DEPARTMENT OF MATHEMATICS

Error Measurements for Semi-Lagrangian Schemes

C.J.Smith

Numerical Analysis Report **3/93**

UNIVERSITY OF READING

Abstract

The semi-Lagrangian method is widely used in numerical weather models. The properties of the numerical solutions obtained by this method, depend strongly on the form of spatial interpolation used. In this report, several commonly used interpolants are reviewed and Fourier analysis is applied to the resulting schemes. Error measurements for the advection of more general data are then established, which build on the results of the Fourier analysis.

Contents

Introduction

This report has two aims. The first is to describe the semi-Lagrangian method. This is an approach to the generation of finite difference schemes for problems involving advection. The importance of this method lies in the fact that it allows the time step to be chosen from accuracy considerations alone; the scheme being unconditionally stable. For this reason it is particularly suitable for weather modelling. In particular semi-Lagrangian advection has been combined with an implicit treatment of sound and gravity wave terms. This results in schemes whose time step far exceeds the stability limit for conventional explicit methods, [8], [9].

An important part of any semi-Lagrangian scheme is the spatial interpolation of the finite difference data. Not only does the choice of interpolant determine the spatial accuracy of the scheme, it also affects certain physically important properties of the solution. These are properties such as monotonicity, or more generally positivity, and conservation of mass. This leads to the second part of this report, which involves assessing the suitability of various interpolants, for use in semi-Lagrangian schemes.

As a first step in finding a means of comparing different interpolants, a review of the standard technique of Fourier analysis is carried out. This provides values for amplitude attenuation and phase error for harmonic waves. The next step builds on these results by considering the advection of periodic waves of arbitrary shape. Such a wave may be decomposed into its component harmonics. The expansion coefficients, used in this decomposition, provide a means of weighting the amplitude and phase errors associated with each harmonic. These weights are then used in the averaging of the errors over all the component harmonics.

The methods described above are applied to a specific class of explicit finite difference schemes. Several interpolants of practical importance fall in to this class, and results are given here for some of these. A separate analysis is made for cubic spline interpolation, which is not a member of this class. Results obtained

for some standard schemes are given for comparison.

1 The Semi-Lagrangian Method

The semi-Lagrangian method is a finite difference technique for the numerical modelling of advection, [8]. It achieves the redistribution, of a transported quantity, by solving local fluid trajectories over each time step. First it is necessary to consider the Lagrangian description of fluid flow. Following this, an overview of the two time-level semi-Lagrangian method is given.

1.1 Lagrangian Description

Let $u(x,t)$ be some property of the fluid and let $a(x,t)$ be the velocity field of the fluid. Passive transport of the quantity u is described by the one dimensional advection equation :

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = 0. \tag{1.1}$$

This equation may be written in terms of the Lagrangian (or convective) derivative,

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + a\frac{\partial}{\partial x},$$

when it becomes

$$\frac{Du}{Dt} = 0. \tag{1.2}$$

The Lagrangian derivative (1.2) provides the time rate of change, of the value of u, associated with any one particular fluid element. Such a fluid element is uniquely identified by specifying its position, x, at some reference time, t_{ref}. If $y(x, t_{ref}; t)$ is the position of this fluid element at time t, then the trajectory of the element is given by the equation,

$$\frac{dy}{dt} = a(y,t). \tag{1.3}$$

1.2 Two Time-Level Scheme

The finite difference solution of (1.2) is to be established on an Eulerian mesh with uniform space and time intervals, $\triangle x$ and $\triangle t$ respectively. Grid nodes are

at the positions $x_j = j\triangle x$ and the solution is to be advanced through successive time intervals, $t_n = n\triangle t$ where $n = 0, 1, 2, \cdots$.

Let $U(x,t)$ be an approximate solution of (1.2) obtained numerically. If U is known up to the time level t_n, then it is advanced to the next time level using a discretization of (1.2). For a two time-level scheme [1] this discretization is

$$\frac{U(x_j, t_{n+1}) - U(y(x_j, t_{n+1}; t_n), t_n)}{\triangle t} = 0. \tag{1.4}$$

where $y(x_j, t_{n+1}; t_n)$ is the position of the fluid trajectory at time t_n, described below. So at time t_{n+1} the value of U at the grid node x_j is given by

$$U(x_j, t_{n+1}) = U(y(x_j, t_{n+1}; t_n), t_n). \tag{1.5}$$

This important equation is the basic statement of the semi-Lagrangian method.

Equation (1.2) states that, for any chosen fluid element, the value of u does not change with time. Equation (1.5) is the discrete equivalent of this : the value of U held by the fluid element at (x_j, t_{n+1}) is exactly that which it held at the earlier time t_n. In order to find this value of U, it is first necessary to discover the position, $y(x_j, t_{n+1}; t_n)$, of the particle at time t_n. This position is known as the upstream departure point. It is found by performing an integration of the trajectory equation (1.3).

The simplest choice of integration is the Euler method. For this particular problem it takes the form,

$$\frac{y(x_j, t_{n+1}; t_{n+1}) - y(x_j, t_{n+1}; t_n)}{\triangle t} = a(y(x_j, t_{n+1}; t_{n+1}), t_n). \tag{1.6}$$

Using the identity,

$$y(x_j, t_{n+1}; t_{n+1}) = x_j,$$

equation (1.6) results in the displacement formula :

$$y(x_j, t_{n+1}; t_n) = x_j - \triangle t \; a(x_j, t_n).$$

In practice the Euler method is considered too inaccurate, but ordinary differential equation solvers of higher order can be applied to (1.3) in an entirely analogous way, [8].

The upstream departure point, once found, will generally not coincide with a grid node. Yet it is only at the nodes that a finite difference solution exists. This raises the need for interpolation to evaluate $U(y, t_n)$.

1.3 Interpolation

A finite difference method provides values of the approximate solution $U(x,t)$ only at grid nodes, x_j. To obtain a value between nodes, any form of interpolation might be used. If a polynomial interpolant is chosen, then the scheme falls into a class of schemes particularly well suited to analysis. These are the explicit polynomial schemes considered in the next section of this report. First a number of interpolation options of practical importance are considered here.

1.3.1 Lagrange Interpolation

This is the simplest form of polynomial interpolation. A small subset of grid nodes is selected, which surround the point at which an interpolated value is required.

Assume that the upstream departure point, y, lies between nodes $k-1$ and k. That is

$$x_{k-1} \leq y \leq x_k.$$

Take a sample set of p nodes either side of the interval $[x_{k-1}, x_k]$:

$$\mathbf{S}_p = \{x_{k-p}, x_{k-p+1}, \cdots, x_{k+p-1}\}.$$

Let $\mathcal{L}_{2p-1}$ be the unique polynomial of degree $2p-1$ which interpolates the finite difference solution at all the points in $\mathbf{S}_p$. Let $u_j = U(x_j, t_n)$ and $u^j = U(x_j, t_{n+1})$, then

$$\mathcal{L}_{2p-1}(x_{k-p}, \cdots, x_{k+p-1}, u_{k-p}, \cdots, u_{k+p-1}; y) = \sum_{r=-p}^{p-1} u_{k+r} L_r$$

where

$$L_r(x_{k-p}, \cdots, x_{k+p-1}; y) = \prod_{\substack{s=-p \\ s \neq r}}^{p-1} \left(\frac{x_{k+s} - y}{x_{k+s} - x_{k+r}} \right). \tag{1.7}$$

For a constant velocity field, the velocity is simply displacement over time,

$$a = \frac{x_j - y}{\triangle t}. \tag{1.8}$$

This provides the following useful formula for the Courant number :

$$\nu \equiv a\frac{\triangle t}{\triangle x} = \frac{x_j - y}{\triangle x}.$$

The set $\mathbf{S}_p$ is centred on the interval $[x_{k-1}, x_k]$, which contains the departure point y. As a consequence y always lies within one grid interval, $\triangle x$, of x_k. Now equation (1.7) indicates that the interpolation is built up of differences between x_k (and surrounding nodes) and y. So if the Courant number is restricted to lie between 0 and 1, then there is no loss of generality putting $k = j$. A further simplification of the equations results by remembering that $x_j = j\triangle x$. Redefining the polynomial in terms of the Courant number, it may now be written

$$U(y, t_n) = \mathcal{L}_{2p-1}(\nu) \equiv \sum_{r=-p}^{p-1} \left\{ u_{j+r} \prod_{\substack{s=-p \\ s\neq r}}^{p-1} \left(\frac{s+\nu}{s-r} \right) \right\}$$

where $0 \leq \nu \leq 1$.

Only polynomials of degree $2p-1$ have been considered. Even degree polynomials are constructed in the same way. However it is not possible to centre the sample set about the upstream departure point : an even degree polynomial requires an odd number of data points. Stable schemes can still be constructed by choosing the sample set correctly. There are two possibilities of sample set for a second degree polynomial. Both lead to stable schemes. These are the unconditionally stable equivalents of the Lax-Wendroff [5] and Warming and Beam [10] schemes. However not all of the possibilities for a fourth degree polynomial give a stable scheme.

1.3.2 Hermite Cubic Interpolation

This form of interpolation is very similar to the Lagrange form. Again the interpolation is done on a small number of nodes surrounding the departure point. A

Hermite polynomial requires two values at each node : the values of the interpolated function and its first derivative. However, the finite difference scheme does not provide values of the first derivative. These must be obtained from the data using a suitable approximation method. This freedom in choosing an estimate for the first derivative allows for great flexibility in designing a semi-Lagrangian scheme. Also limiters can be applied to derivative estimates to obtain desirable properties in the solution, such as monotonicity [7].

Let d_j be an estimate for $\dfrac{\partial U(x_j, t_n)}{\partial x}$. In general such a derivative estimate will be obtained in terms of the discrete slopes :

$$\triangle_{j-1/2} = \frac{u_j - u_{j-1}}{\triangle x}.$$

With these definitions the semi-Lagrangian scheme, with Hermite cubic interpolation, may be written

$$\begin{aligned} U(y, t_n) = \mathcal{H}_3(\nu) \equiv \quad & \triangle x \ (2\triangle_{j-1/2} - d_{j-1} - d_j) \ \nu^3 \\ + \quad & \triangle x \ (d_{j-1} + 2d_j - 3\triangle_{j-1/2}) \ \nu^2 \\ - \quad & \triangle x \ d_j \ \nu + u_j \end{aligned}$$

1.3.3 Derivative Estimates

There are many forms of derivative estimate which might be used in the Hermite cubic interpolation. A few are listed here.

- Arithmetic Mean : $d_j = \dfrac{\triangle_{j-1/2} + \triangle_{j+1/2}}{2}$
- Hyman [3] : $d_j = \dfrac{-\triangle_{j-3/2} + 7\triangle_{j-1/2} + 7\triangle_{j+1/2} - \triangle_{j+3/2}}{12}$
- Priestley [6] : $d_j = \dfrac{-3\triangle_{j-3/2} + 19\triangle_{j-1/2} + 19\triangle_{j+1/2} - 3\triangle_{j+3/2}}{32}$

1.3.4 Cubic Spline Interpolation

This interpolation method does not share the same structure as the Lagrange and Hermite methods. A spline does not depend only on the data near the departure point, but on all the data throughout the region.

The cubic spline is formed using approximations, c_j, for the second derivative at grid points [4] :

$$c_j \approx \frac{\partial^2 U}{\partial x^2}(x_j, t_n).$$

These derivative estimates are related to the finite difference solution by the formula,

$$\left(1 + \frac{1}{6}\delta^2\right) c_j = \frac{1}{\triangle x^2}\ \delta^2 u_j, \tag{1.9}$$

where $\delta^2 u_j = u_{j+1} - 2u_j + u_{j-1}$.

The cubic spline polynomial is

$$\begin{aligned} U(y, t_n) = \mathcal{S}_3(\nu) \equiv\ & -\triangle x^2\ \frac{(c_j - c_{j-1})}{6}\ \nu^3 + \triangle x^2\ \frac{c_j}{2}\ \nu^2 \\ & -\triangle x^2 \left[\triangle_{j-1/2} + \frac{1}{6}(2c_j + c_{j-1})\right] \nu + u_j \end{aligned} \tag{1.10}$$

1.3.5 Fourier Analysis of the Cubic Spline

As an example of the type of analysis used in this report, Fourier transforms are applied to equations (1.9) and (1.10). Equation (1.9) involves the step operator E. The effect of the Fourier transform on this is to introduce the multiplicative factor $e^{-i\phi}$, where $\phi = k\triangle x$ and k is the independent variable of the transform.

Let $\tilde{U}$ and $\tilde{C}$ be the Fourier transforms of U and C, which are related by

$$\left(1 + \frac{1}{6}\delta^2\right) C(x, t_n) = \frac{1}{\triangle x^2}\ \delta^2 U(x, t_n).$$

The transform of this equation is

$$\begin{aligned} \tilde{C}\triangle x^2 &= \lambda(\phi)\tilde{U} \\ \text{where} \quad \lambda(\phi) &= 6\left(1 - \frac{3}{2 + \cos\phi}\right). \end{aligned}$$

A transform equation may be obtained for (1.10) in a similar way :

$$\begin{aligned} \tilde{U}(k, t_{n+1}) &= g(\phi, \nu)\tilde{U}(k, t_n) \\ \text{where} \quad g(\phi, \nu) &= -\frac{\lambda(\phi)}{6}(1 - e^{-i\phi})\nu^3 + \frac{\lambda(\phi)}{2}\nu^2 \\ & \quad - \left[1 - e^{-i\phi} + \frac{\lambda(\phi)}{6}(2 + e^{-i\phi})\right] \nu + 1 \end{aligned}$$

The quantity $g(\phi, \nu)$ is known as the amplification factor for the scheme. In effect it is the response of the scheme to a single Fourier mode. This form of analysis will be presented in the next section for a large class of finite difference schemes. These schemes have a particularly simple polynomial form, and include many of the semi-Lagrangian schemes of interest here.

2 Analysis of Finite Difference Schemes for Advection

2.1 Test Problem and its Discretization

In order to analyse the performance of a numerical solution scheme, a suitable test problem is required. This is usually taken to be the simplest differential equation whose solutions exhibit the process which is of interest. In the case of advection the following test problem is chosen :

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = 0 \tag{2.11}$$

where $u = u(x,t)$ and $a \geq 0$,

on the domain $\mathbf{D}$: $\begin{cases} -\infty < x < \infty \\ t \geq 0. \end{cases}$

The choice of initial condition is dictated by the form of the analysis which is to be carried out. The simplest possibility is considered in section 2.4 and slightly more realistic data will be considered in section 2.5.

To establish a finite difference approximation for this problem, the domain must first be discretized. The following mesh, which is regular in both space and time, is used :

$$\mathbf{M}(\triangle x, \triangle t) = \{(j\triangle x, n\triangle t) : j \in \mathbf{Z}, n \in \mathbf{N}\},$$

where $\triangle x$ and $\triangle t$ are the space and time grid intervals respectively. Now let

$$x_j = j\triangle x \quad \text{and} \quad t_n = n\triangle t, \tag{2.12}$$

and let u_j^n be the finite difference solution at the mesh point (x_j, t_n). That is,

$$u_j^n \approx u(x_j, t_n). \tag{2.13}$$

The time evolution of the finite difference solution is governed by a numerical scheme, which is obtained from some consistent discretization of the equation (2.11). One class of schemes will be considered.

2.2 Explicit Schemes of Polynomial Form

The schemes of this class have the general form,

$$u_j^{n+1} = \mathcal{C}(E, \nu)\ u_j^n \tag{2.14}$$

where

- $\mathcal{C}$ is a polynomial in both arguments,
- E is the space index step operator : $Eu_j = u_{j+1}$,
- and ν is the Courant number : $\nu = a\dfrac{\triangle x}{\triangle t}$.

This is an explicit, two time-level scheme : the solution at the next time level is obtained from data held at the current time level alone.

The polynomial, $\mathcal{C}$, has the general form

$$\begin{aligned} \mathcal{C}(E, \nu) &= \sum_r c_r(\nu)\ E^r \\ \text{where} \quad c_r(\nu) &= \sum_s \alpha_{rs}\ \nu^s \end{aligned} \tag{2.15}$$

and α_{rs} are constant values. In practice both summations will consist of a finite, and usually quite small, number of terms.

2.3 Evolution Operators

Evolution operators exist for both the exact and finite difference solutions. These operators advance their respective solutions, at any point (x_j, t_n), through one time step. The evolution operator of the exact solution is obtained by considering the analytical solution of (2.11),

$$u(x, t) = u(x - at, 0). \tag{2.16}$$

Applying this to the exact solution on the mesh $\mathbf{M}$,

$$\begin{aligned} u(x_j, t_n + \triangle t) &= u(x_j - at_n - a\triangle t, 0) \\ &= u(x_j - at_n - \nu\triangle x, 0) \end{aligned}$$

where the last line follows from $\nu = a\dfrac{\triangle t}{\triangle x}$. Using the space step operator E, the above may be written,

$$\begin{aligned} u(x_j - at_n - \nu\triangle x, 0) &= E^{-\nu}u(x_j - at_n, 0) \\ &= E^{-\nu}u(x_j, t_n) \end{aligned} \tag{2.17}$$

where the final line is obtained from the analytical solution (2.16). Hence the evolution equation for the exact solution of (2.11) is

$$u(x_j, t_n + \triangle t) = E^{-\nu}u(x_j, t_n). \tag{2.18}$$

This equation identifies the evolution operator for the exact solution, on the mesh $\mathbf{M}$: $E^{-\nu}$. It corresponds to linear wave motion; during one time step, it shifts the solution through a distance determined by the mesh velocity ν. The evolution operator for the finite difference solution is simply $\mathcal{C}(E, \nu)$, since (2.14) is the numerical analogue of (2.18).

The effectiveness of the finite difference scheme rests on how well its evolution operator mimics that of the exact solution. A well known technique for investigating this is to compare the effect each evolution operator has on a single harmonic wave.

2.4 Fourier Analysis

Consider the case where the differential equation (2.11) and its discretization (2.14) are supplied with initial data

$$\begin{aligned} u(x, 0) &= e^{ikx} \\ u_j^0 &= e^{ikx_j}, \end{aligned} \tag{2.19}$$

where $k \in \mathbb{R}$ is the wave number of this particular Fourier mode.

It is the ratio of the grid interval $\triangle x$ to the wavelength $2\pi/k$ which determines how well the wave is resolved on the grid. Consequently the performance of any numerical scheme must depend on this ratio. For this reason it is convenient to define the mesh dependent wave number,

$$\phi = k\triangle x.$$

For the above initial data the exact solution of (2.11) is a sinusoidal progressive wave. Such a wave is precisely specified by just two parameters (the wavenumber k and the wave velocity a) and takes the form

$$u(x,t) = e^{ik(x-at)}.$$

If this solution is restricted to the mesh $\mathbf{M}(\triangle x, \triangle t)$, then the relevant parameters are the mesh dependent counterparts of k and a, that is ϕ and ν :

$$u(x_j, t_n) \;\; = \;\; e^{i\phi(j-\nu n)}. \tag{2.20}$$

The space and time parts of this solution are separable allowing the solution to be written as

$$u(x_j, t_n) = (e^{-i\nu\phi})^n \; e^{ij\phi}. \tag{2.21}$$

The quantity $e^{-i\nu\phi}$ is the amplification factor for the exact solution. It is obtained by the action of the exact evolution operator, $E^{-\nu}$, on the Fourier wave $e^{i\phi}$:

$$E^{-\nu} e^{i\phi} = e^{i(1-\nu)\phi} = e^{-i\nu\phi} \; e^{i\phi}.$$

The parameters ϕ and ν also determine the finite difference solution. They enter the solution through the action of the evolution operator, $\mathcal{C}(E,\nu)$ as follows. Using (2.14) and (2.19) the solution after the first time step is

$$u_j^1 = \mathcal{C}(E,\nu) \; e^{ij\phi}.$$

This may be expanded using (2.15),

$$\begin{aligned} \mathcal{C}(E,\nu) \; e^{ij\phi} \;\; &= \;\; \sum_r c_r(\nu) \; E^r \; e^{ij\phi} \\ &= \;\; \sum_r c_r(\nu) \; e^{i(j+r)\phi} \\ &= \;\; e^{ij\phi} \sum_r c_r(\nu) \; e^{ir\phi}. \end{aligned}$$

The summation in the last line may be written in terms of the polynomial $\mathcal{C}$, which leads to

$$u_j^1 = e^{ij\phi} \; \mathcal{C}(e^{i\phi}, \nu).$$

The coefficient, $\mathcal{C}(e^{i\phi}, \nu)$, is independent of the space index and will be unaffected by subsequent applications of the evolution operator. Hence the solution of the difference scheme may be written as

$$u_j^n = [\mathcal{C}(e^{i\phi}, \nu)]^n \; e^{ij\phi}. \tag{2.22}$$

This result is the finite difference analogue of (2.21) and, just as for the exact solution, an amplification factor may be identified. The finite difference amplification factor plays the central rôle in the analysis which follows. It is a function of ϕ and ν, and is assigned the symbol $g(\phi, \nu)$, where

$$g(\phi, \nu) = \mathcal{C}(e^{i\phi}, \nu). \tag{2.23}$$

Equations (2.21) and (2.22) indicate that any error in the finite difference solution must arise through discrepancies between the exact and finite difference amplification factors. One such discrepancy is immediately apparent. The finite difference amplification factor is 2π periodic in ϕ, since it is a polynomial in $e^{i\phi}$. In general $e^{-i\nu\phi}$ does not share the same periodicity.

Both amplification factors are complex, producing changes in the amplitude and phase of the wave solutions over one time step. For this reason the analysis is split into two parts; comparisons are made between the moduli and arguments of the amplification factors. This leads to von Neumann stability and spectral analysis.

2.4.1 Von Neumann Stability

For this form of stability, the modulus of the finite difference amplification factor is required not to exceed that of the exact amplification factor. Define

$$\begin{aligned} \varepsilon_a(\phi, \nu) &= \frac{|g(\phi, \nu)|}{|e^{-i\phi\nu}|} \\ &= |g(\phi, \nu)|\,. \end{aligned} \tag{2.24}$$

The stability requirement now reads $\varepsilon_a \leq 1$.

2.4.2 Spectral Analysis

Phase error in the finite difference solution is a result of error in the argument of $g(\phi, \nu)$. Define

$$\varepsilon_p(\phi, \nu) = \frac{\arg[g(\phi, \nu)]}{\arg(e^{-i\phi\nu})}.$$

Since $g(\phi, \nu)$ is 2π periodic in ϕ, it suffices to restrict ϕ to the principal values $-\pi < \phi \leq \pi$. It is now possible to write the ratio of the finite difference and exact phases in a more useful form,

$$\varepsilon_p(\phi, \nu) = -\frac{1}{\nu\phi}\arg[g(\phi, \nu)] \tag{2.25}$$

where $\arg(z)$ represents the argument of the complex number z.

With the above definition :

$\varepsilon_p < 1$ corresponds to phase lag in the finite difference solution,

$\varepsilon_p > 1$ corresponds to phase advance.

First Order Upwind The first order upwind scheme

$$u_j^{n+1} = \nu u_{j-1}^n + (1-\nu)u_j^n \quad , \quad \nu \geq 0$$

may be written in the form (2.14) :

$$u_j^{n+1} = [\nu E^{-1} + (1-\nu)E^0]u_j^n \equiv \mathcal{C}(E, \nu)u_j^n.$$

This gives

$$g(\phi, \nu) \equiv \mathcal{C}(e^{i\phi}, \nu) = \nu e^{-i\phi} + (1-\nu).$$

Simple calculations then provide

$$\begin{aligned} \varepsilon_a(\phi, \nu) &= [1 - 2\nu(1-\nu)(1-\cos\phi)]^{\frac{1}{2}}, \\ \varepsilon_p(\phi, \nu) &= \frac{1}{\nu\phi}\arctan\left\{\frac{\nu\sin\phi}{1-\nu(1-\cos\phi)}\right\} \end{aligned}$$

for $-\pi < \phi \leq \pi$.

2.4.3 Plots of Modulus and Argument Ratios

Treating $\varepsilon(\phi, \nu)$ and ϕ as polar coordinates, curves of $\{(\varepsilon(\phi), \phi) : -\pi \leq \phi \leq \pi\}$ may be plotted. This is done for both ε_a and ε_p, with a range of values for ν.

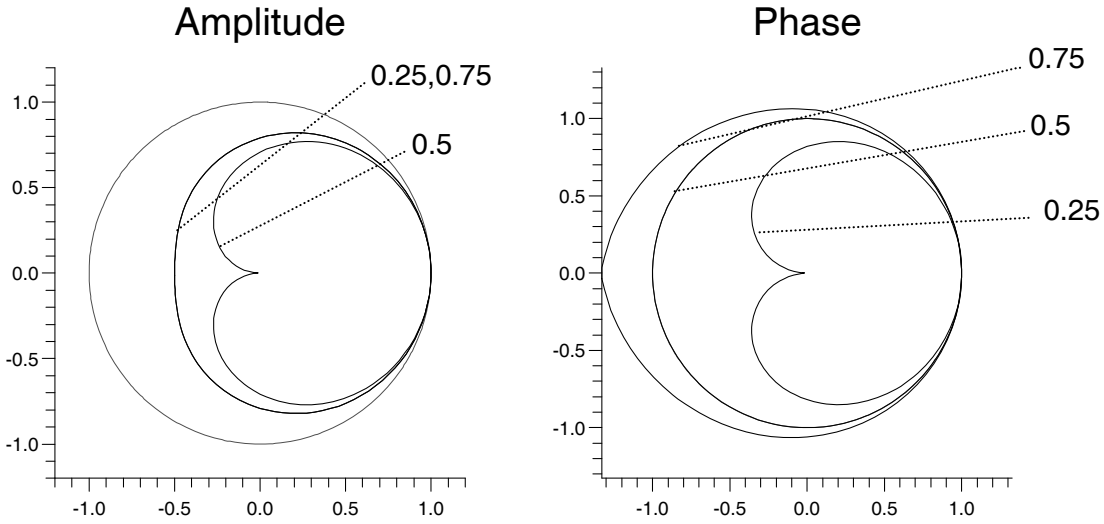

Figure 1: First Order Upwind Scheme

First Order Upwind The amplitude graph of *figure* 1 always remains within the unit circle, indicating that the scheme is stable. At $\phi = 0$, $\varepsilon_a = 1$ but its value rapidly drops away from the unit circle, as $|\phi|$ is increased. This accounts for the well known damping inherent in this scheme. However the extreme damping of the highest frequencies is a desirable property, since these will most often be non-physical.

It is important, in producing these plots, to ensure that $\arg(g)$ is not allowed to change discontinuously with respect to the exact phase. Sudden jumps in $\arg(g)$, relative to the exact phase, occur due to the periodicity of the arg function. Multiples of 2π are added/subtracted to $\arg(g)$ to ensure that it changes smoothly with respect to ϕ.

2.5 Arbitrary Periodic Waves

So far only waves composed of a single frequency have been considered. A more realistic approach is to consider the advection of a periodic wave, which contains a superposition of frequencies. (For instance the initial data may be chosen to be a square or triangular wave.) This allows the possibility of analysing such phenomena as dispersion and frequency dependent attenuation. However the approach which will be made here, is to look for measurements of amplitude and phase error averaged over all frequency components.

2.5.1 Fourier Expansion

At time t_n, let the data take the form

$$u_j^n = \sum_{r=-\infty}^{\infty} \alpha_r(t_n)\ e^{ij\phi_r} \tag{2.26}$$

where $\alpha_r \in \mathbf{C}$ are the Fourier weights and ϕ_r the wave numbers of the component Fourier modes.

When the solution is advanced one time step using (2.14), the linearity of the scheme results in each Fourier mode being separately advected :

$$\begin{aligned} u_j^{n+1} &= \mathcal{C}(E,\nu)\left\{\sum_{r=-\infty}^{\infty} \alpha_r e^{ij\phi_r}\right\} \\ &= \sum_{r=-\infty}^{\infty} \alpha_r\ \mathcal{C}(E,\nu)\ e^{ij\phi_r}. \end{aligned}$$

The evolution operator combines with the complex exponentials, to produce an amplification factor multiplying each Fourier component :

$$\sum_{r=-\infty}^{\infty} \alpha_r\ \mathcal{C}(E,\nu)\ e^{ij\phi_r} = \sum_{r=-\infty}^{\infty} \alpha_r\ g(\phi_r,\nu)\ e^{ij\phi_r}.$$

This equation shows that the relative importance of the amplification factors, $g(\phi_r,\nu)$, is dictated by the Fourier weights, α_r.

Some measure of the deviation of numerical advection from exact advection is required. Again the approach is to analyse amplitude and phase error separately.

2.5.2 Phase Error

The chosen measure of phase error is simply a weighted sum of the phase errors for each Fourier component :

$$\mathrm{E}_n(\nu)^2 = \frac{\sum_{r=-\infty}^{\infty} |\alpha_r(t_n)|^2 \{\arg[g(\phi_r,\nu)] - (-\nu\phi_r)\}^2}{\sum_{r=-\infty}^{\infty} |\alpha_r(t_n)|^2}. \tag{2.27}$$

The corresponding weighted amplitude error would appear to be of little use; it is symmetric about the stability limit $|g| = 1$. However, by making use of Parseval's Theorem, the conserved fraction (defined below) of the second moment of the solution may be obtained.

2.5.3 Conservation of Second Moment

Parseval's Theorem Consider an infinite periodic wave with wavelength $2l$:

$$\text{If} \quad f(x) = \sum_{r=-\infty}^{\infty} \alpha_r \; e^{ir\pi x/l}$$

$$\text{then} \quad \frac{1}{2l}\int_{-l}^{l} f(x)^2 \, dx = \sum_{r=-\infty}^{\infty} |\alpha_r|^2$$

under suitable conditions on $f(x)$. □

Let $U(x,t_n)$ be the continuous extension of the finite difference solution, where the index j is replaced by $x/\triangle x$:

$$U(x,t_n) = \sum_{r=-\infty}^{\infty} \alpha_r(t_n) \; e^{i\phi_r x/\triangle x} \tag{2.28}$$

and let

$$M_n = \int_{-L}^{L} U(x,t_n)^2 \, dx.$$

If the wavelength is $2L$ then Parseval's Theorem provides the following :

$$M_n = 2L \sum_{r=-\infty}^{\infty} |\alpha_r|^2 . \tag{2.29}$$

When the evolution operator $\mathcal{C}(E,\nu)$ is applied to (2.28), each Fourier mode produces an amplification factor :

$$U(x,t_n+\triangle t) = \sum_{r=-\infty}^{\infty} \alpha_r \; g(\phi_r,\nu) \; e^{i\phi_r x/\triangle x}.$$

Applying Parseval's Theorem to this gives

$$M_{n+1} = 2L \sum_{r=-\infty}^{\infty} |\alpha_r \; g(\phi_r, \nu)|^2 . \tag{2.30}$$

Define the conservation fraction per time step for the second moment :

$$\mathrm{C}_n(\nu) = \frac{M_{n+1}}{M_n}$$

From (2.29) and (2.30) this is seen to be

$$\mathrm{C}_n(\nu) = \frac{\sum_{r=-\infty}^{\infty} |\alpha_r(t_n)|^2 \; |g(\phi_r, \nu)|^2}{\sum_{r=-\infty}^{\infty} |\alpha_r(t_n)|^2}. \tag{2.31}$$

For linear advection with a constant velocity field, all the finite difference schemes considered in this report provide perfect conservation of mass. The second moment is not necessarily conserved : non-conservation is a result - though not a necessary one - of changes in the shape of the advected mass distribution.

2.5.4 Convergence of $\mathbf{E}_n(\nu)$ and $\mathbf{C}_n(\nu)$

Convergence of the summations in (2.27) and (2.31) follows from a property of the Fourier expansion, namely, that $\sum |\alpha_r|$ converges. Convergence of C_n is ensured since $|g| \leq 1$, for any stable scheme.

There is a slight problem involved in the calculation of $\mathrm{E}_n(\nu)$. Care must be taken in the calculation of $\arg[g(\phi_r, \nu)] + \nu\phi_r$, which is the phase error in the amplification factor for Fourier mode $e^{i\phi_r}$. The problem arises when $\nu\phi_r$ is outside the range of the arg function, since it is then necessary to find the winding number for $g(\phi_r, \nu)$. However, this may be side-stepped by recalling that waves with wavenumbers ϕ and $\phi + 2\pi$ are indistinguishable on the grid. This allows the phase error to be mapped onto the interval $(-\pi, \pi]$ by the addition/subtraction of multiples of 2π. Provided the scheme is known never to produce phase errors exceeding 2π, then the above procedure provides the correct phase error. This is the case for all the schemes to be considered here.

2.5.5 Courant Average

By averaging $\mathrm{C}_n(\nu)$ and $\mathrm{E}_n(\nu)$ over the range $0 \leq \nu \leq 1$, single values are obtained, corresponding to second moment conservation and average phase error respectively. This is accomplished by dividing the unit ν-interval into N equal parts. Define

$$\begin{aligned} \overline{\mathrm{C}}_n &= \frac{1}{N}\sum_{r=0}^{N} \mathrm{C}_n(r/N) \\ \overline{\mathrm{E}}_n &= \frac{1}{N}\sum_{r=0}^{N} \mathrm{E}_n(r/N). \end{aligned}$$

Finally, we have arrived at two quantities related to the error in numerically advecting an arbitrary periodic wave. Both quantities are time dependent. This is due to the relative changes in the Fourier coefficients, since they each evolve according to their own amplification factor. Unlike the single Fourier wave case, where the wave retains similarity of form throughout all time steps, the data now changes its shape. Since it is the finite difference scheme which causes this change, it is to be expected that the scheme will perform better for this new shape : the shape becomes better suited to the scheme. This indicates that measurements of the error committed during the first time step are sufficient to describe the scheme's performance. It is then only a matter of testing the scheme with a variety of different initial wave shapes. The values of $\overline{\mathrm{C}}_0$ and $\overline{\mathrm{E}}_0$ then indicate the scheme's effectiveness for the chosen shape.

3 Results

The analysis presented in sections 2.4 and 2.5 is applied to two groups of schemes. The first group comprises the schemes of Lax-Wendroff [5], Warming and Beam [10], and Fromm [2] in addition to the first order upwind scheme. In the second group various forms of interpolation are applied to the semi-Lagrangian scheme outlined in section 1. Each scheme is presented together with graphical results of the appropriate Fourier analysis. Then results of second moment conservation and average phase error are given.

3.1 Standard Schemes

Lax-Wendroff

$$u_j^{n+1} = \frac{\nu}{2}(1+\nu)\ u_{j-1}^n + (1-\nu^2)\ u_j^n - \frac{\nu}{2}(1-\nu)\ u_{j+1}^n$$

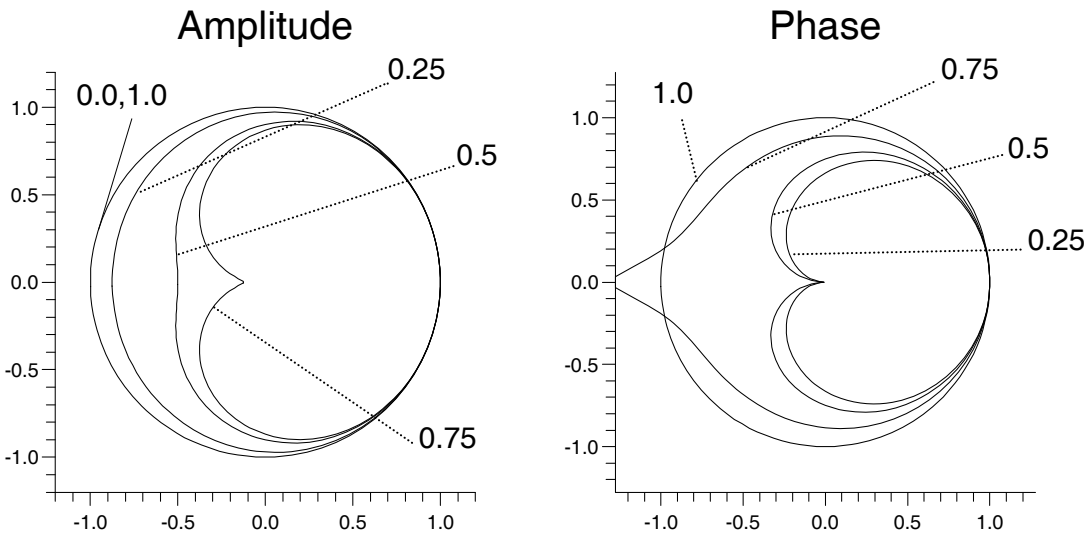

Figure 2: Lax-Wendrfoff

Warming and Beam This is the second order upwind scheme,

$$u_j^{n+1} = \frac{1}{2}\nu(1-\nu)\ u_{j-2}^n + \nu(2-\nu)\ u_{j-1}^n + \frac{1}{2}(\nu-1)(\nu-2)\ u_j^n$$

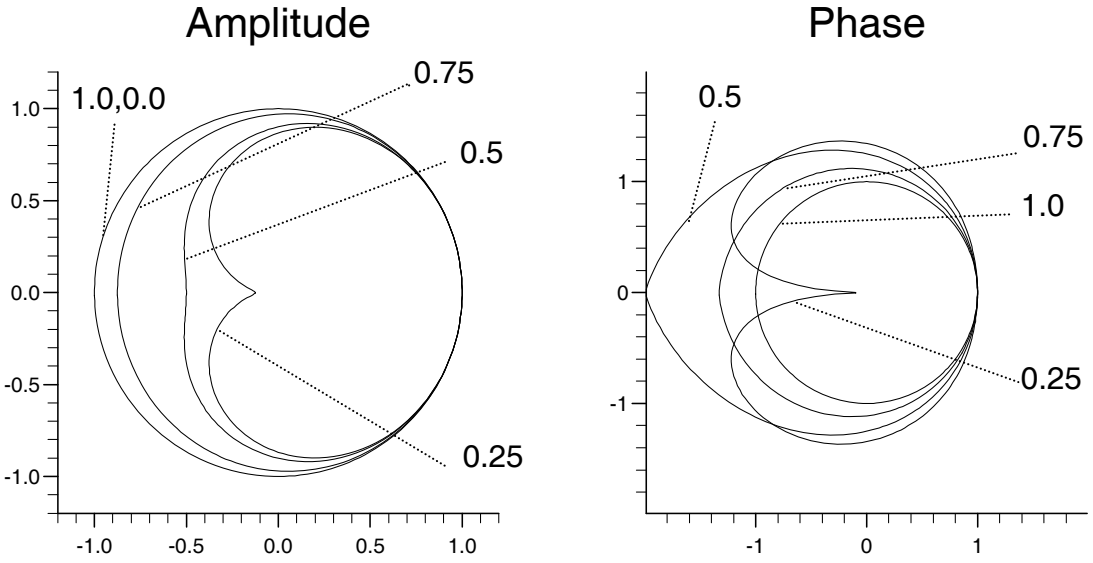

Figure 3: Warming and Beam

Fromm Fromm is the mean of the Lax-Wendroff and Warming and Beam schemes. It has the form

$$u_j^{n+1} = \frac{\nu}{4}(\nu-1)\ u_{j-2}^n + \frac{\nu}{4}(5-\nu)\ u_{j-1}^n - \frac{1}{4}(\nu^2+3\nu-4)\ u_j^n - \frac{\nu}{4}(1-\nu)\ u_{j+1}^n$$

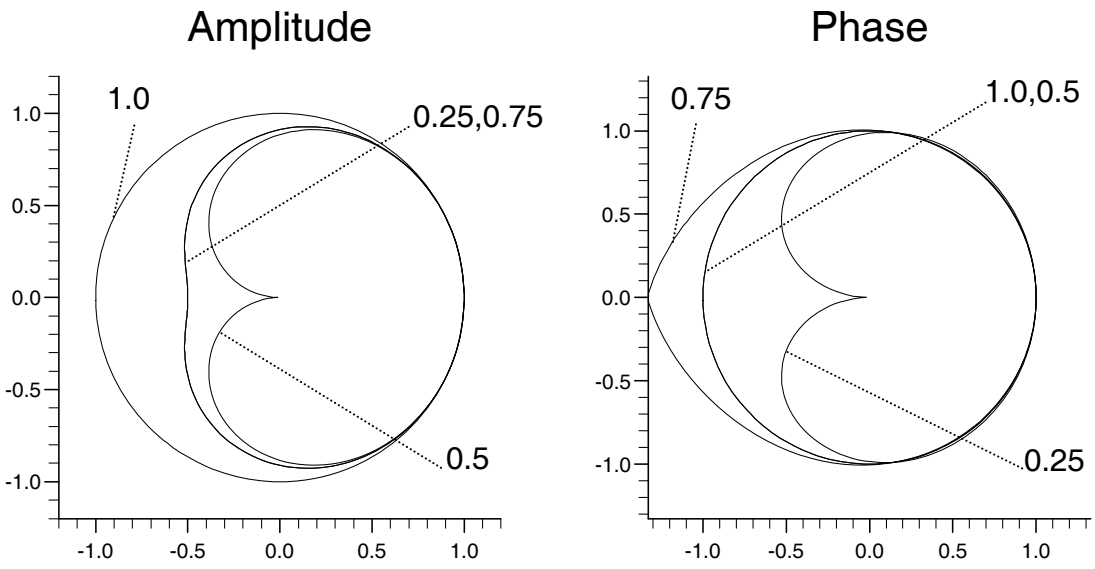

Figure 4: Fromm's Scheme

3.2 Semi-Lagrangian Schemes

In section 1 the semi-Lagrangian finite difference method was presented, along with several possible choices for interpolation. The results of Fourier analysis on these schemes is presented here in graphical form.

Cubic Lagrange The Lagrange interpolating polynomial of degree three is the simplest semi-Lagrangian method with fourth order spatial accuracy.

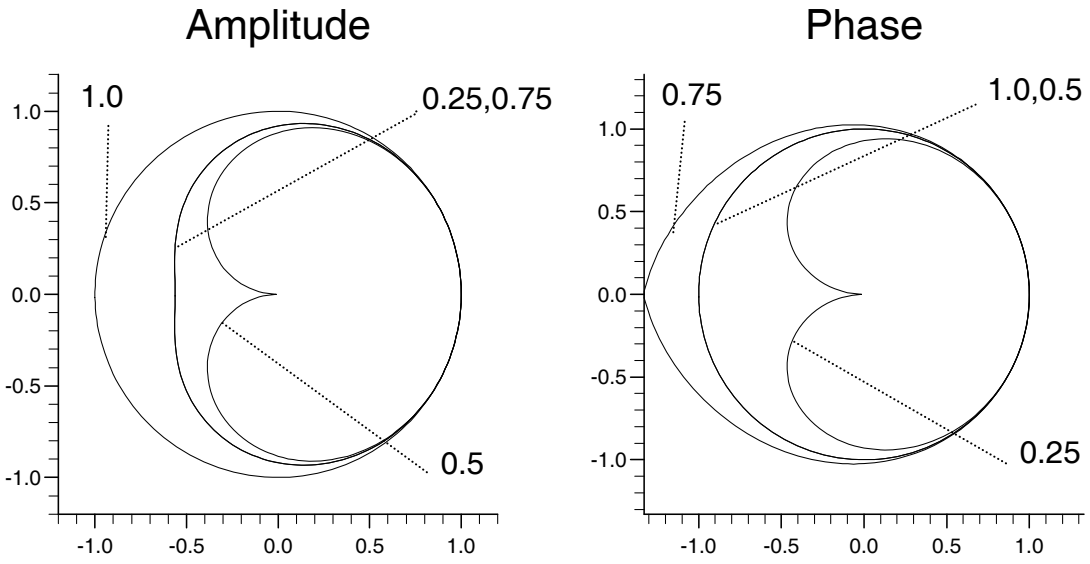

Figure 5: Cubic Lagrange

Quintic Lagrange Despite the extra effort in computing a higher degree polynomial, the quintic Lagrange method shows little improvement over the cubic Lagrange.

Hermite Cubic with Arithmetic Mean Hermite cubic interpolation requires estimates for the derivative at each grid node. These are usually expressed in terms of the discrete slopes of the finite difference solution. A simple form of

derivative estimate is provided by the mean of the discrete slopes, either side of a grid node.

Hermite Cubic with Priestley Derivatives The Priestley derivative estimate [6] provides fourth order accuracy. This is consistent with the accuracy of the Hermite cubic itself.

Hermite Cubic with Hyman Derivatives The Hyman derivative estimate [3] is similar in form to Priestley's.

Cubic Spline This is a global interpolation method requiring considerable computational effort when used in practice.

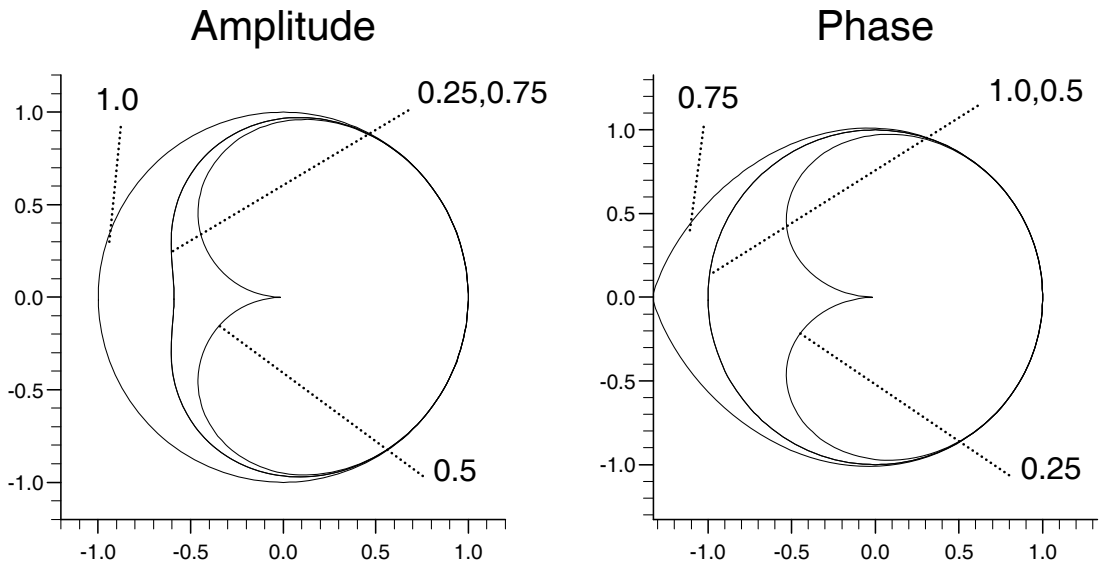

Figure 6: Quintic Lagrange

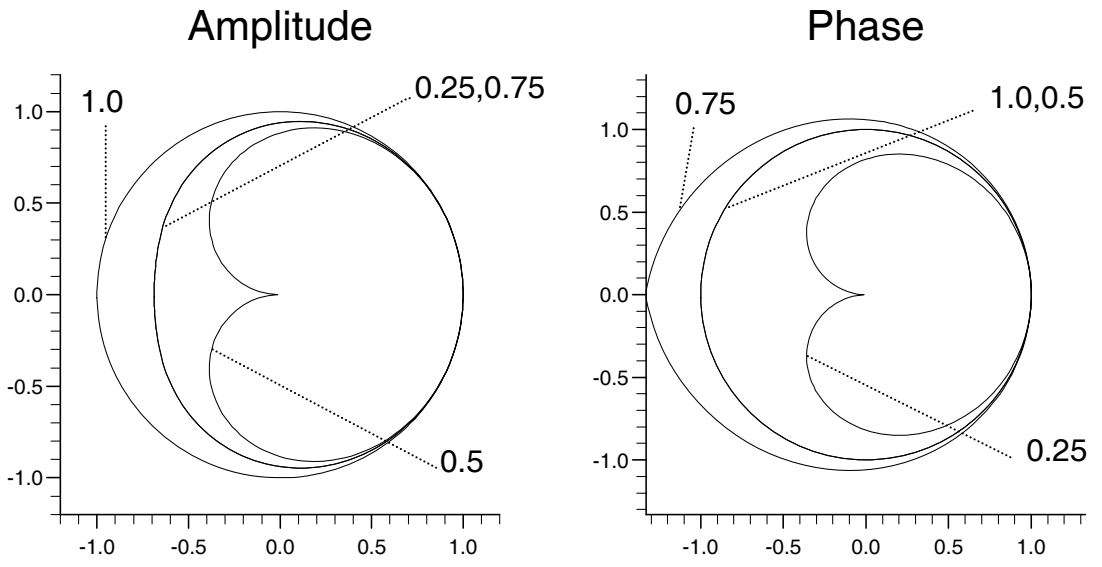

Figure 7: Hermite Cubic with Arithmetic Mean

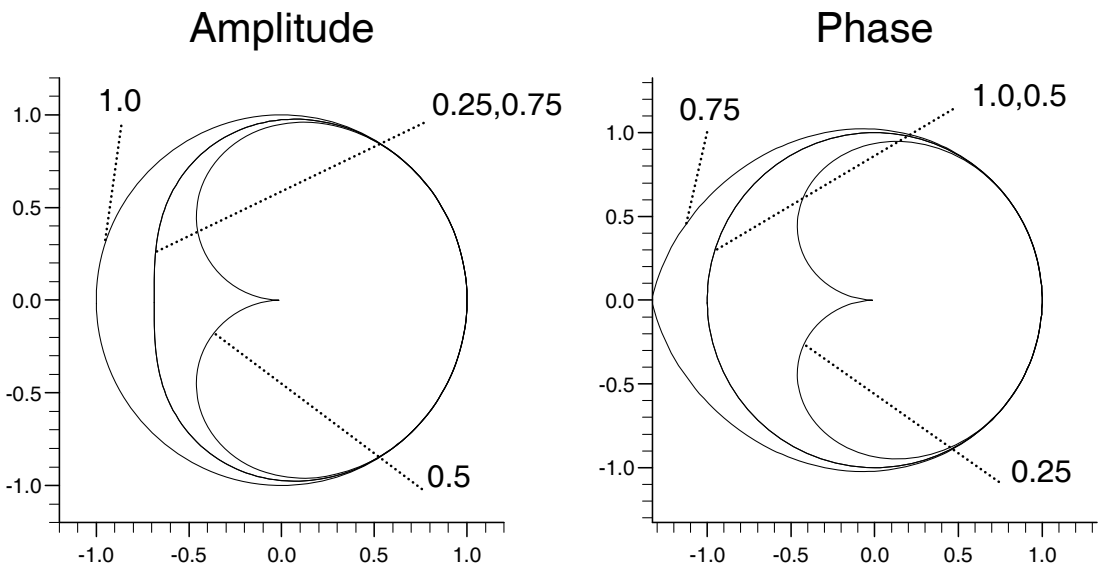

Figure 8: Hermite Cubic with Priestley Derivatives

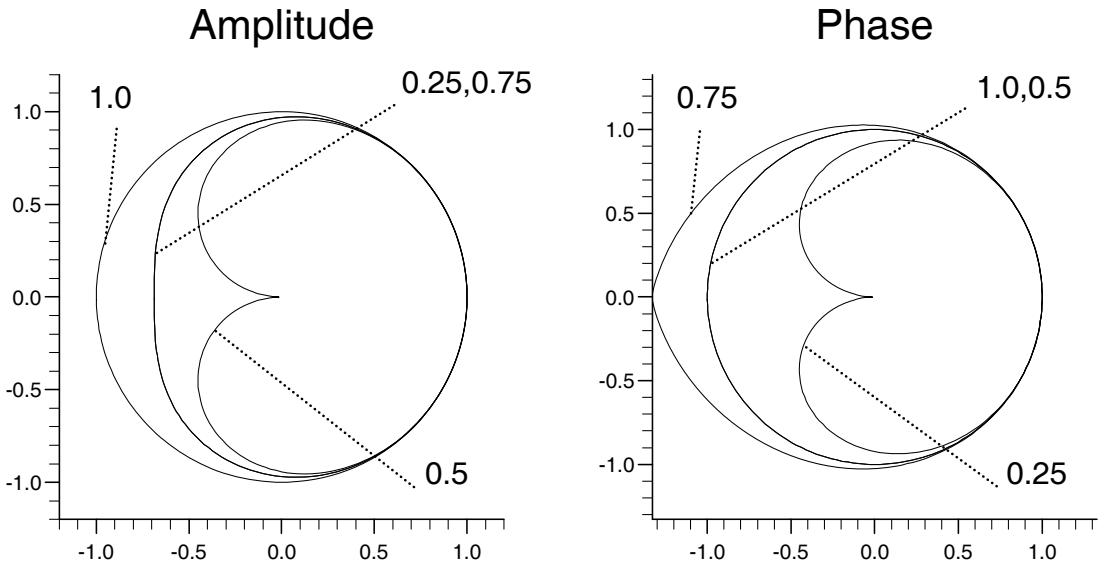

Figure 9: Hermite Cubic with Hyman Derivatives

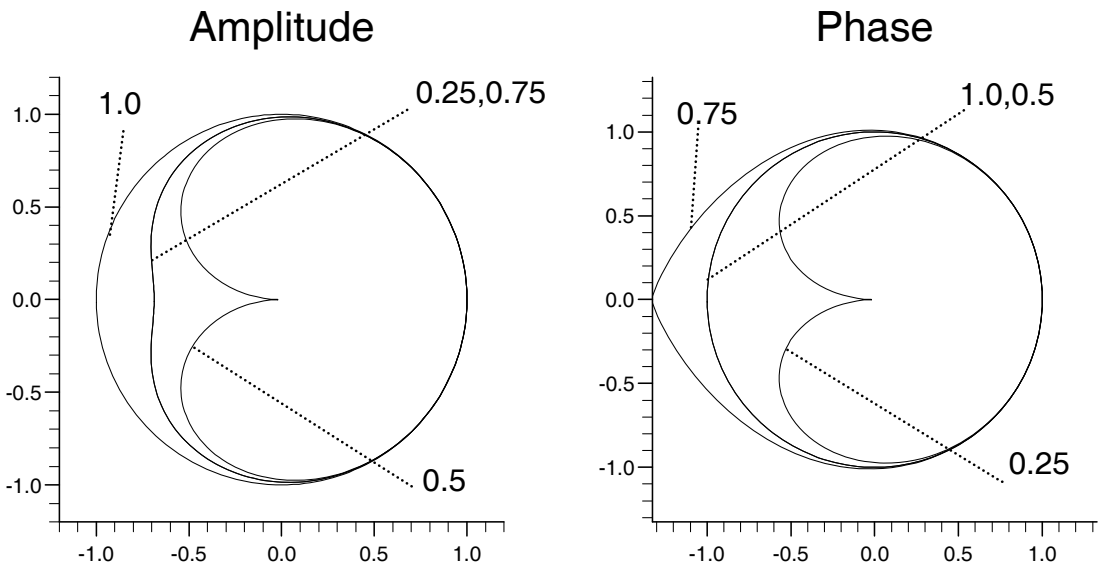

Figure 10: Cubic Spline

3.3 Conservation of 2nd Moment And Average Phase Error

In section 2.5 the errors incurred by numerically advecting an arbitrary periodic waveform were investigated. Quantities were established for measuring second moment conservation and average phase error; theses are $\overline{\mathrm{C}}_0$ and $\overline{\mathrm{E}}_0$ respectively. Both quantities were obtained by averaging over a range of Courant numbers and consequently provide an indication of the overall performance of a scheme. However both $\overline{\mathrm{C}}_0$ and $\overline{\mathrm{E}}_0$ were seen to be dependent on the shape of the advected wave. This shape dependency enters the calculations for $\overline{\mathrm{C}}_0$ and $\overline{\mathrm{E}}_0$ through the Fourier coefficients of the wave. General periodic initial data has a Fourier expansion of the form :

$$u_j^0 = \sum_{r=-\infty}^{\infty} \alpha_r \; e^{ij\phi_r}.$$

The wave is specified by the wave numbers of the Fourier modes, ϕ_r, and the corresponding coefficients, α_r. The quantities $\overline{\mathrm{C}}_0$ and $\overline{\mathrm{E}}_0$ refer to errors averaged over all the component Fourier modes, with weightings determined by the amplitudes of the expansion coefficients. Hence it is the quantities ϕ_r and $|\alpha_r|^2$ which are of importance. Results are obtained for three different test shapes.

Square Wave The general form of the square wave used is depicted in *figure* 11. It has a wavelength of $2l$, and an 'on-fraction' μ. The choice $\mu = 1$ is used throughout.

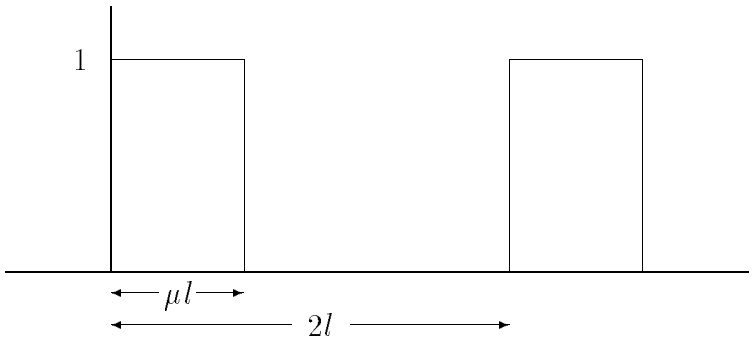

Figure 11: Square Wave

The component wave numbers are

$$\phi_r = r\pi\frac{\triangle x}{l}.$$

where $r = 0, \pm 1, \pm 2, \cdots$.

The corresponding error weights are

$$\begin{aligned} |\alpha_0|^2 &= \left(\frac{\mu}{2}\right)^2 \\ |\alpha_r|^2 &= \frac{\sin^2(r\pi\mu/2)}{(r\pi)^2} \end{aligned}$$

where $r = \pm 1, \pm 2, \cdots$.

Saw-Tooth Wave The saw-tooth wave, with wavelength $2l$, is shown in *figure* 12.

This has component wave numbers

$$\begin{aligned} \phi_{2r} &= 0 \\ \phi_{2r+1} &= (2r+1)\pi\frac{\triangle x}{l} \end{aligned}$$

where $r = 0, \pm 1, \pm 2, \cdots$.

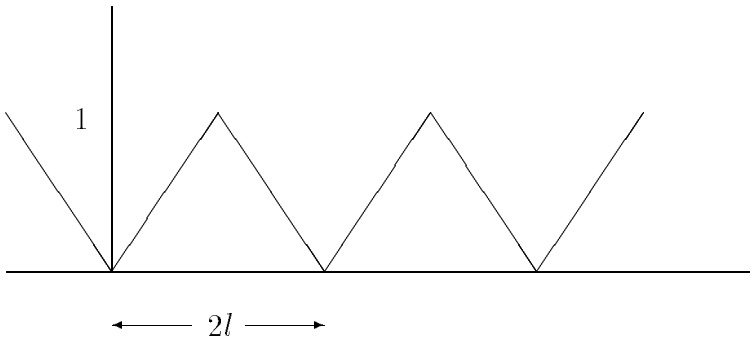

Figure 12: Saw-Tooth

The corresponding weights are

$$\begin{aligned} |\alpha_0|^2 &= \frac{\pi^2}{4} \\ |\alpha_{2r}|^2 &= 0 \qquad r = \pm 1, \pm 2, \cdots \\ |\alpha_{2r+1}|^2 &= \frac{4}{\pi^2(2r+1)^4} \quad r = 0, \pm 1, \pm 2, \cdots \end{aligned}$$

Parabolic Blips This test wave is similar to the square wave described above, except that the square waves are replaced by parabolas. The wave is described by the function :

$$f(x) = \begin{cases} \frac{4}{l^2}x(l-x) & 0 \le x < l \\ 0 & l \le x < 2l \end{cases}$$

where $f(x \pm 2l) = f(x)$. The wave numbers and corresponding weights which this function gives rise to are :

$$\begin{aligned} \phi_0 &= 0 \\ \phi_r &= r\pi\frac{\triangle x}{l} \end{aligned}$$

and

$$\begin{aligned} |\alpha_0|^2 &= \left(\frac{1}{3}\right)^2 \\ |\alpha_{2r-1}|^2 &= \left[\frac{8}{\pi^3(2r-1)^3}\right]^2 \\ |\alpha_{2r}|^2 &= \left(\frac{1}{\pi^2 r^2}\right)^2 \end{aligned}$$

where $r = 1, 2, \cdots$.

The measure of conservation of second moment is an absolute quantity : $\overline{\mathrm{C}}_0$ takes the value 1 if the second moment is conserved exactly; above and below this value correspond to gain and loss respectively. This is not the case for the phase measure, $\overline{\mathrm{E}}_0$. It takes the value 0 if there is no phase error. However no scale has yet been defined for non-zero values. It is chosen here to assign a phase error of 1 to the first order upwind scheme, and scale the errors for other schemes accordingly.

For each of the test waves there is a parameter which must be specified. This is the quantity $\frac{\triangle x}{l}$. It describes how well the wave is represented on the grid : it is the ratio of the grid spacing to a half wavelength. When this parameter is varied, there appears to be little, or no, relative change in performance for the schemes considered here. For this reason $\frac{\triangle x}{l}$ is assigned the value 0.1 throughout.

The results obtained, using the above definitions, are shown in *table* 1.

	Square Wave		Saw-Tooth		Parabolas	
Scheme	$\overline{C}_0$	$\overline{E}_0$	$\overline{C}_0$	$\overline{E}_0$	$\overline{C}_0$	$\overline{E}_0$
Upwind	0.9667	1.0000	1.2079	1.0000	0.98500	1.0000
Lax-Wendroff	0.9867	1.0701	1.1919	0.8898	0.99869	1.1691
Warming-Beam	0.9867	1.0701	1.0000	0.6319	0.99869	1.1691
Fromm	0.9842	1.0235	1.0720	0.8488	0.99856	0.9927
Cubic Lagrange	0.9853	1.0141	1.0000	0.6319	0.99867	0.9902
Quintic Lagrange	0.9895	1.0229	1.0000	0.6319	0.99950	0.9920
Arithmetic Mean	0.9879	1.0000	1.0000	0.9357	0.99890	1.0000
Priestley	0.9911	1.0139	1.0487	0.7405	0.99958	0.9898
Hyman	0.9907	1.0122	1.0409	0.7475	0.99950	0.9899
Cubic Spline	0.9928	1.0275	1.0387	0.6861	0.99968	0.9938

Table 1: Error comparison

Conclusion

The weighted averages for phase error and second moment conservation error appear to have practical significance. They provide measurements for errors committed in numerically advecting reasonably realistic data. A particularly useful feature is, that these error measurements are obtained without the need to run a code of the scheme. They also allow the possibility of testing how well a scheme copes with advecting a given shape of periodic wave.

The results given in *table* 1 show that, in general, all the schemes perform best on data without sudden jumps and with fewest sharp corners.

Throughout all the tests the upwind scheme displays the worst conservation of second moment. This is to be expected, since it is a first order scheme. Its solutions display severe smearing and diffusion.

Fourier analysis shows that the Lax-Wendroff scheme suffers from phase lag at most frequencies. Warming and Beam suffers mainly phase advance. The motivation behind the Fromm scheme, which is the mean of these two schemes, is that there will be cancellation of their respective phase errors. The results in *table* 1 show that this is mostly the case. However, this improvement in phase is achieved at the cost of greater loss of second moment conservation.

The Hermite cubic interpolant with mean derivative estimate consistently lies somewhere between cubic and quintic Lagrange interpolations, whilst the higher order derivative estimates of Priestley and Hyman out-perform quintic Lagrange interpolation. Results for theses two schemes are very similar. Priestley's scheme provides slightly better second moment conservation than Hyman's. The Hyman scheme tends to give better phase, but this difference between the two is less pronounced.

The cubic spline has extremely good second moment conservation. This indicates that the interpolation has captured the wave shape very well. Perhaps this is due to the fact that the spline is a global interpolation. Despite this its phase errors are a little disappointing.

Acknowledgements

I would like to thank Dr. M.J. Baines and Dr. A. Priestley for suggesting the work reviewed here, and for their continual guidance throughout. I acknowledge receipt of an SERC/Meteorological Office CASE Award.

References

[1] Bates J.R. and McDonald A. (1982) *Multiple Upstream, semi-Lagrangian Advective Schemes : Analysis and Application to a Multi-Level Primitive Equation Model.* Mon. Wea. Rev. **112**,2033-2047.

[2] Fromm J.E. (1968) *A Method for Reducing Dispersion in Convective Difference Schemes.* J. Comp. Phys. **3**,176-189.

[3] Hyman J.M. (1983) *Accurate Monotonicity Preserving Cubic Interpolations.* SIAM J. Sci. Stat. Comp. **4**,645-654.

[4] Jacques I. and Judd C. (1987) *Numerical Analysis.* Chapman and Hall.

[5] Lax P.D. and Wendroff B. (1960) *Systems of Conservation Laws.* Comm. Pure and Appl. Math. **X**,537-566.

[6] Priestley A. (1993) To appear, J. Comp. Phys.

[7] Rasch P.J. and Williamson D.L. (1990) *On Shape-Preserving Interpolation and Semi-Lagrangian Transport.* SIAM J. Sci. Stat. Comp. **11**,656-689.

[8] Staniforth A. and Côté J. (1991) *Semi-Lagrangian Integration Schemes for Atmospheric Models - Review.* Mon. Wea. Rev. **119**,2206-2223.

[9] Temperton C. and Staniforth A. (1987) *An Efficient Two-Time Level Semi-Lagrangian Semi-Implicit Integration Scheme.* Q.J.R. Met. Soc. **113**,1025-1039.

[10] Warming R.F. and Beam R.W. (1976) *Upwind Second-Order Difference Schemes and Applications in Aerodynamic Flows.* AIAA J. **24**,1241.