N92-22497

A Simple Method for Simulating Gasdynamic Systems

Tom T. Hartley
Department of Electrical Engineering
The University of Akron
Akron, OH 44325-3904
216-972-5991

December 1990

Introduction

This memo contains a brief discussion of a simple method for performing digital simulation of gasdynamic systems. Basically it is a modification of a method attributed to Courant, Isaacson, & Rees (1952), "On the Solution of Nonlinear Hyperbolic Differential Equations by Finite Differences," Communications on Pure and Applied Mathematics, vol V, pp 243-255. The approach is somewhat intuitive and requires some knowledge of the physics of the problem as well as an understanding of the effect of finite differences. The method is given in Appendix A which is taken from the book by P.J. Roache, "Computational Fluid Dynamics," Hermosa Publishers, 1982. The resulting method is relatively fast while it sacrifices some accuracy.

Spatial Differencing Revisited

The reader is reminded of the general problem associated with simulating nonlinear hyperbolic systems of the form

$$\frac{\partial u}{\partial t} + \frac{\partial F(u)}{\partial x} = S \ .$$

The problem is that the information is allowed to travel in both spatial directions in subsonic flow. It then becomes difficult to choose a spatial differencing operator. A central difference would be the obvious choice, however the resulting difference equation for u will become dominated by high frequency spurious noise or instability. A pure forward or backward difference, on the other hand, will only allow information to travel in one direction, again yielding numerical instability. The clever thing about the Courant, Isaacson, Rees approach is that the actual physics of the process is considered when doing the differencing. The terms in the F(u) vector associated with mass flow and energy are assumed to propagate signals downstream. Thus each of these terms are approximated using backward differences. Alternatively, the terms in the F(u) vector associated with pressures are assumed to propagate information in both directions. Thus each of these terms are approximated using central differences. The resulting method has the remarkable properties of stability, shock capturing, and reasonable accuracy. Both the time responses and steady state spatial distributions have first order accuracy. Furthermore, it also appears that some rather large spatial lumps are possible. A real benefit of this method is its simplicity and computational speed. As it does not usually require explicit artificial dissipation and is a one pass method, it should be approximately two times faster than MacCormack's method.

The method, as applied to quasi-one-dimensional gasdynamic systems, is as follows:

$$\dot{\rho}_i = -\frac{1}{HA_i}\left[A_i m_i - A_{i-1} m_{i-1}\right] + \frac{1}{A_i} M_i$$

$$\dot{m}_i = -\frac{1}{HA_i}\left[\frac{A_i m_i^2}{\rho_i} - \frac{A_{i-1} m_{i-1}^2}{\rho_{i-1}}\right] - \frac{1}{2HA_i}\left[P_{i+1}A_{i+1} - P_{i-1}A_{i-1}\right] + \frac{P_i}{A_i}\left[\frac{dA}{dx}\right]_i + \frac{1}{A_i} F_i$$

$$\dot{E}_i = -\frac{1}{HA_i}\left[\frac{A_i m_i E_i}{\rho_i} - \frac{A_{i-1} m_{i-1} E_{i-1}}{\rho_{i-1}}\right] - \frac{1}{2HA_i}\left[\frac{A_{i+1} m_{i+1} P_{i+1}}{\rho_{i+1}} - \frac{A_{i-1} m_{i-1} P_{i-1}}{\rho_{i-1}}\right] + \frac{1}{A_i} Q$$

The specific method used approximates the time derivatives with Euler's method.

For completeness, the simple first order method of Lax (see Appendix A) was also attempted but was dominated so much by diffusion that no shock capturing was apparent, while some very small spatial oscillations were. This method is not recommended for systems that contain shocks.

40-60 Inlet Validation

The NASA Lewis 40-60 Inlet was simulated in QuickBasic using this approach in order to determine its applicability. The program is given in Appendix B for reference and will be referred to as PHYSL for PHYSical Lumping. Forty-one lumps were used with a timestep of 20 μs, half of the usual. The steady state spatial distributions for several flow variables are given in in the figures. It should be noted that the shock is sitting a little farther back in the inlet with respect to the usual distribution from LAPIN and MACGAS which is given in the NASP paper. Also, the shock is a little more mushed out, but not too bad considering the simplicity of the method. A transient response was also obtained on what has become the standard test problem, that is, the downstream pressure input of +100 psf at t=0.002 seconds. The response has the same shape, however it is a little slower in responding and peaking. It is not clear whether this is "good enough" but would appear to be very promising as it still allows large perturbations. The LAPIN and MACGAS responses are also included for comparison.

Discussion

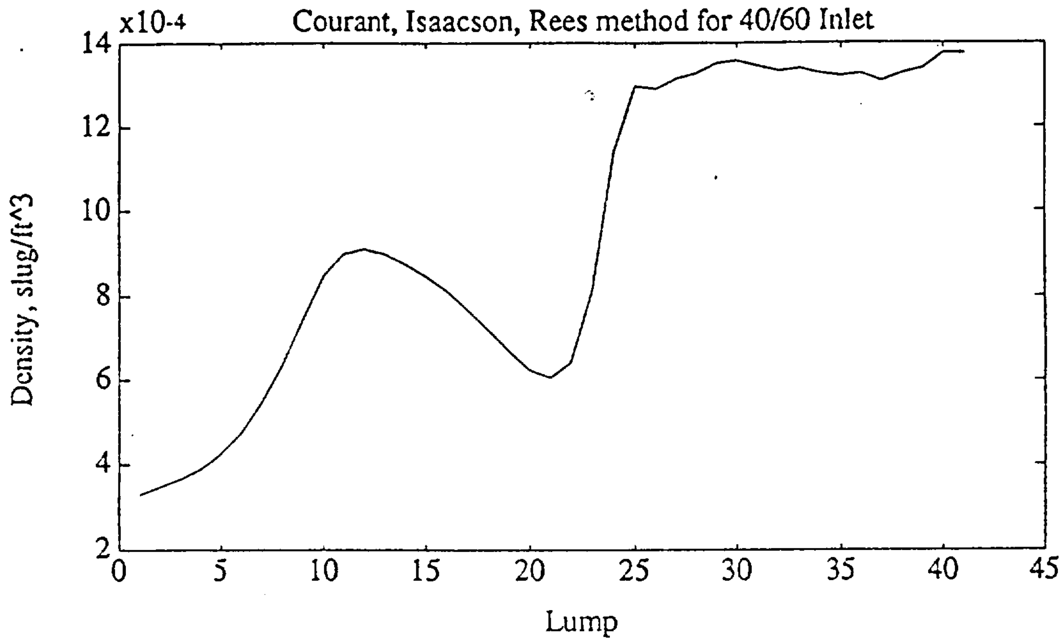
Some of the benefits of the PHYSL approach requiring further study are given below.

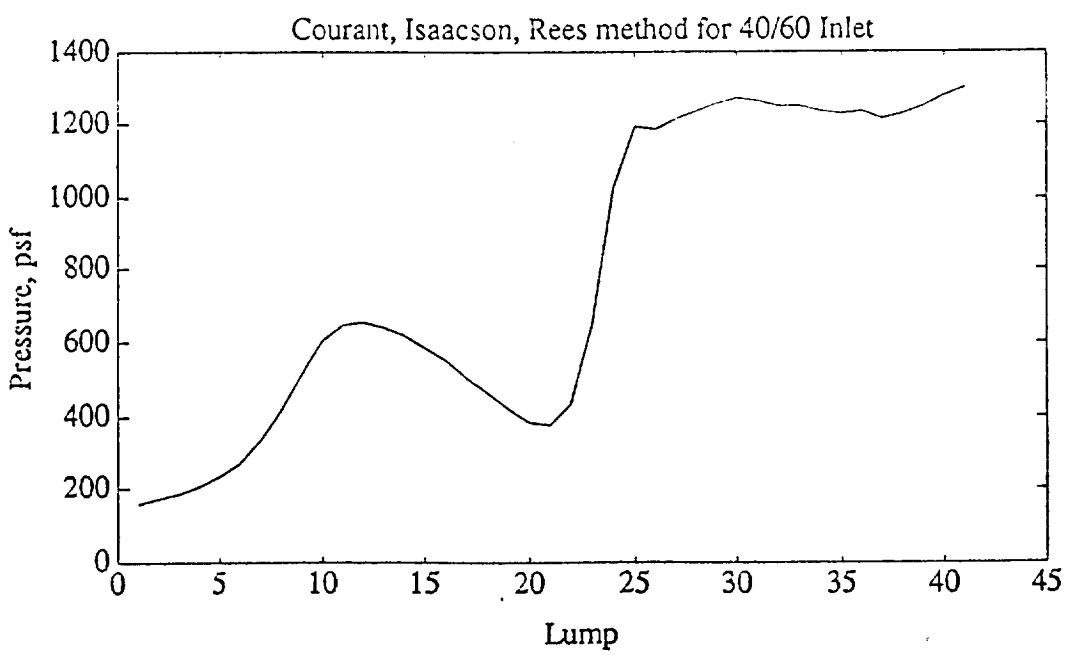
1) The method should be about two times faster than MacCormack's method. It should be noted that PHYSL appears to need a smaller timestep.
2) Larger lumps may be possible which would then allow even further speedup.
3) Large nonlinear models are easily written down, allowing their direct study for possible model reduction (as opposed to methods using Jacobian computations, prediction and correction, or artificial viscosity).
4) It also allows easy linearization of the discrete lumps for linear models and model reduction.
5) Alternate integration methods may be possible as opposed to Euler's method which the PHYSL method presently uses.
6) It may be possible to use different flow variables to allow even more efficient or natural spatial differencing.
7) It may be possible to develop a more useful buzz model using this method.

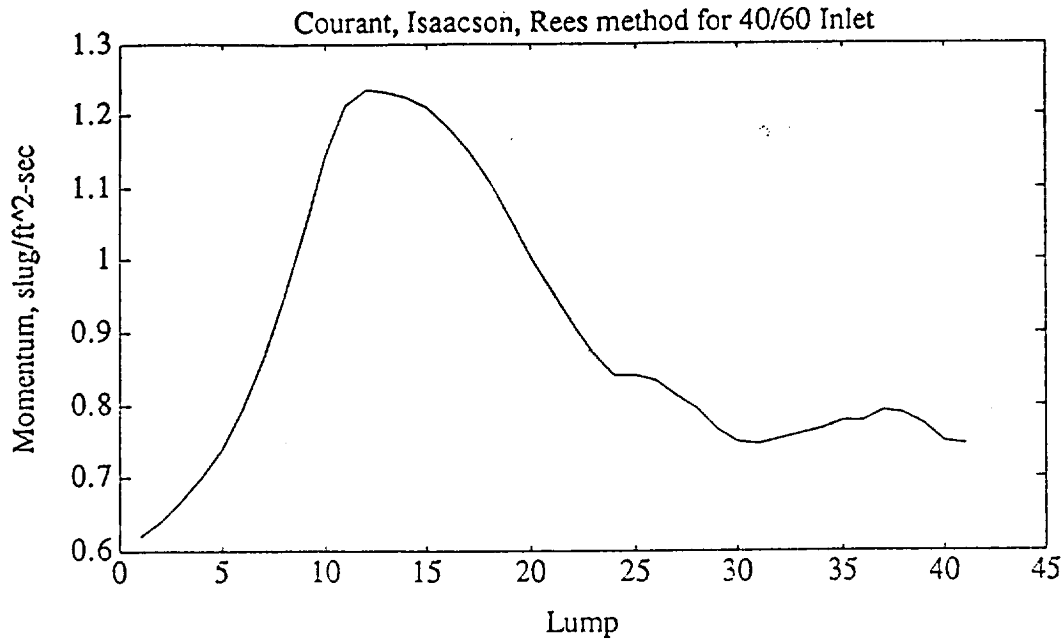
The major disadvantage of the method is basically its lack of accuracy. The methods used in LAPIN and MACGAS are second order accurate methods whereas this is a first order accurate method. It is not clear how bad the transient response can become using this method and still be meaningful.

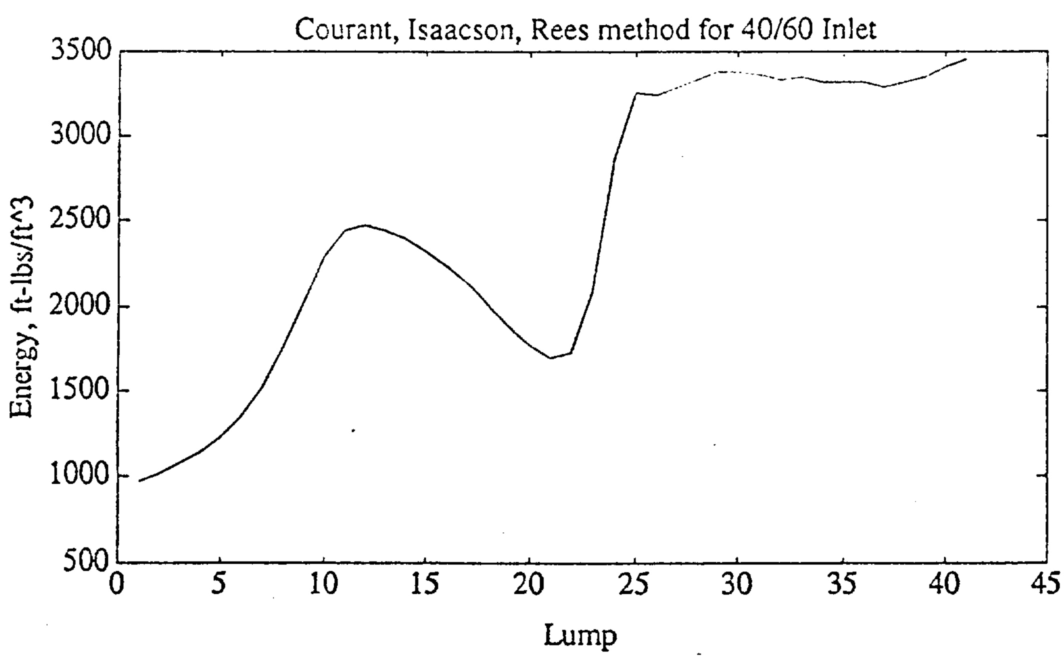
Acknowledgements

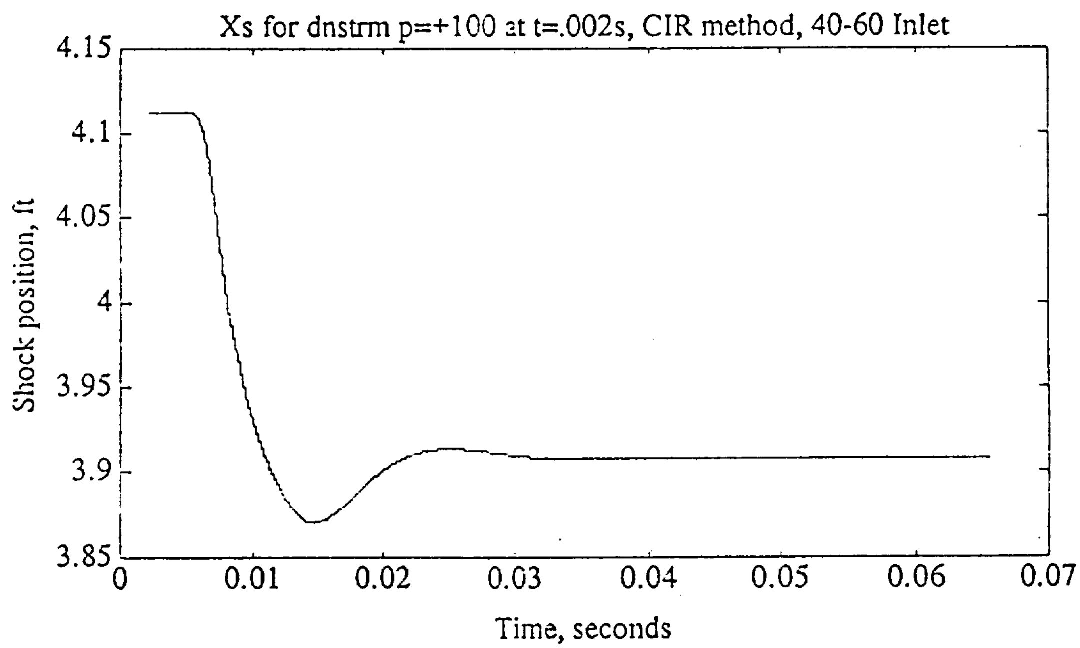
The author wishes to thank Steve and Amy Chicatelli for their reading of this document and their useful comments.


x10-4
Courant, Isaacson, Rees method for 40/60 Inlet
14
12
10
8
6
4
2
Density, slug/ft^3
0
5
10
15
20
25
30
35
40
45
Lump


Courant, Isaacson, Rees method for 40/60 Inlet
1400
1200
1000
800
600
400
200
0
Pressure, psf
0
5
10
15
20
25
30
35
40
45
Lump


Courant, Isaacson, Rees method for 40/60 Inlet
Momentum, slug/ft^2-sec
1.3
1.2
1.1
1
0.9
0.8
0.7
0.6
0
5
10
15
20
25
30
35
40
45
Lump


Courant, Isaacson, Rees method for 40/60 Inlet
Energy, ft-lbs/ft^3
3500
3000
2500
2000
1500
1000
500
0
5
10
15
20
25
30
35
40
45
Lump


Xs for dnstrm p=+100 at t=.002s, CIR method, 40-60 Inlet
4.15
4.1
4.05
4
3.95
3.9
3.85
Shock position, ft
0
0.01
0.02
0.03
0.04
0.05
0.06
0.07
Time, seconds

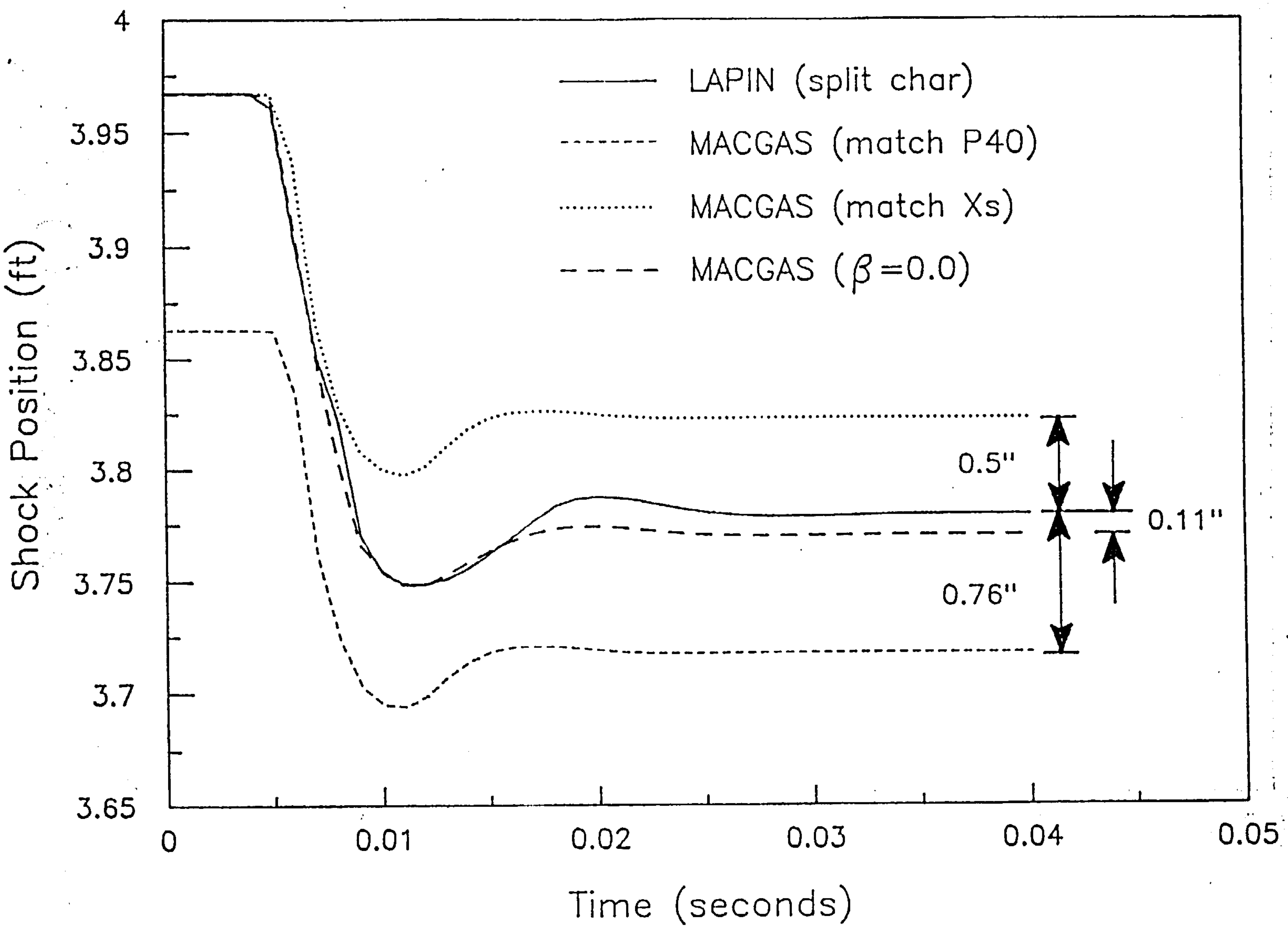
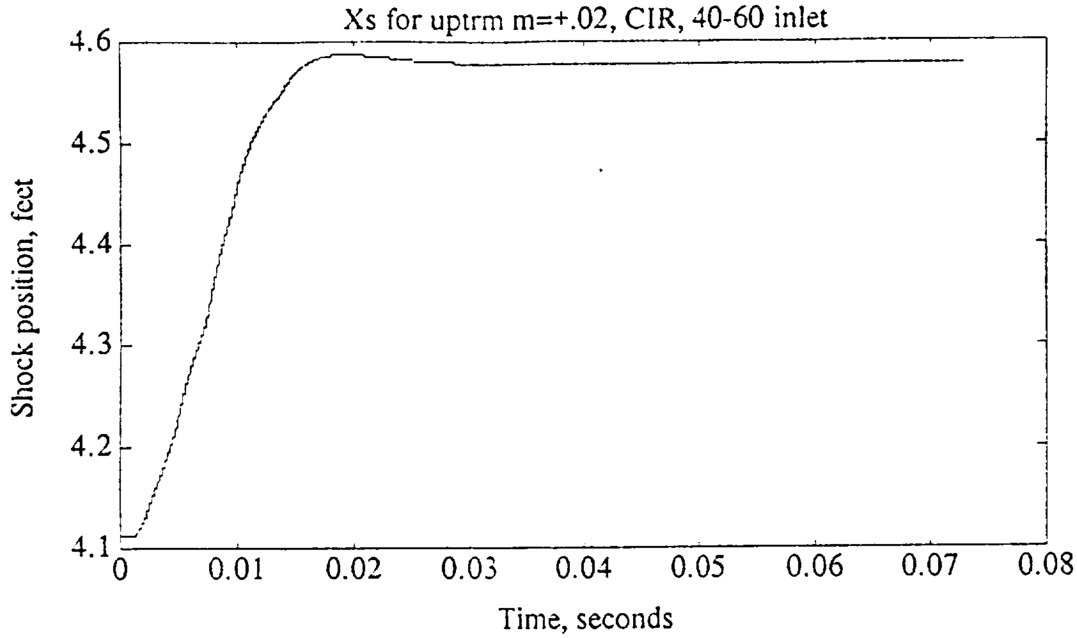


Figure 4 – Small Perturbation Transient Response
(+7.6% step in back pressure at 0.002 seconds)


Xs for uptrm m=+.02, CIR, 40-60 inlet
Shock position, feet
4.1
4.2
4.3
4.4
4.5
4.6
0
0.01
0.02
0.03
0.04
0.05
0.06
0.07
0.08
Time, seconds

Appendix A

V-D-4. Errors Arising from Artificial Viscosities

The use of artificial viscosities is often unavoidable, and it can be acceptable; but some strange errors can arise from explicit artificial viscosities, aside from the obvious ones common to incompressible flow calculations (section III-A-8). Schulz (1964) pointed out that simple application of the von Neumann-Richtmyer q_1 in cylindrical or spherical coordinates causes a diffusion of radial momentum. He extended q_1 to a tensor form which maintains strict conservation of radial momentum. Cameron (1966) showed that explicit artificial viscosities introduced surprising errors in the calculation of shocks propagating across a material interface or across a change in mesh spacing, Δx. The von Neumann-Richtmyer q_1-term causes spurious fluctuations for the changes in entropy and density as the shock crosses a material interface. Also, when Δx changes, a false shock wave is *reflected off the mesh change,* and the speed of the original propagating shock is altered. He also found that Landshoff's q_2 did not adversely affect shock speed at a mesh change, but was less useful than the von Neumann-Richtmyer q_1 because the shock *thickness* now changed abruptly at the mesh change. Cameron used both errors to partially cancel each other. By changing Δx at the material interface, he obtained the correct speed for the propagating shock. The false reflected shock still appeared, however. Higbie and Plooster (1968) varied the von Neumann-Richtmyer q_1 for a shock propagation problem in Lagrangian coordinates in such a way that the shock thickness in mesh increments stayed constant as the mesh spacing continually changed, thus eliminating oscillations.

V-E. Methods Using Implicit Artificial Damping

Instead of adding explicit artificial viscosity terms like q_1 to the equations, artificial damping may be added implicitly, just from the form of the difference equations. Sometimes these methods add an artificial *viscosity* in the sense of a non-zero coefficient of second space derivatives, and sometimes they just add artificial *damping* in the sense of the eigenvalues of the amplification matrix being less than one in magnitude. In either case, these methods may require *additional explicit* artificial viscosities in order to stabilize strong shock calculations.

V-E-1. Upwind Differencing

The second method of Courant, Isaacson, and Rees (1952) is a one-sided or upwind differencing scheme, as described in section III-A-8. It was also suggested by Lelevier (see Richtmyer, 1957) for Lagrangian equations, and is frequently referred to as Lelevier's method (e.g., Crocco, 1965; Roberts and Weiss, 1966; Kurzrock and Mates, 1966). In equation (4-63), each of the advected properties, U, that appear in F and G is differenced according to the sign of the advection velocity, u or v, respectively. However, the pressure gradients in the momentum equations must *not* be evaluated by upwind differences, as will be discussed in the next section. In terms of the 1D inviscid equations (4-66), the first upwind differencing method is as follows.

$$\frac{\rho_i^{n+1} - \rho_i^n}{\Delta t} = -\frac{(\rho u)_i^n - (\rho u)_{i-1}^n}{\Delta x} \quad \text{for} \quad u_i > 0 \tag{5-22A}$$

$$= -\frac{(\rho u)_{i+1}^n - (\rho u)_i^n}{\Delta x} \quad \text{for} \quad u_i < 0 \tag{5-22B}$$

$$\frac{(\rho u)_i^{n+1} - (\rho u)_i^n}{\Delta t} = -\frac{P_{i+1}^n - P_{i-1}^n}{2\Delta x} - \frac{(\rho u^2)_i^n - (\rho u^2)_{i-1}^n}{\Delta x} \quad \text{for} \quad u_i > 0 \tag{5-23A}$$

$$= -\frac{P_{i+1}^n - P_{i-1}^n}{2\Delta x} - \frac{(\rho u^2)_{i+1}^n - (\rho u^2)_i^n}{\Delta x} \quad \text{for} \quad u_i < 0 \tag{5-23B}$$

$$\frac{E_{s_i}^{n+1} - E_{s_i}^n}{\Delta t} = -\frac{[u(E_s + P)]_i^n - [u(E_s + P)]_{i-1}^n}{\Delta x} \quad \text{for} \quad u_i > 0 \tag{5-24A}$$

$$= -\frac{[u(E_s + P)]_{i+1}^n - [u(E_s + P)]_i^n}{\Delta x} \quad \text{for} \quad u_i < 0 \tag{5-24B}$$

The 2D difference equations follow this form in an obvious way. The analysis of Kurzrock (1966) indicates that stability is limited, in addition to the Courant-number restriction, by

$$\Delta t \leq \frac{|u|/\Delta x + |v|/\Delta y}{\left[|u|/\Delta x + |v|/\Delta y + a/\Delta x\sqrt{1 + \beta^2}\right]^2} \tag{5-25A}$$

or, for $\Delta x = \Delta y = \Delta$(or $\beta = 1$),

$$\Delta t \leq \frac{(|u| + |v|)\Delta}{(|u| + |v| + a\sqrt{2})^2} \tag{5-25B}$$

This limitation will become dominant in stagnation regions and in recirculating flow regions, where $u, v \to 0$. (See also section V-E-3.)

The modifications of this first upwind differencing method, which are necessary to achieve strict conservation near a region of velocity reversal, follow the description in section III-A-10. The more accurate second upwind differencing follows the description in section III-A-11.

These upwind differencing methods introduce effective "viscosity" through the truncation errors of the one-sided differences. The method adds artificial diffusion terms to $U = \rho, \rho u, \rho v, E_s$ in equation (4-63). From the analysis in section III-A-8, the x- and y-diffusion terms for the transient analysis are

$$\alpha_x = \frac{1}{2}u\Delta x(1 - u\Delta t/\Delta x)$$
$$\alpha_y = \frac{1}{2}v\Delta y(1 - v\Delta t/\Delta y) \tag{5-26A}$$

and, for the steady-state analysis,

$$\alpha_x = \frac{1}{2}u\Delta x$$
$$\alpha_y = \frac{1}{2}v\Delta y \tag{5-26B}$$

Note that the viscosity effect is not really equivalent to a physical viscosity, since the coefficients are directional and dependent on the velocity components.

Exercise: In a flow parallel to the x-axis with $\partial U/\partial x = 0$, but with an arbitrary density distribution in the y-direction, contrast the artificial diffusion behavior of the upwind differencing method with that of Rusanov's method.

For strong shocks appearing in inviscid calculations, this implicit viscosity is not usually sufficient to stabilize the calculations (Richtmyer, 1957), but Kurzrock and Mates

(1966), Scala and Gordon (1967), and Roache and Mueller (1970) have applied it to low (cell) Reynolds-number flows with success.* This method is also the basis of the PIC and FLIC codes, to be described shortly.

The upwind difference method possesses the transportive property (sections III-A-9, 10) which is significant for both subsonic and supersonic flow. The associated lack of second-order spatial accuracy is somewhat less significant in supersonic than in subsonic flow, as we now discuss.

V-E-2. The Domain of Influence and Truncation Error

In this section, we will compare and relate the domain of influence in continuum and in finite-difference equations. Our objective is to show how upwind differencing maintains something of the correct characteristic sense of the continuum equations and does not necessarily have a worse spatial truncation error than do centered difference methods.

Consider first the incompressible continuum flow equations,

$$\nabla^2\psi = \zeta \tag{5-27}$$

and

$$\frac{\partial\zeta}{\partial t} = -\nabla\cdot(\vec{V}\zeta) + \frac{1}{Re}\nabla^2\zeta \tag{5-28}$$

The vorticity transport equation (5-28) is parabolic and, by itself, represents an initial-value problem with limited spatial domain of influence in the inviscid limit $1/Re = 0$. But the Poisson equation (5-27) is elliptic and represents a boundary-value problem. Therefore, a disturbance in ζ at any point in the flow immediately affects all other points in the field, even with $1/Re = 0$, through the nonlinear term $\vec{V}$ which depends on ψ, and thus ζ, through equation (5-27). This property is shared by the finite-difference equations. We say that the system (5-27) and (5-28) possesses *infinite signal propagation speed,* and so does the finite-difference equation.

The inviscid compressible flow equations are all transport equations like (5-28) and therefore represent initial-value problems. The signal propagation speed is *finite*; for small linearized disturbances, the signal propagates at the isentropic sound speed (a) relative to the fluid, or at $(V + a)$ relative to an Eulerian mesh. Consequently, for $V > a$, i.e., $M > 1$, no disturbance is propagated upstream. This leads directly to the well-known Mach-cone principle, or the principle of limited upstream influence.

Consider now the signal propagation in a finite-difference equation. If space-centered differences are used, any disturbance at (i) at time (n) is felt at $(i\pm1, j\pm1)$ at $(n+1)$, no matter what the value of Δt. Thus, the propagation distances are always the same, Δx and Δy. The propagation speeds are then $\Delta x/\Delta t$ and $\Delta y/\Delta t$. The Courant-Friedrichs-Lewy (1928) or CFL necessary stability requirement is that the finite-difference domain of influence at least include the continuum domain of influence, i.e., $\Delta x/\Delta t < V + a$, or

$$C = \frac{(V + a)\Delta t}{\Delta x} \leq 1 \tag{5-29}$$

where C is the Courant number. In strong shock problems, where the small-disturbance assumption is not valid, replacement of "a" by the nonlinear shock propagation speed $a_s > a$ leads to the von Neumann-Richtmyer (1950) requirement.

Courant et al. (1928) did not require anything else from the finite-difference equations, since their objective was only to demonstrate the existence of solutions. But it clearly

*Scala and Gordon (1967) used upwind differencing for the advection terms, but with a more complex pattern of operations, as in Sheldon's method for the Poisson equation (see section III-B-7).

would be desirable also to maintain something of the limited upstream influence of the continuum system. Working with a rectangular mesh, the most we can accomplish is to restrict the sense, + or −, of perturbations along u and v. This led Courant, Isaacson, and Rees (1952) to their method for differenicng in a rectangular mesh, upwind differencing.

This leads again to the notion of transportive differencing for the advection terms, as discussed in sections III-A-8, 9, 10. But allowance must be made for the possible nonlinear upstream propagation in the case $a_s > V$. This leads to the space-centered differencing of the pressure gradient terms of the momentum equations, so that pressure gradient effects are felt upstream.* Note that P is *not* an advected quantity in $\partial P/\partial x$ and $\partial P/\partial y$, but *is* an advected quantity in the flow-work term, $\nabla \cdot (\vec{V}P)$, of the energy equation; consequently, upwind differencing *is* used on the flow-work term.

The distinction between the behavior of these equations and the incompressible system is that no elliptic equation like (5-27) appears, so the compressible inviscid system is purely hyperbolic.

The second-order accuracy of space-centered difference methods is still highly desirable, of course, as it was in incompressible flow. But in supersonic flow, we sacrifice less to achieve the transportive property. The accuracy evaluation of centered differences of section III-A-1 is based on Taylor series expansions for the flow properties, assuming continuity of the flow variables and their derivatives. But, in inviscid supersonic flow, the inviscid equations do not necessarily display continuity of derivatives. In fact, characteristic curves may be *defined* (Courant and Friedricks, 1948; Shapiro, 1953) as curves across which flow variables may have discontinuous derivatives.** Therefore, the Taylor-series expansion is not always valid, and the loss of truncation order of the differentials is not as important in supersonic flow.***

For viscous flow, the characteristics do not exist and the above arguments are weakened. It does seem reasonable, however, to base arguments on the differencing methods for the advection terms on only the behavior of the inviscid equations. This approach is conceptually vague, but the known success of method-of-characteristics solutions in computing real flows with small viscosity supports the approach.

Lax (1969) has shown that the upwind difference form gives a very good shock calculation in the inviscid form of Burger's equation, but fails for the full system of compressible flow inviscid equations and also, surprisingly, for the *linearized* inviscid Burger's equation. That is, the calculations of the nonlinear equation are *more* accurate than those of the linear equation.

V-E-3. PIC and FLIC

A well known method originally devised by Evans and Harlow (1957) is the Particle-in-Cell or PIC method. The genesis of this method is different from most, in that the attempt is not made to model the differential equations so much as the fundamental physical process, through a finite-particle approach. PIC may unequivocally be called a "simulation" method. The calculations proceed in several phases at each time level, with several key intermediate

---

*Kurzrock (1966) experimented with forward, backward, and centered pressure differences. His experiments and his stability calculations show that centered pressure differencing is preferable for his boundary-layer calculations.

Note the physical absurdity that would result from using upwind differencing for pressure and all advection terms. Then, in the quasi-1D duct flow problem described in section III-C-9, the effects of flow perturbations at outflow (i = I) could never be felt upstream, and a shock could not propagate upstream. It would therefore not be possible to computationally turn off an indraft supersonic wind tunnel!

**It is precisely this property that gives the method of characteristics its utility, allowing different flow regions to be patched together along characteristics.

***McNamara (1967) credits Trulio (1964) for showing that, for time-marching methods with discontinuous derivatives, the truncation error tends to zero no faster than $(\Delta x)^{3/2}$.

cell properties being calculated on the basis of pressure contributions, followed by advection calculations. The method is too complicated to describe in complete detail here, but the most unique aspect is that continuum flow is not modeled; rather, a finite number of particles is used, their locations and velocities being traced by Lagrangian kinematics as they move through a computational Eulerian mesh. They are not merely marker particles as in the MAC code (see section III-G-4), but they actually participate in the calculation, even when free surfaces and interfaces are not present. Cell-averaged thermodynamic properties are calculated, based on the numbers of particles in the cell. As few as six particles/cell on the average and three particles/cell locally have been used. The results display high frequency oscillations in cell density and pressure, as expected.

A continuum method which evolved out of the PIC code is the <u>F</u>luid <u>in</u> <u>C</u>ell or FLIC code of Gentry, Martin, and Daly (1966), based on earlier work by Rich (1963). They departed from the finite particle approach of PIC but retained most of the other aspects. It is a two-step method. In the first part of the first step, provisional values, $\overline{u^{n+1}}$ and $\overline{v^{n+1}}$, are calculated using only the contribution of the pressure gradients and the explicit artificial viscosity terms, if present. [A form like (5-10) is used for the explicit artificial viscosity.] Non-conservation forms are used. Then a provisional internal energy, $\overline{e^{n+1}}$, is calculated only from the pressure term of the equation

$$\frac{\partial e}{\partial t} = -\vec{V}\cdot\nabla e - P\nabla\cdot\vec{V} \qquad (5\text{-}30)$$

plus its artificial viscosity terms. The divergence $\overline{\nabla\cdot V}$ is based on velocities $\tilde{u}_{ij} = 1/2\, u^n_{ij} + \overline{u^{n+1}_{ij}}$ wherein the provisional values $\overline{u^{n+1}_{ij}}$ have already been calculated; likewise for $\tilde{v}$. In the second step, only the contributions of advection terms are calculated. The mass flux across each cell interface is calculated, using donor cell differencing (second upwind difference method, section III-A-11) based on the provisional values of velocities $\overline{u^{n+1}}$ and $\overline{v^{n+1}}$. This mass flux is used to calculate a new density ρ^{n+1}, and then to calculate only the advective contribution to u, v and $e_s = E_s/\rho$. Note that this final advective contribution must be added to the provisional value $\overline{u^{n+1}}$, etc., rather than the original values u^{n+1}, etc.

The PIC calculation is similar, but the mass flux calculation is based on a *finite number of particles* from the donor cell. The particles are not located at the center of the cell, but each particle p has its own Lagrangian coordinates, x_p and y_p. The particles are moved by the same velocity weighting used in the MAC code (see section III-G-4, equation 3-605). If the particle crosses the cell boundary, it contributes its mass, momentum, and internal energy to the averages in the new cell, upon which the pressures for that cell are calculated. As mentioned earlier, momentary crowding or depletion of particles in the cells will occur, producing a random high frequency oscillation of cell properties. This oscillation models the molecular behavior of the gases, but with very few computational molecules.

Both the PIC and FLIC methods use donor cell (second upwind) differencing for the advection terms and therefore have an implicit artificial viscosity (see sections V-E-1,2). Gentry, Martin, and Daly (1966) pointed out that the effect of $q \sim |u|$ in PIC and FLIC means that the artificial diffusion is not Galilean-invariant, i.e., the "wind tunnel transformation" does not apply to these computations.* Also, the method is locally unstable at stagnation points without the additional explicit q terms because the implicit $q \sim |u|$, according to Evans and Harlow (1958, 1959) and Longley (1960). See also equation (5-25) et seq. Both methods are presented in the original papers for both Cartesian and cylindrical coordinate systems.

The PIC method is most advantageously applied to interface problems (free surface or multiple materials), because the discrete particles may be assigned different masses, specific heats, etc., to represent two fluids, a free fluid surface, or even a fluid and a deformable solid. Solutions to the early problems of empty cells, boundary conditions,

*Also true of all upwind differencing methods.

and details of the particle weighting procedures have evolved over the years of successful application (Evans and Harlow, 1957, 1958, 1959; Evans et al., 1962; Harlow, 1963, 1964). A review of these techniques was given by Amsden (1966). Mader (1964) has extended the approach to include chemically reactive fluid dynamics in his Explosive-in-Cell or EIC method; Hirt (1965) also presented PIC calculations of shock detonation by explosives. The PIC approach was extended to plasma stability calculations by Dickman et al. (1969) and Morse and Nielson (1971). Armstrong and Nielsen (1970) demonstrated the good agreement of PIC transient computations with transform method calculations of the nonlinear development of a strong two-stream plasma instability. The accuracy has also been demonstrated by several PIC-like multi-material codes at Physics International (Buckingham et al., 1970; Watson and Godfrey, 1967; Watson, 1969). Amsden and Harlow (1965) calculated the gross features of supersonic turbulent flow in a base region. Crane (1968) attempted an accurate calculation of a hypersonic near wake problem using PIC with inviscid equations; the method is not well suited to this problem, and the calculation was unsuccessful. The accuracy of the FLIC method was independently ascertained by Gururaja and Dekker (1970) on several complex 2D shock-propagation problems, and by Satofuka (1970) in calculating 2D planar and cylindrical shock tube problems, Another FLIC-type code is the TOIL code of Johnson (1967); see also Hill and Larsen (1970) and Reynolds (1970). For references of other work on PIC and FLIC codes performed at Los Alamos Scientific Laboratory, see Harlow and Amsden (1970A).

Butler (1967) included viscosity and heat conduction in both PIC and FLIC, and found that the two methods produced comparable results.

V-E-4. Lax's Method

Lax's method* appears in Lax's((1954) fundamental paper on conservation equations. Lax was most concerned with the conservation principles and only secondarily with the finite-difference scheme. To stabilize calculations of the inviscid 1D equations (4-66) using forward-time, centered space differences, as in

$$U_i^{n+1} = U_i^n - \Delta t \left.\frac{\delta F}{\delta x}\right|_i^n \qquad (5\text{-}31)$$

he replaced the U_i^n in the right-hand member by its space average at time n.

$$U_i^{n+1} = \frac{1}{2}\left(U_{i-1}^n + U_{i+1}^n\right) - \Delta t \left.\frac{\delta F}{\delta x}\right|_i^n \qquad (5\text{-}32)$$

This simple and historically important method has several instructive properties. The space derivatives are centered and therefore appear to be second-order accurate, but the method is also diffusive. (Richtmyer, 1963, identifies it by the term "diffusing".) Consider the model equation (5-1) with $\alpha = 0$. Lax's method then gives

$$u_i^{n+1} = \frac{1}{2}\left(u_{i+1}^n + u_{i-1}^n\right) - \bar{u} \left.\frac{\delta u}{\delta x}\right|_i^n \qquad (5\text{-}33)$$

Expanding in Taylor series, as in Hirt's stability analysis (section III-A-5-c), we obtain

$$u_i^n + \frac{\partial u}{\partial t}\Delta t + \frac{1}{2}\frac{\partial^2 u}{\partial x^2}\Delta t^2 + O(\Delta t^3) = \frac{1}{2}\left[u_i^n + \frac{\partial u}{\partial x}\Delta x + \frac{1}{2}\frac{\partial^2 u}{\partial x^2}\Delta x^2 + \frac{1}{6}\frac{\partial^3 u}{\partial x^3}\Delta x^3 + O(\Delta x^4)\right]$$

$$+ \frac{1}{2}\left[u_i^n - \frac{\partial u}{\partial x}\Delta x + \frac{1}{2}\frac{\partial^2 u}{\partial x^2}\Delta x^2 - \frac{1}{6}\frac{\partial^3 u}{\partial x^3}\Delta x^3 + O(\Delta x^4)\right]$$

$$- \bar{u}\left.\frac{\partial u}{\partial x}\right|_i^n + O(\Delta x^2) \qquad (5\text{-}34)$$

---

*Commonly referred to as Lax's method. It first appears in open literature in a footnote of Courant et al. (1952) as the "scheme of J. Keller and P. Lax." Richtmyer (1963) also mentions K. O. Friedrichs in connection with it.

or

$$\frac{\partial u}{\partial t} = -u\frac{\partial u}{\partial x} + \frac{\Delta x^2}{2\Delta t}\frac{\partial^2 u}{\partial x^2} - \frac{\Delta t}{2}\frac{\partial^2 u}{\partial t^2} + O(\Delta x^2) \quad (5\text{-}35)$$

Using the relation

$$\frac{\partial^2 u}{\partial t} = \frac{\partial}{\partial t}\left(-u\frac{\partial u}{\partial x}\right) = -u\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial t}\right) = +u^2\frac{\partial^2 u}{\partial x^2} \quad (5\text{-}36)$$

we obtain

$$\frac{\partial u}{\partial t} = -u\frac{\partial u}{\partial x} + \left(\frac{\Delta x^2}{2\Delta t} - \frac{\Delta t}{2}u^2\right)\frac{\partial^2 u}{\partial x^2} + O(\Delta x^2) \quad (5\text{-}37)$$

From this transient analysis, Lax's method is seen to introduce an effective artificial diffusion coefficient,

$$\alpha_e = \left(\frac{\Delta x^2}{2\Delta t} - \frac{\Delta t}{2}u^2\right) = \frac{\Delta x^2}{2\Delta t}\left[1 - \frac{u^2\Delta t^2}{\Delta x^2}\right] \quad (5\text{-}38)$$

or

$$\alpha_e = \frac{\Delta x^2}{2\Delta x}\left[1 - c^2\right] \quad (5\text{-}39)$$

Stability in the model equation requires $\alpha_e \geq 0$ or $c \leq 1$, as usual. For $c = 1$, the exact solution of the model equation is obtained. Since the method is applied to all variables $U = \rho$, ρu, E_s, the artificial diffusion represents not only an artificial viscosity, but also artificial mass diffusion and heat conduction.*

The order of the truncation error is determined from equation (5-37) to be

$$E = O\left(\Delta x^2, \Delta t, \frac{\Delta x^2}{\Delta t}\right) \quad (5\text{-}40)$$

This equation indicates that, as $\Delta t \to 0$ for fixed Δx, the truncation error becomes unbounded. This indication *is* meaningful. It is disconcerting in the extreme to accidentally run a shock propagation code with $\Delta t = 0$, as the present author has done, and find that the shock still propagates! [Consider equation (5-32) with $\Delta t = 0$.] The disturbance does not actually propagate with a wave front, as a shock does, but diffuses out from the initial jump condition for $\Delta t = 0$.

For small enough Δt, the method obviously provides sufficient α_B to stabilize a strong shock calculation. For $c = 1$, the damping vanishes and the method cannot be used with shocks.

Lax's method is very easily extended to two and three dimensions, as

$$U_{ij}^{n+1} = \frac{1}{4}\left[U_{i+1,j}^n + U_{i-1,j}^n + U_{i,j+1}^n + U_{i,j-1}\right] + \Delta t\frac{\delta U^n}{\delta t} \quad (5\text{-}41)$$

$$U_{ijk}^{n+1} = \frac{1}{6}\left[U_{i+1,j,k}^n + U_{i-1,j,k}^n + U_{i,j+1,k}^n + U_{i,j-1,k}^n + U_{i,j,k+1}^n + U_{i,j,k-1}^n\right] + \Delta t\frac{\delta U^n}{\delta t} \quad (5\text{-}42)$$

*A diffusive scheme doubly violates the transportive property. Whereas the leapfrog methods (section III-A-6), for example, advect the effect of a perturbation upstream, against the velocity, a diffusive scheme also advects it at right angles to the velocity.

The corresponding stability requirements are

$$C_{2D} \equiv \frac{(V + a)\Delta t}{\Delta x \Delta y} \sqrt{\Delta x^2 + \Delta y^2} \leq 1 \qquad (5\text{-}43)$$

$$C_{3D} \equiv \frac{(V + a)\Delta t}{\Delta x \cdot \Delta y \cdot \Delta z} (\Delta x^2 + \Delta y^2 + \Delta z^2)^{3/2} \leq 1 \qquad (5\text{-}44)$$

Thus, for $\Delta x = \Delta y = \Delta z$, the largest possible Δt is reduced by a factor of $1/\sqrt{3} \approx 0.58$.

Exercise: Derive expressions for α_e of Lax's method in two and three dimensions.

Exercise: Determine the conditions for which Rusanov's method reduces to Lax's method.

Moretti and Abbett (1966A) used the two-dimensional version of Lax's method in conjunction with a patched characteristics solution in an attempt to calculate base flow. They noted a phenomenon which they called "stalling". That is, with a spatial gradient of properties such that

$$U_i^n \neq \hat{U}_i^n \equiv \frac{1}{4}\left(U_{i+1,j}^n + U_{i-1,j}^n + U_{i,j+1}^n + U_{i,j-1}^n\right) \qquad (5\text{-}45)$$

the time solution adjusted to a condition where

$$U_i^n - \hat{U}_i^n = \frac{\delta U^n}{\delta t} \Delta t \qquad (5\text{-}46)$$

so that $U_i^{n+1} = U_i^n$ for all i. The situation could be changed by changing Δt. Of course, the method was not intended to be used on this subsonic shock-free problem, but the example shows up another shortcoming of the method.

In spite of its shortcomings, the method has an important asset: simplicity. It is also easily adapted to cylindrical, spherical, and 3D problems. This appears to be the major reason for its use by Bohachevsky and Rubin (1966), Bohachevsky and Mates (1966), Bohachevsky and Kostoff (1971), Barnwell (1967), Xerikos (1968), and Emery and Ashurst (1971). Kentzner (1970B) experimented with using the Lax method and the midpoint leapfrog method (section III-A-6) at different time steps and in different weighted combinations, in a two-dimensional problem in which the shock discontinuity was treated as a boundary.

Because it is easily programmed and is dependable, Lax's method can be used to advantage in the early stages of program development. The program can be converted to more complex methods afterwards.

Exercise: Show that the use of Lax's method on the advection terms and FTCS differencing on the diffusion term of the model equation results in an unconditionally unstable method.

HINT: Use the analysis for the FTCS method, replacing α by $(\alpha + \alpha_e)$.

Exercise: Show that α_e of Lax's method by the steady-state analysis is $\alpha_e = \Delta x^2/2\Delta t$.

V-E-5. *Lax-Wendroff Method*

Lax and Wendroff (1960, 1964) investigated a class of methods which has attained considerable stature in theoretical studies of difference methods, and which led to a class of two-step methods (next section V-E-6) which are currently the most popular methods for solving compressible flow problems. Like Leith's method (section III-A-13), all these are based on a second-order Taylor series expansion in time, and all are identical to Leith's method for the constant-coefficient model equation.

Compared with Leith's method for incompressible flow, the application of the time expansion to compressible flow is greatly complicated because a *system* of equations is

Appendix B

```
DIM  r(50),m(50),e(50),nr(50),nm(50),ne(50)
DIM  fr(50),fm(50),fe(50),p(50),a(50),dadx(50),np(50),mach(50)
a(1)=1.5173
a(2)=1.462
a(3)=1.4027
a(4)=1.3399
a(5)=1.2675
a(6)=1.1812
a(7)=1.0862
a(8)=.9875
a(9)=.8995
a(10)=.8194
a(11)=.7738
a(12)=.7605
a(13)=.7617
a(14)=.7658
a(15)=.776
a(16)=.7927
a(17)=.8164
a(18)=.8487
a(19)=.8893
a(20)=.9334
a(21)=.9798
a(22)=1.0267
a(23)=1.0756
a(24)=1.1182
a(25)=1.1163
a(26)=1.1278
a(27)=1.1532
a(28)=1.1829
a(29)=1.2215
a(30)=1.2518
a(31)=1.2602
a(32)=1.2463
a(33)=1.2372
a(34)=1.2238
a(35)=1.2092
a(36)=1.2052
a(37)=1.1882
a(38)=1.1905
a(39)=1.2124
a(40)=1.2547
a(41)=1.2986
r(1)=.00033261#
r(2)=.00034841#
```

```
r(3)=.00036703#
r(4)=.0003892#
r(5)=.00041833#
r(6)=.00045941#
r(7)=.00051552#
r(8)=.0005917#
r(9)=.00068451#
r(10)=.00080715#
r(11)=.00091045#
r(12)=.00094969#
r(13)=.000946#
r(14)=.00093331#
r(15)=.00090439#
r(16)=.00086285#
r(17)=.00081293#
r(18)=.00075621#
r(19)=.00069773#
r(20)=.00064524#
r(21)=.00059902#
r(22)=.00055888#
r(23)=.0011533
r(24)=.0013227
r(25)=.0013217
r(26)=.0013278
r(27)=.0013401
r(28)=.001353
r(29)=.0013678
r(30)=.0013781
r(31)=.0013807
r(32)=.0013763
r(33)=.0013733
r(34)=.0013687
r(35)=.0013633
r(36)=.0013619
r(37)=.0013552
r(38)=.0013561
r(39)=.0013646
r(40)=.001379
r(41)=.001392
m(1)=.6187
m(2)=.6422
m(3)=.6692
m(4)=.7006
m(5)=.7407
m(6)=.7948
```

m(7)=.8643
m(8)=.9506
m(9)=1.0436
m(10)=1.1458
m(11)=1.2133
m(12)=1.2344
m(13)=1.2324
m(14)=1.2259
m(15)=1.2098
m(16)=1.1842
m(17)=1.1499
m(18)=1.1061
m(19)=1.0556
m(20)=1.0057
m(21)=.9582
m(22)=.9143
m(23)=.9292
m(24)=.8396
m(25)=.8408
m(26)=.8322
m(27)=.8141
m(28)=.7937
m(29)=.7685
m(30)=.7498
m(31)=.745
m(32)=.7534
m(33)=.7587
m(34)=.7671
m(35)=.7763
m(36)=.7789
m(37)=.79
m(38)=.7886
m(39)=.7744
m(40)=.7481
m(41)=.7229
e(1)=978.86
e(2)=1022.18
e(3)=1073.01
e(4)=1133.21
e(5)=1211.74
e(6)=1321.41
e(7)=1469.42
e(8)=1667.06
e(9)=1903.51
e(10)=2209.09

```
e(11)=2460.65
e(12)=2555.09
e(13)=2546.03
e(14)=2515.9
e(15)=2446.01
e(16)=2345.17
e(17)=2223.24
e(18)=2083.06
e(19)=1936.75
e(20)=1804.08
e(21)=1685.8
e(22)=1582.19
e(23)=2890.29
e(24)=3315.79
e(25)=3313.64
e(26)=3326.71
e(27)=3353.03
e(28)=3380.58
e(29)=3411.96
e(30)=3433.77
e(31)=3439.31
e(32)=3430!
e(33)=3423.6
e(34)=3413.82
e(35)=3402.4
e(36)=3399.3
e(37)=3385.14
e(38)=3387.13
e(39)=3405.1
e(40)=3435.67
e(41)=3463.18
p(41)=1300
np(41)=p(41)
nr(1)=r(1)
nm(1)=m(1)
ne(1)=e(1)
p(1)=.4*(e(1)-.5*m(1)*m(1)/r(1))
np(1)=p(1)
OPEN "friedss" FOR INPUT AS #1
FOR j=1 TO 41
  INPUT  #1,r(j),m(j),e(j),p(j),a(j),dadx(j)
NEXT j
CLOSE
X=.1427
T=.00002
```

```
cfl=T/X
10 k=k+1
LOCATE 1,1
PRINT "Time = ",k*T
PRINT "xs = ",xs
q$=INKEY$
IF q$="c" THEN CLS
IF q$="c" THEN k=2
IF q$="p" THEN np(41)=np(41)+100
IF q$="m" THEN np(41)=np(41)-50
IF q$="f" THEN nm(1)=nm(1)+.02
IF q$="o" THEN GOTO 51
FOR j=1 TO 40
  PSET(k/10+200,1200-50*xs)
 'PSET(k+200,1400-np(25))
  'PSET(j,100-10000*nr(j))
  'PSET(j,150-25*nm(j))
  'PSET(j,200-.01*ne(j))
  'PSET(j,250-25*a(j))
  dadx(j)=(a(j+1)-a(j))/X
  'PSET(j,300-50*dadx(j))
  PSET(j,300-25*mach(j))
  PSET(j,400-.02*np(j))
   np(j)=.4*(e(j)-.5*m(j)*m(j)/r(j))
NEXT j
FOR j=1 TO 41
IF k<2 THEN GOTO 12
  r(j)=nr(j)
  m(j)=nm(j)
  e(j)=ne(j)
12  fr(j)=m(j)*a(j)
  p(j)=np(j)
   PSET(50+4*j+2*k,400-.05*p(j)-3*k)
   fm(j)=a(j)*(m(j)*m(j)/r(j))
   fe(j)=a(j)*m(j)*(e(j))/r(j)
NEXT j
js=0
FOR j =2 TO 40
   nr(j)=r(j)-cfl*(fr(j)-fr(j-1))/a(j)
   nm(j)=m(j)-cfl*(fm(j)-fm(j-1))/a(j)-cfl*.5*(a(j+1)*p(j+1)-a(j-1)*p(j-1))/a(j)+T*p(j)*dac
   ne(j)=e(j)-cfl*(fe(j)-fe(j-1))/a(j)-cfl*.5*(a(j+1)*p(j+1)*m(j+1)/r(j+1)-a(j-1)*p(j-1)*m(
1))/a(j)
  np(j)=.4*(ne(j)-.5*nm(j)*nm(j)/nr(j))
  mach(j)=nm(j)/(nr(j)*SQR(1.4*np(j)/nr(j)))
 IF js>0 THEN GOTO 66
```

```
 IF mach(j)<1 THEN js=j
66 NEXT j
xs=js-1+(mach(js-1)-1)/(mach(js-1)-mach(js))
nr(41)=1.1*nr(40)-.1*nr(39)
'ne(41)=1.1*ne(40)-.1*ne(39)
nm(41)=1.1*nm(40)-.1*nm(39)
ne(41)=np(41)*2.5+.5*nm(41)*nm(41)/nr(41)
'nm(41)=SQR(nr(41)*2*(ne(41)-2.5*np(41)))
GOTO 10
51 OPEN "friedss" FOR OUTPUT AS #1
FOR j=1 TO 41
  WRITE  #1,r(j),m(j),e(j),p(j),a(j),dadx(j)
NEXT j
CLOSE
END
```

3.3261E-04,.6187,978.86,161.3707,1.5173,-.3875262
3.490575E-04,.6421022,1023.219,173.0542,1.462,-.4155576
3.688516E-04,.6692475,1075.901,187.5026,1.4027,-.4400836
3.934306E-04,.7006146,1139.949,206.4511,1.3399,-.5073579
4.271239E-04,.7406338,1226.446,233.7263,1.2675,-.6047654
4.756736E-04,.7947454,1349.906,274.3936,1.1812,-.6657325
5.437855E-04,.8642546,1521.851,334.0234,1.0862,-.6916607
6.352177E-04,.9506363,1752.094,416.3023,.9875,-.6166784
7.418117E-04,1.043639,2023.014,515.5511,.8995,-.5613174
8.449772E-04,1.145659,2292.624,606.3818,.8194,-.3195515
9.016081E-04,1.213173,2445.335,651.653,.7738,-9.320253E-02
9.11371E-04,1.23439,2475.782,655.9336,.7605,8.408975E-03
8.993929E-04,1.232445,2449.963,642.2195,.7617,2.873178E-02
8.787841E-04,1.225846,2403.358,619.348,.7658,7.147879E-02
8.479405E-04,1.209733,2330.654,587.0831,.776,.1170285
8.089402E-04,1.184248,2236.579,547.896,.7927,.1660827
7.638988E-04,1.149869,2125.713,504.114,.8164,.226349
7.15049E-04,1.106107,2002.318,458.7206,.8487,.284513
6.670071E-04,1.055609,1877.251,416.7781,.8893,.3090399
6.273364E-04,1.005734,1767.817,384.6516,.9334,.3251577
6.074699E-04,.9581046,1693.779,375.2855,.9798,.3286618
6.415262E-04,.9143353,1730.304,431.4897,1.0267,.342677
8.079688E-04,.8727631,2087.237,646.3432,1.0756,.2985278
1.138254E-03,.8395169,2871.563,1024.791,1.1182,-1.331435E-02
1.298073E-03,.8409532,3261.311,1195.566,1.1163,8.058865E-02
1.291775E-03,.8323805,3242.571,1189.758,1.1278,.1779961
1.313123E-03,.8140482,3290.748,1215.37,1.1532,.2081284
1.328064E-03,.7936094,3322.475,1234.144,1.1829,.2704982
1.348537E-03,.7685305,3368.435,1259.778,1.2215,.2123329
1.353843E-03,.7499269,3379.432,1268.693,1.2518,5.886533E-02
1.345854E-03,.7449259,3360.645,1261.796,1.2602,-9.740744E-02
1.332856E-03,.7532317,3329.793,1246.784,1.2463,-6.376987E-02
1.335863E-03,.7587691,3338.633,1249.258,1.2372,-9.390384E-02
1.326157E-03,.7670736,3316.679,1237.934,1.2238,-.102312
1.322379E-03,.7763314,3308.145,1232.106,1.2092,-2.803131E-02
1.32531E-03,.7789035,3317.32,1235.374,1.2052,-.1191308
1.310563E-03,.790042,3281.903,1217.51,1.1882,.0161179
1.32432E-03,.78851,3313.474,1231.493,1.1905,.1534684
1.33831E-03,.7742607,3346.921,1249.181,1.2124,.296426
1.370761E-03,.7481508,3399.041,1277.95,1.2547,.3076385
1.374007E-03,.7455398,3452.266,1300,1.2986,0