EXPLICIT FINITE DIFFERENCE SCHEMES FOR THE ADVECTION EQUATION

Mike H. Zhou, Michael Mascagni and Alice Y. Qiao
Program in Scientific Computing, University of Southern Mississippi, Box 6539, Hattiesburg, MS-39406

Abstract. Conventional explicit finite difference schemes for the advection equations are subject to the time step restrictions dictated by the CFL condition. In many situations, time step sizes are not chosen to satisfy accuracy requirements but rather to satisfy the CFL condition. In this paper we present explicit algorithms which are stable far beyond the CFL restriction. Similar or even better accuracy can be achieved with a much larger time step size. The idea is matching the stencil and the real domain of dependence by characteristic analysis. Numerical tests are done for linear advection equations as well as the Burgers equation.

1. Introduction

For a finite difference scheme (FDS) to be useful, it must be convergent. That is, its solution must approximate the solution of the corresponding partial differential equation (PDE) and that this approximation must improves as the grid spacing, h (for space) and k (for time), tends to zero. By the Lax-Richtmyer equivalence theorem, a consistent FDS for a PDE is convergent if and only if it is stable. Consistency of a FDS is easy to achieve, therefore stability has become the focus of the analysis of a FDS.

Implicit schemes usually have better stability properties than explicit schemes. However, explicit schemes have their own advantages, among which are their efficiency and ease of parallelization, which is becoming more and more important.

For hyperbolic equations, conditionally stable explicit schemes have been widely used. The stability condition (Courant number $\lambda \leq 1$) imposes restrictions on the time step size k and thus reduces the efficiency of the scheme in cases where the velocity field is highly peaked. The greater the maximum of the velocity field, the smaller the time step size. When the the Courant condition restricts the time step size to be considerably smaller than what would be imposed by accuracy requirements, a considerable amount of CPU time may be wasted.

The Courant condition can be derived from the domain of dependence analysis[1]. The solution of a linear hyperbolic equation propagates with speed c along characteristics with a slope of $1/c$. In Fig.1.1 we draw three characteristics. At time $t0$, we have the initial values $u^{t0}(x)$. At a future time $t1$, the value u_X^{t0} propagates to position B. Thus u_B^{t1} depends on u_X^{t0}. On the other hand, a conventional explicit finite difference scheme would have u_B^{t1} depending on u_A^{t0}, u_B^{t0}, and u_C^{t0}. Note that to predict the value at B, u_B^{t1}, a stencil (like ABC) covering B is used. For a small time step, such as $(t1-t0)$, X is within the domain of the stencil ABC. A combination of u_A^{t0}, u_B^{t0}, and u_C^{t0} may be a good approximation for u_B^{t1} with the finite difference scheme converging. However, if we want to enlarge the time step size to,

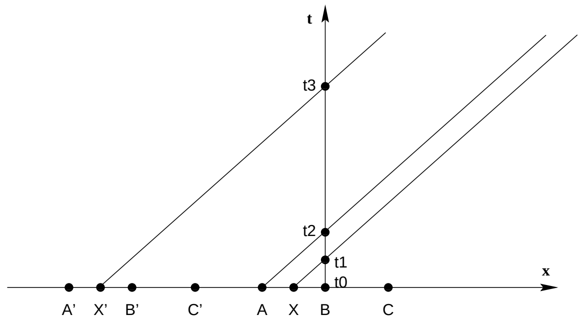

FIGURE 1.1.

say $(t3 - t0)$, the true solution u_B^{t3} will depend on $u_{X'}^{t0}$,while the finite difference scheme still has u_B^{t3} depending on u_A^{t0}, u_B^{t0}, and u_C^{t0}. A combination of u_A^{t0}, u_B^{t0}, and u_C^{t0} can not be a good approximation for u_B^{t3}, and the finite difference scheme will fail to converge. For convergence, a necessary condition is that X must be within the domain ABC,

$$\frac{t - t0}{AB} \equiv \frac{k}{h} \leq \frac{1}{c}. \tag{1.1}$$

This leads to the Courant-Friedrichs-Lewy (CFL) condition,

$$\lambda \equiv \frac{kc}{h} \leq 1. \tag{1.2}$$

A similar argument can be used to obtain the following theorem[2]:

> There are no explicit, unconditionally stable, consistent finite difference schemes for hyperbolic systems of partial differential equations.

However, in the derivation of CFL condition as well as the above theorem, the following assumption was used:

> For the prediction of the value at position B, the chosen stencil contains B, or at least the relative position of the stencil and B is fixed.

In fact, what's really important is not for the stencil to cover B, but for it to cover the B's real domain of dependence. If we discard the above assumption, the theorem will no longer hold and we may try to construct finite difference schemes that break the CFL restriction yet are still convergent. In Fig.1.1, if we choose the stencil $A'B'C'$, instead of the stencil ABC, the scheme may converge with a time step $(t3 - t0)$ which is beyond what the CFL condition would allow.

In the methods proposed in this paper, we still use a finite size stencil. However, the relative position of the stencil with respect to grid B is determined by domain of dependence. For a given time step size, we try to choose a stencil covering the domain of dependence of grid B. The scheme's domain of dependence and the equation's domain of dependence always coincide with each other for the linear advection equation and the scheme are thus unconditionally stable. We therefore call these schemes unconditionally stable explicit finite difference schemes (USE-FDS). Numerical tests are done for linear advection equations as well as Burgers equation.

2. First Order USE Method

2.1. **The Derivation of USE-FDS.** The objective is to derive an FDS for the advection equation:

$$u_t + c\,u_x = 0. \tag{2.1}$$

Finite difference schemes are constructed by replacing continuous derivatives with appropriate finite difference approximations defined on a discrete set of points. In this way consistency is automatically achieved while stability is not guaranteed. Since the exact solution of the linear advection equation is well known, we will derive the USE-FDS directly from this solution. This makes sense in that our ultimate goal is not to solve the linear advection equation, but to apply the scheme to more complex equations.

The exact solution to equation (2.1) is $F(x - ct)$. The shape of F is determined by the initial conditions and will translate with speed c without a change in shape. This situation is illustrated in Figure 2.1. Given the values of u at all grid points at

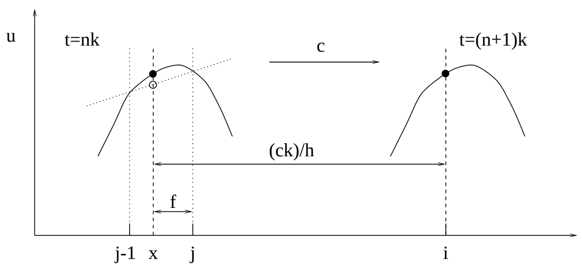

Figure 2.1.

time nk, we want to compute the the values of u at all grid points at time $(n+1)k$. From Fig.2.1 it is obvious that u_i^{n+1} is equal to u_X^n exactly. The distance between X and i, $\frac{kc}{h}$, is the Courant number. In general, X falls between two grid points $(j-1)$ and j. If we approximate the value of u_X^n by interpolating u_{j-1}^n and u_j^n (shown in Fig.2.1), we get the first order USE-FDS:

$$\lambda = \frac{kc}{h},\ j = i - \mathrm{trunc}(\lambda),\ f = \lambda - \mathrm{trunc}(\lambda), \tag{2.2}$$

$$\begin{aligned} u_i^{n+1} &= u_X^n \\ &\approx u_{j-1}^n + (1-f)(u_j^n - u_{j-1}^n) \\ &= f u_{j-1}^n + (1-f) u_j^n. \end{aligned} \tag{2.3}$$

The "trunc" function returns the integer part of λ. In the computer language `C`, this can be implemented simply by a casting λ to an integer. Above f is the fractional part of λ.

If $|\lambda| < 1$, then $f = \lambda$ and the first order USE-FDS becomes equivalent to the explicit Euler/Upwind scheme:

$$\frac{u_i^{n+1} - u_i^n}{k} + c\frac{u_i^n - u_{i-1}^n}{h} = 0,\ \text{for } c \geq 0, \tag{2.4}$$

$$u_i^{n+1} - u_i^n \over k + c \frac{u_{i+1}^n - u_i^n}{h} = 0, \text{ for } c \leq 0. \tag{2.5}$$

2.2. **Explicitness of USE-FDS.** One of the basic differences between explicit and implicit schemes is described in [3]:

> In an explicit method we have a formula for u_i^{n+1} in terms of known values of u_i at previous time levels, whereas with an implicit method we must solve a system of equations to advance to the next time level.

It is obvious that USE-FDS is not implicit, but is clearly explicit.

For an implicit method, after the system of equations are solved, u_i^{n+1} can be expressed in terms of the known values of u_i at the previous time level. Therefore a more rigorous definition of explicit method is defined in [1] stating that an explicit scheme is any scheme that can be written in the form:

$$u_i^{n+1} = \text{ a } finite \text{ sum of } u_{i'}^{n'} \text{ with } n' \leq n.$$

USE-FDS meets this criterion too with the coefficients in the above sum being explicitly defined through interpolation formulae.

2.3. **Consistency of USE-FDS.** Expanding the left hand side (LHS) of Eq.(2.3) around u_j^n, we have

$$\begin{aligned} \text{LHS} &= u_i^{n+1} \\ &= u_i^n + k(u_t)_i^n + O(k^2) \\ &= u_j^n + (\lambda - f)h(u_x)_j^n + k(u_t)_i^n + O(k^2, h^2). \end{aligned} \tag{2.6}$$

Expanding the right hand side (RHS) of Eq.(2.3) about u_j^n, we get

$$\begin{aligned} \text{RHS} &= f\, u_j^n - fh\,(u_x)_j^n + (1-f)u_j^n + O(h^2) \\ &= u_j^n - fh\,(u_x)_j^n + O(h^2). \end{aligned} \tag{2.7}$$

Therefore,

$$\frac{\text{LHS} - \text{RHS}}{k} = (u_t + c\, u_x) + O(k, h). \tag{2.8}$$

Clearly, the USE-FDS scheme, Eq.(2.3), is consistent with the advection equation, Eq.(2.1), and has an accuracy of $O(k, h)$.

2.4. **Stability Check.** Necessary and sufficient conditions for the stability of finite difference schemes for linear equations can be obtained by von Neumann (or Fourier) analysis[4]. This is done by seeking the time evolution of numerical solutions of the form:

$$u_{\omega,j}^n = v_\omega^n e^{i\omega x_j}. \tag{2.9}$$

Inserting the upper expression into USE-FDS, Eq.(2.3), we get a formula for the amplification factor:

$$z(\omega, f) \equiv \frac{v_\omega^{n+1}}{v_\omega^n} = f e^{i\omega h(j-i-1)} + (f-1)e^{i\omega h(j-i)}. \tag{2.10}$$

The magnitude of the amplification factor $z(\omega, f)$ is:

$$|z(\omega, f)| = \sqrt{1 + 2f(f-1)(1 + \cos((2j - 2i - 1)\omega h))}. \tag{2.11}$$

By taking the first and second derivatives of equation (2.11), we find that $|z(\omega, f)|$ reaches its minimum, 0, at $f = 0.5$. In addition, $|z(\omega, f)|$ reaches its maximum, 1, at the two end points $f = 0$ and $f = 1$. Since $0 \le f \le 1$, we conclude that $0 \le |z(\omega, f)| \le 1$ for all values of the Courant number and all frequencies. Obviously, this first order USE-FDS is unconditionally stable.

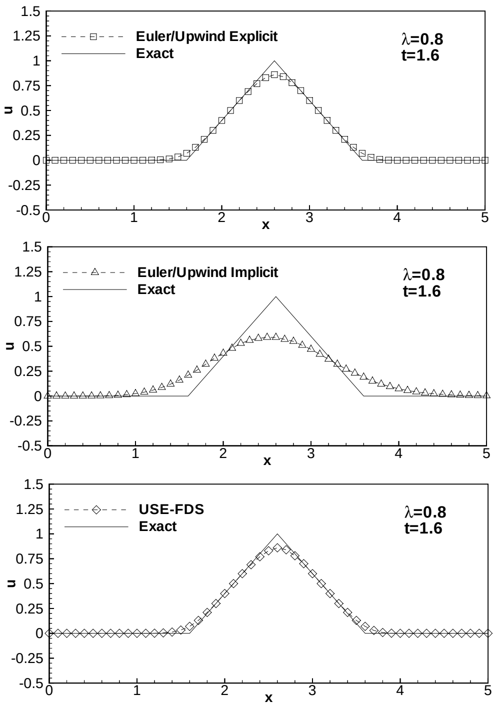

FIGURE 2.2.

2.5. **Numerical Experiments with Linear Advection Equation.** We now solve the advection equation, Eq.(2.1), with $c = 1$ and initial data,

$$u_0(x) = \begin{cases} x & \text{if } 0 \le x \le 1, \\ 2 - x & \text{if } 1 \le x \le 2, \\ 0 & \text{if } 2 \le x \le 5, \end{cases} \tag{2.12}$$

with periodic boundary condition. That is, we simulate the movement of a triangular wave around a one dimensional ring.

We use three different two-point stencil schemes, namely USE-FDS, explicit Euler/Upwind, and implicit Euler/Upwind which is defined by

$$\frac{u_i^{n+1} - u_i^n}{k} + c\frac{u_i^{n+1} - u_{i-1}^{n+1}}{h} = 0, \text{ for } c \ge 0. \tag{2.13}$$

In Fig.2.2 and Fig.2.3 we adopt a grid spacing of $h = 0.1$. In Fig.2.2 we choose a Courant number of $\lambda = 0.8$ and advance the wave packet to time 1.6. We can see that since $\lambda < 1$ all three schemes give stable, non-oscillatory, solutions. The implicit Euler/Upwind is the least accurate and computationally least efficient.

In Fig.2.3 we choose a Courant number of $\lambda = 1.6$ and advance the wave packet to time 0.8. We can see that since $\lambda > 1$ the explicit Euler/Upwind produces an unstable, oscillatory, solution. In the domain, the high frequency components of

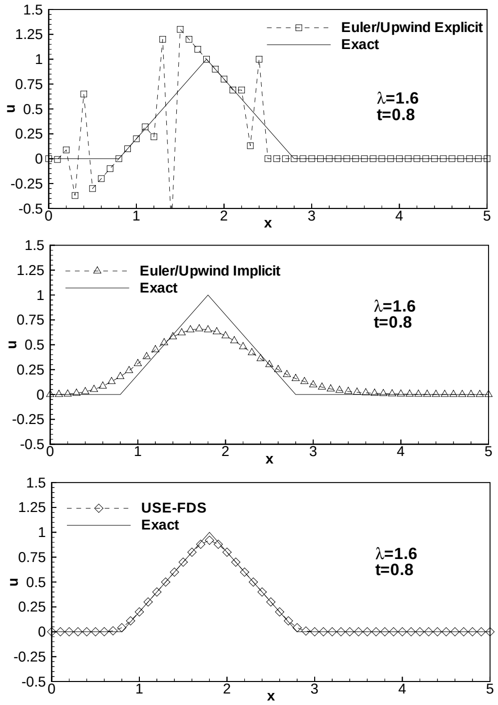

FIGURE 2.3.

the wave around the three derivative discontinuities are amplified and advanced with a wave speed less than c.

The results of the implicit Euler/Upwind scheme remain stable yet are less accurate. On the other hand, the results of USE-FDS remain both stable and accurate.

3. Higher Order USE-FDS

3.1. **The Derivation of Second Order USE-FDS.** Returning to Fig.2.1, we derived the first order USE formula by approximating the value of u_X^n by interpolating u_{j-1}^n and u_j^n. If we use a higher order interpolation, which will utilize the values at more stencil points, we will obtain higher order USE-FDS formulas.

As an example, second order USE-FDS is obtained by interpolating the three points u_{j-1}^n, u_j^n and u_{j+1}^n. The point marked j is the grid point closest to X, that is

$$j = i - \mathrm{trunc}(\lambda + 0.5), \; f = (\lambda + 0.5) - \mathrm{trunc}(\lambda + 0.5). \tag{3.1}$$

Using Newton's interpolation formula, we get

(3.2)
$$\begin{aligned} u_i^{n+1} &= u_X^n \\ &\approx u_{j-1}^n + (1.5 - f)(u_j^n - u_{j-1}^n) + 0.5\,(1.5 - f)(0.5 - f)(u_{j-1}^n - 2\,u_j^n + u_{j+1}^n). \end{aligned}$$

This formula turns out to be second order accurate in time and space if we do a Taylor series expansion. Also, it is unconditionally stable.

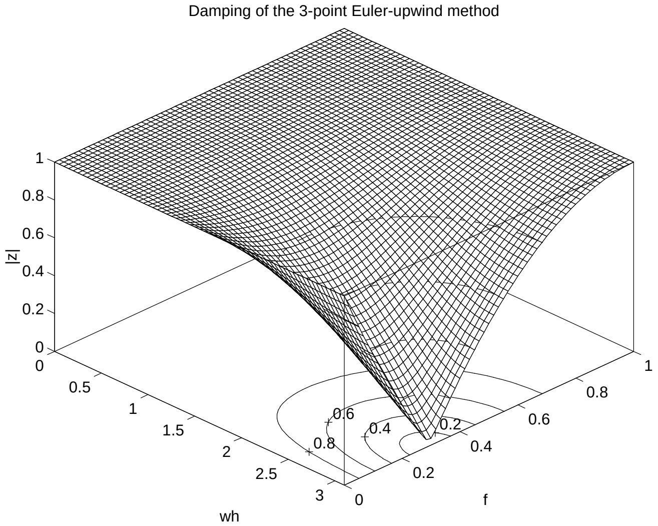

FIGURE 3.1.

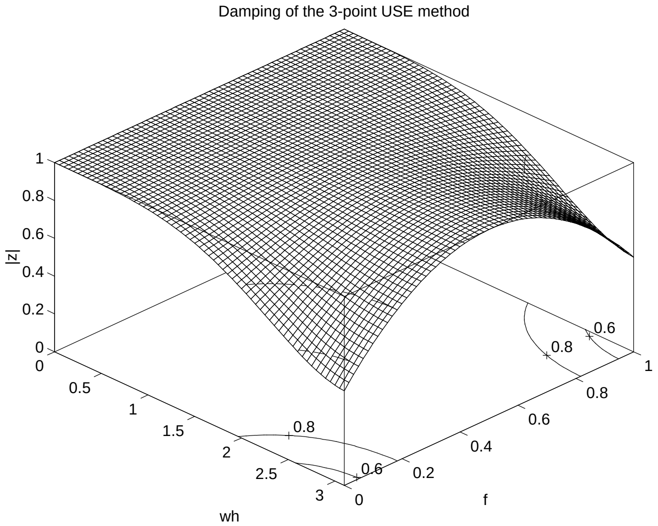

FIGURE 3.2.

Higher order formulas can be readily obtained by employing more stencil points with higher order interpolation. Note that USE-FDS is stable in the full range of the Courant number. In the stability region ($\lambda < 1$) of conventional schemes, USE-FDS is a special case of the General Average method [5] with one more upwind point than downwind point. This kind of Upwind-Biased spatial scheme has been combined with Runge-Kutta time integration to obtain good performance[6]. From the point of view of USE, we see why the Upwind-Biased scheme is a good choice. The fully Upwind method is equivalent to using values at points $j-2$, $j-1$, and j to approximate a value at a point between $j-1$ and j. USE-FDS uses the values

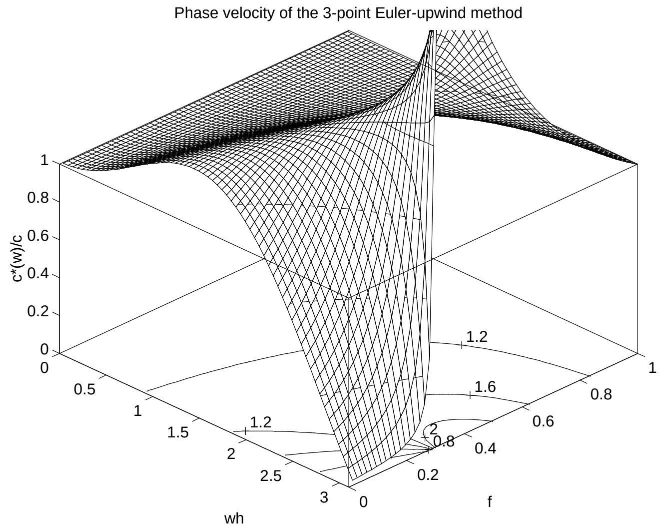

FIGURE 3.3.

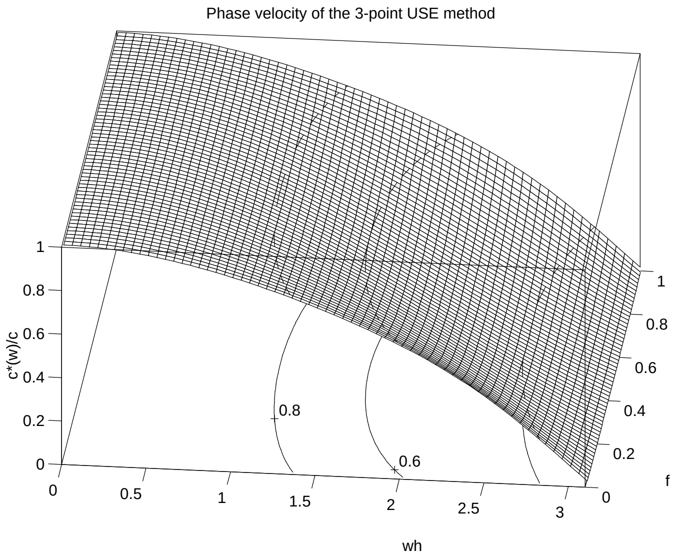

FIGURE 3.4.

at the three points to approximate a value at a point within $(j-1) \pm 0.5$. It is obvious that the latter is better.

3.2. **Amplification factor and phase speed.** The amplification factor for the second order USE-FDS, Eq.(3.1) and Eq.(3.2), is:

(3.3)

$$\begin{aligned} z(\omega, f) \quad &= \quad e^{i\omega h(j-i-1)} + (1.5-f)(e^{i\omega h(j-i)} - e^{i\omega h(j-i-1)}) \\ &\quad +0.5\,(1.5-f)(0.5-f)(e^{i\omega h(j-i-1)} - 2e^{i\omega h(j-i)} + e^{i\omega h(j-i+1)}). \end{aligned}$$

The amplification factor for the second order fully upwind General Average method is:

$$
\begin{aligned}
z(\omega, f) &= e^{i\omega h(j-i-1)} + (2-f)(e^{i\omega h(j-i)} - e^{i\omega h(j-i-1)}) \\
&\quad +0.5\,(2-f)(1-f)(e^{i\omega h(j-i-1)} - 2e^{i\omega h(j-i)} + e^{i\omega h(j-i+1)}).
\end{aligned}
\tag{3.4}
$$

The magnitude of the amplification factor indicates the damping of frequency components of the numerical solutions. The phase speed, which manifests diffusion of the numerical solution, can also be obtained from the amplification factor,

$$
c^*(\omega, f) = \frac{-\angle z(\omega, f)}{\omega k}
\tag{3.5}
$$

where $\angle z(\omega, f)$ is the phase angle of the amplification factor $z(\omega, f)$.

In Fig.3.1 through Fig.3.4 we compare second order USE with second order fully Upwind General Average scheme within $\lambda < 1$. Remember that for $\lambda > 1$, USE-FDS has the same behavior as within $\lambda < 1$ while other explicit schemes will be unstable. Fig.3.1 and Fig.3.2 show that USE-FDS has less damping on average even with $\lambda < 1$.

4. Applying USE-FDS to Variable Coefficient Advection Equation

In this section we test our scheme on the variable coefficient advection equation,

$$
u_t + c(x)\, u_x = 0.
\tag{4.1}
$$

This kind of equation arises in many applications such as the one-dimensional nozzle problem. Consider flow in a pipe with variable cross-sectional area $a(x)$, where x is distance along the length of the pipe. With an incompressible fluid and the assumption that the velocity is constant across each cross section and varies only with x, then we have

$$
c(x) = c_0 / a(x)
\tag{4.2}
$$

for some constant u_0. Now let $u(x, t)$ be the mass per unit length of pipe, then (4.1) holds.

For the test problem, we set

$$
c(x) = \frac{3}{\cos(\pi x) + 2}.
\tag{4.3}
$$

This represents a nozzle with a higher speed in the middle, $x \sim 1$, and lower speed near the two ends $x = 0, 1$. At the left end we feed the nozzle with an oscillating mass,

$$
u_0^t = \sin(5\pi t) + 1.
\tag{4.4}
$$

In Fig.4.1 we show the results of the USE-FDS and Euler Upwind Explicit/Implicit schemes. We can see that USE-FDS gives stable and less dissipative solutions for Courant number $\lambda = 2$ and $\lambda = 3$. Larger Courant number means a smaller number of time steps which produces less dissipation: there are fewer smoothing interpolations. However, from Fig.4.1 we see that this is not true for the implicit Euler Upwind scheme, where larger Courant numbers lead to greater dissipation.

In implementing USE-FDS, we iterate Eq.(2.2) twice. The first time we use $c = c_i$ and then second time $c = (c_i + c_j)/2$.

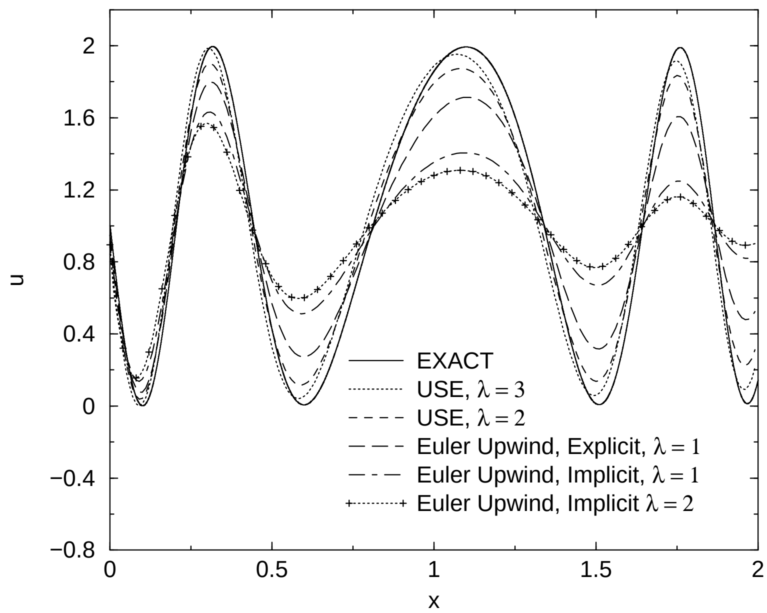

FIGURE 4.1.

5. Applying USE-FDS to Burgers Equation

Burgers equation

$$u_t + u\, u_x = 0, \tag{5.1}$$

is a convection equation with a non-linear, spatially variable wave speed. Thus it is not easy to find a stable numerical scheme for it, even within the usual CFL condition (typical $\lambda \leq 1$). However, in some cases where u is large (high global speed), schemes like USE-FDS can be used to obtain efficient, stable solutions by employing time steps larger than permitted by the CFL condition.

For the Burgers equation, the Lax-Friedrichs scheme [7] is quite effective,

$$u_i^{n+1} = \frac{u_{i-1}^n + u_{i+1}^n}{2} + k\frac{u_{i-1}^n + u_{i+1}^n}{2}\frac{u_{i+1}^n - u_{i-1}^n}{2h}. \tag{5.2}$$

In some sense, the above terms $\frac{u_{i-1}^n+u_{i+1}^n}{2}$ can be viewed as a substitution of u_i^n. They add artificial dissipation to the scheme and maintain its stability. We will use a similar technique to the USE-FDS formula for Burgers equation. Again we iterate Eq.(2.2) twice with $c = \frac{u_{i-1}^n+u_{i+1}^n}{2}$ first, and using $c = \frac{u_{j-1}^n+u_{j+1}^n}{2}$ second. For the initial condition, we superimpose the triangular wave packet (2.12) on a uniform background speed 1. That is, add 1 to $u_0(x)$ in Eq.(2.12).

The results of the simulations are shown in Fig.5.1. With Courant number $\lambda = 2$, USE-FDS gives similar result as Lax-Friedrichs with $\lambda = 1$.

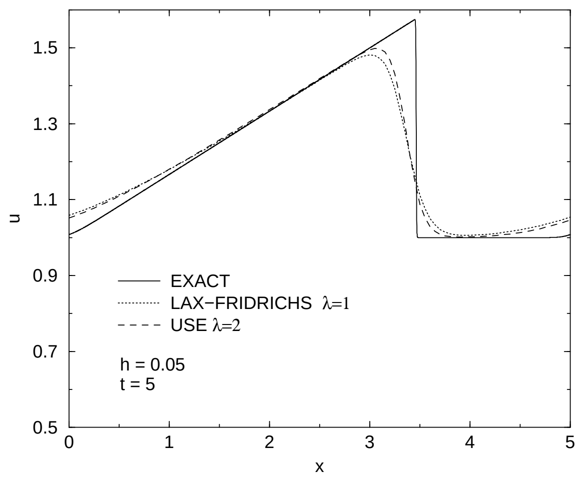

FIGURE 5.1.

6. CONCLUSION

In this paper we present explicit algorithms which are stable far beyond the CFL restriction. The idea is matching the stencil's and the real domain of dependence through use of characteristics. Numerical tests are done for the linear advection equations as well as for Burgers equation. Similar or even better accuracy is achieved with time steps much larger than permitted by the CFL condition. Of course the Courant number cannot be made arbitrarily large for non-linear equations. However, here the Courant number is limited by the maximum local variation in the velocity field, not the maximum of the velocity. In situations such as ocean modeling, one may have a very large maximum velocity and at the same time a relatively small maximum local variation of the velocity field. USE-FDS will be very efficient in this case by permitting a large Courant number to be used.

REFERENCES

[1] John C. Strikwerda, *Finite Difference Schemes and Partial Differential equations,* 28-29. Wadsworth & Brooks/Cole Advanced Books & Software, Pacific Grove, 1989.

[2] R. Courant, K. O. Friedrichs, and H. Lewy. *Uber die partiellen differenzengleichungen der mathematischen physik.* Mathematische Annalen. 32-74 (1928).

[3] Gene. H. Golub and James M. Ortega. *Scientific Computing and Differential equations: an introduction to numerical methods,* p262. Academic Press, Inc, Boston, 1992.

[4] Robert Vichnevetsky and John B. Bowles. *Fourier Analysis of Numerical Approximations of Hyperbolic Equations,* SIAM, Philadelphia, 1982.

[5] Peter A. Orlin and Louise Perkins, *A Frequency Accurate Spatial Derivative Finite Difference Approximation,* Num. Meth. PDE, **13** 549-560 (1997).

[6] Yuguo Li, *Wavenumber-Extended High-Order Upwind-Biased Finite-Difference Schemes for Convective Scalar Transport,* J. Comp. Phys., **133** 235-255 (1997).

[7] Richard Liska and Burton Wendroff, *Composite Schemes for Conservation Laws,* SIAM J. Num. Anal., to appear.