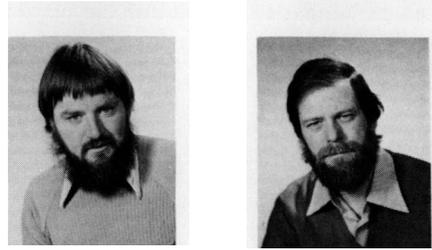


Keywords: differential quadrature, hyperbolic PDE, mathematical techniques, method of lines, partial differential equations

# The method of lines and the advective equation



by

Michael B. Carver and H. W. Hinds  
Atomic Energy of Canada Limited  
Chalk River Nuclear Laboratories  
Chalk River, Ontario, Canada K0J 1J0

MICHAEL B. CARVER, a native of Wales, obtained a BEng in mechanical engineering at McMaster University, Hamilton, Canada in 1963. He has since worked in aerodynamic research at English Electric Company, Rugby, and obtained an MSc in thermodynamics at Birmingham University. After working for several years in thermohydraulics research at Atomic Energy of Canada, Chalk River, he took his current position as numerical analyst in the Mathematics and Computation Branch, where he specializes in software for partial and ordinary differential-equation simulation. He also plays rugby, skis, sails, and writes poetry.

H. W. (TONY) HINDS attended McGill University in his hometown of Montreal, Canada, receiving his BEng in electrical engineering in 1964. Postgraduate studies at the University of Birmingham in England earned him an MSc. For the past 10 years, he has worked at the Chalk River Nuclear Laboratories of Atomic Energy of Canada Ltd., specializing in dynamic simulation of nuclear power plants and system identification by transfer-function measurement. His hobbies include cross-country skiing, canoeing, and beekeeping.

## ABSTRACT

*The linear advective equation is simple in form and yet it is one of the most difficult equations to solve accurately by numerical means. Because its behavior is similar to that of the conservation equations of compressible one- or two-phase flow, the advective equation is extremely useful for testing numerical schemes.*

*This paper summarizes techniques for solving the advective equation using the method of lines on a digital or hybrid computer, and assesses how suitable they are for incorporation into large simulations. Some of the dangers of using artificial dissipation terms are described, and a weighted-residual technique is shown to be very effective.*

## 1 INTRODUCTION

The method of lines, or differential quadrature, is one of the few techniques for solving partial differential equations (PDEs) which can be used with equal success on both digital and analog/hybrid computers. The technique consists of converting the PDEs into ordinary differential equations (ODEs) by finite difference, spline, or weighted-residual techniques, and integrating the resulting ODEs. The technique is problem-independent and is popular because of its versatility, modularity, and ease of implementation. Several recently published software packages for automated method-of-lines solution of arbitrarily defined PDEs have been very successful, particularly for parabolic and elliptic PDE systems.<sup>2,12,15</sup>

Hyperbolic equations are more difficult to solve because of their characteristic behaviour. First-order hyperbolic equations are particularly difficult to solve because they transmit discontinuities without dispersion or dissipation. Unfortunately, any attempt to use a finite number of space intervals introduces dispersion and spurious oscillation. Traditional symmetric techniques can only be used if an arbitrary second-order artificial viscosity or damping term is added to the equation. Directional (or upwind) techniques have proved superior for both finite-difference and finite-element analyses, since they diminish the oscillation problem, although they do not eliminate it.<sup>3,13</sup>

The linear advective equation

$$\frac{\partial u}{\partial t} + v \frac{\partial u}{\partial x} = 0$$

is the simplest imaginable PDE, and the great difficulty of solving it numerically causes considerable embarrassment to those attempting to solve complex PDE systems. However, it does provide a good case for testing methods to be used on systems of hyperbolic equations. Many schemes have been tested on it,

generally by using a propagating step, sine, Gaussian, or Chapeau (triangular) wave form.<sup>6,11</sup>

Frequency analysis of a proposed model is very informative and may be readily done on a hybrid computer if analytical methods are too complex. The frequency response can be obtained by perturbing the model with a pseudorandom binary sequence (PRBS), and analyzing the response with a Fast Fourier Transform (FFT). This gives gain and phase as functions of frequency. Identical results can be obtained by using sine-wave excitations at many different frequencies, but the PRBS/FFT technique is comprehensive.

The above techniques were used to provide a rationale for choosing a spatial scheme for first-order hyperbolic equations. The optimal choice is not invariant, but depends on the application. An optimal scheme minimizes not only errors in amplitude and phase velocity but also spurious oscillations, which can cause serious numerical instability.<sup>17</sup> Thus a measure of the quality of numerical schemes must include the integrated absolute error obtained from the time response in addition to the gain and phase error obtained from the frequency response.

We are interested in developing suitable techniques for solving the conservation equations of compressible one- or two-phase flow. These are normally systems of three coupled hyperbolic equations which can transmit discontinuities in the spatial profiles of one or more variables. Thus we use the advective equation as a test case to develop a suitable technique, and then progress to the conservation equations. Because the technique must be used in large systems, we must add to our criteria of minimum phase, gain, and integration errors the requirements that the technique be computationally fast and problem-independent.

Available schemes were assessed using a hybrid computer and the digital method-of-lines packages FORSIM<sup>4</sup> and PDECOL.<sup>12</sup> We found that we could improve the facilities for hyperbolic equations in the FORSIM package by incorporating an upwind weighted-residual technique. This technique is similar to but superior to the use of an artificial viscosity term and could easily be used in any package.

## 2 METHODOLOGY

Consider the advective equation

$$\frac{\partial u}{\partial t} + v \frac{\partial u}{\partial x} = 0 \quad (1)$$

and represent the distributed variable  $u$  by some approximating function at a finite number of points  $N$ .

$$u_a(x, t) = \sum_{i=1}^N b_i(x) u_i(t) \quad (2)$$

The  $x$  and  $t$  variables are now separated into basis functions  $b_i(x)$  and time functions  $u_i(t)$ .

There are now several recourses, of which we will consider two major categories.

### (a) Finite-difference method

Obtain an approximation to the term  $\partial u / \partial x$  by differentiating (2):

$$\left( \frac{\partial u}{\partial x} \right)_a = \frac{\partial}{\partial x} [u_a(x, t)] = \sum_{i=1}^N u_i(t) \frac{\partial}{\partial x} [b_i(x)] \quad (3)$$

This may then be substituted in (1) to give an explicit definition of  $\partial u / \partial t$  at each station  $j$ :

$$\left( \frac{\partial u}{\partial t} \right)_j = -v \left( \frac{\partial u}{\partial x} \right)_j = -v \sum_{i=1}^N \frac{\partial b_i(x)}{\partial x} u_i \quad (4)$$

Equation 4 may then be integrated as a set of  $N$  explicitly defined ODEs of the form  $\dot{\underline{U}} = \underline{K} \underline{U}$ .

### (b) Finite-element method

In this case substitute (2) in both terms of (1), and obtain an expression for the residual  $R$  that is the absolute value of the deviation:

$$\sum_{i=1}^N b_i(x) \frac{\partial u_i(t)}{\partial t} + v \sum_{i=1}^N u_i(t) \frac{\partial b_i(x)}{\partial x} = R \quad (5)$$

Now attempt to minimize  $R$  by integrating over the domain of interest using an appropriate weighting function  $w_j(x)$ , the choice of which defines the method.

$$\int_{x=x_1}^{x_N} \left( \sum_{i=1}^N w_j(x) b_i(x) \frac{\partial u_i}{\partial t} + v \sum_{i=1}^N w_j(x) \frac{\partial b_i(x)}{\partial x} u_i(t) \right) dx = 0 \quad (6)$$

Thus for either the finite-difference method or the finite-element method, the system of ordinary differential equations becomes

$$\underline{\dot{M}} \underline{U} = \underline{K} \underline{U} \quad (7)$$

where  $\underline{M}$  and  $\underline{K}$  are known as the mass and stiffness matrices in finite-element terminology.

For finite elements,  $\underline{M}$  and  $\underline{K}$  are normally sparse and banded, and  $\underline{M}$  is normally symmetric. For finite-difference methods, the form of  $\underline{K}$  is similar, but  $\underline{M}$  is the diagonal matrix  $\underline{I}$ .

Thus the main difference between the methods is the structure of the  $\underline{M}$  matrix. However, the components of  $\underline{M}$  are independent of time, so the associated standard upper and lower triangle ( $LU$ ) decomposition need be done only once to obtain

$$\underline{\dot{U}} = \underline{M}^{-1} \underline{K} \underline{U} \quad (8)$$

Direct matrix inversion of (8) is not efficient on a hybrid computer. An alternate implicit method can be derived by dividing  $\underline{M}$  into its diagonal ( $\underline{M}_D$ ) and off-diagonal ( $\underline{M}_O$ ) parts.

This leads to

$$\underline{\dot{U}} = \underline{M}_D^{-1} (\underline{K} \underline{U} - \underline{M}_O \underline{\dot{U}}) \quad (9)$$

which can be easily solved on a hybrid computer provided that the elements of  $\underline{M}_D^{-1} \underline{M}_O$  are small compared to unity.

### 3 THE APPROXIMATING FUNCTION

Many different approximating functions can be used, even for the finite-difference formulation; this number is still larger for the finite-element approach. We shall consider the more popular choices.

#### Lagrangian polynomials

One may develop one class of expressions for the spatial derivative by differentiating the Lagrange interpolation polynomial

$$b_j(x) = \frac{\prod_{i=1, i \neq j}^m (x - x_i)}{\prod_{i=1, i \neq j}^m (x_j - x_i)} \quad j \in (1, m) ; \quad i \neq j \quad (10)$$

Using (10), we may obtain the two most common approximations

$$\left(\frac{\partial u}{\partial x}\right)_i = \left(\frac{u_i - u_{i-1}}{\Delta x}\right) \quad (11)$$

(which is the two-point upwind formula when the velocity  $v$  is positive) and the three-point central formula

$$\left(\frac{\partial u}{\partial x}\right)_i = \left(\frac{u_{i+1} - u_{i-1}}{2 \Delta x}\right) \quad (12)$$

Higher-order approximations not only are more accurate but also reduce the stiffness of the resulting ODE set.<sup>10</sup> Thus we may also derive from (10) a three-point upwind formula

$$\left(\frac{\partial u}{\partial x}\right)_i = \frac{3u_i - 4u_{i-1} - u_{i-2}}{2 \Delta x} \quad (13)$$

and a four-point upwind-biased formula

$$\left(\frac{\partial u}{\partial x}\right)_i = \frac{-u_{i+1} + 6u_i - 3u_{i-1} - 2u_{i-2}}{6 \Delta x} \quad (14)$$

Other formulae can be obtained similarly and a complete list is given in Reference 1.

#### Hermitian interpolation

Lagrangian polynomials are based on functional values alone, whereas Hermitian formulae also involve derivatives. Any Hermitian formula can be expressed as

$$u_h(x) = \sum_{i=1}^{n1} \alpha_i(x) u_i + \sum_{i=1}^{n2} b_i(x) \left(\frac{\partial u}{\partial x}\right)_i \quad (15)$$

where the  $\alpha_i$  and  $b_i$  functions are  $m = n1 + n2$  polynomials of order  $m-1$ , in which the coefficients are determined from the identities

$$u(x_i) = u_i, \quad \frac{\partial}{\partial x}[u(x_i)] = \left(\frac{\partial u}{\partial x}\right)_i, \quad \text{etc.}$$

As derivatives of order greater than  $m$  are zero, formulae for first and second derivatives can be developed by differentiating (15). For example, considering (15) in an upwind sense,  $n1 = 2$ ,  $n2 = 1$  gives the formula

$$\left(\frac{\partial u}{\partial x}\right)_i = 2 \frac{u_i - u_{i-1}}{\Delta x} - \left(\frac{\partial u}{\partial x}\right)_{i-1} \quad (16)$$

Repeating for  $n2 = 2$  gives a function  $u_h(x)$  with a continuous first derivative. Other values of  $n1$  and  $n2$  may be used;  $n1 = 3$ ,  $n2 = 1$  gives

$$\left(\frac{\partial u}{\partial x}\right)_i = \frac{1}{2} \left[ \frac{u_{i+1} + 4u_i - 5u_{i-1}}{2 \Delta x} - \left(\frac{\partial u}{\partial x}\right)_{i-1} \right] \quad (17)$$

#### Spline functions

Cubic splines may be generated by assuming a cubic interpolation function between the nodes and forcing continuity of the function and its first two derivatives. In addition, the assumption of a boundary condition at each endpoint is required. The assumption of zero second derivatives at the endpoints is the most convenient in our case. For equally spaced nodes, this will lead to one of the tridiagonal equation sets

$$\frac{u'_{i+1} + 4u'_i + u'_{i-1}}{6} = \frac{u_{i+1} - u_{i-1}}{2 \Delta x} \quad (18)$$

or

$$\frac{u''_{i+1} + 4u''_i + u''_{i-1}}{6} = \frac{u_{i+1} - 2u_i + u_{i-1}}{(\Delta x)^2} \quad (19)$$

where

$$u'_i = \left(\frac{\partial u}{\partial x}\right)_i, \quad u''_i = \left(\frac{\partial^2 u}{\partial x^2}\right)_i$$

The spline interpolation formulae are given by the Hermitian interpolants for  $x_i < x < x_{i+1}$

$$\begin{aligned} u(x) = & u_i \left[ 3 \left(\frac{x_{i+1} - x}{\Delta x}\right)^2 - 2 \left(\frac{x_{i+1} - x}{\Delta x}\right)^3 \right] \\ & + u_{i+1} \left[ 3 \left(\frac{x - x_i}{\Delta x}\right)^2 - 2 \left(\frac{x - x_i}{\Delta x}\right)^3 \right] \\ & + u'_i \left[ \left(\frac{x_{i+1} - x}{\Delta x}\right)^2 - \left(\frac{x_{i+1} - x}{\Delta x}\right)^3 \right] \\ & + u'_{i+1} \left[ - \left(\frac{x - x_i}{\Delta x}\right)^2 + \left(\frac{x - x_i}{\Delta x}\right)^3 \right] \end{aligned} \quad (20)$$

$C^N$  splines of order  $N+1$  have been used for  $N = 0$  to 20,<sup>12</sup> but in general, cubic splines are the highest practical order, as a banded matrix solution analogous to (18) is time-consuming for higher orders.

#### 4 ARTIFICIAL DISSIPATION

Traditional symmetric approaches to defining the interpolation polynomial lead to a solution with severe spurious oscillations. Directional approaches tend to be more stable.

Richtmeyer and Morton<sup>14</sup> have summarized the two approaches for low-order finite-difference techniques. The upwind two-point difference (Equation 11) yields

a non-oscillatory, heavily dispersed result, while the central difference of Equation 12 yields a solution with severe spurious oscillations trailing the signal wave.

They, therefore, suggest the addition of an artificial second-order viscosity term which will dissipate the oscillations, hopefully without greatly affecting the true solution. Thus (1) becomes

$$\frac{\partial u}{\partial t} + v \frac{\partial u}{\partial x} - \alpha \frac{\partial^2 u}{\partial x^2} = 0 \quad (21)$$

where  $\alpha$  is chosen empirically. Again one can use finite-element or finite-difference techniques to solve (21).

This causes some concern, since mathematically (21) now requires an extra boundary condition not apparent from the physics. In practice no extra condition is imposed.

Heydweiler and Sincovec<sup>8</sup> have made an alternative attempt to combat dissipation and oscillation by combining (11) and (12) parametrically as follows

$$\left(\frac{\partial v}{\partial x}\right)_i = \frac{1}{\beta} \left( \frac{\beta-1}{2} \cdot \frac{u_{i+1} - u_i}{\Delta x} + \frac{\beta+1}{2} \cdot \frac{u_i - u_{i-1}}{\Delta x} \right) \quad (22)$$

Note that (21) reduces to (11) for  $\beta = 1$ , and approaches (12) for large  $\beta$ .

#### 5 WEIGHTING FUNCTIONS

The weighting functions used in the finite-element method are normally of the Galerkin type  $w_j(t) = b_j(x)$ . In particular, if one uses the Galerkin weighted-residual method with piecewise linear 'Chapeau' basis functions as described by Strang and Fix<sup>14</sup> on (21), one obtains the formula

$$\frac{1}{1+\gamma} \left( \gamma \frac{du_{i+1}}{dt} + \frac{du_i}{dt} + \gamma \frac{du_{i-1}}{dt} \right) + v \left( \frac{u_{i+1} - u_{i-1}}{2\Delta x} \right) - \alpha \left( \frac{u_{i+1} - 2u_i + u_{i-1}}{(\Delta x)^2} \right) = 0 \quad (23)$$

Equation 23 is stated in parametric form using  $\gamma$ . When  $\gamma = 0$ , Equation 23 reduces to the finite-difference formulation,  $\gamma = \frac{1}{2}$  gives the linear finite-element formulation, and other values of  $\gamma$  may be readily explored. Vichnevetsky<sup>18</sup> has developed optimum combinations of  $\alpha$  and  $\gamma$  for Equation 23, showing that a nonzero value of  $\gamma$  affects accuracy more than a nonzero value of  $\alpha$ . Thus the finite-element formulation is considerably more powerful than the finite-difference formulation. Unfortunately, the optimum  $(\alpha, \gamma)$  combination is a function of dimensionless time  $vt/x$ .

#### The upwind weighted-residual method

Since upwind techniques have proved useful and the finite-element approach is more accurate than centered finite differences, a combination of upwind and finite-element techniques would seem to be desirable.

Upwind weighting of finite elements has been considered by Christie *et al.*<sup>5</sup> for solving the steady-state diffusion equation with convective terms.

$$k \frac{\partial^2 T}{\partial x^2} - u \frac{\partial T}{\partial x} = 0 \quad (24)$$

Here we shall apply a similar philosophy to the advective equation, choosing linear basis functions  $b_i(x)$  in (2) as follows:

$$b_i(z) = z/h \quad ; \quad x_{i-1} < x < x_i, \quad z = x - x_{i-1}, \quad h = \Delta x \\ 1 - z/h \quad ; \quad x_i < x < x_{i+1}, \quad z = x - x_i \quad (25)$$

For the standard Galerkin approach, the weighting functions  $w_j(x)$  in (6) are the shape functions themselves  $w_j(x) = b_j(x)$ . For upwind elements, choose instead

$$w_j = b_j + \epsilon f(x) \quad \text{for the upwind element} \\ = b_j - \epsilon f(x) \quad \text{for the downwind element} \quad (26)$$

where  $\epsilon$  is a weighting parameter to be investigated below. The function  $f(x)$  is chosen to satisfy the requirements  $f(0) = f(h) = 0$  and the simplest form is

$$f(x) = \frac{3x}{h^2} (h - x) \quad (27)$$

Substituting (19) to (21) in (6) and performing the integrations, we get

$$\frac{1}{6} \left[ \left(1 + \frac{3\epsilon}{2}\right) u_{i-1}^! + 4u_i^! + \left(1 - \frac{3\epsilon}{2}\right) u_{i+1}^! \right] + \frac{v}{2h} \left[ (1 - \epsilon) u_{i+1} + 2\epsilon u_i - (1 + \epsilon) u_{i-1} \right] = 0 \quad (28)$$

which can be rearranged to give

$$\frac{1}{6} \left[ \left(1 + \frac{3\epsilon}{2}\right) u_{i-1}^! + 4u_i^! + \left(1 - \frac{3\epsilon}{2}\right) u_{i+1}^! \right] + \frac{v}{2h} (u_{i+1} - u_{i-1}) - \frac{\epsilon v}{2h} (u_{i+1} - 2u_i + u_{i-1}) = 0 \quad (29)$$

This is identical in form to the standard statement of linear finite elements with artificial dissipation, except that the mass matrix is no longer symmetric. The upwind weighting appears in both the stiffness and mass matrices, and it will be seen below that this greatly increases the effectiveness of the method. Also, as (29) is obtained directly from (1) without adding an artificial dissipation term as in (20), no extra boundary condition is required. This is in agreement with the *ad hoc* omission of the extra condition when using (21). It is interesting to note that (14) and (17) can also be rearranged to reveal dissipative terms.

#### 6 ASSESSMENT CRITERIA

By using an error-controlled variable-step algorithm, such as the Hindmarsh-Gear integrator used in FORSIM,<sup>3</sup> or by integrating on the analog computer, one can ensure that the integration error is negligible in comparison with the spatial error.<sup>7</sup> Spatial error can always be reduced by finer discretization, but this is expensive. One may, however, assess the relative efficiency of various spatial schemes using the same integrator and the same spatial grid.

We have performed two basic tests on these systems giving frequency response and time response. The frequency response to sinusoidal inputs is represented by the gain (in dB) and phase shift for each frequency. It contains the same information as the time response, but in a format which is more useful in certain applications. The frequency response is compared to the ideal and the error is quantified in terms of bandwidth (frequency range for gain error of less than 3dB) and phase error.

Bandwidth is strongly related to dispersion, with low bandwidth indicating large dispersion. However, large bandwidths may also result from "peaking" in the gain curve above unity at some frequencies. This high gain will generate excessive spurious oscillation in the time response.

The phase error represents variation of velocity with frequency and also is strongly related to spurious oscillation. A negative phase error indicates that the velocity is too great, and oscillations lead the step response; a positive phase error indicates that the velocity is too low, and oscillations trail the step response.

In most cases, the frequency response may be obtained analytically, as shown by Vichnevetsky.<sup>18</sup> However, in our case we choose to use the Pseudo-Random Binary Sequence/Fast Fourier Transform technique.<sup>9</sup> This is much easier to implement on the analog computer than on the digital computer. A filtered PRBS signal is used as the input perturbation function. This signal is ideal for the purpose as it contains components of significant amplitude over a large frequency range. The input and output signals of the model are then analysed by an FFT-based program. The frequency response, which is also the Fourier transform of the impulse response, is given by

$$F(j\omega) = \frac{U_N(j\omega)}{U_0(j\omega)}$$

where

$U(j\omega)$  = Fourier transform of  $u(t)$

$F(j\omega)$  = frequency response

and the subscripts 0 and  $N$  refer to the input and output nodes, respectively.

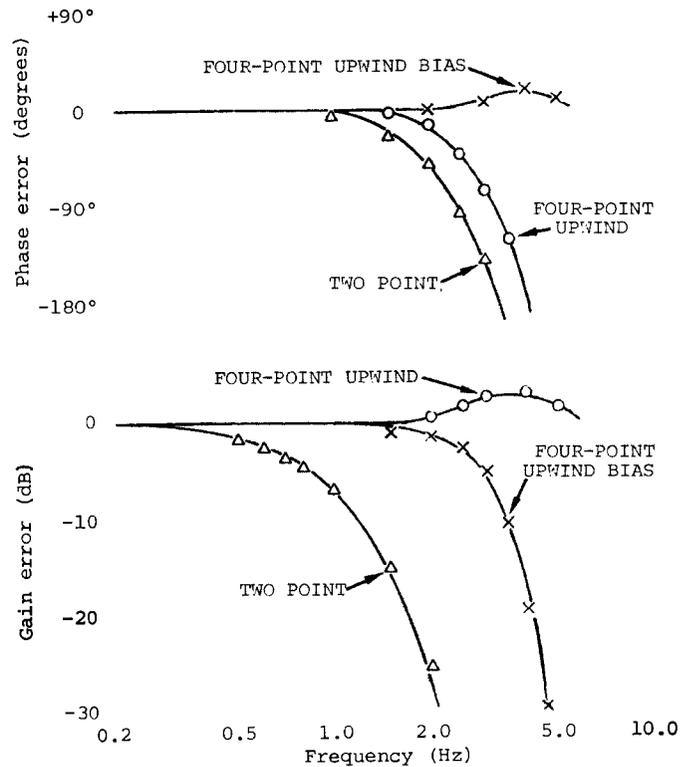
The time response is the transient response at various points  $x_i$  to an input perturbation. Accuracy may be quantified by integrating the error, which appears as

- (a) Spurious oscillations leading or trailing the signal
- (b) Dispersion of the signal over time.

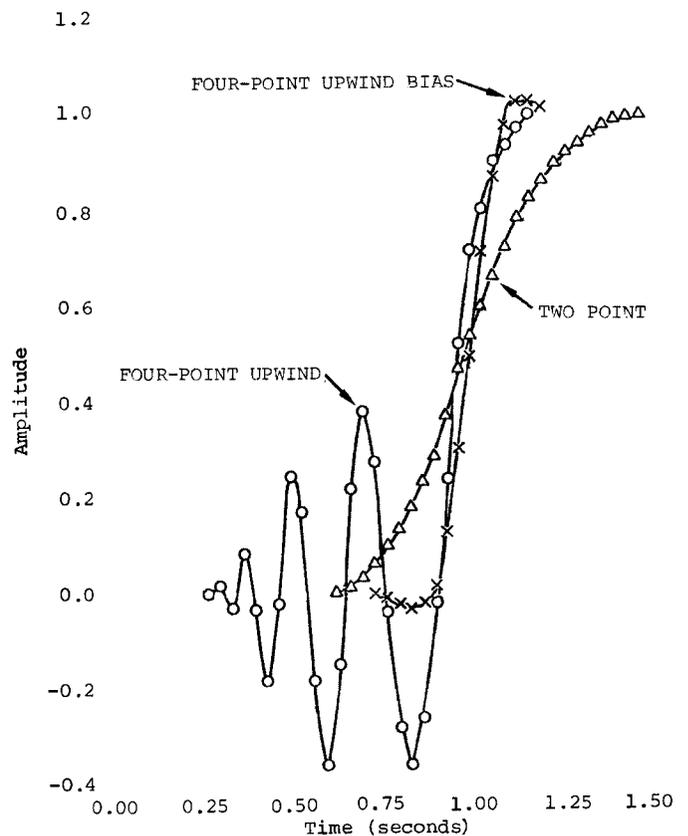
To ensure that both these effects are incorporated in the error assessment, the absolute error was integrated for each scheme over a time interval of twice the theoretical transit time of a perturbation. Thus at any point  $x_i$

$$E(x_i) = \int_0^{2L/v} |U_{\text{exact}} - U_{\text{calculated}}| dt \quad [\text{where } x \in (0, L)] \quad (30)$$

The overall error, obtained by integrating  $x$  from 0 to  $L$ , was approximated by using the four points  $x = 0.25L, 0.5L, 0.75L, \text{ and } L$ .



(1a) Frequency response at  $x = 1$



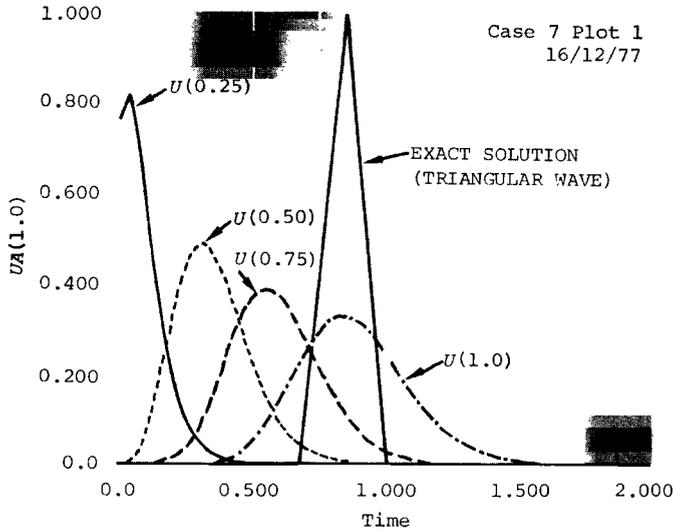
(1b) Time response at  $x = 1$

Figure 1 - Frequency and time responses of selected finite-difference schemes

$$E_0 = \frac{1}{L} \int_0^L E(x) dx \approx \sum_{i=1}^4 E(x_i) \quad (31)$$

$$\bar{E}_j = E_{0j} / E_{03C} \quad (32)$$

Finally, a relative error for each scheme  $j$  is obtained by dividing by the overall error for the three-point central-finite-difference scheme,  $E_{03C}$



Progression of a triangular wave  
2-point upwind finite difference

Figure 2 - Time response of two-point upwind scheme to a triangular wave. Plots are values of time responses  $U$  at four stations  $x_i$

A time response may also be obtained from the frequency response by means of an inverse Fourier transform. To further illustrate the reciprocal relationship between frequency and time responses, we show in Figure 1a the *frequency response* for two-point upwind, four-point upwind, and four-point upwind biased finite-difference simulations at the point  $x=1$ . Figure 1b shows the corresponding *time responses*. Note that the two-point upwind formula has considerable dissipation because of its low bandwidth, and that the peak in the gain curve of the four-point upwind scheme results in large spurious oscillations leading the signal. Finally, the four-point upwind bias method has a good bandwidth, no positive gain error, and a well-behaved time response.

On the basis of the step response test alone, one might be prepared to accept the two-point difference scheme since it behaves very stably. However, the low bandwidth shows that any waveform will be severely attenuated by this scheme. This is illustrated by the progression of a triangular wave in Figure 2.

## 7 BOUNDARY NODES

In both the finite-element and finite-difference approaches, it is normally not possible to use exactly the same scheme at the boundary nodes as at the others. For finite-element schemes the matrix is set up to eliminate  $u_1$ . For finite-difference formulae one can maintain directionality at the expense of order or vice versa, i.e., one can maintain directionality by using progressively fewer points or maintain order by keeping the same number of points and sacrificing directionality. We have found that the former degrades the frequency response considerably, and the latter is preferable even though it may require the use of a downwind formulation for the first node.

Table 1

Normalized integrated error and computation time for selected schemes

Numerical scheme	Equation number	E-error		Computation time*
		Step	Triangle	
Upwind Residual $\epsilon=0.3$	28	0.28	0.21	1.5
PDECOL KO=7	-	0.27	-	121.0
PDECOL KO=6	-	0.28	-	60.0
PDECOL KO=5	-	0.30	-	31.0
Linear Finite-Element $\gamma=0.5, \alpha'=0.08$	23	0.35	0.26	1.5
Hermite 3,1	17	0.35	0.24	1.0
Five-Point Upwind Bias	-	0.36	0.24	1.0
PDECOL KO=4	-	0.39	-	15.0
Four-Point Upwind Bias	14	0.43	0.30	1.0
LFE $\gamma=0.5, \alpha=0$	23	-	-	-
UWR $\epsilon=0$	28	0.57	0.50	1.5
Cubic Spline Finite Difference	18	-	-	-
Hermite 2,1	16	0.58	0.35	1.0
Three-Point Upwind	13	0.60	0.46	1.0
Heyweiler Sincovec Best	22	0.68	0.53	1.0
PDECOL KO=3	-	0.73	-	5.0
Two-Point Upwind	11	0.86	0.75	1.0
Three-Point Central	12	1.00	1.00	1.0
Five-Point Central	-	1.20	1.40	1.0

\* normalized with respect to three-point Central Finite Scheme

## 8 ANALYSIS OF SELECTED SCHEMES

The above techniques were used to provide integrated error and bandwidth as criteria of assessment for selected spatial schemes. To provide a relative efficiency criterion, we also include the digital computation time for the FORSIM package, normalized with respect to the three-point central finite-difference scheme. For 25 nodes, this is about two seconds, or approximately real time, on the CDC CYBER 170 model 175 computer. Analog time is about four milliseconds.

The results are summarized in Table 1. Included also are results from the PDECOL package using a finite-element collocation method with splines of up to sixth order ( $KO=7$ ) to illustrate what accuracy can be obtained if no restriction is placed on computational complexity.

While quite accurate, the high-order spline methods in PDECOL obviously consume excessive time when the banding in the matrices is wider than tridiagonal. The Upwind Weighted-Residual (UWR) method is superior in performance to the Linear Finite-Element method with Artificial Dissipation (LFEAD) because of the additional weighting in the mass matrix. In either case the choice of weighting factor  $\alpha$  or  $\epsilon$  is somewhat arbitrary, but it is important to note that the UWR method is much less sensitive to this choice. In applying (23) to various combinations of  $v$ ,  $\Delta x$ , and signal amplitude, it becomes obvious that the effect of a given  $\alpha$  depends strongly on  $v$  and  $\Delta x$ . This may be rationalized by rearranging (23) as follows:

$$\frac{1}{1+\gamma} \left( \frac{\gamma}{2} \frac{du_{i+1}}{dt} + \frac{du_i}{dt} + \frac{\gamma}{2} \frac{du_{i-1}}{dt} \right) = \frac{-V}{2\Delta x} [U_{i+1} - u_{i-1} + \alpha'(u_{i+1} - 2u_i + u_{i-1})] \quad (33)$$

This indicates that  $\alpha' = 2\alpha/v\Delta x$ , not  $\alpha$ , is the relevant parameter. Physically, this means the diffusion added must be related to the convection. This may also be discovered by comparing (33) and (29), since equality of the right-hand sides occurs when  $\epsilon = 2\alpha/v\Delta x$ . Using this relationship, any difference in the two methods will be due entirely to the upwind weighting of the mass matrix in (29). Figure 3 therefore shows normalized error for step-wave propagation in (23) and (28) for various combinations of  $\gamma$ ,  $\alpha'$ , and  $\epsilon$ . Curve A is Equation 23 with  $\gamma = 0$  or three-point finite difference, curve B is (23) with  $\gamma = 0.5$  or three-point finite element. Curve C is (29), the upwind weighted residual. Similar curves may be obtained for triangular waves.

The UWR method produces only a marginal improvement in minimum attainable error, but this method performs well over quite a wide range of  $\epsilon$  because of the upwind weighting of the mass matrix. The LFE method is much more sensitive to the choice of  $\alpha'$ , having an optimum value  $\alpha' \approx 0.8$ .

However, in applications, the appropriate form of UWR must be developed for each equation set and this may be a disadvantage. The weights can of course be estimated or the integrals evaluated numerically, but this will tend to destroy the superior accuracy margin of the method. For general applications, therefore, it is more straightforward and almost as effec-

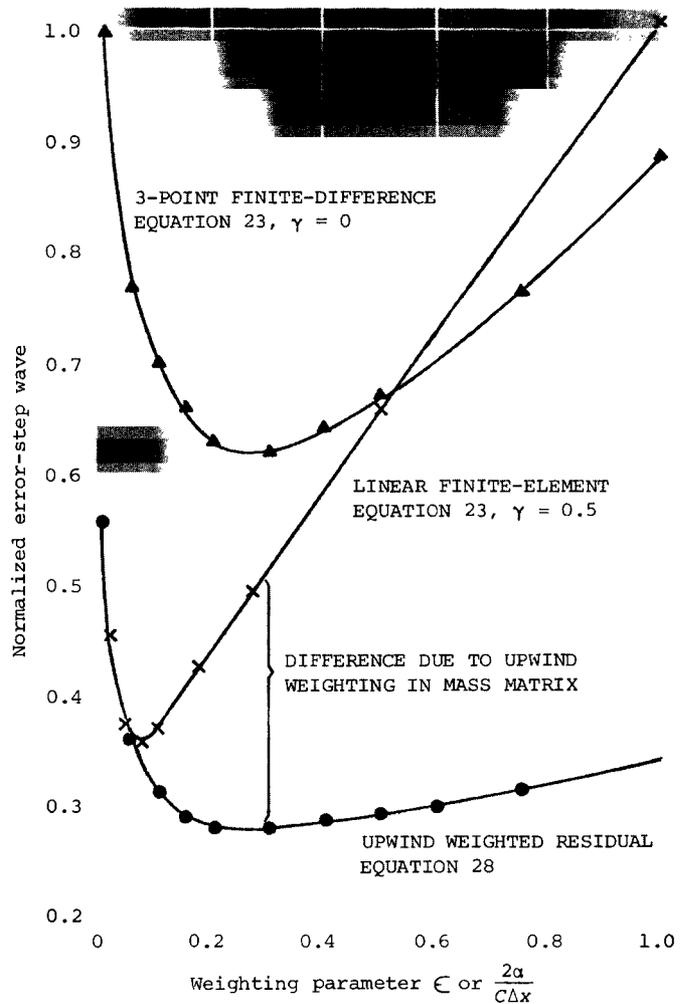


Figure 3 - Effect of weighting parameter

tive to use artificial dissipation in the finite-element formulation by applying a cubic-spline approximating function in the quasi-Lagrangian sense, as long as the weighting parameter is chosen with care.

The Hermite 3,1 formulation, Equation 17, also performs well if one can estimate  $(\partial u/\partial x)_0$ . For our simple test cases we use a first-order estimate. Finally, note that the simple four- and five-point upwind biased formulae are fast, stable and reasonably accurate. The boundary conditions are simpler to include than for the spline case, and one merely ensures that the boundary formulae maintain the order of approximation as mentioned above.

The gain and phase frequency response of the optimum UWR scheme is given in Figure 4, and the time responses at  $x = 1$  for the above four schemes are given for a triangular signal wave in Figure 5.

Finally, Figure 6 shows bandwidth, normalized by multiplying by  $x/v$ , as a function of number of nodes for selected schemes, and may be used to choose an appropriate number of nodes for a required performance. The theoretical maximum bandwidth is given by Shannon's sampling theorem, which states that there must be at least two samples per cycle of the highest frequency present. In practice, attainable bandwidth

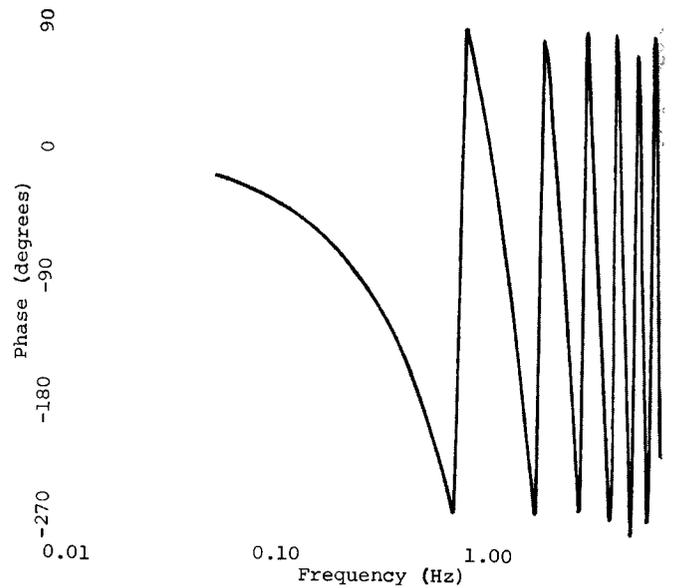
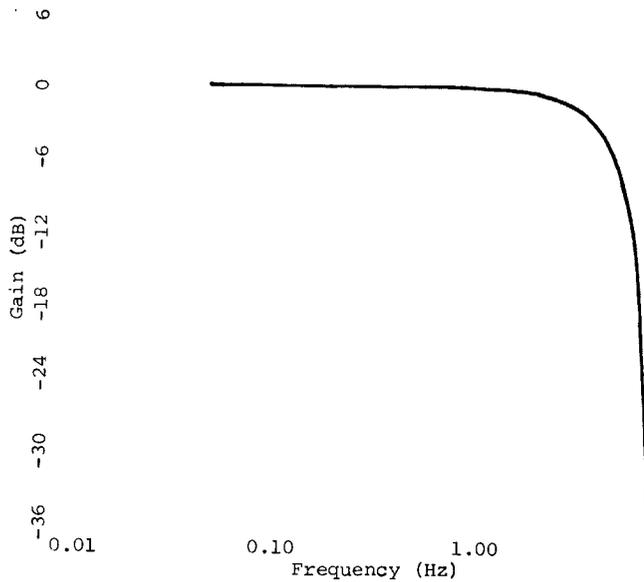
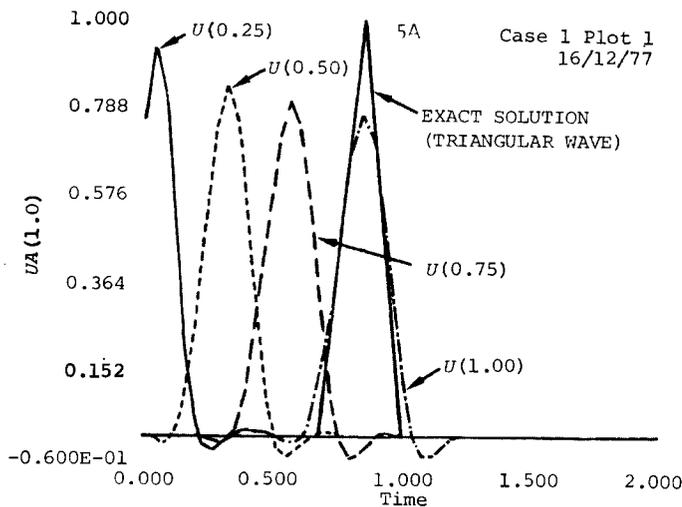
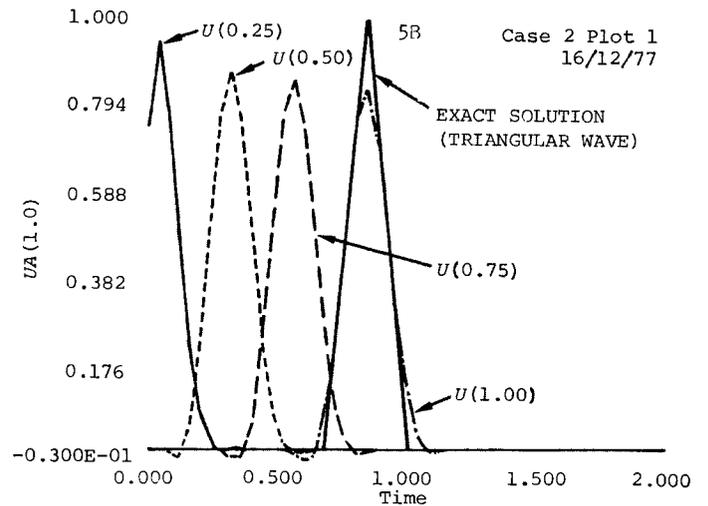


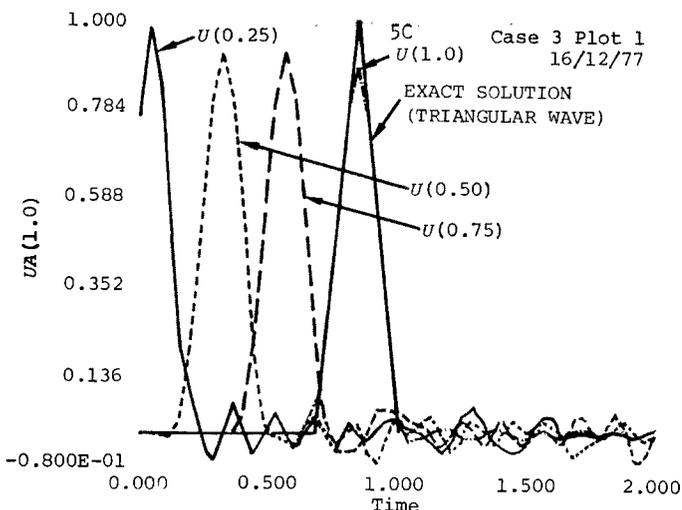
Figure 4 - Linear finite-element frequency response; upwind weighting factor = 0.50



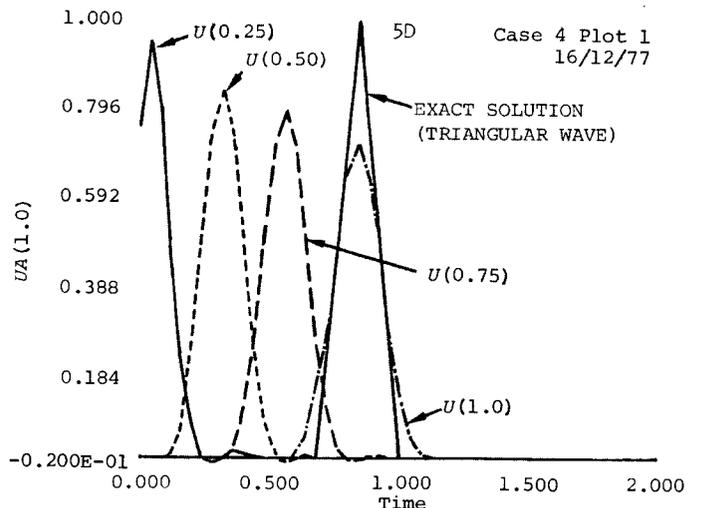
5a Progression of a triangular wave  
4-point upwind bias finite difference



5b Progression of a triangular wave  
3-point upwind Hermite

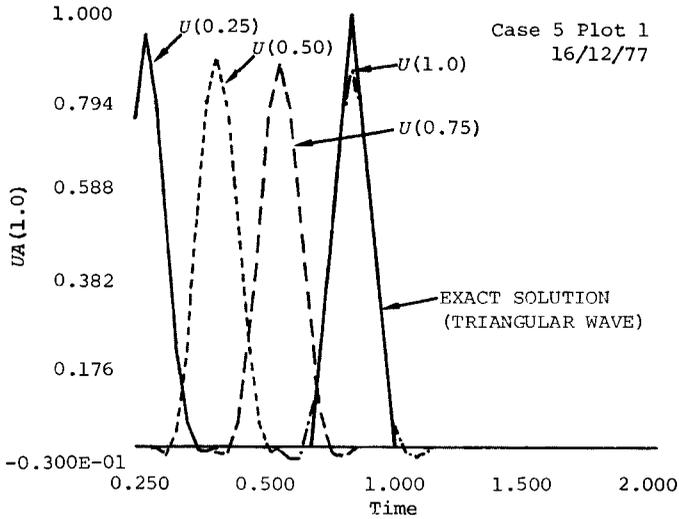


5c Progression of a triangular wave  
lin finite element ALFP = 0



5d Progression of a triangular wave  
lin finite element ALFP = 0.08

Figure 5 - Time response of selected schemes to a triangular wave. Plots are values of  $\underline{U}$  at various stations  $x_i$



5e Progression of a triangular wave  
U.W.R. ETA = 0.4

Figure 5 - Time responses of selected schemes to a triangular wave. Plots are values of  $U$  at various stations  $x_i$

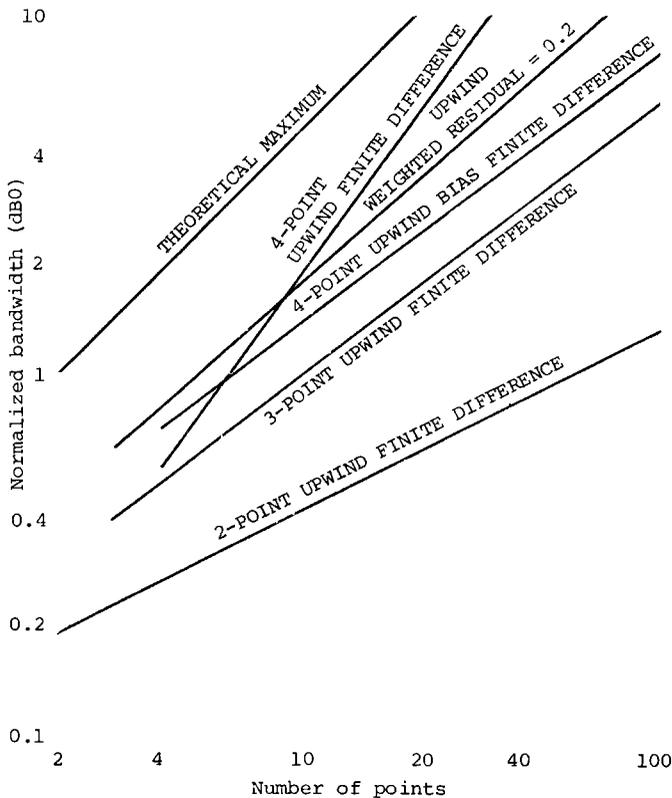


Figure 6 - Bandwidth as a function of discretization framework

is about a third of this value. The required bandwidth for a simulation depends on the nature of the signal being transmitted. Smooth signals may be simulated adequately using a low bandwidth scheme, but signals containing step changes have high bandwidth requirements.

## 9 CONCLUDING REMARKS

Any scheme which is to be stable must have dissipation built in by upwind weighting or included as an additional term. In the latter case the amount of dissipation added must be compatible with the convective term.

The optimum order is cubic, since good accuracy is achieved efficiently. Higher-order methods manipulate matrices denser than tridiagonal, requiring excessive time without greatly improving accuracy.

The Upwind Weighted Residual method is more robust than the linear finite-element method with added dissipation because of the weighting in the mass matrix. However, one can implement the latter much more readily by using a cubic spline in a quasi-Lagrangian sense, and this may be attractive if the appropriate weighting factor is carefully established.

The Hermite 3,1 formula (Equation 17) is also effective if one can realistically approximate  $(\partial u / \partial x)_0$ . Finally, for many applications, the four-point upwind bias formula (Equation 14) will behave quite adequately, since it is fast, stable, extremely easy to implement, and is not subject to the uncertainties of artificial dissipation.

While the results here pertain directly to the advective equation, the properties which this equation has in common with hyperbolic equation sets make it a useful test case. We believe that the conclusions of this study can be extrapolated in a qualitative manner and we plan further studies involving hyperbolic equations.

## APPENDIX A

### Upwind weighted residuals

Choose the approximation

$$U = \sum_{j=0}^1 \phi_j u_j \quad (A1)$$

and linear chapeau-shape function

$$\phi = [(1 - x/h), x/h] \quad (A2)$$

Thus

$$U = (1 - x/h)u_0 + x/h u_1, \quad x \in (x_0, x_1) \quad (A3)$$

Standard Galerkin uses weight functions  $\phi_i$  or  $W_i$  orthogonal to  $\phi_j$ . Applying this to

$$\frac{\partial u}{\partial t} + v \frac{\partial u}{\partial x} = 0 \quad (A4)$$

we have

$$\int_0^L \left[ \phi_i \left( \sum_{j=0}^2 \phi_j \frac{\partial u}{\partial t} \right) + \sum_{j=0}^2 \left( v \phi_i \frac{\partial \phi_j}{\partial x} u_j \right) \right] dx = 0 \quad (A5)$$

where only  $j=0,2$  is required as one of the orthogonal functions is zero elsewhere.

To achieve upwind weighting as shown in the figure, let us instead define

$$W_i = \phi_i \pm \theta_i$$

such that  $W_i = 0, 1$   
at  $x = 0, h$ .

We chose

$$\theta_i = 3\varepsilon x(x-h)/h^2$$

$\varepsilon$  is a parameter  
(0, 1).

Thus (A5) becomes

$$\int_0^L W_i \left( \sum_{j=0}^2 \phi_j \frac{\partial u}{\partial t_j} \right) + v \sum_{j=0}^2 W_i \frac{\partial \phi_j}{\partial x} u_j \Big| dx = 0 \quad (A6)$$

(A6) must be solved term by term for clarity. The first term is

$$\begin{aligned} & \int_0^h \left( W_1 \phi_0 \frac{\partial u}{\partial t_0} + W_1 \phi_1 \frac{\partial u}{\partial t_1} \right) dx + \frac{2h}{h} \left( W_1 \phi_1 \frac{\partial u_1}{\partial t} + W_1 \phi_2 \frac{\partial u_2}{\partial t} \right) dz, \\ & \qquad \qquad \qquad z = x - h \\ & = \int_0^h \left[ \left( \frac{x}{h} - \frac{3\varepsilon x(x-h)}{h^2} \right) \left( 1 - \frac{x}{h} \right) u_0' \right. \\ & \quad \left. + \left( \frac{x}{h} - \frac{3\varepsilon x(x-h)}{h^2} \right) \left( \frac{x}{h} \right) u_1' \right] dx + \text{etc.} \end{aligned}$$

which gives

$$\begin{aligned} & \frac{h}{6} \left[ \left( 1 + \frac{3\varepsilon}{2} \right) u_0' + 4u_1' + \left( 1 - \frac{3\varepsilon}{2} \right) u_2' \right] \\ & - \frac{\varepsilon}{z} \left[ -(1+\varepsilon)u_0 + 2\varepsilon u_1 + (1+\varepsilon)u_2 \right] \quad (A7) \end{aligned}$$

Repeating the analysis for end elements one obtains the matrix

$$\begin{aligned} & \frac{h}{6} \begin{bmatrix} 2 - \frac{3\varepsilon}{2} & 1 - \frac{3\varepsilon}{2} & 0 & 0 \\ 1 + \frac{3\varepsilon}{2} & 4 & 1 - \frac{3\varepsilon}{2} & 0 \\ 0 & 1 + \frac{3\varepsilon}{2} & 4 & 1 - \frac{3\varepsilon}{2} \\ 0 & 0 & 1 + \frac{3\varepsilon}{2} & 2 + \frac{3\varepsilon}{2} \end{bmatrix} U' \\ & = -\frac{v}{2} \begin{bmatrix} -(1-\varepsilon) & (1-\varepsilon) & 0 & 0 \\ -(1+\varepsilon) & 2\varepsilon & (1-\varepsilon) & 0 \\ 0 & -(1+\varepsilon) & 2\varepsilon & (1-\varepsilon) \\ 0 & 0 & -(1+\varepsilon) & (1+\varepsilon) \end{bmatrix} \quad (A8) \end{aligned}$$

Note that for null upwind weighting, i.e.,  $\varepsilon = 0$ , (A8) reduces to the standard Galerkin formulation.

## APPENDIX B

### Equivalence of linear finite-element and finite-difference cubic spline

Consider the advective equation with artificial dissipation

$$\frac{\partial u}{\partial t} = -v \frac{\partial u}{\partial x} + \frac{\partial^2 u}{\partial x^2} \quad (B1)$$

The linear finite-element approximation gives

$$\begin{aligned} & \frac{1}{6} \left( \frac{du_{i+1}}{dt} + 4 \frac{du_i}{dt} + \frac{du_{i-1}}{dt} \right) \\ & = -v \frac{u_{i+1} - u_{i-1}}{2h} + \alpha \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} \quad (B2) \end{aligned}$$

and is solved using

$$\underline{\underline{M}} \dot{\underline{U}} = (\underline{\underline{K}}_1 + \underline{\underline{K}}_2) \underline{U} \quad (B3)$$

or

$$\dot{\underline{U}} = \underline{\underline{M}}^{-1} (\underline{\underline{K}}_1 + \underline{\underline{K}}_2) \underline{U}$$

where the elements of

$$\begin{aligned} \underline{\underline{M}} & \text{ are } \frac{1}{6} \begin{bmatrix} 1 & 4 & 1 \end{bmatrix} \\ \underline{\underline{K}}_1 & \text{ are } \frac{-v}{2\Delta x} \begin{bmatrix} -1 & 0 & 1 \end{bmatrix} \\ \underline{\underline{K}}_2 & \text{ are } \frac{\alpha}{\Delta x^2} \begin{bmatrix} 1 & -2 & 1 \end{bmatrix} \end{aligned}$$

The cubic spline approximation applied to the right-hand side of (B2) defines the spatial derivatives  $U'$  and  $U''$  by (18) and (19) as follows:

$$\frac{u_{i+1}' + 4u_i' + u_{i-1}'}{6} = \frac{u_{i+1} - u_{i-1}}{2h}$$

$$\frac{u_{i+1}'' + 4u_i'' + u_{i-1}''}{6} = \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2}$$

$$\text{i.e., } \underline{\underline{M}} \underline{U}' = \frac{-1}{v} \underline{\underline{K}}_1 \underline{U}$$

$$\underline{\underline{M}} \underline{U}'' = \frac{1}{\alpha} \underline{\underline{K}}_2 \underline{U}$$

$$\text{or } \underline{U}' = \underline{\underline{M}}^{-1} \frac{-1}{v} \underline{\underline{K}}_1 \underline{U}$$

$$\underline{U}'' = \underline{\underline{M}}^{-1} \frac{1}{\alpha} \underline{\underline{K}}_2 \underline{U}$$

Inserting these now into the right-hand side of (21) in the finite-difference manner, we have

$$\begin{aligned} \dot{\underline{U}} & = -v \underline{U}' + \alpha \underline{U}'' \\ & = -v \underline{\underline{M}}^{-1} \frac{-1}{v} \underline{\underline{K}}_1 \underline{U} + \alpha \underline{\underline{M}}^{-1} \frac{1}{\alpha} \underline{\underline{K}}_2 \underline{U} \\ & = \underline{\underline{M}}^{-1} (\underline{\underline{K}}_1 + \underline{\underline{K}}_2) \underline{U} \end{aligned}$$

as in (B3).

## REFERENCES

- 1 BICKLEY, W.G.  
*Formulae for Numerical Differentiation*  
*Mathematical Gazette* vol. 25 1941
- 2 CARVER, M.B.  
*The FORSIM System for Automated Solution of Sets of Implicitly Coupled Partial Differential Equations*  
*Annales de l'Association du Computation Analogique* vol. 3 1975 pp. 195-202
- 3 CARVER, M.B.  
*The Choice of Algorithms in Automated Method of Lines Solution of Partial Differential Equations*  
In L. Lapidus and W.E. Schiesser, editors,  
*Numerical Methods for Differential Systems*  
(Academic Press, New York, 1976), pp. 243-265
- 4 CARVER, M.B. STEWART, D.G. BLAIR, J.M. SELANDER, W.N.  
*The FORSIM VI Package for Automated Solution of Ordinary and/or Partial Differential Equations*  
Atomic Energy of Canada Report AECL-5821  
February 1978
- 5 CHRISTIE, I. GRIFFITHS, D.F. MITCHELL, A.R. ZIENCIEWICZ, O.C.  
*Finite Element Methods for Second Order Differential Equations with Significant First Derivatives*  
*International Journal of Numerical Methods in Engineering* vol. 10 1976 pp. 1389-1396
- 6 FISCHER, K.  
*Sub Grid Numerical Representations and Their Improvement of Convective Difference Schemes*  
In R. Vichnevetsky, editor, *Advances in Computer Methods for Partial Differential Equations II*  
(IMACS Press, 1977)
- 7 GARY, J.  
*The Methods of Lines Applied to a Simple Hyperbolic Equation*  
*Journal of Computational Physics* vol. 22 1976 pp. 131-149
- 8 HEYDWEILLER, J.C. SINCOVEC, R.F.  
*A Stable Difference Scheme for the Solution of Hyperbolic Equations Using the Method of Lines*  
*Journal of Computational Physics* vol. 22 1976 pp. 377-388
- 9 LAWRENCE, C.B. PEARSON, A.  
*Measurement Techniques Using a Pseudo-Random Binary Sequence and Fast Fourier Transformation to Determine a System Transfer Function*  
Atomic Energy of Canada Report AECL-3601 1970
- 10 LOEB, M.A. SCHIESSER, W.E.  
*Stiffness and Accuracy in Method of Lines Integration of Partial Differential Equations*  
*Proceedings 1973 Summer Computer Simulation Conference* AFIPS Press pp. 25-39
- 11 LONG, P.E. PEPPER, D.W.  
*A Comparison of Six Numerical Schemes for Calculating the Advection of Atmospheric Pollution*  
*Proceedings Third Symposium on Atmospheric Turbulence, Diffusion and Air Quality* Raleigh, North Carolina October 1976 pp. 181-187
- 12 MADSEN, N.K. SINCOVEC, R.F.  
*PDECOL: Generalized Software for Partial Differential Equations*  
In L. Lapidus and W.E. Schiesser, editors,  
*Numerical Methods for Differential Systems*  
(Academic Press, New York, 1976), pp. 229-242
- 13 PURNELL, D.K.  
*Solution of the Advective Equation by Upstream Interpolation of a Cubic Spline*  
*Monthly Weather Review* vol. 104 January 1976 pp. 42-48
- 14 RICHTMEYER, R.D. MORTON, K.W.  
*Difference Methods for Initial Value Problems*  
Interscience New York 1967
- 15 SCHIESSER, W.E.  
*LEANS: a Digital Simulation System for Mixed Ordinary/Partial Differential Equation Modes*  
*Proceedings IFAC Symposium on Digital Simulation of Continuous Systems* vol. 2 Gyor, Hungary 1971 pp. S2-1 to S2-9
- 16 STRANG, G. FIX, G.J.  
*An Analysis of the Finite Element Method*  
Prentice-Hall Englewood Cliffs, New Jersey 1973
- 17 VICHNEVETSKY, R. TOMALISKY, A.W.  
*Spurious Wave Phenomena in Numerical Approximations of Hyperbolic Equations*  
*Proceedings Fifth Annual Conference on Information Science and Systems* March 1971 pp. 357-363
- 18 VICHNEVETSKY, R. *et al.*  
*Mean Square Measure of the Effects of Artificial Viscosity*  
In R. Vichnevetsky, editor, *Advances in Computer Methods for Partial Differential Equations II*  
(IMACS Press, 1977)

**ERRATA** for The method of Lines and the Advective Equation, Carver & Hinds Simulation, 1978

Equation 13 should be:

$$(d'u/d'x)_i = (3u_i - 4u_{i-1} + u_{i-2})/2\Delta x$$

Equation 14 should be:

$$(d'u/d'x)_i = (u_{i-2} - 6u_{i-1} + 3u_i + 2u_{i+1})/6\Delta x$$

These are correct in later papers.

M.B. Carver