An Unconditionally Stable Splitting Method Using Reordering for Simulating Polymer Injection

Knut-Andreas Lie

Halvor Møll Nilsen Xavier Raynaud Atgeirr Flø Rasmussen

September 13, 2012

Abstract

We present an unconditionally stable algorithm for sequential solution of flow and transport that can be used for efficient simulation of polymer injection modeled as a two-phase system with rock compressibility and equal fluid compressibilities. Our formulation gives a set of nonlinear transport equations that can be discretized with standard implicit upwind methods to conserve mass and volume independent of the time step. The resulting nonlinear system of discrete transport equations can, in the absence of gravity and capillary forces, be permute to lower triangular form by using a simple topological sorting method from graph theory. This gives a nonlinear Gauss–Seidel method that computes the solution cell by cell with local iteration control. The single-cell systems can be reduced to a nested set of scalar nonlinear equations that can easily be bracketed and solved with standard gradient or root-bracketing methods. The resulting method gives orders-of-magnitude reduction in runtimes and increases the feasible time-step sizes. Hence, sequential splitting combined with standard upwind discretizations can become a viable alternative to streamline methods for speeding up simulation of advection-dominated systems.



Introduction

Polymer polymer flooding is a widely used EOR strategy, in particular for heavy oil recovery [4, 14, 15]. Heavy crude oil resources are estimated to be more than twice the resources of conventional crude oil. On the other hand, such reservoirs typically have a large percentage of residual or non-recoverable oil and enhanced oil recovery is therefore essential to increase each field's lifetime and ultimately the world's oil resources. Many examples of technically successful field-projects with polymer and surfactant-polymer flooding are reported in the literature. A recent review of real-field operations of EOR based on polymer injection [6] shows great promise, but also points out that it is crucial to evaluate the injection strategies.

Heavy oil has much higher viscosity than water, and polymer is therefore injected to establish more favorable mobility ratios. The fluid behaviour is highly nonlinear and the resulting equations are very stiff. In particular, the water viscosity is strongly affected by the polymer concentration and it is crucial to capture polymer fronts sharply to resolve the nonlinear displacement mechanisms correctly. Polymer fronts are, unlike water fronts, not self-sharpening and high numerical resolution is therefore required to limit the numerical diffusion that would otherwise bias or deteriorate simulation results. Altogether, this makes simulation of enhanced recovery of heavy oil computationally challenging. The main purpose of this paper is to develop a highly efficient and unconditionally stable simulation method for simulating polymer injection on high-resolution geo-cellular grid models; such models are currently outside the reach of conventional solvers based on a fully implicit discretization. High-resolution models tend to have large differences in cell volumes and face areas that effectively preclude the use of high-order discretization methods relying on explicit time stepping. We have therefore chosen to develop an implicit method that is implicit and first order in time, accompanied by a set of nonlinear solvers are robust over a large span of CFL numbers and so fast that numerical diffusion can be reduced by small time steps and high resolution. In this context, small means time steps that are near the CFL limit of explicit schemes for the cells that contribute most to the numerical diffusion. In other words, first-order implicit and explicit schemes will in practice have equal numerical diffusion if the time step of the implicit method is approximately the same as the CFL restriction for explicit schemes in cells with high flow and large volumes.

In the following we consider a two-phase model with polymer in the water phase. The model assumes rock compressibility, identical phase compressibilities, pressure-dependent transmissibility, gravity, but no capillary effects. In the recovery of heavy oil, the gravity contrast between the injected water and the oil is typically (much) smaller than in traditional oil field operations, thereby giving a looser coupling between viscous and gravity forces that can be exploited to develop highly efficient finite-volume solvers [7–9]. To this end, we first formulate a sequential solution method that conserves mass and volume and has no restriction on the pressure and transport steps. Then, we show how the resulting transport equations can discretized by a standard, implicit, upstream mobility-weighted method giving a discrete nonlinear system that can be permuted to triangular form and solved cell-by-cell with local control over the nonlinear iterations. Each single-cell problem can furthermore be reduced to a nested set of scalar nonlinear equations that can be bracketed and thus is guaranteed to have a unique solution for arbitrary large time steps. Altogether, this gives a very robust and efficient method that is fully competitive with state-of-the-art streamline methods [2, 3, 13]. The new method is verified and validated on three test cases: a simple 1D case that describes injection of a fully miscible or a partially miscible polymer slug, a 2D case that describes water upconing, and a 3D real-field model of polymer injection.

Mathematical Model

We start with the most general system for which we can devise an unconditionally stable, sequential splitting method, namely a two-phase system with polymer, rock compressibility, and identical fluid



compressibilities for the two phases. The continuous transport equations can be written

$$\frac{\partial}{\partial t}(\rho_{\alpha}\phi S_{\alpha}) + \nabla \cdot (\rho_{\alpha}\vec{v}_{\alpha}) = 0, \qquad \alpha \in \{w, o\},$$

$$\frac{\partial}{\partial t}(\rho_{w}\phi S_{w}c) + \nabla \cdot (c\rho_{w}\vec{v}_{wp}) = 0.$$
(1)

Here, ρ_{α} , \vec{v}_{α} , and S_{α} denote the density, velocity and saturation terms of phase α , with $\alpha = w, o$ denoting the water and oil phases. Furthermore, ϕ is the porosity, *c* the polymer concentration and \vec{v}_{wp} the velocity of water with polymer. Sources and sinks may be included in a manner equivalent to boundary conditions, and are left out of the above equations.

Simple PVT behaviour is modeled through the factor $b = \rho_{\alpha}/\rho_{\alpha,S}$, which is assumed to be the same function of pressure for both phases. The system can be simplified by dividing each equation with the relevant surface density $\rho_{\alpha,S}$.

$$\frac{\partial}{\partial t}(b\phi S_{\alpha}) + \nabla \cdot (b\vec{v}_{\alpha}) = 0, \qquad \alpha \in \{w, o\},$$

$$\frac{\partial}{\partial t}(b\phi S_{w}c) + \nabla \cdot (bc\vec{v}_{wp}) = 0.$$
(2)

Darcy's law for the two phases can be written $\vec{v}_{\alpha} = -K\lambda_{\alpha}(\nabla p - b\rho_{\alpha,S}\vec{g})$, where $\lambda_{\alpha} = k_{\alpha}/\mu_{\alpha}$ denotes the fluid mobility for $\alpha = o, w$. We introduce the total mobility $\lambda = \lambda_o + \lambda_w$ and let $f_{\alpha} = \lambda_{\alpha}/\lambda$. As long as λ_{α} are nonnegative, monotone, and equal zero for $S_{\alpha} = 0$, the fractional flow functions will be monotone and have end-point values $f_{\alpha} = 0$ for $S_{\alpha} = 0$ and $f_{\alpha} = 1$ for $S_{\alpha} = 1$. With this, the equation for the total velocity $\vec{v} = \vec{v}_w + \vec{v}_o$ reads

$$\vec{v} = -K\lambda \left(\nabla p - \sum_{\alpha} b\rho_{\alpha,S} f_{\alpha} \vec{g}\right).$$
(3)

Next, we describe a simple polymer model. Polymer is diluted in the water. We denote by *c* the concentration of polymer. The Todd–Longstaff model introduces a mixing parameter that takes into account the degree of mixing of polymer into water. The viscosity of a fully mixed polymer solution, $\mu_m(c)$, is a function of the concentration. The effective polymer viscosity is defined as

$$\mu_{p,\text{eff}} = \mu_m(c)^{\omega} \mu_p^{1-\omega},$$

where $\mu_p = \mu_m(c_{\text{max}})$ and $\omega \in [0, 1]$ denotes the mixing parameter. The viscosity of the partially mixed water is given in a similar way by

$$\mu_{w,e} = \mu_m(c)^{\omega} \mu_w^{1-\omega}.$$

To define the effective water viscosity, $\mu_{w,eff}$, we interpolate linearly between the inverse of the effective polymer viscosity and the partially mixed water viscosity:

$$\frac{1}{\mu_{w,\text{eff}}} = \frac{1 - c/c_{\text{max}}}{\mu_{w,e}} + \frac{c/c_{\text{max}}}{\mu_{p,\text{eff}}}.$$
(4)

To take into account the incomplete mixing, we introduce the velocity of water that contains polymer, which we denote \vec{v}_{wp} . For this part of water, the relative permeability is denoted by k_{rwp} and the viscosity is equal to $\mu_{p,eff}$. We also consider the total water velocity, which we still denote \vec{v}_w and the viscosity is there given by $\mu_{w,eff}$. Darcy's law then gives us

$$\vec{v}_w = -\frac{k_{rw}}{\mu_{w,\text{eff}}} \mathbf{K}(\nabla p_w - \rho_w g \nabla z), \qquad \vec{v}_{wp} = -\frac{k_{rwp}}{\mu_{p,\text{eff}}} \mathbf{K}(\nabla p_w - \rho_w g \nabla z) = m(c) \vec{v}_w, \tag{5}$$

as we assume that the presence of polymer does not affect the pressure and the density. We have also assumed that the permeability does not depend on mixing so that $k_{rwp} = k_{rw}$. A useful formula for m(c) can easily be derived:

$$m(c) = \frac{\mu_{w,\text{eff}}}{\mu_{p,\text{eff}}} = \left[\left(1 - \frac{c}{c_{\text{max}}} \right) \left(\frac{\mu_p}{\mu_w} \right)^{1-\omega} + \frac{c}{c_{\text{max}}} \right]^{-1}.$$
(6)



Solution Strategy

We now formulate an unconditionally stable, sequential splitting method. By this, we mean a method that is mass conservative, preserves consistency between pressures and masses, has no time-step restriction between pressure and transport steps, has no restrictions on the transport steps, and uses a number of operation that is limited independently of the time step.

Discretization

We start by discretizing in space, first integrating over a grid cell C_i and applying the divergence theorem:

$$\int_{C_{i}} \frac{\partial}{\partial t} (b\phi S_{\alpha}) dx + \int_{\partial C_{i}} b\vec{v}_{\alpha} \cdot \vec{n} dS = 0, \qquad \alpha \in \{w, o\},$$

$$\int_{C_{i}} \frac{\partial}{\partial t} (b\phi S_{w}c) dx + \int_{\partial C_{i}} bc\vec{v}_{wp} \cdot \vec{n} dS = 0.$$
(7)

Replacing b, ϕ , and S_{α} by cell averages in the first integral and splitting the boundary integral into a sum over cell faces we obtain

$$\frac{\partial}{\partial t}(b_i\phi_i S_{\alpha,i}) + \sum_j b_{ij}v_{\alpha,ij} = 0, \qquad \alpha \in \{w, o\},$$

$$\frac{\partial}{\partial t}(b_i\phi_i S_{w,i}c_i) + \sum_j b_{ij}c_{ij}v_{wp,ij} = 0.$$
(8)

where the sum over *j* is taken to mean over all *j* such that the cells C_i and C_j share a common face, the $v_{i,ij}$ are signed fluxes through this common face, and therefore antisymmetric except for outer boundary fluxes and fluxes representing wells, with $v_{i,ij} > 0$ if the direction of flow is $i \rightarrow j$. All other variables indexed by *ij* are assumed to be symmetric in *i* and *j*, they will typically be evaluated upwind.

Discretizing in time with an implicit Euler scheme, we obtain

$$(b_i\phi_i S_{\alpha,i})^{n+1} - (b_i\phi_i S_{\alpha,i})^n + \Delta t \sum_j (b_{ij}v_{\alpha,ij})^{n+1} = 0, \qquad \alpha \in \{w, o\},$$

$$(b_i\phi_i S_{w,i}c_i)^{n+1} - (b_i\phi_i S_{w,i}c_i)^n + \Delta t \sum_j (b_{ij}c_{ij}v_{wp,ij})^{n+1} = 0.$$
(9)

To derive a discrete pressure equation, we sum the two phase equations using the condition $S_w + S_o = 1$ to obtain

$$(b_i\phi_i)^{n+1} - (b_i\phi_i)^n + \Delta t \sum_j (b_{ij}v_{ij})^{n+1} = 0$$
(10)

where v_{ij} denotes the flux corresponding to the total velocity \vec{v} . The equation necessary to compute discrete pressures and fluxes is now formed by combining (10) with a discrete form of (3).

Pressure-transport splitting

In the following we will use a splitting method in which the pressure equation (10) and the transport equations (9) are solved sequentially. Since we will only use the transport equations for water and polymer, from now on we take *S* to mean S_w . If a splitting between pressure and transport is to be mass and volume conserving, it is crucial that the pressure equation does not depend on the saturations calculated in the transport step. We therefore rewrite the phase transport equations from (9) and insert the pressure equation (10) to eliminate b^{n+1} and ϕ^{n+1} . Using the observation that

$$(b\phi S)^{n+1} - (b\phi S)^n = (b\phi)^n (S^{n+1} - S^n) + S^{n+1} ((b\phi)^{n+1} - (b\phi)^n),$$



we obtain

$$(b_i\phi_i)^n [S_i^{n+1} - S_i^n] - S_i^{n+1}\Delta t \sum_j (b_{ij}v_{ij})^{n+1} + \Delta t \sum_j (b_{ij}f_{ij}v_{ij})^{n+1} = 0,$$

$$(b_i\phi_i)^n [(S_ic_i)^{n+1} - (S_ic_i)^n] - (S_ic_i)^{n+1}\Delta t \sum_j (b_{ij}v_{ij})^{n+1} + \Delta t \sum_j (b_{ij}c_{ij}v_{wp,ij})^{n+1} = 0,$$
(11)

where the fractional flow for water f can be used to express the water flux as $\vec{v}_w = f\vec{v}$ in the absence of gravity. As before, f is assumed to be a monotone function of S_w on [0,1] with $f(0,\cdot) = 0$ and $f(1,\cdot) = 1$.

Mass conservation for the full time-step follows from the symmetry of b_{ij} , f_{ij} , and c_{ij} and the antisymmetry of v_{ij} and $v_{wp,ij}$:

$$\Delta M_{w} = \sum_{i} (b_{i}\phi_{i}S_{i})^{n+1} - \sum_{i} (b_{i}\phi_{i}S_{i})^{n} = -\Delta t \sum_{i} \sum_{j} (b_{ij}f_{ij}v_{ij})^{n+1} = 0,$$

$$\Delta M_{\text{polymer}} = \sum_{i} (b_{i}\phi_{i}S_{i}c_{i})^{n+1} - \sum_{i} (b_{i}\phi_{i}S_{i}c_{i})^{n} = -\Delta t \sum_{i} \sum_{j} (b_{ij}c_{ij}v_{wp,ij})^{n+1} = 0.$$
(12)

However, if one wants to use a different time-step for equations (10) and (11), care must be taken with the values used for *b* and ϕ . Letting $\delta \in [0, 1]$ denote the partial time-step, we obtain mass conservation if we use

$$(b\phi)^{n+\delta} = (1-\delta)(b\phi)^n + \delta(b\phi)^{n+1}.$$
(13)

With $\Delta t_{\text{transport}} = \delta \Delta t$, we obtain

$$\Delta M_{w,i}^{\delta} = (b_i \phi_i S_i)^{n+\delta} - (b_i \phi_i S_i)^n$$

$$= (b_i \phi_i)^n (S_i^{n+\delta} - S_i^n) + S_i^{n+\delta} \left((b_i \phi_i)^{n+\delta} - (b_i \phi_i)^n \right)$$

$$= S_i^{n+\delta} \delta \Delta t \sum_j (b_{ij} v_{ij})^{n+1} - \delta \Delta t \sum_j (b_{ij} v_{ij})^{n+1} f_{ij}^{n+\delta}$$

$$+ S_i^{n+\delta} \left((1-\delta) (b_i \phi_i)^n + \delta (b_i \phi_i)^{n+1} - (b_i \phi_i)^n \right)$$

$$= -\delta \Delta t \sum_j (b_{ij} v_{ij})^{n+1} f_{ij}^{n+\delta}$$
(14)

which yields zero when summed over *i* due to antisymmetry of v_{ij} as before. We have now shown that we conserve mass from step *n* to step $n + \delta$; mass conservation can also be shown to hold for arbitrary steps. A similar analysis can be applied to the polymer mass.

Reordering to a sequence of single-cell problems

If gravity is neglected, the velocity field \vec{v} has no loops. The discrete equivalence of this property is that the directed graph describing the fluxes is acyclic and can be topologically sorted. Natvig and Lie [9] used this property to reduce the nonlinear system of discretized transport equations to a triangular system that could be solved using a Gauss–Seidel type algorithm that computes the solution one cell at the time. For incompressible two-phase flow, they showed that the Gauss-Seidel algorithm could achieve two orders-of-magnitude speedup for many cases. Herein, we use the same idea to reduce the transport equations for the polymer model to a sequence of single-cell 2×2 systems. To this end, we assume that the pressure equation is solved using a monotone method, we use the standard two-point flux approximation scheme, which will guarantee that the directed graph induced by the fluxes is acyclic.

We then perform a topological sort on the flux graph and introduce two index sets: U(i) denotes all upwind neighbours of cell *i* and D(i) denotes all downwind neighbours. We use upstream-cell evaluation



for f_{ij} . Using this notation, the residuals of the single-cell system reads

$$R_{w,i}(S_i^{n+1}, c_i^{n+1}) = (b_i\phi_i)^n [S_i^{n+1} - S_i^n] - S_i^{n+1}\Delta t \sum_j (b_{ij}v_{ij})^{n+1} + \Delta t \sum_{j \in U(i)} f(S_j^{n+1}, c_j^{n+1}) (b_{ij}v_{ij})^{n+1} + \Delta t f(S_i^{n+1}, c_i^{n+1}) \sum_{j \in D(i)} (b_{ij}v_{ij})^{n+1}, R_{c,i}(S_i^{n+1}, c_i^{n+1}) = (b_i\phi_i)^n [(S_ic_i)^{n+1} - (S_ic_i)^n] - (S_ic_i)^{n+1}\Delta t \sum_j (b_{ij}v_{ij})^{n+1} + \Delta t \sum_{j \in U(i)} c_j^{n+1} m(c_j^{n+1}) f(S_j^{n+1}, c_j^{n+1}) (b_{ij}v_{ij})^{n+1} + \Delta t c_i^{n+1} m(c_i^{n+1}) f(S_i^{n+1}, c_i^{n+1}) \sum_{j \in D(i)} (b_{ij}v_{ij})^{n+1},$$
(15)

where *m* defined by $\vec{v}_{w,p} = m\vec{v}_w$ is the ratio of the flux density of water with and without polymer. A specific expression for *m* is given in (6). In all models considered herein, *m* is assumed to be a function only of the concentration *c*. Solving the equations in topologically sorted order, we may assume that all upwind saturations and concentrations are computed already. For a single cell, (15) is then a nonlinear system of two variables. Such systems can in general have more than one solution, however in the following we show that the system has a single, unique solution. This solution can be found in a bounded number of steps, independent of the size of the time step.

Bracketing the single-cell system

Scalar nonlinear equations h(x) = 0 can be solved robustly and efficiently by so-called root-bracketing algorithms. If h(x) differs in sign from h(b), the function crosses zero at least once in the interval [a,b], which is then guaranteed to contain a root. Given a valid initial interval, bracketing algorithm cannot fail to find the root provided the function h is well behaved. In the following, we will show how to formulate a similar bracketing technique for the single-cell problem (15) that is guaranteed to converge for all time steps Δt . Bracketing methods only require the evaluation of residuals, which are simple to implement and typically inexpensive to compute compared to the Jacobians required by gradient methods. Although the bracketing method may not always have optimal convergence, it can still be used as an efficient nonlinear solver in its own right or be used as a robust fallback strategy for more efficient gradient-based methods.

To develop the bracketing method, we show that we can reduce the 2 × 2 system (15) to a scalar equation $r_i(c) = R_{c,i}(\tilde{S}(c),c)$, where $\tilde{S}(c)$ is defined implicitly by $R_{w,i}(\tilde{S}(c),c) = 0$. In general, $\tilde{S}(c)$ can not be computed analytically. Our numerical procedure is therefore to evaluate $\tilde{S}(c)$ by solving the equation

$$R_{w,i}(\tilde{S}(c),c) = 0 \tag{16}$$

numerically. The resulting algorithm to solve the single-cell problem is hence a nested loop of iterations of fully stable solutions of scalar equations. There are many robust methods for root bracketing, e.g., bisection, secant, regula falsi, interpolation, and inverse interpolation, as well as hybrid methods like Brent's method, which combines bisection, secant, and inverse quadratic interpolation. Herein, we use a modified regula falsi method described in [5], the Pegasus method, for both solves.

To show that our transformation is valid, it is sufficient to show that $\partial R_{w,i}(S)/\partial S > 0$ in [0,1] and that $R_{w,i}(0,c) \cdot R_{w,i}(1,c) < 0$. This is easily verified:

$$R_{w,i}(0,c) = -(b_i\phi_i)^n S_i^n + \Delta t \sum_{j \in U(i)} f(S_j^{n+1}, c_j^{n+1}) (b_{ij}v_{ij})^{n+1} \le 0$$
(17)

since $v_{ij} < 0$ for $j \in U(i)$, and similarly

$$R_{w,i}(1,c) = (b_i\phi_i)^n [1-S_i^n] + \Delta t \sum_{j \in U(i)} [f(S_j^{n+1}, c_j^{n+1}) - 1] (b_{ij}v_{ij})^{n+1} \ge 0.$$
(18)



Finally,

$$\frac{dR_{w,i}(S,c)}{dS} = (b_i\phi_i)^n - \Delta t \sum_j (b_{ij}v_{ij})^{n+1} + \Delta t \frac{df}{dS}(S,c) \sum_{j \in D(i)} (b_{ij}v_{ij})^{n+1}$$

$$= (b_i\phi_i)^{n+1} + \Delta t \frac{df}{dS}(S,c) \sum_{j \in D(i)} (b_{ij}v_{ij})^{n+1} > 0$$
(19)

by using equation (10) and the fact that df/dS > 0 and $v_{ij} > 0$ for $j \in D(i)$.

Similarly, to show that the outer iteration $r_i(c) = 0$ has a unique solution we need to verify that $r_i(0) < 0$, $r_i(c_{\max}) > 0$, and $dr_i/dc > 0$. The proof depends on details of the polymer model, and for the simple model presented above, we only need to show that $r_i(c)$ has a unique solution in $[0, c_{\max}]$. This will require the assumption that $\mu'_m(c) \ge 0$. First, we establish the lower bracket $r_i(0)$:

$$R_{c,i}(S,0) = -(b_i\phi_i)^n (S_ic_i)^n + \Delta t \sum_{j \in U(i)} c_j^{n+1} m(c_j^{n+1}) f(S_j^{n+1}, c_j^{n+1}) (b_{ij}v_{ij})^{n+1} \le 0,$$
(20)

which follows since $v_{ij} < 0$ for $j \in U(i)$, independent of S. For the upper bracket $r_i(c_{\text{max}})$, we write

$$\begin{aligned} R_{c,i}(S,c_{\max}) &= (b_i\phi_i)^n \big[(Sc_{\max}) - (S_ic_i)^n \big] - (Sc_{\max})\Delta t \sum_j (b_{ij}v_{ij})^{n+1} \\ &+ \Delta t \sum_{j \in U(i)} c_j^{n+1} m(c_j^{n+1}) f(S_j^{n+1},c_j^{n+1}) (b_{ij}v_{ij})^{n+1} + \Delta t c_{\max} m(c_{\max}) f(S,c_{\max}) \sum_{j \in D(i)} (b_{ij}v_{ij})^{n+1} \\ &= R_{w,i}(S,c_{\max}) c_{\max} + S_i^n (c_{\max}-c_i^n) + \Delta t \sum_{j \in U(i)} \big[c_j^{n+1} m(c_j^{n+1}) - c_{\max} \big] f(S_j^{n+1},c_j^{n+1}) (b_{ij}v_{ij})^{n+1} \\ &+ \Delta t \big[c_{\max} m(c_{\max}) - c_{\max} \big] f(S,c_{\max}) \sum_{j \in D(i)} (b_{ij}v_{ij})^{n+1}. \end{aligned}$$

Letting $S = \tilde{S}(c_{\max})$ be the solution of $R_w, i(S, c_{\max}) = 0$ and using that $m(c_{\max}) = 1$ and m(c) < 1 otherwise, we have

$$R_{c,i}(\tilde{S}(c_{\max}), c_{\max}) = S_i^n(c_{\max} - c_i^n) + \Delta t \sum_{j \in U(i)} \left[c_j^{n+1} m(c_j^{n+1}) - c_{\max} \right] f(S_j^{n+1}, c_j^{n+1}) (b_{ij} v_{ij})^{n+1} \ge 0.$$

Now we show that $dr_i/dc > 0$. First, we rewrite both residuals using the pressure equation (10) and drop all constant terms (in *S*, *c*) to write

$$\hat{R}_{w,i}(S,c) = SP + fD, \quad \hat{R}_{c,i}(S,c) = ScP + cmfD, \quad P = (b_i\phi_i)^{n+1}, \quad D = \Delta t \sum_{j \in D(i)} (b_{ij}v_{ij})^{n+1}.$$
(21)

Let the curve in the *S*, *c* plane where $R_{w,i}(S,c) = 0$ be described as before by writing $S = \tilde{S}(c)$. Then

$$\frac{d}{dc}r_i(c) = \frac{d}{dc}\hat{R}_{c,i}(\tilde{S}(c),c) = SP + c\frac{dS}{dc}P + cm\frac{df}{dc}D + \frac{d(cm)}{dc}fD = SP + c\frac{dS}{dc}P(1-m) + \frac{d(cm)}{dc}fD,$$

where we have used that along $\tilde{S}(c)$,

$$\frac{d\hat{R}_{w,i}}{dc} = \frac{dR_{w,i}}{dc} = 0 \implies \frac{dS}{dc}P = -\frac{df}{dc}D.$$

The first term is clearly nonnegative, the other terms are also nonnegative if dS/dc = -(D/P)df/dcand dm/dc are nonnegative:

$$\frac{df}{dc} = \frac{d}{dc} \left(\frac{k_{rw}\mu_o}{k_{rw}\mu_o + k_{ro}\mu_{w,\text{eff}}} \right) = \frac{-k_{rw}\mu_o k_{ro}\frac{d}{dc}(\mu_{w,\text{eff}})}{(k_{rw}\mu_o + k_{ro}\mu_{w,\text{eff}})^2},$$
$$\frac{dm}{dc} = \frac{\left[\left(\frac{\mu_p}{\mu_w}\right)^{1-\omega} - 1 \right] / c_{\text{max}}}{\left[\left(1 - \frac{c}{c_{\text{max}}}\right) \left(\frac{\mu_p}{\mu_w}\right)^{1-\omega} + \frac{c}{c_{\text{max}}} \right]^2}.$$

ECMOR



Figure 1 Convergence of the bracketed and a standard Newton solver for the cell at x = 180 from the simulation shown in Figure 3. The surface is the square of the water residual, $R_{w,i}(S,c)^2$. The Newton method converges in three steps (green dots, black numbers) to the solution indicated by the black dot. The bracketed solver converges in nine outer iterations; red dots and red numbers indicate the corresponding evaluations of the polymer residual $r_i(c)$ in the outer iteration, whereas the blue dots denote the points at which the water residual is evaluated in the inner regula-falsi iterations.

Recall the assumption that $\mu'_m(c) \ge 0$. Using that $d\mu_{w,eff}/dc$ is nonnegative gives that dS/dc. Also $\mu_p/\mu_w \ge 1$ and so dm/dc is also nonnegative. Since the first term, *SP* is only zero for S = 0, we have that $dr_i/dc > 0$ except for the singular S = 0 case.

Figure 1 illustrates the path followed by our bracketing solver, using a modified regula falsi solver for each scalar equation. The robustness of the method is based on finding the zero of the water residual $R_{w,i}(S,c)$ (the lowest point on the surface in Figure 1) for a fixed *c* in the inner iterations, and then follow the path formed by these zero points to the solution in the outer iteration. In the figure, one can clearly observe that the nested use of a modified regula falsi solver is far from optimal because many residuals are evaluated far from the solution even if the initial guess is quite good. On the other hand, the method is robust and simple. In contrast, the Newton method converges in three iterations. However, from the shape of the residual surface, it is evident that the Newton method may encounter problems, e.g., when starting in a region where the residual is almost flat. In our production code that will be used on multimillion cell models, we only use the bracketed solver as a fail-safe fallback option if the Newton method fails.

Operator splitting for gravity

One can easily formulate a similar splitting of the pressure and transport equations that is consistent and devoid of mass conservation errors for models containing gravity and source terms. However, nonzero gravity introduces a stronger coupling between the pressure and transport equations and changes the structure of the velocity field. The first issue mainly limits the feasible sizes of the pressure steps, whereas the second affects the possibility of reordering the discrete transport system to a sequence of



single-cell problems. To be able to use the highly efficient solvers we have discussed above, we therefore introduce an additional operator splitting for the transport equations

$$\frac{\partial}{\partial t} (b\phi S_w) + \nabla \cdot \left[bf \left(\vec{v} + \lambda_o (\rho_w - \rho_o) K \vec{g} \right) \right] = 0$$

$$\frac{\partial}{\partial t} (b\phi S_w c) + \nabla \cdot \left[bcm f \left(\vec{v} + \lambda_o (\rho_w - \rho_o) K \vec{g} \right) \right] = 0.$$
(22)

That is, we solve two sets of equations, a system of advective equations that account for viscous effects

$$\frac{\partial}{\partial t}(b\phi S_w) + \nabla \cdot (bf\vec{v}) = 0, \qquad \frac{\partial}{\partial t}(b\phi S_w c) + \nabla \cdot (bcmf\vec{v}) = 0, \tag{23}$$

and a system of segregation equations that account for gravity effects

$$\frac{\partial}{\partial t}(b\phi S_w) + \nabla \cdot \left[bf\lambda_o(\rho_w - \rho_o)K\vec{g} \right] = 0, \qquad \frac{\partial}{\partial t}(b\phi S_w c) + \nabla \cdot \left[bcmf\lambda_o(\rho_w - \rho_o)K\vec{g} \right] = 0.$$
(24)

The equations are solved sequentially, possibly with different methods. For the segregation equation (24), we may in some cases use solvers exploiting its structure, for example by solving the equation for columns of cells in isolation if the grid cells are vertically aligned. For equation (23), the solver exploiting topological sorting described above can be used.

We still have to handle the possibility that the total velocity \vec{v} contains loops, i.e., blocks of mutually dependent cells that cannot be solved one by one in order. It is possible to split \vec{v} in two parts, one that can be reordered perfectly, and one that contains loops, by the technique described in [8]. This is useful if the loops cover many cells and their flows are large, but this is not the case for the test problems we are investigating in this article, related to EOR of heavy oil with small density differences and large injection rates. In [9, 10], the coupled multi-cell systems were solved using a standard stabilised Newton method. Herein, we will instead use an iterated nonlinear Gauss-Seidel method in which we solve the multicell block one cell at a time, repeating the process until an acceptable converged solution is obtained. We have not yet established a formal convergence analysis, but the method appears to be surprisingly efficient for all test cases we have run so far.

Numerical examples

In the following, we will present three examples that demonstrate the utility of our numerical strategy. To this end, we will use polymer models that have many of the features that are typically found in contemporary commercial simulators; one simple example of such a model was described above. In all the following examples, the nonlinear transport equations will be reduced to a sequence of 2×2 single-cell problems that can either be solved by the bracketed solver discussed above or by a standard Newton–Raphson solver with the nested bracketing method as a fallback if Newton iterations fail.

Example 1: Polymer slug in a 1D reservoir.

In the first example, we consider a 1D homogeneous reservoir that is initially filled with an oil with viscosity 100 cP. To produce the oil, we first inject water with viscosity 1 cP from the left side for 200 days, then a polymer slug for 200 days, before continuing injection of pure water for another 100 days. The polymer will increase the water viscosity by a factor twenty. To illustrate the advantage of our nonlinear Gauss–Seidel solvers, we consider two different flow scenarios, parameterized by the Todd–Longstaff parameter ω that models the effect the effect of the porous media on the mixing of polymer and water.

Figure 2 shows simulation results for mixing parameter $\omega = 1$, which corresponds to full mixing of polymer and water. In the simulations, we have used two different time steps, a small time step $\Delta t = 2.5$





Figure 2 Simulation of the injection of a polymer slug in a 1D reservoir for a scenario with full mixing between polymer and water (Todd-Longstaff parameter $\omega = 1$). Water saturation and normalized polymer concentration are shown as blue and red lines, respectively. The bar plots show the number of evaluations of the single-cell water and polymer residuals $R_{w,i}$ and $R_{c,i}$ from (15).





Figure 3 Simulation of the injection of a polymer slug in a 1D reservoir for a scenario with partial mixing between polymer and water (Todd-Longstaff parameter $\omega = 0.1$). Water saturation and normalized polymer concentration are shown as blue and red lines, respectively. The bar plots show the number of evaluations of the single-cell water and polymer residuals $R_{w,i}$ and $R_{c,i}$ from (15).

days and a large time step $\Delta t = 25$ days. The plots to the left show solutions computed with the bracketed solver and the plots to the right solutions computed with the Newton method. Comparing the solutions computed with short and long time steps, we see a qualitative difference in the region ahead of the polymer slug. With small time steps, the polymer front is not smeared out and the simulation is able to accurately predict the oil bank that is pushed by the polymer slug. As expected, the evaluation of residuals is located to the flooded region for both nonlinear solvers. For the bracketed solver, the number of residual evaluations seems to be relatively uneffected by the size of the time step. For the Newton solver, on the other hand, the number of iterations and hence the number of residual evaluations increases with the time step. For small time steps (close to the CFL limit), the Newton solver needs significantly fewer iterations away from fronts in the solution. Conversely, the method failed to converge in some cells for a few of the earlier steps not shown in the figure, and hence had to fall back to the bracketing method, which introduced extra evaluations of the water residual. Finally, we observe that even though the bracketing solver has a higher number of evaluations of saturation residuals, the computational cost is comparable to that of the Newton solver, which needs to evaluation both residuals and compute corresponding Jacobians in all iteration steps.

Figure 3 shows the same set of simulations for a scenario with partial mixing (parameter $\omega = 0.1$). The main matematical difference between the two scenarios is that the polymer equation now has stronger nonlinearity so that the intruding polymer has a self-sharpening fronts and hence will be less affected by numerical diffusion. Still, we see a significant difference in the ability to capture the oil bank ahead of the polymer slug for small and large time-step sizes. We also see that the number of iterations for both the Newton and the bracketing method is significantly larger than for fully mixed case. In particular, our



current implementation of the bracketing method uses a large number of iterations in the region where polymer is present.

Example 2: Upconing in 2D.

We consider a $1525 \times 1000 \times 100$ m³ reservoir discretised with $61 \times 1 \times 100$ gridblocks that initially has an oil-water contact at the depth of 2335 meters, see Figure 4. The porosity is constant at 0.28. To produce the oil, we use a symmetric well pattern consisting of two production wells at the east and west sides and a single injector in the middle of the reservoir that injects a mixture of polymer and water at a rate of 3000 m³ per day. All three wells are perforated only in the oil zone. The upper plots in Figure 4 show the water saturation and polymer concentration 2000 days after the start of the injection. The drawdown of the interface between water and oil is mainly caused by the large viscosity ratio between oil and water and not the gravity. We notice that main computational costs of both the bracketed and the Newton solvers are focused around steep fronts and in areas where the polymer is present. Here, the Newton method is (significantly) better at localizing the computations. However, we believe that more tuning of the bracketing method will make it more comparable to the Newton.

Example 3: Real-life model.

We consider a realistic case based on a real-field model that consists of 600 000 cells and has approximately forty wells. Polymer is injected along with water from all injection wells at a concentration that increases the water viscosity by a factor fifty. Polymer and water are assumed to perfectly mixed (i.e, we use $\omega = 1$). As in the examples above, we perform simulations using both short and long time steps and use the bracketed and the Newton solver. Figure 5 shows water saturation and polymer concentration after 1000 days along with the number of iterations used in the last time step for all methods. The most important observation from the figure is that the work is limited to regions around wells and near the oil-water contact, where either the water saturation or the concentation of polymer changes.

Using AGMG [1, 11] as linear solver, each pressure solve takes approximately 10 seconds and converges within 4 nonlinear iterations. One transport step, on the other hand, takes approximately 0.5 seconds. This means that we obtain twenty transport steps for the cost of a single pressure step, which gives us the possibility of reducing the time-steps to increase the resolution of polymer fronts. However, further research has to be done to find an optimal time-stepping strategy for achiving the best accuracy for the given computational cost.

Conclusions

We have presented a fast simulation method for a realistic, but simplified version of the polymer injection problem. The method is based on sequential splitting of the pressure and transport equation and a splitting between advection and gravity segregation in the transport step. It also exploits topological sorting based on flow direction. As such, our method is based on the same physical features of the governing equations that are utilized by steamline solvers. Our method, however, is based solely on standard finite-volume discretizations and hence avoid certain difficulties that are inherent in streamline methods: lack of mass conservation, allocation of fluxes to streamlines, mapping between streamlines and physical grid, tracing of streamlines in irregular geometries, etc.

The key points to the high efficiency of our method are: (i) to exploit the loose coupling between flow and transport and use larger steps for the pressure solves than for the transport solves to both decrease the computational cost and increase the numerical resolution of advancing fronts; and (ii) to localize the nonlinear solves of the advective part of transport step to significantly reduce computational costs. A further advantage is that we have an unconditionally stable method for solving the localized single-cell problems that avoids time-step restrictions induced by stability problems. In particular, the method is very robust with respect to the large differences in cell volumes that are typically present in realistic





Figure 4 Water upconing in a 2D reservoir with a semi-realistic polymer model.





Figure 5 Simulation of polymer injection for a real-field model. All plots show quantities from the last step in a simulation of 1000 days

ECMOR X high-resolution geo-cellular models. The method is particularly efficient for systems in which rock

compressibility dominates fluid compressibilities and gravity only governs slow processes; such systems are highly relevant for enhanced oil recovery of heavy oil.

The method is implemented as part of the Open Porous Media (OPM) initiative and is freely available at [12] under the GPLv3 license. The current software implements a standard polymer model that has more features than described herein, including source terms, dead pore space, adsorption, and permeability reduction. Further research and development will be needed to determine for which cases one can achieve similar performance in practice as for the simplified system discussed in this article. Particular challenges include handling operator splitting for fluids with different compressibility and developing efficient localization strategies for the nonlinear iterations for flow fields that contain more loops and have stronger gravity effects.

Acknowledgements

The authors would like to thank Statoil Petroleum A/S for funding and access to data sets for simulator testing. We also thank Vegard Kippe, Alf Birger Rustad, Stein Krogstad, Bård Skaflestad, Kristin M. Flornes, and Ove Sævareid for fruitful discussions and feedback on our work.

References

- [1] AGMG [2012] Iterative solution with AGgregation-based algebraic MultiGrid. http://homepages.ulb.ac.be/~ynotay/AGMG/.
- [2] AlSofi, A.M. and Blunt, M.J. [2010] Streamline-based simulation of non-Newtonian polymer flooding. SPE J., 15(4), 895–905, doi:10.2118/123971-PA.
- [3] Clemens, T., Abdev, J. and Thiele, M.R. [2011] Improved polymer-flood management using streamlines. SPE J., 14(2), 171–181, doi:10.2118/132774-PA.
- [4] Fletcher, P. et al. [2012] Improving heavy oil recovery using an enhanced polymer system. SPE Improved Oil Recovery Symposium, 14-18 April 2012, Tulsa, Oklahoma, USA.
- [5] Ford, J.A. [1995] Improved algorithms of illinois-type for the numerical solution of nonlinear equations. Tech. Rep. CSM-257, University of Essex.
- [6] Gao, C.H. [2011] Scientific research and field applications of polymer flooding in heavy oil recovery. J. Petrol. Explor. Prod. Technol., 1, 65–70.
- [7] Kwok, F. and Tchelepi, H. [2007] Potential-based reduced Newton algorithm for nonlinear multiphase flow in porous media. J. Comput. Phys., 227(1), 706-727, doi:10.1016/j.jcp.2007.08.012.
- [8] Lie, K.A., Natvig, J.R. and Nilsen, H.M. [2012] Discussion of dynamics and operator splitting techniques for two-phase flow with gravity. Int. J Numer. Anal. Mod. (Special issue in memory of Magne Espedal), 9(3), 684-700.
- [9] 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, ISSN 0021-9991, doi:10.1016/j.jcp.2008.08.024. [10] Natvig, J.R. and Lie, K.A. [2008] On efficient implicit upwind schemes. *Proceedings of ECMOR XI, Bergen,*
- Norway, 8–11 September, EAGE.
- [11] Notay, Y. [2010] An aggregation-based algebraic multigrid method. Electron. Trans. Numer. Anal., 37, 123-140.
- [12] OPM [2012] The Open Porous Media (OPM) initiative. url: http://www.opm-project.org/.
- [13] Thiele, M.R., Batycky, R.P., Pöllitzer, S. and Clemens, T. [2010] Polymer-flood modeling using streamlines. SPE J., **13**(2), 313–322, doi:10.2118/115545-PA.
- [14] Wassmuth, F.R., Green, K., Arnold, W. and Cameron, N. [2009] Polymer flood application to improve heavy oil recovery at East Bodo. *J. Can. Petrol. Technol.*, **48**(2), 55–61, doi:10.2118/09-02-550. [15] Xiaodong, K. et al. [2011] A review of polymer EOR on offshore heavy oil field in Bohai Bay, China. *SPE*
- Enhanced Oil Recovery Conference, 19-21 July 2011, Kuala Lumpur, Malaysia.