SPE 119099

Front-Tracking Methods for Use in Streamline Simulation of Compressible Flow

Halvor Møll Nilsen and Knut–Andreas Lie, SINTEF

Abstract

Streamline methods are gaining popularity in the industry by providing fast desktop simulation of large reservoir models or multiple realizations. Traditionally, streamline simulation has been associated with simplified physics, but recent advances have demonstrated its potential also for compressible three-phase or component flows. However, streamline simulation is still most efficient for two-phase incompressible flow, for which one can utilize a particularly efficient front-tracking method to solve 1-D transport equations along streamlines that is unconditionally stable and independent of the strongly irregular time-of-flight grid. In a recent paper (Nilsen and Lie 2008), we presented, for the first time, front-tracking methods for simulating 1-D *compressible* two-phase flow. We also developed two methods that were particularly efficient for solving compressible flow in which one phase is incompressible, motivated by the simulation of CO_2 injection. Here we apply these methods to streamline simulation of 3-D models, including a real-life model of a North Sea formation, which is under consideration as a potential target for CO_2 deposition. Our numerical results demonstrate that streamlines and front tracking together give very efficient simulation of compressible flow. Similar ideas can also be applied for dual-porosity models, but this is not investigated in great detail herein.

Introduction

Fast and robust methods are crucial for modeling and simulating of flow in porous media, in particular because many of the physical and geological parameters are uncertain or unknown and one needs to simulate a large set of equiprobable realizations. Streamline methods (Datta-Gupta and King 2007) are well established as an effcient alternative to traditional finite-volume methods for simulation of (large) models with complex geology for which flow patterns are dictated by heterogeneity in rock parameters and position and rates of wells. Whereas early streamline methods were only applicable to models with simple flow physics, like incompressible two-phase flow without gravity or capillary forces, current streamline technology can include gravity and capillary effects and be applied to compositional, compressible three-phase black-oil, and dual-porosity models. As a result, streamline simulation is gaining (renewed) popularity in areas such as uncertainty quantification, upscaling, history matching, closed-loop management, production optimization, risk assessment of CO_2 deposition, etc.

The efficiency of streamline simulation can basically be attributed to two factors (Thiele 2005): memory efficiency and computational efficiency. Streamline simulation is memory efficient because it uses a sequential splitting to separate the solution of flow and transport so that only the flow equations (for pressure and velocities) need to be solved implicitly over the global grid. This drastically reduces the size of the discrete (non)linear problem and the memory requirements of the corresponding linear solver. The transport along streamlines are independent and can be computed in parallel or sequentially, keeping only a single streamline in memory at a time. The computational efficiency comes from three factors: (i) the number of streamlines required to achieve good accuracy typically scales linearly with the size of a vectical slice; (ii) in many applications, one can use relatively long pressure steps and hence update the streamline distribution infrequently; and (iii) it is more efficient to solve (reduced) 1-D transport problems along streamlines than solving the full 3-D transport problem. In this article, we will focus on two challenges related to the last two points: namely, how to make an efficient 1-D solver for the highly irregular time-of-flight grids, and how to develop an accurate operator splitting that is able to account for the coupling of pressure and saturation/compositions seen in compressible flow.

Tracing of streamlines typically leads to a graded and highly irregular grid with (many) orders of magnitude in differences between the smallest and the largest cells. To simulate transport, most streamline solvers use a finite-volume discretization, for which is imperative to keep a reasonable ratio between the smallest and largest cells to maintain overall computational efficiency, regardless of whether one uses a standard upwind or a higher-order scheme. It is therefore common to introduce a resampling on a more regular grid, or alternatively to introduce an adaptive scheme that may retain most of the irregularity of the grid (and only

merge the cells with the smallest volume). For incompressible, immiscible two-phase flow, however, one may entirely avoid the difficulty of the irregular grid by using an unconditionally stable and grid-independent front-tracking method (Holden and Risebro 2002) that is highly efficient and has superior resolution of discontinuities compared with finite-volume schemes. In previous work, it has been shown that the front-tracking approach can be extended to three-phase (Lie and Juanes 2005) and miscible two-phase flow (Juanes and Lie 2008) under the assumption of incompressibility. Herein, we will assess the applicability and efficiency of a new set of front-tracking and large–time-step methods (Nilsen and Lie 2008) for use in streamline simulation of compressible two-phase flow.

Secondly, we will discuss splitting errors introduced when using a sequential solution procedure for the coupled system of flow and transport equations and try to develop a method that reduces these errors. In an ordinary IMPES formulation, one first discretizes the mass-balance equations before introducing the splitting of flow and transport, which means that one can achieve complete conservation of mass and volumes even in the compressible case. In a streamline method, a sequential splitting of the flow and transport equations is introduced *before* discretizing the equations, which means that it is not straightforward to achieve conservation of both mass and volume. For incompressible flow, the 1-D conservation laws along each streamline do not depend on the absolute pressure level. In other words, saturations or components can be evolved without knowledge of the pressure level because their wave speeds are independent of the absolute pressure. For compressible systems, the 1-D transport equations along each streamline describe conservation of volume and contain coefficients or source terms, depending on the formulation, that depend on both saturation and pressure. A change in pressure without a corresponding change in saturation/components will therefore introduce a violation of mass conservation, and vice versa. Similarly, in a formulation using source terms, it may happen that the phase sources do not sum to zero, thereby resulting in a volume error. Minimizing mass and volume errors is hence a particular challenge when developing compressible streamline methods, and may also impose restrictions on the size of the pressure steps that diminishes the efficiency of streamline simulation over conventional finite-volume simulators. These issues will be discussed in more detail below.

Theoretical Background

We start by outlining the formulation and solution methods used to set up our streamline method for compressible flow.

Mathematical Model. We consider immiscible, two-phase flow in the absence of capillary forces. To model this system, we will use the so-called fractional formulation consisting of an equation for the pressure p (and the total Darcy velocity $\vec{v}_t$)

$$\big(c_r + \phi(c_1 S_1 + c_2 S_2)\big)\frac{\partial p}{\partial t} + \nabla \vec{v}_t - (c_1 f_1 + c_2 f_2)\,\vec{v}_{tg} \cdot (\mathbf{K}\lambda_t)^{-1}\vec{v}_t - (c_1 - c_2)(\rho_1 - \rho_2)\frac{\lambda_1 \lambda_2}{\lambda_t}\,\vec{v}_{tg} \cdot \vec{g} = q_1 + q_2, \quad (1)$$

and a transport equation for the saturation S_i of each phase, here written in non-conservative form,

$$\phi\frac{\partial S_i}{\partial t} + \vec{v}_t \cdot \nabla f_i(S_i) + \nabla \cdot \left((\rho_i - \rho_j)\frac{\lambda_i \lambda_j}{\lambda_t}\mathbf{K}\vec{g}\right) + h_i(S, p, \vec{x}) = q_i, \qquad i = 1, 2, \quad (2)$$

where the source resulting from compressibility is

$$h_i(S, p, \vec{x}) = (c_r S_i + \phi c_i S_i)\frac{\partial p}{\partial t} + \frac{1}{\rho_i}f_i(S_i)\nabla \cdot (\rho_i \vec{v}_t) + \left((\rho_i - \rho_j)\frac{\lambda_i \lambda_j}{\lambda_t}\mathbf{K}\vec{g}\right) \cdot \nabla \ln \rho_i. \quad (3)$$

The parameters in Eq. 1–Eq. 3 have different dependence: the gravity vector $\vec{g}$ is a constant; the permeability $\mathbf{K}$ and the porosity ϕ depend on the spatial coordinate $\vec{x}$; the rock compressibility c_r, the phase compressibilities c_i, and the densities ρ_i depend on pressure; and the phase mobilities λ_i, the total mobility $\lambda_t = \lambda_1 + \lambda_2$, and the fractional flows $f_i = \lambda_i/\lambda_t$ depend strongly on the saturation and weakly on the pressure. Finally, $\vec{v}_{tg}$ is shorthand for $-\lambda_t \mathbf{K}\nabla p$.

Our study is partially motivated by injection and migration of CO_2 in saline aquifers or depleted reservoirs, for which one of the phases is incompressible. Assuming that phase 1 is incompressible, the corresponding transport equation reduces to

$$\phi\frac{\partial S_1}{\partial t} + \vec{v}_t \cdot \nabla f_1 + \nabla \cdot \left((\rho_1 - \rho_2)\frac{\lambda_1 \lambda_2}{\lambda_t}\mathbf{K}\vec{g}\right) + f_1 \nabla \cdot \vec{v}_t = q_1. \quad (4)$$

This special case will be treated separately later in the article.

Sequential Splitting. In streamline simulation, one introduces a sequential splitting to solve the flow and transport in separate steps. The flow equation Eq. 1 is solved implicitly in time on an Eulerian grid to obtain p and $\vec{v}_t$; here we will either use a standard two-point or a mimetic (Aarnes et al. 2008) spatial discretisation. The pressure and velocity are then held fixed as the saturations S_i are advanced a time-step Δt forward in time. To this end, we introduce a further operator splitting (Gmelig Meyling 1991; Bratvedt et al. 1996) and solve the advective part

$$\phi\frac{\partial S_i}{\partial t} + \vec{v}_t \nabla f_i + h_i(S, p, \vec{x}) = q_i \quad (5)$$

along streamlines and the gravity part

$$\phi \frac{\partial S_i}{\partial t} + \nabla \cdot \left((\rho_i - \rho_j) \frac{\lambda_i \lambda_j}{\lambda_t} \mathbf{K} \vec{g} \right) = 0 \tag{6}$$

along gravity lines. The streamlines are traced using Pollock's method (Pollock 1988), noting that whereas all streamlines start and end at wells (or aquifers) for incompressible flow, streamlines may start and end in any cell that acts as a source or sink in the compressible case.

In the sequential splitting, one should take particular care of the fact that the pressure equation Eq. 1 represents conservation of volume. Indeed, the pressure equation is what balances the source terms in the advection equation Eq. 5, and if this is not taken into account when setting up the saturation step, one cannot guarantee that $0 \le S_i \le 1$. For instance, a naïve splitting that keeps $\nabla \cdot \vec{v}_t$ constant for the case with one incompressible phase, may immediately give saturations outside the unit inverval because we do not impose the physical restriction that the divergence of $\vec{v}$ is identically zero when only the incompressible phase is present. To make sure that the transport equations always conserve volume, we can use the pressure equation Eq. 1 to eliminate $\nabla \cdot \vec{v}_t$ from the source term Eq. 3, to get new source terms that sum to zero,

$$\widetilde{h}_i(S, p, \vec{x}) = c_r \frac{\partial p}{\partial t} \big(S_i - f_i(S_i) \big) + \phi \frac{\partial p}{\partial t} \Big(c_i S_i - [c_1 S_1 + c_2 S_2] f_i(S_i) \Big) + \frac{\vec{v}_{tg} \mathbf{K}^{-1} \vec{v}_t}{\lambda_t^2} \Big(-c_i \lambda_i + [c_1 \lambda_1 + c_2 \lambda_2] f_i(S_i) \Big)$$
$$+ \frac{\vec{v}_{tg} \vec{g}}{\lambda_t} \Big[(c_i - c_j)(\rho_i - \rho_j) \frac{\lambda_i \lambda_j}{\lambda_t} f_i - c_i (\rho_i - \rho_j) \frac{\lambda_i \lambda_j}{\lambda_t} \Big]. \tag{7}$$

In addition, these source terms have the property that $\widetilde{h}_i(0, \vec{x}) \equiv 0$ and $\widetilde{h}_i(1, \vec{x}) \equiv 0$. This, together with $f_i(0) \equiv 0$, guarantees that solutions of the transport equation fulfill the inequality $0 \le S_i \le 1$.

1-D Transport Equations. Streamlines are a family of curves that are everywhere tangential to the instantaneous flow field. In the absence of gravity and capillarity forces, there is no mass transfer between individual streamlines, meaning that each streamline can be viewed as an isolated flow system. By introducing the time-of-flight τ, given by $\vec{v}_t \cdot \nabla \tau = \phi$, and the operator identity $\vec{v}_t \cdot \nabla = \phi \partial_\tau$, one can reduce the multidimensional advection equations Eq. 5 to a family of 1-D transport equations along streamlines. The equation reduces to[1]

$$\frac{\partial S}{\partial t} + \frac{\partial f(S)}{\partial \tau} + \frac{h_i(S, \tau)}{\phi} = 0. \tag{8}$$

Because $\vec{v}_t$ is generally not divergence free, Eq. 8 cannot be written in conservative form, even in cases for which it should be possible like in Eq. 4. It may therefore be convenient, at least in the case with one incompressible phase, to introduce a volume factor σ to make a divergence-free field, that is, $\nabla \cdot (\vec{v}_t / \sigma) = 0$. If we define a new spatial coordinate η along each streamline by $(\vec{v}_t / \sigma) \cdot \nabla = \phi \partial_\eta$ and apply this to Eq. 5, we get a form that is particularly well-suited for finite-volume discretisations,

$$\frac{\partial S}{\partial t} + \frac{\partial}{\partial \eta} \big(\sigma f(S) \big) = 0. \tag{9}$$

The above is the same idea as presented by Cheng et al. (2006) and Beraldo et al. (2008). If we instead use the standard time-of-flight coordinate, the transport equation will not be explicitly conservative, but take the form

$$\frac{\partial S}{\partial t} + \frac{1}{\sigma} \frac{\partial}{\partial \tau} \big(\sigma f(S) \big) = 0. \tag{10}$$

Front-Tracking Methods for the 1-D Tranport Equations

Lagrangian methods have a long history and are based on approximating the solution following trajectories inherent in the system. Front tracking (Holden and Risebro 2002) is one such method, which is based on approximating the solution of first-order quasilinear conservation laws by a piecewise constant function and then evolving the solution by solving Riemann problems and tracking discontinuities. For incompressible flow, this is simply done by approximating the fractional flow $f(S)$ by a piecewise-linear function and solving the corresponding approximate transport equation exactly. Recently we have developed a set of new front-tracking and large–time-step methods for compressible flow (Nilsen and Lie 2008), following ideas introduced by Karlsen et al. (2004) and Karlsen et al. (2008) for solving general conservation laws with discontinuous flux functions. For completeness, we will explain the methods in some detail, starting with the special case of one incompressible phase.

---

[1]In streamline simulation, the fluid sources/sinks are modeled as inflow/outflow boundaries rather than source terms. Therefore $q_i \equiv 0$ henceforth, and we also drop the subscripts on S and f to simplify notation. Moreover, since p is assumed fixed during the transport step, we suppress the p-dependence in h_i.
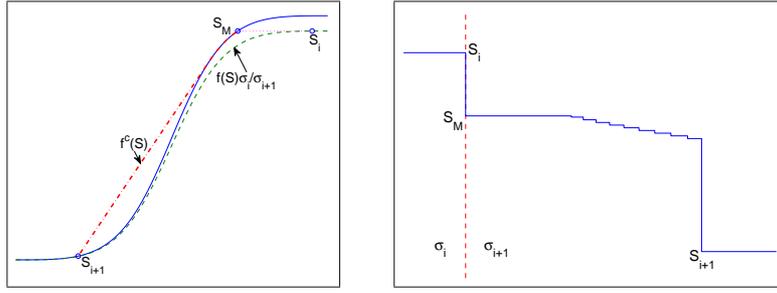
Fig. 1—Constructing the solution to a discontinuous Riemann problem: the left plot shows the construction and the right plot the resulting self-similar solution consisting of a standard Buckley–Leverett profile with left state S_M and right state S_{i+1} followed by a stationary discontinuity (S_i, S_M).

One Incompressible Phase. For the special case of one incompressible and one compressible phase, we can develop a particularly simple front-tracking method by considering Eq. 9 or Eq. 10 rather than Eq. 8. Notice, however, that this means that the second phase will generally not be mass conservative.

We start by considering Eq. 9 for the case when f is a piecewise linear function of S and σ and the initial data S_0 are piecewise constant functions of τ on the underlying time-of-flight grid. This Cauchy problem consists of a series of local Riemann problems

$$\frac{\partial S}{\partial t} + \frac{1}{\sigma}\frac{\partial}{\partial \tau}\big(\sigma f(S)\big) = 0, \qquad S(\tau,0) = \left\{ \begin{array}{ll} S_i, & \tau < \tau_i, \\ S_{i+1}, & \tau > \tau_i, \end{array} \right. \qquad \sigma(\tau) = \left\{ \begin{array}{ll} \sigma_i, & \tau < \tau_i, \\ \sigma_{i+1}, & \tau > \tau_i. \end{array} \right. \qquad (11)$$

Assume first that $\sigma_i = \sigma_{i+1}$, which corresponds to the incompressible case, and introduce a piecewise-linear local convexification[2] f^c of f on the interval $[S_i, S_{i+1}]$ (see **Fig. 1**), where $\{s^0, \ldots, s^N\}$ denotes the nodes of f^c. Then the solution of Eq. 11 is a similarity solution (Riemann fan) consisting of constant states

$$S(\tau,t) = \left\{ \begin{array}{ll} s^0 = S_i, & \tau < r_1(t), \\ s^j, & r_j(t) < \tau < r_{j+1}(t), \quad j = 1, \ldots, N-1. \\ s^N = S_{i+1}, & \tau > r_N(t), \end{array} \right. \qquad (12)$$

separated by discontinuities along straight space-time rays $r_j(t)$

$$\frac{dr_j}{dt} = \frac{f^c(s^{j+1}) - f^c(s^j)}{s^{j+1} - s^j}, \quad r_j(0) = \tau_i. \qquad (13)$$

To construct the global solution, we glue together the local Riemann fans and track the rays (henceforth called fronts) until they collide. Each collision between two (or more) rays will give a new Riemann problem that can be solved by Eq. 12. Mathematically, one can prove that there will only be a finite number of front collisions, and thus a solution to infinite time can be constructed in a finite number of steps.

For compressible flow, we also need to solve Riemann problems and introduce fronts where σ is discontinuous. The solution of the Riemann problem in the general case with $\sigma_i \neq \sigma_{i+1}$ consists of a stationary discontinuity with left state S_i and right state S_M given by $\sigma_i f(S_i) = \sigma_{i+1} f(S_M)$, followed by a Riemann fan Eq. 12 with left state S_M (instead of S_i) and right state S_{i+1}. Notice, however, that in setting up Eq. 10, we have incorrectly assumed that $\nabla \cdot v$ is constant in time within each cell. This means that in some cases it may be impossible to find S_M (e.g., if $\sigma_i f(S_i)/\sigma_{i+1} > 1$). We then introduce a regularisation by setting $S_M = 1$, which corresponds to introducing a more general solution based on a modified entropy condition (Adimurthi et al. 2007). We will come back to these operator splitting anomalies and errors in the numerical examples below.

The overall method will henceforth be referred to as FT1. Because Eq. 10 is in non-conservative form, FT1 will not conserve volumes. On the other hand, this formulation is the one that is closest to the method we use for incompressible flow. Alternatively, one may base the front-tracking method on Eq. 9, giving a method that follows the exact same construction, except that we now need to scale the speed of the fronts Eq. 13 by σ. Notice that both methods require no regularisation if $\nabla \cdot \vec{v}_t$ is positive, which will typically be the case for CO_2 injection when the $\partial_t p$-term does not dominate the pressure equation Eq. 1.

The accuracy of the FT1 is generally determined by three tolerances that control the resolution of the piecewise-linear approximation of $f(S)$, the resolution of the piecewise-constant approximation to $\sigma(\tau)$ (i.e., the resolution of the flux as a function of τ), and the resolution of the piecewise-constant approximation of $S(\tau,0)$. (In the next subsection, we will introduce two new methods that in addition have a time step that decides resolution of the compressible source term as a function of time). However, when FT1 is applied in streamline simulation, the last two tolerances are given implicitly by the time-of-flight grid. In standard

---

[2]The convexification can be visualised as cutting the upper edge of a cardboard to the shape of f and then stretching a rubber band above the cardboard from S_i to S_{i+1}. If $S_i < S_{i+}$, we cut the lower edge to the shape of f and stretch the rubber band below the cardboard.

finite-volum methods, the accuracy is decided by the grid and the CFL target that controls the time steps, but for applications along streamlines, one must typically introduce a regridding step to ensure computational efficiency. For incompressible flow, the propagation of information along the streamline is independent of the pressure, so the regularized grid can be generated with only efficiency and spatial accuracy in mind. For compressible flow, on the other hand, pressure-dependent quantities from the underlying Eulerian grid must also be correctly represented on the regularized grid. This is done automatically in the front-tracking approach by introducing static fronts that represent discontinuities in σ. Hence, in terms of parameters, it is therefore simpler to control the accuracy of FT1 than controlling the accuracy of a finite-volume method.

To consider the number of operation needed to calculate a solution for the discontinuous front-tracking method it is illustrative to consider a solution formed by parallel shock fonts propagating through a uniform and a nonuniform grid as shown in **Fig. 2**. For the uniform case, the number of front collisions is determined by the CLF number and is exactly equal the number of function evaluations needed for an explicit method with CFL number equal one. In terms of efficiency, this is a worst-case scenario in which the front-tracking method will be more expensive than a standard explicit method. For each collision, the front-tracking method must solve a discontinuous Riemann problem, which amounts to computing the intermediate state S_M and solving a regular Riemann problem. Both these operations consist of searching the linear segments of the flux function, and solving a discontinuous Riemann problem is hence logaritmic in the number of linear segments. In a more realistic example (e.g., with a Buckley–Leverett type profile), the number of fronts required to represent the solution will normally be much smaller than the number of grid cells, and hence front tracking will have good performance even for uniform grids.

In the nonuniform case, the time step of an explicit finite-volume method would be limited by the time the fastest discontinuity takes to propagate through the smallest grid cell, which means that the number of time steps will increase drastically. For front tracking, on the other hand, it does not matter if the grid is regular or irregular, as this method has no global CFL restriction. The local CFL number only shows up as small increments between front collisions but does not affect the total number of collisions.

Two Compressible Phases. Because the transport equation has a source term, we cannot use a pure front-tracking method, but will instead use the large–time-step method by Karlsen et al. (2008), in which the front-tracking method is an essential ingredient. To this end, we rewrite Eq. 5 using the streamline coordinate, introduce an internal time-step k, and evaluate the source term using saturation values from the previous internal time-step,

$$\frac{\partial S^{n+1}}{\partial t} + \frac{\partial f(S^{n+1})}{\partial \tau} = -\frac{h(S^n, \tau)}{\phi}, \qquad t \in [\ell \Delta t + nk, \ell \Delta t + (n+1)k). \tag{14}$$

We define the function $H(\tau) = \int_0^\tau h(r, S^n(r))/\phi(r)\, dr$ and may rewrite the above equation as

$$\partial_t \big(S + H(\tau)\big) + \partial_\tau \big(f(S) + H(\tau)\big) = 0. \tag{15}$$

If H is now approximated by a picewise-constant function, we can apply a front-tracking scheme that is similar to FT1 within each time-step. The only difference is that the stationary discontinuity (S_i, S_M) now is determined from $f(S_i) + H_i = f(S_M) + H_{i+1}$. If $H_i > H_{i+1}$, we may have to regularise the solution at the end-point $S = 1$ and similarly at $S = 0$ if $H_i < H_{i+1}$. The overall scheme, consisting of front-tracking with a restart after each time step k, will be referred to as FT2. In the restart we can either keep all discontinuities from one time step to the next, or introduce a projection back to the TOF-grid at the end of each time step.

Alternatively, we may use the source term Eq. 7 rather than Eq. 3, which means that also the effect of $\nabla \cdot \vec{v}_t$ on the source term is updated at the end of each internal time-step. The resulting method will be referred to as FT3.

Test Cases in 1-D
To keep the presentation as simple as possible, we start by discussing injection scenarios on a uniform 1-D grid. Using a uniform grid is not the most efficient way of utilizing the front-tracking methods. Ideally, the spacing in the picewise-constant
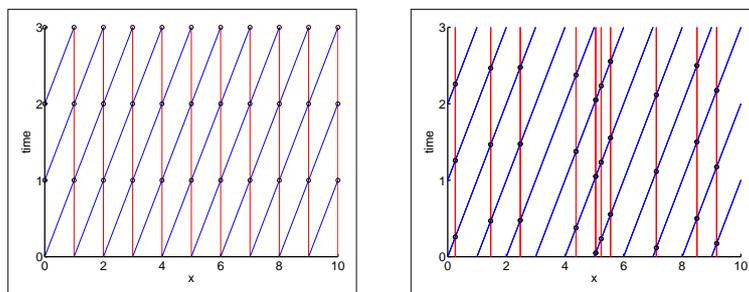
Fig. 2—A conceptual illustration of the number of discontinuous Riemann problem that need to be solved for a uniform and nonuniform grid. The blue lines are fronts, whereas the vertical red lines represent the time-of-flight grid (i.e., discontinuities in p and σ). Front collisions are indicated by small circles.

TABLE 1—RUNTIME STATISTICS FOR CASE 1

	FT1	FT2	FT3	FT2/ FT3	FV
Fronts	99	258	307	183	—
Collisions	26	66	84	51	—
Finding S_M	54	143	137	65	—
Riemann problems	35	105	162	68	—
Internal steps	—	10	10	1	19
CPU time (ms)	0.1	0.3	0.4	0.15	0.1

TABLE 2—RUNTIME STATISTICS FOR CASE 2

	FT1	FT2	FT3	FT2/ FT3	FV
Fronts	184	645	806	71	—
Collisions	51	113	188	20	—
Finding S_M	65	276	420	25	—
Riemann problems	68	277	372	40	—
Internal steps	—	10	10	1	13
CPU time (ms)	0.1	0.5	0.5	0.08	0.1

approximation of $S(\tau, 0)$ and $\sigma(\tau)$ should instead depend on local accuracy requirements so that the spacing is small in regions with large gradients and large in regions where $S(\tau, 0)$ and/or $\sigma(\tau)$ vary slowly. In comparing efficiency to a finite-volume method, uniform grids will generally be least favourable for the front tracking because they do not allow the method to fully focus computations only where most required. In other words, if a front-tracking method is as efficient as a finite-volume method on a uniform grid, it will in general be much more efficient on a highly graded grid.

Unless stated otherwise, all front-tracking methods will use a linearization of the flux function with 40 segments of equal length. For the flux function, we assume quadratic relative permeabilities with zero residual saturations.

Case 1: Injection of a Compressible Phase. We consider the injection of a compressible phase into a homogeneous 1-D reservoir of five length units saturated by an incompressible phase that is ten times more viscous. This type of computational setup is very important in the simulation of CO_2 injection into deep saline aquifers. To compute the solution up to time $t = 100$, we use only five pressure steps, which is too few to capture the coupling of flow and transport accurately, but will enable us to highlight the qualitative differences in the various methods.

Initially, the pressure satisfies an incompressible equation, but as we start injecting the compressible fluid, the nonlinear compressible term starts to dominate the pressure equation, Eq. 1. This will cause the total divergence to be positive, which, in turn, implies that the solution of Eq. 8 will not go out of bounds, even if we do not introduce regularizations at the saturation endpoints. FT1 is particularly suited to simulate this type of scenario because there is no need to approximate the source term in time. The right panels of **Fig. 3** show the fronts in (τ, t) space for FT1 and for FT3 with 10 internal time steps. The main difference in the two solutions is that new static and new dynamic fronts appear during a pressure step for FT3. With many intervals in the picewise-constant approximation of the initial saturation, this will increase the number of fronts significantly. Indeed, **Table 1** shows that the number of fronts increases by a factor of 3. The left plot in Fig. 3 shows saturation profiles for FT1, FT2, FT3, and the standard single-point upwind method compared with a fine-grid reference solution computed using a finer grid and more pressure updates. For completeness, we also include a solution computed by the incompressible front-tracking method FTi (i.e., the method solving Eq. 10 for $\sigma \equiv 1$). We notice that FT3 is a bit nearer the reference solution than the other front-tracking methods, but still has the same error in the front position caused by too few pressure steps.

Case 2: Injection of an Incompressible Phase. In the second test, we consider the same computational setup, except that we now inject the incompressible phase into the compressible phase. In this case, the time derivative of the pressure in Eq. 1 dominates the divergence, in particular in the beginning of the simulation. At later times, the situation is similar as in Case 1.

The computational results are shown in **Fig. 4** and **Table 2**. Studying the fronts depicted in the right panels of Fig. 4, we observe that FT3 has a much lower number of static fronts than FT1. This is caused by the different representation of the source term in the two front-tracking methods. FT1 represents the medium by approximating σ, which is changing in the whole domain, while FT3 approximates the source term, which is zero where there is only one fluid present. Even if FT1 has to represent more static fronts in this case, we see no performance penalty because these extra fronts do not participate in collisions. Indeed, Table 2 shows that FT1 has fewer collisions than FT2 and FT3 with 10 internal time steps. If we use only one internal time step, then FT2 and FT3 are as fast as FT1 with approximately the same accuracy; however, because of too infrequent pressure updates, the error in the front velocity is much larger compared with Case 1.
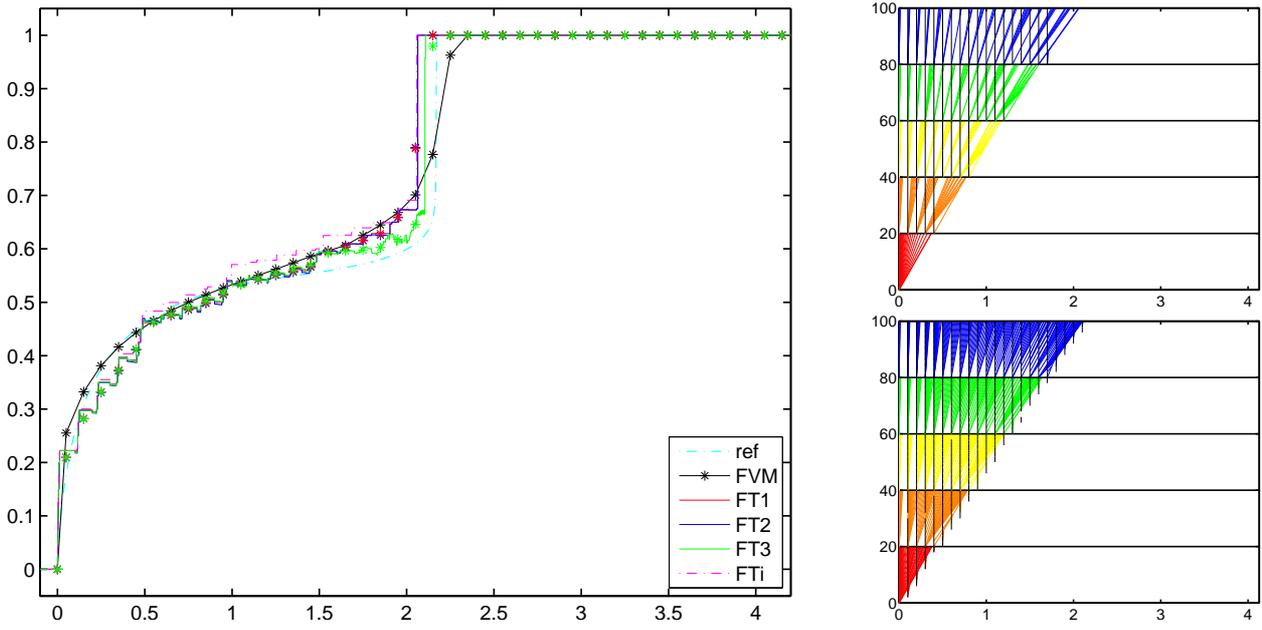
Fig. 3—Injection of a compressible phase into a ten times less viscous incompressible phase. The left plot shows saturation profiles for the front-tracking methods (FT1,FT2,FT3,FTi) and the finite-volume (FV) method. Both FT2 and FT3 used 10 internal timesteps. The upper-right plot shows the fronts of FT1, while the lower-right plot shows the same for FT3. The pressure steps are indicated by black horisontal lines, and the fronts have been given a different color for each pressure step to distinguish them.
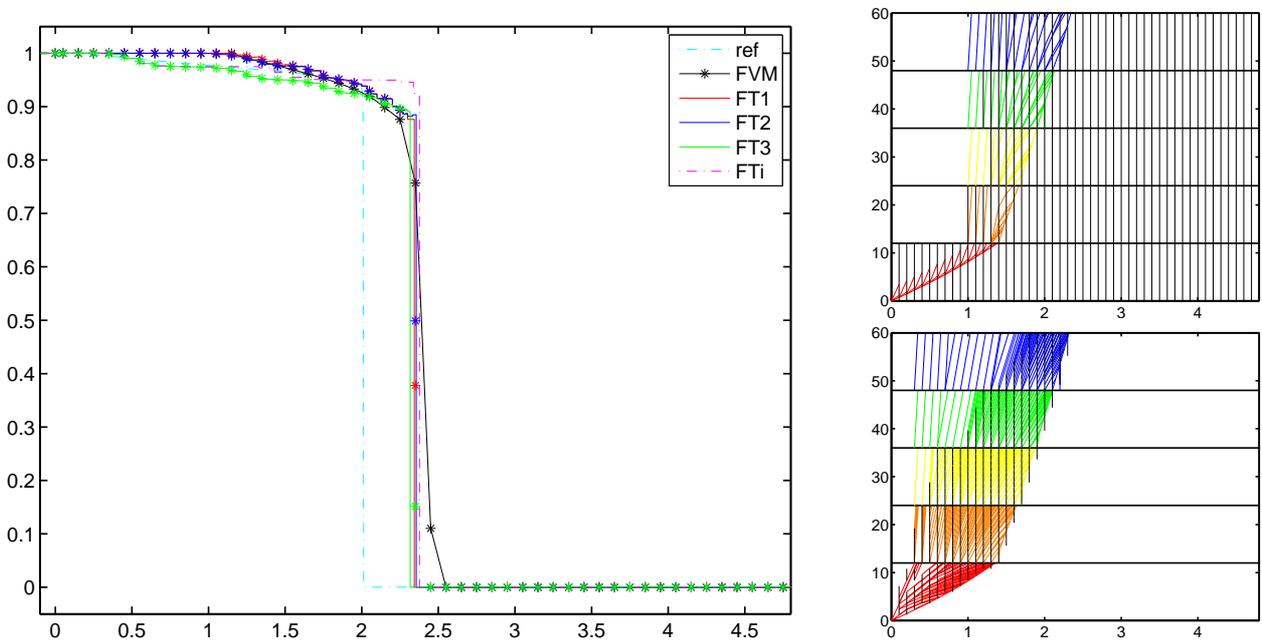
Fig. 4—Injection of an incompressible phase into a ten times less viscous compressible phase. The left plot shows saturation profiles for the front-tracking methods (FT1,FT2,FT3,FTi) and the finite-volume (FV) method. Both FT2 and FT3 used 10 internal timesteps. The upper-right plot shows the fronts of FT1, while the lower-right plot shows the same for FT3. The pressure steps are indicated by black horisontal lines.

Examples of Streamline Simulations

Front-tracking has been shown to be particularly efficient for solving transport equations on irregular grids arising in streamline simulations (Bratvedt et al. 1992). In this section, we will show a few simple examples that highlight the capabilities of our new front-tracking methods.

Case 3: Injection Along a Streamline. In the first example, we will perform a simulation similar to Case 1, but now using a highly irregular grid. To add some realism, we apply the same injection scenario to a real-life real-life geological model of the Johansen formation from the North Sea and consider one transport step along a particular streamline. The Johansen formation lies underneath the Troll field and is considered as a prime site for injection of CO_2 into a deep saline aquifer. The model consists of 27,032 active cells that cover an area of 50×50 km and 1 km in the vertical direction. In this model, we have introduced two synthetic wells and traced streamlines as shown in **Fig. 5**.
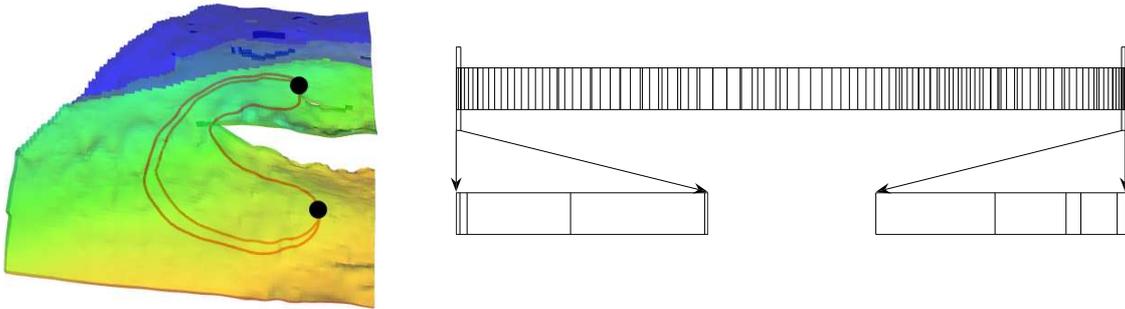
Fig. 5—The right plot shows the Johansen geomodel with three example streamlines. The left plot shows the corresponding time-of-flight grid used for the simulation in 1-D.

To make the situation a bit more complicated than in Case 1, we consider a transport step after some CO_2 has been injected into the formation so that $S(\tau, 0)$ and $\sigma(\tau)$ are not constant along the 1-D grid. Moreover, to make a fair comparison with the FV method, this method is run on a regularized grid in which the ratio between the smallest and the largest cell equals four. The computational results are shown in **Fig. 6** and **Table 3**. Because of the nonuniform initial state along the streamline, the front structures of FT1 and FT3 shown in the right plots of Fig. 6 are slightly more complex than in Case 1 (Fig. 3). Comparing accuracy and the performance of the method, we see that the front-tracking methods are approximately 5 times faster with better accuracy, even if they operate on the original grid and the FV method uses the regularized grid with a lower number of cells. Table 3 also shows that using FT3 together with 10 internal time steps for this case increases the number of collisions quite significantly but does not give better results.

Case 4: One Injection Well and Hydrostatic Boundary. For injection of a CO_2 plume into an aquifer it is often a good model to consider hydrostatic boundary conditions. We therefore present a calculation in which we use our front-tracking method to solve a problem with one well and Dirichlet boundary conditions. This will test how the method works on real cases for the advection term of the transport equation. In the particular case shown in **Fig. 7**, we use a compressibility of $2.3 \cdot 10^{-7}$ Pa^{-1} and inject 10 Mtonn/year. To test the robustness of the front-tracking method, we use time step of 50 years. In our research simulator (which unfortunately is not yet fully optimized), the runtime is dominated by the pressure solution: each pressure step required approximately 20 seconds, whereas a typical transport step took approximately 0.5 (2) seconds with 2268 (3644) streamlines. In future work, we will investigate the use of a multiscale pressure solver (Natvig et al. 2009) to increase performance.

Case 5: One Injection Well and One Production Well. Injection of an incompressible fluid into a compressible fluid is highly relevant for enhanced oil and gas recovery. In our second 3-D test, we have therefore placed one injector and one producer in

TABLE 3—RUNTIME STATISTICS FOR THE 1-D STREAMLINE FROM THE JOHANSEN FORMATION

	FT1	FT2	FT3	FT2/ FT3	FV
Fronts	497	488	1359	610	—
Collisions	241	190	542	289	—
Finding S_M	231	261	777	277	—
Riemann problems	239	199	544	286	—
Internal steps	—	10	10	1	253
CPU time (ms)	4	7	18	6	30
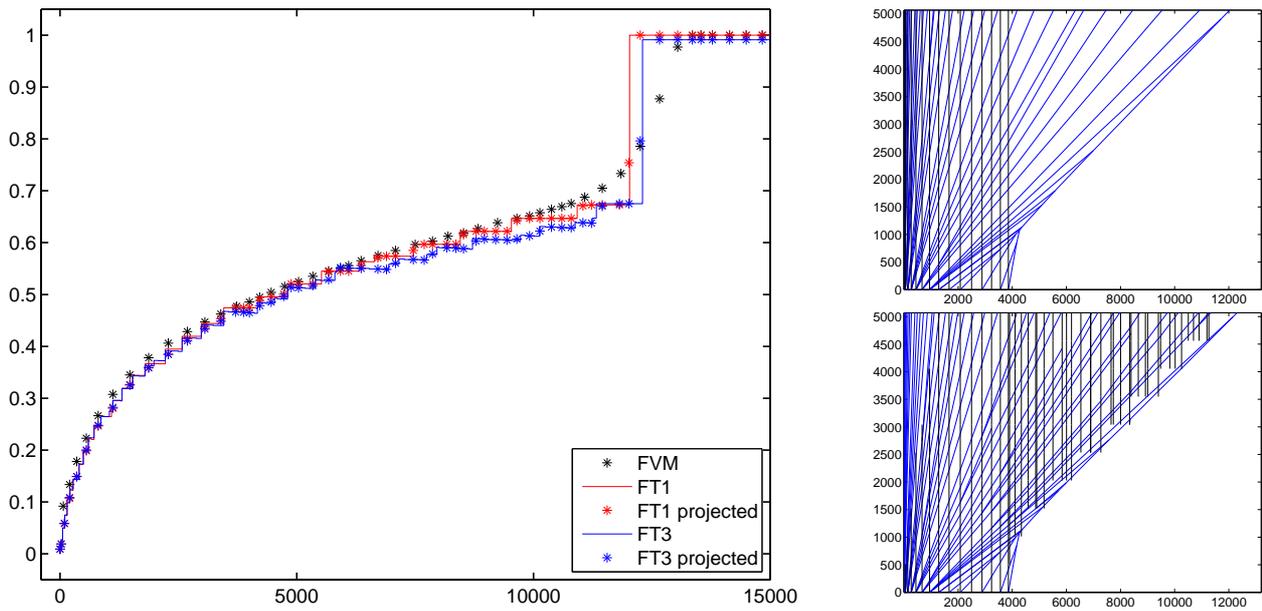
Fig. 6—Solutions along a particular streamline in the simulation of CO_2 injection into the Johansen formation. The left plot shows saturation profiles for FT1, FT3 with 10 internal time step, and the FV method on a regularized grid. The lines show the picewise-constant front-tracking solution, whereas the stars show the solution projected back onto the time-of-flight grid. The upper-right plots shows the fronts for FT1, while the lower-right plot shows the same for FT3.

water saturation   pressure
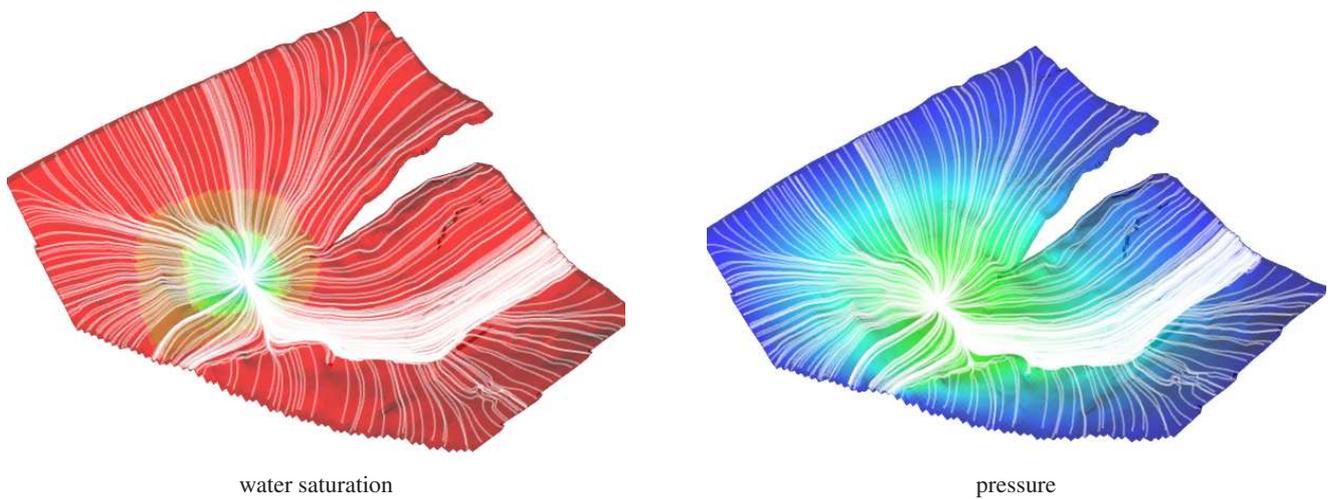
Fig. 7—The Johansen formation with one well and hydrostatic boundary conditions.

water saturation   pressure

Fig. 8—The Johansen formation with one injector and one producer.

the Johansen formation and assumed that the reservoir is efficiently sealed off with no-flow boundaries. For this simulation, the pressure step required approximately 30 seconds, whereas a typical transport step took 0.4 (3) seconds with 1000 (2497) streamlines. As in Case 4, we see that we can almost completely eliminate the computational cost of the transport equation compared with the pressure solution, even without regridding along streamlines.

Case 6: Three Injectors and One Producer In a Syntetic Faulted Grid. **Fig. 9** shows a synthetic test case with three pressure-driven wells that inject a compressible fluid into a reservoir that is initially saturated by an incompressible fluid. Fluids are allowed to escape through an open well in the middle of the domain. The left figure shows the saturation computed by an explicit single-point upwind scheme compared with the saturation profile computed by our streamline method with FT1. The right subplot shows some of the streamlines used for the streamline simulation. We observe that the streamline method gives much sharper resolution of the displacement fronts than the explicit method. (in practice, one would have used an implicit method that would have given even more diffusion). In this calculation we have used compressibility so that the density at the production well for the compressible phase is 10 times less than at the injection well.

Conclusions

We have presented three front-tracking methods for compressible flow that can incorporate the effects of compressibility in two different transport equations used in the framework of sequential splitting. We have shown that the methods are comparable to a standard finite-volume method on idealized cases with uniform grids but become significantly faster when applied to the type of highly irregular grids seen in realistic applications. As opposed to finite-volume schemes, the new methods do not require regridding along the streamlines and will therefore be more true to the input data sampled from the underlying Eulerian grid that is used by the pressure solver.

In the method (FT1) developed especially for the case of one incompressible phase, the accuracy is controlled by a single parameter that specifies the number of segments in the approximation of the flux function. This parameter is easy to choose and gives good control over the spatial accuracy of the method. In the methods for two compressible phases (FT2 and FT3), one must also specify a local time step. The size of this time step may have a significant impact on both the accuracy and the computational time and more research is required before we can provide a strategy for making an optimal choice automatically.

For the simple 3-D simulations reported herein, we see that streamline tracing together with one of our front-tracking methods eliminiates the cost of the transport step almost completely compared with the pressure step without any regularization of the time-of-flight grid.

Acknowledgements

water saturation for euler method   water saturation for streamline method   streamlines
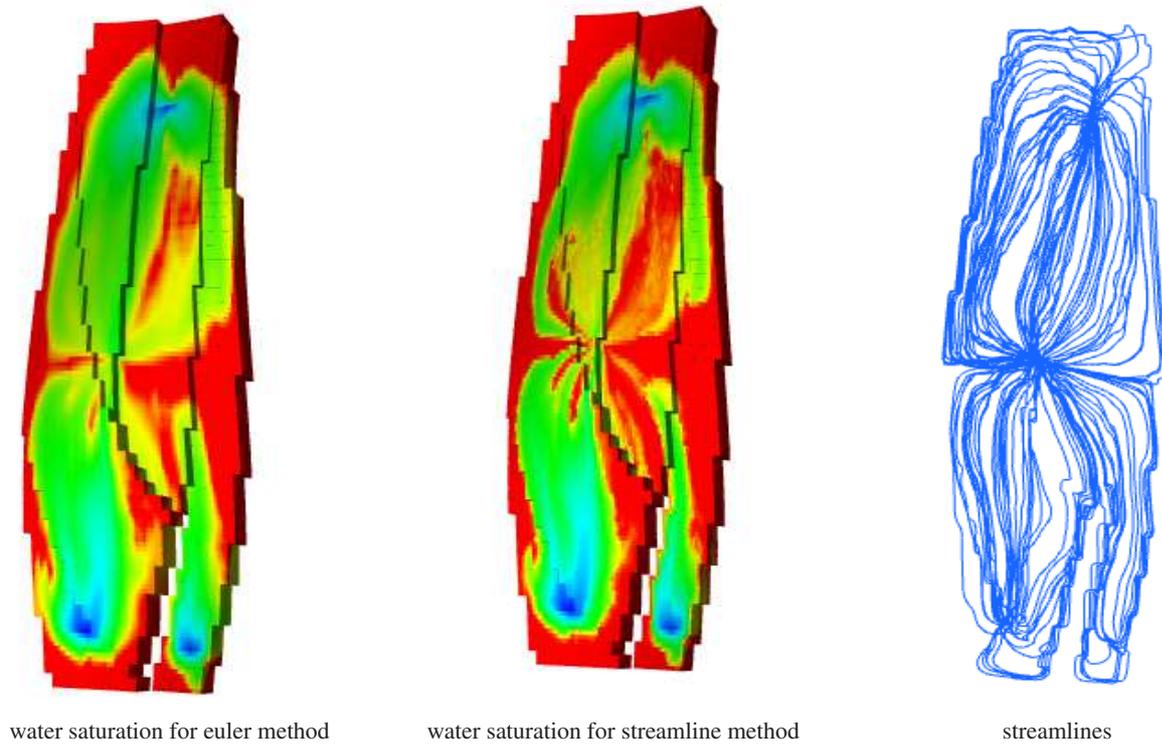
Fig. 9—A synthetic case with three injectors and one producer.

Nomenclature

c_r	=	rock compressibility	S_i	=	saturation of phase i or in gridblock i
c_i	=	phase compressibility	s^i	=	node point in piecewise linear f^c
f_i	=	fractional flow of phase i	t	=	time
f^c	=	convexification of flux function f	$\vec{v}_t$	=	total Darcy velocity
$\vec{g}$	=	gravity vector	$\vec{v}_{tg}$	=	short-hand, $\vec{v}_{tg} = -\lambda_t \mathbf{K}\nabla p$
h_i	=	compressible source term of phase i	$\vec{x}$	=	spatial coordinate
$\mathbf{K}$	=	absolute permeability	Δt	=	step length of sequential splitting
k	=	step length of internal steps	η	=	streamline parameter for divergence-free vector field
ℓ	=	step number in sequential splitting	λ_i	=	mobility of phase i
n	=	internal step number	λ_t	=	total mobility
n	=	number of linear segments in f^c	ρ_i	=	density of phase i
p	=	pressure	σ	=	volume factor to make $\vec{v}/\sigma$ divergence free
q_i	=	volumetric rate of phase i	τ	=	time-of-flight
r_j	=	space-time ray in Riemann problem	ϕ	=	porosity

References

Aarnes, J. E., Krogstad, S., and Lie, K.-A. 2008. Multiscale Mixed/Mimetic Methods on Corner-Point Grids. *Comput. Geosci.*, 12(3):297–315. Doi:10.1007/s10596-007-9072-8.

Adimurthi, Mishra, S., and Veerappa Gowda, G. D. 2007. Conservation Law With the Flux Function Discontinuous in the Space Variable. II. COnvex-Concave Type Fluxes and Generalized Entropy Solutions. *J. Comput. Appl. Math.*, 203(2):310–344.

Beraldo, V. T., Blunt, M. J., and Schiozer, D. J. 2008. Compressible Streamline-Based Simulation with Changes in Oil Composition. Paper SPE 115983, presented at 2008 SPE Annual Technical Conference and Exhibition, Denver, Colorado, USA, 21–24 Septeber.

Bratvedt, F., Bratvedt, K., Buchholz, C., Holden, L., Holden, H., and Risebro, N. 1992. A New Front-Tracking Method for Reservoir Simulation. *SPE Reservoir Eng.*, 7(1):107–116.

Bratvedt, F., Gimse, T., and Tegnander, C. 1996. Streamline Computations for Porous Media Flow Including Gravity. *Transp. Porous Media*, 25(1):63–78.

Cheng, H., Osako, I., Datta-Gupta, A., and King, M. J. 2006. A Rigorous Compressible Streamline Formulation for Two and Three Phase Black Oil Simulation. *SPE J.*, 11(4):407–417.

Datta-Gupta, A. and King, M. J. 2007. *Streamline Simulation: Theory and Practice*, volume 11 of *SPE Textbook Series*. Society of Petroleum Engineers.

Gmelig Meyling, R. H. J. 1991. Numerical Methods for Solving the Nonlinear Hyperbolic Equations of Porous Media Flow. In *Third International Conference on Hyperbolic Problems, Vol. I, II (Uppsala, 1990)*, pages 503–517, Lund. Studentlitteratur.

Holden, H. and Risebro, N. H. 2002. *Front Tracking for Hyperbolic Conservation Laws*, volume 152 of *Applied Mathematical Sciences*. Springer-Verlag, New York.

Juanes, R. and Lie, K.-A. 2008. Numerical Modeling of Multiphase First-Contact Miscible Flows. II. Front-Tracking/Streamline Simulation. *Transp. Porous Media*, 72(1):97–120. DOI: 10.1007/s11242-007-9139-y.

Karlsen, K. H., Mishra, S., and Risebro, N. H. 2008. A Large Time-Stepping Scheme for Balance Equations. *J. Engrg. Math.*, 60(3–4):351–363.

Karlsen, K. H., Risebro, N. H., and Towers, J. D. 2004. Front Tracking for Scalar Balance Equations. *J. Hyperbolic Differ. Equ.*, 1(1):115–148.

Lie, K.-A. and Juanes, R. 2005. A Front-Tracking Method for The Simulation of Three-Phase Flow in Porous Media. *Comput. Geosci.*, 9(1):29–59.

Natvig, J. R., Skaflestad, B., Bratvedt, F., Bratvedt, K., Lie, K.-A., Laptev, V., and Khataniar, S. K. 2009. Multiscale Mimetic Solvers for Efficient Streamline Simulation of Fractured Reservoirs. Paper SPE 119132, presented at 2009 SPE Reservoir Simulation Symposium, The Woodlands, Texas, USA, 2–4 February.

Nilsen, H. M. and Lie, K.-A. 2008. On Front Tracking for Compressible flow. In *Proceedings of ECMOR XI, Bergen, Norway, 8–11 September*. EAGE.

Pollock, D. 1988. Semi-analytical Computation of Path Lines for Finite-Difference Models. *Ground Water*, 26(6):743–750.

Thiele, M. 2005. Streamline Simulation. In *8th International Forum on Reservoir Simulation*, Stresa / Lago Maggiore, Italy.