

FOURIER METHODS IN COMPUTATIONAL

FLUID AND FIELD DYNAMICS

DCS-TR-129

by Robert VICHNEVETSKY

Rutgers University

New Brunswick, New Jersey 08903 U.S.A.

This is the text of a survey paper
presented at the IMACS-International
Symposium on Simulation in Engineering
Sciences - Nantes (France) - May 1983

FOURIER METHODS IN COMPUTATIONAL FLUID AND FIELD DYNAMICS

Robert VICHNEVETSKY
Rutgers University
New Brunswick, New Jersey, 08903, U.S.A.

ABSTRACT

This paper is a review, with examples, of those areas where the theory of Fourier transforms has played a role in the development of computational methods in fluid and field dynamics. These may be separated into (i) the development of computing algorithms using Fourier transforms and (ii) the analysis of standard finite difference and finite element algorithms by Fourier methods.

Recent results in the analysis of spurious reflection phenomena at computational boundaries of fluid flow simulation, which invoke the theory of wave propagation and the concept of group velocity are given.

1. INTRODUCTION

There are essentially two areas where the theory of Fourier transforms plays a role in the computation of flows and fields.

1. Fourier algorithms which rely on the property of Fourier series to carry out specific calculations.

2. Fourier analysis which rely on the theory as a tool for the analysis of error and stability properties of otherwise standard finite difference or finite element algorithms. Albeit theoretical, this second use of Fourier transforms seems, in a way, to have played a role more important than the first.

The relationship of trigonometric series with the hyperbolic partial differential equations of mathematical physics has been known since the 18th century. The first contributions to the theory of vibrating strings by Brook Taylor (1685-1731), Leonhard Euler (1707-1783), Daniel Bernoulli (1700-1782), and Louis Lagrange (1736-1813) among others, do contain indeed many uses of this analytic tool. The name of Fourier became attached later on to trigonometric series, after he applied them to the solution of the heat equation which he had derived (Fourier, - 1807).

2. THE FAST FOURIER TRANSFORM

An algorithm that is fundamental to the implementation of Fourier calculations is of course the Fast Fourier Transform (FFT). It uses the symmetry of trigonometric functions to regroup the equations so as to minimize the computational effort. Whereas computing the discrete Fourier transform of a function represented by N discrete data points requires on the order of N^2 operations, when implemented in the straightforward way, organizing the same calculation as prescribed by the FFT requires only on the order of $N \log_2 N$ operations (for $N=1024$, this represents a saving of more than 100 to 1).

The original description of the fast Fourier transform was given in a paper by Cooley and Tukey published in *Mathematics of Computation* in 1965, although the technique was not unknown in the field. For example, the basic idea of the method may be found on page 239 of C. Lanczos' *Applied Analysis*, published in 1956, and earlier results are described in Danielson and Lanczos (1942). Danielson and Lanczos, in turn, refer to Runge (1903, 1905) for the source of their method. Other pre-1965 users of "special techniques" which were in the same vein as the FFT are given in the book *The Fast Fourier Transform* by Brigham (1974). "Computational tricks" were not held in high esteem in the precomputer days, and attempts to give them visibility under a unified heading were few. The name of "numerical analysis" was coined soon after the advent of electronic computers. The number of people involved in doing scientific calculations grew dramatically, and many techniques that had not been well known came to the surface or were reinvented at that time. The "invention" of the fast Fourier transform in the 1960's is an example.

3. FOURIER METHODS IN COMPUTATIONAL FLUID DYNAMICS

It is after fast, large scale computers had become available (in the early 1960's), and that the fast Fourier transform algorithm had been popularized (in the mid to late 1960's) that Fourier methods in fluid flow calculations made their appearance. We shall briefly describe their principal fields of application :

Computational fluid dynamics imply the approximation of both first and second order spatial derivatives. The equations which illustrate the relevant concepts are :

(i) Those of the motion of a non viscous fluid (often called Euler's equations : he first described them in 1752-1755), which may be written as :

$$\frac{DU}{Dt} = \frac{\partial U}{\partial t} + U \cdot \nabla \cdot U = g - \frac{1}{\rho} \nabla p \quad (1)$$

where $U(x, t)$ is a vector and $\nabla \cdot$ is the gradient operator. Solving these equations numerically requires that the advection term $U \cdot \nabla \cdot U$ be approximated, and Fourier methods have been used mostly with the intent of improving the accuracy of this computation.

(ii) Those of viscous flow, illustrated by the Navier Stokes equation for an incompressible flow, in primitive form :

$$\frac{DU}{Dt} = g - \frac{1}{\rho} \nabla p + \nu \nabla^2 U \quad (2)$$

where ∇^2 is the Laplacian operator and ν is the dynamic viscosity of the fluid. An alternate formulation is the stream function vorticity form :

$$\begin{aligned} \frac{D\zeta}{Dt} &= \nu \nabla^2 \zeta \\ \nabla^2 \psi &= \zeta \end{aligned} \quad U = \begin{bmatrix} \partial \psi / \partial y \\ -\partial \psi / \partial x \end{bmatrix} \quad (3)$$

The numerical treatment of these equations requires that the Laplacian ∇^2 also be approximated. In many cases (in particular if the vorticity form (3) is used, or if (2) is integrated by an implicit time marching method), the solution process requires that a form of Poisson's equation

$$\nabla^2 \psi = f(x, y) \quad (4)$$

be solved at each time step. In this case, Fourier methods are often used to reduce the computing time in the solution of (4).

4. APPLICATIONS IN PLASMA PHYSICS

Questions of the stability of nuclear plasma in the context of the development of nuclear fusion energy devices are often investigated by direct simulation. In one type of approach, the "fluid" consisting of the moving charged particles is represented by a finite number of superparticles : on the order of 10^4 superparticles are used in realistic simulation (it is reported that physicists doing these kinds of simulation are sometimes called

"particle pushers" by those doing the more conventional - continuous model simulations). The particles are in motion in an exterior magnetic field and a self generated electrical field. The latter is computed by solving Poisson's equation, at each time step (or every few time steps). Since many time steps are needed, it is vital, as with Navier Stokes' equations, that the solution of Poisson's equation be obtained as economically as possible.

5. FOURIER METHODS FOR FIRST ORDER HYPERBOLIC EQUATIONS

Consider as a model of (1) the linear, scalar equation :

$$\frac{\partial U}{\partial t} + c \frac{\partial U}{\partial x} = G(x, t) \quad (5)$$

over $x \in [0, l]$ with boundary conditions :

$$U(0, t) = U(l, t) \quad (6)$$

Periodic boundary conditions of this kind occur for instance in mathematical models of the global atmosphere and Fourier methods have indeed been developed and used to a large extent by geophysical fluid dynamicists and meteorologists.

$U(x, t)$ may be expressed in Fourier series form as :

$$U(x, t) = \frac{1}{2} \sum_{k=-\infty}^{\infty} \hat{U}_k e^{2ik\pi x/l} \quad (7)$$

where

$$\hat{U}_k(t) = \int_0^l U(x, t) e^{-2ik\pi x/l} dx \quad (8)$$

The form of $U(x, t)$ which solves (5) is then easily obtained. Since

$$c \frac{\partial U}{\partial x} = \frac{c}{l} \sum \frac{2ik\pi}{l} \hat{U}_k e^{2ik\pi x/l} \quad (9)$$

it follows that each $\hat{U}_k(t)$ satisfies

$$\frac{d\hat{U}_k}{dt} + c \frac{2ik\pi}{l} \hat{U}_k = \hat{G}_k \quad (10)$$

where $\{\hat{G}_k\}$ is the Fourier transform of $G(x, t)$

It is Daniel Bernoulli who, with his 1753 paper, may be credited with paternity of the idea of the separation of the dynamics of a string into sinusoidal modes which do not interact with one another. Once more, history had its word to say, and the method of separation of variables is generally called the method of Fourier, who used it more than half a century later.

Approximations are obtained when k is restricted in (7) to within a finite range of values, say $(-N, N)$, i.e., $U(x, t)$ is approximated by a truncated Fourier series. Using this concept leads to several types of feasible approximations, such as :

(i) truncated Fourier series methods in which time integration is done in Fourier space and all calculations

take place in the frequency domain. This is restricted to linear problems.

(ii) collocation methods in which time integration is done by expressing the solution in a finite number of discrete points in physical space, called collocation points, and trigonometric interpolation is used solely with the intent of approximating the spatial partial derivatives in these points.

(iii) collocation methods in which time integration is done in Fourier space, but transformation is done at each time step to and from physical space, where the non-linear terms are being computed.

Of these, only (ii) and (iii) have general applicability to a variety of problems in fluid dynamics, since they can be applied to non-linear equations.

6. FOURIER METHODS FOR SECOND ORDER EQUATIONS

Special Poisson's equation solvers which use Fourier transforms have been developed in the mid to late 1960's. Their first use was in the simulation of nuclear plasmas, and they have been applied to fluid dynamics later on. Fast Poisson Solvers were initially restricted to simple rectangular regions. (Reference to subsequent work toward the application of Poisson solvers to less regular domains may be found for instance in Proskurowski - 1982; see also Vichnevetsky - 1981-a). Consider for instance the solution of (4) in a square with the simple boundary conditions $U = 0$ on ∂D . The simplest method uses a double Fourier series :

$$U(x,y) = \sum_{p,q} \hat{U}_{p,q} \sin\left(\frac{p\pi x}{x_{max}}\right) \sin\left(\frac{q\pi y}{y_{max}}\right) \quad (11)$$

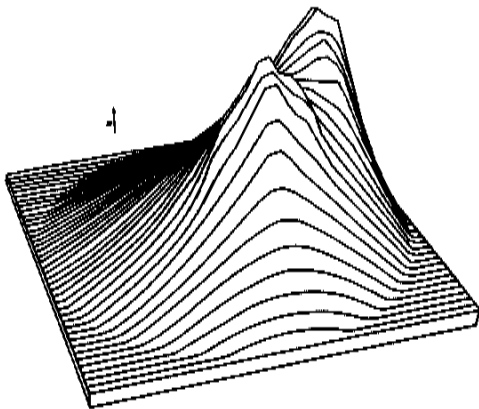


Figure 1 - A solution of Poisson's equation obtained with a solver using fast Fourier transforms.

which is a solution of (4) when the coefficients $\{\hat{U}_{p,q}\}$ are :

$$\hat{U}_{p,q} = \frac{-\hat{f}_{p,q}}{(p\pi/x_{max})^2 + (q\pi/y_{max})^2} \quad (12)$$

and $\{\hat{f}_{p,q}\}$ is the Fourier transform of $f(x,y)$

There is also the popular Hockney method which consist in using Fourier transforms in x and finite differences in y . The resulting system of tridiagonal equations is then solved by recursive cyclic reduction.

Original ideas and analyses of the efficiency of the Fast Poisson solvers with Fourier transforms may be found in Hockney (1965), Gentleman and Sande (1966), Buneman (1969), Colony and Reynolds (1970), and Le Bail (1972).

7. COMPARISONS

Fourier methods achieve one thing that finite difference methods generally do not : they approximate the partial derivatives with the maximum accuracy which is permitted by a representation of the variables in a finite number of discrete points. Essentially, the derivatives are approximated by the analytic derivative of the trigonometric interpolant between all these points.

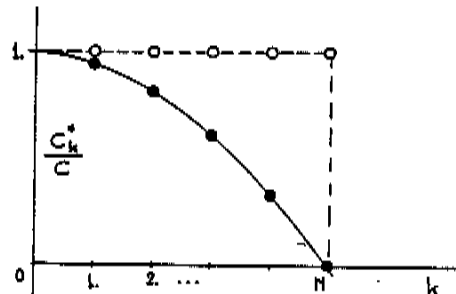


Figure 2 - Numerical phase velocity for Fourier components of wave number k in finite difference \bullet and Fourier \circ approximations.

Consider the following error analysis :

A numerical solution of (5) consisting of $2N$ discrete values in physical space may be expressed in Fourier series forms such as (7) with $k=-N,\dots,N$. The approximation of the advective term $c \partial U / \partial x$ by standard three point finite differences then results in a Fourier series representation of the advective term by :

$$\frac{1}{l} \sum c_k^* \frac{2ik\pi}{l} \hat{U}_k e^{2ik\pi x/l} \quad (13)$$

where c_k^* the phase velocity of the k^{th} Fourier component is given by (see section 9 below) :

$$c_k^* = c \frac{\sin(2k\pi h/l)}{(2k\pi h/l)} \quad (14)$$

The difference $C_k^* - C$ measures the error in the approximation of $C \frac{\partial U}{\partial x}$. By contrast, Fourier methods approximate this term with (9) where the correct velocity C is preserved in all the Fourier components which are included in the computation, and the only source of error is that which results from the elimination or aliasing of the higher wave number components.

Comparisons of the efficiency of Fourier methods with that of standard higher order finite difference approximations have been carried out in some detail in those cases where high accuracy is sought (as in long term atmospheric simulations). These may be found in Ozszag (1971), Kreiss and Olinger (1972) and Fornberg (1975).

There are however, cases in the simulation of engineering systems where a lower order of accuracy is sufficient. The possible advantage of Fourier methods then stems from the fact that a very few number of Fourier components may have the potential of giving a realistic approximation, while a very few number of discrete points in physical space may not have that ability. See for instance, discussions about this in Clymer (1966). Applications of this concept are found in recent work in the simulation and control of what engineers refer to as distributed parameter systems. In fact, these "modal methods" had been known for a long time by nuclear physicists working in the area of the control of nuclear reactors. Not only do they consider sinusoidal or "Fourier" modes, but also other modes which have spatial shapes given by the eigenfunctions of an associated Sturm-Liouville problem - see for instance Kaplan et al. (1965) and Stacey (1967).

The limitation of Fourier methods lies of course in the fact that they prefer regular domains and simple boundary conditions, and that their efficiency fades away when those requisites are not met.

8. SIMULATION OF ATMOSPHERIC TURBULENCE

A typical application is in the simulation of large scale atmospheric turbulence.

An example is illustrated in Figure 3. These results give the stream function and temperature distribution obtained as part of a simulation of the atmosphere in the northern hemisphere by Peskin and Kowalski (1983), using a Fourier method that was originally given by Ozszag (1976). The numerical solution follows (iii) above, namely :

- All the problem's variables are expressed in Fourier space,
- They are transformed (by an FFT) to physical space at each time step for the calculation, point by point, of the non-linear terms,
- Non linear terms are re-transformed to Fourier

space by an FFT.

- Marching to the next time step is done in Fourier space - (a combination of leap-frog and Crank-Nicolson methods was used).

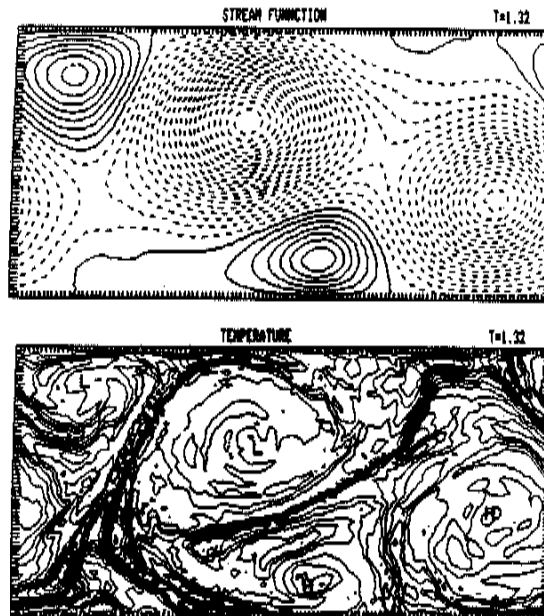


Figure 3 - Numerical simulation of atmospheric turbulence in the Northern hemisphere obtained with a spectral method (Peskin and Kowalski, 1983).

9. FOURIER ANALYSIS

Fourier analysis of the accuracy and stability properties of numerical algorithms may be achieved by observing the time evolution of sinusoidal (sometimes called "trial") solutions. This is the engineers' and physicists' approach. A mathematical refinement consists in observing instead the time evolution of the Fourier transform of numerical solutions. The mathematics are almost identical, but the second viewpoint gives a foundation for more detailed analyses, such as the calculation of energy norms of the error.

In both cases, one assumes that the equations may be locally linearized. The simple scalar equation :

$$\frac{\partial U}{\partial t} + c \frac{\partial U}{\partial x} = 0 \quad (15)$$

is used as a model where C is to assume the value of the characteristic velocities of the system of equations being integrated. In the case of compressible gas dynamics, these are the particle velocity, and the sound

velocities upstream and downstream of the flow, respectively.

The underlying assumption in those analyses is that the models are meant to represent the local behaviour of small perturbations of numerical solutions. It may appear as a paradox (but it is not) that such local analyses are modeled by sinusoidal solutions which must be defined on the entire real axis to allow Fourier transform theory to apply with its full analytic power.

Semi-Discretizations

For illustration we consider typical semi-discretizations such as the explicit (finite-difference)

$$\frac{du_n}{dt} = -c \left(\frac{u_{n+1} - u_{n-1}}{2h} \right) \quad (16a)$$

and implicit (linear finite element/Galerkin) :

$$\frac{1}{6} \left(\frac{du_{n-1}}{dt} + 4 \frac{du_n}{dt} + \frac{du_{n+1}}{dt} \right) = -c \left(\frac{u_{n+1} - u_{n-1}}{2h} \right) \quad (16b)$$

which may both be expressed in operator notations as :

$$\frac{du_n}{dt} = A \cdot u_n \quad (17)$$

where $A \cdot$ is an appropriately defined discrete (or Toeplitz) operator. Fully discrete algorithms are then obtained by applying to these some numerical time-marching procedure. But a first type of error analysis consists in assuming that time-marching errors may be neglected. The error may then be analyzed by defining the discrete Fourier transform of $\{u_n(t)\}$

$$\bar{u}(\omega, t) = h \sum_n u_n e^{-i\omega n h} \quad (18)$$

Taking the Fourier transform of (17) then results in

$$\frac{d\bar{u}}{dt} = \hat{A}(\omega) \cdot \bar{u} \quad (19)$$

where

$$\hat{A}(\omega) = A \cdot e^{i\omega n h} / e^{i\omega n h} \quad (20)$$

is the spectral functions of the operator $A \cdot$. Comparing this with the exact :

$$\frac{d\hat{u}(\omega, t)}{dt} = -i c \omega \hat{u}(\omega, t) \quad (21)$$

defines

$$c^*(\omega) = -\text{Im} \hat{A}(\omega) / \omega \quad (22)$$

as the phase velocity at which numerical Fourier components of spatial frequency, or wave number, ω propagate, and

$$R_e \hat{A}(\omega) \quad (23)$$

as the measure of an amplitude error. Semi-discretizations for which $R_e \hat{A}(\omega) = 0$ are called conservative semi-discretizations. Both (16a) and (16b) are conservative, with finite differences:

$$c^*(\omega) = c \frac{\sin(\omega h)}{\omega h} \quad (24a)$$

and

finite elements :

$$c^*(\omega) = c \frac{3 \sin(\omega h)}{\omega h (2 + \cos(\omega h))} \quad (24b)$$

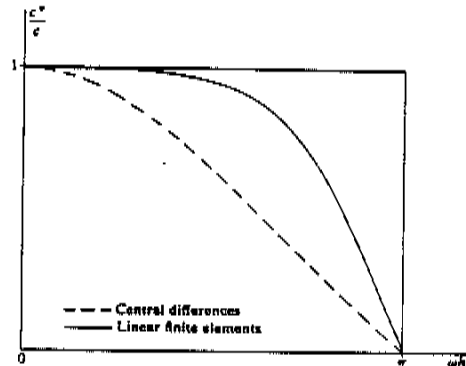


Figure 4 - Phase velocities of semi-discretizations versus wave number.

Fully Discrete Algorithms

Many time-marching methods for a system of equations written in the general form (17) may be expressed in operator notation as :

$$u_n^{j+1} = Z(\Delta t A) \cdot u_n^j \quad (25)$$

where $Z \cdot$ (sometimes called the amplification operator) is a function of the operator $A \cdot$, or more precisely of the dimensionless grouping $\Delta t A \cdot$. We have for example, for some of the well known time-marching methods :

Euler's Method :

$$Z \cdot = 1 + \Delta t A \cdot$$

Crank Nicolson Method :

$$Z \cdot = \frac{1 + \Delta t A \cdot / 2}{1 - \Delta t A \cdot / 2}$$

Leapfrog Method :

$$Z \cdot = \Delta t A \cdot \pm \sqrt{1 + (\Delta t A)^2}$$

To carry out a Fourier analysis of these fully discrete approximations, one replaces, as in the semi-discrete case, u_n^j by its Fourier transform \bar{u}^j

wherupon (25) becomes

$$\hat{u}^{j+1}(\omega) = \hat{Z}(\omega) \cdot \hat{u}^j(\omega) \quad (26)$$

obtained by replacing the operator A by its spectral function $\hat{A}(\omega)$ in the expression of Z . This was first done by von Neumann in his classical analysis of numerical stability (somewhat differently: he used sinusoidal trial solutions and inserted them without preliminaries in the full discretizations), and $\hat{Z}(\omega)$ is traditionally called the amplification factor.

But $\hat{Z}(\omega)$ contains information about accuracy as well: By comparison with (26), we have the ratio of Fourier transforms for an exact solution of (15):

$$\hat{U}(\omega, t^{j+1}) = e^{-i\omega c \Delta t} \hat{U}(\omega, t^j) \quad (27)$$

which allows one to define

$$c^*(\omega) = -\angle \hat{Z}(\omega) / \omega \Delta t \quad (28)$$

as the numerical phase velocity, and

$$|\hat{Z}(\omega)| - 1 \quad (29)$$

as an amplitude error: When $|\hat{Z}(\omega)| < 1$, there is a numerical or spurious damping. And $|\hat{Z}(\omega)| > 1$ corresponds to numerical instability (von Neumann).

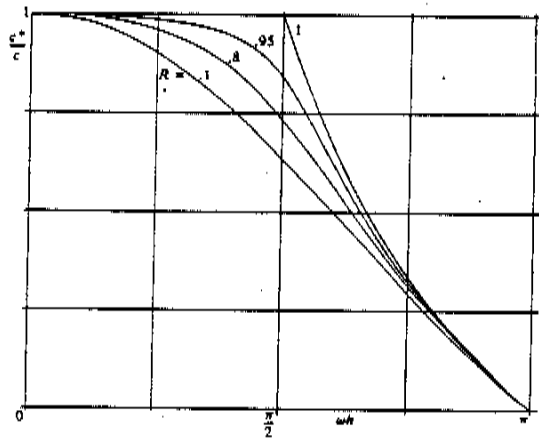


Figure 5 - Phase velocities versus wave number for the finite difference in space, leapfrog in time full discretization of (15). $R =$ Courant number $c \Delta t / h$

Analysis and discussion of specific expressions of numerical phase velocities may be found interspersed in the computational and fluid dynamics literature, for instance in Vichnevetsky and De Schutter (1975), Gray and Pinder (1976), Chin, Hedstrom and Karlsson (1979),

Vichnevetsky and Bowles (1982), and Lapidus and Pinder (1982). Equivalent analyses in the case of the second order wave equation are given in Dougalis and Serbin (1981) and, with specific reference to finite element approximations of the equations describing vibrations in mechanical structures, in Krieg and Key (1973) and in Belytchsko and Mullen (1978). (See also Section 15 below.)

10. TRANSFER FUNCTIONS

Engineers are familiar with the use of transfer function for the analysis of input/output properties of linear systems. Transfer functions are defined as a ratio of Fourier transforms. The same concept may be used to evaluate properties of semi-discrete approximations of hyperbolic equations. Consider the Fourier transforms in time:

$$\hat{U}(x, \Omega) = \int U(x, t) e^{-i\Omega t} dt \quad (30)$$

If $U(x, t)$ is an exact solution of the advection equation (15), then

$$\hat{U}(x, \Omega) = \hat{U}(0, \Omega) e^{-i\Omega x/c} \quad (31)$$

and for two adjacent mesh points:

$$\hat{U}(x_{n+1}, \Omega) = e^{-i\Omega h/c} \hat{U}(x_n, \Omega) \quad (32)$$

where $e^{-i\Omega h/c}$ is the transfer function between those two points.

Semi-discretizations have the effect of replacing this exact transfer function by approximations. For instance, to the simple upwind approximation

$$\frac{du_n}{dt} = -c \frac{u_n - u_{n-1}}{h} \quad (33)$$

corresponds the transfer function

$$\hat{u}_{n+1} / \hat{u}_n = \frac{1}{1 + i\Omega h/c} \quad (34)$$

which we call "cell transfer function" of the approximation. By comparison with (32) we find

$$c^*(\Omega) = \frac{\Omega h}{\arctan(\Omega h/c)} \quad (35)$$

for the numerical phase velocity, and the difference

$$(1 + (\Omega h/c)^2)^{-1/2} - 1$$

is an amplitude error which may be associated to spurious damping or spurious diffusion.

11. DISPERSION AND GROUP VELOCITY

The concept of group velocity occurs in describing the propagation of wave packets in dispersive media (defined as media where sinusoidal waves of the form $e^{i\omega(x - c^*(\omega)t)}$ may exist with a frequency dependent phase velocity $c^*(\omega)$). A wave packet is then

defined as a single wave number sinusoidal function modulated by a smooth envelope $\phi(x,t)$, such as :

$$U = \phi(x,t) e^{i\omega(x-c^*t)} \quad (37)$$

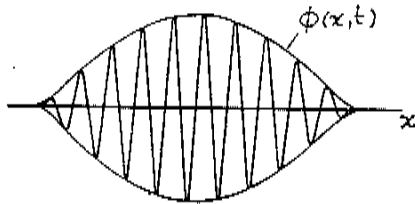


Figure 6 - Wave packet.

It is then found that the envelope of the packet does not propagate in the dispersive medium at the phase velocity c^* , but instead at the group velocity given by :

$$V(\omega) = \frac{d}{d\omega}(\omega c^*(\omega)) \quad (38)$$

Or, in terms of time Fourier transforms,

$$1/V(\Omega) = \frac{d}{d\Omega}(\Omega/c^*(\Omega)) \quad (39)$$

Applying this to (24a) (24b), one finds that the group velocity for sinusoidal solutions of wavelength near $2h$ is negative. This explains the presence of reflected perturbations when a smooth solution passes across a discontinuity or a boundary of the computational domain : since all numerical phase velocities are positive, a cursory observation of the Fourier series form of the numerical solution does not suggest that any part of it might propagate in the negative direction. But the group velocity analysis shows that reflected solutions may indeed exist in the form of wave packets of short wavelength numerical oscillations which propagate upstream of the flow after having been spuriously generated.

The mathematics of propagation in dispersive media and many of the attending developments in Fourier analysis had been well established by the end of the nineteenth century. The concept of group velocity had been used conceptually by Hamilton (1839), was known to Rayleigh (Theory of Sound - 1877) and was fully investigated in the early part of this century with publications of Sommerfeld (1912, 1914) being prominent. An important class of dispersive media is that of crystals, which present periodic structures to the propagation of light. Of interest is to note the fact that numerical discretizations of hyperbolic equations have created new families of similarly periodic structures. It comes

therefore as no surprise that applying the same analysis to those discretizations proves to be a very fruitful endeavor.

12. REFLECTION PHENOMENA

The appropriate tool for the analysis of spurious reflection phenomena in discretizations of hyperbolic equations is that given by time Fourier transforms. If we denote by $\{\hat{u}_n(\Omega)\} = \int \{u_n(t)\} e^{-i\Omega t} dt$ (40)

the set of time Fourier transforms of a solution of (16a) or (16b), then an analytic treatment reveals the following (Vichnevetsky 1981, a, b, 1983 a) :

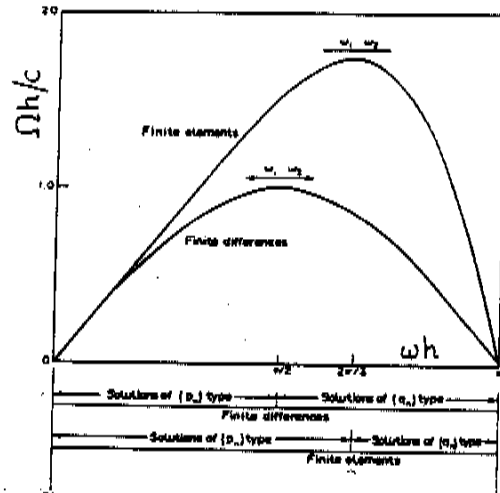


Figure 7 - Dispersion relation (frequency Ω versus wave number ω) To each $\Omega < \Omega_c$ correspond two wave numbers ω_1 and ω_2 which correspond to one another in reflection phenomena. (From reference 62).

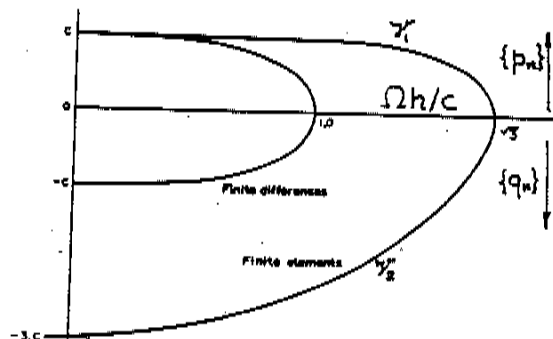


Figure 8 - Group velocities. To each frequency Ω below the cutoff frequency corresponds a positive group velocity (solution of $\{p_n\}$ type) and a negative group velocity (solution of $\{q_n\}$ type).

Any solution may be decomposed into the sum of two fundamental types

$$\{u_n(t)\} = \{p_n(t)\} + \{q_n(t)\} \quad (41)$$

whose Fourier transforms satisfy :

$$\hat{p}_{n+1}/\hat{p}_n = \hat{E}_1(\Omega) ; \hat{q}_{n+1}/\hat{q}_n = \hat{E}_2(\Omega) \quad (42)$$

where \hat{E}_1 and \hat{E}_2 are cell transfer functions of the semi-discretization, and are given in reference (63).

While solutions of the $\{p_n\}$ type have positive phase and group velocities, those of $\{q_n\}$ type have positive phase but negative group velocities : solutions of $\{p_n\}$ type are convergent approximations of genuine solutions, but solutions of $\{q_n\}$ type are entirely spurious. They carry energy upstream of the flow and are precisely the kind of extraneous solutions that are generated when spurious reflection occurs at boundaries of computational domains and at interfaces in numerical mesh refinement.

13. REFLECTION AT BOUNDARIES

When the simple advection equation (15) is approximated over a finite domain (say $x \in [0, l]$), the approximation at the downstream boundary is different from the approximation in the interior points. This creates spurious reflection. Consider, for instance, the simple upstream approximation in

$$\frac{du_n}{dt} = -c \left(\frac{u_n - u_{n-1}}{h} \right) ; (u_n \approx U(l, t)) \quad (43)$$

Observe that :

(i) any incident solution arriving in $x = l$ from $x < 0$ is of $\{p_n\}$ type, since it carries energy downstream.

(ii) the corresponding reflected solution must be of $\{q_n\}$ type, since it carries energy upstream

(iii) to a component of frequency Ω in the incident $\{p_n\}$ correspond a component at the same frequency Ω in the reflected $\{q_n\}$. To quantify the reflection, we take the t-Fourier transform of (43) :

$$i\Omega \hat{u}_n = -\frac{c}{h} (\hat{u}_n - \hat{u}_{n-1}) \quad (44a)$$

which together with

$$\hat{p}_{n-1} = \hat{E}_1^{-1} \hat{p}_n ; \hat{q}_{n-1} = \hat{E}_2^{-1} \hat{q}_n \quad (44b)$$

gives the reflection ratio :

$$r(\Omega) = \frac{\hat{q}_n}{\hat{p}_n} = -\frac{1 + i\Omega h/c - \hat{E}_1^{-1}}{1 + i\Omega h/c - \hat{E}_2^{-1}} = O(\Omega^2 h^2) \quad (45)$$

Better treatments of the boundary may be sought by replacing (43) with improved expressions aimed at making $r(\Omega)$ as small as possible. This has been a subject of considerable interest in recent years .

Less obvious is the question of spurious reflection at an upstream or inlet boundary. The usual numerical treatment consists in letting

$$u_n(t) = U(0, t) \quad (46)$$

i.e. letting the numerical value be equal to the imposed value. But this creates reflection when a spurious solution (of $\{q_n\}$ type) arrives at the boundary. An improved treatment should attempt to absorb these spurious solutions instead of reflecting them. Appropriate non-reflecting inlet boundaries for finite difference discretizations are given in Vichnevetsky, Sciubba and Pak (1981).

14. REFLECTIONS IN MESH REFINEMENT

Mesh refinement is commonly used when a greater accuracy is sought in some subdomain of the problem. A typical example is that of grids used in numerical weather prediction. Whereas coarse grids may be used over the oceans, a fine grid is desirable over land where the predictions count. A simple model of mesh refinement is illustrated in figure 10. The mathematics for the description of reflection at the interface are similar to (44), leading to the reflection ratio for the spurious solution reflected toward $x < 0$:

$$r(\Omega) = \frac{\sqrt{1 - (\Omega h/C)^2} - \sqrt{1 - (\Omega h/\epsilon)^2}}{\sqrt{1 - (\Omega h/C)^2} + \sqrt{1 - (\Omega h/C)^2}} = O(\Omega^2 (h^2 - \epsilon^2))$$

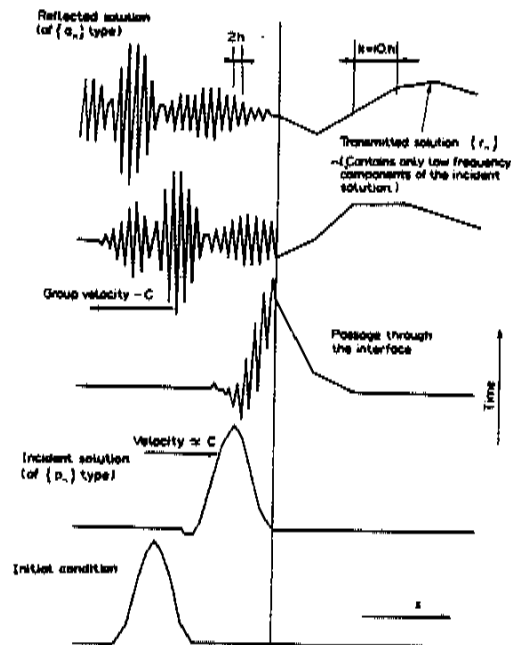


Figure 9 - Reflection of a smooth function at a mesh coarsening interface. The reflected solution is a wave packet of wave length near $2h$.

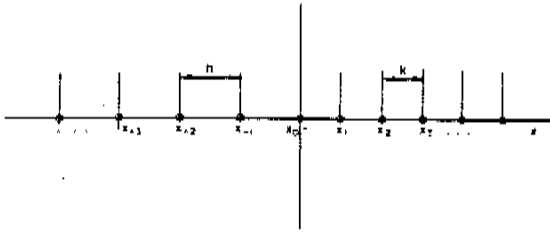


Figure 10 - Mesh refinement.

15. NUMERICAL WAVE PROPAGATION IN 2 DIMENSIONS - ANISOTROPY

The wave equation in two space dimensions

$$\frac{\partial^2 U}{\partial t^2} = c^2 \left(\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} \right) \quad (47)$$

has propagation characteristics which are isotropic : a plane sinusoidal wave

$$U = e^{i\omega(x \cos \alpha + y \sin \alpha \pm ct)} \quad (48)$$

propagates in the 1_α direction at the phase velocity c which is independent of α . By contrast, numerical simulations on a discrete grid introduce, in addition to dispersion (c^* depending on ω) also anisotropy (c^* depending on α). Consider, for instance, the simple semi-discretization of (47) on a square grid

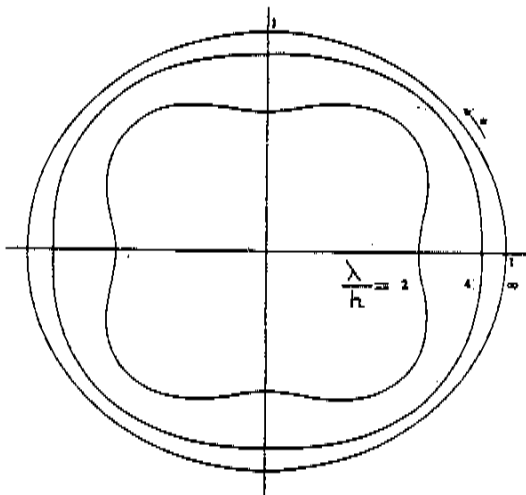


Figure 11 - Anisotropy : phase velocity of the semi-discretization (49) versus direction (α) and wave length (λ).

$$\frac{d^2 u_{m,n}}{dt^2} = c^2 \left[\left(\frac{u_{m+1} - 2u_m + u_{m-1}}{h^2} \right)_n + \left(\frac{u_{n+1} - 2u_n + u_{n-1}}{h^2} \right)_m \right] \quad (49)$$

If we insert the expression of a plane sinusoidal trial solution such as (48), we find, after integration

$$u_{m,n} = e^{i\omega(x_m \cos \alpha + y_n \sin \alpha \pm c^*(\omega, \alpha) t)} \quad (50)$$

where the numerical phase velocity is :

$$c^*(\alpha, \omega) = c \frac{\sqrt{2}}{\omega h} \left(2 - \cos(\omega h \cos \alpha) - \cos(\omega h \sin \alpha) \right) \quad (51)$$

(see also Birkhoff and Dougalis (1975)).

Anisotropy properties of general explicit and implicit semi-discretizations of the wave equation on a square grid are given in Vichnevetsky and Bowles (1982). Similar results for finite element discretizations (bilinear elements on rectangles and linear elements on triangles) are given by Mullen and Belytschko (1982). Anisotropy of phase velocity implies of course anisotropy of group velocity as well. These properties have been analysed by Trefethen (1982...) for semi- and full discretizations. - figure below:

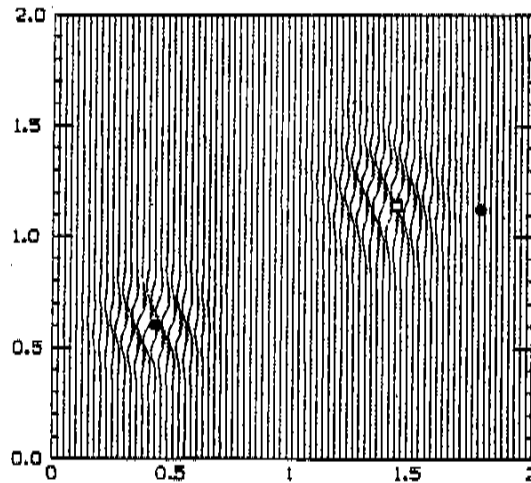


Figure 12 - Propagation of a wave packet in a discretization of the wave equation on a square grid.

● is the ideal position of the packet and □ is the position predicted by the group velocity analysis. (from reference (57) by permission)

16. ACKNOWLEDGMENTS

The assistance of Elissa C. Pariser in the preparation of this paper is gratefully acknowledged.

17. BIBLIOGRAPHY AND REFERENCES

1. Abbot, M. B. "Computational hydraulics, A short pathology." *J. Hydraulic Res.* 14 (1976), 271-285.
2. Belyushko, T. and T. Mullen. "On dispersive properties of finite element solutions". In *Modern Problems in Elastic Wave Propagations*, I. Miklowitz and J. D. Achenbach, Eds., J. Wiley and Sons, 1978.
3. Benyon, P. "A review of numerical methods for simulation." *Simulation* 11 (1968).
4. Bernoulli, Daniel. "Reflexions et éclaircissemens sur les nouvelles vibrations des cordes." *Mem. Acad. Sci. Berlin* 9 (1753a), 147-172. (1753: publ. 1755)
5. Bernoulli, Daniel. "Sur les mélange de plusieurs especes de vibrations simples isochrones." *Mem. Acad. Sci. Berlin* 8 (1753b), 173-195. (1753: publ. 1755)
6. Birkhoff, G. and V. A. Dougalis. "Numerical solution of hydrodynamic problems". In *Advances in Computer Methods for Partial Differential Equations*, R. Vichnevetsky, Ed., AICA, Rutgers University, New Brunswick, NJ, 1975.
7. Birkhoff, G. "Numerical fluid dynamics." *SIAM Review* 25, 1 (1983), 1-34.
8. Brigham, E. O. *The Fast Fourier Transform*. Prentice-Hall, Englewood Cliffs, NJ, 1974.
9. Brillouin, L. *Wave Propagation in Periodic Structures*. McGraw Hill, New York, 1946.
10. Brillouin, L. *Wave Propagation and Group Velocity*. Academic Press, New York, 1960.
11. Buneman, O. "A compact non-iterative Poisson solver." *SUIPR Report* 294 (1974).
12. Carver, M. B. and H. W. Hinds. "The method of lines and the advection equation." *Simulation* 31 (1978), 59-69.
13. Chin, R. C. Y., G. W. Hedstrom, and K. E. Karlsson. "A simplified Galerkin method for hyperbolic equations." *Math. Comput.* 33 (1979), 647-658.
14. Churchill, R. V.. *Fourier Series and Boundary Value Problems*. McGraw Hill, New York, 1941. (revised edition 1963)
15. Clymer, A. B. "Methods of simulating structural dynamics." *Simulation* 7 (1966).
16. Colony, R. and R. R. Reynolds. "An application of Hockney's method for solving Poisson's equation." *Proc. Spring Joint Comp. Conf., AFIPS* (1970).
17. Cooley, J. W. and J. W. Tukey. "An algorithm for the machine calculation of complex Fourier series." *Math. Comput.* 19 (1965), 297-301.
18. Dorr, F. W. "The direct solution of the discrete Poisson equation on a rectangle." *SIAM* 12, 8 (March 1970), 248-263.
19. Dougalis, V. A. and G. Birkhoff. "A comparison of numerical methods for solving wave equations". In *First International Congress on Numerical Ship Hydrodynamics*, J. W. Schot and N. Salvesen, Eds., U.S. Naval R&D Center, (D. W. Taylor Model Basin), 1975.
20. Dougalis, V. A. and S. M. Serbin. "On the efficiency of some fully discrete Galerkin methods for second order hyperbolic equations." *Computers and Math with Appl.* (1981).
21. Engquist, B. and A. Majda. "Absorbing boundary conditions for the numerical simulation of waves." *Math. Comp.* 31, 139 (July 1977), 629-651.
22. Fix, G. and G. Strang. "Fourier analysis of the finite element method in Ritz-Galerkin theory." *Studies in Appl. Math* 48 (1969), 265-273.
23. Fornberg, B. "On a Fourier method for the integration of hyperbolic equations." *SIAM J. Num. Anal.* 12 (1975), 509-528.
24. Fourier, J. "Theorie de la propagation de la chaleur dans les solides." *Monograph presented to the Institut de France* (1807). reproduced in Grattan-Guinness (1972)
25. Fromm, J. E. "A method for reducing dispersion in convective difference schemes." *J. Comp. Phys.* 3 (1968), 176-189.
26. Gentleman, W. M. and G. Sande. "Fast Fourier Transforms - for fun and profit". Proc. Spring Joint Comp. Conf., AFIPS, 1966".
27. Gilliland, M. C. "A spectral stability analysis of finite-difference operations." *IEEE Trans. Electronic Computers EC-15* (1966), 849-854.
28. Gottlieb, D. and S. A. Orszag. "Numerical Analysis of Spectral Methods". In *CBMS Regional Conference Series*, SIAM, Philadelphia, 1977.
29. Grattan-Guinness, I. *The Development of the Foundations of Mathematical Analysis from Euler to Riemann*. MIT Press, Cambridge, MA., 1970.
30. Grattan-Guinness, I. *Joseph Fourier 1768-1830 - His Life and Work*. MIT Press, Cambridge, MA, 1972.
31. Gray, W. G., and G. F. Pinder. "An analysis of the numerical solution of the transport equation." *Water Resources Res.* 12, 3 (1976), 547-555.
32. Hamming, R. W.. *Digital Filters*. Prentice Hall, Englewood Cliffs, NJ, 1977.
33. Hockney, R. W. "A fast direct solution of Poisson's equation using Fourier analysis." *JACM* 12, 1 (1965), 95-113.
34. Hyman, J. M. "A method of lines approach to the numerical solution of conservation laws". In *Advances in Computer Methods for Partial Differential Equations*, R. Vichnevetsky, Ed., IMACS, New Brunswick, NJ, 1979, pp. 313-321.
35. Kaplan, S., O. J. Marlowe, and J. Berwick. "Applications of synthesis techniques to problems with time-dependence." *Nuclear Science and Engineering* 18 (1964), 163-176.
36. Kreiss, H. O. and J. Olliger. "Comparison of accurate methods for the integration of hyperbolic equations." *TELLUS XXIV* (1972), 119-215.

37. Kreiss, H. O. and J. Olinger. "Methods for the approximate solution of time dependent problems." *World Meteorological Organization/International Council of Scientific Unions* (1973).
38. Krieg, D. R. and S. W. Key. "Transient shell response by numerical time integration." *Internat. J. for Num. Meth. in Eng.* 7 (1973), 273-286.
39. Lagrange, J. L. "Recherches sur la nature de la propagation du son." *Miscell. Taurin* 1 (1759), also in *Works*, Vol. 1, pp. 39-148
40. Lanczos, C. *Applied Analysis*. Prentice Hall, Englewood Cliffs, NJ, 1956.
41. Lapidus, L. and G. F. Pinder. *Numerical Solutions of Partial Differential Equations in Science and Engineering*. J. Wiley and Sons, 1982.
42. Le Bail, R. C. "Use of Fast Fourier Transforms for solving partial differential equations in physics." *J. Comp. Phys.* 9 (1972), 440-465.
43. Miranker, W. L. "Difference schemes with best possible truncation error." *Num. Math.* 17 (1971), 124-142.
44. Mullen, R. and I. Belytschko. "Dispersion analysis of finite element semidiscretizations of the two-dimensional wave equations." *Internat. J. for Num. Meth. in Eng.* 18 (1982), 11-29.
45. O'Brien, G., M. A. Hyman, and S. Kaplan. "A study of the numerical solution of partial differential equations." *J. Math and Physics* (1952), 223 ff..
46. Orszag, S. "Numerical simulation of incompressible flow within simple boundaries: Accuracy." *J. Fluid Mech.* 49 (1971), 75 ff..
47. Papoulis, H. *The Fourier Integral and Its Application*. McGraw Hill, New York, 1962.
48. Peskin, R. L. and A. D. Kowalski. "A Simulation of Large Scale Atmospheric Turbulence and Diffusion". Proceedings Sixth Symposium on Turbulence and Diffusion, 1983.
49. Proskurowski, W. "Capacitance Matrix Methods - Available software and possible extensions". In *Proceedings, 10th IMACS World Congress*, IMACS, New Brunswick, NJ, 1982.
50. Lord Rayleigh, 3rd Baron (J. W. Strutt). *The Theory of Sound*. Macmillan, London, 1894. reprinted by Dover, New York, 1945
51. Roache, P. J.. *Computational Fluid Dynamics*. Hermosa Publisher, Albuquerque, NM, 1972.
52. Roberts, R. V. and N. O. Weiss. "Convective difference schemes." *Math. Comput.* 20 (1966), 272-299.
53. Rosenbrock, H. H. "Distinctive problems of process control." *Chem. Eng. Progress* 9 (1962), 43-50.
54. Shannon, C. E. and W. Weaver. *The Mathematical Theory of Communication*. University of Illinois Press, Urbana, 1949.
55. Stacey, W. M. Jr.. *Modal Approximations*. MIT Press, Cambridge, MA, 1967.
56. Strang. "Trigonometric polynomials and difference methods of maximum accuracy." *J. Math and Phys.* 41 (1962), 147-354.
57. Swartz, B. "Comparing certain classes of difference and finite element methods for a hyperbolic problem". In *Advances in Computer Methods for Partial Differential Equations*, R. Vichnevetsky, Ed., AICA, Rutgers University, New Brunswick, NJ, 1975.
58. Trefethen, L. F. "Group velocity in finite difference schemes." *SIAM Review* 24 (1982), 113-136.
59. Vichnevetsky, R. "Physical criteria in computer methods for partial differential equations". Proc. 7th AICA International Congr., Prague, 1973. reprinted in Proc. AICA, XVI, January, 1974. Brussels.
60. Vichnevetsky, R. "Stability charts in the numerical approximation of partial differential equations: a review." *Math Comput. Simulation* 21 (1979), 170-177.
61. Vichnevetsky, R. "Energy and group velocity in semi-discretizations of hyperbolic equations." *Math Comput. Simulation* 23 (1981a), 333-343.
62. Vichnevetsky, R. "Propagation through numerical mesh refinement for hyperbolic equations." *Math Comput. Simulation* 23 (1981b), 344-353.
63. Vichnevetsky, R.. *Computer Methods for Partial Differential Equations*. Prentice Hall, Englewood Cliffs, NJ, 1981c.
64. Vichnevetsky, R. "Group Velocity and Reflection Phenomena in Numerical Approximations of Hyperbolic Equations." *Journal of the Franklin Institute* (1983).
65. Vichnevetsky, R. and J. B. Bowles. *Fourier Analysis of Numerical Approximations of Hyperbolic Equations*. SIAM, Philadelphia, PA, 1982.
66. Vichnevetsky, R. and F. De Schutter. "A frequency analysis of finite element methods for initial value problems". In *Advances in Computer Methods for Partial Differential Equations*, R. Vichnevetsky, Ed., AICA, Rutgers University, New Brunswick, NJ, 1975.
67. Vichnevetsky, R. and B. Peiffer. "Error waves in finite element and finite difference methods for hyperbolic equations". In *Advances in Computer Methods for Partial Differential Equations*, R. Vichnevetsky, Ed., AICA, Rutgers University, New Brunswick, NJ, 1975.
68. Vichnevetsky, R., E. Sciubba and Y. Pak. "Non-reflecting upstream boundary conditions". In *Advances in Computer Methods for Partial Differential Equations / V*, R. Vichnevetsky, Ed., IMACS, Rutgers University, New Brunswick, NJ, 1981.
69. Vichnevetsky, R. and A. W. Tomalesky. "Spurious error waves in numerical approximations of hyperbolic equations". Proc. 5th Princeton Conference on Information Science and Systems, 1971.
70. Vliegthart, A. C. "Dissipative difference schemes for shallow water equations." *J. Engrg. Math.* 3 (1969), 81-94.
71. Von Neumann, J. and R. D. Richtmyer. "A method for the numerical calculation of hydrodynamic shocks." *J. Appl. Phys.* 21 (1950), 232-237.
72. Wesseling, P. "Accuracy of the third-order predictor-corrector difference schemes for hyperbolic problems." *AIAA J.* 10 (1972), 948-949.