JOURNAL OF COMPUTATIONAL PHYSICS 47, 211–228 (1982)

A Front Tracking Method Applied to Burgers' Equation and Two-Phase Porous Flow*

PER LÖTSTEDT†

Lawrence Berkeley Laboratory, University of California, Berkeley, California 94720

Received September 30, 1981

A method is presented that is capable of following discontinuities in the solution of hyperbolic partial differential equations. At every time step for each cell in the neighborhood of the discontinuity, the fraction of the cell lying behind the discontinuity curve is updated. From this data the front is reconstructed. The method is applied to three scalar differential equations: inviscid Burgers' equation, the Buckley–Leverett equation for immiscible porous flow, and the equation for two-phase miscible flow in a porous medium.

1. INTRODUCTION

Discontinuities in the solution of hyperbolic partial differential equations appear in many physical applications, e.g., gas dynamics, flame propagation, and petroleum reservoir simulation. Methods to follow these time-dependent fronts in two space dimensions have been described by Richtmyer and Morton [17] and more recently by Noh and Woodward [15], Chorin [3], Glimm *et al.* [11], and Hirt and Nichols [14]. The last reference contains a review of several front tracking techniques. In [11, 17], the discontinuity curve is approximated by a number of points. The front is advanced in time by moving the points with the front speed, either in the direction normal to the front [17] or along the characteristics [11]. Another way of representing the front is suggested in [3, 14, 15]. For each mesh cell covering the domain of the solution, the fraction F of the cell that lies behind the front is stored. Fraction F satisfies $0 \leqslant F \leqslant 1$ and consequently, $1 - F$ is the fraction of the cell that lies ahead of the front. Using this information, an approximate line of discontinuity can be constructed. The fractions are transported by a velocity field $v = (v_x, v_y)$ determined by the position of the front and the particular hyperbolic equation to be solved. The fractions are moved in two steps: one in the x direction using v_x and one in the y direction using v_y. At each step, the part of the volume behind the front in every cell

* This work was supported in part by the Director, Office of Energy Research; Office of Basic Energy Sciences; Engineering, Mathematical, and Geosciences Division of the U. S. Department of Energy under Contract No. W-7405-ENG-48.

† Present address: Department of Numerical Analysis and Computing Science, Royal Institute of Technology, S-100 44 Stockholm, Sweden.

0021-9991/82/080211-18$02.00/0

581/47/2-4

that flows over a grid line to an adjacent cell is determined. Then the fractional values are updated. The approach taken here is a development of these latter ideas presented in [3, 15].

The front tracking method is applied here to the numerical solution of three examples of scalar hyperbolic partial differential equations (PDEs),

$$u_t + \nabla \cdot (qf(u)) = 0, \tag{1.1}$$

with discontinuous initial data on the domain $(x, y) \in \Omega = [0, 1] \times [0, 1]$. The front speed is given by the Rankine–Hugoniot condition and the velocity field $q = (q_x, q_y)$. The three hyperbolic PDEs are inviscid Burgers' equation, the Buckley–Leverett equation, and the equation for porous, two-phase miscible flow.

(1) Inviscid Burgers' equation

$$u_t + \nabla \cdot (\tfrac{1}{2}qu^2) = 0, \qquad q(x, y) = (1, 1). \tag{1.2}$$

(2) The Buckley–Leverett equation

$$u_t + \nabla \cdot (qf(u)) = 0,$$

$$\partial u/\partial n = 0 \quad \text{on} \quad \partial\Omega, \quad \text{the boundary of} \quad \Omega, \qquad u(t, 0, 0) = 1, \tag{1.3a}$$

$$q = -\lambda(u)\nabla p, \qquad \nabla \cdot (\lambda(u)\nabla p) = g, \qquad \partial p/\partial n = 0 \quad \text{on} \quad \partial\Omega. \tag{1.3b}$$

The fractional flow function $f(u)$ and the total mobility $\lambda(u)$ are taken to be

$$f(u) = u^2/(u^2 + a(1-u)^2), \qquad a \in (0, 1], \qquad \lambda(u) = u^2 + a(1-u)^2. \tag{1.3c}$$

(3) The equation for porous, two-phase, miscible flow,

$$u_t + \nabla \cdot (qu) = 0, \qquad \partial u/\partial n = 0 \quad \text{on} \quad \partial\Omega, \qquad u(t, 0, 0) = 1. \tag{1.4a}$$

The value of q satisfies Eq. (1.3b) and $\lambda(u)$ has the definition

$$\lambda(u) = (u + \beta(1-u))^4, \qquad \beta > 0. \tag{1.4b}$$

The inital data for Burgers' equation are such that the solution contains both shocks and rarefaction waves; see the numerical results in Section 4. The saturation u for the flow of two immiscible, incompressible fluids through a homogeneous porous medium satisfies the Buckley–Leverett equation if the effects of gravity and capillary pressure are negligible. The quantity p is the pressure and a is the viscosity ratio between the two phases. Then we have $g(x, y) = 0$ except for a source at $(0, 0)$ and a sink at $(1, 1)$, each of unit strength. In $\tilde{\Omega} = \Omega \backslash \{(0, 0), (1, 1)\}$, we have $\nabla \cdot q = 0$. Therefore, the solutions u of (1.3a) and (1.4a) satisfy

$$u_t + q \cdot \nabla f(u) = 0 \tag{1.5}$$

in $\tilde{\Omega}$, where f is defined by (1.3c) and $f(u) = u$, respectively. Equations (1.3) and (1.4) are of importance in the simulation of petroleum reservoirs, see Peaceman [16].

The hyperbolic equations (1.2), (1.3a), and (1.4a) are solved by operator splitting. Glimm's method or the upstream difference method combined with Godunov's method is used for the integration of the continuous parts of the solution. The solution to the elliptic equation (1.3b) for the porous flow examples is obtained at each time step by a finite difference approximation.

2. The Front Tracking Method

Place a uniform square grid of mesh spacing h in the x and y directions over the domain Ω of interest. Suppose that a value of v is available everywhere in Ω. For each mesh cell (i,j), define the value F_{ij} to be the fraction of the cell that is behind the front. The discussion is simplified if the region behind the front henceforth is identified as the *black fluid* and the region ahead of the front is identified as the *white fluid.*

At each time step, the fractions in the mesh cells are moved using v in two one-dimensional steps, first in the x direction and then in the y direction. A local approximation of the interface between black and white fluid is performed in each cell (i,j) for which $0 < F_{ij} < 1$ before the fluid in the cell is moved in a coordinate direction. This approximation is based on F in the neighboring cells and is not necessary for the majority of the cells for which $F_{ij} = 0$ or $F_{ij} = 1$. The interpretation of the front in the x direction may be different from that in the y direction.

In order to construct a local front in a cell (i, j) for a step in the horizontal direction, it is sufficient to consider the three cases in Fig. 1. The conditions on F in the neighboring cells in each case are:

Case 1 (Fig. 1.1).

$$F_4 \neq 0 \qquad \text{and} \qquad F_5 = 0$$

and

$$((F_7 \neq 0 \text{ and } (F_2 \neq 0 \text{ or } F_3 \neq 0)) \text{ or}$$
$$(F_2 \neq 0 \text{ and } (F_7 \neq 0 \text{ or } F_8 \neq 0))).$$

Case 2 (Fig. 1.2).

$$((F_4 \neq 0 \text{ and } F_5 \neq 0) \text{ or } F_4 = F_5 = 0) \quad \text{and} \quad F_7 > F_2.$$

Case 3 (Fig. 1.3).

$$F_4 \neq 0 \quad \text{and} \quad F_7 \neq 0 \quad \text{and} \quad F_2 = F_3 = F_5 = 0.$$

The other nine possible configurations satisfy conditions that are easily derived by

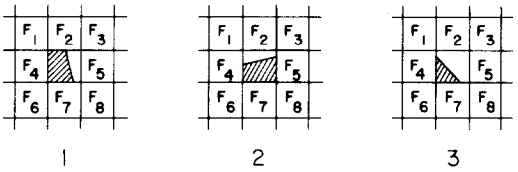

FIG. 1. Three different cases in the procedure for transportation of the fractional volumes.

moving the triangle in Case 3 to the other three corners and by interchanging the role of black and white. If none of the cases above applies, then Case 1 is chosen if $F_4 \geqslant F_5$, and if $F_5 > F_4$, the opposite case is chosen.

The slope of the interface is determined as follows:

Case 1. The upper and lower edge, x_u and x_d, of the trapezoid are proportional to F_2 and F_7,

$$x_u/x_d = (F_2/F_7)^z, \qquad z \in [0, 1]. \tag{2.1}$$

If $F_2 = F_7 = 0$, then $x_u/x_d = 1$. Furthermore, x_u and x_d satisfy the area condition

$$0.5h(x_u + x_d) = h^2 F_{ij}. \tag{2.2}$$

If Eqs. (2.1) and (2.2) imply that $x_u > h$ ($x_d > h$), then solve Eq. (2.2) for x_d (x_u) with $x_u = h$ ($x_d = h$).

Case 2. This case is equivalent to Case 1 with F_2 and F_7 in Eq. (2.1) replaced by F_4 and F_5.

Case 3. Let a right triangle T have its two perpendicular sides on the grid lines and the right angle at the lower left corner of the cell (i, j). Let x_T and y_T be the lengths of the sides in the x and y directions, respectively. Compute x_T and y_T such that

$$x_T/y_T = (F_7/F_4)^z, \tag{2.3}$$

$$0.5x_T y_T = h^2 F_{ij}. \tag{2.4}$$

If $x_T \leqslant h$ and $y_T \leqslant h$, then take x_d and y_l, the left edge of the triangle in Fig. 1.3, to be x_T and y_T. Otherwise, determine T such that Eq. (2.3) is satisfied and the area of T inside the cell is $h^2 F_{ij}$. This polygonal area contains the black fluid.

The value of z in Eqs. (2.1) and (2.3) is 0, 0.5, or 1.0 in the numerical examples. Noh and Woodward [15] allow only interfaces parallel to the grid. Chorin [3] introduces black and white rectangles inside the mesh cells to increase the accuracy. With $z = 0$, the local front line in Cases 1 and 2 is parallel to the mesh.

When the interface between the black and white fluid has been established, the fluids in a cell are advected with the velocity $v_x(x, y)$. The time step taken at time t_n to advance the front is k.

Case 1. Let v_u and v_d be the values of v_x at t_n along the upper and lower mesh line, respectively. Then

$$x_u(t_n + k) = x_u(t_n) + v_u k, \qquad x_d(t_n + k) = x_d(t_n) + v_d k.$$

If $x_u(t_n + k) > h$ or $x_d(t_n + k) > h$, then determine the volume of black fluid ΔF_{ij} that has entered the right cell and update F_{ij} and $F_{i,j+1}$,

$$F_{ij}(t_n + k) = F_{ij}(t_n) - \Delta F_{ij}, \qquad F_{i,j+1}(t_n + k) = F_{i,j+1}(t_n) + \Delta F_{ij}.$$

Proceed in a similar manner if $x_u(t_n + k) < 0$ or $x_d(t_n + k) < 0$. If $v_l = 0.5(v_u + v_d) < 0$, then the contribution to the left cell is $-v_l k$. If $F_{i,j-1} = 1$ and $v_l > 0$, then the cell (i, j) receives $\Delta F_{ij} = v_l k$ from its left-hand neighbor.

Case 2. Let v_l and v_r be the values of v_x at the left and right boundaries of the cell. Depending on the signs of v_l and v_r, the original trapezoid is stretched or contracted and advected. The parallel edges remain constant. Then the possible contributions to the cells $(i, j-1)$ and $(i, j+1)$ are computed.

Case 3. Stretch or contract the triangle with v_u at the left edge and v_d at the lower right corner. Then decide if any black fluid has moved into the left or right cell.

For each cell, F_{ij} must fulfill $0 \leqslant F_{ij} \leqslant 1$. Hence, if $F_{ij}(t_n + k) > 1$, then $F_{ij}(t_n + k) = 1$, and if $F_{ij}(t_n + k) < 0$, then $F_{ij}(t_n + k) = 0$. To prevent the black fluid in a cell from flowing in one direction into more than one neighboring cell, k must satisfy the restriction

$$k \leqslant h/|v|. \tag{2.5}$$

The domain Ω of the solution in the numerical experiments is $[0, 1] \times [0, 1]$ with Neumann boundary conditions or with the directional derivative $\partial u/\partial s = 0$ on $\partial\Omega$, $s = x - y$. The grid on Ω is augmented by extra rows and columns so that the area $[-h, 1 + h] \times [-h, 1 + h]$ is covered. When one sweep in a coordinate direction is completed, the values of F in the extra cells outside of Ω are determined by reflection. Thus, all cells in Ω have a left and right neighbor as required by the algorithm.

In order to compute the solution of the differential equation using the information on the front available from the front tracking method, the position of the discontinuities on the grid lines must be calculated. If the front crosses a horizontal line with black fluid to the left and white fluid to the right, two possibilities are distinguished, see Fig. 2.

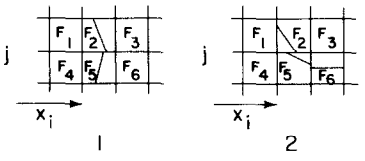

FIG. 2. Two different cases for determination of the point of discontinuity on grid line j.

Case 1 (Fig. 2.1). $F_3 = F_6 = 0$, $F_k \neq 0$, $k = 1, 2, 4, 5$. The discontinuity point on line j is approximated by

$$x = x_i + 0.5h(F_2 + F_5). \tag{2.6}$$

Case 2 (Fig. 2.2). $F_3 = 0$, $F_k \neq 0$, $k = 1, 2, 4, 5, 6$. The approximation of the discontinuity point on line j is

$$x = x_i + 0.5h(F_2 + F_5 + F_6). \tag{2.7}$$

By using Eq. (2.7), a mesh point lying behind the front in the coordinate direction will also do that in the other direction. When the order of black and white at the front is reversed the formulas (2.6) and (2.7) are modified correspondingly.

The coordinates of the discontinuities and their character (black-left or black-right) are stored for each line in the horizontal and vertical directions in vectors for later use when the differential equation is solved. The coordinates are connected to form the front curve in the graphical output. Knowledge of the character of a discontinuity on a line simplifies the task of following its path in time.

It is possible that, e.g., a previous shock wave is replaced by a rarefaction wave due to a reversal of the flow or, as in two of the examples, that an initial discontinuity develops into a rarefaction wave. Whether such a wave starts in a cell (i,j) or not is determined by the hyperbolic PDE and the normal to the constructed front line. The contributions to neighboring cells ΔF_{ij} are computed as before, but F_{ij} is held constant.

Computational experience (cf. Fenimore [8]) indicates that the zone of cells at the front for which $0 < F_{ij} < 1$ is seldom more than two cells wide. Thus, the number of front cells for which calculations are necessary at each time step is $O(1/h)$. The front tracking method requires extra memory to store the $O(1/h^2)$ components of F_{ij}. This number could be reduced to $O(1/h)$ by storing the value F_{ij} and the indices i and j of only the cells in the neighborhood of the front at the cost of a more complicated program. Evidently the formulas (2.6) and (2.7) for computing the points of discontinuity on a grid line are at least $O(h)$ accurate for exact values F_{ij}. For straight lines, however, the error is sometimes proportional to h. Hence, the accuracy of Eqs. (2.6) and (2.7) is $O(h)$. Fenimore [8] demonstrates that the magnitude of the errors introduced in F_{ij} by the method of Noh and Woodward [15] in a special case is $O(h)$.

With a simplified procedure for computing the local interfaces, the method can be

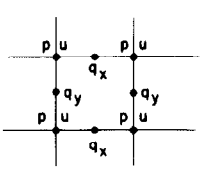

FIG. 3. The approximate values of u, p, and q are computed at mesh points and grid lines.

extended to problems in three spatial dimensions. The discontinuity point on a grid line is then determined by the values of F in the four surrounding rows of grid cubes.

The approximate values of u and q in Eq. (1.1), u^n and q^n, are known at grid points and on grid lines as displayed in Fig. 3. For Eq. (1.1), the velocity field v for advection of the front cells is given by q and the Rankine–Hugoniot condition. In each of the three cases in Fig. 1, v_x is computed as follows:

Case 1 (Fig. 1.1). Let u_0 and u_1 be the values of u^n surrounding the closest discontinuity on the upper grid line. Furthermore, let q_* be the value of q_x on the upper edge of the cell (i,j). Then

$$v_u = q_*(f(u_1) - f(u_0))/(u_1 - u_0). \tag{2.8}$$

We determine v_d in a similar way.

Case 2 (Fig. 1.2). Let u_0 and u_1 be the values of u^n surrounding the closest discontinuity on the left grid line. Let q_* be the value of q_x on the lower edge of the cell. Then v_l is given by the same expression as v_u in Eq. (2.8). We calculate v_r from u_0 and u_1 on the right grid line and q_*.

Case 3 (Fig. 1.3). Case 3 is equivalent to Case 1.

3. Time-Integration of the Differential Equations

The values of p^n, $q^n = (q_x^n, q_y^n)$, and u^n at the points depicted in Fig. 3 are calculated by the following procedures: In the two-phase flow equations, p^n satisfies a finite difference approximation of the elliptic equation (1.3b). The derivative $\partial(\lambda(u)\,\partial p/\partial x)/\partial x$ is replaced by

$$(\lambda_{i+1/2,j}(p_{i+1,j} - p_{ij}) - \lambda_{i-1/2,j}(p_{ij} - p_{i-1,j}))/h^2, \tag{3.1}$$

where $\lambda_{i+1/2,j} = 0.5(\lambda(u_{ij}^n) + \lambda(u_{i+1,j}^n))$, or if u is discontinuous between x_i and x_{i+1}, then let $u_*^n = \max(u_{ij}^n, u_{i+1,j}^n)$ and $\lambda_{i+1/2,j} = \lambda(u_*^n)$. This choice of u_*^n is particularly suitable here since a thin finger of the least viscous fluid with width less than h containing the point (i,j) will have influence on λ and p. We define $\lambda_{i-1/2,j}$ in a similar manner and the counterpart of Eq. (3.1) in the y direction is derived analogously. The resulting system of linear equations is solved by the preconditioned conjugate gradient method, as described in Albright and Concus [1]. The values of q_x^n and q_y^n are approximated by standard difference formulas.

Assume that u^n and q^n are known at t_n. At $t_{n+1} = t_n + k$, u^{n+1} is obtained in two steps by operator splitting of (1.5), see Richtmyer and Morton [17]. First solve

$$u_t + q_x \frac{\partial}{\partial x}(f(u)) = 0, \tag{3.2}$$

followed by

$$u_t + q_y \frac{\partial}{\partial y}(f(u)) = 0. \tag{3.3}$$

The algorithm for computing $u^{n+1/2}$ from Eq. (3.2) in a sweep in the horizontal direction is:

(i) Determine the old points of discontinuity on all the lines in the horizontal direction.

(ii) Transport the black and white fluids using v_x and the method described in the previous section.

(iii) Determine the new points of discontinuity on all the lines in the horizontal direction.

(iv) Compute the continuous parts of $u^{n+1/2}$ on each line by a temporal one-step method, taking both the old and new points of discontinuity into account.

With $u^{n+1/2}$ as input in Eq. (3.3), u^{n+1} is computed in a sweep in the vertical direction. In three space dimensions, an additional step in the z direction is taken.

The last step above requires further comments. The methods chosen in the examples are:

(1) the upstream difference method combined with Godunov's method (see Richtmeyer and Morton [17], Colella [5]), and

(2) a modified Glimm's method (also termed piecewise sampling method, random choice method, uniform sampling method) without a staggered grid due to Colella [5].

The random choice method for the solution of the Buckley–Leverett equation (1.3) was first used by Concus and Proskurowski [7]. The upstream difference formula used here for advancing the solution of Eq. (1.1) one time step in the x direction is

$$u_{ij}^{n+1/2} = u_{ij}^n - (k/h)\, q_{i-1/2r,j}^n (f(u_{ij}^n) - f(u_{i-r,j}^n)), \tag{3.4}$$

where $r = 1$ (resp. $r = -1$) if $q_{i-1/2,j}^n$ and $q_{i+1/2,j}^n$ are both positive (resp. negative). If $q_{i-1/2,j}^n$ and $q_{i+1/2,j}^n$ have different signs, then $u_{ij}^{n+1/2}$ is computed with Godunov's method. The upper bound for the step size k for stability in Eq. (3.4) and Godunov's method is

$$k \leqslant h/|qf'(u)|, \tag{3.5}$$

and in Glimm's method

$$k \leqslant 0.5h/|qf'(u)|. \tag{3.6}$$

Upstream differencing and Godunov's method are first-order accurate [17] and the

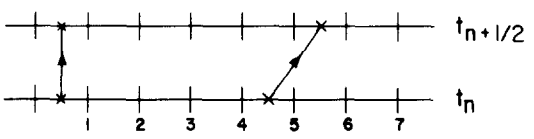

FIG. 4. A discontinuity moves on a grid line when time increases. A mesh point is marked by (|) and a discontinuity by (×).

order of Glimm's method approaches first order, Colella [4]. Equations (3.2) and (3.3) specify a first-order splitting scheme.

Consider a part of a grid line in Fig. 4 with two discontinuities at t_n between which the solution is continuous. A value u_5^* is extrapolated from u_3^n and u_4^n,

$$u_5^* = u_4^n + (u_4^n - u_3^n). \tag{3.7}$$

The values u_i^n, $i = 1,..., 4$, and u_5^* are then used by one of the above methods to produce $u_i^{n+1/2}$, $i = 1,..., 5$. Then $u_i^{n+1/2}$, $i = 6, 7,...$, are determined by u_i^n, $i = 5, 6,...$. The modified Glimm's method does not always provide solution values at boundary points or mesh points close to a discontinuity. The missing values are calculated with Eq. (3.4). At a discontinuity point, q is assumed to have constant sign so that Eq. (3.4) can be used. For the equation

$$u_t + vu_x = 0, \qquad v = \text{const},$$

computing $u_5^{n+1/2}$ by extrapolation (Eqs. (3.7) and (3.4)) is equivalent to following the characteristics, along which u is constant. If a rarefaction wave starts in a front cell, then the point of discontinuity in the cell is ignored. There has been no indication of instability in the front values in the numerical test runs.

Let $\Omega_0(t) \in \Omega$ be the set such that if $(x, y) \in \Omega_0(t)$, then $u(t, x, y) = 0$. Moreover, let $\Omega_1(t)$ be the set $\Omega_1 = \Omega \backslash \Omega_0$. The initial data for Eqs. (1.2–1.4) are such that $\Omega_0(0) \neq \emptyset$. The hyperbolic PDEs have the following property: if $u_{ij}^n = 0$ and the point (i, j) is not reached by the front at $t_{n+1/2}$, then $u_{ij}^{n+1/2} = 0$. Hence, no computation is required to update many values of $u_{ij}^{n+1/2}$. The set $\Omega_1(t)$ grows with the expansion of the shock front. There is no wave interaction between the advancing front and the material ahead of the front.

4. NUMERICAL RESULTS

The numerical solutions to Eqs. (1.2)–(1.4) with discontinuous initial data obtained by the method described in the previous two sections are presented and discussed here. All the examples are solved on a square grid with $h = 1/40$ and k fulfills Eq. (2.5) and Eq. (3.5) or (3.6). Glimm's method was used for Eqs. (1.2), and the upstream difference method for Eqs. (1.3) and (1.4). In the figures, the development of $\partial\Omega_1^n$, the approximation of the contour of $\Omega_1(t_n)$, is drawn for every

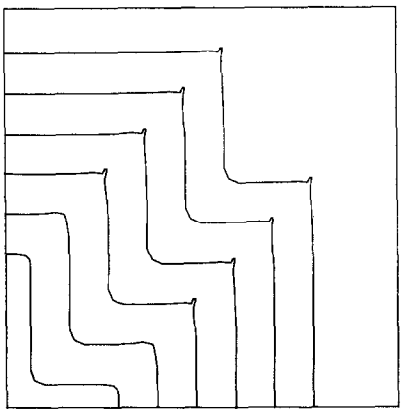

FIG. 5. Burgers' equation, $z = 0.0$, $k = 0.01$, $\Delta t = 20k$. The spurious wrinkles at some of the corners are caused in part by the routine for constructing the front line. The best approximation in the corners is achieved with $z = 0.0$.

example. The initial $\Omega_1^0 = \Omega_1(0)$ is in the lower left corner. The difference in time between the contours is Δt.

The exact solution to Burgers' equation (1.2) is also derived easily in two spatial dimensions. After a simple transformation of the independent variables, Eq. (1.2) becomes a one-dimensional problem with known solution. Recently, Eq. (1.2) with discontinous initial data has been solved numerically by Gropp [12] with a method based on mesh refinement.

The initial data for Eq. (1.2) in the numerical experiments are $u(0, x, y) = 1$, $(x, y) \in \Omega_1(0)$. The boundary conditions are of Neumann type $\partial u/\partial n = 0$ except for Fig. 6, where $\partial u/\partial s = 0$, $s = x - y$. Different random numbers in Glimm's method are used in the two fractional steps Eqs. (3.2) and (3.3). In Figs. 5 and 6, the initial front

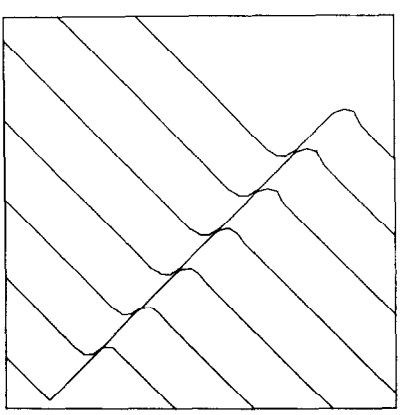

FIG. 6. Burgers' equation, $z = 0.5$, $k = 0.01$, $\Delta t = 20k$.

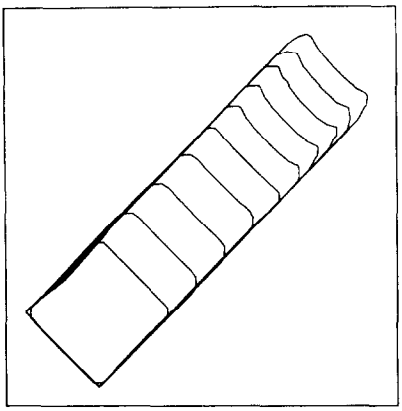

FIG. 7. Burger's equation, rarefaction wave, $z = 0.5$, $k = 0.01$, $\Delta t = 15k$. The initial configuration $\Omega_1(0)$ is the square in the lower left corner.

line retains its shape where the line is straight and the corners are rounded slightly. Part of the rounding effect, at least initially, is caused by the technique for constructing front curves. The error in the speed of the flat parts is on the level of the round-off errors. The analytical solution satisfies $u(t, x, y) = 1$, $(x, y) \in \Omega_1(t)$, $t \geqslant 0$, and the computed solution is $u^n = 1$ in Ω_1^n and $u^n = 0$ in $\Omega_0^n = \Omega \backslash \Omega_1^n$. The solution of the rarefaction wave example in Figs. 7 and 8 on $x = y$ is compared with the analytical solution in Fig. 9. The agreement is very good. Another example with a rarefaction wave is displayed in Figs. 10 and 11. The original method by Noh and Woodward [15] was tested on the initial data of Fig. 5 but was found to be too inaccurate at the corners of the front. In the absence of round-off errors, Chorin's method of determining the local front in [3] yields the exact answer for the problem in Fig. 5.

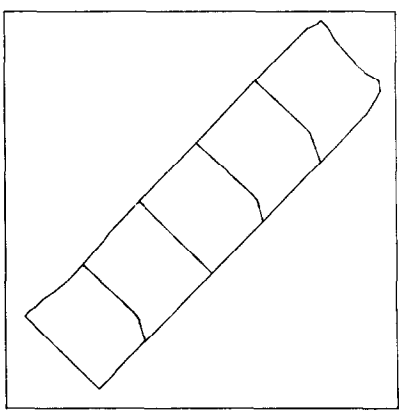

FIG. 8. The contours of $u(1.5, x, y)$ for the problem in Fig. 7 are drawn for $u = 0.1$, 0.2, 0.3, 0.4 going from the left to the right.

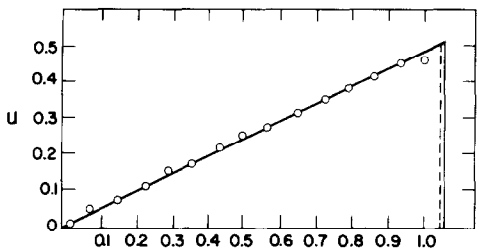

FIG. 9. The exact solution (—) on the diagonal $x - y = 0$ is compared with the computed values (∘) and the computed front (---) in Fig. 8.

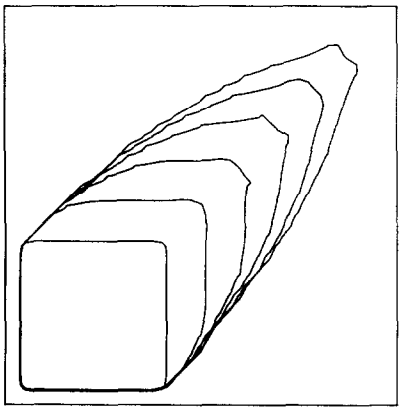

FIG. 10. Burgers' equation, rarefaction wave, $z = 0.5$, $k = 0.01$, $\Delta t = 20k$. The initial configuration $\Omega_1(0)$ is the square in the lower left corner.

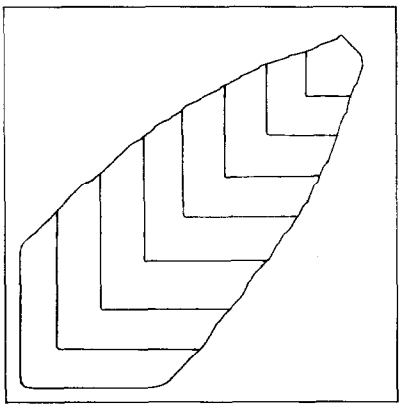

FIG. 11. The contours of $u(1.05, x, y)$ for the problem in Fig. 10 are plotted for $u = 0.1$, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7 going from the left to the right. The top of the front curve is rounded and is about 3% slower than the exact solution. The exact u at the top is approximately 0.83.

The solution to the problem in Fig. 6 obtained with that method had oscillations in the front in the vicinity of the corners.

Numerical solutions to the Buckley–Leverett equation (1.3) in two space dimensions have been obtained by a number of authors [1, 2, 9 – 11, 18]. They use the same $f(u)$ and $\lambda(u)$ as in Eq. (1.3c), although other choices are possible. In the five-spot oil flow test problem (Collins [6]), water is injected into the source at (0, 0) and oil is recovered from the sink at (1, 1). We denote by u the saturation of water. A shock front, where u is discontinuous, propagates from (0, 0) toward (1, 1). For certain values of α in Eq. (1.3c), the front is stable to small perturbations but for other values the front is unstable and *fingering* of the front occurs. Whether the front is stable or not depends on the mobility ratio M [6, 9, 10, 13]. M has the definition

$$M = \lambda(u_b)/\lambda(u_a), \tag{41}$$

where u_b is the value of u immediately behind the front and u_a is the value ahead of the front. In the examples $u_a = 0$. The solution of the Riemann problem at the front yields

$$u_b = (\alpha/(1+\alpha))^{1/2}, \tag{4.2}$$

and therefore,

$$M = 2(1 - (\alpha/(1+\alpha))^{1/2}). \tag{4.3}$$

The front is stable if $M < 1$ and unstable if $M > 1$.

The normal component of q at the front q_n is continuous but the tangential component q_t is discontinuous. When the fractions of black fluid at the front are transported in the horizontal direction by v_x, only v_n, the component normal to the front, actually moves the front curve. Even if v_t and q_t advect the fractions of black

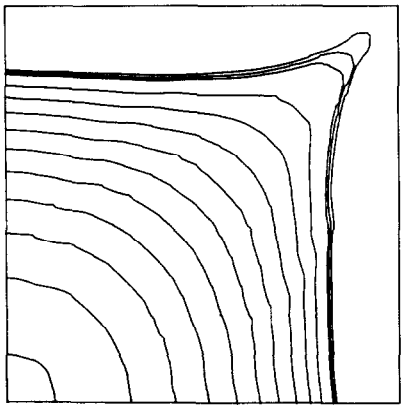

FIG. 12. The Buckley–Leverett equation (Eqs. (1.3a) and (1.3b)), with $\lambda(u) = 1$ and $\alpha = 0.5$, $z = 0.5$, $k = 0.005$. Between the first eleven lines $\Delta t = 40k$, and between the last three lines $\Delta t = 10k$.

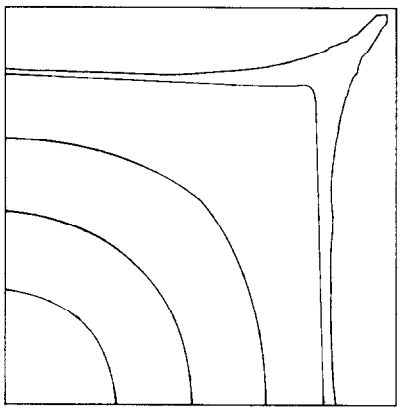

FIG. 13. The contours of $u(2.125, x, y)$ for the problem in Fig. 12 immediately before breakthrough are drawn for $u = 0.9, 0.8, 0.7, 0.6$ going from the left to the right. Breakthrough occurs at $t = 2.13$, about 1–2% later than the analytical solution. When the difference in initial conditions has been taken into account, u is in good agreement with the saturation contours in [2].

fluid, they have little effect on a moderately curved front and no effect on a straight front line.

In Figs. 12 and 13, Eq. (1.3) is solved with $\lambda(u) = \text{const} = 1$ for comparison with the exact solution in [2] for $\alpha = 0.5$, $u_b = 0.577$, $M = 0.845$. In this case, q is independent of t and is computed only initially. $\Omega_1(0)$ is a quarter of a circle and $u(0, r) = 1 + ar + br^2$, $u(0, r_0) = u_b = 0.577$, where $r = (x^2 + y^2)^{1/2}$. The front is stable for $\alpha = 0.5$. The initial data for the example in Fig. 14 are computed starting from the same initial data as in Fig. 12. Then a small disturbance is created on the front. The perturbation disappears in accordance with theory. This is in contrast to the results with $\alpha = 0.1$, $u_b = 0.302$, $M = 1.397$. The initial data are prepared as

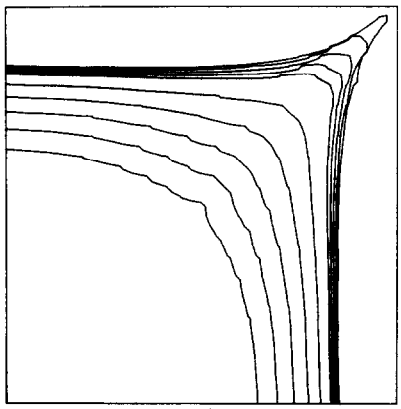

FIG. 14. The Buckley–Leverett equation as in Eq. (1.3) with $\alpha = 0.5$, $z = 1.0$, $k = 0.005$. Between the first six lines $\Delta t = 40k$, and between the last five lines $\Delta t = 10\,k$. The initial perturbation disappears.

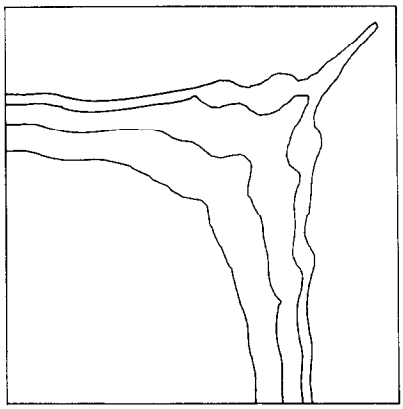

FIG. 15. The Buckley–Leverett equation as in Eq. (1.3) with $\alpha = 0.1$, $z = 0.5$, $k = 0.004$. Between the first three lines $\Delta t = 50\,k$ and between the last two lines $\Delta t = 32\,k$. The initial perturbation grows.

above. A small perturbation develops into a finger in Figs. 15 and 16. The shock profiles are compared in Figs. 15 and 16 for two different values of z in Eqs. (2.1) and (2.3). The most significant difference is the shape of the finger when the front is close to breakthrough at (1,1). From an initially smooth front, however, no spontaneous formation of fingers has been observed. This is probably due to the fact that M is not sufficiently large and to the inaccuracy of the solution to the elliptic equation (1.3b) near the front. Note that M in Eq. (4.3) satisfies $M \in [2 - \sqrt{2}, 2)$. In the last example another model for flow is used allowing a larger M.

The equations of miscible flow (1.4) have been solved by Glimm's method without front tracking in [10] and by using a front tracking technique in [11]. The initial

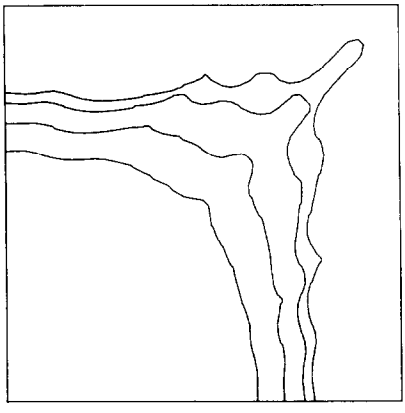

FIG. 16. For comparison, the problem in Fig. 15 is rerun with $z = 1.0$. The difference between the two solutions is not great considering the unstable nature of the problem.

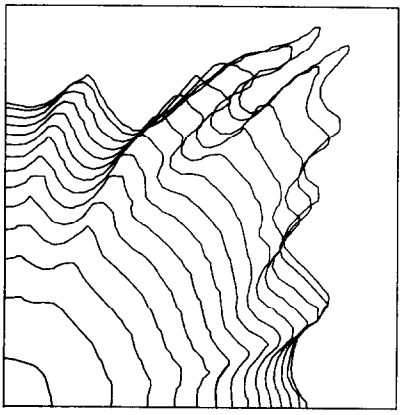

FIG. 17. The equations for two-phase, miscible flow $M = 2.0$, $k = 0.005$, $z = 1.0$. In this case $\Delta t = 40\,k$ except for the last two lines, where $\Delta t = 20\,k$.

data here are such that $u(0, x, y) = 1$, $(x, y) \in \Omega_1(0)$. It follows from Eq. (1.4) that $u(t, x, y) = 1$ when $(x, y) \in \Omega_1(t)$, $t \geqslant 0$. The mobility ratio is $M = 1/\beta^4$. Fig. 17 displays the solution of Eq. (1.4) with $M = 2$. The front is unstable and a *fingering* effect is clearly visible. The computed solution u^n fulfills $u^n = 1$ in Ω_1^n and $u^n = 0$ in Ω_0^n. (See also Fig. 18.) The solution of the same problem in [11] with the front tracking method described there was depicted as having a stable front curve and no fingers.

The typical CPU time required by a FORTRAN code on a VAX-11/780 under VMS to compute the results in Figs. 7, 16, and 17 is presented in Table I.

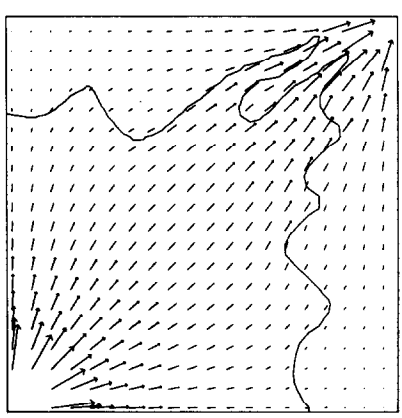

FIG. 18. Here q (2.5, x, y) is displayed for the last profile in Fig. 17.

TABLE I

Example	Total CPU time	CPU time per step	Percentage of total CPU time (a)	(b)	(c)	(d)
Fig. 7	191 s	1.27 s	14.3	58.3	2.5	25.0
Fig. 16	516 s	4.13 s	3.7	16.7	63.6	16.0
Fig. 17	2153 s	4.49 s	6.3	14.9	68.7	10.1

Note. (a) Computation of points of discontinuity, front curve, and rarefaction squares. (b) Computation of F_{ij}^n. (c) Computation of q^n. (d) Computation of u^n.

5. Conclusions

The front tracking method described in Sections 2 and 3 has been tested on three scalar hyperbolic PDEs with discontinuous initial data. The numerical solution of inviscid Burgers' equation agrees very well with the analytical solution for shock waves and rarefaction waves. The same holds true also for the Buckley–Leverett equation with $\alpha = 0.5$ and $\lambda(u) \equiv 1$ in Eq. (1.3b), when compared with the analytically derived solution. When a small perturbation is introduced on the front curve in the general system (Eq. (1.3)), it grows in the unstable case, $\alpha = 0.1$, and disappears in the stable case, $\alpha = 0.5$. Fingers develop spontaneously when the mobility ratio is unfavorable in the two-phase, miscible flow equation. The method is able to resolve and follow unstable irregularities in the interface between shocked and unshocked fluid. The computer time required for the front tracking is comparable to the time used for the solution of the differential equations. The properties of the method when applied to nonscalar conservation laws, such as the equations of gasdynamics, have not been tested or analyzed.

Acknowledgments

The problem was suggested to me by Alexandre J. Chorin and Paul Concus. I have had illuminating discussions with Phillip Colella on the hyperbolic equations in general and with Paul Concus on the Buckley–Leverett equation. This work was supported in part by the Director, Office of Energy Research, Office of Basic Energy Sciences, Engineering, Mathematical, and Geosciences Division of the U. S. Department of Energy under Contract No. W-7405-ENG-48.

References

1. N. Albright and P. Concus, "On Calculating Flows with Sharp Fronts in a Porous Medium," Lawrence Berkeley Laboratory Report LBL–9286, 1979.

2. N. ALBRIGHT, P. CONCUS, AND W. PROSKUROWSKI, *Soc. Pet. Eng. of AIME, Pap.* (1979), 7681.
3. A. J. CHORIN, *J. Comput. Phys.* **35** (1980), 1.
4. P. COLELLA, PH. D. THESIS, Univ. of California, Berkeley, 1978.
5. P. COLELLA, *SIAM J. Sci. Stat. Comput.* **3** (1982), 76.
6. R. E. COLLINS, "Flow of Fluids Through Porous Materials," Petroleum Pub., Tulsa, Okla., 1976.
7. P. CONCUS AND W. PROSKUROWSKI, *J. Comput. Phys.* **30** (1979), 153.
8. C. FENIMORE, PH. D. THESIS, Univ. of California, Berkeley, 1979.
9. J. GLIMM, D. MARCHESIN, AND O. MCBRYAN, *J. Comput. Phys.* **39** (1981), 179.
10. J. GLIMM, D. MARCHESIN, AND O. MCBRYAN, *Comm. Pure Appl. Math.* **34** (1981), 53.
11. J. GLIMM, E. ISAACSON, D. MARCHESIN, AND O. MCBRYAN, *Adv. Appl. Math.* **2** (1981), 91.
12. W. D. GROPP, *SIAM J. Sci. Stat. Comput.* **1** (1980), 191.
13. B. HABERMANN, *Trans. AIME* **219** (1960), 264.
14. C. W. HIRT AND B. D. NICHOLS, *J. Comput. Phys.* **39** (1981), 201.
15. W. F. NOH AND P. WOODWARD, *in* "Proceedings, Fifth International Conf. on Numerical Methods in Fluid Dynamics" (A. I. van de Vooren ad P. J. Zandbergen, Eds.), Springer-Verlag, New York/Berlin, 1976.
16. D. W. PEACEMAN, "Fundamentals of Numerical Reservoir Simulation," Elsevier, Amsterdam/New York, 1977.
17. R. D. RICHTMYER AND K. W. MORTON, "Difference Methods for Initial-Value Problems," 2nd ed., Interscience, New York, 1967.
18. A. SPIVAK, H. S. PRICE AND A. SETTARI, *Soc. Pet. Eng. J.* **17** (1977), 27.