Note

A More Accurate Method for the Numerical Solution of Nonlinear Partial Differential Equations

INTRODUCTION

Numerical solutions of nonlinear partial differential equations are inherently inaccurate due both to round-off errors and to the fact that they can only approximate the true differentials. In this paper we propose a general two-step method which reduces the latter source of error by partially solving the equations analytically rather than totally numerically. Recent studies [1, 2] performed independently from the present study discuss in detail the additive splitting of hyperbolic partial differential equations in order to solve such equations. In our investigation we apply such a technique to a simple model in order to illustrate the value of solving, at each time step, part of the system analytically, and then using this result in the numerical computation needed to complete the time step.

As a concrete example of our method we shall concentrate on solving the system

$$u_t = -uu_x - gh_x, \qquad h_t = -uh_x - hu_x. \tag{1}$$

These are the equations for one-dimensional fluid flow in a shallow tank, where u is the velocity in the positive x direction, h is the height of the fluid surface above the tank bottom, and g is a constant (gravity).

As an illustration of the type of error we hope to reduce, we look at the linearized version of (1)

$$u_t' = -gh_x', \qquad h_t' = -h_0 u_x'. \tag{2}$$

Here we have let $u(x, t) = u_0 + u'(x, t)$ and $h(x, t) = h_0 + h(x, t)$ and assumed $u_0 \equiv 0$.

If we impose initial conditions of

$$h'(x, 0) = \cos(kx) \qquad \text{and} \qquad u'(x, 0) = (g/h_0)^{1/2} h'(x, 0) \tag{3}$$

and periodic boundary conditions (i.e., no reflection), we can solve (2) analytically as

$$h'(x, t) = \cos(kx - \omega t), \qquad u'(x, t) = (g/h_0)^{1/2} h'(x, t). \tag{4}$$

Here $k = 2\pi/(\lambda \Delta x)$ is the wave number (where $\lambda \Delta x$ is one wavelength) and $\omega = 2\pi/P$, where the period P is the time required for a wave to traverse the distance $\lambda \Delta x$.

In [3] the finite difference version of (2) is analyzed as to its accuracy. The finite difference approximation does not introduce any amplification factor but does exhibit a pronounced phase error. This is expressed as the ratio of calculated (c) to exact (c_a) wave speeds:

$$c/c_a = (\pm k\, \Delta t (gh_0)^{1/2})^{-1} \sin^{-1}(\gamma (4 - \gamma^2)^{1/2}/2)$$

is one of three equivalent expressions, where $\gamma^2 = gh_0 (\Delta t/\Delta x)^2 \sin^2 k\, \Delta x$. This error can be appreciable; for the parameters used in our examples of the next section this ratio would be .901.

Admittedly, a simple finite-difference scheme may not be the most accurate numerical method for (2), but we shall see that our two-step scheme performs well even when its numerical portion is done using finite differences. Intuitively, we expect that the sort of error encountered in the numerical solution of (2) also contributes to the error of the numerical solution of (1), and that techniques which improve the accuracy of (2) have the same effect on (1). Our two-step method makes use of this idea.

NUMERICAL EXPERIMENTS

In the numerical work reported here we use initial condition (3) with $g = 10$ and $h_0 = 1.6$, a $\Delta t/\Delta x$ ratio of 0.125 $(gh_0)^{-1/2}$ unless otherwise stated, and a wavelength of 8 Δx. We drop the primes from u' and h'. All calculations were coded in Fortran 77 and run on a VAX 11/780 minicomputer.

In order to evaluate the accuracy of our proposed scheme vis-à-vis a totally numerical technique, we would like to know the exact solution to (1). The best we can do is compute a highly accurate solution (which we shall call "exact") using a spectral method.

To effect this we express u and h in terms of truncated Fourier series. Our periodicity of 8 Δx dictates the form

$$u = \sum_{n=1}^{3} (a_n \cos nx + b_n \sin nx) + a_4 \cos 4x + b_4 \cos 8x, \qquad (5)$$

where a number of the trigonometric terms are skipped because they add no new information on the eight grid points; h is expressed analogously.

For use in the spectral scheme, (1) becomes

$$u^{\tau+1} = u^\tau - \Delta t(u^\tau u_x^\tau - gh_x^\tau), \qquad (6a)$$

$$h^{\tau+1} = h^\tau - \Delta t(u^{\tau+1} h_x^\tau - hu_x^{\tau+1}), \qquad (6b)$$

where the superscripts denote the time step. At each time step we calculate $u^{\tau+1}$ and

$h^{\tau+1}$ for the eight grid points then fit these new values to series of form (5) to get new coefficients, thus always working with 8-term series at each step.

For increased accuracy we reduce the time step by a factor of five compared to that specified above. These "exact" results are graphed in Fig. 1.

A spectral method is generally too expensive for detailed geophysical calculations. More practical is the simple finite difference approximation to (1) given by the following equations:

$$u_j^{\tau+1} = \frac{\Delta t}{\Delta x}\left[-u_j^{\tau}(u_j^{\tau}-u_{j-1}^{\tau})-\frac{g}{2}(h_{j+1}^{\tau}-h_{j-1}^{\tau})\right]+u_j^{\tau}, \qquad \text{if} \quad u_j^{\tau}>0,$$

$$= \frac{\Delta t}{\Delta x}\left[-u_j^{\tau}(u_{j+1}^{\tau}-u_j^{\tau})-\frac{g}{2}(h_{j+1}^{\tau}-h_{j-1}^{\tau})\right]+u_j^{\tau}, \qquad \text{if} \quad u_j^{\tau}\leqslant 0;$$
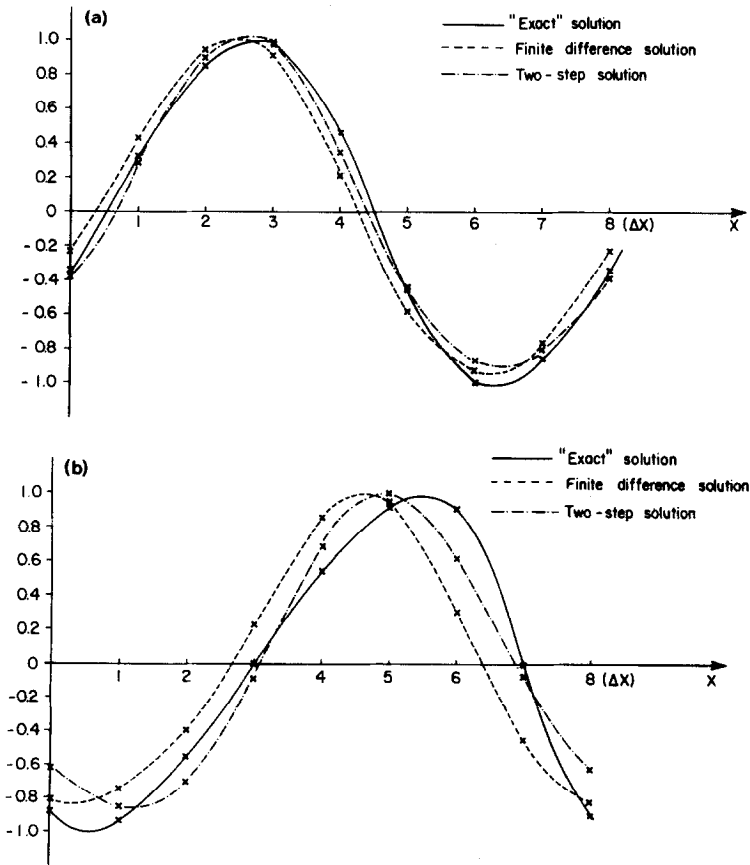
(7a)

FIG. 1. Wave amplitude h as computed by the compared methods (wave speed u behaves similarly, (a) after 20 time steps, (b) after 40 time steps.

$$h_j^{\tau+1} = \frac{\Delta t}{\Delta x}\left[-u_j^{\tau+1}(h_j^{\tau} - h_{j-1}^{\tau}) - h_j^{\tau}\frac{1}{2}(u_{j+1}^{\tau+1} - u_{j-1}^{\tau+1})\right] + h_j^{\tau}, \qquad \text{if} \quad u_j^{\tau+1} > 0,$$

$$\tag{7b}$$

$$= \frac{\Delta t}{\Delta x}\left[-u_j^{\tau+1}(h_{j+1}^{\tau} - h_j^{\tau}) - h_j^{\tau}\frac{1}{2}(u_{j+1}^{\tau+1} - u_{j-1}^{\tau+1})\right] + h_j^{\tau} \qquad \text{if} \quad u_j^{\tau+1} \leqslant 0.$$

The superscripts here refer to the time step and the subscripts to the grid point. The solution as computed by (7) is also plotted in Fig. 1.

With the two-step method of solution we first solve linear system (2) analytically. The result is (4), which we propagate by one time step, giving us intermediate solution values denoted by h^* and u^*,

$$h^* = \cos(kx - \omega \Delta t), \qquad u^* = (g/h_0)^{1/2} h^*. \tag{8}$$

As the second step we use this intermediate solution in the numerical computation necessary to complete the time step. Analogous to (7) we have

$$u_j^{\tau+1} = \frac{\Delta t}{\Delta x}\left[-u_j^{\tau}(u_j^{\tau} - u_{j-1}^{\tau})\right] + u^*, \qquad \text{if} \quad u_j^{\tau} > 0,$$

$$\tag{9a}$$

$$= \frac{\Delta t}{\Delta x}\left[-u_j^{\tau}(u_{j+1}^{\tau} - u_j^{\tau})\right] + u^*, \qquad \text{if} \quad u_j^{\tau} \leqslant 0;$$

$$h_j^{\tau+1} = \frac{\Delta t}{\Delta x}\left[-u_j^{\tau+1}(h_j^{\tau} - h_{j-1}^{\tau}) - h_j^{\tau}\frac{1}{2}(u_{j+1}^{\tau+1} - u_{j-1}^{\tau+1})\right] + h^*, \quad \text{if} \quad u_j^{\tau+1} > 0,$$

$$\tag{9b}$$

$$= \frac{\Delta t}{\Delta x}\left[-u_j^{\tau+1}(h_{j+1}^{\tau} - h_j^{\tau} - h_j^{\tau}\frac{1}{2}(u_{j+1}^{\tau+1} - u_{j-1}^{\tau+1})\right] + h^*, \quad \text{if} \quad u_j^{\tau+1} \leqslant 0.$$

Note that except for u^* and h^* all the terms on the right-hand sides of (9a) and (9b) are nonlinear.

The solution as computed with scheme (9) is also displayed in Fig. 1. It is apparent that the two-step solution eliminates much of the phase error found in the straight finite-difference solution. Both the finite-difference and two-step solutions show an asymmetry (overshoot of positive peak and spreading in the trough) that is not true to the exact solution.

The two-step method is also competitive with the straight numerical computation as far as computer time goes. Compilation time for (7) was 3.2, 3.3, and 3.4 seconds of CPU in three separate trials, while (9) required 4.3, 5.1, and 5.2 seconds. To run (7) through 65 time steps took 0.4, 0.5, and 0.6 seconds; 65 steps of (9) used 0.4, 0.6, and 0.6. Thus the run times are essentially equivalent while the difference in complilation time is of no practical concern.

GENERALIZATION AND THEORY

The two-step procedure outlined above is valid for a large class of problems. Suppose we have a system of the form

$$\frac{\partial \mathbf{y}(x_1, ..., x_n, t)}{\partial t} = A\mathbf{y}(x_1, ..., x_n, t), \tag{10}$$

where A is some operator on an $(n + 1)$-dimensional complex-valued vector space and $\mathbf{y}(x_1, ..., x_n, t)$ is an $n + 1$ vector. We split A into the sum of operators L and N, where L is such that

$$\partial \mathbf{y}/\partial t = L\mathbf{y}, \tag{11}$$

which with chosen initial and boundary conditions, can be solved analytically. Operator L must be a closed operator defined on a dense linear subspace of a Banach space B. We interpret $L\mathbf{y}$ as being the linear terms from $A\mathbf{y}$, while $N\mathbf{y}$ consists of the nonlinear terms, although this labeling is not critical.

Thus, (10) becomes

$$\partial \mathbf{y}/\partial t = L\mathbf{y} + N\mathbf{y}. \tag{12}$$

This sort of equation can be studied with functional analysis techniques; see, for instance [4]. We proceed formally to solve (12) with a variation of parameters argument.

We first look to solve homogeneous equation (11) with the given initial and boundary conditions. In our two-step method we deliberately choose L such that (11) has an analytic solution which we may denote $\mathbf{y}_a(t)$. Equation (8) then corresponds to $\mathbf{y}_a(t + \Delta t)$ which we denote $\mathbf{y}^*$.

If we also solve (11) formally we find

$$\mathbf{y}_a = e^{(t - t_0)L}\mathbf{y}_0, \tag{13}$$

where t_0 is some initial time and $\mathbf{y}_0$ is the solution at t_0. The exponential factor is well defined for L as specified; e^{tL} is then a bounded-operator-valued function on B. (For background, see [5, Chap. VIII].)

Following the lead of the classic variation of parameters derivation we now replace $\mathbf{y}_0$ with $\mathbf{y}_0(t)$ and use (13) in (12). The resulting expression for the particular solution is

$$\mathbf{y}_p = \int_{t_0}^{t} e^{(t - \tau)L} N\mathbf{y}(\tau) \, d\tau \tag{14}$$

(This is well defined whenever $N\mathbf{y}$ is an integrable function.)

Thus, the total solution to (12) may be written as

$$\mathbf{y}(t) = \mathbf{y}_a(t) + \int_{t_0}^{t} e^{(t-\tau)L} N \mathbf{y}(\tau) \, d\tau. \tag{15}$$

So, in a given time step, we compute $\mathbf{y}(t_0 + \Delta t)$ as

$$\mathbf{y}(t_0 + \Delta t) = \mathbf{y}^* + \int_{t_0}^{t_0 + \Delta t} e^{(t_0 + \Delta t - \tau)L} N(\mathbf{y}(\tau)) \, d\tau. \tag{16}$$

Our two-step scheme is, therefore, approximating the above integral by $(\Delta t) N \mathbf{y}(t_0)$, the computing the term numerically; compare with (9). Of course, this is just the simplest such approximation of the integral in (16); one could evaluate $N \mathbf{y}$ at some midpoint rather than at t_0, or approximate the integral with three points, for example.

CONCLUSIONS

We have introduced a two-step scheme for improved accuracy in the numerical solution of nonlinear partial differential equations. In a simple experimental case this method reduced the error of a basic numerical scheme without appreciable increase in computing time.

This paper is meant to be just an introduction to the possibilities of this two-step method. As a next move it would be very illuminating to try an experiment such as the one we have performed on a solvable nonlinear P.D.E.; that is, one for which the exact solution is known in closed form. In that case we would be able to see very clearly how well the two-step version works compared to a straight numerical scheme, for a variety of such schemes. Work must also be done with real-life problems to see what difficulties arise in actually implementing a two-step scheme with more involved equations.

Mathematically there is much analysis to be done, complicated by the fact that most of it is nonlinear analysis. We need to know sources and magnitudes of errors with the two-step scheme, and optimum approximations for the integral in (16) in order to minimize errors. There are also questions of convergence of the method. It may be that with certain approximations to the integral in (16) we could increase the size of our time steps significantly.

The method proposed here may have utility for a wide range of geophysical problems including numerical weather prediction. By computing a linear solution at each time step, and only evaluating the nonlinear components numerically, a major source of computational error may be avoided.

REFERENCES

1. R. J. LeVeque and J. Oliger, "Numerical Methods Based on Additive Splittings for Hyperbolic Partial Differential Equations," Numerical analysis project manuscript NA-81-16, Computer Science Department, Stanford University, Stanford, Calif., October 1981.
2. K. L. LeVeque, "Time Split Methods for Partial Differential Equations," Ph. D. Thesis, Computer Science Department, Stanford University, Stanford, Calif.. 1982.
3. R. A. Pielke, "Numerical Mesoscale Modeling," Academic Press, New York, 1983, in preparation.
4. M. Schechter, "Modern Methods in Partial Differential Equations," McGraw–Hill, New York, 1977.
5. N. Dunford and J. T. Schwartz, "Linear Operators, Part 1," Interscience, New York, 1967.

SCOTT WEIDMAN

Department of Applied Mathematics and Computer Science,
University of Virginia,
Charlottesville, Virginia 22901

ROGER A. PIELKE

Department of Atmospheric Science,
Colorado State University,
Fort Collins, Colorado 80523