SPE
Society of Petroleum Engineers

SPE 21230

Numerical Simulation of Two-Phase Flow Using Locally Refined Grids in Three-Space Dimensions

W A. Mulder and R.H.J. Gmelig Meyling, Koninklijke/Shell E&P Laboratorium

This paper was prepared for presentation at the 11th SPE Symposium on Reservoir Simulation held in Anaheim, California, February 17-20, 1991.

Abstract

This paper presents a numerical method for solving incompressible, immiscible, two-phase flow problems in three space dimensions. The grid consists of rectangular blocks. Blocks can iteratively be refined in areas with active wells, with steep saturation fronts, or with large permeability contrasts. The grid is dynamically adapted as the numerical simulation proceeds.

The equations governing two-phase flow are treated separately, which leads to an elliptic equation for pressure and velocity, and a parabolic equation for the water saturation. The solution technique combines mixed finite elements with the method of characteristics. Multigrid is used to solve pressure and velocity from the mixed finite element discretization. The water saturation is determined by a convection-dominated parabolic equation. Operator splitting allows a separate treatment of convection and diffusion.

The nonlinear convection problem is treated by the method of characteristics. A one-dimensional problem is formulated for each grid block by tracking along the stream line through the block center. The resulting convection problem is integrated in time by using co-moving co-ordinates. Nonlinear interactions are treated on the basis of Roe's approximate Riemann solver, which in the co-moving frame leads to insertion and deletion of points, and the recomputation of effective speeds. The resulting algorithm is quite simple, allows for arbitrary large time steps, and introduces almost no numerical diffusion.

The effect of gravity is incorporated by a fractional-step strategy, leading to another nonlinear convection problem. Capillary pressure is modeled as diffusion and treated implicitly by mixed finite elements and multigrid.

The solution method permits large time steps, if the flow field does not change substantially between time steps. Accurate results can be obtained with only a limited number of grid blocks. The method has been used to study both stable and unstable, immiscible displacement of oil by water. Results for some test problems are given.

Introduction

The objective of reservoir simulation is the computation of fluid flow through porous media with an accuracy that is sufficient for the prediction of the hydrocarbon recovery. The mathematical model for porous media flow consists of a coupled system of nonlinear partial differential equations that represents the basic physical properties of the true solution. Numerical methods are required to determine an approximate solution of the system.

Some difficulties associated with these numerical methods are: numerical diffusion, severe time step restrictions, and high computational complexity (see [2] for a survey). Certain fluid flow processes, such as viscous fingering, depend critically on the amount of physical diffusion. Computational techniques that introduce strong numerical diffusion and smear sharp fluid interfaces are therefore unacceptable. Stability conditions associated with some methods require very small time steps and lead to large computational costs. Many flow problems involve highly localized phenomena, such as sharp fluid interfaces, point or line sources and sinks, and geological heterogeneities. For large-scale simulations, it is impossible to accurately represent all the local phenomena by covering the entire domain with a uniform fine grid.

References and illustrations at end of paper.

In this paper, we present a numerical method for solving two-phase, immiscible flow problems. The method avoids the above mentioned computational difficulties through the combination of adaptive local grid refinement and special numerical techniques. Grid refinement can be carried out an arbitrary number of times in any region of space (see [3] for a survey of refinement strategies). The refinement is adaptive and dynamic, allowing fine blocks to move with fluid fronts. Static refinement is carried out near wells and near strong permeability contrast.

Operator splitting turns the mathematical model into a sequence of elliptic, hyperbolic, and parabolic problems. The elliptic and parabolic problem are discretized in space by the lowest order mixed finite element method of Raviart and Thomas [8]. This method provides a better accuracy near wells and strong heterogeneities than standard finite difference or finite element methods. The large linear system arising from the mixed finite element discretization is solved efficiently with multigrid [11]. The equation for the saturation has a convection and diffusion part. The diffusion models the effect of capillary pressure and is generally small compared to the convection term. The convection part is treated separately by transforming to stream line co-ordinates and integrating the one-dimensional nonlinear hyperbolic problem with the use of Riemann solvers. This characteristic technique has a very low numerical diffusion, captures sharp fronts, and allows for arbitrary large time steps. Here we use a slightly more general variant of the technique described in [4] and [5], employing Roe's approximate Riemann solver [9,10] rather than exact Riemann solvers.

The plan of the paper is as follows. First, we state the mathematical model for two-phase, immiscible flow. Secondly, the different aspects of the numerical method, namely operator splitting, mixed finite elements, multigrid, grid adaptation, convection, and diffusion, are treated. Finally, we present some numerical examples to demonstrate the capabilities of the method.

Governing equations

The flow of oil and water in a porous medium can be modeled by the following equations, derived in [1]:

$$\nabla.\mathbf{u} = q(\mathbf{x},t), \tag{1}$$

$$\mathbf{u} = -\mathcal{K}(\mathbf{x})\lambda(s)\left(\nabla p - \rho(s)\mathbf{g}\right), \tag{2}$$

$$\phi\frac{\partial s}{\partial t} + \nabla.\Big(f(s)\mathbf{u} + g(s)\mathbf{v} - D(s,\mathbf{x})\nabla s\Big) = c(s,\mathbf{x})q(\mathbf{x},t). \tag{3}$$

For reservoir simulation, we have nonzero $q(\mathbf{x},t)$ at the wells, with $\int_\Omega q(\mathbf{x},t)d\mathbf{x} = 0$. Boundary conditions will be specified further on. We choose a composition $c(s,\mathbf{x}) = f(s)$ at the wells.

The incompressibility condition (1) and Darcy's law (2) form an elliptic system of equations. The convection-diffusion equation (3) is of parabolic type. Without diffusion, it is a hyperbolic equation.

Numerical method

Operator splitting

The system of equations (1)-(3) is integrated in time by means of operator splitting. First, we solve a mixed finite element discretization of (1) and (2) with a multigrid solver [11]. Secondly, the convection part of (3) is treated by a nonlinear characteristic method. Thirdly, the diffusion part of (3) is discretized in time by a fully implicit scheme and in space by the mixed finite element method. The resulting system of equations is solved by multigrid and successive substitution (or Picard iteration).

We will describe the methods for convection and diffusion in more detail below. For the total flow $\mathbf{u}$ and pressure p, the lowest order mixed finite element representation of Raviart and Thomas [8] is used. The system (1)-(2) is solved with the multigrid method described in [11]. This method finds an approximate solution to

$$\begin{aligned} c\psi + \nabla.\mathbf{u} &= q, \\ \nabla\psi + \mathbf{W}\mathbf{u} &= \mathbf{g}, \end{aligned} \tag{4}$$

subject to the boundary condition

$$\mathbf{n}.\mathbf{u} = 0, \quad \text{on (part of) } \partial\Omega, \tag{5a}$$

or

$$\psi - \alpha\,\mathbf{n}.\mathbf{u} = \beta, \quad \text{on (part of) } \partial\Omega. \tag{5b}$$

Here $\mathbf{n}$ is the outward normal on the boundary $\partial\Omega$ of the domain Ω. The scalar quantities c and q are defined per block. The unknown scalar ψ is also defined per block, whereas the unknown vector $\mathbf{u}$ is defined on faces. Each face carries only the normal component of $\mathbf{u}$. The tensor $\mathbf{W}$ and vector $\mathbf{g}$ are specified per block. Equations (1) and (2) can be rewritten in the form (4) and solved by the multigrid method.

Once the total flow $\mathbf{u}$ at a given time is computed for a saturation distribution s, the convection-diffusion equation (3) is solved with a frozen flow field. Operator splitting is applied. First, we solve two convection equations of the form

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

and

$$\phi\frac{\partial s}{\partial t} + \nabla.g(s)\mathbf{v} = 0. \tag{6b}$$

The order in which this is done is alternated at every time step. Next, the source term $c(s,\mathbf{x})q(\mathbf{x},t)$ is treated. Finally, diffusion is taken into account by considering

$$\phi\frac{\partial s}{\partial t} - \nabla.(D(s,\mathbf{x})\nabla s) = 0. \tag{7}$$

Grid adaptation

For problems with different length scales, adaptive local grid refinement is highly desirable. We employ the approach described in [12], which allows for dynamic regridding and is geared towards the mixed finite element method and the multigrid solver.

A grid consists of *blocks* on which scalar quantities are defined, separated by *faces*, on which fluxes are stored. Each face carries only the normal component of the flux. The subroutines in our code are organized in such a way that the dimension can be chosen as 1, 2, or 3. A hierarchical grid structure is employed. We start with a static rectangular base grid created by orthogonal "planes" (constant value of a space co-ordinate). The rectangular boxes are called *blocks*, and the separating "planes" *faces*. The base grid can be refined and unrefined by the *basic refinement*, which divides a block into 2^{N_d} identical smaller parts. Our code uses a cartesian co-ordinate system. The extension to curvi-linear co-ordinates is straightforward, but has not been carried out so far.

The grid is statically refined near wells. Dynamic grid adaptation is used for the saturation s and the flow field $\mathbf{u}$. The regridding is controlled by the requirement

$$tol_{min}^{(m)} \leq \varepsilon^{(m)} \leq tol_{max}^{(m)}, \tag{8}$$

where the error indicator $\varepsilon^{(m)}$ is determined from numerical approximations to the following quantities:

$$\begin{aligned} &h_{i\,k}|\tfrac{\partial}{\partial x_k}(\phi s)|_i, && \text{for } m=1,\\ &h_{i\,k}|\tfrac{\partial^2}{\partial x_k^2}(\phi s)|_i, && \text{for } m=2,\\ &h_{i\,k}|\mathcal{K}^{-1}\tfrac{\partial}{\partial x_k}\mathcal{K}|_i, && \text{for } m=3,\\ &a_j|u_j|, && \text{for } m=4, \end{aligned} \tag{9}$$

and the maximum is taken over $k = 1, \ldots, N_d$. Here $h_{i\,k}$ is the width of a block i in the k^{th} co-ordinate direction, a_j the area of a face, and u_j the normal component of the flow stored at face j. If the upper (lower) bound of (8) is violated, the block is refined (coarsened).

The refinement based on the first and second derivatives of the saturation, is carried out iteratively during a time step. New saturation values are computed, the grid is refined where necessary, new saturations are computed again, etc. The number of iterations does not exceed the maximum number of levels, as refinement can only be carried out to one level finer. At the end of the time step, refinement is suppressed and only coarsening is carried out to remove the redundant refinements caused by the old location of the saturation front.

Convection

We will concentrate on the numerical solution of (6a). First, we transform this equation to the stream line co-ordinate ξ, defined by

$$d\mathbf{x} = \frac{\mathbf{u}}{\phi} d\xi. \tag{10}$$

The transformed equation is

$$\frac{\partial s}{\partial t} + \frac{\partial f(s)}{\partial \xi} = 0. \tag{11}$$

Because the normal components of $\mathbf{u}$ are given on cell faces and vary linearly between opposing faces, we have $u_k = u_k(x_k)$ as a linear function in x_k. The system (10) decouples in the co-ordinates and each equation can be solved analytically per block.

For each block, we start at the block center and track backward and forward along the stream line, integrating (10) per block. In this way, a one-dimensional list of saturation values s_j is obtained, as found in each block intersected by the stream line. These values are defined on intervals $(\xi_{j-1/2}, \xi_{j+1/2})$, with each $\xi_{j+1/2}$ corresponding to a block face in the original domain. If the index $j = 0$ is assigned to the original block, the minimum and maximum values of ξ must obey

$$\begin{aligned} &\xi_{min} \leq -(t^{n+1} - t^n)\max_s f'(s), && \xi_{min} \leq \xi_{0-1/2},\\ &\xi_{max} \geq -(t^{n+1} - t^n)\min_s f'(s), && \xi_{max} \geq \xi_{0+1/2}. \end{aligned} \tag{12}$$

Given this list of saturation values s_j^n, $(j = j_{min}, \ldots, j_{max})$, at time t^n that are piecewise constant in cells separated by faces $\xi_{j-1/2}$, $(j = j_{min}, \ldots, j_{max}+1)$, we have to determine a new saturation value s_0^{n+1} in cell $j = 0$ at time t^{n+1}. We choose the solution method in [7], which is based on co-moving co-ordinates and Roe's approximate Riemann solver [9,10]. (Methods based on co-moving co-ordinates with a projection step to the original grid are also known as Euler-Lagrange or Lagrangian-Eulerian methods.) In [7], a system of hyperbolic equations is considered. Our scalar case is much simpler. We extend the method by a different treatment of shocks and expansion fans, to allow for larger time steps and a better resolution of the fans.

Before going to co-moving co-ordinates, we review Roe's scheme. Define $\Delta_j h = h_{j+1/2} - h_{j-1/2}$ for any quantity h, and let $\Delta t = t^{n+1} - t^n$. An explicit time discretization of (11) leads to

$$\frac{s_j^{n+1} - s_j^n}{\Delta t} = -\frac{\Delta_j f}{\Delta_j \xi}. \tag{13}$$

Here $f_{j+1/2}$ is a numerical flux depending on s_j and s_{j+1}. The exact solution of the *Riemann problem*

$$s = \begin{cases} s_j, & \text{for } \xi < \xi_{j+1/2},\\ s_{j+1}, & \text{for } \xi > \xi_{j+1/2}, \end{cases} \tag{14}$$

at $t = t^n$, yields a unique state $s_{j+1/2} = s(\xi_{j+1/2}, t)$ for $t > t^n$ that can be used to evaluate $f_{j+1/2} = f(s_{j+1/2})$. Roe's scheme uses an *approximate* solution of the Riemann problem by setting

$$f_{j+1/2} = \tfrac{1}{2}\left[f_j + f_{j+1} - |\hat{a}_{j+1/2}|(s_{j+1} - s_j)\right]. \tag{15}$$

Here $f_j = f(s_j)$ and the Roe speed

$$\hat{a}_{j+1/2} = \begin{cases} \frac{f_{j+1} - f_j}{s_{j+1} - s_j}, & \text{if } s_{j+1} \neq s_j,\\ f'(s_j), & \text{if } s_{j+1} = s_j. \end{cases} \tag{16}$$

Eq.(15) can be interpreted as a central difference scheme with an added viscosity term proportional to $|\hat{a}_{j+1/2}|$. Another interpretation is obtained by defining

$$a^+ = \max(0, a), \quad a^- = \min(0, a). \tag{17}$$

Then

$$\Delta_j f = \hat{a}^+_{j-1/2}(s_j - s_{j-1}) + \hat{a}^-_{j+1/2}(s_{j+1} - s_j), \qquad (18)$$

which demonstrates the upwind character of the scheme. The two terms on the right-hand side can be viewed as *linear waves*, one moving to the right from $\xi_{j-1/2}$ with speed $\hat{a}^+_{j-1/2}$ and strength $|s_j - s_{j-1}|$, and one moving to the left from $\xi_{j+1/2}$ with speed $\hat{a}^-_{j+1/2}$ and strength $|s_{j+1} - s_j|$. The *nonlinearity* of the problem is accounted for in the computation of the speed $\hat{a}_{j+1/2}$.

The explicit scheme (13) can be rewritten as

$$\begin{aligned} s^{n+1}_j = &\Big[1 - \frac{\Delta t}{\Delta_j \xi}(\hat{a}^+_{j-1/2} - \hat{a}^-_{j+1/2})\Big] s^n_j \\ &+ \Big[\frac{\Delta t}{\Delta_j \xi}\hat{a}^+_{j-1/2}\Big] s^n_{j-1} - \Big[\frac{\Delta t}{\Delta_j \xi}\hat{a}^-_{j+1/2}\Big] s^n_{j+1}. \end{aligned} \qquad (19)$$

It is *positive* for

$$\frac{\Delta t}{\Delta_j \xi}(\hat{a}^+_{j-1/2} - \hat{a}^-_{j+1/2}) \le 1. \qquad (20)$$

Positivity implies stability in ℓ_∞.

We will not use the method in this form, because of the severe time step restriction (20). Instead, we transform to co-moving co-ordinates by letting each cell face $\xi_{j+1/2}$ move with a speed $\hat{a}_{j+1/2}$. The transformation to co-moving co-ordinates (ζ, τ) is given by

$$\xi_{j+1/2} = \zeta_{j+1/2} + \int_{t^n}^{\tau} \hat{a}_{j+1/2} d\tau', \qquad (21)$$

where $\hat{a}_{j+1/2}$ is computed from s_j and s_{j+1} according to (16). Note that the subscripts now refer to cells in the co-moving frame. For small $\Delta\tau = \tau - t^n$, the Roe speed in the transformed co-ordinates approximates $f(s) - sf'(s)$ and is zero. This implies that s_j is constant, as long as the cell size does not vanish.

The discrete result is consistent with the well-known fact that *s is constant on characteristics*, but is more general: it holds even in the case of shocks, until the moment that the cell size $\Delta_j\xi = \Delta_j\zeta + \Delta\tau\,\Delta_j\hat{a}$ vanishes. If that happens earlier than time t^{n+1}, we integrate up to the instant where the cell size of a cell j vanishes, and then remove this cell from our list of saturations. After this has been done, the Riemann problem between states s_{j-1} and s_{j+1} of the original list has to be solved, which is done approximately by recomputing $\hat{a}(s_{j-1}, s_{j+1})$ at $\xi_{j-1/2} = \xi_{j+1/2}$. In this way, the computation can be continued up to an arbitrary large time t^{n+1} in quite an efficient manner.

However, there is one problem which is caused by a fundamental flaw in Roe's approximate Riemann solver. Roe's scheme allows for unphysical expansion shocks. An initial discontinuity that should break up into an expansion fan, remains as it is. A costumary remedy [6] is to add a small amount of extra dissipation into (15) by replacing $|\hat{a}_{j+1/2}|$ with $h(\hat{a}_{j+1/2})$, where

$$h(a) = \begin{cases} |a|, & \text{if } |a| \ge \varepsilon_a, \\ (a^2 + \varepsilon_a^2)/2\varepsilon_a, & \text{if } |a| < \varepsilon_a. \end{cases} \qquad (22)$$

Here ε_a is a user-specified constant. In our case, this would imply that s_j is no longer constant, so we choose another remedy.

If an expansion fan occurs between two neighboring states s_j and s_{j+1}, extra saturation values are inserted at the face $\xi_{j+1/2}$ using a fixed spacing δs, which can be user-specified and is typically around 0.01. A new value s_i is inserted *only* if the resulting $\Delta_i\hat{a}$ is *strictly* positive. This excludes saturation values that lead to compression or linear convection with constant speed.

The actual subroutine that performs this insertion is somewhat complicated. To accelerate the process, a precomputed table with saturation values between 0 and 1 at increments of δs, and their corresponding fluxes and speeds, is used. The method works for any C^1 flux function $f(s)$.

A short-coming of the approach outlined so far, is it's tendency to produce "staircases" after a number of time steps. The non-smoothness is a direct result of the piecewise constant representation of saturation values and is also noted in [7]. The virtual absence of numerical dissipation causes the initial piecewise constant representation to persist, which does not "look nice", but actually is consistent with the first-order accuracy of the scheme. The effect of "staircases" is reduced by going to a second-order scheme, using a linear distribution of the data injected into expansion fans. This linear distrubution is forced to lie between the cell centers of the orginal two cells, and preserves mass. Details are given in the Appendix.

Once the above techniques have been used to integrate the one-dimensional data from t^n to t^{n+1}, the resulting saturations s^{n+1}_j are projected back to the original cell $(\zeta_{0-1/2}, \zeta_{0+1/2})$ according to

$$\bar{s}^{n+1}_0 = \frac{1}{\Delta_0\zeta}\int_{\zeta_{0-1/2}}^{\zeta_{0+1/2}} s(\xi, t^{n+1})\, d\xi, \qquad (23)$$

where the subscript 0 now refers to the original co-ordinate system and $s(\xi, t^{n+1})$ denotes the piecewise constant values s^{n+1}_j in the cells of the co-moving frame. The result $\bar{s}^{n+1}_0$ becomes the new saturation in the original block of the multi-dimensional problem. The method is conservative in one dimension, but not in more than one.

Diffusion

The effect of capillary pressure is modeled by diffusion. We discretize the equation

$$\phi\frac{\partial s}{\partial t} = \nabla\cdot(D(s)\nabla s), \qquad (24)$$

in time by

$$\phi \frac{s^{n+1} - \tilde{s}^{n+1}}{\Delta t} = \nabla . \left(D(s^{n+1}) \nabla s^{n+1} \right), \qquad (25)$$

where $\tilde{s}^{n+1}$ is the result obtained after convection. In the absence of convection, we simply have $\tilde{s}^{n+1} = s^n$. The spatial part of (24) is discretized by the lowest order mixed finite element method of [8]. We obtain a system of the form (4):

$$\begin{aligned} \tfrac{\phi}{\Delta t} s^{n+1} + \quad \nabla . \mathbf{F} \quad &= \tfrac{\phi}{\Delta t} \tilde{s}^{n+1}, \\ \nabla s^{n+1} + \left[D^{n+1} \right]^{-1} \mathbf{F} &= 0. \end{aligned} \qquad (26)$$

The nonlinear diffusion term D^{n+1} is treated by using successive substitution (or Picard iteration). The linear system (26) is solved with multigrid. We use $\tilde{s}^{n+1}$ as initial guess for D^{n+1}, solve the linear system with multigrid, substitute the resulting s^{n+1} in D^{n+1}, and continue to repeat this for a number of times.

Time step control

The total flow $\mathbf{u}$ is kept constant during a time step with the convection scheme. Accurate results can be obtained with large time steps only if the total flow remains fairly constant in time. This is true for stable displacement with small or negligible capillary pressure and no gravity. For unstable displacement, $\mathbf{u}$ varies strongly with the saturation and large time steps cannot be taken. The same is true in the presence of gravity, especially if gravity acts perpendicular to the overall flow direction. We therefore relate the time step Δt to the relative change of the total flow $\mathbf{u}$ over a time step, in order to keep this change within reasonable bounds.

In the presence of gravity, very small time steps are required if gravity acts in a direction perpendicular to the front. In that case, the total flow changes direction abruptly across the front, and any change in the position of the front results in a large change of the total flow field. Under these circumstances, it is better to use an intermediate value of the total flow field. A predictor-corrector approach is employed. Given $\mathbf{u}^n$ and s^n at time t^n, we perform convection and diffusion steps to determine s^{n+1} at time $t^{n+1} = t^n + \Delta t$. Next, the new total flow $\mathbf{u}^{n+1}$ is computed, and the convection and diffusion steps are repeated with $\frac{1}{2}(\mathbf{u}^n + \mathbf{u}^{n+1})$. This approach allows us to take substantially larger time steps.

Numerical examples

We present a number of numerical examples to illustrate the capabilities of the method. The fluid, oil and water, are assumed to be incompressible, as is the rock formation.

Unstable displacement

A two-dimensional rectangular domain with constant absolute permeability and porosity has inflow of water from the left and outflow of oil at the right side. Initial conditions are shown in Fig. 1a. We choose an S-shaped fractional flow curve that leads to unstable displacement. The endpoint mobility ratio is 7.262 and the mobility ratio across the shock is 2.051. Capillary pressure is included. Gravity is absent.

The result after 41 time steps is shown in Fig. 1b, using variable time steps. The amplification of the initial perturbation, caused by the unstable displacement process, is obvious. The capillary pressure suppresses small-scale instabilities (cf. [4]).

Cusping

A three-dimensional rectangular domain $[0,4]\times[0,1]\times[0,\frac{1}{2}]$ has a constant permeability and porosity. Water is injected at the origin of the 15° updip reservoir. Oil is produced at $(4,0,\frac{1}{2})$. The computation includes gravity and capillary pressure. Figure 2 shows the result at 0.25 PVI after 4 time steps for unstable displacement with the same fractional flow curve as in the first example. The result for stable displacement is shown in Fig. 3. The fractional flow curve used here has an endpoint mobility ratio of 2, whereas this ratio across the shock is 0.845.

Conclusions

A numerical method has been presented for accurately solving the problem of two-phase, immiscible flow through porous media. The method is based on truely adaptive local grid refinement. A mixed finite element discretisation and multigrid are used to solve elliptic problems. Nonlinear hyperbolic problems are treated by the method of characteristics and approximate Riemann solvers.

The method introduces almost no numerical diffusion and can be used with large time steps if the physics of the problem allow this. The method has been applied to stable and unstable displacement problems in two and three dimensions.

Nomenclature

$c(s,\mathbf{x})$	fluid composition at the wells
$\mathcal{D}(s) = g(s)\frac{dp_c}{ds}\mathcal{K}(\mathbf{x})$	diffusion coefficient
$f(s) = \lambda_w/\lambda$	fractional flow curve (flux)
$g(s) = \lambda_o f(s)$	gravity flux
$\mathbf{g}$	gravity acceleration vector
$\mathcal{K}(\mathbf{x})$	absolute permeability
$k_o(s), k_w(s)$	relative permeability of oil, water
$p_c(s)$	capillary pressure
p	total pressure
$q(\mathbf{x},t)$	source term, nonzero at wells
$s_w = s$	reduced water saturation
$s_o = 1 - s$	reduced oil saturation
t	time, $t \in [0,T]$
$\mathbf{u}_o$, $\mathbf{u}_w$	oil and water flow, respectively
$\mathbf{u} = \mathbf{u}_o + \mathbf{u}_w$	total flow
$\mathbf{v} = (\rho_w - \rho_o)\mathcal{K}(\mathbf{x})\mathbf{g}$	gravity flow
$\mathbf{x}$	position vector, $\mathbf{x} \in \Omega$
$\lambda_o(s) = k_o/\mu_o$	relative mobility of oil
$\lambda_w(s) = k_w/\mu_w$	relative mobility of water
$\lambda(s) = \lambda_o + \lambda_w$	total mobility
μ_o, μ_w	viscosity of oil, water
ρ_o, ρ_w	density of oil, water
$\rho(s) = (\rho_o\lambda_o + \rho_w\lambda_w)/\lambda$	average fluid density
$\phi(\mathbf{x})$	porosity
Ω, $\partial\Omega$	reservoir domain and boundary

Acknowledgements

The authors are indebted to the management of Shell Internationale Research Maatschappij BV for permission to publish this paper. They thank Geert Schmidt for providing the multigrid and adaptive gridding routines, and for many stimulating discussions.

References

1. Chavent, G., Cohen, G., and Jaffre, J.: "Discontinuous Upwinding and Mixed Finite Elements for Two-Phase Flows in Reservoir Simulation," *Computer Methods in Appl. Mechanics and Eng.* **47** (1984) 93–118.
2. Ewing, R. E.: "Problems Arising in the Modeling of Processes for Hydrocarbon Recovery," in *The Mathematics of Reservoir Simulation*, R. E. Ewing (ed.), Frontiers in Applied Mathematics, SIAM, Philadelphia (1983), 3–34.
3. Ewing, R. E.: "Adaptive Grid-Refinement Techniques For Treating Singularities, Heterogeneities, and Dispersion," in *Numerical Simulation in Oil Recovery*, M. F. Wheeler (ed.), Springer-Verlag, Nev' York/Berlin (1988) 133–148.
4. Gmelig Meyling, R. H. J., Mulder, W. A., and Schmidt, G. H.: "Porous Media Flow on Locally Refined Grids," in *Proceedings of the Workshop on Numerical Methods for the Simulation of Multi-Phase and Complex Flow*, T. Verheggen (ed.), Springer-Verlag, New York/Berlin (1990), to appear.
5. Gmelig Meyling, R. H. J.: "Numerical Methods for Solving the Nonlinear Hyperbolic Equations of Porous Media Flow," in *Proceedings of the Third International Conference on Hyperbolic Problems*, Uppsala, June 1990.
6. A. Harten, : "High Resolution Schemes for Hyperbolic Conservation Laws," *J. Comput. Phys.* **49** (1983) 357-393.
7. Harten, A., and Hyman, J. M.: " Self Adjusting Grid Methods for One-Dimensional Hyperbolic Conservation Laws," *J. Comput. Phys.* **50** (1983) 235–269.
8. Raviart, P. A., and Thomas, J. M.: "A Mixed Finite Element Method for Second Order Elliptic Problems," in *Mathematical Aspects of Finite Element Methods*, Lecture Notes in Mathematics Vol. 606, Springer-Verlag, New York/Berlin (1977) 292–302.
9. Roe, P. L.: "Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes," *J. Comput. Phys.* **43** (1981) 357–372.
10. Roe, P. L.: "Characteristic-Based Schemes for the Euler Equations," *Ann. Rev. Fluid Mech.* **18** (1986) 337–365.
11. Schmidt, G. H., and Jacobs, F. J.: "Adaptive Local Grid Refinement and Multi-grid in Numerical Reservoir Simulation," *J. Comput. Phys.* **77** (1988) 140–165.
12. Schmidt, G. H.: "A Dynamic Grid Generator and Multigrid for Numerical Fluid Dynamics," in *Proceedings of the 8^{th} GAMM Conference on Numerical Methods in Fluid Mechanics*, P. Wesseling (ed.), Notes in Numerical Fluid Mechanics Vol. 29, Vieweg, Braunschweig (1990) 493–502.

Appendix. Insertion in expansion fans

To avoid "staircases" in the solution, extra points inserted at the face between two cells j and $j+1$ are spread over a finite interval between the two cell centers. Here we describe the algorithm.

Suppose that N new points have been injected at $\xi_{j+1/2}$, the face between cells j and $j+1$. New indices $i = 0, \ldots, N+1$ are introduced, with $s_{i=0} = s_j$ and $s_{i=N+1} = s_{j+1}$. Let $\xi_{i=0} = \frac{1}{2}(\xi_{j-1/2} + \xi_{j+1/2})$ be the center of the original cell j, and $\xi_{i=N+1} = \frac{1}{2}(\xi_{j+1/2} + \xi_{j+3/2})$ the center of $j+1$. The new cells $i = 1, \ldots, N$ must have sizes $\Delta_i\xi$ such that

$$\begin{aligned} &\Delta_i\xi = \alpha(s_{i+1} - s_{i-1}), \quad i = 1, \ldots, N, \\ &\sum_{i=0}^{N+1} \Delta_i\xi = \xi_{N+1} - \xi_0 \equiv L, \\ &\sum_{i=0}^{N+1} s_i\Delta_i\xi = (\xi_{j+1/2} - \xi_0)s_0 + (\xi_{N+1} - \xi_{j+1/2})s_{N+1} \equiv M. \end{aligned} \tag{A-1}$$

The first equation describes the linear distribution of saturations on terms of ξ, the second places this distribution between ξ_0 and ξ_{N+1}, the original cell centers, and the third imposes conservation. Two nonstandard quantities in the above are

$$\begin{aligned} \Delta_0\xi &\equiv \xi_{1/2} - \xi_0 \equiv h_0, \\ \Delta_{N+1}\xi &\equiv \xi_{N+1} - \xi_{N+1/2} \equiv h_{N+1}. \end{aligned} \tag{A-2}$$

The system (A-1) has no unique solution, and $\alpha = 0$ is always a solution. With the definitions

$$\begin{aligned} &F \equiv \frac{s_N s_{N+1} - s_0 s_1}{\sum_{i=1}^{N}(s_{i+1} - s_{i-1})}, \quad G \equiv M - FL, \\ &F_0 \equiv s_0 - F, \quad F_{N+1} = s_{N+1} - F, \end{aligned} \tag{A-3}$$

the system (A-1) reduces to

$$F_0 h_0 + F_{N+1} h_{N+1} = G. \tag{A-4}$$

We pick the solution that minimizes h_0 and h_{N+1}, subject to $h_0 \geq 0$ and $h_{N+1} \geq 0$. Without the bounds, the solution is

$$h_0 = \frac{GF_0}{F_0^2 + F_{N+1}^2}, \quad h_{N+1} = \frac{GF_{N+1}}{F_0^2 + F_{N+1}^2}. \tag{A-5}$$

If this violates the bounds, either $h_0 = 0$ or $h_{N+1} = 0$ is substituted into (A-4) to find the other h_{N+1} or h_0, respectively. Next, we let

$$\alpha = \frac{L - h_0 - h_{N+1}}{\sum_{i=1}^{N}(s_{i+1} - s_{i-1})}, \tag{A-6}$$

and compute the new $\xi_{i+1/2}$.

The entire procedure is suppressed (implying $\alpha = 0$), if the fractional flow curve $f(s)$ gives rise to a shock between s_j and s_{j+1}, or if either state is close to 0 or 1 (within $\frac{1}{2}\delta s$).

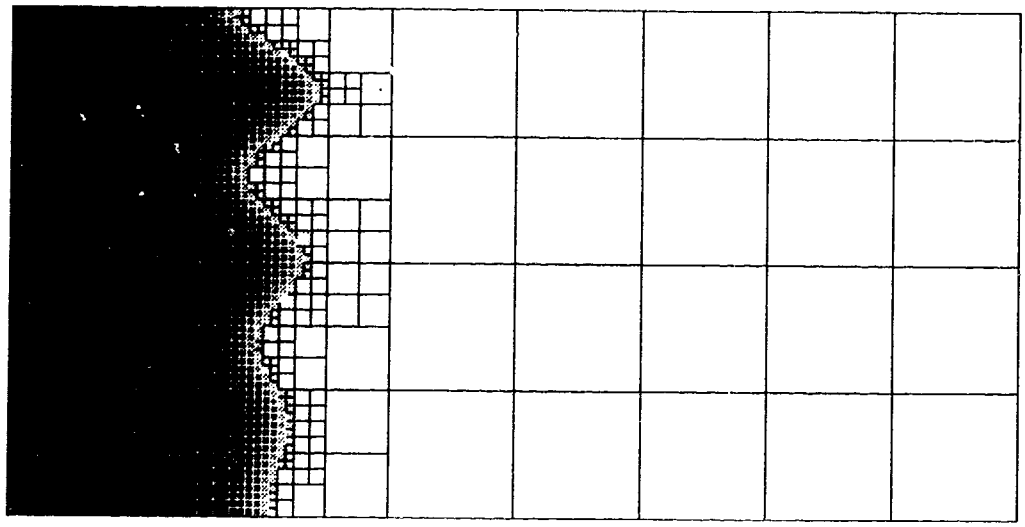


Fig. 1a. Initial conditions for the unstable displacement problem. Darker halftones correspond to higher water saturations.

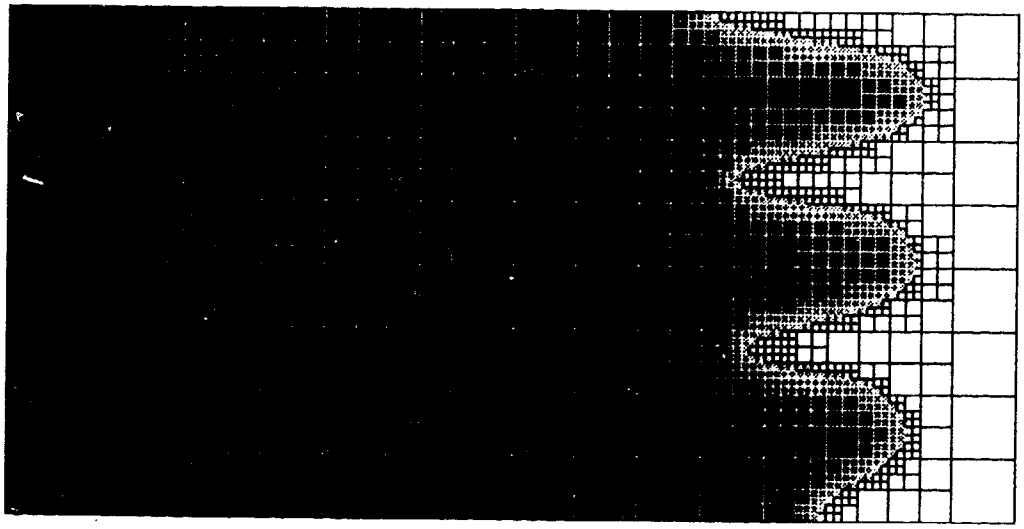
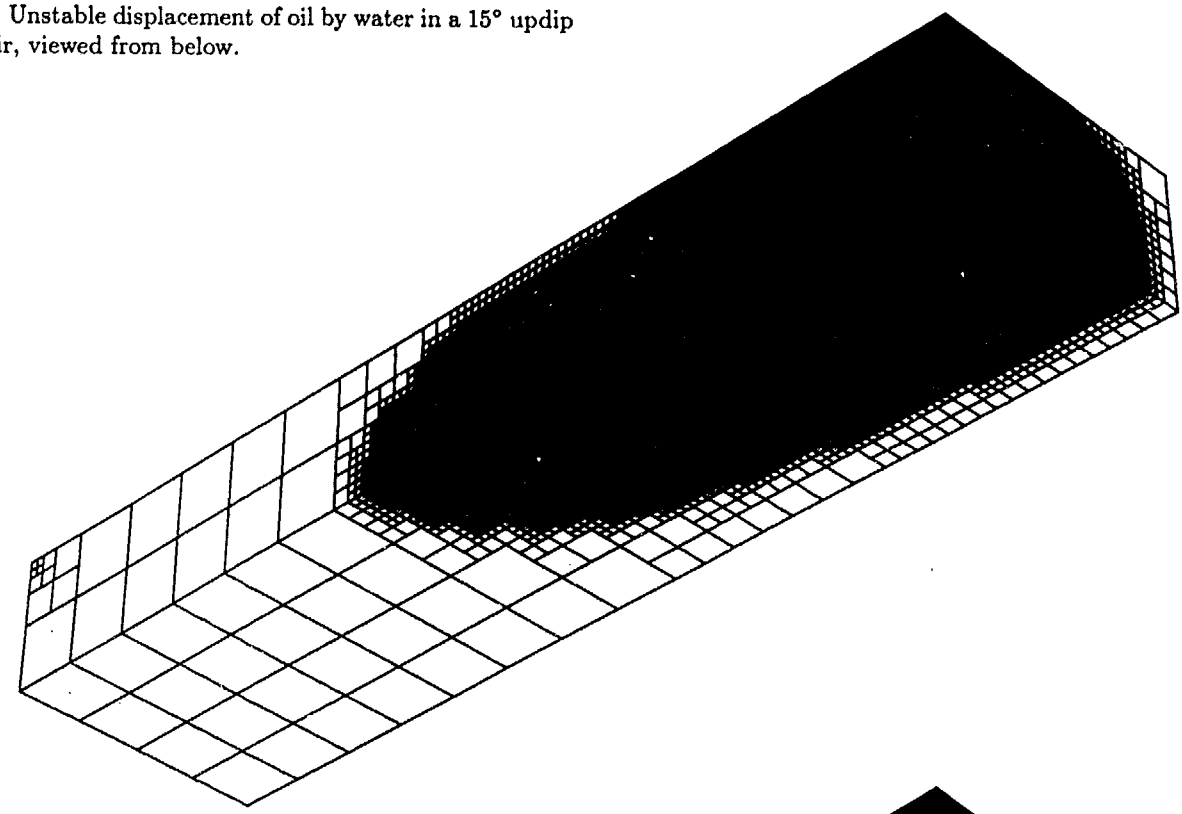


Fig. 1b. Solution at later time, showing the growth of fingers.

Fig. 2. Unstable displacement of oil by water in a 15° updip reservoir, viewed from below.



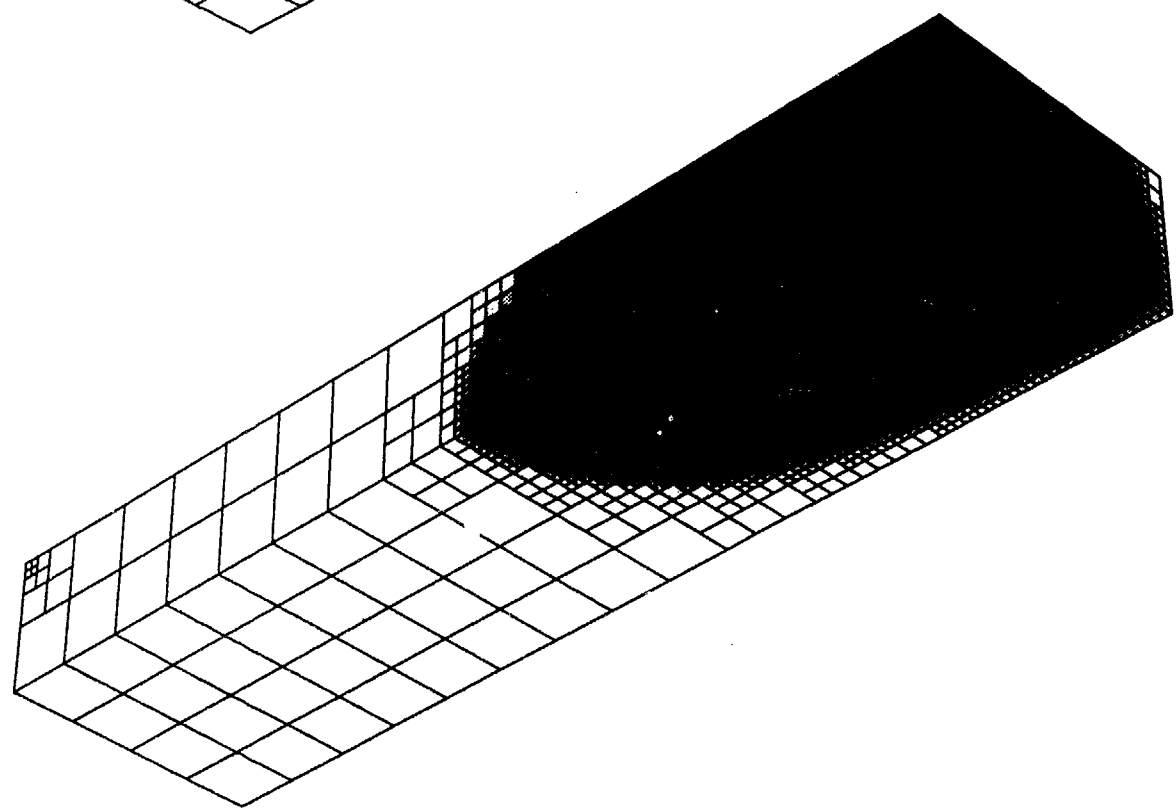


Fig. 3. Stable displacement of oil by water in a 15° updip reservoir, viewed from below.