Noname manuscript No.
(will be inserted by the editor)

Efficient Flow Diagnostics Proxies for Polymer Flooding

Stein Krogstad · Knut-Andreas Lie · Halvor Møll Nilsen · Carl Fredrik Berg · Vegard Kippe

Received: date / Accepted: date

Abstract Flow diagnostics refers to a family of numerical methods that within a few seconds can compute visually intuitive quantities illuminating flow patterns and well connections for full 3D reservoir models. The starting point is a flow field, extracted from a previous multiphase simulation or computed by solving a simplified pressure equation with fixed mobilities. Time-of-flight (TOF) and stationary tracer equations are then solved to determine approximate time lines and influence regions. From these, one can derive sweep or drainage regions, injector-producer regions, and well allocation factors, as well as dynamic heterogeneity measures that characterize sweep and displacement efficiency and correlate (surprisingly) well with oil recovery from waterflooding processes.

This work extends flow diagnostics to polymer flooding. Our aim is to develop inexpensive flow proxies that can be used to optimize well placement, drilling sequence, and injection strategies. In particular, we seek proxies that can distinguish the effects of improved microscopic and macroscopic displacement. To account for the macroscopic effect of polymer injection, representative flow fields are computed by solving the reservoir equations with linearized flux functions. Although this linearization has a pronounced smearing effect on water and polymer fronts, we show that the heterogeneity of the total flux field is adequately represented. Subsequently, transform the flow equations to streamline coordinates, map saturations from physical coordinates to time-of-flight, and (re)solve a representative 1D flow problem for each well-pair region. A recovery proxy is then obtained by accu-

Stein Krogstad, Knut-Andreas Lie, Halvor Møll Nilsen
SINTEF Digital, Mathematics and Cybernetics
P.O. Box 124 Blindern, N–0314 Oslo, Norway
E-mail: {Stein.Krogstad,Knut-Andreas.Lie,HalvorMoll.Nilsen}@sintef.no

Carl Fredrik Berg
Department of Geoscience and Petroleum, NTNU
Trondheim, Norway
E-mail: carl.f.berg@ntnu.no

Vegard Kippe
Statoil Research Centre, Rotvoll
Arkitekt Ebbells vei 10, N-7005 Trondheim, Norway
E-mail: vkip@statoil.com

mulating each 1D solution weighted by a distribution function that measures the variation in residence times for all flow paths inside each well-pair region. We apply our new approach to 2D and 3D reservoir simulation models, and observe close agreements between the suggested approximations and results obtained from full multiphase simulations. Furthermore, we demonstrate how two different versions of the proxy can be utilized to differentiate between macroscopic and microscopic sweep improvements resulting from polymer injection. For the examples considered, we demonstrate that macroscopic sweep improvements alone correlate better with measures for heterogeneity than the combined improvements.

Keywords Flow diagnostics, tracer distributions, time-of-flight, polymer flooding, simplified physics proxy

1 Introduction

Modern reservoir simulators provide detailed forecasts of hydrocarbon recovery based on a description of reservoir geology, flow physics, well controls, and couplings to surface facilities. To interpret these simulations, it is common to study well profiles and 3D visualization of pressure, saturation, and component distributions in the reservoir. However, this is seldom sufficient to develop an understanding of how the reservoir reacts to changes in production strategies. A reservoir engineer will also want to know which injection and production wells are in communication; what is the sweep and displacement efficiency within a given drainage, sweep, or well-pair region; which regions of the reservoir are likely to remain unswept, and so on. Likewise, one must understand how different parameters in the reservoir model and their inherent sensitivity affect the recovery forecasts. Detailed simulations of field models take hours or days, and this limits the ability to iteratively perturb simulation input to evaluate and build *cause-and-effect* knowledge of the model. Rapid screening capability and simple, efficient, and interactive tools that can be used to develop basic understanding of how the fluid flow is affected by reservoir geology and how the flow patterns in the reservoir respond to engineering controls are needed to accelerate modelling workflows, make better use of time-consuming simulation runs, and provide better data for decision support.

The term *flow diagnostics*, as used here, denotes a class of simple and controlled numerical flow experiments 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, either as a standalone prescreening tool or to post-process standard multiphase simulations [32, 22]. Flow diagnostics can also be used to compute quantitative information about the recovery process in settings somewhat simpler than what would be encountered in actual fields, or be used to perform what-if and sensitivity analyzes in a parameter region surrounding a pre-existing simulation. As such, these methods offer a computationally inexpensive alternative to full-featured multiphase simulations to provide flow information in various reservoir management workflows.

Two quantities are fundamental in flow diagnostics: time-of-flight and volumetric (tracer) partitions. Time-of-flight τ denotes the time it takes a neutral particle to flow from the nearest inlet to a given point in the reservoir and defines natural

time lines that describe how displacement fronts will propagate under prevailing flow conditions for an instantaneous flow field $\mathbf{v}$. Time-of-flight has traditionally been associated with streamline methods [8, 35], but can equally well be computed by standard finite-volume methods [24, 25, 32]. Using a finite-volume formulation extends better to unstructured grids and provides more seamless integration with standard modelling tools currently used in industry. On differential form, τ is given as

$$\mathbf{v} \cdot \nabla \tau = \phi, \quad \tau|_{\text{inflow}} = 0, \tag{1}$$

where ϕ is the porosity of the reservoir. Similarly, we can define an equation that follows the reverse velocity field $-\mathbf{v}$ from the outflow boundary to compute travel time from a point in the reservoir to the outflow boundary.

Volumetric partitions and measures of to what extent each cell in the reservoir is in communication with the different fluid sources and sinks can be determined by computing numerical tracer distributions. These distributions can be though of as resulting from artificial tracer injections continued until time infinity under steady flow conditions. Normalized tracer concentrations are given by simple advection equations on the form

$$\mathbf{v} \cdot \nabla c = 0, \quad c|_{\text{inflow}} = 1. \tag{2}$$

The tracer concentration will equal one in all points in communication with the inflow boundary and be undetermined elsewhere. The inflow boundary typically consists of multiple wells, or well segments, and/or interfaces between the reservoir and aquifers. To derive a volumetric partition, we associate a unique tracer to each part of the inflow boundary (e.g., one tracer for each injector), and solve the corresponding tracer equations numerically by a finite volume method. The default choice would be the single-point upwind method commonly used in multiphase reservoir simulators. If a grid cell is in communication with a single injector only, the corresponding tracer concentration equals one and the others are zero. If a grid cell is in communication with multiple injectors, each nonzero tracer value is the fraction of the volumetric flow through the cell that can be attributed to the corresponding injector. Tracer distributions associated with outflow boundaries (producers) are computed similarly from the *reverse* flow field. From time-of-flight and tracer distributions, one can derive various quantities that express volumetric connections and flow patterns such as drainage and sweep regions, well-pair connections and flow volumes, and well-allocation factors, which all are visually intuitive quantities giving enhanced understanding during pre- and post-processing [22]; see the illustration in Figure 1.

The ultimate goal of most reservoir simulation studies is to contribute to maximize profit given a set of operational and economic constraints. To this end, one needs to explore various production strategies and perform a number of what-if and sensitivity analyzes. Sweep theory from classical reservoir engineering includes a number of heterogeneity measures for the variation in petrophysical properties like flow and storage capacity, the Lorenz and Dykstra–Parsons coefficients, etc. [17]. It has been shown that time-of-flight can be used to generalize this theory to a dynamic setting to provide measures of the heterogeneity in flow paths rather than in static reservoir properties. Heterogeneity measures like sweep efficiency, Lorenz coefficient, and vorticity index have proved to correlate well with recovery [13, 29, 22]. These measures are all inexpensive to compute, and with a finite-volume formulation it is also straightforward to develop adjoint equations

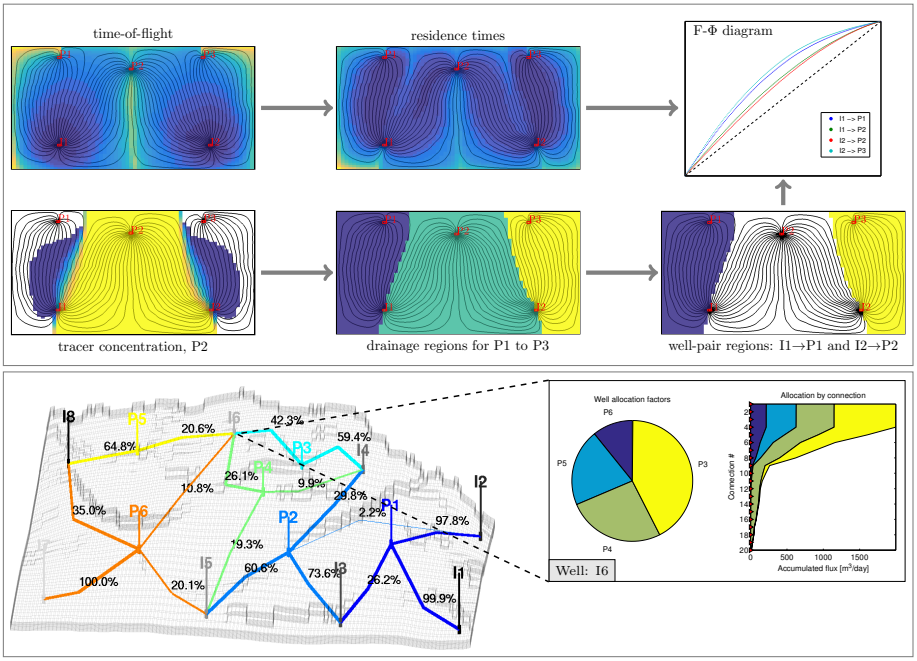

Fig. 1: Conceptual illustration of flow diagnostics. Time-of-flight gives travel time along streamlines, whereas numerical tracers provide partition of unity for the reservoir volume. From these quantities, one can derive residence times of flow paths, drainage and sweep regions, well-pair regions, etc. The F-Φ diagram shows how y percent of the flow can be attributed to x percent of the flow volume. The Lorenz coefficient, which is twice the area between this curve and the straight line $y = x$, correlates well with oil recovery in waterflooding. The lower plots show well connections and flux allocation for one injector in a field model.

to compute gradients and parameter sensitivities, which in turn can be utilized in effective optimization methods. In previous research [22], we have used this idea to develop efficient workflows for optimizing well placement, drilling sequence, and production rates. We have also shown how effective proxies for economic objectives like net-present value can be derived from time-of-flight and tracer partitions, and how these in turn can be used to formulate highly efficient optimization loops for suggesting plausible sequences of rate targets, which subsequently can be slightly adjusted by a full-fledged simulation to derive production schedules that fulfill multiphase well constraints. Often, it is more difficult to formulate the objective and economic and engineering constraints in a precise mathematical form than solving the resulting problem. Exploring a large number of alternative formulations is usually prohibitive when relying on full-fledged multiphase simulators. Various forms of flow diagnostics, on the other hand, are inexpensive to compute and therefore ideal in the exploratory part of an optimization workflow.

Using time-of-flight and tracer distributions to generate flow-based proxies for accelerating reservoir management workflows is not a new idea. Diagnostic tools formulated on top of streamline simulation have been applied in ranking and up-

scaling of geostatistical models [11, 1], to optimize well rates in waterflooding [36, 26, 13, 37], for flood surveillance on a pattern-by-pattern basis [3], and to optimize fracture stages and well completions in tight gas reservoirs [31].

Herein, we will discuss to what extent flow-diagnostic ideas developed for waterflooding scenarios can be extended to polymer flooding. To this end, we first discuss alternative ways of computing the distribution of time-of-flight and residence times (i.e., the time a neutral particle spends traveling from an inflow to an outflow point) that utilize ideas from tracer modelling [33, 10]. Then, we move on to discuss how to forecast the macroscopic effect of polymer flooding and provide inexpensive forecasts of hydrocarbon recovery. Viscosity change due to polymer flooding improves both the microscopic and macroscopic sweep efficiency [34, 17]. Polymers increase the viscosity of the displacing fluid and hence increase the fractional flow of oil to the flow of the displacing fluid, which in turn improves the microscopic sweep efficiency [27]. This effect is most pronounced when the waterflooding has an unfavorable mobility ratio.

Polymers also improve the macroscopic sweep by reducing channeling through heterogeneous reservoirs and through viscous cross-flow between layers of different permeability [7]. We investigate polymer efficiency by comparing polymer flooding simulations to corresponding waterflooding scenarios. As numerical examples we apply both single layers from SPE 10 Model 2 [6] and the more complex Norne field model [12]. By comparing an explicit proxy that only accounts for the improved microscopic sweep along streamlines to an implicit calculation that also accounts for macroscopic effects, we can distinguish the microscopic and macroscopic polymer effects. As the macroscopic effects are linked to viscous cross-flow and conformance, they are expected to correlate with heterogeneity measures [38]. Correlation with the Lorenz coefficient and the vorticity index [28] is explored for the models under consideration.

2 Time-of-flight and distributions of residence time

Time-of-flight can essentially be computed in three different ways for an instantaneous flow field $\mathbf{v}$. The most obvious approach is to trace streamlines and compute time-of-flight τ in a *pointwise sense* by integrating the interstitial velocity field along these streamlines [8]

$$\frac{d\mathbf{x}}{ds} = \mathbf{v}(\mathbf{x}), \qquad \frac{d\tau}{ds} = \frac{\phi(\mathbf{x})}{|\mathbf{v}(\mathbf{x})|}, \tag{3}$$

where s denotes curve length along individual streamlines. Using streamlines to compute τ gives high pointwise accuracy. Unfortunately, it is not always straightforward to trace streamlines in complex reservoir grids having polyhedral cell geometries and all sorts of challenging degeneracies. In particular, it is challenging to reconstruct a consistent velocity field $\mathbf{v}$ from the numerical fluxes that are typically available from a finite-volume reservoir simulator, associate the correct flux to each flow path, etc. There are general and versatile methods available, see e.g., [15], but these are relatively expensive for large and complex geological models. Likewise, there are problems associated with distributing well fluxes to streamlines and ensuring mass conservation, see e.g., [14].

Alternatively, one can use a finite-volume discretization of (1), which approximates the volume-averaged value of τ in each grid cell. Assuming incompressible flow, (1) can be written $\nabla \cdot (\tau \mathbf{v}) = \phi$. Integrating this equation over a single cell C_j and using the divergence theorem gives us

$$\int_{\partial C_j} \tau \mathbf{v} \cdot \mathbf{n} \, ds = \int_{C_j} \phi \, dx,$$

where $\mathbf{n}$ is the normal vector to the cell faces. Using the same finite-volume method as for multiphase flow with upwind evaluation of fluxes (i.e., the single-point upwind method in [32]), we can write the flux over the face Γ_{jk} between cells C_j and C_k as

$$\int_{\Gamma_{jk}} \tau \mathbf{v} \cdot \mathbf{n} \, ds = \begin{cases} v_{jk}\tau_j, & \text{if } v_{jk} \geq 0, \\ v_{jk}\tau_k, & \text{otherwise,} \end{cases} \qquad v_{jk} = |\Gamma_{jk}|(\mathbf{v} \cdot \mathbf{n})|_{\Gamma_{jk}}.$$

In vector notation[1] , this discretization reads $\boldsymbol{A}\boldsymbol{\tau} = \boldsymbol{V}_{\boldsymbol{\phi}}$, where $\boldsymbol{A}$ is the flux matrix, $\boldsymbol{\tau}$ is the vector of unknown TOF values, and $\boldsymbol{V}_{\boldsymbol{\phi}}$ is the vector of pore volumes per cell. The discretization of the tracer equation is similar. This discretization preserves the *causality* of the underlying continuous equation (1) (all information follows streamlines), which in turn ensures that the resulting linear system can be permuted to (block) triangular form by performing a topological sort of the grid cells. Hence, (1) can be solved very memory-efficiently in $\mathcal{O}(n)$ operations for a grid with n cells, see [24, 25, 22]. This solution procedure is also possible if one uses a higher-order discontinuous Galerkin discretization.

To shed more light into the finite-volume approach and its potential limitations, let us consider a discrete incompressible flux field $\boldsymbol{v}$ and a grid cell j with total influx v_j. Let $c_{(j)}$ denote the vector of *backward* tracer concentrations corresponding to an imaginary experiment in which a tracer is injected in cell i and allowed to flow in the reverse direction of $\boldsymbol{v}$. Moreover, let

$$\boldsymbol{A}\boldsymbol{\tau} = \boldsymbol{V}_{\boldsymbol{\phi}} \quad \text{and} \quad \boldsymbol{A}^T \boldsymbol{c}_{(j)} = \boldsymbol{e}_j v_j \tag{4}$$

be the discrete TOF equation and the *backward* tracer-equation, respectively. Here, $\boldsymbol{e}_j$ is a unit vector equal one in cell j and zero elsewhere. For the TOF-value τ_j of cell j, we then have the following:

$$\tau_j = \boldsymbol{e}_j^T \boldsymbol{A}^{-1} \boldsymbol{V}_{\boldsymbol{\phi}} = \frac{1}{v_j} \boldsymbol{c}_{(j)}^T \boldsymbol{V}_{\boldsymbol{\phi}}. \tag{5}$$

Accordingly, τ_j equals the pore volume of the upstream region of cell j (i.e., the drainage region) divided by the flux. For a highly heterogeneous drainage region, this means that τ_j will be the average of a distribution of potentially large variance. This averaging introduces a systematic bias in dynamic heterogeneity measures, which may be acceptable in some applications and can be somewhat reduced by a higher-order spatial discretization [30].

Despite this bias, dynamic heterogeneity measures like the Lorenz coefficient computed from the average residence time defined in each grid cell (more details will be given below), have previously shown to correlate well with secondary oil

[1] Henceforth, we will use the notation that a bold italic symbol $\boldsymbol{v}$ denotes a discrete quantity, whereas a bold upright symbol $\mathbf{v}$ denotes a continuous vector in physical space.

recovery by waterflooding [13, 22]. However, as we will see later (e.g., in Figure 9), these simple measures do not provide satisfactory correlation with tertiary recovery by polymer flooding. In the next section we therefore develop a simplified physics proxy that maps one-dimensional displacement profiles onto time-of-flight. For this purpose it is not sufficient to know the average TOF values. Instead, we need to know the *distribution* of breakthrough times for all flow paths. Hence, to provide more accurate production forecasts, we consider the distribution of τ for each grid cell and in particular for cells containing production wells. At an outflow boundary, τ equals the *residence time*, i.e., the total time a neutral particle has spent traveling from the inflow to the outlet.

Let $\mathbf{v}$ be an incompressible flux field in a 3D domain Ω with $\nabla\cdot\mathbf{v} = 0$ inside the domain, $\mathbf{v}\cdot\mathbf{n} = q_i$ on the inlet boundary Γ_i and $\mathbf{v}\cdot\mathbf{n} = q_o$ on the outlet boundary Γ_o, and $\mathbf{v}\cdot\mathbf{n} = 0$ elsewhere on $\partial\Omega$. Consider the linear transport equation

$$\phi\frac{\partial c}{\partial t} + \mathbf{v}\cdot\nabla c = 0, \quad c|_{\Gamma_i} = \delta(t), \tag{6}$$

with $c(\mathbf{x}, 0) = 0$. Thus, (6) describes the transport of a unit pulse through Ω. For each point $\mathbf{x}$, the TOF-distribution $p(\cdot;\mathbf{x})$ is simply the Dirac function

$$p(t;\mathbf{x}) = c(\mathbf{x},t) = \delta\big(t - \tau(\mathbf{x})\big), \tag{7}$$

while at the outlet Γ_o, the TOF/residence-time distribution is given as

$$p_o(t) = \frac{1}{F_o}\int_{\Gamma_o} c\,\mathbf{v}\cdot\mathbf{n}\,ds, \qquad F_o = \int_{\Gamma_o}\mathbf{v}\cdot\mathbf{n}\,ds. \tag{8}$$

It follows from the definition of the Dirac distribution that $\int p_o(t)\,dt = 1$. Moreover, for the mean $\bar{t}$ of the distribution we have

$$\begin{aligned}\bar{t} &= \int_0^\infty \frac{t}{F_o}\int_{\Gamma_o} c\mathbf{v}\cdot\mathbf{n}\,ds\,dt \\ &= \frac{1}{F_o}\int_{\Gamma_o}\int_0^\infty t\delta(t-\tau)\mathbf{v}\cdot\mathbf{n}\,dt\,ds \\ &= \frac{1}{F_o}\int_{\Gamma_o}\tau\mathbf{v}\cdot\mathbf{n}\,ds = \frac{1}{F_o}\int_\Omega \mathbf{v}\cdot\nabla\tau\,d\mathbf{x} = \frac{1}{F_o}\int_\Omega \phi\,d\mathbf{x}.\end{aligned} \tag{9}$$

Accordingly, the mean of $p_o(t)$ is given by $\bar{t} = \Phi_t/F_o$, where Φ_t is the total pore volume.

To develop discretized equations for the TOF/residence-time distribution, we write the semi-discrete version of the pulse-equation (6) as a linear set of ODEs of the form

$$\frac{d\boldsymbol{c}}{dt} + \boldsymbol{M}\boldsymbol{c} = 0, \quad \boldsymbol{c}(0) = \boldsymbol{c}_0 = \frac{\boldsymbol{q}_i}{\boldsymbol{V}_\phi}, \tag{10}$$

where $\boldsymbol{M} = \boldsymbol{V}_\phi^{-1}\boldsymbol{A}$ is the discretization of the linear operator $\frac{1}{\phi}\mathbf{v}\cdot\nabla$ and $\boldsymbol{q}_i$ is the vector of injection source terms. The discrete linear operator $\boldsymbol{M}$ is constructed using the standard upwind scheme introduced for $\boldsymbol{A}$ above. The solution of (10) is given in terms of matrix exponentials by $\boldsymbol{c}(t) = e^{-t\boldsymbol{M}}\boldsymbol{c}_0$. Hence, the discrete counterparts of (7)–(8) can be represented by

$$p_j(t) = \boldsymbol{e}_j^T e^{-t\boldsymbol{M}}\boldsymbol{c}_0 \quad \text{and} \quad p_o(t) = \boldsymbol{q}_o^T e^{-t\boldsymbol{M}}\boldsymbol{c}_0/\boldsymbol{q}_o^T\boldsymbol{e}, \tag{11}$$

where p_j is the distribution in cell number j, p_o is the distribution in the producers (fluid sinks) and $\boldsymbol{q}_o$ the corresponding vector of source terms, and $\boldsymbol{e}$ is a vector of ones. Given the distribution p_o at the outlet, we can define flow capacity and storage capacity curves as [33]

$$F(t) = \int_0^t p_o(s)\, ds, \qquad \Phi(t) = \frac{F_o}{\Phi_t} \int_0^t s\, p_o(s)\, ds, \tag{12}$$

where Φ_t is the total pore volume of the reservoir volume that is drained by the outflow boundary Γ_o and F_o is the corresponding total outflux. Notice that both quantities are normalized so that $F(\infty) = \Phi(\infty) = 1$. From this definition, it also follows that the mean value of $p_o(t)$ corresponds to the time $t = \Phi_t/F_o$ it takes to inject one pore volume, which we will later refer to as 1 PVI.

For efficient computation of residence-time distributions we employ a rational Padé approximation to evaluate the action of the matrix exponential. By collecting all the p_j's in a vector $\boldsymbol{p}$, the first equation in (11) can be reformulated and approximated as follows

$$\boldsymbol{p}(t + \Delta t) = e^{-\Delta t \boldsymbol{M}} \boldsymbol{p}(t) \approx P(-\Delta t \boldsymbol{M})\, Q(-\Delta t \boldsymbol{M})^{-1} \boldsymbol{p}(t), \tag{13}$$

for suitable polynomials[2] P and Q. Herein, we use first-order polynomials to reduce fill-in, i.e., $P(x) = 1 + x/2$ and $Q(x) = 1 - x/2$. Accordingly, for each successive value of the distribution we compute, we need to solve a linear system. However, for the problems we consider, the matrix $\boldsymbol{M}$ is triangular possibly after permutation [25], and hence each linear solve is highly efficient. The approximation (13) obviously depends on the choice of Δt. For the cases considered here, we found (heuristically) that splitting the time-interval of interest into 200 uniform steps, gave sufficient accuracy for the approximation.

The upper plots in Figure 2 show $p_o(t)$ as function of dimensionless time (PVI) for two different permeability fields. The solid lines are distribution computed numerically by (13), i.e., by tracing a unit pulse through the model. For comparison, we also include estimates of the same distributions obtained by first solving the forward and backward TOF equations, $\boldsymbol{A}^{\pm}\boldsymbol{\tau}^{\pm} = \boldsymbol{V}_{\boldsymbol{\phi}}$, to obtain the total travel time $\tau_r = \tau^+ + \tau^-$, and then use the relationship $F_j = V_{\phi,j}/\tau_{rj}$ to back out the flux F_j associated with cell j. In principle, the residence-time distributions is now obtained by sorting the τ_{rj} values in ascending order and plotting F_j against τ_{rj} normalized by 1 PVI. The resulting plots are highly irregular, and in Figure 2 we have therefore binned the τ_r values and instead plot the total flux associated with each bin, shown as dashed lines. It is clear, especially from the most heterogeneous case, that the averaging in the TOF-equation introduces a delay in e.g., breakthrough-time. By construction, the mean of the distributions equals 1 PVI, shown as red dashed lines in the figure. This may not be apparent from the plots since particularly the channelized case has a very long tail. The lower plots in the figure show the resulting F-Φ diagrams and report the Lorenz coefficient, defined as twice the area between the curve $F(\Phi)$ and the straight line $F = \Phi$. This coefficient is a measure of dynamic heterogeneity and has previously been shown to correlate well with recovery for waterflooding [13, 29, 22]. Even though there are large differences in the residence-time distributions computed by the two methods, the F-Φ diagrams and Lorenz coefficients are not very different.

[2] We note that $P = 1 + x$ and $Q = 1$ gives forward Euler for *linear* equations, whereas $P = 1$ and $Q = 1 - x$ gives backward Euler.

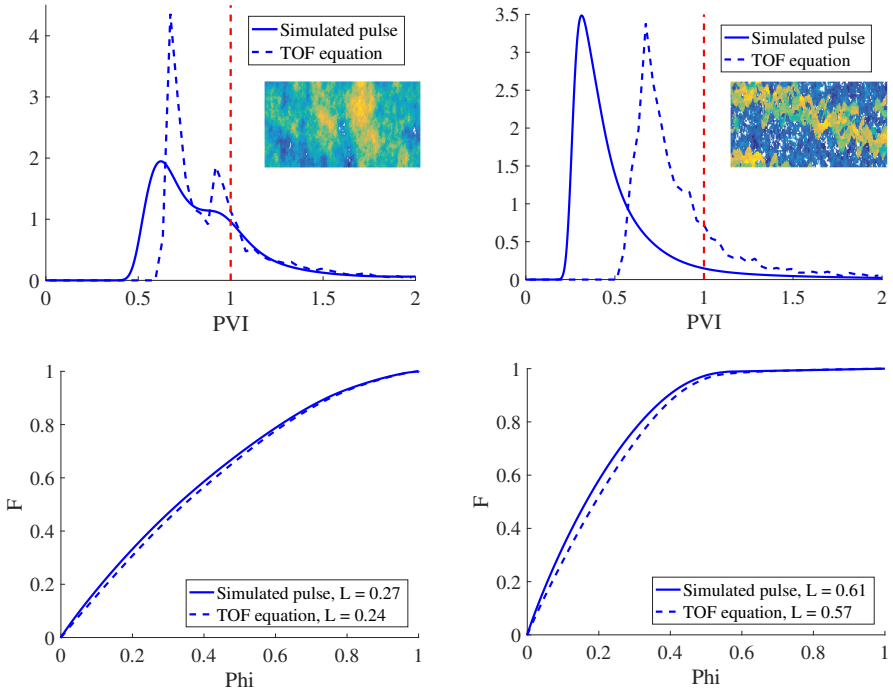

Fig. 2: The upper plots report residence-time distributions (8) for two different permeability fields in a left-to-right displacement scenario. Solid lines are obtained by tracing a unit pulse through the reservoir to determine the distribution at the outlet, whereas dashed lines are obtained by solving the time-of-flight equation by a finite-volume method and backing out data for representative flow paths through each of the cells of the model. The lower plots show comparisons of the corresponding flow F-Φ diagrams and Lorenz coefficients.

Distributions of residence times are used e.g., in the study of chemical reactors and tracer tests [33, 10]. We end the section by going through some derivations that hopefully contribute to tie connections for those familiar with analysis of tracer tests. To see the connection between (1) and (6), we consider the first-order moment $m_1 = \int_0^\infty tc\ dt$, which can be obtained by multiplying (6) with t and taking the integral

$$\int_0^\infty \Big[\phi\frac{\partial c}{\partial t}t + \mathbf{v}\cdot\nabla(tc)\Big]\ dt = \phi\big([tc]_{t=0}^{\infty} - m_0\big) + \mathbf{v}\cdot\nabla\left(m_1\right) = 0. \tag{14}$$

This equation simplifies to

$$\mathbf{v}\cdot\nabla m_1 = \phi, \quad m_1|_{\Gamma_i} = 0, \tag{15}$$

since $m_0 = \int_0^\infty c\,dt = 1$ and $\lim_{t\to\infty} c(t) = 0$. Accordingly, m_1 equals τ as defined by (1). Equation (15) is the first of a family of moment equations [18], for which the higher-order (raw) moments can be computed according to

$$\mathbf{v}\cdot\nabla m_k = k\ \phi\ m_{k-1}, \quad m_k|_{\Gamma_i} = 0. \tag{16}$$

Note that by (7), for any point $\mathbf{x}$, $m_k(\mathbf{x}) = 0$ for $k \geq 2$, while this is not the case for residence-time distributions of the form (8).

Analogous to (16), the moments $m_{o,k}$ of $p_o(t)$ for $k \geq 1$ can be obtained by

$$m_{o,k} = \frac{\boldsymbol{q}_p^T \boldsymbol{m}_k}{\boldsymbol{q}_p^T \boldsymbol{e}}, \quad \boldsymbol{M}\boldsymbol{m}_k = k\,\boldsymbol{m}_{k-1}, \tag{17}$$

with $\boldsymbol{m}_0 = \boldsymbol{e}$. We note that an alternative approach to using matrix exponentials is to solve the *truncated moment problem*, i.e., to compute the first n moments of the distribution from (16), and then try to find a distribution sharing the same moments. One approach towards this is the *maximum entropy method* (see e.g., [20]), which involves solving a set of n non-linear equations. In our initial tests, however, we found that obtaining convergence for these equations could be difficult, especially for distributions with long, slim tails towards infinity. This is typically the case for residence-time distributions from highly heterogeneous permeability fields like the one shown to the right in Figure 2.

3 A recovery proxy for polymer flooding

In the following, we will use the residence-time distribution to develop a proxy for evaluating the performance of polymer flooding. The word 'proxy' is often used to denote response surface models derived from a series of full flow simulations. Herein, we will use the same word to denote a reduced model with simplified flow physics that can approximate recovery curves.

To describe polymer flooding, we consider an immiscible, two-phase model with three fluid components (oil, water, and polymer) on the form,

$$\begin{aligned} \partial_t(\phi b_\alpha s_\alpha) + \nabla \cdot (b_\alpha \mathbf{v}_\alpha) - b_\alpha q_\alpha = 0, \quad \alpha = o, w \\ \mathbf{v}_\alpha = -\lambda_\alpha K(\nabla p_\alpha - \rho_\alpha g \nabla z), \\ \partial_t(\phi_{dpv} b_w s_w c_p) + \partial_t(\rho_r c_a (1-\phi_r)) + \nabla \cdot (b_w c_p v_p) - b_w q_p = 0, \\ \mathbf{v}_p = -\lambda_p K(\nabla p_w - \rho_w g \nabla z). \end{aligned} \tag{18}$$

This model is sufficiently general to incorporate most of the fluid effects found in commercial simulators, like adsorption of polymer onto the reservoir rock, reduction in permeability, inaccessible pore space, mixing of polymer in water, compressibility of fluids and rock, as well as pseudoplastic effects of the diluted polymer solution. As our multiphase reference, we will use an open-source simulator [2] that includes all these effects.

3.1 Capturing macroscopic sweep effects in a single step

The first goal of the current flow-diagnostics approach is to efficiently obtain a flux field that takes into account changing mobility effects originating from injection of polymer. To simplify our discussion, we omit adsorption, dead pore space, and pseudoplastic effects. In addition, the proxy will neglect gravity and compressibility

for efficiency (our reference simulations does not). Equation (18) can then be written in total flux form:

$$\begin{aligned} \mathbf{v} &= -[\lambda_w(s,c_p)+\lambda_o(s)]\mathsf{K}\nabla p, \quad \nabla\mathbf{v} = q, \\ \partial_t(\phi s_w) + \nabla\cdot(\mathbf{v}f_w) = q_w, &\qquad \partial_t(\phi s_w c_p) + \nabla\cdot(\mathbf{v}f_p c_p) = q_w c_{p,\mathrm{inj}}, \end{aligned} \tag{19}$$

where $\mathbf{v} = \mathbf{v}_w + \mathbf{v}_o$ is the total flow rate of both phases and we have introduced the fractional flow functions $f_w = \lambda_w/(\lambda_w+\lambda_o)$ and $f_p = \lambda_p/(\lambda_w+\lambda_o)$. (For the model used herin, $f_p(s,c) = m(c)f_w(s,c)$, where $m(c) = \lambda_p/\lambda_w$.)

Although the equations in (19) are greatly simplified compared to (18), they are still highly nonlinear, and obtaining a flux field $\mathbf{v}$ at some finite end time T requires a simulation. To reduce the computational cost of the proxy, we will try to perform this simulation as efficient as possible. Assuming constant well controls and injection compositions, we use a single *implicit* time step $\Delta t = T$. To enable this computation for large Δt, we linearize f_w and f_p between their endpoints. The fully-coupled system is still nonlinear, but using linear flux functions improves the convergence of the nonlinear Newton solver. As an alternative or complement to linearization of the flux functions, one could use a trust-region solver [21], which recently has been extended to include all the pertinent flow physics in (18), see [16]. Taking extremely long time steps like this will obviously lead to severe smearing of saturation and concentration fronts, and hence the solution cannot be used to predict fluid production in wells. However, the sole purpose of the computation is to obtain *representative* flux fields that account for how polymer injection and/or other changes in the injection setup affect the time-of-flight and tracer distributions in the reservoir.

To illustrate, Figure 3 depicts how the instantaneous residence-time distributions vary throughout a polymer injection scenario following an initial waterflooding phase for the two permeability fields in Figure 2. Residence-time curves are specific to instances in time, and curves are obtained at different times by extracting instantaneous velocity fields and then using each such field to trace a unit pulse through the whole domain from inlet to outlet. The solid blue curves represent the velocity field at the end of waterflooding, whereas the gray curves represent instantaneous velocity fields from times equally spaced throughout the polymer injection period. As can be observed, the residence-time distributions can vary substantially during the polymer injection period. However, if we average all the instantaneous velocity fields, the residence-time distribution (solid red lines) associated with this averaged flow field seems to be well matched by the corresponding distribution (dashed red lines) computed for the velocity field used in our one-step proxy. In the next step of the proxy, we reevaluate the fluid distribution based on the residence-time distributions for the flux field $\mathbf{v}$ obtained from (19).

3.2 Mapping 1D displacement fronts to residence-time distributions

To account more accurately for the fluid transport, we compute numerical tracers for all wells and use these to partition the flux field for each simulation period into injector–producer interaction regions, and solve representative 1D transport problems along τ for each region. The interaction regions are obtained by solving (forward and backward) stationary tracer equations, see [22] for details. Let $c(\mathbf{x},t)$ be the solution of the delta-pulse equation (6), and $s(\mathbf{x},t)$ a saturation field in Ω

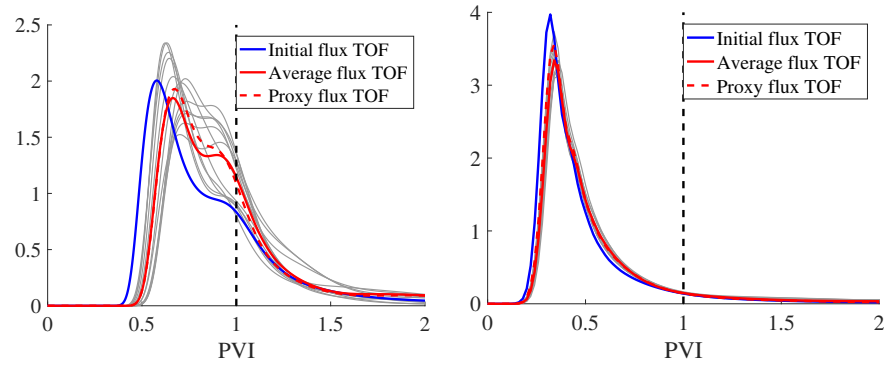

Fig. 3: Residence-time distributions for the two permeability fields of Figure 2. Blue line is initial distribution (prior to polymer injection), gray lines show distributions at selected times during the injection period, red line is the distribution for the average flux field over the period, and finally, the red dashed line shows the distribution obtained from the single-step proxy.

(or any other time-dependent field on Ω). Then, the corresponding 1D field $s(\tau, t)$ along τ can be computed by

$$s(\tau, t) = \int_{\Omega} s(\mathbf{x}, t) c(\mathbf{x}, \tau)\, d\mathbf{x}. \tag{20}$$

Correspondingly, a 1D field $s(\tau, t)$ is mapped to $s(\mathbf{x}, t)$ by

$$s(\mathbf{x}, t) = \int_{0}^{\infty} s(\tau, t) c(\mathbf{x}, \tau)\, d\tau. \tag{21}$$

For a true delta pulse (exact solution of Eq. (6)), the composition of the mappings (21)–(20) equals identity, while the opposite composition equals identity only if the field $s(\mathbf{x}, t)$ is aligned with $\tau(\mathbf{x})$, i.e., constant along the time-of-flight contours. For the discrete case, however, compositions (in either direction) will only approximate identity since the delta pulses are approximated by smooth functions.

Disregarding compressibility, the recovered oil $r(t)$ from time t_0 to time t, can be estimated by

$$r_o(t) = \int_{\Omega} \phi \Delta s(\mathbf{x}, t)\, d\mathbf{x}, \tag{22}$$

where $\Delta s(\mathbf{x}, t) = s(\mathbf{x}, t) - s(\mathbf{x}, t_0)$ is the pointwise saturation change. The integral (22) can be transformed to an integral in τ (i.e., omitting the mapping (21)) as

follows

$$
\begin{aligned}
r_o(t) &= \int_\Omega \phi(\mathbf{x}) \Delta s(\mathbf{x}, t)\, d\mathbf{x} \\
&= \int_\Omega \phi(\mathbf{x}) \int_0^\infty \Delta s(\tau, t) c(\mathbf{x}, \tau)\, d\tau\, d\mathbf{x} && (23)\\
&= -\int_\Omega \phi(\mathbf{x}) \int_0^\infty \left(\int_0^\tau \Delta s(\tilde{\tau}, t)\, d\tilde{\tau} \right) \partial_\tau c(\mathbf{x}, \tau) d\tau\, d\mathbf{x} && (24)\\
&= -\int_\Omega \int_0^\infty S(\tau, t) \phi(\mathbf{x}) \partial_\tau c(\mathbf{x}, \tau) d\tau\, d\mathbf{x} && (25)\\
&= \int_0^\infty S(\tau, t) \int_\Omega \mathbf{v} \cdot \nabla c(\mathbf{x}, \tau)\, d\mathbf{x}\, d\tau && (26)\\
&= \int_0^\infty S(\tau, t) \int_{\Gamma_o} c(\mathbf{x}, \tau) \mathbf{v} \cdot \mathbf{n}\, d\Gamma_o\, d\tau \\
&= q_o \int_0^\infty S(\tau, t) p_o(\tau)\, d\tau. && (27)
\end{aligned}
$$

In the above derivation, (24) follows from (23) by partial integration in τ and that $c(\mathbf{x}, 0) = \lim_{\tau \to \infty} c(\mathbf{x}, \tau) = 0$, and $S(\tau, t)$ denotes the integral function of $\Delta s(\tau, t)$ such that $\partial_\tau S = \Delta s$. Moreover, (26) follows from (25) by equation (6), and finally q_o is the total production rate and $p_o(\tau)$ the residence-time distribution as defined in (8). Accordingly, the recovery can be estimated solely by considering the TOF distribution at producers and the (integral of the) 1D solution profile along τ.

To sum up, a single step of the suggested proxy proceeds as follows. First, by solving a series of normalized tracer equations (2) using the representative velocity field, we split the reservoir into a set of well-pair regions with associated total fluxes $q_i = q_o$. For each well-pair region, we perform the following three steps:

1. Compute the TOF/residence-time distribution for the region.
2. Map the saturation/concentration fields of the region onto a 1D TOF-grid using (20), i.e., $s(\mathbf{x}, 0) \mapsto s(\tau, 0)$ and $c_p(\mathbf{x}, 0) \mapsto c_p(\tau, 0)$. Run a 1D simulation from time zero to time T. As a result, we get the saturation changes as a function of τ, i.e., $\Delta s(\tau, T) = s(\tau, T) - s(\tau, 0)$.
3. Estimate the total volume of produced oil for the region by (27). That is, the produced volume of oil r_o from the beginning of the period (time 0) to the end of the period (time T) for the region is estimated as

$$
r_o = q_o \int_0^\infty S(\tau, T) p_o(\tau)\, d\tau, \tag{28}
$$

   where q_o is the total production rate and $S(\tau, T) = \int_0^\tau \Delta s(\tilde{\tau}, T)\, d\tilde{\tau}$. This oil production is computed for each well-pair region and summed up to give total field production.

The overall procedure for evolving saturation and concentration during a single simulation period is illustrated in Figure 4 and has obvious similarities with streamline simulation (think of each region as a bundle of streamlines). If necessary, the proxy can be refined by computing the volumetric partition based on well segments instead of individual wells.

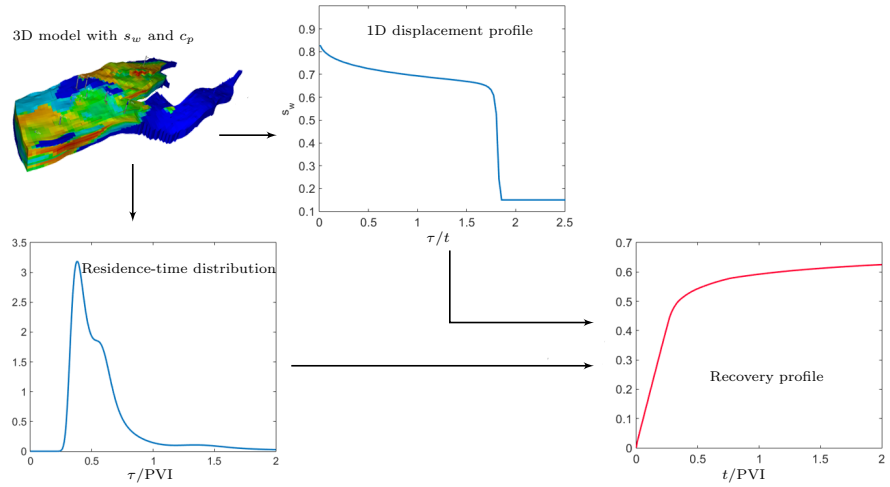

Fig. 4: Illustration of the recovery proxy. The reservoir is partitioned into injector–producer regions. Then, residence-time distributions and representative 1D displacement profiles are computed for each region and convolved to compute the recovery proxy.

3.3 Overall procedure

The overall procedure for computing the proxy is summarized in Figure 5. For efficiency, our default setup is to use a single time step to compute the flux field from (19). In the numerical examples presented in the next section, the proxies are applied only to cases with continuous water injection or continuous polymer injection. To handle cases in which the well controls and injection compositions vary significantly throughout the simulation period (injection of polymer slugs, injection of chase water, etc), multiple time-steps must be considered. In this situation, one can divide the simulation history into multiple periods, and approximate each period with the proxy. Between periods, the one-dimensional displacement profiles must then be mapped back to the physical grid by (21) as illustrated by the red box in Figure 5. This is analogous to streamline methods. However, our proxy is a more crude appxoimation and cannot generally be expected to provide the same spatial accuracy as a multiphase streamline simulation. Restarting from inaccurate saturation/concentration fields will eventually affect the accuracy of the linearized multiphase simulation used to compute a representative velocity field for the next proxy stage. Hence, accuracy is generally expected to decay somewhat when the proxy is applied to simulate multiple injection periods. Likewise, the computational cost will increase since one would need to recompute representative velocity fields and the residence-time distributions for each simulation period.

To discriminate macroscopic displacement effects, we introduce a second proxy that follows the same steps outlined above, but computes each flow field *explicitly*, i.e., using mobility resulting from the fluid distribution at the outset of each simulation period. This proxy does not account for the fact that injected fluids will

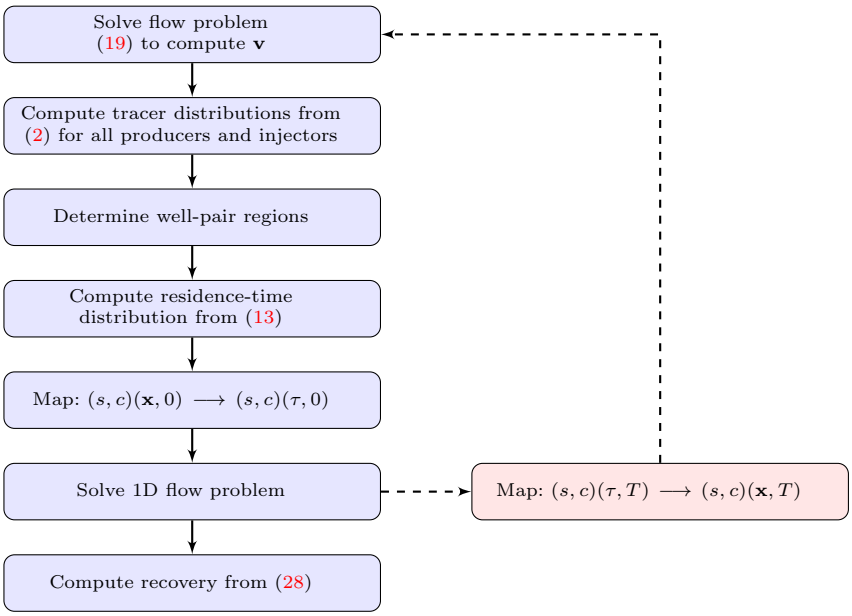

Fig. 5: Flow-chart for the proxy computation. The blue boxes represent key steps in the method. The red box is an optional step the could be included to make our proxy behave more like a streamline method.

affect the flow paths during the simulation period, but only accounts for changes in displacement efficiency along each flow path through the 1D simulations.

In our explanation of the proxy method, we neglected various polymer effects to make the presentation as brief as possible. All the effects of the underlying multiphase model can in principle be included in the proxy, and most of them are implemented in our prototype code.

4 Numerical examples

To validate the practical usefulness of flow diagnostics for EOR, the methods introduced above were implemented as an enhancement to the `diagnostics` module from the open-source Matlab Reservoir Simulation Toolbox (MRST) [23, 19]. The multiphase reference simulations reported in the following were conducted with the `ad-eor` module of MRST [2], and include the effects of dead pore space, gravity, and fluid compressibility, but not pseudoplasticity. The explicit and implicit proxies include dead pore space, but neglect gravity and compressibility.

4.1 Horizontal layers from SPE 10

In our first numerical example, we consider the horizontal layers of the synthetic Brent model used in the 10th SPE Comparative Solution Project [6]. The full

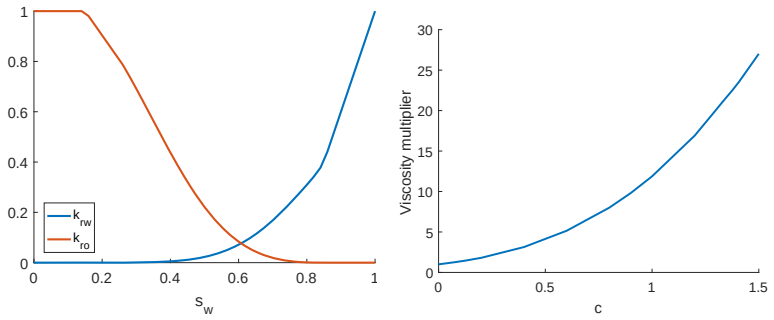

Fig. 6: Relative permeability (left) and water viscosity multiplier as function of polymer concentration (right).

model consists of a grid with $60 \times 220 \times 85$ cells, where the top 35 layers represent the shallow-marine Tarbert formation, which has a log-normal permeability distribution, while the lower 50 layers represent the fluvial Upper Ness formation with distinct permeability distributions for the high-permeable channels and the lower-permeable background. For the experiments, we utilize relative permeabilities and water viscosity multiplier as function of polymer concentration as depicted in Figure 6.

The water and oil viscosities are set to 0.5cp and 1.5cp, respectively, while water, oil and rock compressibilities are set to $c_w = 4.94 \times 10^{-10}\,\text{bar}^{-1}$, $c_o = 6.65 \times 10^{-10}\,\text{bar}^{-1}$ and $c_r = 6.82 \times 10^{-10}\,\text{bar}^{-1}$. We consider a scenario with a single injector and a single producer, each well perforating an entire side of the model (see Figure 7). The injector is controlled by a constant (surface) volume rate (3m^3/day), while the producer is controlled by a constant pressure (150bar). Initially, the fluid mixture is assumed to be at connate water saturation of 0.15 at 270 bar. The reference simulations are run as follows

1. From $t_0 = 0$, inject water only until the water cut in the producer reaches 0.9, which defines a time t_1.
2. From t_1, inject water with a polymer concentration $1\,\text{kg/m}^3$ until a total of 51 000 kg polymer has been injected. For most layers this amounts to approximately 0.8 PVI (t_2). For comparison, we also simulate a scenario in which pure water is injecated also during the period $[t_1, t_2]$.

The middle and right plots in Figure 7 show the reference solution with and without polymer depicted as solid and dashed lines, respectively, for one layer in each of the two different formations. Next, we evaluate the proxy for all layers. For comparison and subsequent approximation of macroscopic versus microscopic sweep improvements, we compute both the implicit proxy, which accounts for changes in flow paths due to changes in mobility, and the explicit proxy, in which the flow pattern will be locked to the current state and not represent changes in streamlines due to mobility changes. This way, the explicit version will only include the effect of improved microscopic sweep due to changes in fractional flow and not reflect improved macroscopic sweep. We note that for evaluation purposes, results from the explicit proxy were compared to results from a sequential simulator which was modified to use fixed velocity for all time steps. This comparison (not reported

here) showed close agreement between the proxy and modified simulation. The proxies are run as follows:

1. For the first period $[0, t_1]$, we evaluate the explicit and implicit proxies using flux fields computed over various time horizons $T <= t_1$.
2. For the second period $[t_1, t_2]$, we start from the state computed by the multiphase simulator and evaluate the explicit and implicit proxies using flux fields computed over various time horizons $T <= t_2 - t_1$ both with and without polymer injection.

Figure 7 shows the proxy predictions for Layers 23 and 75 for the first period (water only) and for the second period (polymer or water only). As observed, there is little difference between explicit and implicit during the first period, which indicates that the injected water causes insignificant changes in the flow paths and residence-time distribution. For the polymer injection in the second period, the two proxies differ substantially for the highly heterogeneous Upper Ness case (Layer 75), but only marginally for the more homogeneous Tarbert case (Layer 23). This indicates that the recovery increase due to polymer for Layer 23 is mainly caused by improvements in microscopic sweep, whereas improvements in macroscopic sweep dominates for Layer 75.

Figure 8 shows the correlation between recovery factors computed by the proxies and reference recovery factors obtained from full simulations for all layers. The upper row shows all proxy evaluations for the waterflooding scenario. For the first period (upper right), a (slight) bias is observed when the flux field is computed with the longer time steps. Note however, that the reservoir is almost completely flooded during this period (water cut from zero to 0.9), so some discrepancies are expected when using the flux field with strong inter-well water communication to compute residence-time distributions. For the second period (upper left), the proxy predictions are very well correlated with the reference. Note also that there are no significant differences between the explicit and implicit proxies for these cases. The lower-left figure compares recovery factors predicted by the implicit proxy and by a full multiphase simulation. Results from the explicit proxy are omitted since they fail to give adequate predictions. Although the implicit proxy correlates well with full multiphase simulations, it has a tendency to overpredict recovery if the flux is computed over a shorter time horizon (prior to polymer breakthrough). This effect becomes more pronounced when we plot the predicted increase in oil recovery due to polymer, i.e., recovery from polymer injection minus recovery from water only (lower-right plot).

Finally, we wish to isolate the macroscopic and microscopic sweep improvements by comparing the implicit and explicit proxies. Since a standard simulator cannot give predictions of these quantities, we rather compare them with measures of heterogeneity, i.e., the Lorenz coefficient and the vorticity index. As above, we estimate the improvement in total recovery by comparing the implicit proxy for polymer injection to the implicit proxy for pure water injection. We estimate the improved recovery due to microscopic effects by comparing the explicit proxy for polymer injection to the implicit proxy for pure water injection. And finally, we estimate the improved recovery due to macroscopic effect as the difference between total improved recovery and recovery improved by microscopic effects. Figure 9 depicts the resulting improved recoveries for all layers. As expected, there is no apparent correlation between heterogeneity and improvements in microscopic sweep.

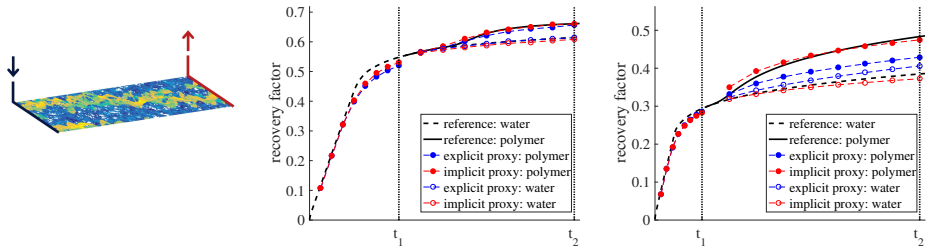

Fig. 7: Well setup for each layer of the SPE10 model (left), and reference and proxy output for Layer 23 (middle) and Layer 75 (right). Dashed black line is reference recovery factor without polymer injection, solid black line is reference recovery for polymer injection starting at t_1 (water cut 0.9). Red dots (polymer injection) and circles (water injection) show proxy predictions for various time horizons using the implicit proxy, while blue dots and circles correspond to explicit proxy.

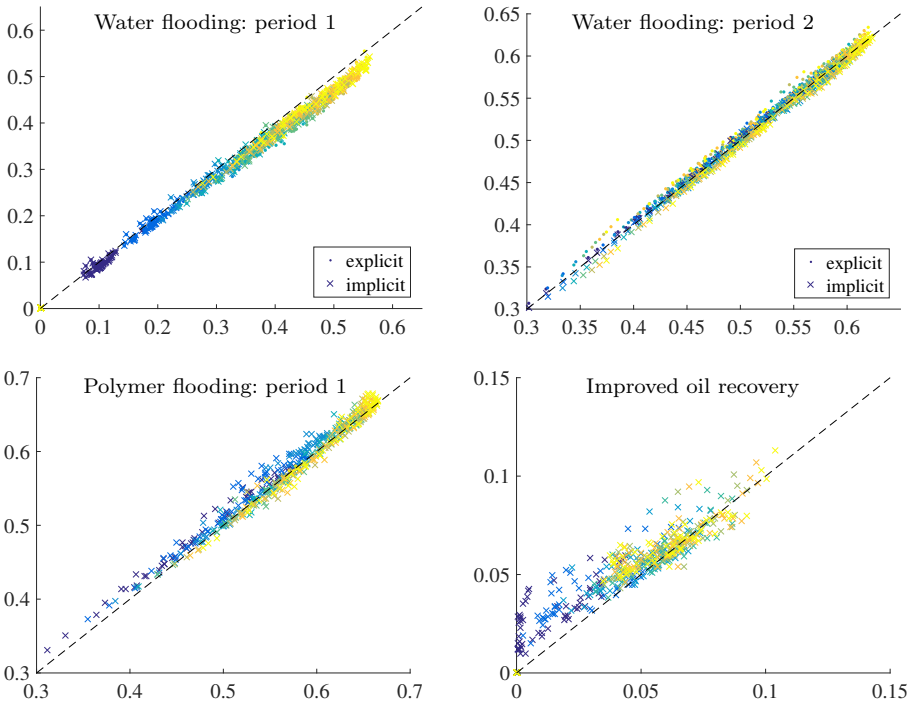

Fig. 8: Correlation between recovery predicted by the proxy (y-axis) and by a full simulation (x-axis) for all horizontal layers of the SPE 10 model. The proxy is run with different end times for the one-step computation of the representative flow field, varying from short (blue) to the whole period (yellow).

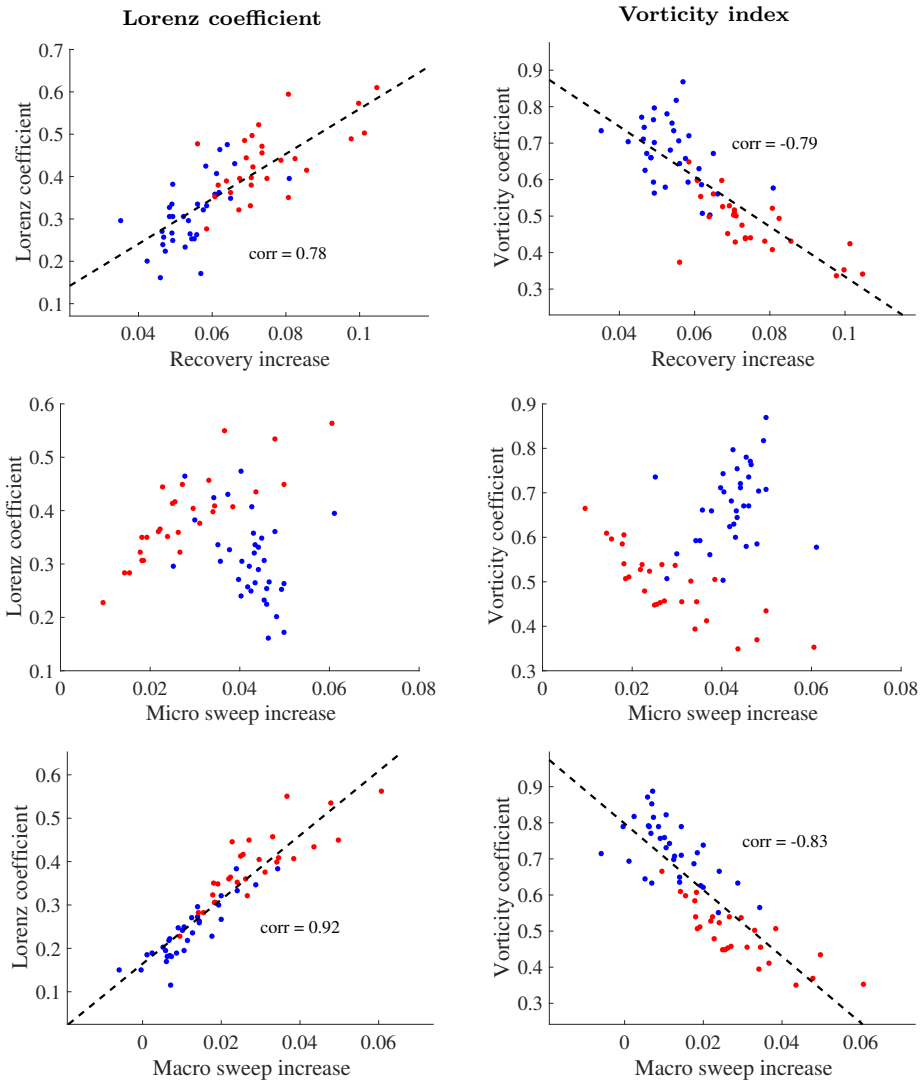

Fig. 9: Plot of heterogeneity measure (Lorenz coefficient or vorticity index) versus estimated increase in total (top), microscopic (middle), and marcroscopic (bottom) recovery due to polymer. Red dots correspond to the fluvial Upper Ness layers, blue to log-normal permeability fields of the Tarbert formation.

The best correlation (for both measures) is observed between heterogeneity and macroscopic improvements. Accordingly, these results illustrate that the largest gain for polymer injection is obtained for high heterogeneity. We note that the line of best fit for macroscopic improvements versus Lorenz coefficient does not pass trough the origin, hence it appears that the relation is not linear even though the two different estimates are strongly correlated.

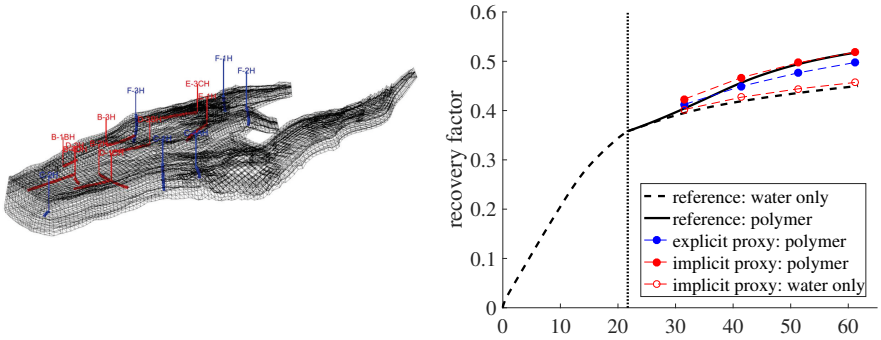

Fig. 10: Grid and well positions for the Norne model (left), and evolution of total field recovery factors from simulations and proxies (right).

In the experiments reported above, we used constant injection rates to make sure that equal amounts of fluids were injected in all layers. The SPE 10 model has very strong heterogeneity and this inevitably caused injection pressures to exceed what realistic ranges in several layers. (As a result, the commercial simulator we used to verify our multiphase simulations in MRST failed to converge for some of the layers.) In a sense, the above experiments can therefore be considered as hard test cases, and we believe that the validy of our approximations would be at least as good for simulations conducted with more realistic simulation setups with the same type of polymer model. Similar considerations apply also in the next example.

4.2 The Norne field model

In this example we adapt a model of the Norne field [12], to test our suggested proxy in a realistic setting with multiple wells (see Figure 10). Again, we consider waterflooding with subsequent polymer injection of a reservoir that initially is filled with oil. Relative permeabilities, viscosity multipliers and PVT-properties are the same as in the previous example. The wells included in the experiment corresponds to those active in the benchmark-case [12] at simulation date 25.04.2006. The injection rates for the experiment are set equal to those in the benchmark at this date which gives a total field injection rate of about 3.7×10^4 m^3/day. Producers are set to operate under constant bhp at 150 bar. Also for this experiment, we assume that the initial fluid mixture is uniform at connate water saturation ($s_w = 0.15$), and initial reservoir pressure about 270bar. Water is injected at constant rate for approximately 20 years, while producers are shut when a water-cut of 0.9 is reached. As a result, at the end of the first period, five of the nine producers have been closed due to excess water-cut. In the second period, all wells are opened, and set to produce at a constant liquid rate for about 40 years (about 0.8 PVI). As in the previous example, we use the proxy to predict both pure water and polymer injection. In the polymer case, all six injectors are set to inject a mixture of concentration 1 kg/m^3. Starting from the multiphase simulation at 20 years, the implicit and explicit proxies are run with time horizons of $10, 20, 30$

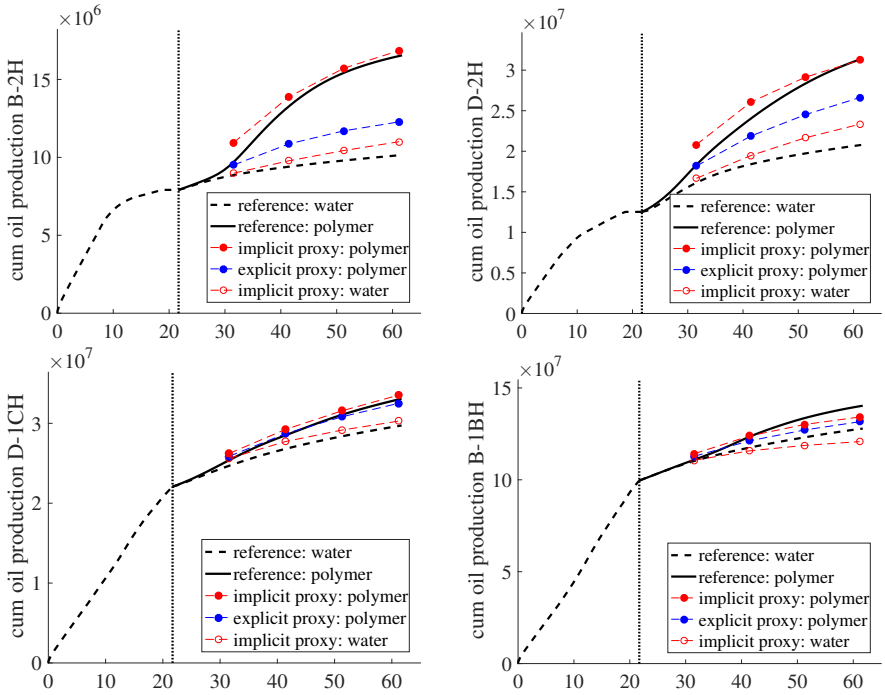

Fig. 11: Estimated evolution of recovery (simulation and proxy) for four of the producers.

and 40 years for the flux calculation for both injection scenarios. With six injectors and nine producers, there are potentially 54 well-pair regions. However, in the current scenarios there are 14 producer-injector pairs with zero or negligible communication, and hence residence-time distributions and 1D displacement profiles are computed for 40 regions for each proxy evaluation. In our current MATLAB implementation, the computation of interaction regions and corresponding residence-time distributions takes about 5 seconds on a standard laptop using the built-in linear solver in MATLAB. We anticipate, however, that employing a tailored solver based on optimal ordering [24] can speed up this process substantially. In lack of an efficient 1D solver, each of the displacement profiles are computed using a fully implicit solver (with large relative overhead for small problems) and hence constitutes the bottleneck of the current implementation. Again, a tailored solver, as used in streamline simulators [8], would eliminate this bottleneck, and as a result, a single proxy evaluation for a model of this size should be on the order of a second in an efficient implementation.

In Figure 10 (right), the evolution of the overall recovery factor is shown. Similarly to the previous example, the proxy for polymer injection slightly overestimates the recovery for the shortest time horizon, but matches perfectly as the polymer mixture reaches most producers. We also observe a considerable difference in explicit versus implicit proxy, suggesting macroscopic sweep improvement due to polymer are present.

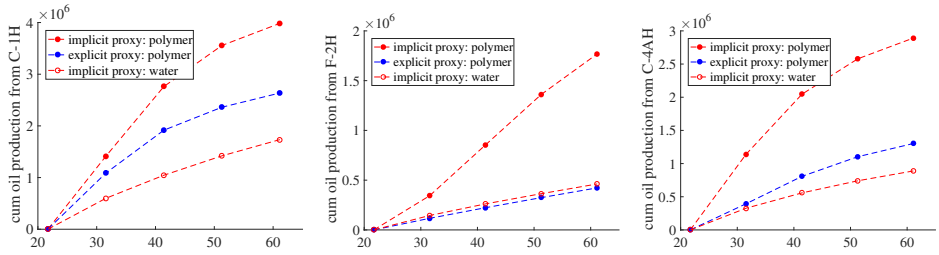

Fig. 12: Estimated evolution of partial recovery for producer B-2H from the three most contributing injectors.

Since the proxy computes recovery for each (communicating) well-pair region, we can also estimate recovery for each of the wells. Figure 11 reports cumulative recovery factors for four of the producers. For wells D-1CH and B-1BH, we observe that the polymer injection has a somewhat marginal effect, while the opposite is true for B-2H and D-2H. Note that the large improvements seen in these plots not only come from improved sweep in the drainage regions, but also from the fact that the drainage regions are enlarged.

Last, we look more closely at the recovery profile of producer B-2H (Figure 11, top-left) and decompose it according to the communicating injectors. The three injectors contributing most to production in B-2H (measured in total flux) are C-1H, F-2H and C-4AH. Recovery plots for the corresponding interaction regions are shown in Figure 12. Since our multiphase simulator cannot provide us with recovery estimates broken down to individual well-pair regions, only the proxy-values for the second period are plotted. We observe that the polymer injection apparently has a large influence on the recovery from these regions.

5 Concluding remarks

Reduced-order and proxy type methods have received much attention as an approach to accelerate forward simulations in optimization and data assimilation workflows. Many such methods – like the trajectory piecewise linearization method [4, 5] – require an initial stage in which a (potentially extensive) set of multiphase simulations is run to tune the proxy model. The proxy can then be expected to provide fast and accurate predictions for simulation input *in the neighborhood* of the tuning run(s). The proxy method presented herein serves a different purpose, as it does not require tuning and hence can be used to quickly screen a number of very different polymer injection scenarios, including new well placements, etc. Technically, our method is somewhat similar to the recent INSIM method [9] in the sense that it relies on one-dimensional simulations within well-pair regions. However, unlike the INSIM method, our method assumes that a geo-cellular model is available and tries to utilize as much information about flow paths from this model as possible. This is a key characteristics of all flow-diagnostics computational tools. Our new proxy also takes inspiration from streamline methods, but utilizes the same grid and finite-volume discretization as a conventional reservoir simulator, and hence accounts for the effects of heterogeneity in the same way.

For the examples considered, the implicit version of the proxy appeared to approximate both water and polymer flooding to a fair degree of accuracy. Thus it appears as a good candidate for optimization applications. In addition, the combination of the explicit and the implicit proxies appeared to adequately differentiate between macroscopic and microscopic sweep improvements. This could be used to asses the impact of reservoir heterogeneity on polymer efficiency.

Apart from the flow model used, there is nothing specific to polymer in our proxy, and we therefore expect that similar ideas can be applied also to other (similar) EOR processes. The main limitation is in the ability to obtain representative flow fields, and if the proxy is applied in multiple stages, in the ability to predict fluid compositions at the end of each stage that are sufficiently accurate to not have a strong adverse effect on the approximate flow field computed for the subsequent step. Likewise, using multiple stages will incur a higher computational costs since we then need to compute additional flow fields and associated residence-time distributions. As discussed herein, our proxy used a unique 1D transport solve for each well-pair region. Higher granularity can easily be added by subdividing the well-pair regions further, e.g., by considering regions associated with subsegments of the well, or even individual well perforations. We have used such subregions in our previous work on flow diagnostics visualization, but have not yet investigated their use together with the proxy for polymer flooding.

6 Acknowledgements

The authors wish to thank Statoil for partial funding of this work and for permission to publish the results. Also, we thank Statoil (operator of the Norne field) and its license partners ENI and Petoro for the release of the Norne data. Further, the authors acknowledge the IO Center at NTNU for coordination of the Norne cases.

References

1. Ates, H., Bahar, A., El-Abd, S., Charfeddine, M., Kelkar, M., Datta-Gupta, A.: Ranking and upscaling of geostatistical reservoir models using streamline simulation: A field case study. SPE Res. Eval. Eng. **8**(1), 22–32 (2005). DOI 10.2118/81497-PA
2. Bao, K., Lie, K.A., Møyner, O., Liu, M.: Fully-implicit simulation of polymer flooding with MRST. Comp. Geosci. (2017). DOI 10.1007/s10596-017-9624-5
3. Batycky, R.P., Thieles, M.R., Baker, R.O., Chugh, S.H.: Revisiting reservoir flood-surveillance methods using streamlines. SPE Res. Eval. Eng. **11**(2), 387–394 (2008). DOI 10.2118/95402-PA
4. Cardoso, M., Durlofsky, L.: Linearized reduced-order models for subsurface flow simulation. J. Comput. Phys. **229**(3), 681–700 (2010). DOI 10.1016/j.jcp.2009.10.004
5. Cardoso, M.A., Durlofsky, L.J.: Use of reduced-order modeling procedures for production optimization. SPE J. **15**(02), 426–435 (2010)

6. Christie, M.A., Blunt, M.J., others: Tenth SPE comparative solution project: A comparison of upscaling techniques. In: SPE Reservoir Simulation Symposium. Society of Petroleum Engineers (2001). DOI 10.2118/66599-MS
7. Clifford, P.J., Sorbie, K.S.: The effects of chemical degradation on polymer flooding. In: SPE Oilfield and Geothermal Chemistry Symposium. Society of Petroleum Engineers (1985). DOI 10.2118/13586-MS
8. Datta-Gupta, A., King, M.J.: Streamline Simulation: Theory and Practice, *SPE Textbook Series*, vol. 11. Society of Petroleum Engineers (2007)
9. Guo, Z., Reynolds, A.C., Zhao, H.: A physics-based data-driven model for history-matching, prediction and characterization of waterflooding performance. In: SPE Reservoir Simulation Conference. Society of Petroleum Engineers (2017). DOI 10.2118/182660-MS
10. Huseby, O., Sagen, J., Dugstad, Ø.: Single well chemical tracer tests – fast and accurate simulations. In: SPE EOR Conference at Oil and Gas West Asia, Muscat, Oman, 16-18 April (2012). DOI 10.2118/155608-MS
11. Idrobo, E.A., Choudhary, M.K., Datta-Gupta, A.: Swept volume calculations and ranking of geostatistical reservoir models using streamline simulation. In: SPE/AAPG Western Regional Meeting. Long Beach, California, USA (2000). SPE 62557
12. IO Center, NTNU: The Norne benchmark case. url: `http://www.ipt.ntnu.no/~norne/wiki/doku.php` (2012)
13. Izgec, O., Sayarpour, M., Shook, G.M.: Maximizing volumetric sweep efficiency in waterfloods with hydrocarbon F-ϕ curves. J. Petrol. Sci. Eng. **78**(1), 54–64 (2011). DOI 10.1016/j.petrol.2011.05.003
14. Kippe, V., Hægland, H., Lie, K.A.: A method to improve the mass-balance in streamline methods. In: SPE Reservoir Simulation Symposium. SPE 106250, Houston, Texas, U.S.A (2007). DOI 10.2118/106250-MS
15. Klausen, R.A., Rasmussen, A.F., Stephansen, A.: Velocity interpolation and streamline tracing on irregular geometries. Comput. Geosci. **16**, 261–276 (2012). DOI 10.1007/s10596-011-9256-0
16. Klemetsdal, Ø., Møyner, O., Raynaud, X., Lie, K.A.: Non-linear Newton solver for a polymer two-phase system using interface-localized trust regions. In: IOR 2017 – 19th European Symposium on Improved Oil Recovery, 24–27 April, Stavanger, Norway. EAGE (2017). DOI 10.3997/2214-4609.201700356
17. Lake, L.W.: Enhanced Oil Recovery. Prentice-Hall (1989)
18. Leube, P.C., Nowak, W., Schneider, G.: Temporal moments revisited: Why there is no better way for physically based model reduction in time. Water Resour. Res **48**(11) (2012). DOI 10.1029/2012WR011973. W11527
19. Lie, K.A.: An Introduction to reservoir simulation using MATLAB: User guide for the Matlab Reservoir Simulation Toolbox (MRST). SINTEF ICT, www.sintef.no/Projectweb/MRST/publications (2016)
20. Mead, L.R., Papanicolaou, N.: Maximum entropy in the problem of moments. Journal of Mathematical Physics **25**(8), 2404–2417 (1984). DOI 10.1063/1.526446
21. Møyner, O.: Nonlinear solver for three-phase transport problems based on approximate trust regions. In: ECMOR XV – 15th European Conference on the Mathematics of Oil Recovery. EAGE, Amsterdam, The Netherlands (2016)
22. Møyner, O., Krogstad, S., Lie, K.A.: The application of flow diagnostics for reservoir management. SPE J. **20**(2), 306–323 (2014). DOI 10.2118/

171557-PA
23. MRST: The MATLAB Reservoir Simulation Toolbox. www.sintef.no/MRST (2016b)
24. Natvig, J.R., Lie, K.A., Eikemo, B.: Fast solvers for flow in porous media based on discontinuous Galerkin methods and optimal reordering. In: P. Binning, P. Engesgaard, H. Dahle, G. Pinder, W. Gray (eds.) Proceedings of the XVI International Conference on Computational Methods in Water Resources. Copenhagen, Denmark (2006). URL http://proceedings.cmwr-xvi.org/
25. Natvig, J.R., Lie, K.A., Eikemo, B., Berre, I.: An efficient discontinuous Galerkin method for advective transport in porous media. Adv. Water Resour. **30**(12), 2424–2438 (2007). DOI 10.1016/j.advwatres.2007.05.015
26. Park, H.Y., Datta-Gupta, A.: 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 (2011). DOI 10.2118/144580-MS
27. Pope, G.A.: The application of fractional flow theory to enhanced oil recovery. Society of Petroleum Engineers Journal **20**(03), 191–205 (1980). DOI 10.2118/7660-PA
28. Rashid, B., Bal, A.L., Williams, G.J., Muggeridge, A.H.: Using vorticity to quantify the relative importance of heterogeneity, viscosity ratio, gravity and diffusion on oil recovery. Comput. Geosci. **16**(2), 409–422 (2012). DOI 10.1007/s10596-012-9280-8
29. Rashid, B., Muggeridge, A., Bal, A.L., Williams, G.J.J.: Quantifying the impact of permeability heterogeneity on secondary-recovery performance. SPE J. **17**(2), 455–468 (2012). DOI 10.2118/135125-PA
30. Rasmussen, A.F., Lie, K.A.: Discretization of flow diagnostics on stratigraphic and unstructured grids. In: ECMOR XIV – 14^{th} European Conference on the Mathematics of Oil Recovery, Catania, Sicily, Italy, 8-11 September 2014. EAGE (2014). DOI 10.3997/2214-4609.20141844
31. Sehbi, B.S., Kang, S., Datta-Gupta, A., Lee, W.J.: Optimizing fracture stages and completions in horizontal wells in tight gas reservoirs using drainage volume calculations. In: Proceedings of the North American Unconventional Gas Conference and Exhibition. 14-16 June 2011, The Woodlands, Texas, USA (2011). DOI 10.2118/144365-MS
32. Shahvali, M., Mallison, B., Wei, K., Gross, H.: An alternative to streamlines for flow diagnostics on structured and unstructured grids. SPE J. **17**(3), 768–778 (2012). DOI 10.2118/146446-PA
33. Shook, G.M., Forsmann, J.H.: Tracer interpretation using temporal moments on a spreadsheet. Tech. Rep. INL report 05-00400, Idaho National Laboratory (2005)
34. Sorbie, K.S.: Polymer-improved oil recovery. Springer Science & Business Media (1991)
35. Thiele, M.R.: Streamline simulation. In: 8th International Forum on Reservoir Simulation. Stresa / Lago Maggiore, Italy (2005)
36. Thiele, M.R., Batycky, R.P.: Water injection optimization using a streamline-based workflow. In: Proceedings of the SPE Annual Technical Conference and Exhibition. 5-8 October 2003, Denver, Colorado (2003). DOI 10.2118/84080-MS

37. Wen, T., Thiele, M.R., Ciaurri, D.E., Aziz, K., Ye, Y.: Waterflood management using two-stage optimization with streamline simulation. Comput. Geosci. **18**(3-4), 483–504 (2014). DOI 10.1007/s10596-014-9404-4
38. Zhou, Y., Muggeridge, A.H., Berg, C.F., King, P.R.: Quantifying viscous cross-flow and its impact on tertiary polymer flooding in heterogeneous reservoirs. In: IOR 2015-18th European Symposium on Improved Oil Recovery (2015)