JOURNAL OF COMPUTATIONAL PHYSICS 52, 569–591 (1983)

An Adaptive Grid Finite Difference Method for Conservation Laws

J. B. BELL AND G. R. SHUBIN

Exxon Production Research Company,
Houston, Texas 77001

Received January 4, 1983

Adaptive grid finite difference methods for computing time-accurate solutions of nonlinear hyperbolic conservation laws in one space dimension are studied. The basic approach is to decouple the determination of the moving grid from the solution of the differential equation on the moving grid. The grid is determined by an elliptic grid generation technique suitably modified for time-accurate computations. Two methods, Godunov's scheme and an artificial viscosity method, are used to discretize the differential equation. The technique is applied to problems involving the Buckley–Leverett equation modelling the flow of two immiscible fluids in a porous medium. Substantial improvement in computational efficiency is obtained by using the adaptive grid technique. Extensions of this approach to several space dimensions and systems of equations are discussed.

1. INTRODUCTION

In this paper we consider adaptive grid finite difference methods for obtaining time-accurate solutions of hyperbolic conservation laws characteristic of the flow of immiscible fluids in porous media. In particular we consider the single equation

$$u_t + f_x = 0, \qquad 0 < x < 1, t > 0 \tag{1.1a}$$

subject to the initial and boundary conditions

$$u(x, 0) = u_0(x) \tag{1.1b}$$

$$u(0, t) = g(t). \tag{1.1c}$$

The flux function $f(u)$ will be assumed to be C^1. For cases of interest in porous media flow $f(u)$ is usually not convex.

Since equations of the form of (1.1a) can develop discontinous solutions from smooth initial data it is necessary to recast (1.1a) into a "weak" or integrated form (cf. [1]). However, the weak solution is not unique, so to obtain the correct physical solution (representing the limit as diffusion vanishes) one must also impose an entropy condition such as the E-condition of Oleinik [2].

0021-9991/83 $3.00

The nonconvexity of the flux functions found in two-phase immiscible flow in porous media raises serious difficulties in obtaining accurate numerical solutions to (1.1) on relatively coarse grids. Harten, Hyman and Lax [3] show that nonmonotone schemes such as Lax–Wendroff can converge to nonentropy satisfying solutions for nonconvex fluxes. Alternatively, monotone schemes are inherently first-order accurate and thus excessively smear out discontinuities. Some theoretical work of LeRoux [4] and Douglas [5] suggests that a first-order numerical diffusion term (of magnitude comparable to that inherent in upwind schemes) must be present in order to guarantee convergence of the numerical solution to the correct physical solution.

Seemingly, to prevent spurious weak solutions from arising one must resort to a first-order finite difference method. Recent work on higher-order methods for gas dynamics [6, 7] suggest that requiring monotone schemes is not necessary and that first-order diffusion is only needed locally near discontinuities. Preliminary work on adapting these methods for nonconvex flux functions appears promising but will be considered elsewhere (see [8]). An alternative to using complicated finite difference schemes to obtain increased accuracy is to allow the finite difference grid to adapt in time, placing more grid points in regions of rapid spatial changes in the solution. The clustering of grid points around sharp changes in the solution should reduce the smearing caused by the inherent numerical diffusion in first-order monotone schemes.

Many investigators have successfully applied adaptive grid methods to the solution of ordinary differential equations. The use of adaptive grid refinement for multidimensional elliptic boundary value problems has received considerable attention in the finite element literature (e.g., [9–12]). Treatment of time-dependent problems has been primarily limited to "time-asymptotic" methods for solving steady problems in which temporal accuracy is unimportant (e.g. [13–15]).

Recently some work has appeared on truly time-dependent adaptive grid methods (e.g. [16–20]). Of particular note is the moving finite element method [21–24], which yields good results in one space dimension. Unfortunately, in this method the grid motion is implicitly coupled to the solution of the equation (except see [25]) so that in several space dimensions an extremely large linear system must be solved at each time step.

For reasons of computational simplicity we consider an adaptive grid finite difference method in which the grid motion is decoupled from the solution of the partial differential equation on the moving grid. Our approach is based on a modification of elliptic grid generation techniques (see Thompson [26, 27] for a survey of this field, and Brackbill and Saltzman [28, 29] for the development of adaptive elliptic grid generation). The idea behind elliptic grid generation is to numerically compute a transformation from a computational rectangle to the physical domain so that the image of a uniform grid on the rectangle produces the nonuniform grid in the physical domain. When applied in multiple dimensions, this method produces grids in which the number of grid points remains fixed for all time, and the grids are always "logically rectangular." These desirable features obviate the need for the complicated data structures and computational logic involved in grids that are not logically rectangular. For the present one-dimensional problem, these features simply

imply that the initial ordering of the grid points remains fixed for all time; we never remove grid points from one location and reinsert them elsewhere.

In the next section we discuss in some detail the use of grid generation for time-dependent problems, in particular the modifications of the elliptic grid generation technique that we have made in order to obtain a smooth time evolution of the grid.

Solving the differential equation on a moving grid introduces some subtle difficulties for nonconvex flux functions. In particular, "upwind" schemes of various sorts become ambiguous in this setting. Section 3 addresses this issue and discusses two schemes that successfully treat these problems.

In Section 4 we present numerical results that illustrate several features of grid generation and corresponding solution techniques.

2. Method for Determining the Moving Grid

In this section we discuss issues concerning the extension of elliptic grid generation techniques to truly time-dependent problems. The goal is to construct a moving grid in the physical space (x, t); see Fig. 1a. To this end we consider a mapping

$$\begin{aligned} \xi &= \xi(x, t) \\ \tau &= t \end{aligned} \tag{2.1}$$

to a computational (ξ, τ) space in which the grid lines are equally spaced and τ-independent; see Fig. 1b. The grid will be generated by numerically determining the transformation at time $t + \Delta t$ from the transformation at time t and the solution at time t.

The usual approach is to construct an equation to determine $\xi(x)$; this results in a nonlinear elliptic equation that determines the transformation [27–29]. We are more inclined to directly determine the inverse mapping $x(\xi)$ as the solution of a linear elliptic system. In two and three dimensions such an approach has been used for

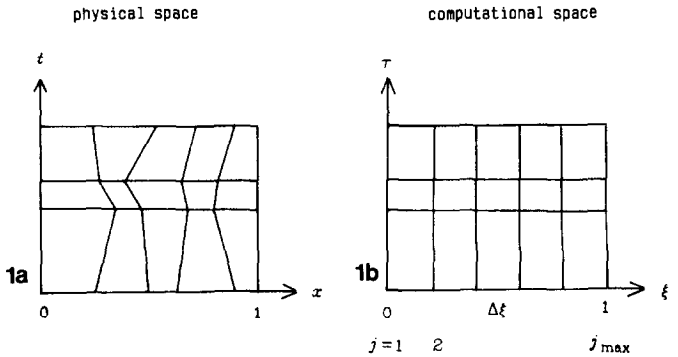


FIGURE 1

nonadaptive grid generation where biharmonic equations determine the transformation [30, 31]. In general, this approach sacrifices a maximum principle (which prevents grid overlapping) in favor of computational simplicity. In the present context the impetus for a fourth-order system is missing; we therefore construct a second-order linear equation to determine the transformation. Here, we are able to obtain a maximum principle. Indeed, for time-dependent problems in several space dimensions the computational savings associated with having linear equations will be reduced and methods satisfying a maximum principle may have more appeal.

Following the variational formulation of elliptic grid generation [29], we seek to minimize

$$\int_0^1 [x_\xi^2 + \lambda_a w(\xi) x_\xi^2 + \lambda_b (x_\xi - x_\xi^*)^2] \, d\xi \tag{2.2}$$

where λ_a and λ_b are parameters, $w(\xi)$ is a clustering function scaled to that $0 \leqslant w(\xi) \leqslant 1$, and x_ξ^* is the value of x_ξ from the previous time step. To understand (2.2) we may think of x_ξ as $\Delta x/\Delta\xi$. The first term of (2.2) attempts to equidistribute the grid size Δx since the minimum value of this term is obained for a uniform grid. The second term attempts to make Δx small where the clustering function w is large. The third term keeps the grid from changing too rapidly in time. This is desirable for computational accuracy and for maximizing the allowable time step, as discussed later. The choice of λ_a and λ_b determines the relative emphases given to these effects. In order to numerically determine $x(\xi)$ we first find the Euler equation for (2.2),

$$(1 + \lambda_a w + \lambda_b) x_{\xi\xi} + \lambda_a w_\xi x_\xi - \lambda_b x_{\xi\xi}^* = 0. \tag{2.3}$$

Replacing (2.3) with a finite difference scheme and using the boundary conditions $x_1 = 0$ and $x_{j\max} = 1$ yields a tridiagonal linear system to determine the new grid point locations. (While we use a simple centered scheme for (2.3), if drastic grid motion is anticipated it would be better to use another scheme which numerically preserves the maximum principle.)

Although we choose to solve (2.3) numerically (and in more general multidimensional problems one must do so), it is possible to directly integrate (2.3) to give

$$x(\xi) = \int_0^\xi \frac{k + \lambda_b x_\xi^*(\xi)}{1 + \lambda_a w(\xi) + \lambda_b} \, d\xi. \tag{2.4}$$

Here k is chosen so that $x(\xi = 1) = 1$. It can easily be seen that $k > 0$; hence $x(\xi)$ is a monotone function and a maximum principle holds. (We again note that the maximum principle only pertains to the one-dimensional case; in higher dimensions the maximum principle is lost.) Since the local grid spacing $\Delta x = x_\xi \Delta\xi$ is $\Delta\xi$ times the integrand of (2.4), the way that parameters λ_a and λ_b affect the grid can be seen by

examining this integrand. Taking the case $\lambda_b = 0$ (no grid smoothing in time), we obtain the local grid spacing

$$\Delta x = \frac{\Delta \xi}{(1 + \lambda_a w)(1/(1 + \lambda_a w))_{average}}$$

so that the grid, as expected, equidistributes $1 + \lambda_a w$. Note that indefinitely increasing λ_a will not result in increasingly finer grid spacing where w is a maximum; instead it is necessary to alter the clustering function w to more sharply discriminate where small Δx's are required. In the case $\lambda_a = 0$ (no clustering in space), we get

$$\Delta x = \frac{\Delta \xi + \lambda_b \Delta x^*}{1 + \lambda_b}$$

corresponding to a combination of smoothing in space and in time (Δx^* is the grid spacing at the previous time step).

We now discuss some aspects of the clustering function $w(\xi)$. We want w to be large where small grid size Δx is desired. A frequently used technique in adaptive grid methods is to try to equidistribute some measure of the error in the computed solution. However, since the choice of an appropriate error measure depends on the solution method, we postpone discussion of the specific choice of w to the fourth section.

Due to our basic approach, we are determining the grid at time t^{n+1} from the solution at time t^n. To allow the grid to "anticipate" shock motion, it is desirable to smear out $w(\xi)$ by making several (we use three) passes of smoothing wherein w_j is replaced by $(w_{j-1} + 2w_j + w_{j+1})/4$.

3. Methods for Solving the Conservation Law on a Moving Grid

Certain subtleties arise in solving the differential equation (1.1) on a moving grid when the flux function is not convex. The problem is caused by the possible ambiguity of the "upwind" direction. For example, the wave speeds corresponding to the solution values on two adjacent intervals may both be positive while the interaction between these two states can generate waves that propagate with negative speeds. In this type of situation it becomes impossible to apply standard upwind switching difference schemes. The most natural way to overcome these problems is to directly apply Godunov's method [32] in the moving physical domain. Since the extension of Godunov's scheme to multiple dimensions is rather uncertain we also consider an artificial diffusion scheme applied to the equation in the transformed computational domain.

We assume that the new grid has been determined at time t^{n+1} and that we now wish to advance the solution of (1.1a) from t^n to t^{n+1}. Consider the region Ω shown

in Fig. 2 whose boundaries are the moving grid lines. The approximation of (1.1a) in Ω is given by

$$0 = \int_\Omega \int \nabla \cdot \begin{pmatrix} u \\ f \end{pmatrix} dt\, dx = \int_{\partial\Omega} (u\mathbf{n}_t + f\mathbf{n}_x)\, dS \tag{3.1}$$

where $\nabla \cdot$ is the $t - x$ divergence and $\mathbf{n}$ is the unit outer normal to Ω. Rearranging we find that

$$\begin{aligned} \int_{x_j^{n+1}}^{x_{j+1}^{n+1}} u\, dx = \int_{x_j^n}^{x_{j+1}^n} u\, dx + \int_{t^n}^{t^{n+1}} \left[f(u_L) - u_L \left[\frac{x_j^{n+1} - x_j^n}{\Delta t} \right] \right] dt \\ - \int_{t^n}^{t^{n+1}} \left[f(u_R) - u_R \left[\frac{x_{j+1}^{n+1} - x_{j+1}^n}{\Delta t} \right] \right] dt \end{aligned} \tag{3.2}$$

where u_L and u_R are the values of u along the left and right boundaries, respectively. At time t^n the approximate solution is represented by piecewise constants, i.e.,

$$u = u_{j+1/2}^n \qquad \text{for} \quad x_j^n \leqslant x \leqslant x_{j+1}^n.$$

Using this approximation, u_R and u_L are constants that can be determined by solving a Riemann problem. In particular, at x_j^n we solve the Riemann problem for a left state of $u_{j-1/2}^n$ and the right state is $u_{j+1/2}^n$ and evaluate the similarity solution at

$$\frac{x - x_j^n}{t - t^n} = \frac{(x_j^{n+1} - x_j^n)}{\Delta t}.$$

Using this form, and making a suitable restriction on time step to ensure that the Riemann solution at one interface does not interact with adjacent boundaries, we see

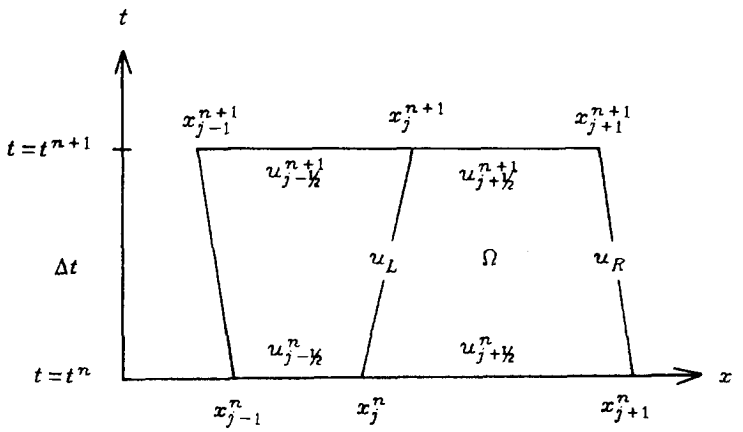


FIGURE 2

that (3.2) corresponds to solving (1.1a) with piecewise constant initial data and replacing the solution at t^{n+1} by the average

$$\int_{x_j^{n+1}}^{x_{j+1}^{n+1}} u\, dx.$$

We note that for the Buckley–Leverett flux function considered in Section 4 the Riemann problem solution can be found in [33]. (Note that the solution there contains an error for the case where the left and right states lie on opposite sides of an inflection point of $f(u)$, but it is easily corrected.)

We next consider a time step selection method in the spirit of the Courant–Friedrichs–Levy (CFL) condition. We want to determine Δt so that no wave arising from the Riemann problem solution centered at x_j^n can reach the grid lines $(x_{j-1}^n, x_{j-1}^{n+1})$ or $(x_{j+1}^n, x_{j+1}^{n+1})$ in the time step Δt. This means that we require

$$x_j^n + \Delta t\, s_{\max} \leqslant x_{j+1}^{n+1}$$

and

$$x_j^n + \Delta t\, s_{\min} \geqslant x_{j-1}^{n+1}$$

where $s_{\max}$ and $s_{\min}$ are the largest and smallest wave speeds in the Riemann solution at x_j^n. We define CFL = 1 to be the largest Δt for which these inequalities hold.

We note that for a single equation this time step selection algorithm has an interesting effect. The adaptive grid method will place the smallest grid spacing near shock waves and larger spacings elsewhere. One would tend to suspect that the presence of these small intervals would substantially restrict the time step. However, near the shock the grid lines tend to essentially follow the shock as do the waves generated by the Riemann interaction. This has the effect of increasing the maximum allowable Δt where Δx is small near a shock wave.

Although very natural in one space dimension, the construction of the Godunov scheme above would be difficult to extend to multiple dimensions. We therefore consider an artificial viscosity based on the results of LeRoux [4] and Douglas [5]. This approach will easily extend to higher dimensions although it gives somewhat poorer results in one dimension than does Godunov's scheme. We explicitly use the mapping (2.1) that defined the grid generation scheme to transform (1.1) into an equation to be solved on the uniform grid in (ξ, τ) space.

First we point out that there is a slight difference between the following two processes: (a) add an artifical diffusion to (1.1a) and then transform using (2.1), and (b) transform (1.1a) using (2.1), and then add an artificial diffusion term. The latter process leads more directly to an indication of what artificial diffusion to add, while the former yields a scheme giving better computational results. We first pursue (a) and then appeal to (b) for the exact diffusion term.

We assume an artificial diffusion of the form

$$u_t + (f(u))_x = (g u_x)_x \tag{3.3}$$

or

$$u_t + (f - gu_x)_x = 0$$

and then transform using (2.1) to obtain (in conservation form [34])

$$U_\tau + [F(U, \xi)]_\xi = 0 \tag{3.4}$$

where

$$U = x_\xi u$$

and

$$F = -\frac{x_t}{x_\xi} U + f\left(\frac{U}{x_\xi}\right) - \frac{g(U/x_\xi)_\xi}{x_\xi}.$$

Equation (3.4) is almost of the same form as (1.1a) except that F depends on ξ in addition to U (we consider x_t to be a function of ξ once the grid is determined). A conservative numerical scheme for Eq. (3.4) (with $g = 0$) possesses a natural interpretation as a "cell balance" where $x_\xi u$ represents the amount of u in a cell of size Δx and the extra flux terms involving x_t are due to the motions of the cell sides. Indeed, the Godunov scheme (3.2) can be readily interpreted as an approximation method for (3.4). As before, let $u^n_{j+1/2}$ represent an approximate value for u in $x^n_j \leqslant x \leqslant x^n_{j+1}$. Let (3.4) be approximated by

$$\frac{U^{n+1}_{j+1/2} - U^n_{j+1/2}}{\Delta\tau} + \frac{F^n_{j+1} - F^n_j}{\Delta\xi} = 0$$

where

$$F_j = \frac{(-x_t)_j}{2}\left[u_{j+1/2} + u_{j-1/2}\right] + \frac{1}{2}\left[f(u_{j+1/2}) + f(u_{j-1/2})\right]$$

$$- g_j\left[\frac{u_{j+1/2} - u_{j-1/2}}{x_{j+1/2} - x_{j-1/2}}\right].$$

Here we have written

$$u_{j+1/2} = \left(\frac{U}{x_\xi}\right)_{j+1/2} \qquad \text{where } (x_\xi)_{j+1/2} = \frac{(x_{j+1} - x_j)}{\Delta\xi}$$

and $x_{j+1/2} = \frac{1}{2}(x_j + x_{j+1})$. We note that it is easy to treat (3.4) in linearized implicit "delta" form [35]. However, implicit schemes are extremely diffusive at large CFL values which would defeat our present purpose. We therefore consider only an explicit scheme.

It remains to choose the artificial diffusion term g. To do so we think of the transformation of (3.3) into (3.4) with $g = 0$ and ignore the dependence of F on ξ. Our previous experience with artificial diffusion suggests that we consider

$$U_\tau + F_\xi = (GU_\xi)_\xi$$

with

$$G \approx \frac{\Delta\xi}{2}|F'(U)| = \frac{\Delta\xi}{2}\frac{|-x_t + f'|}{x_\xi}.$$

Noting the difference between u and U, we therefore choose

$$g_j = \frac{(x_{j+1/2} - x_{j-1/2})}{2} \max_{z \in (u_{j-1/2},\, u_{j+1/2})} |-x_{tj} + f'(z)|$$

by analogy with the results of LeRoux [4].

We observe that a CFL time step criterion for (3.4) would be

$$\frac{|F'|\,\Delta\tau}{\Delta\xi} = \frac{|-x_t + f'|}{x_\xi}\frac{\Delta\tau}{\Delta\xi} = \frac{|-x_t + f'|\,\Delta\tau}{\Delta x} \leqslant 1$$

which is in accord with the time step considerations discussed previously. (In the numerical results of Section 4 we use the previous criterion.)

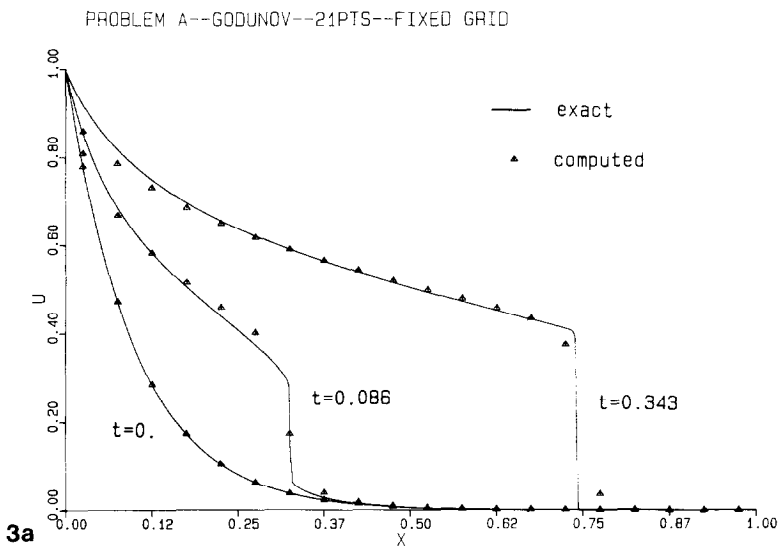
An alternative solution method would be to project the solution at t^n onto the new grid and advance the equation on a nonuniform grid that is fixed in time (for each step); see [19]. This approach has two drawbacks. First the allowable time step becomes very small because of the small grid spacing near the shock. Also the projection introduces additional numerical diffusion into the solution, degrading the results.

4. Computational Results

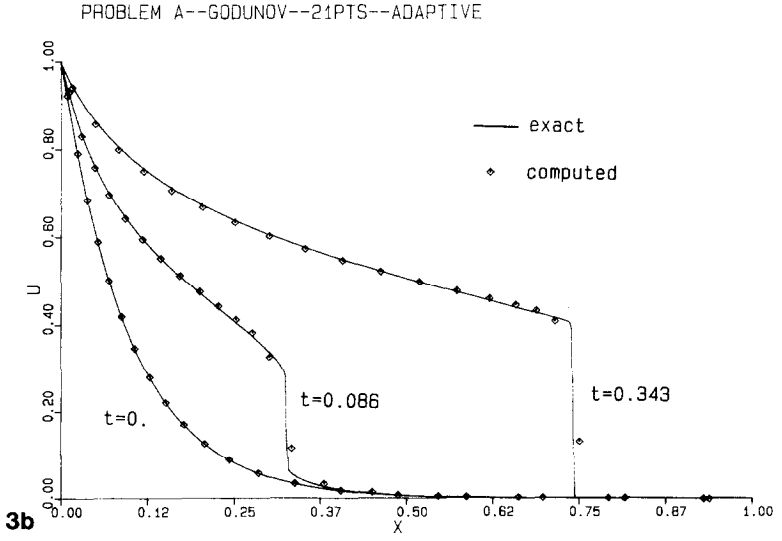
Before discussing the numerical experiments, we must specify the function $w(\xi)$ that determines the clustering of the grid. The specific form of the clustering function depends on both the norm in which a "good" approximate solution is sought and the characteristics of the underlying numerical solution technique. The error measure we wish to use is the L^1 error

$$E = \int_0^1 |u_h(x) - u_e(x)|\,dx,$$

the natural norm for hyperbolic conservation laws. Here, u_h is the piecewise constant approximate solution (consistent with the underlying difference schemes) and u_e is the exact solution. Approximation theory considerations (although not precisely

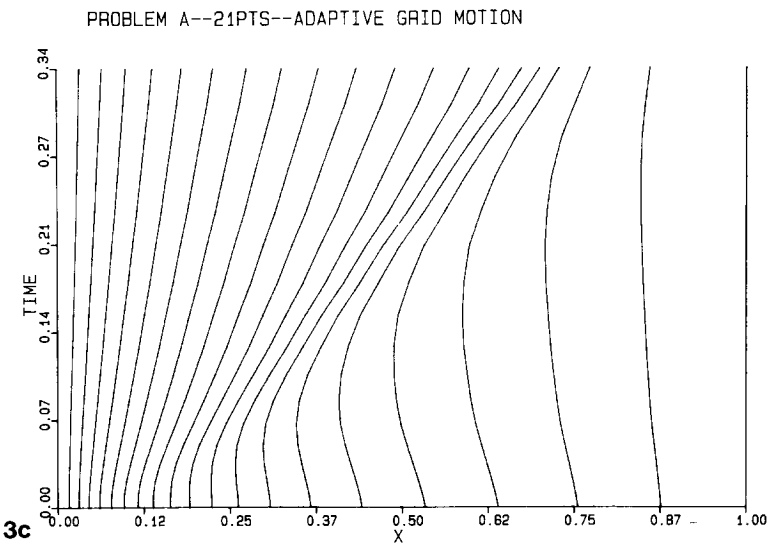


3a



3b

FIGURE 3



3c

FIG. 3—*continued*

applicable here because of the presence of shock waves) indicate that $w(\xi)$ should look like u_ξ^2. Numerical experiments showed that the method performed better when $w(\xi) = |u_\xi|$. This definition of w, using a finite difference of u_h to approximate u_ξ and scaling so that w lies between 0 and 1, was used in all of the numerical experiments.

We apply our adaptive grid methods to solve (1) with

$$f(u) = \frac{u^2}{u^2 + \alpha(1-u)^2}.$$

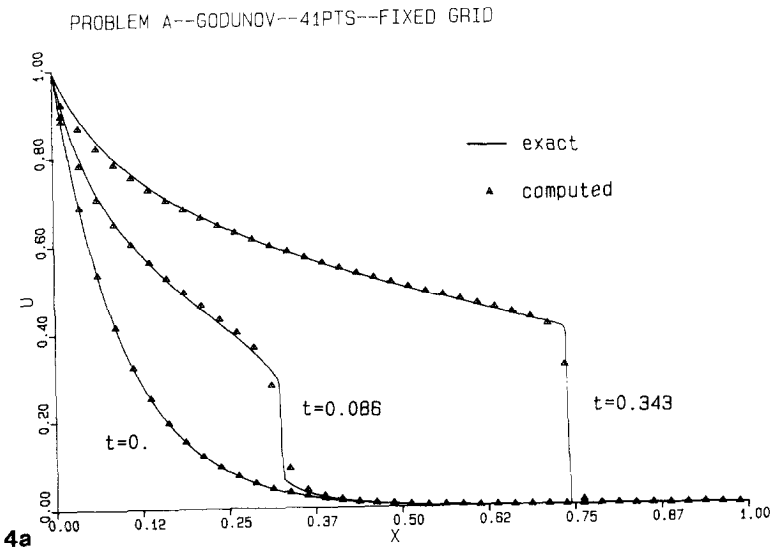
For this choice of f, (1) describes two-phase immiscible flow of oil and water in a linear reservoir in the absence of diffusive (capillary pressure) effects [36]. Here u represents the saturation of water and α is the mobility ratio of oil to water, taken here to be 0.25. We consider three different sets of initial and boundary data:

Problem A

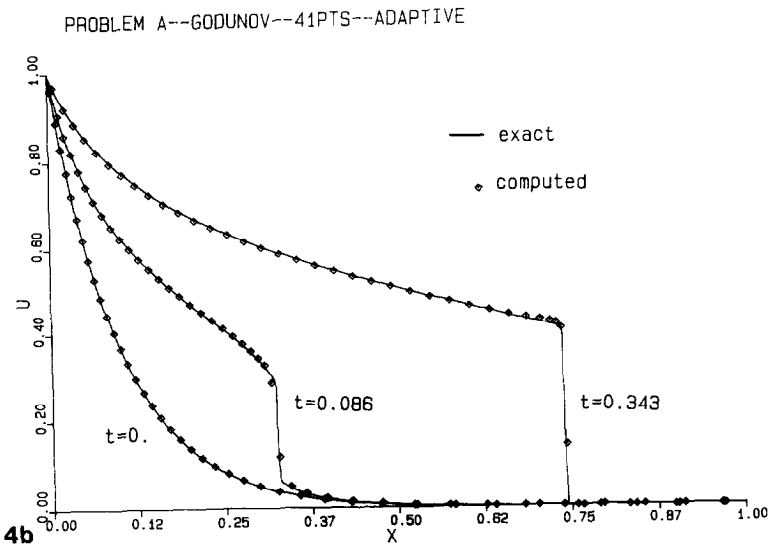
$$u_0^A(x) = e^{-10x}, \qquad 0 \leqslant x \leqslant 1$$

$$u(0, t) = 1$$

$$0 < t \leqslant 0.343$$



4a



4b

FIGURE 4

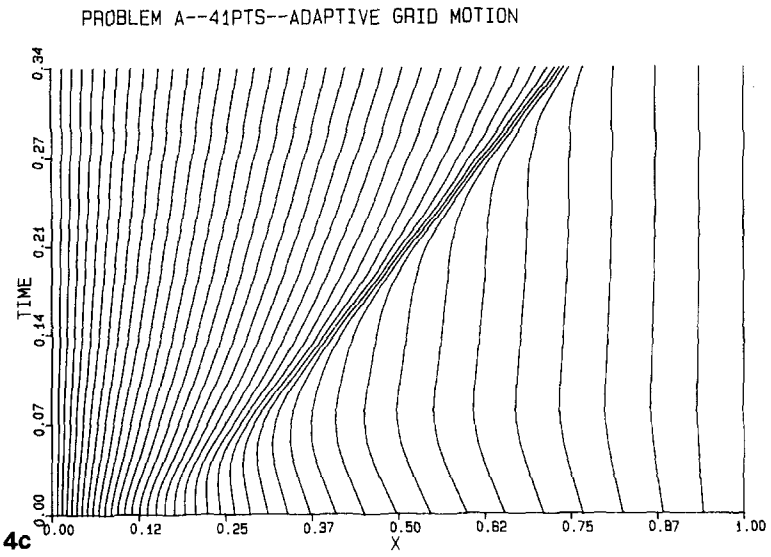


FIG. 4—*continued*

Problem B

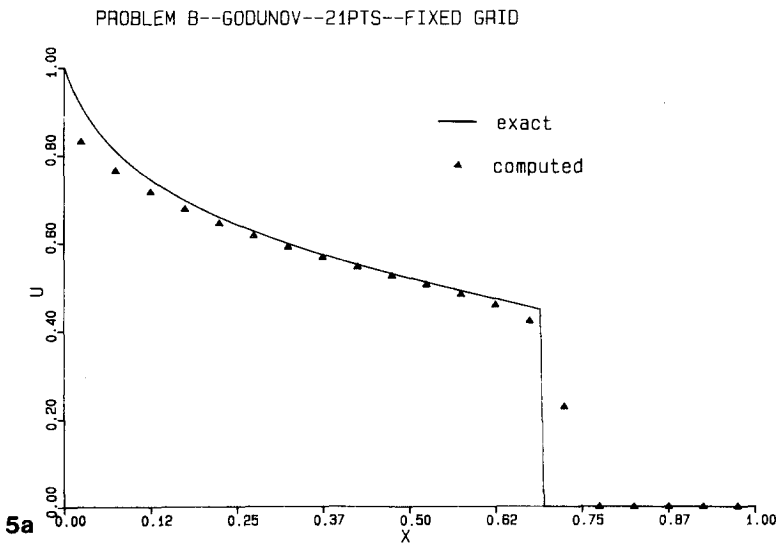
$$u_0^B(x) = 1 \qquad x = 0$$
$$= 0 \qquad 0 < x \leqslant 1$$
$$u(0, t) = 1$$
$$0 < t \leqslant 0.4288$$

Problem C

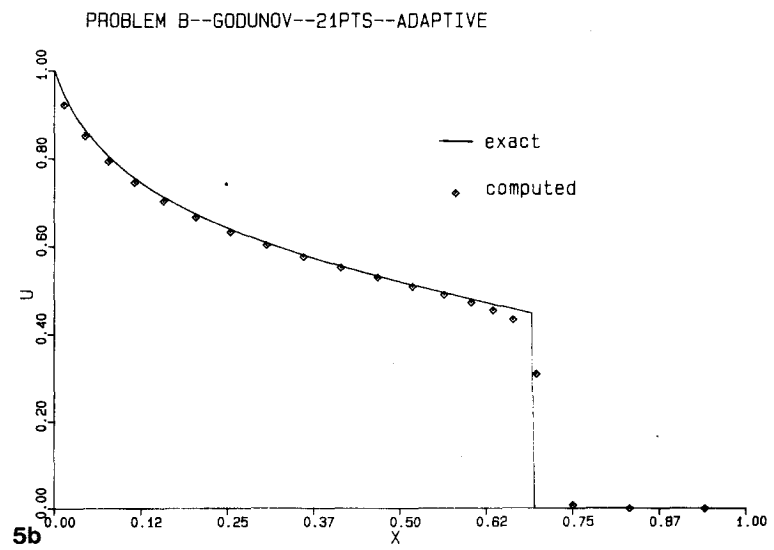
$$u_0^C(x) = \text{exact solution for Problem B at time } 0.1$$
$$u(0, t) = 1$$
$$0 < t \leqslant 0.4288$$

We stop all computation before any significant disturbance reaches the right boundary $x = 1$. To assess the accuracy of our computational results, we obtain the "exact" solution for Problem A by running Godunov's methods on very fine uniform grids until numerical convergence is obtained. The exact solutions for Problems B and C are simply the Riemann problem solutions evaluated at the final time. We will present graphical results for Problems A, B, and C, and numerically compute errors for Problems B and C.

The initial grids were obtained by repeated application of the grid generation algorithm of Section 2. The process was iterated until the change in the grid became small. Unless otherwise noted, the parameters in the adaptivity scheme were set to



5a



5b

FIGURE 5

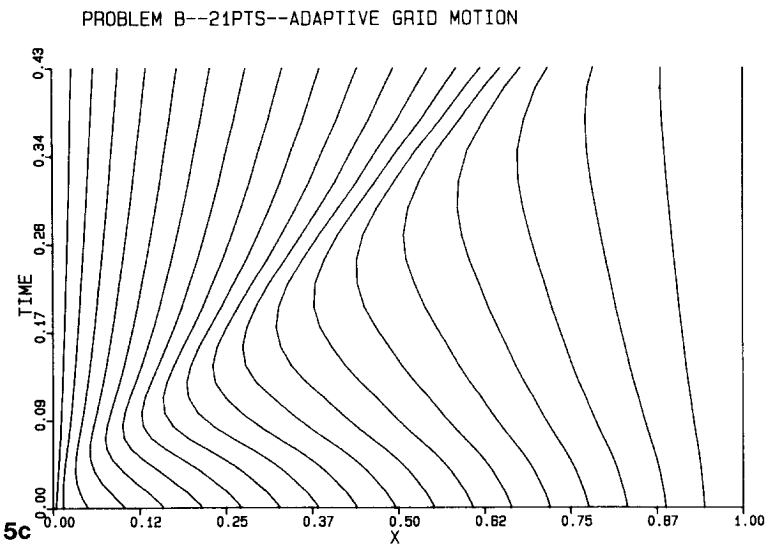
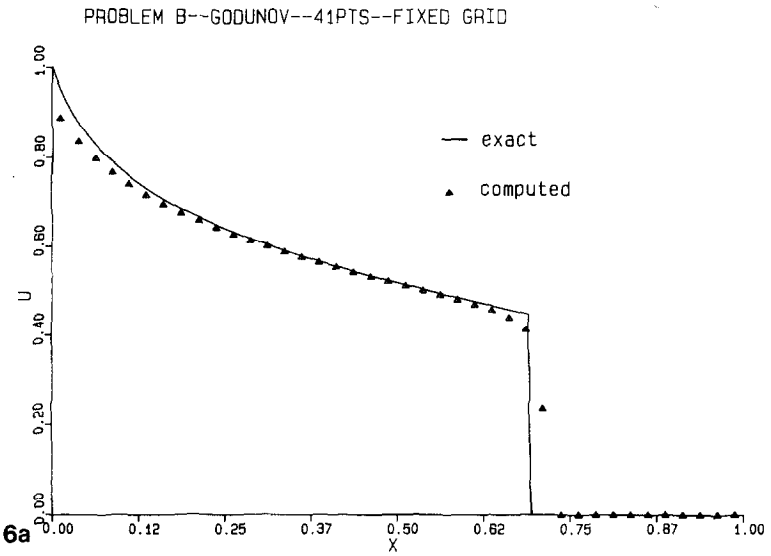


FIG. 5—*continued*

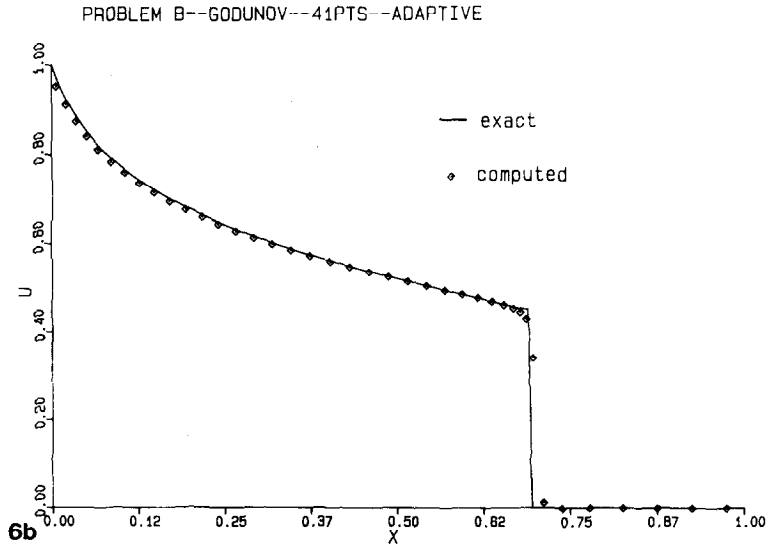
$\lambda_a = 10$ and $\lambda_b = 20$. As discussed later, this choice of the parameters gives typical but nonoptimal results. All runs were made at CFL = 1 and the size of the final time step was adjusted to exactly reach the final time.

For Problem A we show in Figs. 3a and b the exact solution at various times compared with the computed solution using Godunov's scheme with 21 grid points. Results are shown for both fixed and adaptive grids. Each computation took 16 time steps. In all of the figures, the piecewise constant computed solution is represented by a symbol positioned halfway between grid point locations. Figure 3c shows the motion of the adaptive grid where each line represents the trajectory of a grid point. It is clear that the qualitative behavior of the grid is correct and that a reasonable increase in accuracy has been obtained for a modest cost (i.e., the cost of generating the grid). We note that increased accuracy is obtained both at the shock and in the strong rarefaction region near $x = 0$. In Figs. 4a, b, and c we repeat these experiments with 41 grid points. In this case the fixed grid computation took 32 time steps compared to 58 steps for the adaptive grid. The improved accuracy and shock resolution is apparent.

We next consider Problem B. We expect poorer performance than was obtained for Problem A because the adaptive grid cannot resolve the discontinuity in the initial data at the boundary. In Figs. 5a and b we show results at the final time using Godunov's method with 21 points on nonadaptive and adaptive grids. Figure 5c shows the motion of the grid points, and again we note that small grid blocks follow the front. Figures 6a, b, and c show the same experiments with 41 grid points. For both of these cases (and others), the L^1 error is displayed in Tables Ia and b. As

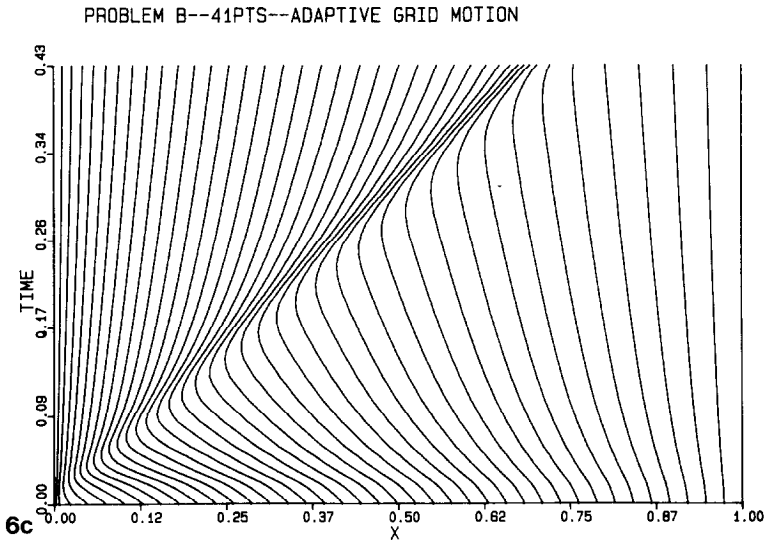


6a



6b

FIGURE 6



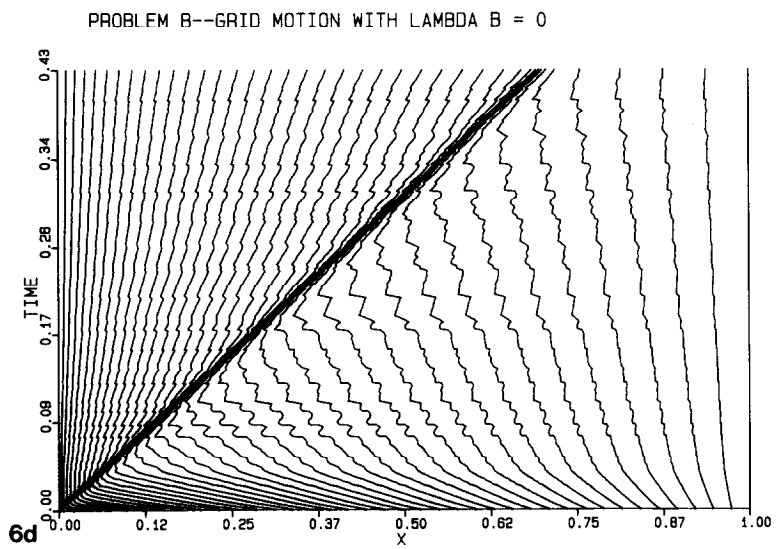


FIG. 6—*continued*

TABLE Ia

Problem B: Fixed Grid Godunov

Grid points	Time steps	Error
11	10	0.0457
21	20	0.0281
41	40	0.0173
81	80	0.0103

expected, we see essentially linear convergence. We note that approximating a function with piecewise constants induces a limitation on how small the error can be made. For this problem the best approximation of the final solution by piecewise constants on an arbitrary grid yields errors of about 0.007 and 0.003 with 21 and 41 points, respectively.

We next consider the effects of varying the grid generation parameters λ_a and λ_b. The L^1 errors obtained with the Godunov scheme for Problem B using various choices of λ_a and λ_b are displayed in Tables II and III for 21 and 41 grid points, respectively. Also shown is the number of time steps necessary for each calculation. We can see that the results presented in Figs. 5a–b and 6a–b using the "default" values of $\lambda_a = 10$, $\lambda_b = 20$ can be improved by about a factor of 2. This is accomplished by choosing larger values of λ_a, resulting in a more accurate solution because grid points are more closely clustered at the front. However, such small grid spacing requires very small time steps unless λ_b is chosen appropriately. If λ_b is too large, the grid cannot adequately follow the front and the error becomes larger. Alternatively, if λ_b is too small, there is a lot of grid jerkiness and the taking of large time steps near the front is inhibited. In Fig. 6d we show the jerky grid motion obtained with $\lambda_a = 10$, $\lambda_b = 0$.

TABLE Ib

Problem B: Adaptive Grid Godunov

Grid points	Time steps	Error
21	26	0.0176
41	57	0.0072
81	150	0.0038

TABLE II

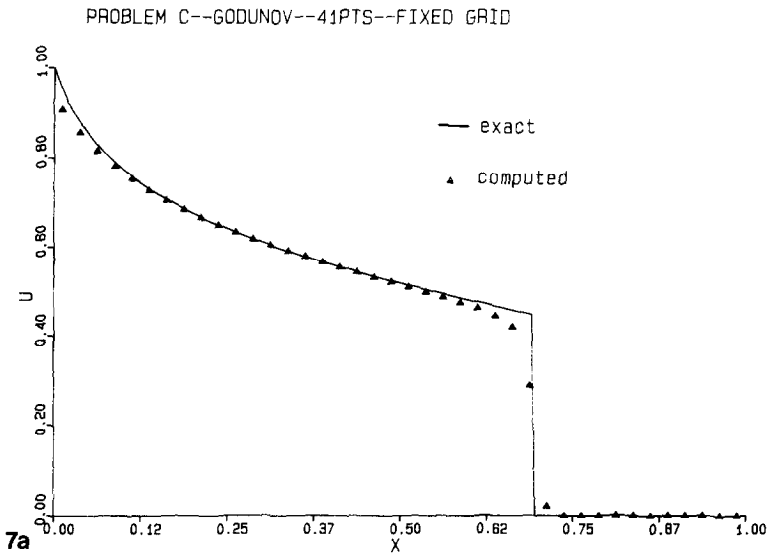
Problem B: Adaptive Grid Godunov—21 Grid Points, Error/Time Steps

λ_a \ λ_b	0	10	20	30	40
0	0.0281 20	0.0281 20	0.0281 20	0.0281 20	0.0281 20
10	0.0104 61	0.0114 35	0.0176 26	0.0224 25	0.0235 24
20	0.0095 124	0.0095 69	0.0104 43	0.0130 30	0.0160 27
30	0.0092 203	0.0087 113	0.0092 69	0.0099 46	0.0118 33
40	0.0095 268	0.0086 184	0.0085 101	0.0088 67	0.0095 49

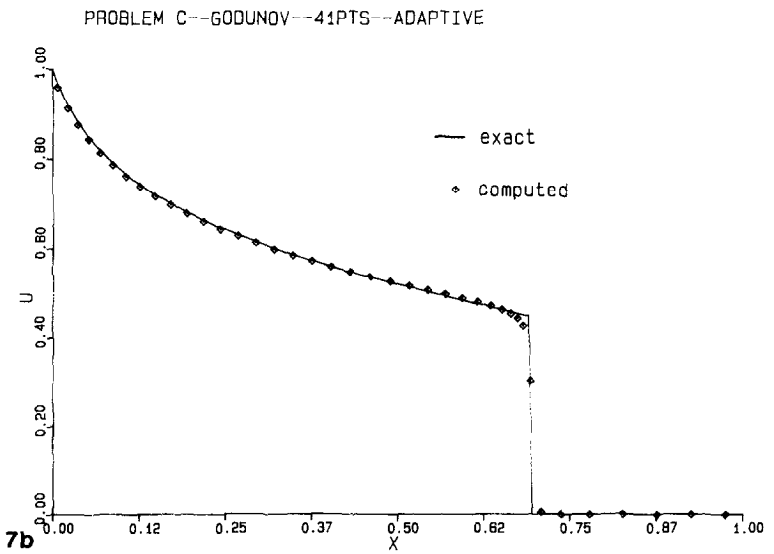
TABLE III

Problem B: Adaptive Grid Godunov—41 Grid Points, Error/Time Steps

λ_a \ λ_b	0	10	20	30	40
0	0.0173 40	0.0173 40	0.0173 40	0.0173 40	0.0173 40
10	0.0057 207	0.0062 122	0.0072 57	0.0096 50	0.0116 48
20	0.0052 552	0.0046 341	0.0049 207	0.0053 102	0.0065 59
30	0.0054 941	0.0043 585	0.0037 476	0.0043 268	0.0049 151
40	0.0059 950	0.0040 804	0.0040 756	0.0039 522	0.0041 311



7a



7b

FIGURE 7

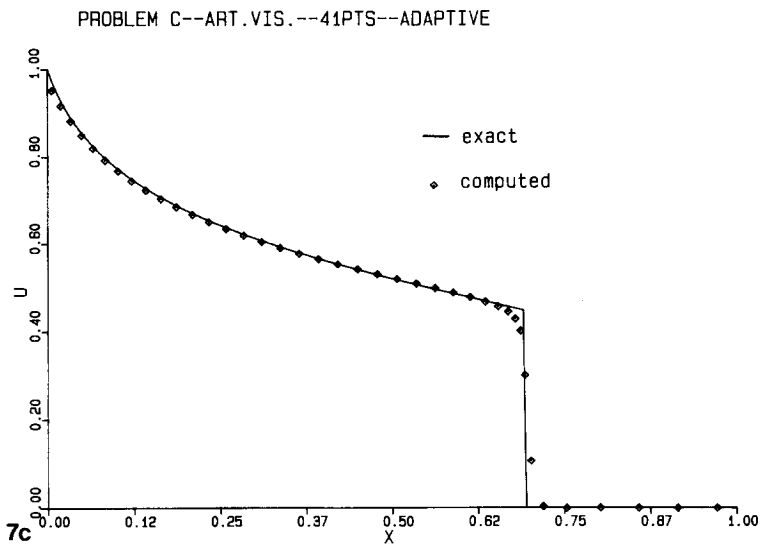


FIG. 7—*continued*

Finally, we consider Problem C. Here, the discontinuity in initial data is interior to the domain and can be resolved by the adaptive grid. Figures 7a, b and c and Table IV present results for fixed grid Godunov, adaptive grid Godunov and adaptive grid-artificial viscosity experiments with 41 points. The modest degradation of results using the artificial viscosity scheme is typical and occurs to varying degrees in the other problems studied.

TABLE IV

Problem C—41 Grid Points

Method	Time steps	Error
Fixed grid Godunov	31	0.0119
Adaptive grid Godunov	38	0.0060
Adaptive grid artificial viscosity	49	0.0066

5. Concluding Remarks

A number of issues regarding this approach need to be addressed. An unfortunate aspect of this and many other adaptive grid techniques is the presence of several parameters that must be chosen. In particular, the choice of λ_b greatly affects the performance of the present method. If it is too large, the grid cannot change rapidly enough to follow the shock; if it is too small, dramatic increases in the number of time steps occur. Once an appropriate value of λ_b is determined, it seemingly works well for a reasonable class of problems; but, in general, it must be set empirically.

Perhaps the most interesting aspect of this method is its ability to take large time steps in the presence of fine spatial grids. To a great degree this phenomenon is an artifact of considering a single conservation law. For systems of equations where different characteristic families have widely varying wave speeds (such as in gas dynamics) the maximum allowable time step will be substantially reduced. The implication of this observation for other types of systems such as those arising in petroleum reservoir simulation remains an open question.

The extension of the adaptive grid generation part of the method to multiple dimensions is relatively straightforward. However, solving hyperbolic conservation laws in several space dimensions on fixed grids is not well understood; extensions to moving grids is by no means easy.

So far we have not addressed the question of computational cost. Compared to a fixed grid method, the major additional computational work comes in determining the grid motion, which essentially involves only the evaluation of an error criterion, some simple smoothing, and a tridiagonal system solution. Also, the difference schemes become slightly more complicated, and it is a little more difficult to determine suitable time steps. However, at least for Godunov schemes, the largest part of the work is devoted to solving Riemann problems—and this is no more difficult (in general) for moving grids than it is for fixed grids.

We also note that, for a comparable amount of work, our adaptive grid computations show only a modest increase in accuracy over the corresponding fixed grid results. Furthermore, the clustering of the grid near the shock in the displayed results is not overly dramatic (though it can be made more so by adjusting parameters). The problem is not the adaptive grid method; rather, it is the underlying nature of the approximating schemes that are based on piecewise constants. Accurately approximating a smoothly varying region of the solution with piecewise constants requires a large number of grid points. This suggests that the adaptive method would be more successful if used with a higher-order numerical scheme such as higher-order Godunov methods based on discontinuous linear approximate solutions. This idea will be pursued more fully in subsequent work.

REFERENCES

1. P. D. LAX, "Hyperbolic Systems of Conservation Laws and the Mathematical Theory of Shock Waves," SIAM Regional Conference Series in Applied Mathematics, 1973.
2. O. A. OLEINIK, *Amer. Math. Soc. Trans.* 2 **26** (1963), 95–172.
3. A. HARTEN, J. M. HYMAN, AND P. D. LAX, *Comm. Pure Appl. Math.* **29** (1976), 297–322.
4. A. Y. LEROUX, *Math. Comp.* **31** (140) (1977), 848–872.
5. J. DOUGLAS, JR., *Simulation of a linear waterflood*, to appear.
6. P. COLELLA, "A Direct Eulerian MUSCL Scheme for Gas Dynamics," LBL-14104, Lawrence Berkley Laboratory, February 1982.
7. A. HARTEN, "High Resolution Schemes for Hyperbolic Conservation Laws," DOE/ER/03077–175, New York University.
8. J. BELL, P. COLELLA, P. CONCUS, AND H. GLAZ, *Application of a higher order Godunov scheme to petroleum resevoir simulation problems*, to appear.
9. GRAHAM F. CAREY, *Comp. Meth. Appl. Mech. Eng.* **17/18** (1979), 541–560.
10. WERNER C. RHEINBOLDT, *Int. J. Numer. Meth. Engrg.* **17** (1981), 649–662.
11. I. BABUSKA AND W. C. RHEINBOLDT, *SIAM J. Numer. Anal.* **15**(4) (1978), 736–754.
12. I. BABUSKA AND W. C. RHEINBOLDT, *Int. J. Numer. Meth. Engrg.* **12** (1978), 1597–1615.
13. R. G. HINDMAN, PAUL KUTLER, AND DALE ANDERSON, "A Two-Dimensional Unsteady Euler-Equation Solver For Flow Regions With Arbitrary Boundaries," AIAA Paper 79–1465, Williamsburg VA., July 1979.
14. MAN MOHAN RAI AND D. A. ANDERSON, *J. Comp. Phys.* **43** (1981), 327–344.
15. B. L. PIERSON AND P. KUTLER, "Optimal Nodal Point Distribution for Improved Accuracy in Computational Fluid Dynamics," AIAA Paper 79–0272, New Orleans LA., Jan. 1979.
16. H. A. DWYER, R. J. KEE, AND B. R. SANDERS, *AIAA J.* **18**(10) (1980), 1205–1212.
17. BABUSKA AND M. LUSKIN, *in* "Advances in Computer Methods For Partial Differential Equations IV" (R. S. Stepleman, Ed.), pp. 5–8, 1981.
18. T. H. CHONG, *SIAM J. Numer. Anal.* **15** (4) (1978), 835–857.
19. J. DOUGLAS, JR., B. L. DARLOW, WHEELER, AND P. KENDALL, *in* "SPE Fifth Symposium on Reservoir Simulation," pp. 65–72, 1979.
20. W. D. GROPP, *SIAM J. Sci. Stat. Comp.* **1**(2) (1980), 191–197.
21. K. MILLER AND R. MILLER, *SIAM J. Numer. Anal.* **18** (6) (Dec. 1981), 1019–1032.
22. K. MILLER, *SIAM J. Numer. Anal.* **18**(6) (1981), 1033–1057.
23. R. J. GELINAS, S. K. DOSS, AND K. MILLER, *J. Comp. Phys.* **40** (1981), 202–249.
24. T. DUPONT, *Math. Comp.* **39** (159) (1982), 85–107.
25. F. DAVIS AND E. FLAHERTY, *SIAM J. Sci. Stat. Comp.* **3** (1) (1982), 6–29.
26. J. THOMPSON, EDITOR, "Numerical Grid Generation," Elsevier/North-Holland, New York, 1982.
27. F. THOMPSON, ZAHIR U. A. WARSI, AND C. WAYNE MASTIN, *J. Comp. Phys.* **47** (1982), 1–108.
28. J. U. BRACKBILL AND J. S. SALTZMAN, *J. Comp. Phys.* **46** (1982), 342–368.
29. J. SALTZMAN AND J. BRACKBILL, *in* "Numerical Grid Generation," Elsevier/North-Holland, New York, 1982.
30. G. R. SHUBIN, A. B. STEPHENS, AND J. B. BELL, *in* "Numerical Grid Generation," Elsevier/North-Holland, New York, (1982).
31. J. B. BELL, G. R. SHUBIN, AND A. B. STEPHENS, *J. Comp. Phys.* **47** (3) (1982).
32. S. K. GODUNOV, *Ma. Sb.* **47** (1959), 271–295.
33. P. CONCUS AND W. PROSKUROWSKI, *J. Comp. Phys.* **30** (1979), 153–166.
34. H. VIVIAND, *Recherche Aerospatiale* **1** (1974), 65–68.
35. R. M. BEAM AND R. F. WARMING, *J. Comp. Phys.* **22** (1976), 87–110.
36. D. W. PEACEMAN, "Fundamentals of Numerical Reservoir Simulation." Elsevier, New York, 1977.