# An Unsplit, Higher Order Godunov Method for Scalar Conservation Laws in Multiple Dimensions

JOHN B. BELL, CLINT N. DAWSON, AND GREGORY R. SHUBIN

*Exxon Production Research Company, P.O. Box 2189,*
*Houston, Texas 77252-2189*

Received April 18, 1986; revised February 20, 1987

In this paper we develop an unsplit, higher order Godunov method for scalar conservation laws in two dimensions. The method represents an extension, for the special case being considered, of methods developed by Colella and by Van Leer. In our method we begin with a piecewise bilinear representation of the solution on each grid cell. A new piecewise bilinear representation at the next time level is then obtained from a two-step procedure. In the first step, a conservative predictor–corrector scheme derived from the integral form of the differential equation and characteristic considerations is used to obtain average values of the solution over grid cells at the new time level. Next, these new average values are then used to construct a limited, piecewise bilinear profile for each cell at the new time level. The resulting method is shown to satisfy a maximum principle for constant coefficient linear advection. Computational results are presented comparing the new method to Colella's method for linear advection. The method is also applied to two model problems from porous media flow, miscible and immiscible displacement. The new scheme provides accurate resolution of sharp fronts without any significant distortion. © 1988 Academic Press, Inc.

## 1. INTRODUCTION

In this paper we develop an unsplit, higher order Godunov method for scalar conservation laws in two dimensions. The intended application of the method is the modeling of unstable, adverse mobility displacements in porous media. Numerical modeling of unstable displacements in porous media places very severe demands on computational methods. These problems typically entail propagating a sharp, unstable front. Standard finite difference methods applied to these problems produce smeared and wildly distorted fronts. The distortion of fronts, referred to as the grid orientation effect, has received considerable attention in the reservoir simulation literature. A variety of methods have been found that dramatically reduce the grid orientation effect; however, these methods typically include (either implicity or explicitly) a first-order dissipation term that stabilizes the front. A review of this literature can be found in Shubin and Bell [1]. They show that, for first-order schemes, grid orientation effects depend on both diffusive and dispersive anisotropies inherent in the discretization. This observation can be used to con-

0021-9991/88 $3.00

struct methods that are essentially free from any grid orientation dependence. Unfortunately, these methods do not reduce the smearing of the front.

A number of methods have also been proposed to reduce the smearing of fronts. Some of these methods are discussed in Bell and Shubin [2]. Unfortunately, these methods typically exacerbate grid orientation errors rather than reducing them. In [2], a method for reducing the smearing of fronts is developed with special attention paid to avoiding anisotropy in the scheme. The method uses directional fluxes computed using a one-dimensional higher order Godunov procedure (see [3, 4]) to reduce front smearing. Although this method does reduce front smearing and suffers minimal grid orientation, it incorporates an inherent dissipation transverse to the flow. This transverse dissipation reduces the formal accuracy of the scheme to first order and limits its utility in more general circumstances.

Our goal then, is to construct an unsplit Godunov scheme that is formally second order when unlimited (and consequently does not contain any first-order transverse dissipation to regularize the front) and that does not suffer grid orientation anomalies. Before discussing additional considerations relevant to development of the scheme, we first address the initial choice of an unsplit Godunov scheme. We use an unsplit scheme for two reasons. Primarily, directional operator splitting simply does not work well for incompressible and nearly incompressible flow models characteristic of porous media flow. For the model problems discussed in Section 5, operator splitting generates large oscillations (over 10%) and predicts nonphysical profiles that exhibit massive grid orientation dependence. Furthermore, an unsplit scheme can be used for more general types of grid systems.

In the present context, we refer to the scheme as a Godunov scheme because "upwind" fluxes are computed by solving Riemann problems. Riemann problems play an important role in porous media flow because, in some regimes, the flux functions are typically not convex; hence, methods that use some type of linearized flux computation can fail to adequately predict nonlinear dynamics.

We will not attempt to develop a method for general systems of conservation laws. (In fact, the method we define cannot be easily generalized.) Instead, we focus on scalar conservation laws of the form

$$s_t + (uf(s))_x + (vg(s))_y = qh(s), \tag{1.1}$$

where $(u, v)$ represents a spatially dependent velocity field. The right-hand side represents sources and sinks of fluid of strength $q(x, y)$ and composition $h$. (The function $h$ may also contain explicit spatial dependence; however, we suppress that dependence to simplify the notation.) We assume incompressibility so that

$$u_x + v_y = q; \tag{1.2}$$

inclusion of compressibility is straightforward. Equally obvious is the extension of the method to diagonalizable systems of conservation laws. The framework of (1.1) is adequate for a wide variety of porous media flow problems. For example, for areal simulation in which gravitational forces are ignored, the two fluxes $f$ and $g$ are always the same so that the system is diagonalizable. Even when gravitational effects are included, several systems of interest are diagonalizable.

Other authors have also pursued the development of unsplit, higher order Godunov methods. Colella [5] derives a scheme using Taylor series expansions to obtain higher order accuracy. Van Leer [6] independently obtains essentially the same scheme from a convection principle. However, both of these authors treat general systems of conservation laws. Thus, their methods focus on more general issues than are required for porous media flow. Colella's scheme can, however, be directly applied to (1.1) and will be used for comparison purposes in the later sections.

In addition to the form of the equation given by (1.1), the intended application to modeling porous media flow imposes certain additional constraints on the design of the method. Foremost, the velocity field in (1.1) is determined by solving an elliptic pressure equation at each step of the calculation. The linear algebra costs associated with this elliptic equation consume the bulk of the computational cost; hence the method used to solve (1.1) can be fairly complicated without noticeably affecting the overall computational time. Also, since the velocity is being determined from a finite difference pressure equation we typically obtain only an average normal velocity over the edges of grid cells. Therefore, we will assume, for the development, we are only given constant normal velocities along each edge. Furthermore, the velocity field need not be continuous. In particular, at sharp fronts and across changes in material properties, the tangential component of velocity is discontinuous. These properties will be explicitly reflected in details of the method to be discussed in Section 2. When the velocity field is known more precisely or is known to be smooth the formalism developed in the next section can easily be modified to account for the additional information.

The new scheme is based on a piecewise approximation to the conserved variable. In a rough outline form, the algorithm begins with a bilinear approximation to the solution on each grid block. The quasilinear form of the differential equation is then used to estimate the average value of $s$ on the edges of the grid cells during a time step. For each edge, average values from the cells on either side provide initial data for an appropriate, one-dimensional Riemann problem. The state which propagates with zero speed in the Riemann problem solution is used to evaluate the numerical flux through the edge. These numerical flux values are then used in a conservative difference scheme to obtain average values over each cell at the new time level. An interpolation procedure with a suitable limiting algorithm is then used to construct a new bilinear profile from the new cell averages.

In the next section of the report we develop the basic predictor–corrector form of the scheme. In Section 3, we discuss the construction of piecewise bilinear profiles for the solution on the grid cells from cell averages. In Section 4, the method is tested on two constant coefficient linear advection problems, propagation of a smooth profile and propagation of a square pulse. The method is also tested on two variable coefficient advection examples. In Section 5, the method is applied to adverse mobility miscible and immiscible displacement problems. The paper concludes with some final remarks about the algorithm.

## 2. Scheme Development

In this section we develop the basic predictor–corrector form of the scheme from the integral form of the differential equation. We will present the corrector first; the requirements of the corrector motivate the predictor step. For simplicity we assume a cell-centered, rectangular grid with constant $\Delta x$ and $\Delta y$. For grid block $B_{ij}$ we denote by $i+\frac{1}{2}, j$ and $i-\frac{1}{2}, j$ the right and left edges and by $i, j+\frac{1}{2}$ and $i, j-\frac{1}{2}$ the top and bottom edges. As noted above we assume that we only have average values of $u$ at the left and right faces and that we only have values of $v$ at the top and bottom faces. We also assume that the source strength, $q$, is given only in terms of cell values $q_{ij}$. The discrete velocities are assumed to satisfy a discrete analog of (1.2); namely,

$$u_{x,ij}+v_{y,ij}\equiv\frac{u_{i+1/2,j}-u_{i-1/2,j}}{\Delta x}+\frac{v_{i,j+1/2}-v_{i,j-1/2}}{\Delta y}=q_{ij}. \tag{2.1}$$

The corrector, which is the same as that used by Colella [5], is obtained by integrating (1.1) over $B_{ij}\times(t^n, t^{n+1})$ and applying the divergence theorem. If we denote by $s_{ij}^n$ the average value of $s$ on block $B_{ij}$ at time $t^n$, we obtain

$$\begin{aligned} s_{ij}^{n+1}&=s_{ij}^n-\frac{\Delta t}{\Delta x}\left(u_{i+1/2,j}F_{i+1/2,j}-u_{i-1/2,j}F_{i-1/2,j}\right)\\ &\quad-\frac{\Delta t}{\Delta y}\left(v_{i,j+1/2}G_{i,j+1/2}-v_{i,j-1/2}G_{i,j-1/2}\right)+\Delta t H_{ij}q_{ij}, \end{aligned} \tag{2.2}$$

where $F$ and $G$ represent the average values of the respective fluxes $f$ and $g$ over the appropriate faces and $H_{ij}q_{ij}$ represents a space-time volume average of the source term. Equation (2.2) does not contain any approximation (except for the aforementioned assumptions on the velocity field); the accuracy of the scheme is determined by the representation of the initial distribution and by the accuracy of the computation of the flux averages $F$ and $G$ and of the source term $H$. For the examples considered in this paper, the $Hq$-terms are localized and quite simple. A straightforward treatment was found to be adequate for the cases considered.

Thus, the essential ingredient of the procedure is the computation of the numerical fluxes $F_{i+1/2,j}$ and $G_{i,j+1/2}$. The remainder of this section is devoted to the predictor step in which we estimate these values. Our approach is based on the geometry of the characteristic surfaces as proposed by Van Leer [6] rather than the more formal approach used by Colella [5]. We assume that we are given a piecewise polynomial profile for $s$ in each grid block. The geometry of the characeristic surfaces is used to estimate the average value of $s$ reaching each edge during a time step. For each edge there are two separate estimates; one from each of the blocks that share that edge.

For concreteness, we consider the treatment of edge-$i+\frac{1}{2}, j$. Other edges are treated analogously. In this case the objective of the predictor step is to estimate averages values on edge-$i+\frac{1}{2}, j$ using the equation linearized about $s_{ij}$ to obtain $s^L_{i+1/2,j}$ and linearized about $s_{i+1,j}$ to obtain $s^R_{i+1/2,j}$. Amibiguities in these states are resolved by solving an appropriate Riemann problem (or, in more general circumstances, some suitable approximate flux computation).

We will only give the computation of $s^L_{i+1/2,j}$; $s^R_{i+1/2,j}$ is analogous. To compute $s^L_{i+1/2,j}$ we linearize Eq. (1.1) in the direction normal to the edge to obtain

$$s_t + u_{i+1/2,j} f' s_x + (vg)_y + u_{x,ij} f - q_{ij} h = 0, \tag{2.3}$$

where $f'$ is evaluated using the average block $s_{ij}$ and $u_{x,ij}$ is computed as in (2.1). For purposes of the derivation we assume that $u_{i+1/2,j} f'(s_{ij})$ is strictly positive. The modifications to the final predictor formulae when this is not the case will be indicated later. (Note that we would like to linearize the system about values at the interface where we have evaluated the velocity; however, technical considerations regarding systems necessitate linearization about cell averages.) We have not linearized the derivative terms transverse to the edge. This allows a more accurate representation of the transverse flux.

The estimate for the predictor is obtained by integrating (2.3) over the space-time region $ABCDEF$ indicated in Fig. 1. This region is determined from the characteristic domain of dependence of (2.3). In particular, all characteristics traced backward in time from the edge $BCEF$ intersect either the base rectangle $ABDE$ or one of the triangular regions $ABC$ or $DEF$. Furthermore, any characteristic traced forward in time from the $ABDE$ to either intersects the edge $BCEF$ of one of the triangles.

Thus, to estimate $s^L_{i+1/2,j}$ we form

$$\iiint_{ABCDEF} s_t + u_{i+1/2,j} f' s_x + (vg)_y + u_{x,ij} f - q_{ij} h \, dx\, dy\, dt = 0. \tag{2.4}$$

We now integrate (2.4) by parts, noting that $u_{i+1/2,j} f'$ is a constant in (2.3) to obtain

$$\begin{aligned} u_{i+1/2,j} f' \, \Delta t\, \Delta y\, s^L_{i+1/2,j} = u_{i+1/2,j} f' \iint_{BCEF} s\, dy\, dt = \iint_{ABDE} s\, dx\, dy + \iint_{ABC} vg\, dx\, dt \\ - \iint_{DEF} vg\, dx\, dt - \iiint_{ABCDEF} q_{ij} h - u_{x,ij} f\, dx\, dy\, dt. \end{aligned} \tag{2.5}$$

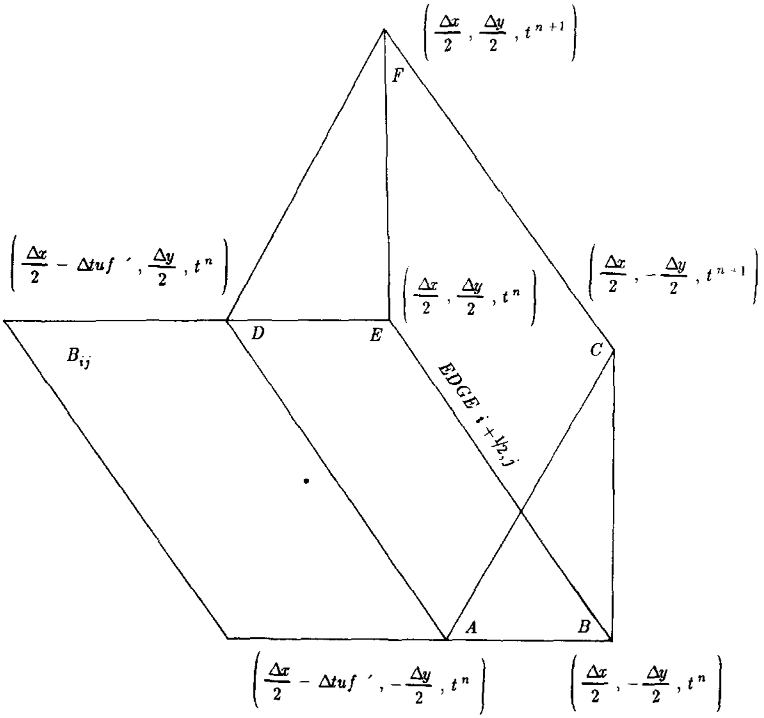

$\left(\frac{\Delta x}{2}, \frac{\Delta y}{2}, t^{n+1}\right)$
F
$\left(\frac{\Delta x}{2} - \Delta t u f', \frac{\Delta y}{2}, t^n\right)$
$\left(\frac{\Delta x}{2}, \frac{\Delta y}{2}, t^n\right)$
$\left(\frac{\Delta x}{2}, -\frac{\Delta y}{2}, t^{n+1}\right)$
$B_{ij}$
D
E
C
EDGE $i+\frac{1}{2},j$
A
B
$\left(\frac{\Delta x}{2} - \Delta t u f', -\frac{\Delta y}{2}, t^n\right)$
$\left(\frac{\Delta x}{2}, -\frac{\Delta y}{2}, t^n\right)$

FIG. 1. Characteristic domain of dependence of edge $i+1/2, j$, $x-y$ coordinates specified relative to block center.

Two of these integral terms are treated in straightforward manner. Since we are assuming that at $t^n$, $s$ is represented by a bilinear function, the integral over $ABDE$ is exactly computed using the midpoint rule. Thus, we have

$$\iint_{ABDE} s\,dx\,dy = u_{i+1/2,j} f' \,\Delta t\, \Delta y\, s_M$$

where

$$s_M = \left( s_{ij} + \frac{(\Delta x - u_{i+1/2,j} f' \,\Delta t)}{2} s_{x,ij} \right).$$

The volume integral in (2.5) is evaluated at the midpoint of $ABDE$. The nonlinearities in $h$ and $f$ preclude the use of more accurate temporal quadrature rules.

If we substitute these formulae into (2.5) we obtain the predictor equation

$$s^L_{i+1/2,j} = s_M + \frac{\Delta t}{2} (q_{ij} h(s_M) - u_{x,ij} f(s_M)) + \frac{\Delta t}{2\Delta y} (\Gamma^- - \Gamma^+), \tag{2.6}$$

where $\Gamma^-$ and $\Gamma^+$ represent the average values of the flux $vg$ over the triangles $ABC$ and $DEF$, respectively. The only modification required when $u_{i+1/2,j}f'$ is negative (in which case $s_M$ lies outside block $B_{ij}$) is to replace $s_M$ by its projection onto block $B_{ij}$, namely, $s_{ij} + (\Delta x/2)\, s_{x,ij}$.

To complete the procedure we must evaluate the integrals of $vg$ over the triangles $ABC$ and $DEF$. We restrict the development to the integral over $DEF$; the other is computed in exactly the same way. Note that the integrand for this term represents the nonlinear flux in the $y$-direction. For this reason we estimate an average value of $s$ using linearization on each of the cells bounding the edge containing the triangle and solve an appropriate Riemann problem to estimate the flux. Thus, the integral over $DEF$ is, itself, computed in a two step process.

The initial step of the process is to estimate the average value of $s$ on the triangle as predicted from block $ij$ and block $i, j+1$. We discuss only the estimate for block-$ij$. To obtain this estimate we use the quasilinear form of the equation

$$s_t + u_{i+1/2,j} f' s_x + v_{i,j+1/2} g' s_y + f u_{x,ij} + g v_{y,ij} - q_{ij} h = 0, \tag{2.7}$$

where the derivative term $f'$ and $g'$ are evaluated at $s_{ij}$. We are assuming for the development that both $uf'$ and $vg'$ are positive; the necessary modifications when this is not the case are discussed later.

We can now proceed as in the previous case and integrate (2.7) over the tetrahedral region $DEFG$, depicted in Fig. 2, that is determined by the domain of dependence of $DEF$. If we now integrate by parts we obtain

$$\begin{aligned} s^L_{DEF} &\equiv \frac{1}{m(DEF)} \iint_{DEF} s\, dx\, dt \\ &= \frac{1}{m(DEG)} \left( \iint_{DEG} s\, dx\, dy + \iiint_{DEFG} f u_{x,ij} + g v_{y,ij} + q_{ij} h\, dx\, dy\, dt \right). \end{aligned} \tag{2.8}$$

As before we evaluate the volume term in (2.8) using the midpoint rule in space and explicit Euler in time to obtain

$$\begin{aligned} &\iiint_{DEFG} f u_x + g v_y - q h\, dx\, dy\, dt \\ &\qquad \approx m(DEG) \frac{\Delta t}{3} \left( f(s_m)\, u_{x,ij} + g(s_m)\, v_{y,ij} - q_{ij} h(s_m) \right), \end{aligned}$$

where $s_m$ is the value of our piecewise bilinear representation of $s$ at time $t^n$ evaluated at the midpoint of $DEG$ which is located at the $(x, y)$ point $(\Delta x/2 - 2\Delta t\, uf'/3,\ \Delta y/2 - \Delta t\, vg'/3)$ (relative to the cell center).

The integral of $s$ over $DEG$ can also be computed to the required accuracy using the midpoint formula and, we have, in fact, used that procedure for the linear part of our solution representation. However, although it is second-order accurate the

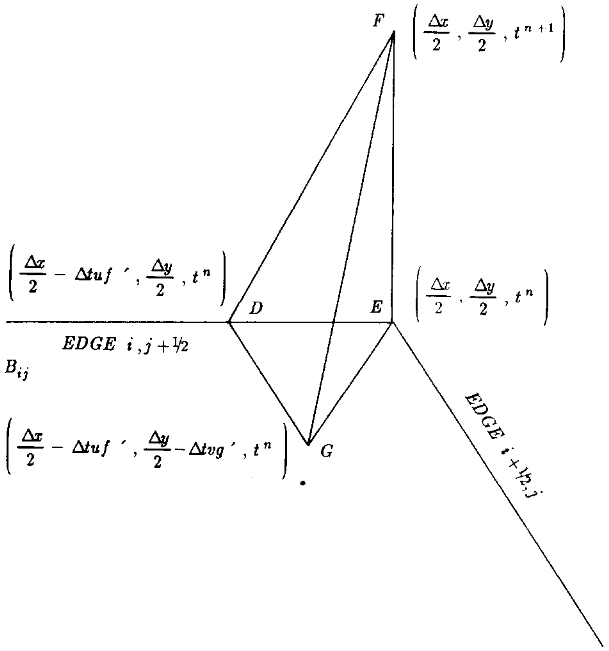

F
$\left(\frac{\Delta x}{2}, \frac{\Delta y}{2}, t^{n+1}\right)$
$\left(\frac{\Delta x}{2} - \Delta t u f', \frac{\Delta y}{2}, t^{n}\right)$
D
E
$\left(\frac{\Delta x}{2}, \frac{\Delta y}{2}, t^{n}\right)$
EDGE $i, j+\frac{1}{2}$
$B_{ij}$
$\left(\frac{\Delta x}{2} - \Delta t u f', \frac{\Delta y}{2} - \Delta t v g', t^{n}\right)$
G
EDGE $i+\frac{1}{2}, j$

FIG. 2. Characteristic domain of dependence of triangle *DEF*, $x-y$ coordinates specified relative to block center.

midpoint rule does not integrate the bilinear term exactly. To exactly integrate the bilinear term we have used the midpoints of each of the edges of *DEG* for integration of the bilinear term. (Of course, all of the integrals over *DEG* could have been evaluated using the edges of the sides. The major reason why this was not done is historical; however, it would also have required more function evaluations.) As before, when one of the quadrature points falls outside the grid block we project it back onto that grid block.

A similar procedure is used to estimate $s^R_{DEF}$. However, it should be noted that in this case the analog of *DEG* will not, in general, be a right triangle. This is because one vertex of the triangle is determined by the wave speed in $B_{ij}$ whereas one of the others is determined by the wave speed in $B_{i,j+1}$. Once $s^L_{DEF}$ and $s^R_{DEF}$ are determined, $\Gamma^+$ is determined by solving an appropriate Riemann problem. More precisely,

$$\Gamma^+ = v_{i,j+1/2}\, g(s_{DEF}),$$

where $s_{DEF}$ is defined by

$$s_{DEF} = \begin{cases} \mathbf{R}_g(s^L_{DEF}, s^R_{DEF}, 0) & \text{if} \quad v_{i,j+1/2} \geqslant 0 \\ \mathbf{R}_g(s^R_{DEF}, s^L_{DEF}, 0) & \text{if} \quad v_{i,j+1/2} < 0, \end{cases}$$

where $\mathbf{R}_g(s^L, s^R, x/t)$ represents the state that propagates with speed $x/t$ in the Riemann problem solution for $s_t + g(s)_x = 0$ with initial left and right states given by $s^L$ and $s^R$.

We can now describe the computation of the numerical flux values appearing in the corrector step (2.2). Given the left and right estimates $s^L_{i+1/2,j}$ and $s^R_{i+1/2,j}$, we define $F_{i+1/2,j}$ as $f(s^{n+1/2}_{i+1/2,j})$, where

$$s^{n+1/2}_{i+1/2,j} = \begin{cases} \mathbf{R}_f(s^L_{i+1/2,j}, s^R_{i+1/2,j}, 0) & \text{if} \quad u_{i+1/2,j} \geqslant 0 \\ \mathbf{R}_f(s^R_{i+1/2,j}, s^L_{i+1/2,j}, 0) & \text{if} \quad u_{i+1/2,j} < 0, \end{cases}$$

where $\mathbf{R}_f$ is defined analogously to $\mathbf{R}_g$.

Before discussing the construction of piecewise bilinear profiles from the cell averages, is will be useful to make a number of remarks about the predictor–corrector formulation. First, we have assumed that the normal velocity was constant across the blocks in determining the characteristic domain of dependence of the edges of the grid cells. (However we have included a finite difference approximation to the normal velocity derivative in the linearization.) For the intended applications the velocity normal to each face is smoothly varying in the direction normal to the edge so that this is a reasonable approximation. If the structure of the velocity is available by some means (for example, from a finite element velocity computation) and the structure of the velocity is considered important for the particular application, this variation can be included in the predictor step; however, this introduces considerable complication into the method.

We have not assumed that the velocity normal to an edge varies smoothly tangent to that edge; in particular, we have allowed for discontinuous jumps in the velocity from block to block. (This can occur in reservoir simulation calculations because of discontinuous variations in material properties.) This gives the scheme an interesting property for linear advection. The triangular region on edge $i, j+\frac{1}{2}$ used in projecting $s^L_{i+1/2,j}$ is not the same as the triangular region on that face used in predicting $s^L_{i+1/2,j+1}$. The material that flows through edge $i+\frac{1}{2}, j+1$ that came from block $ij$ is not equal to the transverse flux correction in the predictor for $i+\frac{1}{2}, j$ for flow through edge $i, j+\frac{1}{2}$.

We have defined the procedure assuming a bilinear representation of the solution on each cell. (We can obtain a linear scheme by setting the bilinear term to zero.) However, since the predictor has been reduced to quadratures, the entire procedure could equally well be applied to any representation that is desired, e.g., higher order polynomials. However, as noted by Colella [7], the predictor–corrector form is inadequate for polynomials of degree higher than two.

The final remark concerns the particular form of the scheme when reduced to constant coefficient linear advection. For this special case, all of the integrals in the predictor step of the method are exact. Thus, for this case the method corresponds to exact advection of the piecewise bilinear profile and its subsequent reaveraging onto the grid (analogous to Van Leer's procedure for a linear profile [6]). This assures that the method satisfies a maximum principle provided the profile reconstruction phase does not introduce any new extrema.

## 3. Profile Reconstruction

In this section we describe the procedure used to construct the piecewise bilinear representation of the solution from the cell averages. This is accomplished in two steps. First, a high order accurate interpolation of the solution is obtained from which derivatives can be estimated. Then a limiting procedure is introduced that forces the fit profile to satisfy a maximum principle.

A variety of approaches can be used to construct the initial bilinear profile. The performance of the scheme is relatively insensitive to the method that is used; the limiting is the crucial step. The particular approach we have taken is a multidimensional analog of the procedure used by Colella and Woodward [8]. The idea here is that for block $B_{ij}$ we interpolate points values at each corner $i \pm \frac{1}{2}$, $j \pm \frac{1}{2}$ from neighboring cell averages. Appropriate difference formulae can then be used to compute the required $x$, $y$, and $xy$ derivatives. The interpolations will be based on fourth-order formulae so that the computed $xy$-derivative is second-order accurate. (This represents overkill to some extent; however, this stage of the computation is very inexpensive so it is not really disadvantageous.)

The interpolation formula used is obtained by finding the bicubic polynomial whose cell averages coincide with cell averages for the sixteen cells surrounding the corner. The value of that polynomial at the corner point gives the required value. For equally spaced grids this formula is given by

$$\begin{aligned} s_{i+1/2,j+1/2} = (&s_{i-1,j-1} - 7(s_{i,j+1} + s_{i+1,j-1}) + s_{i+2,j-1} \\ &- 7s_{i-1,j} + 49(s_{i,j} + s_{i+1,j}) - 7s_{i+2,j} \\ &- 7s_{i-1,j+1} + 49(s_{i,j+1} + s_{i+1,j+1}) \\ &- 7s_{i+2,j+1} + s_{i-1,j+2} - 7(s_{i,j+2} + s_{i+1,j+2}) + s_{i+2,j+2})/144 \end{aligned}$$

which is nothing more than the tensor product of Colella and Woodward's formula. These four corner values then determine a bilinear representation of the solution on the grid. Note that this bilinear function does not, in general, give the correct cell average.

Conceptually, the limiting process corresponds to finding the bilinear function closest to the function determined by interpolation that satisfies some collection of limiting constraints. (The limiting process is performed on each block independently; there is no interblock coupling at this stage of the algorithm.) This process can be viewed from two different perspectives. One may view the process in terms of adjusting the derivatives determined from the initial corner values or one may view the process as adjusting the corner values themselves. We will take the latter approach because it is conceptually simpler and more easily lends itself to approximation.

In terms of the corner values the limiting constraints we want to impose require that the average over the cell be preserved and that the interpolation procedure not introduce any local extrema not present in the neighboring cell averages. For block $B_{ij}$ we let $s^{--}$, $s^{+-}$, $s^{-+}$, and $s^{++}$ denote the bottom-left, bottom-right, upper-left, and upper-right limited corner values, respectively. Then the desired constraints are that

$$s_{ij} = \frac{(s^{++} + s^{+-} + s^{-+} + s^{--})}{4} \tag{3.1}$$

and

$$\begin{aligned} \alpha_{i-1/2,j-1/2} &\leqslant s^{--} \leqslant \beta_{i-1/2,j-1/2} \\ \alpha_{i+1/2,j-1/2} &\leqslant s^{+-} \leqslant \beta_{i+1/2,j-1/2} \\ \alpha_{i-1/2,j+1/2} &\leqslant s^{-+} \leqslant \beta_{i-1/2,j+1/2} \\ \alpha_{i+1/2,j+1/2} &\leqslant s^{++} \leqslant \beta_{i+1/2,j+1/2}, \end{aligned} \tag{3.2}$$

where

$$\alpha_{i+1/2,j+1/2} = \min(s_{ij}, s_{i+1,j}, s_{i,j+1}, s_{i+1,j+1})$$

and

$$\beta_{i+1/2,j+1/2} = \max(s_{ij}, s_{i+1,j}, s_{i,j+1}, s_{i+1,j+1}).$$

Our procedure for adjusting the corner values to satisfy these constraints is based on a fairly simple heuristic idea. First, we subtract an appropriate constant from each of the corner values so that the cell average is correct, i.e., (3.1) is satisfied. We then adjust each of the corner values so that the inequality constraints (3.2) are satisfied. Unfortunately, (3.1) will no longer be satisfied.

The question remains as to how the excess (deficit) should be distributed among the other values. Our approach is to attempt to equidistribute the excess (deficit) among all of the other points subject to not causing any value to violate its inequality constraint. This can be done in a simple vectorized manner by making several passes over the points for equidistributing the values.

The limiting procedure described above is, admittedly, based on a very simple heuristic algorithm. To test its effectiveness, we make precise the notion of "closest" by seeking the bilinear function closest to our originally fit function in $\mathbf{L}^2$. The limiting process then reduces to a quadratic minimization problem subject to constraints (3.1) and (3.2). We can then test the performance of our heuristic procedure by comparing its results to results obtained where the limiting was done by actually solving the minimization problem. For a variety of tests for linear advection, the heuristic procedure obtained results within 10% of the minimization procedure in terms of overall $l^1$ error.

From the description of the limiting procedure, the first step in which the values are adjusted to have the correct cell average appears to be superfluous. However, this step was found to be necessary to obtain good performance from the method. We conjecture that this initial step is, in most cases, sufficient for the limiting. The effect of suboptimal performance of the redistribution algorithm is then reduced in the overall limiting algorithm.

## 4. Linear Advection

In this section we test the method developed in the previous sections on the linear advection equation

$$s_t + (us)_x + (vs)_y = 0.$$

First we perform a convergence study for two constant coefficient examples: advection of a smooth bump and advection of a sharp pulse. The new method is compared to Colella's method [5] for velocity fields at two different angles to the grid.

[redacted]

advection of a smooth bump.

First, let us summarize Colella's scheme. (For a more complete discussion see [5].) Colella fits $x$ and $y$ slopes using a one-dimensional slope limiting procedure; no attempt is made to ensure that the resulting plane does not introduce new extrema. His scheme also uses a different procedure for computing the transverse flux corrections in the predictor ($\Gamma$'s). He first considers simply using Godunov's first-order method using cell averages which we will refer to as the lower order transverse flux (LOTF) approximation. For strongly nonlinear waves he advocates using a one-dimensional higher order Godunov procedure that accounts for transverse gradients in computing the transverse flux (HOTF). As will be seen, this correction improves the performance of the scheme even for constant coefficient linear advection. However, it should be noted that neither version of his scheme satisfies a maximum principle, even if an additional limiting step is introduced to force the planes determined by the $x$ and $y$ slopes to satisfy the inequality constraints (3.2).

Before discussing the examples, some remarks about the relationship of the schemes are merited. When reduced to the case of constant coefficient linear advection, there are basically two differences between our scheme and Colella's schemes. First, our scheme uses the bilinear term whereas his do not. Inclusion of the bilinear term can have two effects on the algorithms. In the actual predictor–corrector scheme, the bilinear term only enters in the estimation of the transverse flux correction term. It is easy to see that its contribution to the scheme will be maximized for flow along the diagonal of the grid; as the angle the velocity field makes with the grid lines goes to zero the effect of the bilinear term goes to zero.

The other effect of the bilinear scheme is to reduce the amount of limiting of the $x$ and $y$ slopes required to satisfy (3.2). However, Colella's scheme makes no attempt to satisfy corner conditions such as (3.2). Thus, for accuracy considerations on linear problems, the effect of the bilinear term is to help recover some of the loss incurred from limiting to satisfy the corner contraints.

Our first examples are for constant coefficients. We consider two different sets of initial data: a discontinuous diamond profile given by

$$s(x, y, 0) = \begin{cases} 1.0 & \text{if} \quad |x| + |y| \leqslant 0.25 \\ 0.0 & \text{if} \quad |x| + |y| > 0.25 \end{cases}$$

and a smooth bump given by

$$s(x, y, 0) = e^{-60(x^2+y^2)}.$$

For each initial profile we take $u \equiv 1$ and consider two different values of $v$, namely, $v \equiv 1$ and $v \equiv 0.2$. We solve each problem with $\Delta x = \Delta y$ to $t = 1$ at $CFL = \frac{1}{2}$ and measure errors in $l^1$; i.e.,

$$E = \Delta x \, \Delta y \sum_{ij} |s_{ij} - s(x_i, y_j, 1)|.$$

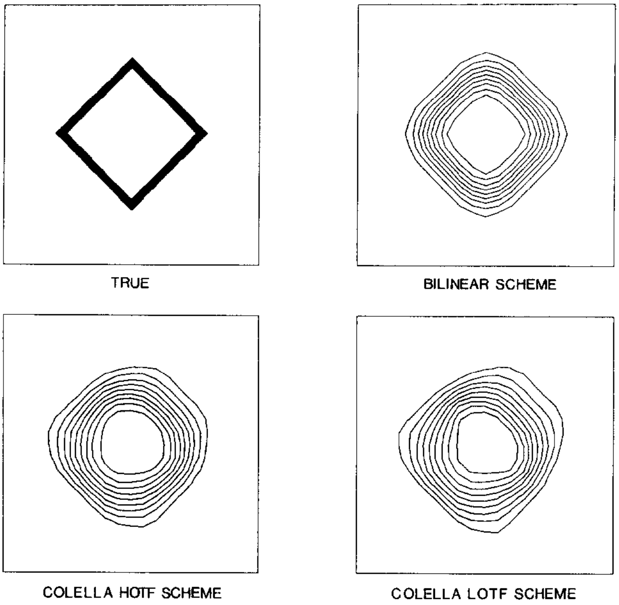
The computational results for the discontinuous data are summarized in Table I.

The results obtained with Colella's scheme are not monotone. For $v = 1.0$ the scheme exhibits roughly 5% overshoot and undershoot. For $v = 0.2$ the undershoot and overshoot are reduced to roughly 1.7%. For discontinuous initial data the errors are dominated by the effects of limiting; thus, we see ony a modest reduction in error (10–20%) with the bilinear scheme. To provide a more detailed examination of the approximate solutions, in Fig. 3 we plot contours from 0.1 to 0.9 for the exact solution and for each scheme for $v = 1.0$ and $\Delta x = 0.5$. For completeness we also show results obtained using Colella's scheme with lower order transverse fluxes. For all of the methods the fronts are considerably smeared; however, the bilinear scheme does a better job of preserving the shape of the initial data than do the other methods. Note that the use of the higher order transverse flux term greatly improves the performance of Colella's method.

TABLE I

$l^1$ Errors for the Diamond Example

| | $\Delta x$ | Bilinear Scheme | Colella Scheme—HOTF |
|---|---|---|---|
| $v = 1.0$ | 0.05 | $6.24E-2$ | $7.31E-2$ |
| | 0.025 | $3.47E-2$ | $4.20E-2$ |
| | 0.0125 | $1.97E-2$ | $2.52E-2$ |
| $v = 0.2$ | 0.05 | $5.81E-2$ | $6.26E-2$ |
| | 0.025 | $3.24E-2$ | $3.50E-2$ |
| | 0.0125 | $1.85E-2$ | $2.02E-2$ |



TRUE

BILINEAR SCHEME

COLELLA HOTF SCHEME

COLELLA LOTF SCHEME

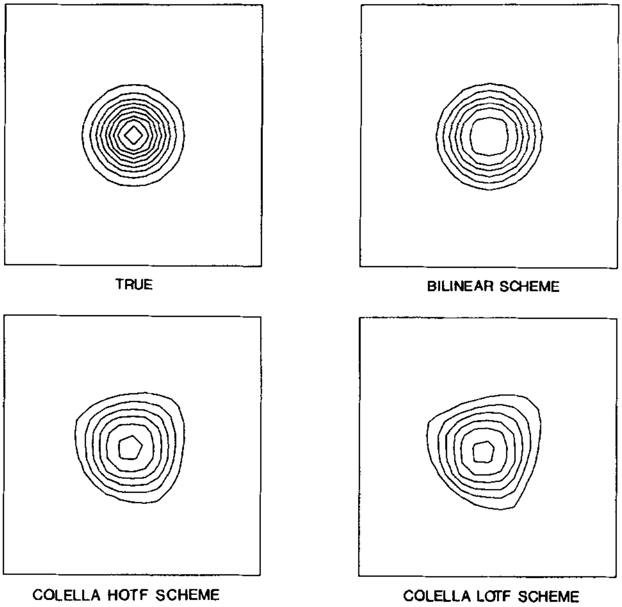
FIG. 3. Coarse grid numerical results fo advection of a diamond.

We now repeat the same computational experiments using the smooth initial data. Errors obtained for this case are summarized in Table II. For smooth initial data the effects of limiting are substantially reduced and we see a dramatic improvement in resolution for the bilinear scheme. Although Colella's scheme still exhibits minor undershoot for this example on coarse grids, the effect disappears as the grid is refined. We again show, in Fig. 4, the solutions obtained with the two methods and with Colella's scheme using lower order transverse fluxes on the coarsest grid. In all cases, the approximate solution has been dissipated considerably. However, the bilinear scheme is doing a good job of maintaining the shape of the profile compared to the other schemes.

TABLE II

$l^1$ Errors for the Smooth Example

| | $\Delta x$ | Bilinear Scheme | Colella Scheme—HOTF |
|---|---|---|---|
| $v = 1.0$ | 0.05 | $9.22E-3$ | $1.39E-2$ |
| | 0.025 | $1.37E-3$ | $4.44E-3$ |
| | 0.0125 | $1.77E-4$ | $1.08E-3$ |
| $v = 0.2$ | 0.05 | $7.55E-3$ | $1.00E-2$ |
| | 0.025 | $1.29E-3$ | $2.86E-3$ |
| | 0.0125 | $2.61E-4$ | $6.32E-4$ |



TRUE

BILINEAR SCHEME

COLELLA HOTF SCHEME

COLELLA LOTF SCHEME

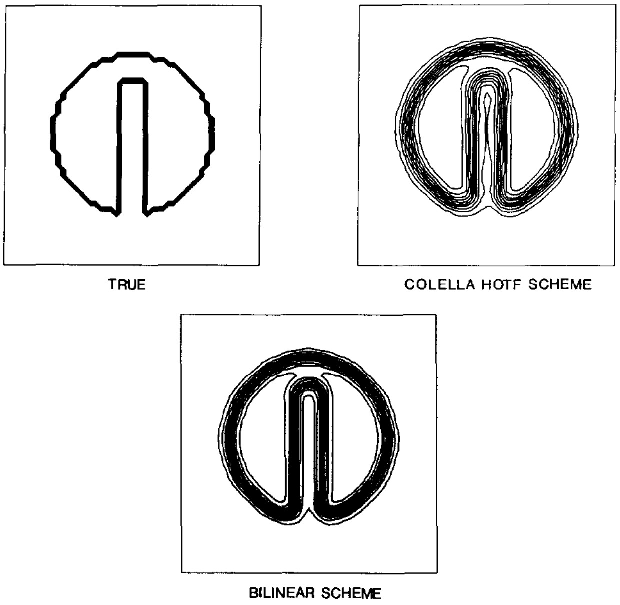
FIG. 4. Coarse grid numerical results for advection of a smooth bump.

In the next two examples we test the method on advection in a divergence free velocity field. The first example, taken from Zalesak [9], is a cylinder that is 15 cells in a radius that is located 25 cells from the origin. In the cylinder, a slot has been cut that is five cells wide extending from near the origin toward the outer edge of the cylinder leaving a five-zone-wide neck at the outer edge of the cylinder. The velocity field is given by

$$u = -y$$

$$v = x.$$

(See Zalesak's paper for more details.) In our implementation we have not smoothed the cylinder by any type of averaging procedure; we simply assign an initial value for each cell based on the location of its center. We have performed the calculation using the bilinear scheme and Colella's HOTF scheme with a global *CFL* slightly less than one. (This makes the *CFL* value approximately one half at the center of the cylinder.) Computational results after one quarter revolution and after one complete revolution are shown in Fig. 5 and Fig. 6. At the end of one complete revolution Colella's scheme exhibits roughly a 2% undershoot. The bilinear scheme offers a number of improvements over Colella's scheme. The resolution of the fronts has been improved. A larger portion of the sides of the cylinder on either side of the slot has been preserved. The slot has filled in

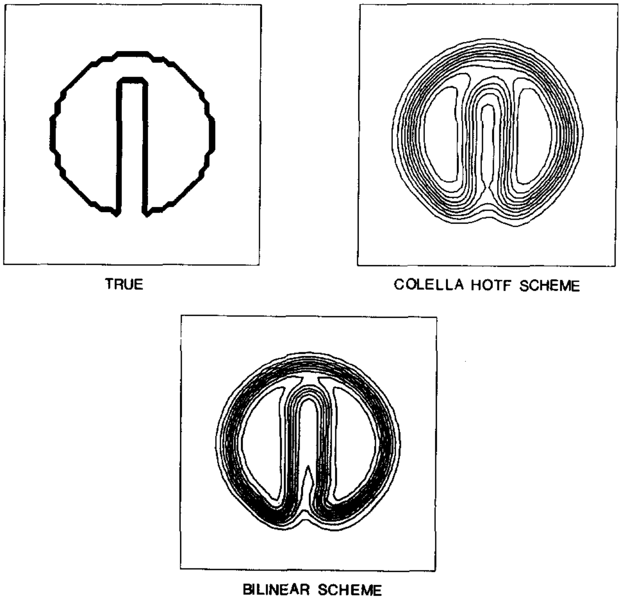


TRUE

COLELLA HOTF SCHEME

BILINEAR SCHEME

FIG. 5. Numerical results for Zalesak's example after one quarter revolution.



TRUE

COLELLA HOTF SCHEME

BILINEAR SCHEME

FIG. 6. Numerical result for Zalesak's example after one complete revolution.

somewhat less and more of the neck is maintained. Furthermore, the bilinear scheme does a better job of preserving the shape of the slot: Colella's scheme causes the slot to distort slightly to the right near the neck. The error in $l^1$ is reduced by about 17% for the bilinear scheme.

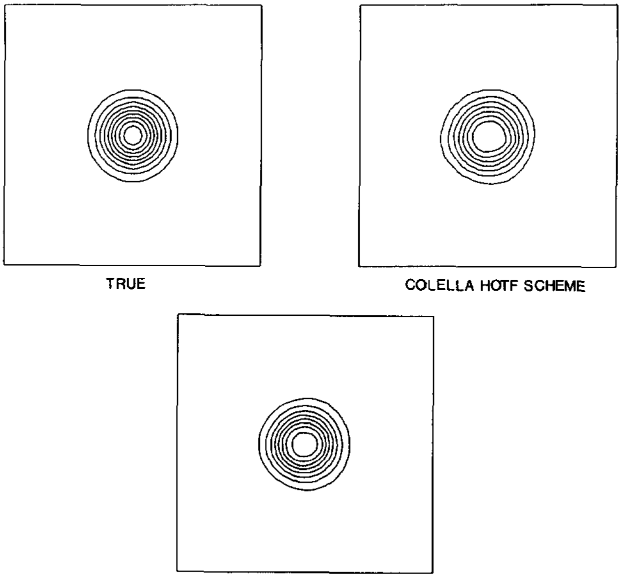
Our final linear advection example uses smooth initial data given by

$$s(x, y, 0) = e^{-300(x^2 + (y - 1/2)^2)}.$$

We use the same velocity field and time step as the Zalesak example with $\Delta x = \Delta y = 0.01$. Contour plots of the computed solutions after one complete revolution are shown in Fig. 7. Again, both the resolution and the shape of the profile are improved with the bilinear scheme. For this example the error is reduced by 50% for the bilinear scheme.

## 5. Reservoir Simulation

In this section we apply the scheme developed in Sections 2 and 3 to model problems arising in petroleum reservoir simulation. Both problems are characterized by unstable fronts that arise by displacing a resident fluid with a less viscous invading fluid. In each case we require the solution of an elliptic pressure equation as well as a hyperbolic conservation law of the form of (1.1). The first problem assumes that the two fluids are completely miscible. For this problem the conser-



TRUE

COLELLA HOTF SCHEME

BILINEAR SCHEME

FIG. 7. Numerical results for rotation of a smooth bump.

vation law is linear. The second problem assumes that the two fluids are completely immiscible. For this problem the instability is much weaker, but the conservation law is nonlinear with a nonconvex flux function. These two simple models are representative of the types of behavior encountered in reservoir modeling and provide a good test of the utility of a method for these types of problems.

*Miscible Displacement*

Our first examples are for incompressible miscible displacement. This system describes the displacement of a resident fluid by a fluid with which is it fully miscible. We have normalized the problem by expressing time in terms of pore volume injected. The flow is then described by the coupled nonlinear system of equations

$$-\nabla\cdot\frac{K}{\mu(c)}\nabla p\equiv\nabla\cdot\mathbf{v}=q \tag{5.1}$$

$$c_t+\nabla\cdot(\mathbf{v}c)-\nabla\cdot D\,\nabla c=qh, \tag{5.2}$$

where $\mathbf{v}=(u, v)$ represents the fluid velocity, $p$ the pressure, and $c$ the concentration of the invading fluid. The dependence of the viscosity $\mu$ on concentration $c$ is given.

computation. For the computations present in this paper we have used a finite difference analog of Galerkin's method. More precisely, the difference equations used for (5.1) are

$$-\left(I+\frac{\delta_y^2}{6}\right)\left(\left(\frac{K}{\mu}\right)_{i+1/2,j}\frac{p_{i+1,j}-p_{ij}}{\Delta x}-\left(\frac{K}{\mu}\right)_{i-1/2,j}\frac{p_{ij}-p_{i-1,j}}{\Delta x}\right)\Big/\Delta x$$
$$-\left(I+\frac{\delta_x^2}{6}\right)\left(\left(\frac{K}{\mu}\right)_{i,j+1/2}\frac{p_{i,j+1}-p_{ij}}{\Delta y}-\left(\frac{K}{\mu}\right)_{i,j-1/2}\frac{p_{ij}-p_{i,j-1}}{\Delta y}\right)\Big/\Delta y=q_{ij}, \quad (5.3)$$

where $\delta_x^2$ and $\delta_y^2$ are the usual undivided second difference operators.

The frontal instabilities associated with this problem for high mobility ratios require that the structure of the solution within grid cells be modeled in the pressure equation. We define $K/\mu$ coefficients in the pressure equation using an integral harmonic average; i.e.,

$$\left(\frac{K}{\mu}\right)_{i+1/2,j}=2\Delta x\,\Delta y\left(\iint_{B_{ij}}\frac{\mu}{K}\,dx\,dy+\iint_{B_{i+1,j}}\frac{\mu}{K}\,dx\,dy\right)^{-1}. \quad (5.4)$$

The use of harmonic averages is suggested by physical principles and by the analogy between block-centered discretizations of elliptic equations and the finite element mixed method. (See Weiser and Wheeler [11] for details.) We perform the integrations in (5.4) using Simpson's rule normal to the edge and the midpoint rule parallel to the edge. This allows us to incorporate the gradient of the approximate solution normal to the edge, as determined by the limited slope calculation, into the pressure coefficients. (We use midpoint integration parallel to the edge for two reasons. First, the normal gradient is physically more important in determining the velocity. Second, by using the midpoint rule we do not introduce quadrature points for Colella's scheme that have nonphysical values of concentration because of the one-dimensional nature of his limiting scheme.)

The pressure discretization implicitly defines a discrete divergence operator. We want to define the velocities used in the computation so that they will be compatible with this discrete divergence operator. This gives

$$u_{i+1/2,j}=-\left(I+\frac{\delta_y^2}{6}\right)\left(\left(\frac{K}{\mu}\right)_{i+1/2,j}\frac{p_{i+1,j}-p_{ij}}{\Delta x}\right)$$

with $v_{i,j+1/2}$ defined analogously.

Exhaustive study of miscible displacement is beyond the scope of this paper. We will confine our computational experiments to assessing the grid orientation dependence of the schemes. Our tests will be made on a repeated five-spot pattern depicted in Fig. 8. We choose the dispersive mixing lengths so that the longitudinal and transverse Peclet numbers are 100 and 1000, respectively. (The length scale used in this computation is the distance from injector to producer divided by $\sqrt{2}$.) These values correspond to the values suggested by Russell and Wheeler [10].

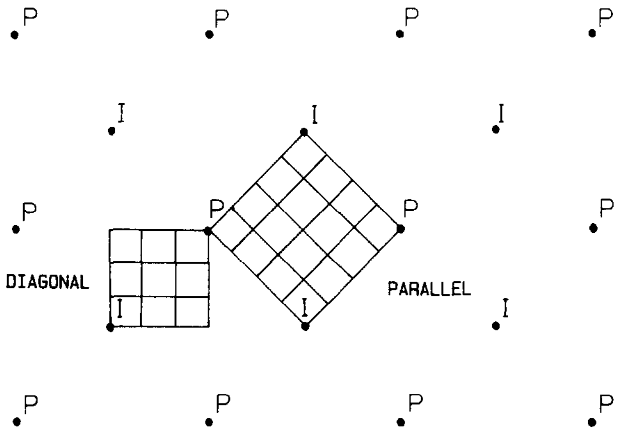


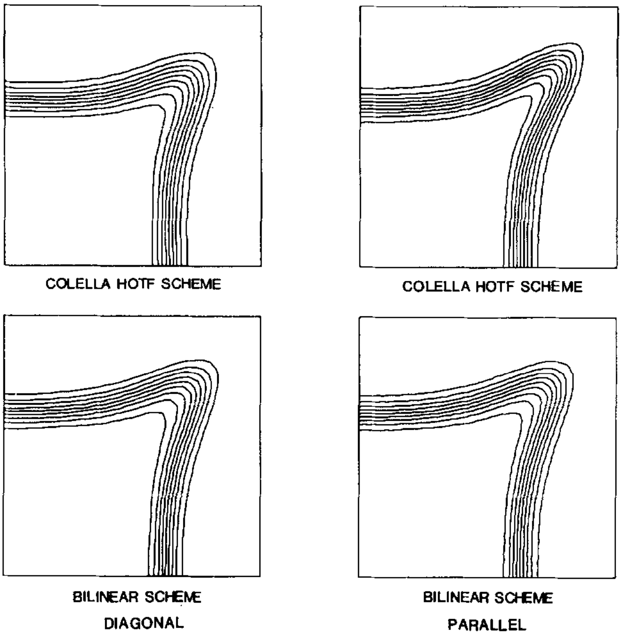
FIG. 8. Repeated fivespot well pattern.

For this particular well pattern there are two different computational grids that can be used, commonly referred to as the diagonal and parallel grids because of their relative orientation to the primary flow path from injector to producer. For our comparisons we will use a 40 × 40 diagonal grid and a 56 × 56 parallel grid (which gives approximately the same size grid cell). At this level of resolution the results of the bilinear scheme appear to compare favorably to fine grid results obtained by Russell *et al.* [12].

In Figs. 9–11, we show computational results for $M = 10,20,40$ for Colella's scheme with higher order transverse fluxes and for the bilinear scheme. The times for each case are chosen to be just before the finger breaks through to the production well. We have found that this is the most sensitive part of the calculation. At $M = 10$, both schemes perform fairly well, however, for Colella's scheme, the finger has penetrated further on the parallel grid. At $M = 20$, the results for the bilinear scheme are still in excellent agreement; results with Colella's scheme are beginning to deteriorate markedly. Finally, at $M = 40$, we begin to see some discrepancy in the results obtained with the bilinear scheme. The finger has penetrated somewhat further on the parallel grid than on the diagonal grid. Results using Colella's scheme have become quite distorted. The diagonal grid results show a nonphysical bowing of the finger along the sides of the grid. On the parallel grid the scheme predicts breakthrough to have already occurred.

### *Immiscible Displacement*

Our final example demonstrates the performance of the scheme on a problem that has nonlinearities. This problem also characterizes a two component, incompressible flow; however, in this case the fluids are assumed not to mix. We let sub-



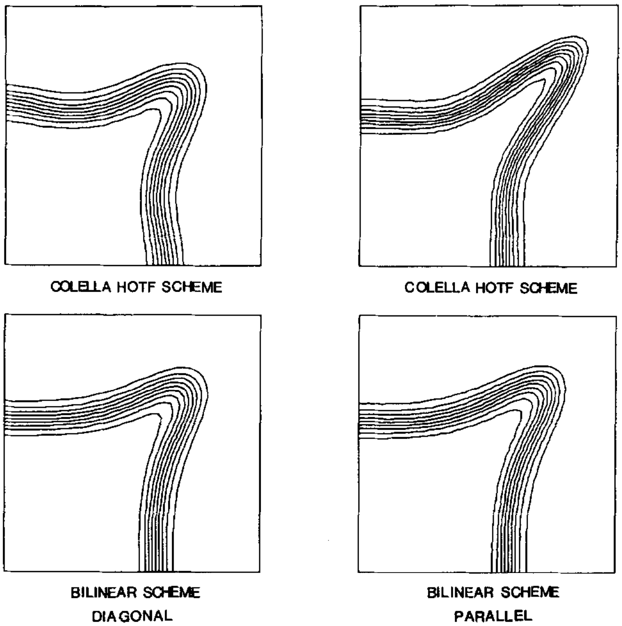
COLELLA HOTF SCHEME

COLELLA HOTF SCHEME

BILINEAR SCHEME
DIAGONAL

BILINEAR SCHEME
PARALLEL

FIG. 9. Numerical results for miscible displacement at 0.45 PVI, $M = 10$.



COLELLA HOTF SCHEME

COLELLA HOTF SCHEME

BILINEAR SCHEME
DIAGONAL

BILINEAR SCHEME
PARALLEL

FIG. 10. Numerical results for miscible displacement at 0.45 PVI, $M = 20$.

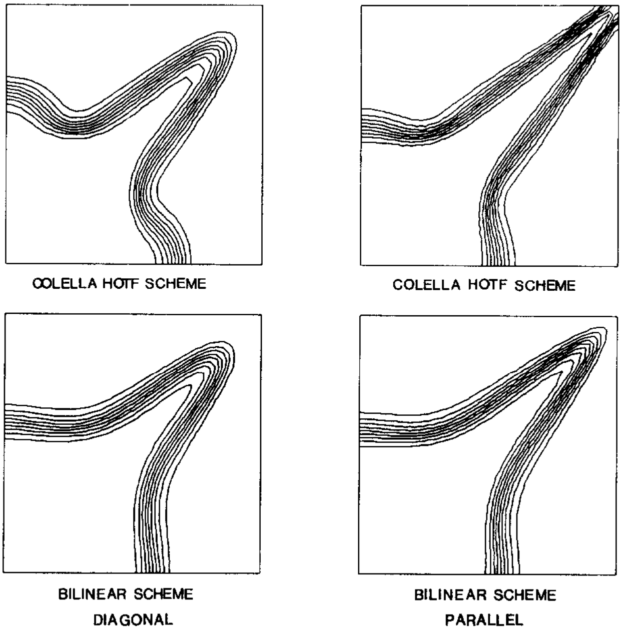


FIG. 11. Numerical results for miscible displacement at 0.45 PVI, $M=40$.

script *o* refer to oil and *w* refer to water. For this problem, the equations describing the flow are

$$-\nabla \cdot (\lambda_o + \lambda_w)\,\nabla p \equiv \nabla \cdot \mathbf{v} = q \tag{5.5}$$

$$s_t + \nabla \cdot \mathbf{v} f(s) = q h_w \tag{5.6}$$

Here, $s$ is the saturation of water, $\mathbf{v}$ is the total velocity, $\lambda_o = K(1-s)^2/\mu_o$, $\lambda_w = Ks^2/\mu_w$, $f = (\lambda_w/(\lambda_w + \lambda_o))$, and $q_w$ represents the wells. We assume that the viscosities $\mu_o$ and $\mu_w$ are constant ($\mu_o/\mu_w = 4$), and we have specialized to quadratic relative permeabilities. For this problem we ignore capillary pressure which introduces a small nonlinear diffusion term. The pressure equation is discretized using the same procedure as was used for the pressure equation in miscible displacement.

For this problem the instability of the front is much less severe than for miscible displacement. We will examine the convergence behavior of the bilinear scheme for two different grid sizes. In Fig. 12, we show computational results for 20 × 20 and 40 × 40 diagonal grids and their corresponding 28 × 28 and 56 × 56 parallel grids. (The jagged appearance of the contours near the shock on the parallel grid results is caused by the interpolation to the diagonal grid.)

The solution in this case involves considerably more structure. A one-dimensional Riemann problem analysis indicates that for the problem considered we

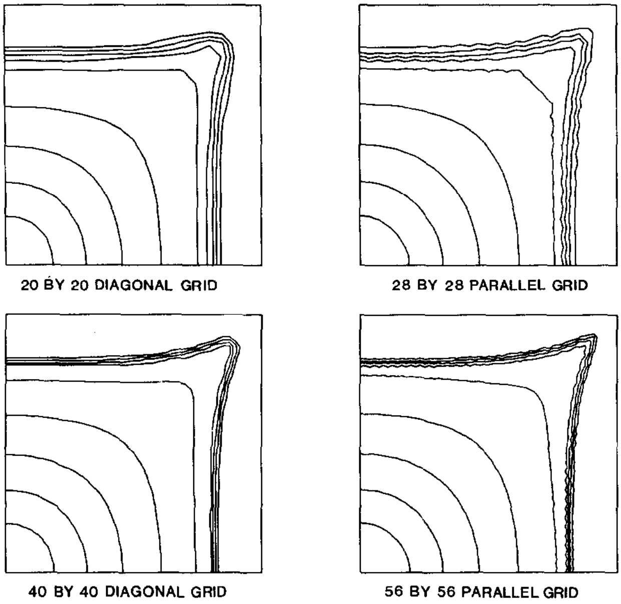


FIG. 12. Numerical results for immiscible displacement at 0.43 PVI, bilinear scheme.

expect a shock with shock height $s = 0.44$ followed by a smooth transition region up to the injected value of $s = 1.0$. All of the computational results appear to predict the correct shock height. The smooth structure of the solution behind the shock is well resolved even on the coarse grids. The two fine grid computations show excellent agreement; there appear to be no grid orientation effects.

## 6. CONCLUSIONS

In this paper we have developed an unsplit, two-dimensional higher order Godunov method for a hyperbolic conservation law. The method is based on discontinuous bilinear approximation of the solution on each grid cell. (We note that although presented only for a single conservation law, the method can be easily extended to diagonalizable systems. An extension to general systems would be considerably more difficult.)

Tests of the method on a variety of linear advection problems indicate that the method is more accurate than existing methods of this type. For propagation of a pure discontinuity, the improvement is rather modest; the improvement for smooth structure is fairly dramatic. In particular, other methods lead to considerable distortion of the shape of the profile as it is propagated; our method does a much better job of preserving its shape.

The method was applied to two model problems for flow in porous media. These two problems display the types of behavior that are typical of reservoir simulation calculations. The problems are characterized by unstable fronts. The bilinear scheme performs well for both problems. It allows us to treat a much wider range of problems than is possible with other methods.

The major difficulty with the scheme is its complexity. This renders the scheme very costly for general application. However, for applications such as porous media flow the computational cost is dominated by the solution of an elliptic pressure equation. For this type of situation where a conservation law is being solved as part of a larger computational task, the complexity of the scheme does not present a problem.

## ACKNOWLEDGMENTS

The authors would like to thank Alan Weiser for providing the linear algebra software which was used to solve the pressure equations for the reservoir simulation examples in Section 5.

## REFERENCES

1. G. R. SHUBIN AND J. B. BELL, *Comput. Methods Appl. Mech. Eng.* **47**, 47 (1984).
2. J. B. BELL AND G. R. SHUBIN, in *Eighth SPE Symposium on Reservoir Simulation, Dallas, Texas, February 10–13, 1985,* p. SPE 13514.
3. P. COLELLA, *SIAM J. Sci. Stat. Comput.* **6** 104 (1985).
4. P. COLELLA, P. CONUS, AND J. SETHIAN, "Some Numerical Methods for Discontinuous Flows in Porous Media," in *The Mathematics of Reservoir Simulation,* edited by R. Ewing (SIAM, Philadelphia, 1983), p. 161.
5. P. COLELLA, A multidimensional second order Godunov scheme for conservation laws, to appear.
6. B. VAN LEER, "Multidimensional Explicit Difference Schemes for Hyperbolic Conservation Laws," *Computing Methods in Applied Sciences and Engineering, VI* (Elsevier Science, New York, 194), p. 493.
7. P. COLELLA, Private communication, 1985.
8. P. COLELLA AND P. R. WOODWARD, *J. Comput. Phys.* **54** 174 (1984).
9. STEVEN T. ZALESAK, *J. Comput. Phys.* **31** 335 (1979).
10. T. F. RUSSELL AND M. F. WHEELER, "Finite Element and Finite Difference Method for Continuous Flow Problems," in *The Mathematics of Reservoir Simulation,* edited by R. E. Ewing (SIAM, Philadelphia, 1983), p. 35.
11. A. WEISER AND M. WHEELER, On convergence of block-centered finite difference methods for elliptic problems, to appear.
12. T. F. RUSSELL, M. F. WHEELER, AND C. CHIANG, in *Proceedings, SEG/SIAM/SPE Conference of Mathematical and Computational Methods in Seismic Explorations and Reservoir Modeling,* (Siam, Philadelphia, 1985).