We now have all we need to formulate and solve a coarse system. Inserting Eq. 5 into Eq. 4 and pre-multiplying by one of the restriction operators from Eq. 6, we obtain

$$\boldsymbol{R}(\boldsymbol{A}(\boldsymbol{P}\boldsymbol{p}_{c})) = (\boldsymbol{R}\boldsymbol{A}\boldsymbol{P})\boldsymbol{p}_{c} = \boldsymbol{R}\boldsymbol{q}, \qquad \longrightarrow \qquad \boldsymbol{A}_{c}\boldsymbol{p}_{c} = \boldsymbol{q}_{c}. \tag{7}$$

The physical interpretation of this system depends on the restriction operator; for a finite-volume summation operator, the system simply describes conservation of the coarse-scale fluxes. In general, p_f will not be an exact solution of Eq. 4; what we hope for is to compute a good approximation more efficiently than by solving Eq. 4 directly. This means that if we use p_f directly to compute approximate fluxes on the fine grid, they will not be conservative. To get conservative fluxes, we use Darcy's law on

the coarse scale to compute mass-conservative fluxes v_c from p_c using Darcy's law and then impose these fluxes as Neumann boundary conditions on local flow problems formulated on the coarse blocks B_j to reconstruct fine-scale fluxes v_f that can be used to compute time-of-flight and tracer partition as outlined in the previous section.

Uncertainty in Fault Multipliers

The primary elements that impact uncertainty in a reservoir characterization include the structural model, petrophysical properties, barriers to flow, the location of the interface between different fluids, and to what degree faults act as barrier to fluid flow. As our first example of a workflow, we will present a method that can rapidly evaluate the impact varying fault transmissivity has on flow patterns. Faults can either act as conduits for fluid flow or create flow barriers and introduce compartmentalization that affects fluid distribution and reduces hydrocarbon recovery. It is therefore crucial to understand and accurately account for the impact that faults have on the fluid flow. Although faults are volumetric objects, the predominant way of modeling faults in commercial reservoir simulators is to introduce so-called transmissibility multipliers that reduce (or increase) the flow between adjacent cells on the opposite side of the fault. The value of the fault multipliers usually vary in the interval [0, 1], where a zero value means that the fault is closed to fluid flow, a unit value characterizes an open fault, whereas a value in between means that the fault is a partial barrier to flow. In principle, it is also possible to have values larger than unity to model increased permeability inside the grid cells, close to the fault plane. Fault multiplier values generally have large associated uncertainties and it is therefore interesting to evaluate flow responses for many and distinctly different values in the unit interval. The number of possible combinations that need to be investigated increases dramatically with the number of faults in the reservoir, and using traditional forward simulations to investigate them all quickly becomes computationally intractable. We therefore propose an alternative method that combines flow diagnostics and multiscale pressure solvers.

We consider water-flooding scenarios with relatively long inter-well distances. If the main driving mechanism is the pressure difference between injection and production wells, the recovery factor is dictated by the areal sweep efficiency, which again is determined mainly by the heterogeneity in the reservoir. We have previously shown that the Lorenz coefficient correlates well with the amount of hydrocarbons recovered after a fixed time for cases in which the areal sweep efficiency is the main factor (Møyner et al. 2014). The logic behind this correlation is that the lower the Lorenz coefficient becomes, the closer the recovery is to a piston-like displacement and hence a high recovery factor. Herein, we will consider cases in which the areal sweep is determined by a combination of heterogeneity and fault transmissivity. To investigate the impact of fault transmissivity, we use the Lorenz coefficient to rank model ensembles that represent different combinations of fault transmissivity levels. The fault system will introduce a certain degree of compartmentalization in the reservoir, where the transmissivity across faults will determine the level of fluid communication between different compartments and affect the global flow directions, whereas flow patterns in the interior of each compartment are mainly determined by local heterogeneity. This observation motivates a partition of the problem into two types of regions: regions close to the fault where the local flow directions may change significantly (e.g., from being orthogonal to being transverse) depending on the degree of transmissivity, and regions away from faults where the fault transmissivities influence the boundary conditions but not the local heterogeneity of the flow field in the interior. We will consider cases where our grid model contains a number of faults, but it is unclear which faults actually impede flow and to what degree.

Multiscale method. The partition outlined above can now be utilized to formulate an efficient multiscale approach for the model ensemble. The first step is to introduce a coarse partition that distinguishes between coarse blocks that belong to the 'fault region' and coarse blocks that are well away from the faults and in what we will refer to as the 'heterogeneity region'. Fig. 3 shows two examples of such partitions, that also include a refinement of near-well regions. The first case is a 200×200 fine grid with uniform aspect ratio, lognormal permeability distribution, and three different faults. The second case has the same type of heterogeneity but contains eight faults. Both cases have the same quarter five-spot well configuration with an injector in the southwest corner and a producer in the northeast corner. In the heterogeneity and the near-well region, we compute basis functions once and use these precomputed quantities to determine local variations in the flow field when the global flow directions have been determined by the coarse-scale system. Likewise, for blocks in the fault region we compute one basis function for each fault-multiplier value that will be included when forming the model ensemble, see Fig. 4. Basis functions for the MsRSB method are straightforward to construct for general coarse partitions. However, to obtain good quality approximation near faults, we ensure that the coarse blocks do not cross the fault lines and that the block centers are chosen to be close to the fault faces. The constructed basis will then only have support over the fault if the multiplier is nonzero. Looking at Fig. 4, we can see how the fault multiplier influences the basis functions. Since the basis functions have limited support, it is straightforward to swap between different basis functions in the vicinity of uncertain features while keeping the remaining basis functions static. In other words, to compute the pressure for a realization it is then sufficient to pick the correct basis functions corresponding to the combination of multipliers under evaluation.

In theory, the largest computational effort for the multiscale solver is the construction of basis functions. If we assume that this cost dominates the runtime of any multiscale evaluation and that the cost of each basis function is roughly the same, we see that the cost to construct all basis functions for a case with N_c coarse blocks in the heterogeneity region and N_f faults with N_m possible multipliers each is

$$C_{MS} = \mathcal{O}(N_c + (N_m - 1)N_f). \tag{8}$$



Fig. 3—Two examples of coarse partitions that adapt to faults and decomposes the global domain into 'fault regions' and 'heterogeneity regions'. The colors represent the logarithm of the isotropic permeability in units $[m^2]$. The left plot shows a 200×200 case with a mild heterogeneity and three faults, whereas the right plot shows a 200×100 case with eight faults.



Fig. 4—Definition of fault regions and computation of basis functions for various values of the fault transmissibility multipliers. In the left plot, light blue color indicates the heterogeneity region whereas darker color denotes the fault region.

The cost for computing a fine-scale flow field for each realization in a straightforward (N_m, N_f) -way combinatorial ensemble using a standard linear solver is

$$C_{fine} = \mathcal{O}(N_m^{N_f}). \tag{9}$$

Obviously, Eq. 9 scales worse than Eq. 8. The question is whether a multiscale approximate fluxes obtained by the use of precomputed basis functions, and possibly also with some inexpensive iterative updates, correlate well with the true fine-scale fluxes. If so, we have a method that can rapidly evaluate fluxes for a large ensemble of realizations. To answer this, we will use the two cases shown in Fig. 3.

Three-fault case. Relatively large changes in the fault transmissivities are generally needed to significantly change the flow pattern in the reservoir. To build our ensemble, we therefore consider four multiplier values [0, 0.1, 0.5, 1] that span the whole interval from closed to open. With three faults in the model, this gives an ensemble of $4^3 = 64$ different members that need to be evaluated. **Fig. 5** shows the resulting ranking and spread in Lorenz numbers for the ensemble computed by a standard fine-scale solver and by our multiscale method. The figure also shows pressure fields and the F- Φ diagrams for the end-members corresponding to all faults open and all faults closed. We clearly see that when the faults act as barriers to flow, they increase the areal sweep and hence reduce the Lorenz coefficient compared to the case with open faults. Secondly, we observe that away from the faults, the flow pattern is quite invariant to whether the faults are open or closed, which was our hypothesis when formulating our multiscale solution strategy. Although certain discrepancies can easily be spotted in the predicted pressure fields, in particular in the fault region for the case with closed faults, the fine-scale and the multiscale solvers compute flow patterns that are qualitatively in close correspondence. There is also a close match between the corresponding F- Φ curves.

For the ranking, we used the Lorenz coefficient computed by the fine-scale solver to first sort the realizations in ascending order and then plot the corresponding Lorenz coefficients computed by the multiscale solver. From Fig. 5 we see that the multiscale are consistently larger than those computed by the fine-scale solver. Having a consistent bias, e.g., as can be observed when comparing low and high-order discretizations of the time-of-flight equation (Rasmussen and Lie 2014), is generally not problematic for most uses of flow diagnostics. Here, however, the multiscale Lorenz coefficients do not form a monotone



Fig. 5—Ranking of model ensemble for the simple box model. The left and middle columns report pressure fields computed by the multiscale solver and the fine-scale reference solver for two realizations of the model ensemble. The right column shows the corresponding F- Φ diagrams and the ranking in terms of increasing Lorenz coefficient obtained for the whole ensemble with four multiplier values.

sequence, which implies that the resulting ranking will not be the same as for the fine-scale solver. Fortunately, this deficiency is quite simple to cure: By increasing the number of iteration cycles in the multiscale method, see (Møyner and Lie 2015) for details, the resulting Lorenz values approach a monotone sequence. These multiscale cycles are inexpensive to compute and even if they may double the computational cost of the multiscale solve, computing the pressure approximation will be significantly faster than on the fine grid. However, to compute the flow diagnostics we need a fine-scale flow field. In Fig. 5 we have used a conservative fine-scale fluxes that can be reconstructed by solving one local flow problem per coarse block with the coarse-scale fluxes as Neumann conditions. When formulated as one global linear system (as we have done in our simple prototype Matlab solver), this operation is almost as expensive as solving the fine-scale problem directly. However, because each local reconstruction is independent of the other, the problem has a large degree of parallelism that can potentially be exploited to increase efficiency. The reconstruction cost can be avoided by the use of the MsMFE method Aarnes et al. (2008), for which conservative fine-scale fluxes are obtained directly by prolongation. This may work well in many cases, but in other, the approximate flux fields contain small cycles that may locally make the time-of-flight problem ill posed.

Eight-fault case. The three fault model was conceptually easy to examine. We will now look briefly at a more complex case that contains eight faults, see the plot to the right in Fig. 3. We compute approximate fluxes using a multiscale method with five or fifteen cycles. To avoid the cost of our inefficient flux reconstruction, we attempt to compute the fine-scale fluxes directly from the multiscale pressure solution, which will give fluxes that are not conservative. The multiscale solver produces fluxes that are qualitatively correct after five cycles. However, the time-of-flight discretization is quite sensitive to minor inaccuracies and inconsistencies in the input fluxes and will every now and then fail to give sensible travel times. With only five multiscale iterations, we could therefore not use the non-conservative fluxes but needed to reconstruct conservative fluxes to be able to compute time-of-flight. Qualitatively, these time-of-flight solutions are correct and hard to distinguish visually from the reference solution. However, the relatively pointwise discrepancy can be up to ten percent. This significantly changes the Lorenz coefficient, which in our experience is very sensitive to local changes in time-of-flight. With fifteen multiscale cycles, on the other hand, the prolongated pressure solution gives high-quality fine-scale fluxes. Thus, we do not need the expensive reconstruction step to compute time-of-flight fields and accurately reproduce the fine-scale Lorenz ranking.



Fig. 6—Ranking by Lorenz coefficient for an ensemble with 6561 members for the case with eight faults. In the plot, the ensemble members are sorted according to the ranking predicted by the fine-scale solver. The five-cycle multiscale solution uses reconstruction of conservative fluxes, whereas the fifteen-cycle multiscale solution computes fine-scale fluxes directly from the prolongated pressures.

Fig. 6 shows the ranking of an ensemble consisting of $3^8 = 6561$ ensemble members corresponding to multiplier values 0, 0.1, and 1. With five iterations (and flux reconstruction) the multiscale solver is able to get the general trend. Unfortunately, the prediction for each specific ensemble member is often is far off and the method can therefore not be used in workflows that directly compare individual ensemble members. In the figure, we observe that multiscale method is more accurate for lower Lorenz coefficients, which correspond to cases dominated by closed faults. For ensemble members with high values, the differences in Lorenz coefficient are dominated by cells with high time-of-flight values. These cells are found near the boundary of the domain, where the subscale resolution of the multiscale method is lower and, our experience, needs to be improved by additional iterations. By increasing the number of iterations to fifteen, the prediction is well within the discretization error of the time-of-flight equation (see e.g., (Rasmussen and Lie 2014)) and can be used for optimization workflows as outlined in (Møyner et al. 2014).

The multiscale solver outlined above can also be used to accelerate full multiphase simulations. However, unless one also accelerates the transport part of the problem, one cannot expect the multiscale simulations to be more than a few times faster compared with a sequential fine-scale solver and up to one-order of magnitude faster than a fully-implicit simulator, see (Møyner and Lie 2015).

Waterflood Optimization

Mathematical methods for waterflood optimization typically require a large number of forward simulations to converge to an optimal well configuration, and since full flow simulations generally are costly, rigorous optimization runs tend to be prohibitively expensive. In (Møyner et al. 2014), we demonstrated how to use quantities derived from time-of-flight as proxies for rapid optimization of reservoir management problems such as finding optimal well-control and placements with respect to recovery or net-present value (NPV). Herein, we demonstrate how such methods can be even further accelerated by using multiscale methods for the pressure solves that are part of the methodology.

Optimization approach. We start by briefly outlining the mathematical formulation and our optimization approach. Let x denote a state-vector containing grid-cell pressures, well rates, well bottom-hole pressures, and any number of time-of-flight and tracer fields. Moreover, let u denote the vector of controls, i.e., well-rates. Given an objective function J(x, u), we consider optimization problems of the form

$$\max_{\boldsymbol{u}} J(\boldsymbol{x}(\boldsymbol{u}), \boldsymbol{u}), \quad \text{s.t.} \quad \boldsymbol{g}(\boldsymbol{x}(\boldsymbol{u}), \boldsymbol{u}) = \boldsymbol{0}, \quad \boldsymbol{A}_i \boldsymbol{u} \leq \boldsymbol{b}_i, \quad \boldsymbol{A}_e \boldsymbol{u} = \boldsymbol{b}_e, \quad (10)$$

where g(x, u) = 0 are the *reservoir equations* resulting from our discretization of Eq. 1 in combination with Eqs. 2 and 3. Furthermore, we include linear inequality and equality constraints, which typically will be box constraints, upper total injection/production limits, and/or volume balance. As in (Møyner et al. 2014), we employ an adjoint evaluation of the reservoir equations to obtain a gradient $\nabla_{u_i} J$ of the objective with respect to the current control. Here, we utilize a quasi-Newton method,



Fig. 7—Global basis approach. Five basis functions corresponding to five wells span the solution space. Any pressure solution is simply obtained by a linear combination (with weights equal to prescribed bhp values) of the five basis functions

namely BFGS (Wright and Nocedal 1999), to iteratively construct approximations H_i^{-1} of the Hessian inverse, which is subsequently used to compute a search direction $d_i = -H_i^{-1} \nabla_{u_i} J$. When active constraints are present (equality constraints are always active), the gradient and Hessian approximations need to be projected onto the null-space of the active constraints. Accordingly, searching along a direction involves iteratively modifying the null-space projection as more constraints become active. We use a line-search algorithm based on cubic interpolation with termination criteria based on the Wolfe conditions Wolfe (1969).

The efficiency of the optimization algorithm as described above, rests on the assumption that the response surfaces for our problems are smooth and can be approximated locally by quadratic functions. This is not always the case for objectives based on flow diagnostics, as especially idealized simple problems may have optima that are sharp peaks. However, for more complex problems, we have not observed such optima and the approach outlined above appears to be efficient.

Global basis functions. For optimization of bottom-hole pressures (bhp) and rates based on flow diagnostics for deterministic problems (one single realization), the dimension of the solution space for the linear pressure equation becomes particularly low. In particular, if the controls u are well-controls (either rate or bhp) for all n_w wells, one can span the entire space by solving the pressure equation with n_w right-hand-sides. One way of producing a spanning basis, is to pick one well, impose and compute the pressure field resulting from imposing unit bhp in this well and zero bhp in all other wells, and then do the same for all other wells. This approach is illustrated in Fig. 7 for an idealized problem with five wells. If all wells are controlled by bottom-hole pressure, solving the pressure equation merely becomes computing a linear combination of the global basis functions. For rate-controlled wells or a mixture of rate and bhp-controls, the basis functions enable us to compute a linear mapping

$$\boldsymbol{q}_w = \boldsymbol{M} \boldsymbol{p}_{bhp}, \tag{11}$$

from the vector of bottom-hole pressures to the vector of (total) well rates. Note that the *small* $n_w \times n_w$ matrix M has rank $n_w - 1$ since the problem is incompressible. For a given combination of rate/bhp-controlled wells, M can be split accordingly, and a subsystem of Eq. 11 can be solved for the unknown bottom-hole pressures. Alternatively, since M typically will be small, one can invert the corresponding block directly prior to the optimization loop. We also remark that once we have constructed the mapping in Eq. 11, we can with no extra cost easily include linear constraints on both well rates and bhps in the optimization problem given by Eq. 10 regardless of the choice of control. Note that this is contrary to simulation-based optimization, in which e.g., a bhp-constraint on a rate-controlled well is a nonlinear output constraint. We finally note the global-basis approach outlined here will be the most efficient *multiscale* method if we disregard the CPU-time spent computing basis functions. For large cases with many wells, the computation cost of the global basis functions can be considerable, and thus an approach that uses local bases may still be appropriate even when optimizing deterministic problems.

Numerical example. In this example we consider a realistic representation of a progradational shallow-marine reservoir generated by the multidisciplinary SAIGUP modeling project (Manzocchi et al. 2008), which focused on assessing the influence of geological factors on production in a large suite of synthetic models. The faulted model consists of 78 720 cells, has anisotropic permeability, and contains both channel-like and layered structures. We consider a two-phase flow model, in which the oil and water phase initially are completely segregated. Five vertical producers are located in the oil-zone, while five water injectors are located in the water-zone encompassing the oil. Well configuration, model permeability, and initial saturation are depicted in **Fig. 8**. In (Møyner et al. 2014) we presented a flow-diagnostics *proxy* for computing the net-present-value (NPV) of a field. Here we will employ and optimize a slightly modified NPV-proxy by controlling producer and injector rates. Recall that the NPV is



Fig. 8—Well configuration and horizontal permeability (left) and initial saturation (right).

the discounted, accumulated, net cash-flow, and that a simple version for two-phase flow can be expressed as

$$NPV(T) = \int_{t=0}^{T} \sum (r_o q_o - r_{wi} q_{wi} - r_{wp} q_{wp}) (1+d)^{-t} dt,$$
(12)

where T is the time horizon, q_o , q_{wi} and q_{wp} are the field-oil, water-injection, and water-production rates. Furthermore, r_o is the oil revenue, while r_{wi} and r_{wp} are costs per unit volume associated with water injection and production, respectively. We then define our NPV-proxy as follows:

$$NPV\text{-}proxy(T) = \sum_{\ell=1}^{N} NPV_{\ell}(\tau_{b,\ell}, T) - \int_{t=0}^{T} \sum r_{wi} q_{wi} (1+d)^{-t} dt,$$
(13)

where NPV $_{\ell}$ is an estimate of the contribution from grid cell ℓ . In particular, we let

$$NPV_{\ell} = PV_{\ell}(r_o f_{o,\ell} - r_{wp} f_{w,\ell})(1+d)^{-\tau_{b,\ell}} c_T(\tau_{b,\ell}),$$
(14)

where $\tau_{b,\ell}$ is the backward time-of-flight of grid cell ℓ and $f_{o,\ell}, f_{w,\ell}$ are the fractional flows for oil and water for grid cell ℓ . Since we are considering a fixed time horizon T, grid cells with $\tau_{b,\ell} > T$ should not contribute to the NPV, and hence a cutoff is appropriate. However, for optimization purposes we wish to maintain a smooth response, and therefore make a smooth approximation of the cut-off:

$$c_T(\tau) = \frac{T - \tau}{2\sqrt{(T - \tau)^2 + \epsilon^2}} + \frac{1}{2}.$$
(15)

Here, we use a *smoothing*-factor $\epsilon = 0.1T$, which for the case considered appeared to be a good compromise between smoothness and accuracy. We finally note that if the time horizon T is longer than the shortest travel time from injector to producer, an extra term in Eq. 13 is needed to account for inter-well water breakthrough, but is skipped here for brevity.

For our numerical experiment we set $r_o = 100$ \$/stb, $r_{wp} = 5$ \$/stb, $r_{wi} = 10$ \$/stb, and use a discount factor of 10 %/year. Producers are limited to produce at liquid rates $\leq 3500 \text{ m}^3/\text{day}$, and injectors $\leq 6000 \text{ m}^3/\text{day}$. Furthermore the total (field) injection rate is set to $15000 \text{ m}^3/\text{day}$ and the time horizon is six years. In the base configuration (prior to optimization) all wells are set to inject/produce at $3000 \text{ m}^3/\text{day}$. We consider three different fluid configurations with water-to-oil mobility ratios of three and ten. For each of the two cases, we

- 1. perform rapid optimization of the NPV-proxy to obtain optimal controls u_{opt}^{proxy} ,
- 2. run a full simulation to evaluate NPV(u_{opt}^{proxy}),
- 3. perform a lengthy simulation-based optimization using u_{opt}^{proxy} as initial guess to obtain optimal controls u_{opt} .

The numerical experiment was run on a standard laptop where the average time for solving the discretized pressure equation was about 1.3 seconds using an efficient algebraic multigrid solver. For the (backward) TOF equation, we used the built-in linear solver in Matlab with an average solution time of 0.05 seconds. The initial optimization of the NPV-proxy required about 15 forward and adjoint function evaluations, so for the global basis approach, the total linear solver time (basis computations)



Fig. 9—Evolution of NPV-functions for cases $\mu_w/\mu_o = 3$ (left) and $\mu_w/\mu_o = 10$ (right).



Fig. 10—Optimal controls for cases $\mu_w/\mu_o = 3$ (left) and $\mu_w/\mu_o = 10$ (right). Dashed lines indicate upper rate limits for injectors (blue) and producers (red)

+ optimization) amounted to about $10 \times 1.3 + 2 \times 15 \times 0.05 = 14.5$ seconds. Accordingly, in an efficient implementation, diagnostics-based optimization of a problem of this size should be achievable in less than 30 seconds, while for our prototype code, it took about two minutes. The final simulation-based optimization, however, took approximately two hours, even though the total runtime was reduced considerably by the fact that the proxy-optimization provided an initial guess *close* to an optimal setting. The evolution of the NPV-curves for the two cases are depicted in **Fig. 9**. We observe that the improvement we obtain by performing a computationally intensive, simulation-based optimization using the controls resulting from the proxy optimization as well targets is only marginal. **Fig. 10** depicts the optimal controls for the two cases.

Concluding Remarks

In this paper we have presented an computational methodology for efficient ranking of reservoir model and fast optimization of reservoir management. The methodology is based on a recent multiscale finite-volume method in combination with single-phase flow-diagnostics. The suggested approach does not simulate dynamic multiphase flow as in standard reservoir simulation. Instead, it computes qualitatively correct flow patterns and measures of displacement efficiency within an interactive time-frame (within seconds). Accordingly, the methodology is tailored towards workflows like screening and ranking of multiple models and visualization of fluid communication in the reservoir. The same approach can be used to compute flow proxies and optimize aggregated measures such as long-term recovery, net-present value, etc.

The performance measures used in flow diagnostics are based on time-of-flight and stationary, single-phase tracer equations. These equations require a flux-field, typically obtained by solving a discretized pressure equation. On large models, solving the pressure equation requires orders-of-magnitude higher CPU-time than computing time-of-flight and tracers. To speed up the combined methodology, we therefore utilize multiscale pressure solvers.

We demonstrate the utility of our approach on two classes of problems. First, we discuss the use of the Lorenz coefficient to rank model ensembles with uncertainty localized to fault multipliers. An important advantage of using a multiscale pressure

solver for this problem, is that the majority of multiscale basis functions can be reused over the entire set of model realizations. However, we did observe that the Lorenz coefficient is (perhaps surprisingly) sensitive to small perturbations in the time-of-flight fields. Accordingly, a number of iterations were required for the multiscale solver to produce results in close agreement with corresponding fine-scale solvers.

For rate-optimization of net-present value, we considered the use global basis functions. When optimizing a single model realization, this is the optimal approach with respect to coarsening-factor, as the number of basis functions and the dimension of the single-phase solution space are the same. The optimal schedules obtained within a couple of minutes by the diagnostics optimization were validated against a CPU-intensive, simulation-based optimization, which only gave marginal improvements.

Acknowledgments

This work was funded in part by Chevron, Schlumberger Information Solutions, and the Research Council of Norway under grants no. 215665 and 226035. We thank Brad Mallison and Jostein R. Natvig for stimulating discussions.

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. doi: 10.1137/030600655.
- Aarnes, J. E., Kippe, V., and Lie, K.-A. 2005. Mixed multiscale finite elements and streamline methods for reservoir simulation of large geomodels. Adv. Water Resour., 28(3):257–271. doi: 10.1016/j.advwatres.2004.10.007.
- Aarnes, J. E., Krogstad, S., and Lie, K.-A. 2006. A hierarchical multiscale method for two-phase flow based upon mixed finite elements and nonuniform coarse grids. *Multiscale Model. Simul.*, 5(2):337–363. doi: 10.1137/050634566.
- Aarnes, J. E., Krogstad, S., and Lie, K.-A. 2008. Multiscale mixed/mimetic methods on corner-point grids. *Comput. Geosci.*, 12(3):297–315. doi: 10.1007/s10596-007-9072-8.
- Alpak, F. O., Pal, M., and Lie, K.-A. 2012. A multiscale method for modeling flow in stratigraphically complex reservoirs. SPE J., 17(4):1056– 1070. doi: 10.2118/140403-PA.
- Arbogast, T., Pencheva, G., Wheeler, M. F., and Yotov, I. 2007. A multiscale mortar mixed finite element method. *Multiscale Model. Simul.*, 6(1):319–346. doi: 10.1137/060662587.
- Ates, H., Bahar, A., El-Abd, S., Charfeddine, M., Kelkar, M., and Datta-Gupta, A. 2005. Ranking and upscaling of geostatistical reservoir models using streamline simulation: A field case study. SPE Res. Eval. Eng., 8(1):22–32. doi: 10.2118/81497-PA.
- Babuka, I., Caloz, G., and Osborn, J. 1994. Special finite element methods for a class of second order elliptic problems with rough coefficients. *SIAM J. Numer. Anal.*, 31(4):945–981. doi: 10.1137/0731051.
- Batycky, R. P., Thieles, M. R., Baker, R. O., and Chugh, S. H. 2008. Revisiting reservoir flood-surveillance methods using streamlines. SPE Res. Eval. Eng., 11(2):387–394. doi: 10.2118/95402-PA.
- Chen, Z. and Hou, T. 2003. A mixed multiscale finite element method for elliptic problems with oscillating coefficients. *Math. Comp.*, 72:541–576. doi: 10.1090/S0025-5718-02-01441-2.
- Datta-Gupta, A. and King, M. J. 2007. Streamline Simulation: Theory and Practice, volume 11 of SPE Textbook Series. Society of Petroleum Engineers.
- Datta-Gupta, A., Xie, J., Gupta, N., King, M. J., and Lee, W. J. 2011. Radius of investigation and its generalization to unconventional reservoirs. *J. Petrol. Technol.*, 63(7):52–55.
- Efendiev, Y. and Hou, T. Y. 2009. *Multiscale Finite Element Methods*, volume 4 of *Surveys and Tutorials in the Applied Mathematical Sciences*. Springer Verlag, New York.
- Eikemo, B., Lie, K.-A., Dahle, H. K., and Eigestad, G. T. 2009. Discontinuous Galerkin methods for transport in advective transport in single-continuum models of fractured media. *Adv. Water Resour.*, 32(4):493–506. doi: 10.1016/j.advwatres.2008.12.010.
- Hajibeygi, H. 2011. Iterative multiscale finite volume method for multiphase flow in porous media with complex physics. PhD thesis, ETH Zurich. doi: 10.3929/ethz-a-006696714.
- Hajibeygi, H. and Tchelepi, H. A. 2014. Compositional multiscale finite-volume formulation. SPE J., 19(2):316–326. doi: 10.2118/163664-PA.
- He, Z., Parikh, H., Datta-Gupta, A., Perez, J., and Pham, T. 2004. Identifying reservoir compartmentalization and flow barriers from primary production using streamline diffusive time of flight. *SPE J.*, 7(3):238–247. doi: 10.2118/88802-PA.
- Hou, T. and Wu, X.-H. 1997. A multiscale finite element method for elliptic problems in composite materials and porous media. *J. Comput. Phys.*, 134:169–189. doi: 10.1006/jcph.1997.5682.

- Idrobo, E., Choudhary, M., and Datta-Gupta, A. 2000. Swept volume calculations and ranking of geostatistical reservoir models using streamline simulation. In SPE/AAPG Western Regional Meeting, Long Beach, California, USA. SPE 62557.
- Izgec, O., Sayarpour, M., and Shook, G. M. 2011. Maximizing volumetric sweep efficiency in waterfloods with hydrocarbon f-φ curves. *Journal* of Petroleum Science and Engineering, 78(1):54–64. doi: 10.1016/j.petrol.2011.05.003.
- Jenny, P., Lee, S. H., and Tchelepi, H. A. 2003. Multi-scale finite-volume method for elliptic problems in subsurface flow simulation. *J. Comput. Phys.*, 187:47–67. doi: 10.1016/S0021-9991(03)00075-5.
- Krogstad, S. 2011. A sparse basis POD for model reduction of multiphase compressible flow. In 2011 SPE Reservoir Simulation Symposium, The Woodlands, Texas, USA, 21-23 February 2011. doi: 10.2118/141973-MS.
- Krogstad, S., Lie, K.-A., Møyner, O., Nilsen, H. M., Raynaud, X., and Skaflestad, B. 2015. Mrst-ad an open-source framework for rapid prototyping and evaluation of reservoir simulation problems. In SPE Reservoir Simulation Symposium, 23–25 February, Houston, Texas. doi: 10.2118/173317-MS.
- Krogstad, S., Lie, K.-A., Nilsen, H. M., Natvig, J. R., Skaflestad, B., and Aarnes, J. E. 2009. A multiscale mixed finite-element solver for threephase black-oil flow. In SPE Reservoir Simulation Symposium, The Woodlands, TX, USA, 2–4 February 2009. doi: 10.2118/118993-MS.
- Lake, L. W. 1989. Enhanced Oil Recovery. Prentice-Hall.
- Lallier, F. L., Montouchet, M., Bergey, P., and Vignau, S. 2014. An efficient well test forward model based on the fast marching method application to geomodel sorting. In *ECMOR XIV 14th European conference on the mathematics of oil recovery*. EAGE. doi: 10.3997/2214-4609.20141869.
- Lie, K.-A., Krogstad, S., Ligaarden, I. S., Natvig, J. R., Nilsen, H. M., and Skaflestad, B. 2012. Open source MATLAB implementation of consistent discretisations on complex grids. *Comput. Geosci.*, 16:297–322. doi: 10.1007/s10596-011-9244-4.
- Lie, K.-A., Natvig, J. R., Krogstad, S., Yang, Y., and Wu, X.-H. 2014. Grid adaptation for the Dirichlet–Neumann representation method and the multiscale mixed finite-element method. *Comput. Geosci.*, 18(3):357–372. doi: 10.1007/s10596-013-9397-4.
- Manzocchi, T. et al. 2008. Sensitivity of the impact of geological uncertainty on production from faulted and unfaulted shallow-marine oil reservoirs: objectives and methods. *Petrol. Geosci.*, 14(1):3–15.
- Møyner, O., Krogstad, S., and Lie, K.-A. 2014. The application of flow diagnostics for reservoir management. SPE J. accepted, doi: 10.2118/171557-PA.
- Møyner, O. and Lie, K.-A. 2014a. The multiscale finite-volume method on stratigraphic grids. SPE J., 19(5):816–831. doi: 10.2118/163649-PA.
- Møyner, O. and Lie, K.-A. 2014b. A multiscale two-point flux-approximation method. J. Comput. Phys., 275:273–293. doi: 10.1016/j.jcp.2014.07.003.
- Møyner, O. and Lie, K.-A. 2015. A multiscale method based on restriction-smoothed basis functions suitable for general grids in high contrast media. In SPE Reservoir Simulation Symposium held in Houston, Texas, USA, 23-25 February 2015. SPE 173265-MS, doi: 10.2118/173256-MS.
- MRST 2014. The MATLAB Reservoir Simulation Toolbox, version 2014b. http://www.sintef.no/MRST/.
- Natvig, J. R. and Lie, K.-A. 2008. Fast computation of multiphase flow in porous media by implicit discontinuous Galerkin schemes with optimal ordering of elements. J. Comput. Phys., 227(24):10108–10124. doi: 10.1016/j.jcp.2008.08.024.
- Natvig, J. R., Lie, K.-A., and Eikemo, B. 2006. Fast solvers for flow in porous media based on discontinuous Galerkin methods and optimal reordering. In Binning, P., Engesgaard, P., Dahle, H., Pinder, G., and Gray, W., editors, *Proceedings of the XVI International Conference on Computational Methods in Water Resources*, Copenhagen, Denmark.
- Natvig, J. R., Lie, K.-A., Eikemo, B., and Berre, I. 2007. An efficient discontinuous Galerkin method for advective transport in porous media. *Adv. Water Resour.*, 30(12):2424–2438. doi: 10.1016/j.advwatres.2007.05.015.
- Natvig, J. R., Skaflestad, B., Bratvedt, F., Bratvedt, K., Lie, K.-A., Laptev, V., and Khataniar, S. K. 2011. Multiscale mimetic solvers for efficient streamline simulation of fractured reservoirs. *SPE J.*, 16(4):880–888. doi: 10.2018/119132-PA.
- Park, H.-Y. and Datta-Gupta, A. 2011. Reservoir management using streamline-based flood efficiency maps and application to rate optimization. In Proceedings of the SPE Western North American Region Meeting, 7-11 May 2011, Anchorage, Alaska, USA. doi: 10.2118/144580-MS.
- Rasmussen, A. F. and Lie, K.-A. 2014. Discretization of flow diagnostics on stratigraphic and unstructured grids. In ECMOR XIV 14th European Conference on the Mathematics of Oil Recovery, Catania, Sicily, Italy, 8-11 September 2014. EAGE. doi: 10.3997/2214-4609.20141844.
- Shahvali, M., Mallison, B., Wei, K., and Gross, H. 2012. An alternative to streamlines for flow diagnostics on structured and unstructured grids. *SPE J.*, 17(3):768–778. doi: 10.2118/146446-PA.

- Sharifi, M. and Kelkar, M. 2014. Novel permeability upscaling method using fast marching method. *Fuel*, 117, Part A(0):568–578. doi: 10.1016/j.fuel.2013.08.084.
- Sharifi, M., Kelkar, M., Bahar, A., and Slettebo, T. 2014. Dynamic ranking of multiple realizations by use of the fast-marching method. *SPE J.* doi: 10.2118/169900-PA.
- Shook, G. and Mitchell, K. 2009. A robust measure of heterogeneity for ranking earth models: The F-Phi curve and dynamic Lorenz coefficient. In SPE Annual Technical Conference and Exhibition, 4-7 October, New Orleans, Louisiana. doi: 10.2118/124625-MS.
- Stenerud, V. R., Kippe, V., Lie, K.-A., and Datta-Gupta, A. 2008. Adaptive multiscale streamline simulation and inversion for high-resolution geomodels. SPE J., 13(1):99–111. doi: 10.2118/106228-PA.
- Thiele, M. R. and Batycky, R. P. 2003. Water injection optimization using a streamline-based workflow. In *Proceedings of the SPE Annual Technical Conference and Exhibition*, 5-8 October 2003, Denver, Colorado. doi: 10.2118/84080-MS.
- Wang, Y., Hajibeygi, H., and Tchelepi, H. A. 2014. Algebraic multiscale solver for flow in heterogeneous porous media. J. Comput. Phys., 259:284–303. doi: 10.1016/j.jcp.2013.11.024.
- Wolfe, P. 1969. Convergence conditions for ascent methods. SIAM review, 11(2):226–235.
- Wright, S. J. and Nocedal, J. 1999. Numerical optimization, volume 2. Springer New York.
- Zhang, Y., Yang, C., King, M. J., and Datta-Gupta, A. 2013. Fast-marching methods for complex grids and anisotropic permeabilities: Application to unconventional reservoirs. In SPE Reservoir Simulation Symposium, 18-20 February, The Woodlands, Texas, USA. doi: 10.2118/163637-MS.