JOURNAL OF COMPUTATIONAL PHYSICS 115, 376–389 (1994)

On an Implicit ENO Scheme

MARK WILCOXSON AND VASILIOS MANOUSIOUTHAKIS*

Chemical Engineering Department, University of California, Los Angeles, California 90024

Received August 17, 1992; revised May 2, 1994

Essentially non-oscillatory (ENO) schemes, which have high order accuracy in regions where solutions are smooth and effectively capture shocks and discontinuities, have been developed for numerical solution of hyperbolic equations. These methods have traditionally been implemented using explicit Euler or explicit Runge–Kutta time marching schemes and consequently suffer from the Courant–Friedrichs–Lewy (CFL) time step restriction. This restriction is significant for steady-state problems and for transient problems with fronts whose speed is slow compared to the propagation speed in the rest of the domain. In this paper, a novel implicit time marching implementation of the ENO Roe scheme is developed. Simulation results which demonstrate the benefits of this implicit implementation are presented. © 1994 Academic Press, Inc.

1. INTRODUCTION

Hyperbolic partial differential equations result from modeling systems whose dissipative or dispersive terms are vanishingly small. Systems of this type studied by engineers and physicists include chromatography [1, 2], enhanced oil recovery [1–3], low temperature plasmas [4, 5], inviscid compressible fluid flow [6, 7], and cosmic flow problems [8]. The non-linear nature of these equations usually precludes analytic solution and necessitates the use of numerical solution techniques. A large body of literature has been devoted to the development of algorithms for the solution of these equations. However, the majority of these schemes are explicit in time. Due to the Courant–Friedrichs–Lewey (CFL) time step restriction, explicit time marching schemes can be inefficient. Implicit time dependent methods are needed to decrease computing time requirements. This is of particular value when limiting computing resources are available and/or multi-dimensional problems are studied.

Several alternatives to explicit schemes for hyperbolic equations have been previously presented. Beam and Warming described an implicit scheme which switches difference operators around the shock to improve stability and reduce spurious oscillations [6]. Shubin *et al.* [9] discussed a shock tracking algorithm combined with Newton iterations applicable to certain steady-state problems. Harten introduced a one-parameter family of second order accurate implicit TVD schemes [10]. Subsequently, Yee *et al.* [7] demonstrated these schemes on one- and two-dimensional problems. Also, Mulder and van Leer [8] developed an implicit implementation of Roe's linearized Riemann solver [11] and van Leer's flux-vector splitting technique [12]. More recently, Blunt and Rubin [3] investigated both fully and partially implicit flux-limiting schemes for reservoir simulation. Most of these methods are spatially second-order accurate. In the following we demonstrate that arbitrarily high order implicit schemes can be developed within the ENO framework of Shu and Osher [13, 14].

Shu and Osher described several efficient implementations of essentially non-oscillatory (ENO) schemes [13, 14]. Using numerical fluxes, rather than cell averages as in [15, 16], schemes of arbitrary spatial and temporal order were developed and presented in an explicit time stepping framework. In this paper, an implicit implementation of the ENO–Roe algorithm will be developed. Examples of the method's utility for transient and steady-state calculations will be presented and discussed. Furthermore, the techniques employed here are sufficiently general to be applicable to other ENO schemes developed in [14] such as ENO–LLF.

2. EXPLICIT ENO–ROE

To allow description of the implicit implementation of the ENO–Roe scheme, a brief review of the explicit methodology will be presented here. Shu and Osher [13, 14] have developed several ENO schemes for approximating systems of m hyperbolic conservation laws of the type

$$u_t + f(u)_x = g(u, x, t), \tag{1}$$

where $u(x, 0) = u_0(x)$, $u = (u_1, u_2, \ldots, u_m)^T$, $x = (x)$, $f(u)_x = \partial f(u)/\partial x$, and the matrix $(\partial f(u)/\partial u)$ always has m real eigenvalues and a complete set of eigenvectors. The conservative spatial approximation of (1) can be written as

$$u_t + \frac{(\hat{f}_{j+1/2} - \hat{f}_{j-1/2})}{\Delta x} = g(u, x, t). \tag{2}$$

The ENO nature of the scheme results from the method used to approximate the numerical flux $\hat{f}$.

* Author to whom correspondence should be addressed.

0021-9991/94 $6.00

2.1. *Spatial Approximation*

To develop a high order approximation of $\hat{f}$, a function $h(x)$ is defined such that its average value between $x - \Delta x/2$ and $x + \Delta x/2$ is $f(u(x))$, that is,

$$f(u(x)) = \frac{1}{\Delta x}\int_{x-\Delta x/2}^{x+\Delta x/2} h(\xi)\, d\xi. \tag{3}$$

Then

$$f(u(x))_x = \frac{h(x + \Delta x/2) - h(x - \Delta x/2)}{\Delta x}. \tag{4}$$

Equations (2) and (4) indicate that $\hat{f}_{j+1/2}$ should approximate $h(x + \Delta x/2)$ to high order. Because it is not easy to obtain $h(x)$ directly from (3), the "reconstruction via primitive function" technique [15, 16] is employed. The primitive function of $h(x)$ is introduced as

$$H(x) \triangleq \int_{-\infty}^{x} h(\xi)\, d\xi \tag{5}$$

and can be evaluated at $x_{j+1/2}$ as

$$\begin{aligned} H(x_{j+1/2}) &= \int_{-\infty}^{x_{j-1/2}} h(\xi)\, d\xi \\ &= \sum_{k=-\infty}^{j} \int_{x_{k-1/2}}^{x_{k+1/2}} h(\xi)\, d\xi = \Delta x \sum_{k=-\infty}^{j} f(u(x_k)). \end{aligned} \tag{6}$$

As will be shown later, the lower limit of $-\infty$ is irrelevant and can be changed to any fixed grid point $x_{j_o+1/2}$. In either case, Eq. (5) implies that

$$\frac{dH(x)}{dx} = h(x). \tag{7}$$

Thus, $h(x)$ can be approximated by first constructing and subsequently differentiating an interpolating polynomial for $H(x)$. This approximating polynomial is termed the reconstruction polynomial, $R(x)$. Defining the ith divided difference of $H(x)$ with leftmost point $x_{m_l-1/2}$ as $H[x_{m_l-1/2}, \ldots, x_{m_l+i-1/2}]$, the $(r+1)$th-order accurate approximation of $H(x)$ can be written as

$$\begin{aligned} R_{j+1/2}(x) = {} & H(x_{j+1/2}) + H[x_{m_1-1/2}, x_{m_1+1/2}](x - x_{m_1-1/2}) + \cdots \\ & + H[x_{m_r-1/2}, \ldots, x_{m_r+r-1/2}](x - x_{m_r-1/2}) \\ & \cdots (x - x_{m_r+r-3/2}). \end{aligned} \tag{8}$$

The interpolating polynomial $R_{j+1/2}(x)$ is constructed from the values of $H(x)$ at $r+1$ adjacent grid points and the subscript $j + 1/2$ denotes that $x_{j+1/2}$ belongs to the domain of interpolation. These grid points are termed the stencil. Approximation of $dH(x_{j+1/2})/dx$ only requires that the point $x_{j+1/2}$ be one of the points in the stencil. Thus, there are $r+1$ possible stencils for approximation of $R_{j+1/2}(x)$. The ENO scheme employs two rules to select the stencil for $R(x)$. The first rule dictates that the left point of the first divided difference (x_{m_1}) is chosen to ensure upwinding. The second rule specifies that each point added to the stencil is chosen to minimize the absolute value of the resulting divided difference and construct the interpolating polynomial in an ENO fashion.

The above formula for $R_{j+1/2}(x)$, Eq. (8), is computationally useful because the divided differences $H(x)$ can be expressed in terms of those of $f(u(x))$ as

$$H[x_{l-1/2}, x_{l+1/2}] = f[u(x_l)] \tag{9}$$

and

$$H[x_{l-1/2}, x_{l+1/2}, \ldots, x_{l+k+1/2}] = \frac{1}{k+1} f[u(x_l), \ldots, u(x_{l+k})]. \tag{10}$$

Incorporating Eqs. (6), (9), and (10), Eq. (8) can be written as

$$\begin{aligned} R_{j+1/2}(x) = {} & \Delta x \sum_{k=-\infty}^{j} f(u(x_k)) + f[u(x_{m_1})](x - x_{m_1-1/2}) + \cdots \\ & + \frac{1}{r} f[u(x_{m_r}), \ldots, u(x_{m_r+r-1})](x - x_{m_r-1/2}) \\ & \cdots (x - x_{m_r+r-3/2}). \end{aligned} \tag{11}$$

The infinite sum in Eq. (11) can be made finite by employing a fixed lower bound (i.e., $j_o + \frac{1}{2}$) in the definition of the primitive function $H(x)$. However, the lower bound is unimportant since only the derivative of $R(x)$ is of interest. Differentiating and evaluating Eq. (11) at $x_{j+1/2}$ yields the $(r+1)$th-order accurate approximation of $h(x + \Delta x/2)$,

$$\begin{aligned} \hat{f}_{j+1/2} = {} & \left(\frac{dR_{j+1/2}(x)}{dx}\right)_{x=x_{j+1/2}} = f[u(x_{m_1})] \\ & + \sum_{k=1}^{r-1} \frac{1}{k+1} f[u(x_{m_{l+1}}), \ldots, u(x_{m_{l+1}+k})] \\ & \times \sum_{l=0}^{k} \prod_{\substack{i=0 \\ i\neq l}}^{k} (x_{j+1/2} - x_{m_{l+1}+i-1/2}). \end{aligned} \tag{12}$$

2.2. *Algorithm*

The first rule of the ENO scheme stipulates that the first point (x_{m_1}) in the stencil is chosen to ensure upwinding. To quantify upwinding, we evaluate the Roe speed at $j + \frac{1}{2}$, $\hat{a}_{j+1/2}$, which is defined as

$$\hat{a}_{j+1/2} = \frac{f(u_{j+1}) - f(u_j)}{u_{j+1} - u_j}. \tag{13}$$

If $\hat{a}_{j+1/2}$ is nonnegative (negative) then the wind is from j ($j+1$) to $j+1$ (j). The first stencil point is then defined as

$$m_1 = \begin{cases} j & \text{if } \hat{a}_{j+1/2} \geq 0 \\ j+1 & \text{if } \hat{a}_{j+1/2} < 0. \end{cases} \tag{14}$$

The second rule requires that subsequent points of the stencil are chosen so as to achieve the locally smoothest interpolating polynomial. Thus, for $i = 1$ to $r-1$, the $(i+1)$th point which produces the smaller absolute value of the ith divided difference is added to the stencil. Then, m_{i+1}, the leftmost point used in the ith divided difference, is

$$m_{i+1} = \begin{cases} m_i - 1, & |a^i| \geq |b^i| \\ m_i, & |a^i| < |b^i| \end{cases} \tag{15}$$

with $a^i = f[u(x_{m_i}), \ldots, u(x_{m_i+i})]$ and $b^i = f[u(x_{m_i-1}), \ldots, u(x_{m_i+i-1})]$. To minimize the number of mathematical operations, the smaller divided difference is saved,

$$c^i = \begin{cases} b^i, & |a^i| \geq |b^i| \\ a^i, & |a^i| < |b^i| \end{cases}, \tag{16}$$

and Eq. (12) is used to calculate $\hat{f}_{j+1/2}$,

$$\hat{f}_{j+1/2} = f(u_{m_1}) + \sum_{k=1}^{r-1} \frac{1}{k+1} c^k \sum_{i=0}^{k} \prod_{\substack{l=0 \\ l \neq i}}^{k} (x_{j+1/2} - x_{m_{k-1}+i-1/2}). \tag{17}$$

2.3. *Entropy Fix*

To exclude solutions containing non-physical "expansion shocks," the following modification is incorporated in cells where $f'(u)$ has a zero. Let $\alpha_{j+1/2} = \max_I |f'(u)|$, where I is the interval $[u_j, u_{j+1}]$. The fluxes $f^+(u)$ and $f^-(u)$, whose characteristic directions are non-negative and non-positive, respectively, are then defined as

$$f^+(u) \triangleq \tfrac{1}{2}(f(u) + \alpha_{j+1/2}u), \quad f^-(u) \triangleq \tfrac{1}{2}(f(u) - \alpha_{j+1/2}u). \tag{18}$$

Then $\hat{f}^+_{j+1/2}$ and $\hat{f}^-_{j+1/2}$ are calculated as before with initially $m_1 = j$ and $m_1 = j+1$, respectively. The flux is then found as

$$\hat{f}_{j+1/2} = \hat{f}^+_{j+1/2} + \hat{f}^-_{j+1/2}. \tag{19}$$

If $f(u)$ is convex, this is needed only if $f'(u_j) < 0 < f'(u_{j+1})$.

2.4. *System of Equations*

For a system of equations, the fluxes must be decomposed or diagonalized locally. The jacobian of the system, $(\partial f/\partial u)$, is assumed to have distinct real eigenvalues and a complete set of eigenvectors. The matrix W is defined such that its columns are the eigenvectors of the jacobian. Letting $\bar{u}_{j+1/2} = u^{\text{Roe}}(u_j, u_{j+1})$ [11], the decomposed fluxes can be defined as

$$\bar{f}(z) = W^{-1}(\bar{u}_{j+1/2}) f(z). \tag{20}$$

Letting $\bar{a}_{j+1/2} = \Lambda(\bar{u}_{j+1/2})$ for each field, the divided differences and the interpolating polynomial can be calculated exactly as in the scalar equation case. Thus, after $\hat{\bar{f}}(z)$ is calculated, transformation back to the original variables is then performed as

$$\hat{f}_{j+1/2} = W(\bar{u}_{j+1/2}) \hat{\bar{f}}_{j+1/2}. \tag{21}$$

3. IMPLICIT ENO–ROE

Following Harten [10], an implicit approximation of (1) can be written as

$$\begin{aligned} u_j^{n+1} - u_j^n + \lambda[\theta(\hat{f}^{n+1}_{j+1/2} - \hat{f}^{n+1}_{j-1/2}) + (1-\theta)(\hat{f}^n_{j+1/2} - \hat{f}^n_{j-1/2})] \\ = \Delta t(\theta g_j^{n+1} + (1-\theta) g_j^n) \end{aligned} \tag{22}$$

with $\lambda \triangleq \Delta t/\Delta x$.

The parameter θ controls the degree of implicitness of (22). For $\theta = 0$, 0.5, and 1.0, the scheme becomes first-order explicit (forward Euler), second-order implicit (trapezoidal), and first-order implicit (backward Euler), respectively. The parameter θ can be chosen so as to retain the ENO properties of the scheme. Below, it is demonstrated that, for a scalar homogeneous equation, (22) is guaranteed to be ENO if $\lambda \leq \lambda_e/(1-\theta)$, where λ_e is the stability limit for the implicit scheme.

FACT. *Consider the homogeneous version of Eq.* (1) *and its* r*th spatial order implicit and explicit approximations corresponding to* (22) *with* $\theta \neq 0$ *and* $\theta = 0$, *respectively. Let the total variation of a function* u *be defined as* $\text{TV}(u) \triangleq \sum_j |u_{j+1} - u_j|$. *Let also the explicit version of* (22) *satisfy the* ENO *criterion* ($\text{TV}(u^{n+1}) \leq \text{TV}(u^n) + O(h^r)$) *under the* CFL *restriction,* $\lambda_e \triangleq \Delta t_e/\Delta x \leq \lambda_e$. *Then the implicit version of* (22) *satisfies the* ENO *criterion under the modified* CFL *restriction,* $\lambda_i \triangleq \Delta t_i/\Delta x \leq \lambda_e/(1-\theta)$.

Proof. For a scalar homogeneous equation, the spatially rth-order implicit and explicit schemes can be written as

$$u_j^{n+1} = u_j^n - \theta\lambda_i \Delta f_j(u^{n+1}) - (1-\theta)\lambda_i \Delta f_j(u^n) \tag{23}$$

$$u_j^{n+1} = u_j^n - \lambda_e \Delta f_j(u^n), \tag{24}$$

respectively, where

$$\Delta f_j(u^n) = \hat{f}(u^n_{j+s_{j+1/2}}, \ldots, u^n_{j+s_{j+1/2}+r-1}) - \hat{f}(u^n_{j+s_{j-1/2}}, \ldots, u^n_{j+s_{j-1/2}+r-1}) = \hat{f}^n_{j+1/2} - \hat{f}^n_{j-1/2} \tag{25}$$

and $\hat{f}_{j+1/2}$ is a consistent numerical flux,

$$\hat{f}(u, \ldots, u) = f(u). \tag{26}$$

Defining β as the ratio of implicit to explicit time steps ($\beta = \lambda_i/\lambda_e$), the implicit scheme (23) can be written as

$$u_j^{n+1} = u_j^n - \theta\beta\lambda_e \Delta f_j(u^{n+1}) - (1 - \theta)\beta\lambda_e \Delta f_j(u^n). \tag{27}$$

Equation (24) can be written for time levels n and $n + 1$ as

$$w_j^{n+2} = u_j^{n+1} - \lambda_e \Delta f_j(u^{n+1}) \tag{28}$$

$$v_j^{n+1} = u_j^n - \lambda_e \Delta f_j(u^n). \tag{29}$$

Equations (28) and (29) can be used to replace $\lambda_e \Delta f_j(u^{n+1})$ and $\lambda_e \Delta f_j(u^n)$, respectively:

$$u_j^{n+1} = u_j^n + \theta\beta(w_j^{n+2} - u_j^{n+1}) + (1 - \theta)\beta(v_j^{n+1} - u_j^n) \tag{30}$$

or

$$(1 + \theta\beta)u_j^{n+1} = (1 - \theta)\beta v_j^{n+1} + \theta\beta w_j^{n+2} + (1 - (1 - \theta)\beta)u_j^n. \tag{31}$$

Choosing θ, such that $\beta(1 - \theta) = 1$ simplifies (31) to

$$(1 + \theta\beta)u_j^{n+1} = v_j^{n+1} + \theta\beta w_j^{n+2}. \tag{32}$$

Substituting $j + 1$ for j in Eq. (32) and subtracting (32) from this yields

$$(1 + \theta\beta)(u_{j+1}^{n+1} - u_j^{n+1}) = (v_{j+1}^{n+1} - v_j^{n+1}) + \theta\beta(w_{j+1}^{n+2} - w_j^{n+2}). \tag{33}$$

Employing the triangle inequality and summing over all j yields

$$(1 + \theta\beta)\mathrm{TV}(u^{n+1}) \le \mathrm{TV}(v^{n+1}) + \theta\beta\mathrm{TV}(w^{n+2}). \tag{34}$$

If $\lambda_e \le \lambda_a$ then

$$\mathrm{TV}(v^{n+1}) \le \mathrm{TV}(u^n) + O(h^r) \tag{35}$$

$$\mathrm{TV}(w^{n+2}) \le \mathrm{TV}(u^{n+1}) + O(h^r). \tag{36}$$

Incorporating (36), (34) can be rewritten as

$$(1 + \theta\beta)\mathrm{TV}(u^{n+1}) \le \mathrm{TV}(v^{n+1}) + \theta\beta\mathrm{TV}(u^{n+1}) + O(h^r) \tag{37}$$

or, equivalently,

$$\mathrm{TV}(u^{n+1}) \le \mathrm{TV}(v^{n+1}) + O(h^r). \tag{38}$$

Incorporating (35), (38) becomes

$$\mathrm{TV}(u^{n+1}) \le \mathrm{TV}(u^n) + O(h^r). \tag{39}$$

Equation (39) guarantees that the implicit scheme is ENO. The restriction on λ_i can be determined, given that $\lambda_e \le \lambda_a$, $\beta = \lambda_i/\lambda_e$, and $\beta(1 - \theta) = 1$. Then

$$\lambda_e = \frac{\lambda_i}{\beta} = \lambda_i(1 - \theta) \le \lambda_a \Rightarrow \lambda_i \le \frac{\lambda_a}{(1 - \theta)}. \tag{40}$$

The general applicability of the proof shown above should be emphasized. If the explicit scheme (i.e., Eqs. (28) and (29)) is assumed to be TVD, the implicit analog (23) would be TVD as long as (40) is satisfied. This is consistent with previous results for implicit TVD schemes [10].

Equation (40) is a sufficient condition for the implicit scheme to remain ENO. However, determining u_j^{n+1} requires solving Eq. (23). Since, the algebraic equations resulting from the spatial discretization are generally nonlinear, an iterative technique such as Newton's method is required. The use of (28) in the derivation of (40) requires the stencil chosen for $\hat{f}^{n+1}_{j+1/2}$ to be ENO in terms of u^{n+1}. Thus, the stencil at time $n + 1$ will depend on the unknown u^{n+1}. This may give rise to non-smooth jacobian elements and hamper convergence. However, this did not occur for the examples considered here. Solution typically required three or fewer iterations. Furthermore, non-smoothness of jacobian elements is not unique to this implicit ENO implementation. Implicit TVD schemes also generate this phenomenon. Finally, to be computationally efficient, compact relations for the jacobian elements are needed. This can be achieved by providing simple analytical expressions for the approximate flux $\hat{f}_{j+1/2}$.

3.1. *Flux Formulation*

Evaluation of $\partial g/\partial u$ presents no difficulties conceptually; however, determining analytic expressions for $\partial\hat{f}_{j+1/2}/\partial u$ using the ENO implementation described by Shu and Osher seems to be a formidable task. An equivalent method for calculating $\hat{f}_{j+1/2}$ which is amenable to analytic expression for the jacobian elements is needed.

As discussed above, for the scheme to remain ENO, the implicit implementation must use the Roe speed and the comparison of divided differences to determine the stencil. The points chosen in this way to evaluate $\hat{f}_{j+1/2}$ will be denoted as $x_{j+s_{j+1/2}+i-1/2}$ for $i = 0, \ldots, r$, where r is the order of the polynomial. To simplify notation, the j dependence of s will be understood. Evaluation of $dH(x_{j+1/2})/dx$ can be accomplished either by differentiation of the interpolating polynomial as done by Shu and Osher [13, 14] or as shown here by using the rth-order finite difference approximation based on the values of $H(x)$ at the points $x_{j+s+i-1/2}$ for $i = 0, \ldots, r$. These two approaches are

equivalent, but the latter can be used to give compact jacobian expressions in a straightforward manner. Thus, for the scalar equation the expression for calculating $\hat{f}_{j+1/2}$ can be written as

$$\frac{dH(x_{j+1/2})}{dx} \approx \hat{f}_{j+1/2} = \frac{1}{\Delta x}\sum_{i=0}^{r} c_i H(x_{j+s+i-1/2}). \tag{41}$$

The coefficients c_i depend on the stencil chosen. For example, if $r = 2$ and $s = 0$ then $c_0 = -\frac{1}{2}$, $c_1 = 0$, $c_2 = \frac{1}{2}$ while for $s = -1$, $c_0 = \frac{1}{2}$, $c_1 = -2$, $c_2 = \frac{3}{2}$. Again, Eq. (6) can be used to replace $H(x)$. After rearrangement,

$$\begin{aligned}\hat{f}_{j+1/2} = &\sum_{i=0}^{r} c_i \sum_{k=-\infty}^{j+s-1} f(u_k) + \sum_{i=1}^{r} c_i f(u_{j+s}) \\ &+ \sum_{i=2}^{r} c_i f(u_{j+s+1}) + \cdots + c_r f(u_{j+s+r-1}).\end{aligned} \tag{42}$$

If the $r + 1$ coefficients c_i are chosen to give an rth-order-accurate approximation, then $\sum_{i=0}^{r} c_i = 0$. With this fact, the notation can be simplified by defining a new set of coefficients (d_k),

$$d_k = \sum_{i=k}^{r} c_i \quad \text{for } k = 1, r. \tag{43}$$

With these coefficients $\hat{f}$ becomes

$$\hat{f}_{j+1/2} = \sum_{i=1}^{r} d_i f(u_{j+s+i-1}). \tag{44}$$

The coefficients c_i and therefore d_i depend on the relation between $j + s - \frac{1}{2}$ and $j + \frac{1}{2}$. Since s can vary from $1 - r$ to 1, there are $r + 1$ sets of coefficients d_i for $i = 1, \ldots, r$. One can place these sets of coefficients into an $r + 1$ by r matrix D with the sets of coefficients ordered so that those for $s = 1 - r$ are in row one and those for $s = 1$ are in row $r + 1$. Then for an arbitrary choice of stencil $(j + s)$, we have

$$\hat{f}_{j+1/2} = \sum_{i=1}^{r} D_{s+r,i} f(u_{j+s+i-1}) \tag{45}$$

and the jacobian of the scalar flux can then be written simply as

$$\frac{\partial \hat{f}_{j+1/2}}{\partial u_{j+s+i-1}} = D_{s+r,i}\left(\frac{\partial f}{\partial u}\right)_{u_{j+s+i-1}}. \tag{46}$$

The entropy fix can be written as

$$\hat{f}^{\pm}_{j+1/2} = \frac{1}{2}\sum_{i=1}^{r} D_{s^{\pm}+r,i}[f(u_{j+s^{\pm}+i-1}) \pm \alpha_{j+1/2} u_{j+s^{\pm}+i-1}]. \tag{47}$$

For the p_ith flux of a system of p equations, (45) can be generalized to

$$\begin{aligned}\hat{f}_{p_i,j+1/2} = &\sum_{q=1}^{p}\sum_{i=1}^{r} W_{j+1/2,p_i,q} D_{s_q+r,i} \\ &\times \sum_{t=1}^{p} W^{-1}_{j+1/2,q,t} f_{t,j+s_q+i-1}.\end{aligned} \tag{48}$$

3.2. *Scalar Flux Example*

To clarify the compact expressions in the preceding section, consider the third-order spatial approximation for a scalar homogenous equation. In this case, r equals 3, s varies from -2 to 1 and the coefficient matrix D (4×3) can be shown to be

$$D = \begin{bmatrix} \frac{2}{6} & -\frac{7}{6} & \frac{11}{6} \\ -\frac{1}{6} & \frac{5}{6} & \frac{2}{6} \\ \frac{2}{6} & \frac{5}{6} & -\frac{1}{6} \\ \frac{11}{6} & -\frac{7}{6} & \frac{2}{6} \end{bmatrix}. \tag{49}$$

Thus, from (45) the possible flux formulations are

$$\begin{aligned}\hat{f}_{j+1/2} &= \tfrac{2}{6}f_{j-2} - \tfrac{7}{6}f_{j-1} + \tfrac{11}{6}f_j \\ &= -\tfrac{1}{6}f_{j-1} + \tfrac{5}{6}f_j + \tfrac{2}{6}f_{j+1} \\ &= \tfrac{2}{6}f_j + \tfrac{5}{6}f_{j+1} - \tfrac{1}{6}f_{j+2} \\ &= \tfrac{11}{6}f_{j+1} - \tfrac{7}{6}f_{j+2} + \tfrac{2}{6}f_{j+3}.\end{aligned} \tag{50}$$

For any of the flux expressions shown above, the jacobian elements can be readily determined. For example, the jacobian for the stencil $[x_j, x_{j+1}, x_{j+2}]$, $s = 0$, is

$$\frac{\partial \hat{f}_{j+1/2}}{\partial u_j} = \frac{2}{6}\left(\frac{\partial f}{\partial u}\right)_j \tag{51}$$

$$\frac{\partial \hat{f}_{j+1/2}}{\partial u_{j+1}} = \frac{5}{6}\left(\frac{\partial f}{\partial u}\right)_{j+1} \tag{52}$$

$$\frac{\partial \hat{f}_{j+1/2}}{\partial u_{j+2}} = -\frac{1}{6}\left(\frac{\partial f}{\partial u}\right)_{j+2}. \tag{53}$$

The simple relations shown above are important since they provide complete expressions for the jacobian which can dramatically improve the convergence properties of the Newton iterations.

4. NUMERICAL RESULTS

In this section, the convergence and resolution properties of the implicit ENO scheme are investigated. For the following discussion, the spatial order of the scheme refers to the order of approximation of $\hat{f}$. The scheme's applicability to transient

solutions was first explored by considering several versions of the linear advection equation. For these solutions, the Newton iterations employed at each time step were considered converged if,

$$\left|\frac{u_j^{k+1} - u_j^k}{u_j^k}\right| \leq 10^{-7} \quad \forall j; \tag{54}$$

u^{k+1} and u^k refer to the value of u at the $(k + 1)$th and kth Newton iterations, respectively. For all values of θ and Δt considered here, convergence typically required three or fewer iterations. However, in a few instances, the iterations became trapped in a limit cycle. To break this cycle, the stencil from the fifth iteration was used for subsequent iterations. Convergence was then reached by the sixth iteration. The convergence of the Newton iterations for a non-linear problem was also investigated. Although detailed results are not shown here, the implicit scheme was also applied to Burgers' equation with initial conditions as given in Eq. (56). Typically, three of fewer iterations were required per time step and the non-smoothness of the jacobian was not problematic.

To demonstrate the scheme's characteristics for steady-state calculations, a quasi-one-dimensional nozzle flow with shock was investigated. The convergence criteria (54) was employed for all variables. Typically, four or fewer iterations per time step were needed for CFL numbers up to 1000. Again, for the few instances that the iterations became trapped in a limit cycle, the stencil from the fifth iteration was used for subsequent iterations. Convergence was then reached by the seventh iteration.

4.1. *Constant Coefficient Linear Advection Equation*

To investigate the attributes of the proposed implicit ENO scheme, the linear advection equation

$$u_t + u_x = 0 \tag{55}$$

with initially discontinuous profile

$$u_o = \begin{cases} 1.0, & 0.2 \geq x \geq 0 \\ 0.5, & 1 \geq x > 0.2 \end{cases} \tag{56}$$

was considered. Figure 1 depicts the dependence of the solution profiles, at $t = 0.4$, on the parameter θ which controls the degree of implicitness of the scheme. The spatially third-order ENO scheme was employed on a grid of 100 points. The CFL number was fixed at 0.5. The profiles become increasingly smeared with increasing θ. Thus, although for $\theta = 0.5$ the scheme has second-order temporal accuracy, it produced more diffuse results than a temporally first-order accurate explicit scheme. The diffusive nature of the scheme can be understood by examining the truncation error dependence on θ. This analysis requires a Taylor series representation of $u(x)$. For this to be valid, the initial condition $u_o(x)$ is assumed to vary from 1.0 to 0.5 over the domain $0.2 \leq x \leq 0.2 + \varepsilon$ in such a way that $u_o(x)$ is continuous and continuously differentiable. The parameter ε is a constant which can be made arbitrarily small. After some manipulation, the implicit scheme can be shown to satisfy

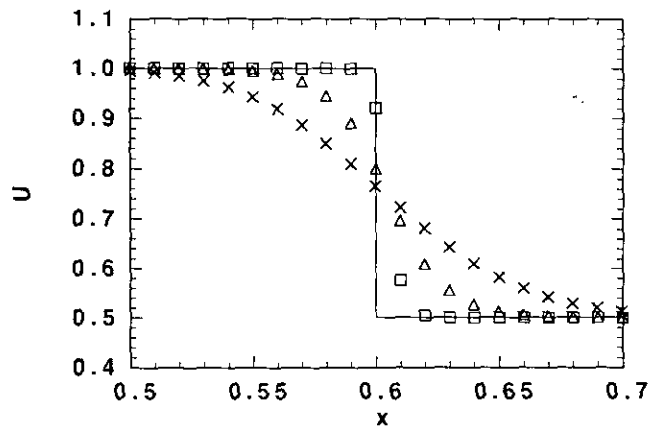

FIG. 1. Step discontinuity initial condition at $t = 0.4$ with $N = 101$ and CFL = 0.5: (———) exact; (□) $\theta = 0$; (△) $\theta = 0.5$; (×) $\theta = 1.0$.

$$\frac{u^{n+1} - u^n}{\Delta t} + \theta u_x^{n+1} + (1 - \theta)u_x^n = u_t^{n+1/2} + u_x^{n+1/2} + \frac{\Delta t}{2}(1 - 2\theta)u_{xx}^{n+1/2} + O(\Delta t^2) + \cdots. \tag{57}$$

The first two terms on the right-hand side of Eq. (57) correspond to the original equation (55). The third term, whose coefficient contains the parameter θ, represents some artificial diffusion which has been introduced by the time discretization. If θ is less than 0.5, the coefficient is positive and this term represents "anti" or "backward" diffusion. Therefore, the leading order error term from the time discretization opposes any numerical diffusion introduced in the spatial approximation of u_x. For $\theta = 0.5$, this error term vanishes and the approximation is second order in time. However, since the error term caused sharpening of the discontinuity, the second-order results are more diffuse than the first-order ones. For $\theta > 0.5$, the coefficient is negative, and the diffusion of the scheme increases with θ. Thus, the results continue to deteriorate with increasing θ. Equation (57) and the results in Fig. 1 indicate that the sharpest profile will result if θ is chosen to be as small as possible.

Solution profiles for different CFL numbers are shown in Fig. 2. For these profiles, θ was chosen to satisfy the following form of Eq. (40):

$$\theta = \max(0, 1 - 0.5\ \Delta x/\Delta t). \tag{58}$$

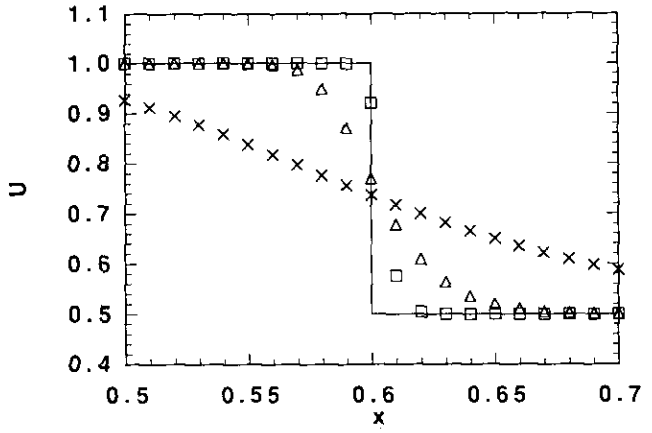

FIG. 2. Step discontinuity initial condition at $t = 0.4$ with $N = 101$: (———) exact; (□) CFL ≈ 0.5 and $\theta = 0$; (△) CFL ≈ 1.0 and $\theta = 0.5$; (×) CFL = 5 and $\theta = 0.9$.

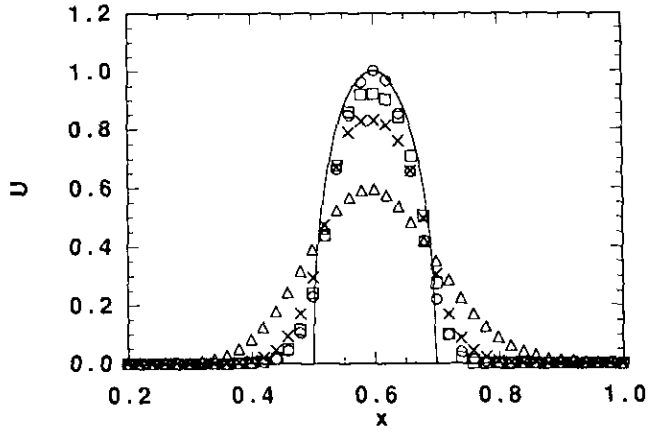

FIG. 4. Solution dependence on spatial order of approximation for semi-ellipse initial condition at $t = 0.4$ with $N = 51$, CFL = 0.4, and $\theta = 0.5$: (———) exact; (△) upwind; (×) ENO-2; (○) ENO-3; (□) TVD-2.

For this form of (40), the scheme remains explicit for CFL numbers less than 0.5 and subsequently uses the lowest value of θ possible. The smearing with increasing CFL number is quite pronounced. Again temporal truncation error is the cause of this trend. An expression for the error induced diffusion in terms of Δt and Δx can be found by replacing θ with Eq. (58),

$$D_{an} = \frac{\Delta t}{2}(1 - 2\theta) = \frac{\Delta t}{2}\min\left(1, \frac{\Delta x}{\Delta t} - 1\right). \tag{59}$$

Thus, the error term causes sharpening if the CFL number is less than 1 and smearing for CFL numbers greater than 1. The smearing in Fig. 2 is more substantial than that in Fig. 1. This is because both θ and Δt are increasing in Fig. 2, while the solutions in Fig. 1 represent a fixed Δt.

In Fig. 3, the impact of the spatial approximation of u_x is illustrated. Results employing first-order upwinding, second- and third-order ENO, and the second-order implicit TVD scheme of Blunt and Rubin are shown. For these results, the CFL number and θ were specified as 0.4 and 0.5, respectively. Upwinding gives more smeared results than any of the high-order schemes. All three high-order schemes give comparable results. The third-order ENO and the second-order TVD schemes give nearly identical results which are slightly superior to the second-order ENO results. These profiles indicate that, as noted by Blunt and Rubin [3], high-order accurate implicit schemes offer some improvement over standard first-order upwind implicit schemes.

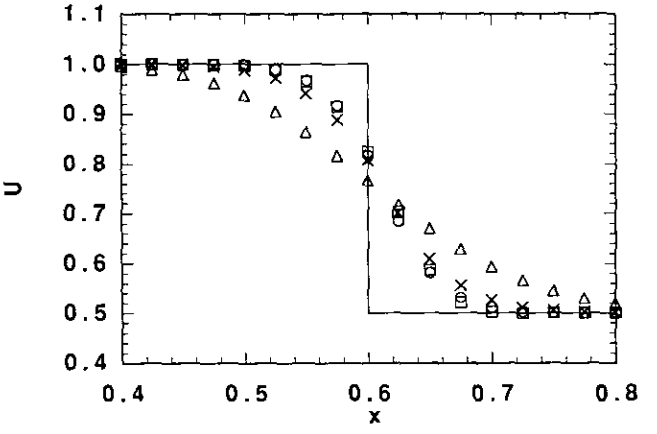

FIG. 3. Solution dependence on spatial order of approximation for step discontinuity initial condition at $t = 0.4$ with $N = 41$, CFL = 0.4, and $\theta = 0.5$: (———) exact; (△) upwind; (×) ENO-2; (○) ENO-3; (□) TVD-2.

The comparable resolution of the propagating step by the second-order TVD and third-order ENO schemes demonstrates that higher order approximations do not always improve accuracy. In some instances, the lower order error terms may be beneficial. For example, the first-order explicit scheme produced better results than the first-order backward Euler as well as the second-order implicit scheme for the propagating step discontinuity (see Fig. 1). However, temporally second-order accurate schemes generally provide significant improvement over first-order temporal schemes (see Fig. 7). To demonstrate that higher order spatial approximations also often lead to significant improvement, two other initial conditions, a semi-ellipse and a continuous $\sin^2(x)$ pulse, were also considered.

For the semi-ellipse initial condition,

$$u_0 = \begin{cases} 0.0, & 0.1 > x \geq 0 \\ 1.0 - \dfrac{(x - 0.2)^2}{0.01}, & 0.3 \geq x \geq 0.1 \\ 0.0, & 1 \geq x > 0.3, \end{cases} \tag{60}$$

the resulting profiles at $t = 0.4$ with $N = 51$, CFL = 0.4, and $\theta = 0.5$ are shown in Fig. 4. Again upwinding gives very

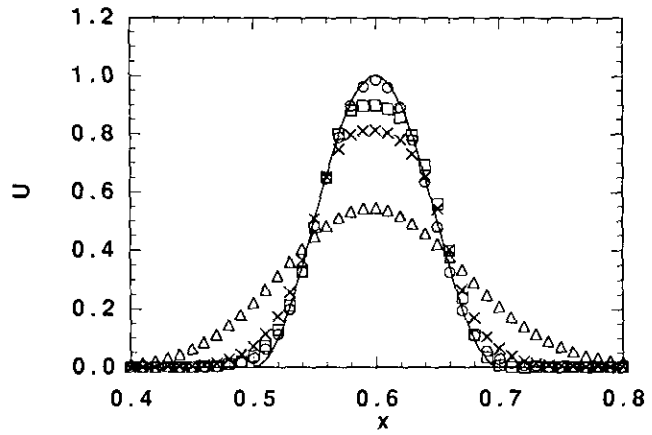

FIG. 5. Solution dependence on spatial order of approximation for a $\sin^2$ pulse initial condition at $t = 0.4$ with $N = 101$, CFL = 0.4, and $\theta = 0.5$: (———) exact; (△) upwind; (×) ENO-2; (○) ENO-3; (□) TVD-2.

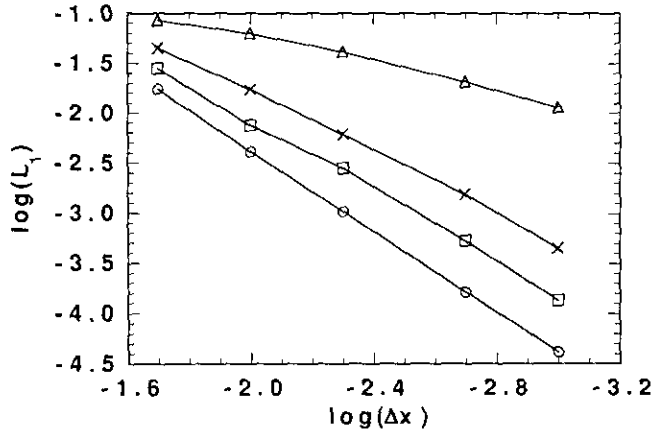

FIG. 6. L_1 error dependence on grid spacing for CFL = 0.4 and $\theta = 0.5$: (△) upwind; (×) ENO-2; (○) ENO-3; (□) TVD-2.

smeared results with an error of approximately 40% in the peak value. The second-order TVD and ENO results are approximately 10% and 20% below the peak, while the third-order ENO scheme provides an accurate peak value. Thus, the resolution near the extrema improved with increasing order of spatial approximation.

The behavior of these schemes for the initially continuous profile,

$$u_0 = \begin{cases} 0.0, & 0.1 > x \geq 0 \\ \sin^2(5\pi(x - 0.1)), & 0.3 \geq x \geq 0.1 \\ 0.0, & 1 \geq x > 0.3, \end{cases} \tag{61}$$

at $t = 0.4$ with $N = 101$, CFL = 0.4, and $\theta = 0.5$ is shown in Fig. 5. Again, as expected the accuracy of the solutions decreases with decreasing order of spatial approximation. The peak value for the third-order ENO scheme is 0.985 while the second-order ENO scheme gives 0.811. The second-order TVD scheme provides more accurate results than the second-order ENO scheme and has a peak value of 0.897. The first-order upwinding scheme smears the profile and produces a peak value of 0.545.

In summary, the results in Figs. 3, 4, and 5 demonstrate that the resolution of the proposed implicit ENO scheme improves with increasing order of spatial approximation for the initial conditions considered here. For the step discontinuity initial condition, the second-order TVD and third-order ENO schemes gave comparable results. However, for two other initial conditions, the third-order ENO scheme showed significant improvement around the extrema compared to the second-order TVD scheme. Interestingly, for all three initial conditions, the second-order TVD scheme produced more accurate results than the second-order ENO scheme.

To demonstrate the second-order temporal accuracy obtained with the implicit scheme when θ is fixed at 0.5, the initial condition (61) was employed. Figure 6 shows the L_1 error as a function of grid spacing for different spatial approximations. The L_1 error is defined as

$$L_1 = \sum_{i=1}^{N} \frac{1}{N} |u_i - u_i^{\text{exact}}|. \tag{62}$$

For the upwind scheme, the error is dominated by the first-order spatial approximation and first-order convergence is observed. All three high-order schemes show second-order convergence. The overall second-order accuracy of the spatially third-order ENO scheme is due to the temporal discretization. As expected the third-order ENO scheme has a smaller error for all values of Δx than any of the lower order schemes. The degradation of the solution for other values of θ is shown in Fig. 7. Figure 7 depicts the dependence of the solution profile

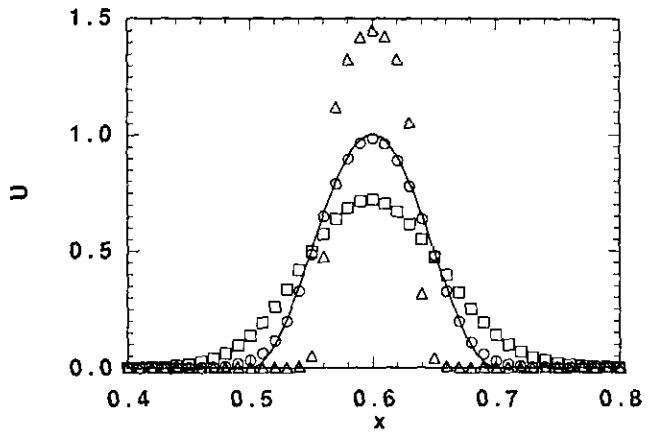

FIG. 7. Impact of temporal approximation for a $\sin^2$ pulse initial condition at $t = 0.4$ with $N = 101$ and CFL = 0.4: (△) $\theta = 0$; (○) $\theta = 0.5$; (□) $\theta = 1.0$.

at $t = 0.4$ on the parameter θ for the third-order ENO scheme employing 101 grid points and a CFL number of 0.4. An accurate solution is obtained with $\theta = 0.5$. However, for $\theta = 0$ and $\theta = 1$, the impact of temporal truncation error is significant. The compressive nature of the first-order explicit time discretization ($\theta = 0$) causes the peak value to be approximately 50% too large and the width of the pulse to be too narrow. The diffusive nature of the backward Euler temporal truncation error ($\theta = 1.0$) causes broadening of the pulse as well as a low peak value. These results as well as Eq. (57) demonstrate the second-order temporal nature of the scheme for $\theta = 0.5$.

For the linear advection examples considered here, the computing requirements per time step for the implicit scheme are approximately an order of magnitude larger than those for the explicit scheme. This causes the implicit scheme to be computationally efficient only at large CFL numbers. However, for large CFL numbers, Eq. (40) forces the value of θ to be approximately one. This produces significant smearing of steep profiles (Figs. 1, 2, and 7). Reducing θ to 0.5 significantly improves resolution. Unfortunately, for this value of θ, implicit implementation increases the maximum allowable time step by only a factor of two. Such modest gains in the maximum time step will not offset the increased computation time needed for the matrix inversions performed in the implicit scheme. However, there exist transient problems for which the proposed implicit scheme can both provide accurate resolution of steep profiles and decrease computational time as compared to the explicit scheme. Consider, for example, a problem whose solution exhibits significant spatial variation in the eigenvalues of $\partial f/\partial u$. In such a problem, the solution may remain smooth throughout most of the domain except in a region with small eigenvalues, where a front may develop. Since the time step of an explicit scheme is limited by the largest eigenvalue in the domain, a small time step would have to be used to maintain stability. As noted by Blunt and Rubin [3], this limitation can be overcome by an implicit scheme with a spatially varying θ. Close to the front, a value of θ near 0.5 can be employed so as to provide accuracy, while away from the front, values of θ close to one can be employed allowing the use of a large time step without significant deterioration in accuracy. In the following section, an example which develops a slow moving front is investigated.

4.2. *Variable Coefficient Linear Advection Equation*

As an example of an equation which develops a slow moving front, consider the variable coefficient linear advection equation, with $a(x)$ and $u_a(x)$ given as

$$u_t + (a(x)u)_x = 0 \tag{63}$$

$$u_a(x)a(x) = \begin{cases} 1.0, & 0.2 \geq x \geq 0 \\ 0.0, & x > 0.2 \end{cases} \tag{64}$$

$$a(x) = 25.5 - 24.5 \tanh(20(x - 0.25)). \tag{65}$$

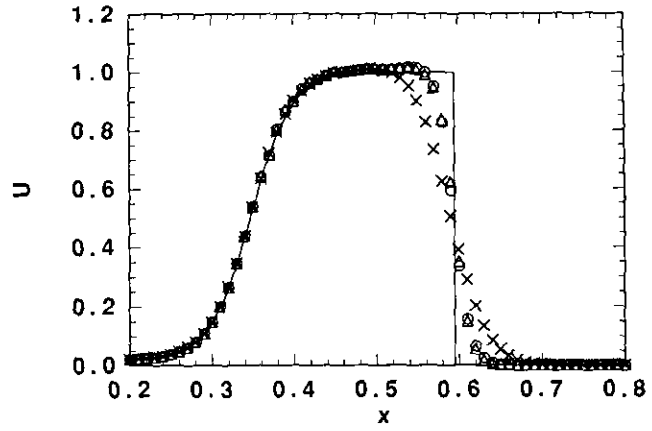

FIG. 8. Solution of the variable coefficient advection equation with discontinuous initial data using third-order ENO: (———) exact, (△) explicit ($CFL_{max} = 0.5$); (×) implicit spatially uniform θ ($CFL_{max} = 25$); (○) implicit spatially varying θ ($CFL_{max} = 25$).

The front is initially located in a region of high velocity ($a(0.2) \approx 44$). However, as time progresses, the front moves to the right and its propagation speed decreases. For $x > 0.5$ the propagation speed is approximately 1. Thus, the propagation speed changes by more than an order of magnitude in the computational domain. An explicit scheme is constrained to use a time step which satisfies the CFL restriction at $x = 0$ regardless of the location of the discontinuity. An implicit scheme can employ a variable time step which increases as the time rate of change decreases. For the implicit results shown in Figs. 8 and 9, the time step is initially set to $0.5\ \Delta x/50$ and allowed to increase to a maximum value of $0.5\ \Delta x$ during the computation. The time step is increased by a factor of 1.75 from one time step to the next if $\Delta u/u$ is less than 0.3. Other strategies which more accurately address temporal truncation error could be employed. However, this simple criterion was

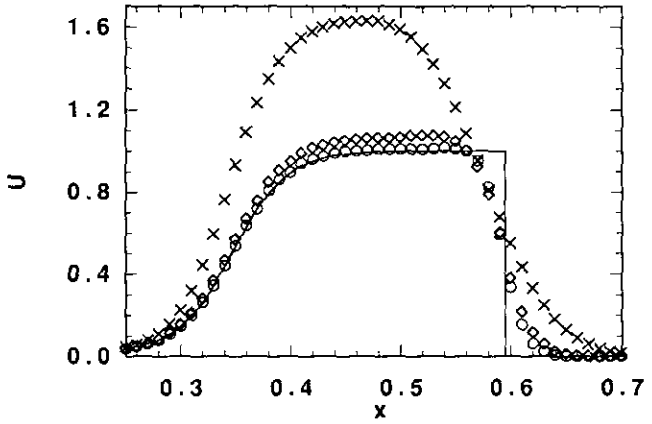

FIG. 9. Solution of the variable coefficient advection equation with discontinuous initial data using the spatially varying θ implicit scheme: (———) exact; (×) upwind; (◇) ENO-2; (○) ENO-3.

TABLE I

CPU Time Requirements

Scheme	Max CFL number	Time steps	CPU time
Explicit ($\theta = 0$)	0.5	2500	1.9
Implicit (spatially uniform θ)	25.0	70	1.0
Implicit (spatially varying θ)	25.0	74	1.1

TABLE II

Boundary Conditions

Diverging		Con-div
$(\rho k)_{\text{inlet}} = 0.5277$	$(\rho k)_{\text{inlet}} = 0.5277$	$(\rho k)_{\text{inlet}} = 1.2550$
$(mk)_{\text{inlet}} = 0.6855$	$(mk)_{\text{inlet}} = 0.6855$	$(mk)_{\text{inlet}} = 3.2630$
$(ek)_{\text{inlet}} = 1.4465$	$(ek)_{\text{inlet}} = 1.4465$	$(ek)_{\text{inlet}} = 6.6225$
$(ek)_{\text{outlet}} = 3.4354$	$(ek)_{\text{outlet}} = 3.9850$	$(ek)_{\text{outlet}} = 31.300$
$x_s = 4.816$	$x_s = 3.316$	$x_s = 6.816$

sufficient for this example. The modified CFL restriction for the implicit ENO scheme can be implemented as

$$\theta = \max\left(0.5, 1 - \frac{0.5\,\Delta x}{a(0)\,\Delta t}\right). \tag{66}$$

Since $a(0)$ is the maximum value of $a(x)$, requiring a spatially uniform θ forces θ to be near 1 for large CFL numbers. Blunt and Rubin [3] found that a spatially varying θ scheme could give superior results compared to a fixed θ scheme. To implement a variable θ scheme, θ was forced to satisfy (66) locally,

$$\theta(x) = \max\left(0.5, 1 - \frac{0.5\,\Delta x}{a(x)\,\Delta t}\right). \tag{67}$$

Choosing θ in this manner allows the scheme to retain second-order accuracy for $x > 0.35$ even at CFL numbers as large as 25.

The solutions computed at $t = 0.25$ with 101 grid points using the explicit, fixed θ implicit, and variable θ implicit third-order ENO schemes are shown in Fig. 8. For the fixed θ implementation, θ is required to be 0.98 at a CFL number of 25. The diffusive nature of this temporal discretization causes the fixed θ scheme to smear the front over approximately twice as many grid points as the explicit scheme. The variable θ implementation allows θ to remain at approximately 0.5 at the front during the computation. This produces results which are comparable to the explicit scheme.

The computation time for the three third-order schemes is shown in Table I. Implicit implementation allows a more than 30-fold decrease in the number of time steps needed. This translated into approximately a factor of two decrease in computing time. This demonstrates that the implicit scheme can decrease computing time and give results comparable to the explicit scheme for slow moving front problems. The differences in computing time become more significant as the differences in the propagation speed become larger throughout the domain.

The impact of the spatial order of approximation is shown in Fig. 9 for the variable θ scheme. The first-order results significantly deviate from the true solution. The smearing at both the front and throughout the section of increasing u ($0.3 < x < 0.4$) produces results which have essentially no plateau. Significant improvement occurs for the second-order scheme. There is now qualitative agreement with the solution. The smearing of the front is less severe. However, some noticeable error is still present at the plateau. The third-order results accurately represent the solution up to and after the discontinuity and the front is resolved over approximately six grid points.

4.3. *Quasi-One-Dimensional Nozzle*

To investigate the steady-state shock capturing and convergence properties of the proposed implicit ENO scheme, the quasi-one-dimensional nozzle was considered [9, 7]. The system of equations describing the nozzle can be written as

$$U_t + F(U)_x + H(U) = 0, \tag{68}$$

where

$$U = \begin{bmatrix} \rho k \\ mk \\ ek \end{bmatrix}, \quad F(U) = \begin{bmatrix} mk \\ (m^2/\rho + p)k \\ (e + p)mk/\rho \end{bmatrix}, \quad H(U) = \begin{bmatrix} 0 \\ -p\dfrac{\partial k}{\partial x} \\ 0 \end{bmatrix} \tag{69}$$

with $m = \rho u$ and the cross sectional area, k, given as

$$k = 1.398 + 0.347 \tanh(0.8x - 4). \tag{70}$$

As in [7], ρ, u, and e were specified for supersonic flow entering at $x = 0$ and e was specified for subsonic flow exiting at $x = 10$. The numerical values for the boundary conditions and the resulting shock location, x_s, are shown in the first column of Table II. Linear interpolation between the exact steady-state boundary values of ρ, mk, and e was used as the initial condition. Furthermore, Λ, $W_{j+1/2}$, $W^{-1}_{j+1/2}$ and the corresponding Roe-averaged forms were also taken directly from [7]. For the nozzle

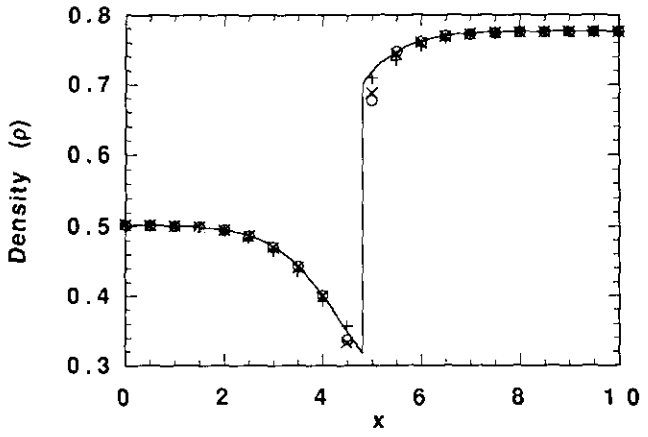

FIG. 10. Steady-state density profile in a diverging nozzle: (———) exact; (+) ENO-1; (×) ENO-2; (○) ENO-3.

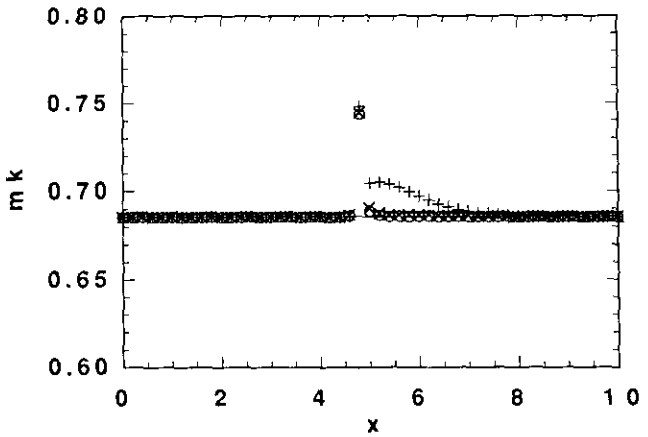

FIG. 12. Steady-state flux and nozzle area product (mk) in a diverging nozzle with $N = 51$: (———) exact; (+) ENO-1; (×) ENO-2; (○) ENO-3.

shapes considered here, the energy and density profiles behaved similarly and therefore only the density profiles are shown.

The density (ρ) profiles determined with 21 grid points are shown in Fig. 10. All three orders of spatial approximation produced similar results which are comparable to the TVD results of [7]. At points away from the shock, the second- and third-order schemes are slightly superior, while on either side of the shock, the first-order results are closer to the exact solution. If more (e.g., 51) grid points are used, the accuracy at all points increases with increasing order of spatial approximation. In Fig. 11, the product of the flux and nozzle area (mk) profiles computed using 21 grid points are shown. These profiles are affected by the order of spatial approximation. This is an interesting and somewhat unexpected result since the exact solution for mk is a flat line. The first-order profiles match the exact solution in the supersonic region. However, significant deviations occur at points where there is appreciable variation in the nozzle area and a numerical flux $\hat{f}$ from a subsonic region is

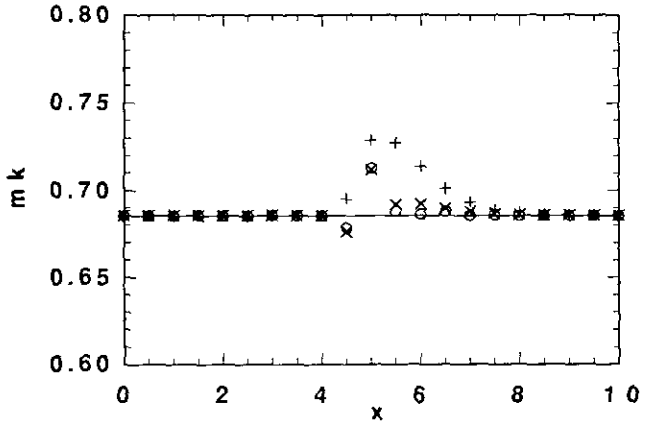

FIG. 11. Steady-state flux and nozzle area product (mk) in a diverging nozzle with $N = 21$: (———) exact; (+) ENO-1; (×) ENO-2; (○) ENO-3.

incorporated in the calculation of $f(u)_x$. These errors result from the transformation to characteristic fields. Since $m^2k/\rho + pk$ varies with x, all three transformed fluxes $\bar{f}$ vary with x. Therefore, significant errors in $\hat{\bar{f}}$ and subsequently $\hat{f}$ are caused by the first-order approximation. This is not always the case. If at a given spatial location, all fields use the same stencil, Eq. (48) can be simplified as

$$\hat{f}_{p_i,j+1/2} \approx \sum_{i=1}^{r} D_{j+s-(j-r),i} f_{p_i,j+s+i-1}. \tag{71}$$

Thus, each numerical flux ($\hat{f}_{p_i}$) is calculated using only values of the corresponding flux (f_{p_i}) and the transformation is unimportant. In the supersonic region, the first-order scheme always uses the same stencil points for all fields. Since steady state requires the numerical flux $\hat{f}_i$ to be constant, (mk) is constant in the supersonic region. In other regions (e.g., subsonic), or when high order schemes are employed, all fields may not use the same stencil. Then, the transformation becomes important and high-order approximations reduce the errors induced in $\hat{\bar{f}}$.

The flux profiles calculated with 51 points are shown in Fig. 12. Errors in the first-order scheme persist over a wide domain, while for the high-order schemes, deviations are localized near the shock. The deviation of the value of mk at the shock may persist regardless of the number of grid points employed, because the ENO–Roe algorithm is based on the first-order Roe scheme. As a result, the value of $\bar{f}_{p_i,j}$ may not be used to calculate any $\hat{\bar{f}}_{p_i}$ when $\Lambda_{p_i,j-1/2}$ and $\Lambda_{p_i,j+1/2}$ are positive and negative, respectively.

Although, the density profile results in Fig. 10 are relatively insensitive to the order of spatial approximation, this is not generally true. Consider the boundary conditions in the second column of Table II. These boundary conditions force the shock further upstream. Then, differences between the first-order and

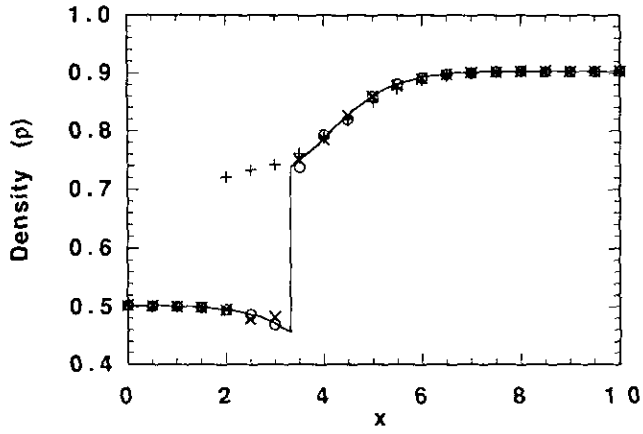

FIG. 13. Steady-state density profile in a diverging nozzle with inlet and outlet conditions as specified in the second column of Table II: (———) exact; (+) ENO-1; (×) ENO-2; (○) ENO-3.

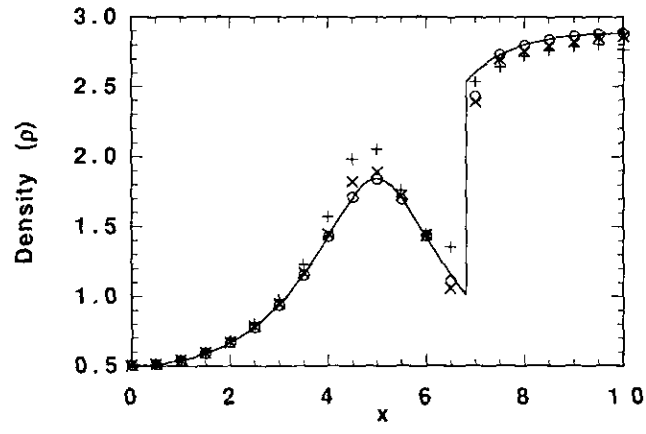

FIG. 15. Steady-state density profile in a converging–diverging nozzle: (———) exact; (+) ENO-1; (×) ENO-2; (○) ENO-3.

high-order schemes become apparent. The resulting solutions with 21 grid points are shown in Fig. 13. The first-order scheme predicts a shock several grid points closer to the inlet than its true location. The second- and third-order schemes give comparable results and accurately identify the shock location. Furthermore, the flux profiles (Fig. 14) compuated using the first-order scheme contain significant error in the subsonic region, while the high-order schemes exhibit only small deviations. Again, as expected there is a small improvement from second to third order.

The scheme was also used to resolve a shock in a converging–diverging nozzle. The shape function shown below was taken from [20]. The inlet and outlet conditions given in the third column of Table II produce a steady-state solution with a shock in the diverging section of the nozzle,

$$k = 1.75 - 0.75 \cos\left(\frac{x}{5} - \pi\right). \tag{72}$$

To achieve a supersonic velocity beyond the throat of the nozzle, the initial conditions for ρk, mk, and ek were set equal to the inlet conditions in the first half of the nozzle. Linear interpolation between these values and the steady-state exit values was employed in the diverging section. Because the flow remains supersonic in the converging section, the density reaches a local maximum and the velocity reaches a local minimum at the throat ($x = 5.0$). With 21 grid points, the first-order scheme produces large errors in density and velocity near these extrema as well as at the first point before the shock (see Figs. 15 and 16). The second-order scheme exhibits improved accuracy at these points although increased error appears at the first point after the shock. The improvement in resolution between the second- and third-order schemes is less pronounced

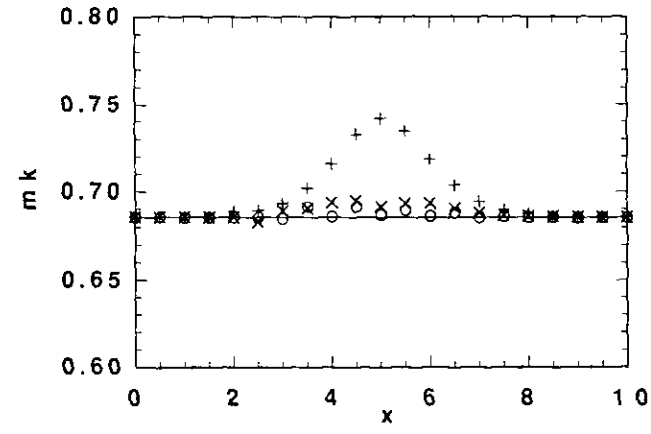

FIG. 14. Steady-state flux and nozzle area product (mk) in a diverging nozzle with inlet and outlet conditions as specified in the second column of Table II: (———) exact; (+) ENO-1; (×) ENO-2; (○) ENO-3.

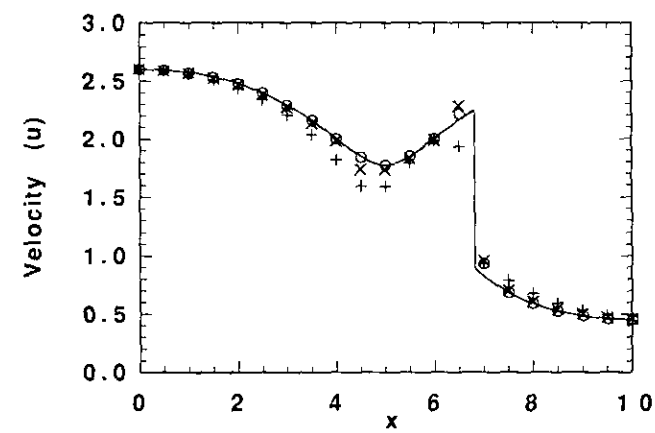

FIG. 16. Steady-state velocity profile in a converging–diverging nozzle: (———) exact; (+) ENO-1; (×) ENO-2; (○) ENO-3.

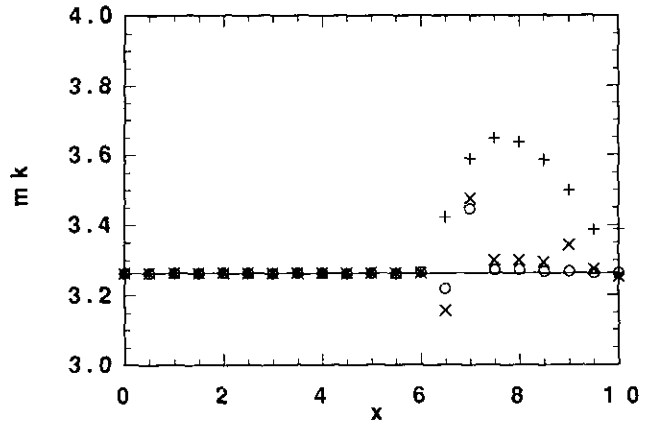

FIG. 17. Steady-state flux and nozzle area product (mk) in a converging–diverging nozzle: (———) exact; (+) ENO-1; (×) ENO-2; (○) ENO-3.

than the differences between the first- and second-order results. However, there is noticeable improvement near the throat and around the shock. The third-order results nearly match the exact solution everywhere except at the first point after the shock. The computed profiles of the product of the flux and nozzle area are shown in Fig. 17. Again large errors occur for the first-order scheme in the subsonic region. The second-order scheme significantly decreases the errors and the third-order scheme effectively has errors only at the first points on either side of the shock.

The results shown in Figs. 15, 16, and 17 demonstrate the qualitative features of the dependence of the solution profiles on the spatial order of approximation. If fewer grid points or conditions which generate more severe extrema are employed, the deviations between second and third orders become larger. In Fig. 18, the second- and third-order schemes with 15 grid points are compared. The third-order scheme provides an accurate solution while the second-order results deviate noticeably at the throat and throughout the subsonic region. Furthermore, the third-order scheme is capable of good resolution of the density profile even when only 11 grid points are used (see Fig. 18). The first-order scheme did not converge when either 15 or 11 grid points were used, and the second-order approximation did not converge with 11 grid points. This is because subsonic velocities occurred at the throat. If the velocity at the throat becomes subsonic, a shock in the diverging section cannot develop. Thus, the results in Fig. 18 demonstrate that the improved resolution which occurs with higher order approximations can remove errors which would otherwise prevent convergence.

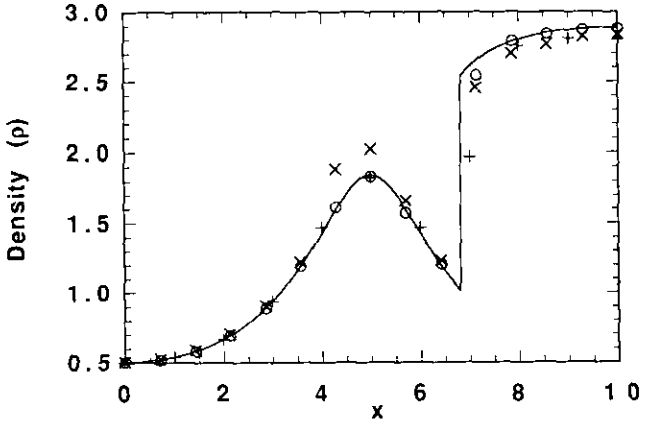

FIG. 18. Steady-state density profile in a converging–diverging nozzle: (———) exact; (+) ENO-3 with $N = 11$; (×) ENO-2 with $N = 15$; (○) ENO-3 with $N = 15$.

TABLE III

CPU Time Requirements

Scheme	Max CFL number	Time steps	CPU time	CPU time per time step
Explicit 1st order	0.8	868	4.7	1.0
Explicit 2nd order	0.8	635	4.5	1.3
Explicit 3rd order	0.8	597	5.2	1.6
Implicit 1st order	1000	14	1.0	13
Implicit 2nd order	1000	14	1.6	20
Implicit 3rd order	1000	14	2.1	27

For steady-state calculations, some simple modifications of the implicit scheme can be incorporated to decrease computing time. First, θ can be set to 1 so that (40) imposes no time step restriction. Furthermore, in the spirit of [17, 6, 18] a linearized version of (68) can be solved. This is implemented by limiting the number of Newton iterations to one per time step. The method remains first order in time, but the ENO properties can no longer be guaranteed at each time step. However, since steady state requires u^{n+1} and u^n to be nearly identical, the approximation will be ENO at steady state. Finally, satisfaction of the following criterion was required at steady state:

$$\left|\frac{u_j^{n+1} - u_j^n}{u_j^n \, \Delta t}\right| \leq \delta \quad \forall j. \tag{73}$$

The parameter δ was specified as 10^{-6} and 10^{-9} for the explicit and implicit schemes, respectively.

For these examples, both the non-linear (requiring convergence of the iterations) and the linearized implementations were employed. Both methods needed approximately the same number of time steps to reach steady state, and the non-linear implicit scheme typically required four or fewer iterations per time step. The difference in iterations per time step resulted in the linearized scheme, requiring significantly less computing time. Thus, the linearized version is more suitable for steady-state calculations.

In Table III, the normalized computing time requirements

for the linearized implicit and the explicit schemes are shown. For the implicit and explicit schemes, the maximum CFL numbers were fixed at 1000 and 0.8, respectively. For the hyperbolic tangent nozzle shape, the higher order explicit schemes required some modifications to reach steady state. Small changes in the smooth, almost flat, regions of the solution profile can change the stencil from one time step to the next. This change in stencil affects the approximation and slightly changes the calculated gradients. In turn, this leads to continuous stencil changes and convergence to steady state is hampered. To alleviate this problem, the stencil is biased slightly so that small fluctuations in the solution profile do not change the stencil [19]. This allows steady state to be reached. However, the number of time steps required to reach steady state is dependent on the choice of biasing. Since the explicit scheme reached steady state without any biasing for the converging–diverging nozzle, this example was used to compare the implicit and explicit computing time requirements.

For all orders of spatial approximation, the implicit scheme decreased computing requirements by at least a factor of 2.5. Furthermore, for this example, the third-order implicit scheme is more than twice as fast as the spatially first-order explicit scheme. While a two or threefold decrease in computing time is not dramatic, it is important since for this problem only 21 grid points were used. For problems with more significant curvature away from the shock, more grid points may be required. As the number of grid points increases, the CFL restriction for the explicit scheme becomes more severe and the computation time reduction provided by the implicit scheme should increase because for $\theta = 1$; there is no CFL restriction.

5. CONCLUSION

In this paper, a novel implicit implementation of the ENO–Roe scheme has been presented. Sufficient conditions to ensure that a scheme which is ENO (TVD) in its forward Euler implementation will remain ENO (TVD) in its implicit implementation were derived. The implicit ENO scheme was shown to be capable of producing comparable front resolution and decreasing computing requirements for transient solutions containing fronts moving slowly compared to the speed of propagation in the rest of the domain. Finally, a linearized version of the implicit ENO scheme was shown to reduce computing time requirements for steady-state calculations as compared to the explicit implementation.

ACKNOWLEDGMENTS

The authors thank Emad Fatemi and Stanley Osher for several helpful discussions. The financial support of NASA under Grant NAG2-822 and NSF under PYI Award No. CBT 88 57867 is gratefully acknowledged.

REFERENCES

1. H. Rhee, R. Aris, and N. Amundson, *First-Order Partial Differential Equations Volume I* (Prentice–Hall, Englewood Cliffs, NJ, 1986).
2. H. Rhee, R. Aris, and N. Amundson, *First-Order Partial Differential Equations Volume II* (Prentice–Hall, Englewood Cliffs, NJ, 1989).
3. M. Blunt and B. Rubin, *J. Comput. Phys.* **102,** 194 (1992).
4. J. M. Burgers, *Flow Equations for Composite Gases* (Academic Press, New York, 1969).
5. M. H. Wilcoxson, Master's thesis, UCLA Dept. of Chem. Eng., 1991.
6. R. M. Beam and R. F. Warming, *J. Comput. Phys.* **22,** 87 (1976).
7. H. C. Yee, R. F. Warming, and A. Harten, *J. Comput. Phys.* **57,** 327 (1985).
8. W. A. Mulder and B. van Leer, *J. Comput. Phys.* **59,** 232 (1985).
9. G. R. Shubin, A. B. Stephens, and H. M. Glaz, *J. Comput. Phys.* **39,** 364 (1981).
10. A. Harten, NYU Report, New York University, New York, October 1982 (unpublished).
11. P. L. Roe, *J. Comput. Phys.* **43,** 357 (1981).
12. B. van Leer, "Lecture Notes in Physics," Vol. 141, p. 354, (Springer-Verlag, Berlin, 1982).
13. C. Shu and S. Osher, *J. Comput. Phys.* **77,** 439 (1988).
14. C. Shu and S. Osher, *J. Comput. Phys.* **83,** 32 (1989).
15. A. Harten and S. Osher, *SIAM J. Numer. Anal.* **24**(2), 279 (1987).
16. A. Harten, B. Enquist, S. Osher, and S. R. Chakravarthy, *J. Comput. Phys.* **71,** 231 (1987).
17. W. R. Briley and H. McDonald, *J. Comput. Phys.* **24**(4), 372 (1977).
18. R. D. Richtmyer and K. W. Morton, *Difference Methods for Initial-Value Problems,* 2nd ed. (Interscience, New York, 1967), p. 203.
19. E. Fatemi, J. Jerome, and S. Osher, *IEEE Trans. Comput. Aided Des. Int. Cir. and Sys.* **10**(2), 232.
20. R. Enander, *J. Comput. Phys.* **107,** 291 (1993).