A FAST MARCHING METHOD FOR 3D RESERVOIR SIMULATION

K. H. KARLSEN, K.–A. LIE, J. R. NATVIG, AND N. H. RISEBRO

1. Introduction

The motivation for doing reservoir simulation is to optimize the production of oil for different scenarios of enhanced recovery, such as water flooding, or polymer flooding. To give accurate predictions, one is faced with the problem of tracking the motion of the resident and injected fluid (i.e. oil and water). This motion is typically modelled by a coupled system of nonlinear equations in total pressure and fluid saturation; an elliptic equation for the pressure and a convection-diffusion equation for the saturation.

In many situations, the fluid motion from injection to production wells is dominated by convection. In such cases it is reasonable to neglect capillary pressure. The saturation equation then reduces to a nonlinear first order hyperbolic conservation law, which may develop discontinuous solutions even for smooth initial data. These discontinuities correspond to sharp interfaces between fluids, and are of great importance in determining how much and where the oil is in a reservoir. Moreover, exact predictions of water breakthrough in production wells are important to determine when separation equipment is needed.

In recent years, many methods have been devised to resolve these sharp fluid interfaces numerically. We present here the extension to three spatial dimensions of a novel method introduced by Karlsen, Lie and Risebro [1], based on the level set idea of Osher and Sethian [3] and the fast marching method of Sethian [4]. An important feature of the new method is that it has no inherent time step restriction; the length of each step can be chosen from accuracy considerations only.

2. The Mathematical Model

We consider a standard model for the flow of two immiscible fluids in a medium Ω with porosity ϕ and permeability K. Injection of one fluid and production of the other in different areas of the domain gives rise to a pressure gradient in the reservoir and a displacement of the two fluids. The relation between the flow rate and the pressure gradient in each phase is modelled by Darcy's law

$$v_i = -\lambda_i\big(\nabla P_i - \rho_i g \nabla D\big), \tag{1}$$

where i refers to the fluid phase (i.e., wetting $\mathbf{w}$ or non-wetting $\mathbf{n}$), g is the gravitational acceleration, D measures the vertical distance in the medium, P_i is the pressure, and ρ_i the mass density of phase i. The proportionality factor λ_i is called the mobility of phase i. In this paper, we assume that $P_{\mathbf{n}} = P_{\mathbf{w}} = P$, thereby neglecting capillary pressure. Furthermore, we assume incompressible medium and fluids. Conservation of mass for each fluid phase reads

$$\frac{\partial(\phi\rho_i S_i)}{\partial t} + \nabla\big(\rho_i v_i\big) = q_i, \tag{2}$$

where ϕ is the total available pore volume, S_i is the saturation of phase i, and q_i represents any sources or sinks for phase i. The saturation of phase i is the percentage of the available pore volume occupied by this phase, hence $S_n + S_w = 1$. Under these assumptions, the equation for the pressure reads

$$\nabla(\lambda_T \nabla P) + Q_T = \nabla\big[(\lambda_{\mathbf{n}}\rho_{\mathbf{n}} + \lambda_{\mathbf{w}}\rho_{\mathbf{w}}) g \nabla D\big], \tag{3}$$

where $\lambda_T = \lambda_{\mathbf{n}} + \lambda_{\mathbf{w}}$ is the total mobility. In the following, we will neglect gravity. By introducing the fractional flow function f_i defined by $\lambda_i = f_i(\lambda_{\mathbf{n}} + \lambda_{\mathbf{w}})$, we get a hyperbolic equation for the saturation

$$\phi\frac{\partial S_{\mathbf{n}}}{\partial t} + v_T \nabla f_{\mathbf{n}}(S_{\mathbf{n}}) = -q_{\mathbf{n}}, \tag{4}$$

with total Darcy velocity given by $v_T = -\lambda_T \nabla P$. In the following we will drop the subscripts in the saturation equation.

We are mainly concerned with the initial value problem given by (3)–(4) and the initial data

$$S_0(\mathbf{x}) = \begin{cases} 1, & \text{if } |\mathbf{x} - \mathbf{x}_w| < r_w \text{ and } q_{\mathbf{n}}(\mathbf{x}_w) > 0, \\ 0, & \text{otherwise}, \end{cases} \tag{5}$$

Date: October 3, 2000.

Key words and phrases. oil reservoir simulation, level set method, fast marching method, numerical examples.

for $\mathbf{x} \in \Omega$ and $r_w > 0$. This corresponds to injection of water ($\mathbf{n}$) into a reservoir fully saturated with oil ($\mathbf{w}$). On the boundary $\partial\Omega$ we impose homogeneous Neumann boundary conditions both for the pressure equation (3) and the saturation equation (4). To make a well-posed method, we assume that the fractional flow function is S-shaped, i.e., non-decreasing with one inflection point and $f(0) = 0$, $f(1) = 1$.

3. The Numerical Method

3.1. **Overall Strategy.** To solve the coupled system of differential equations (3)–(4), we use a standard operator splitting scheme, commonly referred to as IMPES (IMplicit Pressure, Explicit Saturation). The overall strategy is to advance the solution (S, P) in steps Δt to a final time $t_f = N\Delta t$. Assume that the solution (S^n, P^n) at time $t_n = n\Delta t$ has been computed. To advance the solution to t_{n+1} we use the following splitting strategy. First we solve the pressure equation (3) with the saturation S^n held fixed,

$$\nabla(\lambda_T(S^n)\nabla P) + Q_T = 0,$$

using a standard finite element method. We then compute the total Darcy velocity v_T^{n+1} from P^{n+1} using (1). Finally we solve the saturation equation

$$\phi\frac{\partial S}{\partial t} + v_T^{n+1}\nabla f(S) = -q,$$

with fixed v_T^{n+1}. Starting with a suitable initial condition, this splitting will compute the solution to any prescribed time $t_f = N\Delta t$.

3.2. **The Saturation Solver.** Under the assumptions made in the IMPES strategy, the saturation equation (4) reduces to a hyperbolic conservation law

$$\phi(\mathbf{x})\frac{\partial u}{\partial t} + \mathbf{v}(\mathbf{x})\nabla f(u) = 0. \tag{6}$$

Here u is the unknown function (saturation), ϕ is the porosity of the medium and $\mathbf{v}$ is a divergence free vector field. We observe that for initial value problems of (6), the information only flows in certain directions, i.e., along what is commonly referred to as streamlines. To exploit this, we reformulate the evolution equation as a boundary value problem for a stationary Eikonal equation. This boundary value problem can be solved very rapidly by employing the *fast marching method* [4].

Consider now the isosurface corresponding to a constant value $u(\mathbf{x}, 0) = k$. This isosurface will evolve in time as the solution $u(\mathbf{x}, t)$ evolves according to (6). Let $\Xi_k(t)$ be a level set of the solution $u(\mathbf{x}, t)$ in the sense that

$$\Xi_k(t) = \{\mathbf{x} | u(\mathbf{x}, t) = k\},$$

that is, $\Xi_k(t)$ is the isosurface corresponding to $u(\mathbf{x}, t) = k$ at any time t. Now, assume that $\mathbf{y}_0 \in \Xi_k(0)$ and $\mathbf{y} \in \Xi_k(t)$ are points on the same characteristic curve $(\mathbf{x}(\tau), t(\tau))$ described by the characteristic equations

$$\dot{\mathbf{x}}(\tau) = \mathbf{v}(\mathbf{x})f'(u(\mathbf{x}, t)), \qquad \mathbf{x}(0) = \mathbf{y}_0, \tag{7}$$

$$\dot{t}(\tau) = \phi(\mathbf{x}), \qquad t(0) = 0. \tag{8}$$

We know that $u(\mathbf{x}(\tau), t(\tau)) = k$ is constant on the characteristic curve. Then there exists a $\tau(\mathbf{y})$ such that $\mathbf{y} = \mathbf{x}(\tau(\mathbf{y}))$. Since u is constant on $\Xi_k(t)$, (7)-(8) imply that the isosurface moves with a (particle) speed given by

$$\frac{d\mathbf{x}}{dt} = \frac{\dot{\mathbf{x}}(\tau)}{\dot{t}(\tau)} = \mathbf{v}(\mathbf{x})\frac{f'(k)}{\phi(\mathbf{x})}. \tag{9}$$

Then, from (7)-(9) we may compute

$$\phi(\mathbf{y}) = \frac{dt(\tau)}{d\tau} = \nabla_{\mathbf{y}}\tau(\mathbf{y}) \cdot \frac{d\mathbf{y}}{d\tau} = \nabla_{\mathbf{y}}\tau(\mathbf{y}) \cdot \frac{d\mathbf{x}(\tau)}{d\tau}\Big|_{\mathbf{x}(\tau)=\mathbf{y}} = \nabla_{\mathbf{y}}\tau(\mathbf{y}) \cdot \mathbf{v}(\mathbf{y})f'(k). \tag{10}$$

Let $T_k(\mathbf{y}) = t(\tau(\mathbf{y}))$ be the time of arrival for the level set Ξ_k at $\mathbf{y}$. Then T_k is governed by (10). However, since $f'(k)$ is constant on the contour $\Xi_k(\cdot)$, we may as well solve the equation

$$\nabla T \cdot \mathbf{v}(\mathbf{x}) = \phi(\mathbf{x}). \tag{11}$$

Here we have defined $T = T_k f'(k)$. To supply the boundary conditions, we consider the initial data for the conservation law (6), i.e., $u_0(\mathbf{x}) = S_0(\mathbf{x})$. For the Eikonal equation this corresponds to boundary data of the form $\Xi_k(0) = \{\mathbf{x} |\, |\mathbf{x} - \mathbf{x}_w| = r_w,\, q_{\mathbf{n}}(\mathbf{x}_w) > 0\}$. In this case the solution of (6) is not smooth. We now claim that the mapping $s = T(\mathbf{x})$ renders the initial value problem for (6) on a simpler one-dimensional Riemann problem. To see this, we set $w(s,t) = w(T(\mathbf{x}),t) = u(\mathbf{x},t)$ and compute, using (11)

$$\begin{aligned} \frac{\partial}{\partial t}u(\mathbf{x},t) + \mathbf{v}\nabla f(u(\mathbf{x},t)) &= \frac{\partial}{\partial t}w(s,t) + \frac{\partial}{\partial s}f(w(s,t)) = 0, \\ u(\mathbf{x},0) = w(s,t) &= \begin{cases} 1, & s < 0, \\ 0, & s \geq 0. \end{cases} \end{aligned} \tag{12}$$

Due to the form of f, the solution will consist of a shock $(0,\bar{v})$ and a rarefaction wave $(\bar{v},1)$, where $\bar{v}$ satisfies $f'(\bar{v})\bar{v} = f(\bar{v})$. We can now write down the solution of the (equivalent) problems (12) and (6):

$$w(s,t) = \begin{cases} (\tilde{f}')^{-1}(\frac{s}{t}), & \text{if } s < t\tilde{f}'(s/t), \\ 0, & \text{otherwise}, \end{cases} \quad \text{or} \quad u(\mathbf{x},t) = \begin{cases} (\tilde{f}')^{-1}(\frac{T(\mathbf{x})}{t}), & T(\mathbf{x}) \leq t\tilde{f}'(T(\mathbf{x})/t), \\ 0, & \text{otherwise}. \end{cases}$$

Here $\tilde{f}'$ is the derivative of the upper convex envelope of f and $(\tilde{f}')^{-1}$ is its inverse. This gives an implicit formulae for computing the saturation at an arbitrary point, given the time-of-flight. For a more complete description of this background, see Karlsen, Lie and Risebro [1].

3.2.1. *The fast marching method.* In order to solve the Eikonal equation (11), we use the fast marching method of Sethian [4]. The basic idea here is that information only flows in one direction, along the characteristics. Therefore, given a point $\mathbf{x}$, the solution at $\mathbf{x}$ only depends on the solution at points $\mathbf{y}$ in a neighbourhood of $\mathbf{x}$ with $T(\mathbf{y}) < T(\mathbf{x})$. The strategy in the fast marching method is to use this *causality* to arrange the nodes in increasing order, such that the solution can be computed *explicitly* using an upwind difference formula. This ordering cannot be known in advance, so it is constructed dynamically. To this end, we store a selection of nodes $\mathcal{N}$that are candidates for the next update. We label these nodes as *narrow-band* nodes, while nodes that have already been visited are *alive* nodes; the rest are *faraway* nodes. The set of alive nodes and faraway nodes are denoted $\mathcal{A}$ and $\mathcal{F}$, respectively.

Assume that the solution has been constructed for all alive nodes. Then the next step in the solution procedure is to pick the node $n \in \mathcal{N}$ with the smallest T. We update the arrival time for n and then the time for each of its neighbours. The solution at $\mathbf{x}_n$ has now been computed; to update the sets $\mathcal{A}$, $\mathcal{N}$ and $\mathcal{F}$, we label n as an alive node, and all n's neighbours that are faraway nodes are relabelled as narrow-band nodes. To compute the solution up to a time t_f, we continue this process until all nodes in $\mathcal{N}$ have $T \geq t_f f'(\bar{v})$ (recall definition of T). The solution is then computed on $\mathcal{A}$, while $T(\mathcal{N} \cup \mathcal{F})$ is arbitrarily set to $t_f f'(\bar{v})$. (We do not compute the complete solution since we only need to know arrival times $T \leq t_f f'(\bar{v})$.)

By storing $\mathcal{N}$ in a min-heap, the node with smallest time are extracted in $\mathcal{O}(1)$ operations and the faraway nodes are inserted in the heap in $\mathcal{O}(\log|\mathcal{N}|)$ operations. If the total number of nodes in the grid is m^3, then $|\mathcal{N}| = \mathcal{O}(m^2)$ and the solution of (11) is computed in $\mathcal{O}(m^3 \log m)$ operations.

3.2.2. *Restarting.* To apply the IMPES time stepping strategy, we need to be able to handle a time varying (piecewise constant) velocity field. In principle, the causality we based the fast marching method on is lost. However, since the total Darcy velocity is piecewise constant in the IMPES splitting, we are still able to apply the fast marching idea on each time step. When the velocity changes, we need to update the time-of-flight in the whole grid. To do this, we initialize several narrow bands corresponding to different level sets, and march each outwards, starting with the outermost band. This way, an approximate causality is maintained. Assuming the velocity field does not change much in each step, as is normally the case, the error introduced is small.

Therefore, assume that we have stopped the fast marching algorithm at time t_n. Then the largest time-of-flight in the domain is $t = t_n f'(\bar{v})$. We wish to restart the algorithm with a different velocity field. Let $t = N\delta$. For each $1 \leq j \leq N$, $j\delta$ corresponds to a level set Ξ_{k_j} with $j\delta = t_n f'(k_j)$. (Thus $k_N = \bar{v}$.) Initially, we construct N narrow bands $\mathcal{N}_j$, with corresponding sets $\mathcal{A}_j$ and $\mathcal{F}_j$, all of which are defined by

$$\begin{aligned} \mathcal{A}_j &= \{n : T(\mathbf{x}_n) < t_n f'(k_j)\}, \\ \mathcal{N}_j &= \{n : T(\mathbf{x}_n) > t_n f'(k_j),\, \mathcal{S}_n \cap \mathcal{A}_j \neq \emptyset\}, \end{aligned}$$

and $\mathcal{F}_j = \Omega_h - (\mathcal{A}_j \cup \mathcal{N}_j)$, where $\mathcal{S}_n$ is the set of neighbours of n. To construct the solution for times greater than t_n, we apply the fast marching algorithm to each narrow band $\mathcal{N}_j$ in turn, starting with $\mathcal{N}_N$, then $\mathcal{N}_{N-1}$, etc.

The remaining question is: how do we determine when to stop each band $\mathcal{N}_j$ to compute the solution at a later time t_{n+1}? Since each band corresponds to a different saturation value k_j, the characteristic

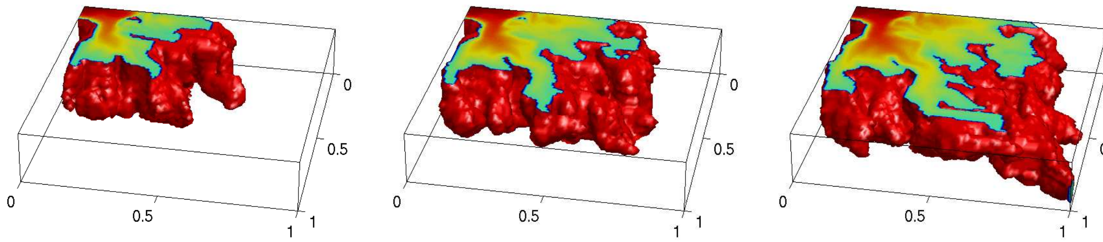

FIGURE 1. Example 1. Saturation component of the solution at time $t = 0.15$ (left), $t = 0.25$ (middle) and $t = 0.35$ (right). The level set (i.e. the front) in each figure is the actual oil-water front Ξ_0, while the shading on top show the saturation behind the oil-water interface.

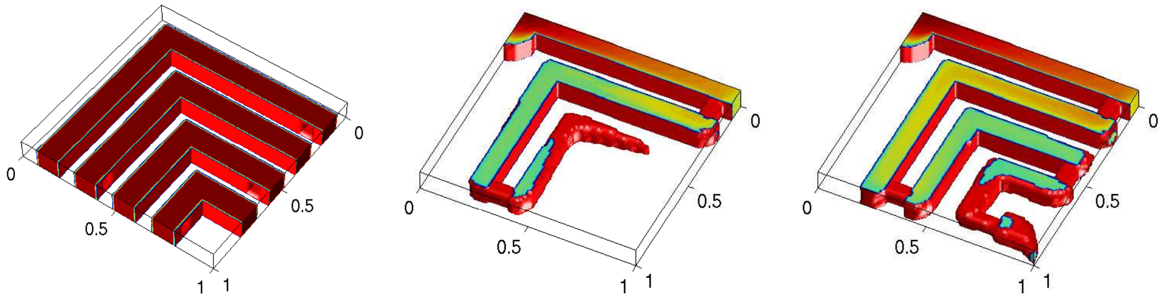

FIGURE 2. Example 2. The layered permeability field (left) and the saturation component of the solution of the artificial channel problem at time $t = 0.04$ (middle) and $t = 0.06$ (right).

speed $f'(k_j)$ is different for each band. Therefore it follows that the stopping time in the Eikonal equation (11) for band j is given by $t_{n+1}f'(k_j) = j\delta t_{n+1}/t_n$.

4. Numerical examples

4.1. **Example 1: Inhomogeneous Quarter-Five-Spot.** We consider a rectangular domain with dimensions $1.0 \times 1.0 \times 0.2$, partitioned uniformly in $100 \times 100 \times 20$ blocks. The mobilities of water and oil are $\lambda_w(S) = S^2$ and $\lambda_o(S) = 4(1-S)^2$ respectively; thus $f(S) = S^2/S^2 + 4(1-S)^2$.

Further, the permeability was generated with a log-normal distribution and subsequent smoothing. The result is a somewhat realistic-looking isotropic permeability field with $K_{11} = K_{22} = K_{33}$; the ratio between largest and smallest permeability is roughly 10^5.

The wells are vertical line sources; an injection well located at $(0,0,\cdot)$ and a production well at $(1,1,\cdot)$, both with intensities 0.2. The results are shown in Figure 1. We see that the structure of the oil-water interface, which is a shock, is well resolved.

4.2. **Example 2: Channel Problem.** The second test is an artificial channel problem. The case is designed to test how the method copes with water-oil fronts resulting from large jumps in the permeability. In reservoir simulation, such jumps occur in the presence of impermeable layers and faults.

The domain is rectangular with dimensions $1.0 \times 1.0 \times 0.1$. A series of Γ-shaped layers with low permeability are present in the slab, each having a small highly permeable "hole". In between these layers, the medium is highly permeable, see Figure 2. The ratio of the high and low permeabilities is 10^5 making the Γ-shaped regions virtually impermeable. The injector and productor are located at $(0,0,\cdot)$ and $(1,1,\cdot)$ respectively, with intensities 0.1. The available pore volume is $5.27 \cdot 10^{-2}$. Plot of saturation is given at two different times in Figure 2. The plots show that the fast marching method resolves the fluid interface accurately.

References

[1] K. H. Karlsen, K.-A. Lie and H. H Risebro. A Fast Marching Method for Reservoir Simulation. To appear in *Computational Geosciences*.

[2] M. S. Espedal, K. H. Karlsen. Numerical solution of reservoir flow models based on large time step operator splitting algorithms. In A. Fasano and H. van Dujin, editors, *Filtration in Porous Media and Industrial Applications*, Lecture Notes in Mathematics, Springer. To appear.

[3] S. Osher, J. A. Sethian. Fronts propagating with curvature dependent speed: algorithms based on Hamilton-Jacobi formulations. J.Comput.Phys., 79(1):12-49, 1988.

[4] J. A. Sethian. Level Set Methods and Fast Marching Methods. 2. edition. Cambridge University Press,1999.