BIT
1999, Vol. 39, No. 4, pp. 683–700

0006-3835/99/3904-0683 $12.00

A DIMENSIONAL SPLITTING METHOD FOR QUASILINEAR HYPERBOLIC EQUATIONS WITH VARIABLE COEFFICIENTS*

KNUT–ANDREAS LIE[†]

Department of Informatics, University of Oslo, P.O. Box 1080 Blindern,
N–0316 Oslo, Norway. email: kalie@ifi.uio.no

Abstract.

A numerical method is presented for the variable coefficient, nonlinear hyperbolic equation $u_t + \sum_{i=1}^{d} V_i(x,t) f_i(u)_{x_i} = 0$ in arbitrary space dimension for bounded velocities that are Lipschitz continuous in the x variable. The method is based on dimensional splitting and uses a recent front tracking method to solve the resulting one-dimensional non-conservative equations. The method is unconditionally stable, and it produces a subsequence that converges to the entropy solution as the discretization of time and space tends to zero. Four numerical examples are presented; numerical error mechanisms are illustrated for two linear equations, the efficiency of the method compared with a high-resolution TVD method is discussed for a nonlinear problem, and finally, applications to reservoir simulation are presented.

AMS subject classification: 35L65, 65M12, 65M99

Key words: Quasilinear hyperbolic equations, variable coefficients, dimensional s-plitting, front tracking

1 Introduction

We consider variable coefficient, hyperbolic equations of the form

$$u_t + \sum_{i=1}^{d} V_i(x,t) f_i(u)_{x_i} = 0, \qquad u(x,0) = u_0(x). \tag{1.1}$$

An important example is nonlinear advection of a scalar quantity u in a specified velocity field $V(x,t)$ in arbitrary space dimension. If the flow is incompressible, the conservation law governing the process can be written in the non-conservative form (1.1). This type of equation is frequently occurring in simulation of flow phenomena. An important application—which was the initial motivation for this study—is the modeling of displacement of oil by water in petroleum reservoirs. In this case the flux function has a typical s-shape and the initial value problem is sometimes called a Buckley–Leverett problem. Other applications

*Received November 1997. Revised May 1999. Communicated by Petter Bjørstad.

[†]This work was partially supported by the Research Council of Norway under grant 100555/410.

are containment transport in saturated groundwater flow, atmospheric models of trace gases, and models for chemical reactors. The equation also occurs when simulation advection processes on curvilinear grids.

In the following we will develop a numerical method for (1.1). The development is strongly inspired by a dimensional splitting method proposed by Holden and Risebro [7] for the constant coefficient equation $u_t + \sum_{i=1}^{d} f_i(u)_{x_i} = 0$. Dimensional splitting decomposes the problem into a series of one-dimensional problems, one in each direction, and is a widely used technique to extend one-dimensional numerical methods to multi-dimensional problems. Holden and Risebro used a very efficient front tracking method [4] to solve the one-dimensional equations, and thus obtained a numerical method that is unconditionally stable and very efficient [13]. The front tracking method has recently [12] been extended to the variable coefficient equation

$$u_t + V(x,t) f(u)_x = 0, \qquad u(x,0) = u_0(x). \tag{1.2}$$

We use this new method as a building block for generalizing Holden and Risebro's method to the variable coefficient equation (1.1). See Ref. [11] and the references therein for other approaches, mainly for linear flux functions, for which our method coincides with the method of characteristics.

The outline of the paper is as follows. Section 2 describes the construction of approximate solutions, and thereby introduces the numerical method implicitly. In Section 3 we verify that the approximate solutions converge to the entropy weak solution. Finally, four numerical experiments are presented in Section 4: two simple linear advection problems, a nonlinear problem with time dependent velocities, and a more realistic problem coming from two–phase reservoir flow.

2 Construction of Approximate Solutions

For simplicity, the dimensional splitting method will be presented for an equation in two spatial dimensions with time independent velocity field. That is, we consider the equation

$$u_t + U(x,y) f(u)_x + V(x,y) g(u)_y = 0, \qquad u(x,y,0) = u_0(x,y). \tag{2.1}$$

It is straightforward to generalize the method to higher dimensions and time dependent velocities.

Our interest is with solutions that may contain discontinuities. Existence and uniqueness of discontinuous solutions were proved by Kružkov [10], who also proposed an *entropy condition* to identify the unique physical solution, see Section 3. Let $u(t) = \mathcal{S}(t) u_0$ denote the entropy solution of (2.1). Then dimensional splitting approximations can be defined as

$$\begin{aligned} \mathcal{S}(n\Delta t) u_0(x,y) &\cong \left[\mathcal{S}^{g,V}(\Delta t)\mathcal{S}^{f,U}(\Delta t)\right]^n u_0(x,y), && \text{(Godunov)} \\ \mathcal{S}(n\Delta t) u_0(x,y) &\cong \left[\mathcal{S}^{f,U}(\Delta t/2)\mathcal{S}^{g,V}(\Delta t)\mathcal{S}^{f,U}(\Delta t/2)\right]^n u_0(x,y), && \text{(Strang)} \end{aligned}$$

where $\mathcal{S}^{f,U}(t)u_0$ and $\mathcal{S}^{g,V}(t)u_0$ are the exact solutions of (1.2) with flux functions f and g and velocities U and V, respectively.

Here we consider numerical approximations defined on a uniform Cartesian grid with nodes $\{(i\Delta x, j\Delta y)\}$, where $\Delta x, \Delta y$ are given positive numbers and $i, j \in \mathbf{Z}$. The one-dimensional solution operators will be approximated using front tracking.

One-Dimensional Front Tracking

In the front tracking method for (1.2) we seek to construct an approximation within the class of piecewise constant functions (step functions). First the initial data u_0 are approximated by a step function. This defines a sequence of local Riemann problems. If the flux function f is approximated by a piecewise linear function, the similarity solution of each Riemann problem is again a step function. Each discontinuity (or *front*) in the Riemann solution propagates along a path $x(t)$ described by an ordinary differential equation $\dot{x} = V(x)s$, where s is the Rankine–Hugoniot velocity $s = (f(u^+) - f(u^-))/(u^+ - u^-)$. If we approximate the velocity by a piecewise linear function, each differential equation can be solved explicitly. The local Riemann problems are then connected, and we track the outgoing fronts up to the time of the first wave interaction. This interaction is a new Riemann problem which is solved by a step function, and so on. The front tracking algorithm thus consists of solving Riemann problems and tracking fronts until they collide. The method is first order and has no associated time step [12]. If the velocity is time dependent, the algorithm is essentially the same, except that special care should be taken when the velocity changes sign in time, see Lie [12]. (The velocity approximation is now, for instance, piecewise constant in time).

Dimensional Splitting

Let π be the usual grid block averaging operator defined on the uniform grid $\{(i\Delta x, j\Delta y)\}$; that is,

$$\pi u(x,y) = \frac{1}{\Delta x \Delta y} \iint_{z_{i,j}} u(\xi,\eta)\, d\xi d\eta, \qquad \forall (x,y) \in z_{i,j}, \tag{2.2}$$

where $z_{i,j}$ is grid block number (i,j) with lower left corner in $(i\Delta x, j\Delta y)$. This projection has the property that $\iint |\pi u - u| = \mathcal{O}(\max(\Delta x, \Delta y))$. For the velocity approximations U^δ, V^δ we assume that U^δ (V^δ) is piecewise linear in the x-direction (y-direction) and piecewise constant in the y-direction (x-direction). Furthermore, assume that $\|U^\delta_{j+1} - U^\delta_j\|_\infty \leq C_U \Delta y$ and $\|V^\delta_{i+1} - V^\delta_i\|_\infty \leq C_V \Delta x$ for positive constants C_U, C_V.

Now, let $\mathcal{S}^{f,U}_\delta(t)$ and $\mathcal{S}^{g,V}_\delta(t)$ denote the front tracking solution operators. Using Godunov splitting, the dimensional splitting solution $\{u^n_{i,j}\}^N_{n=1}$ is then defined as

$$u^n_{i,j}(x,y) = \left[\pi \mathcal{S}^{g,V}_\delta(\Delta t) \pi \mathcal{S}^{f,U}_\delta(\Delta t)\right]^n \pi u_0(x,y). \tag{2.3}$$

The approximate solution is defined analogously for the Strang splitting, and the extension to arbitrary space dimension is obvious.

3 Convergence to the Entropy Solution

In the following we will assume that the flux functions f_i are Lipschitz continuous and that the velocities V_i are bounded and Lipschitz continuous. To simplify the presentation, we only present the proof in two dimensions for the case where the velocities only depend on the spatial position.

The one-dimensional solutions constructed by front tracking have the three following important properties [12, Lemma 2.3]: Suppose that f is continuous and piecewise linear, V is bounded, and u_0 is a bounded step function with finite total variation. Moreover, assume that for each finite $T > 0$ there are a finite number of τ's less than T such that $\operatorname{sgn} V(x, \tau^-) \neq \operatorname{sgn} V(x, \tau^+)$, and assume that $\operatorname{sgn} V(a^+, t) = \operatorname{sgn} V(a^-, t)$ at every point of discontinuity (a, t). Then the solution of $u_t + V(x,t) f(u)_x = 0$ satisfies

(1) a maximum principle, $\|u\|_\infty \leq \|u_0\|_\infty$,

(2) Lipschitz continuity of the L^1-norm, $\|u(t_2) - u(t_1)\|_1 \leq C|t_2 - t_1|$,

(3) TVD-property, $\mathrm{TV}(u) \leq \mathrm{TV}(u_0)$.

We will now prove that the dimensional splitting solution has the same three properties. This will in turn ensure the existence of a convergent subsequence. For convenience we assume that $\Delta x = C_1 \Delta y = C_2 \Delta t$, for nonzero constants C_1, C_2.

LEMMA 3.1. *The solution $u^n(x,y)$ remains bounded in time,*

$$\|u^n(x,y)\|_\infty \leq \|u_0(x,y)\|_\infty.$$

PROOF. The one-dimensional solution operators $\mathcal{S}_\delta^{f,U}$ and $\mathcal{S}_\delta^{g,V}$ do not introduce new maxima or minima into the solution, and neither does the projection operator π. □

LEMMA 3.2. *If u_0 has compact support, then there is a positive constant C such that*

$$\sum_{ij} |u_{i,j}^m - u_{i,j}^n| \Delta x \Delta y \leq \left[C\Delta t + \mathcal{O}\big(\max(\Delta x, \Delta y)\big)\right] |m - n|.$$

PROOF. The lemma follows by induction if we can prove that

$$\sum_{ij} |u_{i,j}^{n+1} - u_{i,j}^n| \Delta x \Delta y \leq C\Delta t + \mathcal{O}\big(\max(\Delta x, \Delta y)\big).$$

One time step in our algorithm can be described as

$$u_{i,j}^n = u_j^n(0) \xrightarrow{\mathcal{S}_\delta^{f,U}} u_j^n(\Delta t) \xrightarrow{\pi} u_{i,j}^{n+1/2} = u_i^{n+1/2}(0) \xrightarrow{\mathcal{S}_\delta^{g,V}} u_i^{n+1/2}(\Delta t) \xrightarrow{\pi} u_{i,j}^{n+1}.$$

The L^1-norm of the one-dimensional front tracking solution is Lipschitz continuous in time, and hence we see that

$$\sum_k |u_j^n(\xi_k, \Delta t) - u_j^n(\xi_k, 0)|(x_{k+1} - x_k) \leq D\Delta t,$$

where $\xi_k \in [x_k, x_{k+1}]$ and the intervals $\{[x_k, x_{k+1}]\}$ are given such that *both* $u^n_j(x, \Delta t)$ and $u^n_j(x, 0)$ are constant on each interval. A similar inequality is valid for the y-sweep from $u_i^{n+1/2}(0)$ to $u_i^{n+1/2}(\Delta t)$. For convenience we assume that the compact support of u^n for all finite n is embedded in the rectangle with corners $x = \pm N\Delta x$ and $y = \pm M\Delta y$. This is a reasonable assumption, as u^0 has finite support and all waves propagate with finite speed. Using the above inequality and the properties of the projection operator now gives

$$\begin{aligned}\sum_{ij} |u^{n+1}_{i,j} - u^n_{i,j}|\Delta x\Delta y &\le \iint_{\mathbf{R}^2} |u^{n+1}_{i,j} - u_i^{n+1/2}(\Delta t)|\, dxdy \\ &\quad + \iint_{\mathbf{R}^2} |u^{n+1/2}_{i,j} - u^n_j(\Delta t)|\, dxdy \\ &\quad + \sum_{i=-N}^{N} \Delta x \sum_k |u_i^{n+1/2}(\eta_k, \Delta t) - u_i^{n+1/2}(\eta_k, 0)|(y_{k+1} - y_k) \\ &\quad + \sum_{j=-N}^{N} \Delta y \sum_k |u^n_j(\xi_k, \Delta t) - u^n_j(\xi_k, 0)|(x_{k+1} - x_k) \\ &\le C\Delta t + \mathcal{O}\big(\max(\Delta x, \Delta y)\big).\end{aligned}$$

□

LEMMA 3.3. *There is a constant C such that* $\mathrm{TV}(u^n) \le e^{Cn\Delta t}\,\mathrm{TV}(u_0)$*; that is, the total variation of the solution $u^n(x, y)$ is finite in finite time.*

PROOF. Using induction, it is sufficient to prove that $\mathrm{TV}(u^{n+1}) \le e^{C\Delta t}\,\mathrm{TV}(u^n)$. The total variation for a two-dimensional function $v(x, y)$ can be written

$$\mathrm{TV}(v(\cdot,\cdot)) = \int_{\mathbf{R}} \mathrm{TV}_x(v(\cdot, y))\, dy + \int_{\mathbf{R}} \mathrm{TV}_y(v(x, \cdot))\, dx.$$

Let us first consider the x-sweep of the algorithm. The front tracking solution is total variation diminishing by construction. If v is a step function, then $\mathrm{TV}_x(\pi v) \le \mathrm{TV}_x(v)$, and we get that

$$\sum_i |u^{n+1/2}_{i+1,j} - u^{n+1/2}_{i,j}| \le \sum_i |u^n_{i+1,j} - u^n_{i,j}|. \tag{3.1}$$

Let v denote the solution of $v_t + U(x)f(u)_x = 0$ with initial data v_0 and w the solution of $w_t + V(x)f(u)_x = 0$ with initial data w_0. Then we have that [12, Theorem 3.1]

$$\|v(\cdot, t) - w(\cdot, t)\|_{L^1(\mathbf{R})} \le e^{\gamma t}\|v_0 - w_0\|_{L^1(\mathbf{R})} + te^{\gamma t}\|f\|_{\mathrm{Lip}}\|U - V\|_\infty \mathrm{TV}(v_0),$$

for some constant γ. Now let $u^n_j(x, \Delta t)$ and $u^n_{j+1}(x, \Delta t)$ be two step functions obtained by front tracking in tube j and $j + 1$, respectively. The corresponding velocities are $U_j(x)$ and $U_{j+1}(x)$. Assume that u^n_j and u^n_{j+1} are constant on the intervals $\{[x_k, x_{k+1}]\}$. If $\xi_k \in [x_k, x_{k+1}]$, then

$$\begin{aligned}\int_{\mathbf{R}} |u^n_{j+1}(x,\Delta t) - u^n_j(x,\Delta t)|\,dx &\equiv \sum_k |u^n_{j+1}(\xi_k,\Delta t) - u^n_j(\xi_k,\Delta t)|(x_{k+1}-x_k)\\ &\le e^{\gamma\Delta t}\sum_i |u^n_{i,j+1} - u^n_{i,j}|\Delta x + \Delta t e^{\gamma\Delta t}\|f\|_{\mathrm{Lip}} C_U \Delta y \mathrm{TV}_x(u^n_{\cdot,j}).\end{aligned}$$

After applying the projection operator we have that

$$\sum_i |u^{n+1/2}_{j+1} - u^{n+1/2}_j|\Delta x = \sum_k |u^n_{j+1}(\xi_k,\Delta t) - u^n_j(\xi_k,\Delta t)|(x_{k+1}-x_k),$$

and hence

$$\begin{aligned}\sum_i |u^{n+1/2}_{j+1} - u^{n+1/2}_j|\Delta x \quad &\le \quad e^{\gamma\Delta t}\sum_i |u^n_{i,j+1} - u^n_{i,j}|\Delta x\\ &\qquad +\Delta t e^{\gamma\Delta t}\|f\|_{\mathrm{Lip}} C_U \Delta y\, \mathrm{TV}_x(u^n_{\cdot,j}).\end{aligned}\tag{3.2}$$

Summing (3.2) and (3.1) times Δy over j and adding the result gives

$$\begin{aligned}\mathrm{TV}(u^{n+1/2}) \quad &\le \quad \int_{\mathbf{R}} \mathrm{TV}_x(u^n)\,dy + e^{\gamma\Delta t}\int_{\mathbf{R}} \mathrm{TV}_y(u^n)\,dx\\ &\qquad +\Delta t\|f\|_{\mathrm{Lip}}\|U\|_{\mathrm{Lip}} e^{\gamma\Delta t}\int_{\mathbf{R}} \mathrm{TV}_x(u^n)\,dy\\ &\le \quad (1+\lambda\Delta t)e^{\gamma\Delta t}\,\mathrm{TV}(u^n).\end{aligned}$$

A similar results holds for the y-sweep. Using the fact that $(1+a) \le e^a$ for positive a gives that $\mathrm{TV}(u^{n+1}) \le e^{C\Delta t}\,\mathrm{TV}(u^n)$. □

Assume first that u^0 has compact support. Let Δ denote all the discretization parameters, $\Delta = (\Delta x, \Delta y, \delta, \Delta t)$. Then the construction in Section 2 gives a family $\{u_\Delta(t)\}$ of approximate solutions. We can now show that the sequence $\{u_\Delta(t)\}$ is convergent by using Lemmas 3.1–3.3 and Helly's theorem, see Crandall and Majda [2]. Moreover, the sequence converges in $L^1_{\mathrm{loc}}(\mathbf{R}^2 \times [0,T])$ for any $T > 0$, and the limit $u(x,y,t)$ is a solution of (2.1). Using a similar argument as in Ref. [2], we can then dispense with the assumption of compact support on the initial data.

The entropy solution of (2.1) satisfies the following Kružkov entropy condition

$$\begin{aligned}\iint_{\mathbf{R}^2}\int_0^T \Big\{\phi_t|u-k| + F(u,k)\big(U\phi\big)_x + G(u,k)\big(V\phi\big)_y\Big\}\,dxdydt&\\ -\iint_{\mathbf{R}^2}|u(T)-k|\phi(T)\,dxdy + \iint_{\mathbf{R}^2}|u(0)-k|\phi(0)\,dxdy \ge 0.&\end{aligned}\tag{3.3}$$

where $F(u,k) = \mathrm{sgn}(u-k)(f(u)-f(k))$ and $(U\phi)_x$ is used as a shorthand for the generalized derivative $\phi U_x + \phi_x U$.

We want to show that the limit solution $u(x,y,t)$ satisfies this entropy condition and that it hence is a unique entropy solution. The argument is a slight variation to that by Crandall and Majda [1]. For the approximate solutions we use the time interpolation

$$u_\Delta(x,y,t) = \begin{cases} \mathcal{S}_\delta^{f,U}\big(2(t-t_n)\big)u_{i,j}^n(x,y), & \text{for } t \in [t_n, t_{n+1/2}),\\ \mathcal{S}_\delta^{g,V}\big(2(t-t_{n+1/2})\big)u_{i,j}^{n+1/2}(x,y), & \text{for } t \in [t_{n+1/2}, t_{n+1}). \end{cases}$$

The solution obtained after one sweep in either direction is the entropy solution of an equation similar to (1.2) and thus satisfies a corresponding one-dimensional entropy condition. Note that time now goes 'twice as fast'. We therefore introduce a modified test function $\tilde{\phi}(x,y,t) = \phi(x,y,2t)$. Then $u_j^n(x,t)$ satisfies

$$\begin{aligned} 0 \quad \leq \quad & \int_{\mathbf{R}}\int_{t_n}^{t_{n+1/2}} \Big\{\frac{1}{2}\phi_t\big|u_j^n(x,t)-k\big| + F_\delta(u_j^n(x,t),k)\big(U^\delta\phi\big)_x\Big\}\,dxdt \\ & -\frac{1}{2}\int_{\mathbf{R}}\big|u_j^n(x,t_{n+1/2}^-)-k\big|\phi(x,t_{n+1/2})\,dx \\ & +\frac{1}{2}\int_{\mathbf{R}}\big|u_j^n(x,t_n^+)-k\big|\phi(x,t_n)\,dx. \end{aligned}$$

A similar inequality is valid for the y-sweep

$$\begin{aligned} 0 \quad \leq \quad & \int_{\mathbf{R}}\int_{t_{n+1/2}}^{t_{n+1}} \Big\{\frac{1}{2}\phi_t\big|u_i^{n+1/2}(y,t)-k\big| + G_\delta(u_i^{n+1/2}(y,t),k)\big(V^\delta\phi\big)_y\Big\}\,dydt \\ & -\frac{1}{2}\int_{\mathbf{R}}\big|u_i^{n+1/2}(y,t_{n+1}^-)-k\big|\phi(y,t_{n+1})\,dy \\ & +\frac{1}{2}\int_{\mathbf{R}}\big|u_i^{n+1/2}(y,t_{n+1/2}^+)-k\big|\phi(y,t_{n+1/2})\,dy. \end{aligned}$$

We integrate the two inequalities over y and x, respectively, add them, and sum over n

$$\begin{aligned} \iint_{\mathbf{R}^2}\int_0^T & \Big\{\frac{1}{2}\phi_t|u_\Delta-k| + \chi_N F_\delta(u_\Delta,k)\big(U^\delta\phi\big)_x + \\ & \qquad (1-\chi_N)G_\delta(u_\Delta,k)\big(V^\delta\phi\big)_y\Big\}\,dxdydt \\ & -\frac{1}{2}\iint_{\mathbf{R}^2}\big|u_\Delta(x,y,T)-k\big|\phi(x,y,T)\,dxdy \\ & +\frac{1}{2}\iint_{\mathbf{R}^2}\big|u_\Delta(x,y,0)-k\big|\phi(x,y,0)\,dxdy \\ \geq \frac{1}{2}\sum_{n=1}^{2N-1} & \iint_{\mathbf{R}^2}\big\{\big|u_\Delta(x,y,t_{n/2}^-)-k\big| - \big|u_\Delta(x,y,t_{n/2}^+)-k\big|\big\}\phi(x,y,t_{n/2})\,dxdy. \end{aligned}$$

Here χ_N denotes the characteristic function of $\{(x,y,t)|t_n \leq t \leq t_{n+1/2}, n = 0,\dots,N\}$ and $N\Delta t = T$. In the limit $N \to \infty$ we have that $\chi_N \to \frac{1}{2}$. The sum on the right-hand side reflects the projection operator used after each one-dimensional sweep. As the spatial discretization tends to zero, the projection operator tends to the identity operator and the right-hand side disappears. We then let the parameters δ in the approximation of flux and velocity tend to zero. By using the bounded convergence theorem for Lebesgue integrals we obtain the entropy condition (3.3). Thus we have proved the following lemma:

LEMMA 3.4. *The limit solution $u(x,y,t)$ is a weak entropy solution of (2.1); that is, it fulfills the entropy condition (3.3).*

In the case of time dependent velocities, the above analysis goes through with the obvious modifications. Similarly, extending these results to arbitrary space dimension is also straightforward. The results can be summarized in a theorem:

THEOREM 3.5. *Let u_0 be a bounded function with bounded total variation. Let f_i be Lipschitz continuous functions and $V_i(x,t)$ be bounded functions that are Lipschitz continuous in x. Moreover, assume that for each finite $T > 0$ there are a finite number of points $\tau < T$ such that $\operatorname{sgn}\big(V_i(x,\tau^-)\big) \neq \operatorname{sgn}\big(V_i(x,\tau^+)\big)$. Then the sequence of approximate solutions defined by*

$$u^n(x,t) = \big[\pi S_\delta^{f_d,V_d}(\Delta t) \cdots \pi S_\delta^{f_1,V_1}(\Delta t)\big]^n \pi u_0(x)$$

converges to the unique entropy solution of (1.1).

4 Numerical Examples

The front tracking method has been tested on various examples. Here we present four problems. The first is a simple counterclockwise rotation of a solid body. Its purpose is to illustrate the two error mechanisms in the dimensional splitting; the splitting and the projection. The second example introduces an important application of the method; solution of equations on curvilinear grids. Although both examples are simple advection problems, this is by no means a prerequisite for the method. The front tracking method was developed for, and is most successfully applied to *nonlinear* problems. Linear advection problems are in many aspects a worst case for the method, since they contain no self-sharpening to counteract the smearing introduced by the projection operator. In the third example, we therefore demonstrate the ability to take large time steps for a nonlinear equation with time dependent velocities.

The last example is from simulation of water injection in heterogenous petroleum reservoirs and has a slight touch of realism. Neglecting capillary forces, this two-phase flow process is described by a hyperbolic–elliptic system. Two quarter five-spot cases are presented; one with low-permeable barriers and one giving viscous fingering. In both cases we use a sequential splitting method to solve the nonlinear system of PDEs.

A more thorough discussion of the front tracking method applied to reservoir simulation problems can be found elsewhere. In Refs. [5, 9], front tracking constitutes an important part of a novel methodology for obtaining the correct balance of convection and diffusion in petroleum reservoirs. In Ref. [3], the front tracking method is extended to the simulation of two-phase, multi-component flow without capillary forces. This paper also demonstrates that a sequential splitting method with dimensional splitting for the saturation equation may fail to converge to the correct solution for unstable displacements.

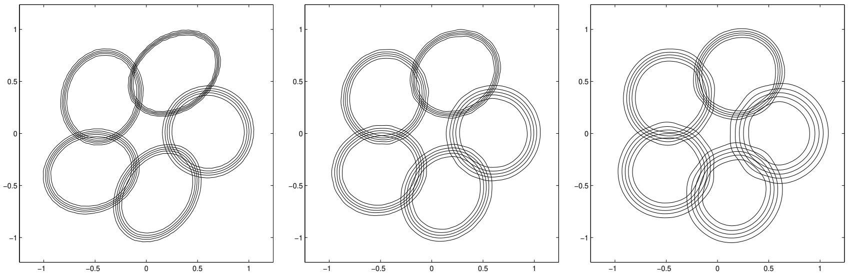

Figure 4.1: Example 1. Contour plots of the cylinder at five equally spaced points in time during the first rotation. The number of time steps is 20, 40, and 80 from left to right corresponding to CFL numbers 15.7, 7.9, and 3.9, respectively.

Example 1: Solid body rotation.

Consider solid body rotation counterclockwise around the origin in a square domain $[-1.25, 1.25] \times [-1.25, 1.25]$,

$$u_t - yu_x + xu_y = 0, \qquad u(x, y, 0) = u_0(x, y),$$

where the initial function $u_0(x, y)$ either equals

$$u_0^1(x, y) = \begin{array}{ll} 1, & \text{if } (x - 0.6)^2 + y^2 \leq 0.16 \\ 0, & \text{otherwise,} \end{array}$$

or

$$u_0^2(x, y) = \exp\bigl[-20\bigl((x - 0.6)^2 + y^2\bigr)\bigr].$$

By introducing cylindrical coordinates (see Example 2 below), this problem can be transformed to a family of one-dimensional problems and solved exactly. However, since the problem is a pure linear advection problem, it allows us to study the two factors contributing to the total error; the errors introduced by the splitting and by the projection. We have earlier pointed out that the front tracking method is unconditionally stable. Hence very large time steps can be used to reduce the number of splitting steps. Although this is favorable in terms of computer efficiency, it obviously leads to large splitting errors. On the other hand, the number of projections is reduced and hence the corresponding smearing.

In Figure 4.1 we have plotted the contours of the cylinder described by u_0^1 at five equally spaced points in time during one revolution using 20, 40, and 80 time steps. The number of unknowns in each direction is 100. The effect described above is clearly evident. With 20 time steps there is little smearing after one rotation. However, the cylinder appears rather deformed during the revolution. On the other hand, using 80 time steps gives an adequate representation of the shape throughout the rotation, but also introduces considerable smearing due to the large number of projections.

Table 4.1: Example 1. Convergence study for the discontinuous initial data u_0^1. The errors are measured after one revolution on a $n \times n$ grid.

	CFL=2.0		CFL=4.0		CFL=8.0		CFL=16.0	
n	L^1-error	rate	L^1-error	rate	L^1-error	rate	L^1-error	rate
25	5.042e-01	—	3.829e-01	—	3.203e-01	—	8.718e-01	—
50	3.578e-01	0.49	2.558e-01	0.58	1.879e-01	0.77	1.736e-01	2.33
100	2.531e-01	0.50	1.815e-01	0.49	1.293e-01	0.54	9.713e-02	0.84
200	1.797e-01	0.49	1.278e-01	0.51	9.132e-02	0.50	6.497e-02	0.58
400	1.274e-01	0.50	9.036e-02	0.50	6.424e-02	0.51	4.577e-02	0.51
800	9.030e-02	0.50	6.390e-02	0.50	4.535e-02	0.50	3.216e-02	0.51

n	L^2-error	rate	L^2-error	rate	L^2-error	rate	L^2-error	rate
25	1.748e-01	—	1.245e-01	—	1.117e-01	—	6.629e-01	—
50	1.127e-01	0.63	7.779e-02	0.68	5.766e-02	0.95	7.669e-02	3.11
100	7.694e-02	0.55	5.432e-02	0.52	3.876e-02	0.57	3.118e-02	1.30
200	5.359e-02	0.52	3.781e-02	0.52	2.702e-02	0.52	1.949e-02	0.68
400	3.765e-02	0.51	2.660e-02	0.51	1.891e-02	0.51	1.353e-02	0.53
800	2.655e-02	0.50	1.876e-02	0.50	1.332e-02	0.51	9.457e-03	0.52

Table 4.2: Example 1. Convergence study for the continuous initial data u_0^2. The errors are measured after one revolution on a $n \times n$ grid.

	CFL=2.0		CFL=4.0		CFL=8.0		CFL=16.0	
n	L^1-error	rate	L^1-error	rate	L^1-error	rate	L^1-error	rate
25	1.420e-01	—	1.023e-01	—	8.624e-02	—	2.715e-01	—
50	9.362e-02	0.60	5.976e-02	0.78	3.870e-02	1.16	5.913e-02	2.20
100	5.621e-02	0.74	3.376e-02	0.82	1.936e-02	1.00	1.703e-02	1.80
200	3.158e-02	0.83	1.797e-02	0.91	9.960e-03	0.96	6.727e-03	1.34
400	1.690e-02	0.90	9.323e-03	0.95	5.024e-03	0.99	3.132e-03	1.10
800	8.798e-03	0.94	4.764e-03	0.97	2.535e-03	0.99	1.536e-03	1.03

n	L^2-error	rate	L^2-error	rate	L^2-error	rate	L^2-error	rate
25	3.192e-02	—	1.989e-02	—	1.490e-02	—	1.022e-01	—
50	1.714e-02	0.90	8.134e-03	1.29	3.742e-03	1.99	8.180e-03	3.64
100	7.197e-03	1.25	2.895e-03	1.49	1.017e-03	1.88	7.892e-04	3.37
200	2.506e-03	1.52	8.716e-04	1.73	2.793e-04	1.86	1.283e-04	2.62
400	7.608e-04	1.72	2.427e-04	1.84	7.245e-05	1.95	2.795e-05	2.20
800	2.122e-04	1.84	6.430e-05	1.92	1.857e-05	1.96	6.715e-06	2.06

n	L^∞-error	rate	L^∞-error	rate	L^∞-error	rate	L^∞-error	rate
25	6.786e-01	—	5.501e-01	—	4.865e-01	—	9.093e-01	—
50	5.609e-01	0.27	4.060e-01	0.44	2.808e-01	0.79	3.561e-01	1.35
100	3.824e-01	0.55	2.550e-01	0.67	1.537e-01	0.87	1.289e-01	1.47
200	2.367e-01	0.69	1.454e-01	0.81	8.344e-02	0.88	5.427e-02	1.25
400	1.346e-01	0.81	7.895e-02	0.88	4.347e-02	0.94	2.484e-02	1.13
800	7.265e-02	0.89	4.139e-02	0.93	2.231e-02	0.96	1.207e-02	1.04

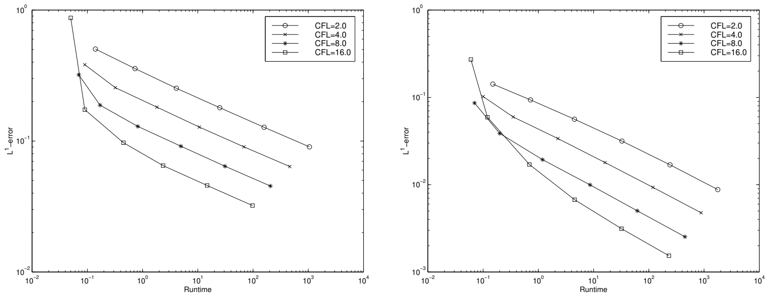

Figure 4.2: Example 1. Errors in L^1-norm after one rotation versus runtime for initial data u_0^1 (left) and u_0^2 (right).

Tables 4.1 and 4.2 give the result of a convergence study for the two initial functions u_0^1 and u_0^2. The errors in L^1 and L^2-norms are measured after one revolution and are computed using a fourth order numerical quadrature on each rectangle in the grid. Based on the measurements of the error, we can determine instantaneous convergence rates. If e_n denotes the error on a $n \times n$ grid, then the instantaneous convergence rate is defined as $r_{2n} = \log_2(e_n/e_{2n})$. For initial data u_0^1 the solution is discontinuous and convergence rates of order one half are not surprisingly low, since there is no self-sharpening effect in the linear equation to counteract the $\mathcal{O}(\Delta x^{1/2})$ smearing introduced by applying the projection operator to discontinuous data. In the smooth case, we observe first order convergence in L^1-norm, which should be expected since the one-dimensional front tracking method is first order [12]. The convergence rates are considerably higher in the L^2-norm and slightly lower in the L^∞-norm. For *nonlinear equations* convergence rates close to one can be observed also for discontinuous data, see Ref. [13].

For a fixed grid (with $n > 25$), the error decreases with increasing CFL numbers, indicating that the projection gives the major contribution to the total error for small CFL numbers. Taking runtime into account (see Figure 4.2), we can conclude that the best performance is observed for CFL number 16. The only exception is the 25 × 25 grid, where a CFL number of 16 corresponds to only 5 splitting steps, which does not resolve the dynamics of the problem completely. These observations are consistent with observations in Ref. [13].

Example 2: Curvilinear grids

The second example introduces an important application of the method—computations on curvilinear grids. For simplicity consider the linear advection equation

$$u_t + U(x,y)u_x + V(x,y)u_y = 0, \qquad u(x,y,0) = u_0(x,y),$$

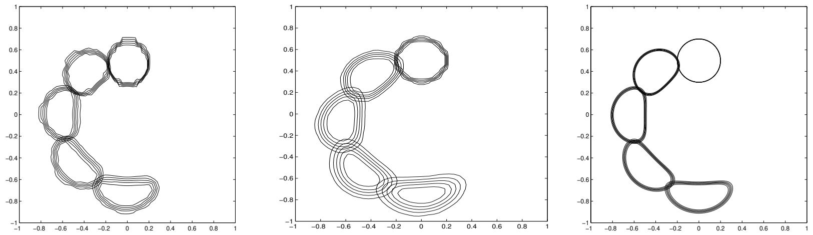

Figure 4.3: Example 2. Snapshots of solution during time interval $t \in [0, \pi]$. (Left) Cylindrical 15 × 150 grid with 4 time steps. (Middle) Cartesian 50 × 50 grid with 16 time steps. (Right) A fine grid solution on a 800 × 800 Cartesian grid.

where the velocities are superpositions of the rotation from Example 1 and a radial movement ($a = 0.05$)

$$U(x,y) = -y + \frac{ax}{x^2+y^2}, \qquad V(x,y) = x + \frac{ay}{x^2+y^2}.$$

The initial function is equal one inside a circle with radius 0.2 centered at $(0, 0.5)$ and zero elsewhere. The cylinder will rotate around the origin and at the same time be deformed due to the decreasing radial velocity component. Now, introduce cylindrical coordinates (r, θ) by the coordinate transform

$$x = r\cos\theta, \qquad y = r\sin\theta.$$

In the new coordinates the equation becomes

$$u_t + R(r,\theta)u_r + \Theta(r,\theta)u_\theta = 0,$$

where

$$\begin{aligned} R(r,\theta) &= \cos\theta\, U(r\cos\theta, r\sin\theta) + \sin\theta\, V(r\cos\theta, r\sin\theta) \\ \Theta(r,\theta) &= r^{-1}\sin\theta\, U(r\cos\theta, r\sin\theta) + r^{-1}\cos\theta\, V(r\cos\theta, r\sin\theta). \end{aligned}$$

We thus get the simple advection equation $u_t + u_r + (a/r)u_\theta = 0$. To avoid the singularity at the origin, the computational domain is chosen equal $[0.2, 1.0] \times [0, 2\pi]$, and we impose periodic boundary conditions at $\theta = 0$ and $\theta = 2\pi$. Figure 4.3 shows five snapshots of the solution in the time interval $t \in [0, \pi]$ computed on a 15 × 150 grid with 4 time steps and on a 50 × 50 Cartesian grid with 16 time steps. For reference, a solution computed on a fine Cartesian grid has been included. Not surprisingly, we observe that the curvilinear grid gives a superior resolution of the movement of the cylinder.

Example 3: Time dependent velocities

In the two previous examples, we have studied linear equations with time independent velocities and observed that the front tracking method has the

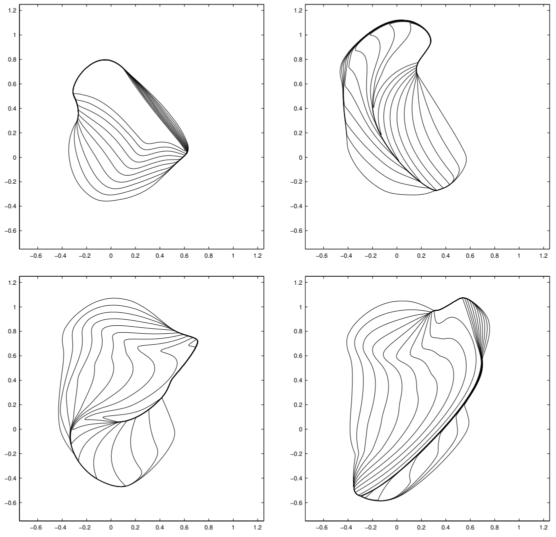

Figure 4.4: Example 3. Contour plots of the solution at times t=0.5, 1.0, 1.5, and 2.0 (from upper left to lower right) computed on a 800 × 800 grid using 400 time steps.

ability to take large time steps. A natural question to ask, is whether this ability carries over to nonlinear equations and time dependent velocities. Consider therefore a variation over Burgers' equation

$$u_t + U(y,t)(u^2)_x + V(x,t)(u^2)_y = 0, \qquad u(x,y,0) = u_0(x,y),$$

where

$$U(y,t) = \cos\bigl(\pi(y+t)\bigr), \qquad V(x,t) = \sin\bigl(\pi(x+t)\bigr)$$

and u_0 is equal one inside a circle with radius 0.4 centered at the origin and zero elsewhere. Figure 4.4 shows the solution at four distinct times up to $t = 2.0$.

To investigate the splitting errors, we use a fixed grid and vary the number of splitting steps in powers of 2. Table 4.3 gives the L^1-errors at four different times, measured relative to a fine grid solution. To reduce the spatial error contribution, we have used a 200 × 200 grid. By increasing the number of splitting steps, we get a better approximation of the time dependent velocity field and hence it is not surprising that the error decreases with decreasing CFL. At the same time, the contribution from the projection error increases and leads to an increase in the total error at 256 splitting steps. It is also worthwhile to consider the *qualitative* behavior of the solution as in Figure 4.5. With only 6 splitting steps (corresponding to a CFL number of 50!), the major features in the solution are reproduced. When the number of steps is increased to 24 (CFL number 12.4), we obtain a fairly accurate solution.

Since the front tracking method is able to utilize large time steps, it has very high *computational efficiency.* To support this claim, we make a comparison

Table 4.3: Example 3. Errors in L^1-norm relative to a fine grid solution (800 × 800 with 400 steps) for runs with n time steps on a 200 × 200 grid.

n	CFL	$t = 0.5$	$t = 1.0$	$t = 1.5$	$t = 2.0$
8	49.8	7.663e-02	5.233e-02	3.202e-02	4.148e-02
16	24.9	4.055e-02	2.850e-02	1.636e-02	2.327e-02
32	12.4	2.023e-02	1.522e-02	8.317e-03	1.208e-02
64	6.2	9.976e-03	8.318e-03	5.082e-03	6.639e-03
128	3.1	5.795e-03	6.574e-03	5.549e-03	6.214e-03
256	1.6	6.570e-03	8.617e-03	8.512e-03	9.526e-03

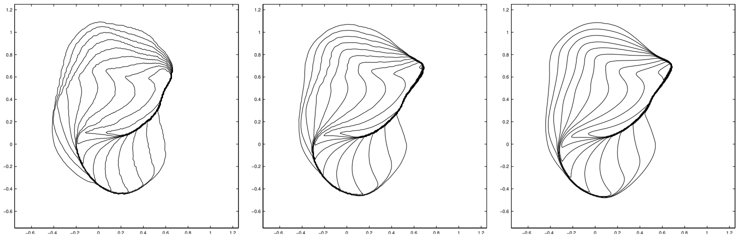

Figure 4.5: Example 3. Contour plots of the solution at time t=1.5 computed on a 200 × 200 grid with 6 (left), 24 (middle), and 96 time steps (right).

with two finite difference methods; a standard first-order, upwind method and a nonoscillatory, second-order, central difference scheme [8]. Figure 4.6 shows a plot of numerical errors in L^1-norm at time $t = 1.0$ versus runtime for the three methods. The numerical error is estimated relative to a solution computed on a 1000 × 1000 grid by the central difference scheme. We used a CFL number of 0.475 for both finite difference methods and CFL 4.0 for front tracking. The front tracking method is obviously the most efficient. Similar comparisons were reported in Ref. [13] for problems with constant coefficients.

Example 4: Water injection in oil reservoirs.

Consider a horizontal reservoir where the flow of oil and water is assumed to be immiscible and incompressible and capillary forces are neglected. We choose total Darcy velocity V, total fluid pressure p, and water saturation s as primary variables. Then the displacement of oil by water is governed by a coupled system of nonlinear partial differential equations. We have the pressure equation

$$\nabla\big((\lambda_w + \lambda_o)\nabla p\big) = 0,$$

where λ_w and λ_o are the mobilities (relative permeability k_{ri} divided by viscosity μ_i) of water and oil and are assumed to be known functions of the saturation. As a simple model we use $\lambda_w(s) = s^2/\mu_w$ and $\lambda_o(s) = (1-s)^2/\mu_o$.

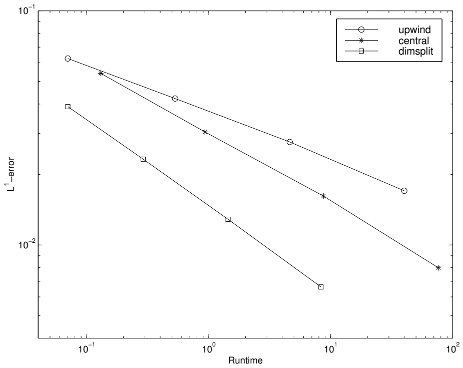

Figure 4.6: Example 3. Numerical error versus runtime for the three methods on a sequence of grids with $25 \cdot 2^k$ $(k = 0, \ldots, 3)$ unknowns in each direction.

The elliptic pressure equation is coupled via the total Darcy velocity $V = -(\lambda_w + \lambda_o)\nabla p$ to the hyperbolic saturation equation

$$s_t + V \cdot \nabla f(s) = 0, \qquad f = \frac{\lambda_w}{\lambda_w + \lambda_o}$$

where the $f = f(s)$ is the fractional flow function. The system is constrained by appropriate initial and boundary conditions; we assume $s = 0$ initially and $V \cdot n = 0$ on the boundary. In the saturation equation, the production wells are modeled as outflow boundaries and the injection wells as inflow boundaries with a specified saturation value.

A sequential time stepping procedure is used to decouple the equations. First the pressure equation is solved with the initial saturation as input to generate a total velocity field. Subsequently, the velocity field is held constant when solving the saturation equation. Then the pressure is recomputed, and so on. This strategy is common and reflects the different nature of the two equations; see Ref. [14] for an analysis of this procedure.

To discretize the pressure equation, we have used a nine–point, finite–difference method on a block–centered grid as described by Yanosik and McCracken [15] with harmonic averaging of the mobilities. The saturation equation is solved by the dimensional splitting approach outlined above. To avoid massive material balance errors in near-well regions, the CFL number is reset to 1.0 at water breakthrough, see [3, 5] for more details.

The first case is a quarter five-spot with three low-permeable blocks, see Figure 4.7. We ran two simulations; in the first simulation we used CFL number 8.0 and 16 pressure updates up to final time $t = 0.8$. In the second simulation we used CFL number 2.0 and 160 pressure updates. The viscosity ratio $\mu_o : \mu_w$ is equal $5 : 1$. The saturation contours are almost equal for the two simulation-s, except that the first simulation gives slightly sharper fingers and and better resolution of the discontinuous water front.

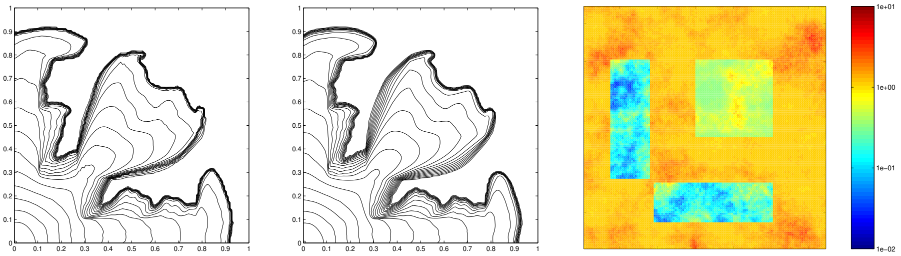

Figure 4.7: Example 4. Saturation contours at time $t = 0.35$ computed on a 129 × 129 grid in simulation 1 (left) and simulation 2 (middle). (Right) permeability field.

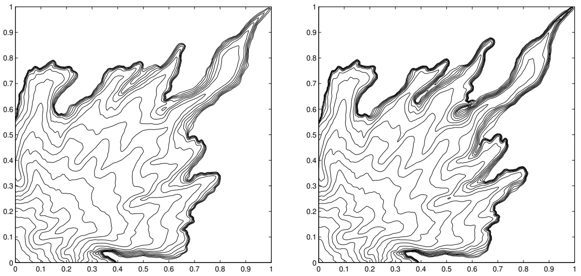

Figure 4.8: Example 4. Saturation contours at time $t = 0.25$ computed with a CFL number of 2.0 (left) and 8.0 (right).

Table 4.4 gives various statistics for the two simulations. Although the times to water breakthrough differ slightly, the oil and water productions are almost similar. The difference in times to breakthrough is due to the different accuracy in the sequential splitting of the equations. The higher material balance error in the first simulation is caused by the higher CFL number. To support this, we have included the statistics of a third simulation with CFL number 8.0 and 160 pressure updates.

The method gives rather sharp resolution of viscous fingers, as is demonstrated in the second case. Here we have used a random permeability field on a 257 × 257 grid and viscosity ratio $\mu_o : \mu_w$ equal $8 : 1$. Fifty sequential splitting steps were used to reach final time $t = 0.25$. Figure 4.8 shows saturation contours obtained with fixed CFL numbers 2.0 and 8.0. The numerical dissipation is less in the latter simulation and hence the fingers are sharper.

Final remarks

The strength of the front tracking method is the ability to use large time steps. The philosophy of the method is that the splitting step should be restricted by the *dynamics* of each problem, and not by the spatial discretization. When large

Table 4.4: Example 4. Various statistics for the first quarter five-spot case; w.b.t = time to water breakthrough, m.b.e = maximum material balance error in percent relative to the initial amount of oil inplace.

	oil prod.	water prod.	w.b.t.	m.b.e
simulation 1	0.5885	0.2115	0.411	0.187
simulation 2	0.5844	0.2156	0.399	0.034
simulation 3	0.5860	0.2140	0.401	0.257

time steps can be used, the method delivers solutions with surprising accuracy (for a first-order method) and high efficiency. Numerical examples supporting these claims have been presented in the current and other studies, e.g., [3, 5, 6, 9, 13].

Acknowledgement

I thank Vidar Haugse, Helge Holden, Kenneth Hvistendahl Karlsen, and Nils Henrik Risebro for many fruitful discussions.

REFERENCES

1. M. G. Crandall and A. Majda. *The method of fractional steps for conservation laws.* Numer. Math., 34 (1980), pp. 285–314.
2. M. G. Crandall and A. Majda. *Monotone difference approximations for s-calar conservation laws.* Math. Comp., 34 (1980), pp. 1–21.
3. V. Haugse, K. H. Karlsen, K.–A. Lie, and J. R. Natvig. *Numerical solution of the polymer system by front tracking.* Preprint, 1999.
4. H. Holden, L. Holden, and R. Høegh–Krohn. *A numerical method for first order nonlinear scalar conservation laws in one-dimension.* Comput. Math. Applic., 15 (1988), pp. 595–602.
5. H. Holden, K. H. Karlsen, and K.–A. Lie. *Operator splitting methods for degenerate convection–diffusion equations II: numerical examples with emphasis on reservoir simulation.* Preprint, 1999.
6. H. Holden, K.–A. Lie, and N. H. Risebro. *An unconditionally stable method for the Euler equations.* J. Comp. Phys., 150 (1999), pp. 76–96.
7. H. Holden and N. H. Risebro. *A method of fractional steps for scalar conservation laws without the CFL condition.* Math. Comp., 60 (1993), pp. 221–232.
8. G.–S. Jiang and E. Tadmor. *Nonoscillatory central schemes for multidimensional hyperbolic conservation laws.* SIAM J. Sci. Comput., 19 (1998), pp. 1892–1917.

9. K. H. Karlsen, K.–A. Lie, N. H. Risebro, and J. Frøyen. *A front-tracking approach to a two-phase fluid-flow model with capillary forces.* In Situ, 22 (1998), pp. 59–89.

10. S. N. Kružkov. *First order quasi-linear equations in several independent variables.* Math. USSR Sbornik, 10 (1970), pp. 217–243.

11. R. J. LeVeque. *High-resolution conservative algorithms for advection in incompressible flow.* SIAM J. Numer. Anal., 33 (1996), pp. 627–665.

12. K.–A. Lie. *A front tracking method for one-dimensional nonlinear advection equations with variable coefficients.* Preprint. Mathematics No. 16, NTNU, Trondheim, Norway, 1997.

13. K.–A. Lie, V. Haugse, and K. H. Karlsen. *Dimensional splitting with front tracking and adaptive grid refinement. Numer. Methods Partial Differential Equations*, 14 (1998), pp. 627-648.

14. H. J. Schroll and A. Tveito. *Local existence and stability for a hyperbolic–elliptic system modeling two–phase reservoir flow.* Preprint 136, RWTH Aachen, 1997.

15. J. L. Yanosik and T. A. McCracken. *A nine point finite difference simulator for realistic predictions of adverse mobility ratio displacements. SPE Journal*, 1979, pp. 252–262.