Journal of Hydroscience and Hydraulic Engineering
Vol. 6, No. 1, July, 1988, pp.1-11

HYBRID COMPUTATION OF NONLINEAR ADVECTION-DIFFUSION

By

Keiichi Toda
New Japan Engineering Consultants, Inc.
Shimanouchi 1-Chome 20-19, Minami-Ku Osaka, 542, Japan

and

Forrest M. Holly, Jr.
Institute of Hydraulic Research, The University of Iowa
Iowa City, Iowa 52242-1585 USA

SYNOPSIS

A hybrid method comprising the Holly-Preissmann characteristics scheme for advection and the Crank-Nicholson finite-difference scheme for diffusion is presented and analyzed. This method provides accurate solutions to the one-dimensional Burger's equation. Application of the method to practical problems involving inertia-dominant flow reveals the need for additional development before the method can be considered ready for generalized industrial use.

INTRODUCTION

Numerical simulation of inertia-dominated engineering flow problems such as surges or bores in open channel flow, or shock propagation in a compressible fluid, require careful treatment of the nonlinear inertia terms in the momemtum-conservation equation. Considerable effort has been devoted to the development of simple and accurate schemes to handle such highly nonlinear problems, and many useful schemes have been obtained (see, for example Cunge et al (2); Roache (9); Lohner et al (7); Hughes (4)). However, if accuracy, brevity of the algorithm, and cost of computation are all taken into account, there is still room for improvement of existing methods and/or development of new alternative methods.

This paper explores use of a hybrid characteristics/finite-difference method to solve the nonlinear advection-diffusion problem. The Holly-Preissmann characteristics method (Holly and Preissmann (3)), based on a Hermite cubic interpolation polynomial, has proven to be a powerful tool for simulation of linear advection problems, and for linear advection-diffusion problems when combined with a Crank-Nicholson finite-difference diffusion scheme (Toda (10)). The key to this method is construction of an interpolating polynomial of higher order between only two computational points using both the dependent variable and its derivatives (which also become dependent variables at those points). The extension to nonlinear advection-diffusion is rather straightforward, the only difficulty being determination of the location of the foot of the characteristic trajectory in nonlinear advection by solving a cubic trajectory equation. The question at hand is whether the method's success for linear problems extends to nonlinear ones.

DESCRIPTION AND ANALYSIS OF METHOD

The model equation used herein is the one-dimensional Burger's equation:

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \nu\frac{\partial^2 u}{\partial x^2} \qquad (1)$$

where u is the dependent variable (usually a velocity) and ν is a diffusion coefficient. The hybrid numerical method for approximate solution of Eq. 1 is based on recognition of the left-hand side as a total derivative along a certain trajectory, and on use of a finite-difference scheme for the diffusion of the right-hand side. Equation 1 can thus be written

$$\frac{Du}{Dt} = \nu \frac{\partial^2 u}{\partial x^2} \tag{2}$$

along

$$\frac{dx}{dt} = u\ (x,t) \tag{3}$$

where $\frac{D}{Dt}$ denotes the total, or substantial, derivative. The.integration of Eq. 2 along the trajectory Eq. 3 leads to

$$u_\eta - u_\xi = \int_{t_\xi}^{t_n} \nu \frac{\partial^2 u}{\partial x^2} dt \approx \{\theta\ (\nu \frac{\partial^2 u}{\partial x^2})_\eta + (1-\theta)\ (\nu \frac{\partial^2 u}{\partial x^2})_\xi\}\ \Delta t \tag{4}$$

where ξ and η denote the foot and head of the trajectory, respectively (see Figure 1). Quantitites subscripted by η are unknowns attributed to the computational point x_i at the n+1 time-step level. θ is a weighting parameter for two-point integration of the diffusion term of Burger's equation, $0 < \theta < 1$.

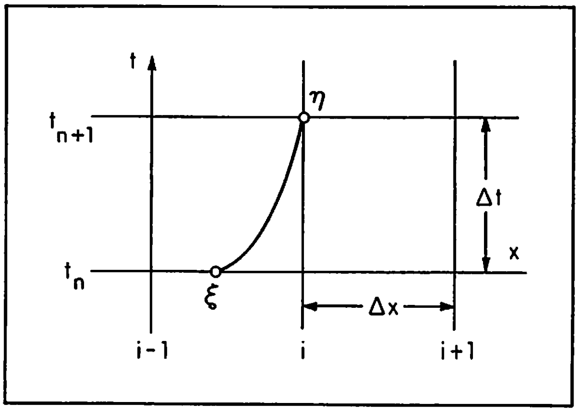

Fig. 1. Definition sketch for hybrid computational scheme.

The Holly-Preissmann method evaluates u_ξ using a cubic interpolating polynomial based on the known quantities, u^n_{i-1}, u^n_i, $u^n_{x_{i-1}}$, and $u^n_{x_i}$ (u_x denotes $\frac{\partial u}{\partial x}$):

$$u_\xi = a_0 + a_1\alpha + a_2\alpha^2 + a_3\alpha^3 \tag{5}$$

and

$$u_{x_\xi} = \frac{\partial u}{\partial x}\Big|_\xi = -\frac{1}{\Delta x}\ (a_1 + 2a_2\alpha + 3a_3\alpha^2) \tag{6}$$

where $\alpha = (x_\eta - x_\xi)/\Delta x$, and $a_o = u^n_i$, $a_1 = -u^n_{x_i}\ (\Delta x)$, $a_2 = -3u^n_i + 3u^n_{i-1} + 2u^n_{x_i}\ (\Delta x) + u^n_{x_{i-1}}\ (\Delta x)$, $a_3 = 2u^n_i - 2u^n_{i-1} - u^n_{x_i}\ (\Delta x) - u^n_{x_{i-1}}\ (\Delta x)$. These expressions for a_0-a_3 result from the requirement that for $\alpha = 0$, $u_\xi = u^n_i$, and $u_{x\xi} = u^n_{x_i}$, and for $\alpha = 1$, $u_\xi = u^n_{i-1}$ and $u_{x\xi} = u^n_{x_{i-1}}$. As this interpolation introduces u_x as a new dependent variable, u_x must also be transported. This transport equation is obtained by taking the derivative of Eq. 2:

$$\frac{\partial u_x}{\partial t} + u\frac{\partial u_x}{\partial x} = -(u_x)^2 + \nu\left(\frac{\partial^2 u_x}{\partial x^2}\right) \tag{7}$$

Again this can be rewritten, using the total derivative form,

$$\frac{Du_x}{Dt} = -(u_x)^2 + \nu\left(\frac{\partial^2 u_x}{\partial x^2}\right) \tag{8}$$

along

$$\frac{dx}{dt} = u(x,t) \tag{3}$$

Integration of Eq. 8 along the trajectory of Eq. 3 yields

$$u_{x\eta} - u_{x\xi} \approx [\theta\{-(u_x)^2 + \nu(\frac{\partial^2 u_x}{\partial x^2})\}_\eta + (1-\theta)\{-(u_x)^2 + \nu(\frac{\partial^2 u_x}{\partial x^2})\}_\xi]\,\Delta t \tag{9}$$

If one recognizes $u_\eta = u_i^{n+1}$ and $u_{x\eta} = u_{x_i}^{n+1}$, uses the Crank-Nicholson diffusion scheme for the second derivatives of u and u_x, and introduces the following linearization for $(u_x)^2$:

$$\begin{aligned}(u_x)_\eta^2 = \{u_{x\xi} + (u_{x\eta} - u_{x\xi})\}^2 &\approx (u_{x\xi})^2 + 2\,(u_{x\xi})\,(u_{x\eta} - u_{x\xi}) \\ &= 2(u_{x\xi})\,(u_{x\eta}) - (u_{x\xi})^2\end{aligned} \tag{10}$$

then the discretized forms of Eqs. 4 and 9 become

$$u_i^{n+1} - u_\xi = [\theta\nu\{\frac{u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1}}{(\Delta x)^2}\} + (1-\theta)\nu\{\frac{u_{i+1}^{n} - 2u_i^{n} + u_{i-1}^{n}}{(\Delta x)^2}\}]\Delta t \tag{11}$$

$$\begin{aligned}u_{x_i}^{n+1} - u_{x_\xi} = {} & [\theta\{-2(u_{x\xi})(u_{x_i}^{n+1}) + (u_{x\xi})^2 + \nu\,\frac{u_{x_{i+1}}^{n+1} - 2u_{x_i}^{n+1} + u_{x_{i-1}}^{n+1}}{(\Delta x)^2}\} \\ & + (1-\theta)\{-(u_{x\xi})^2 + \nu\,\frac{u_{x_{i+1}}^{n} - 2u_{x_i}^{n} + u_{x_{i-1}}^{n}}{(\Delta x)^2}\}]\Delta t\end{aligned} \tag{12}$$

Solution of the above system of equations requires determination of u_ξ and $u_{x\xi}$ for each computational grid interval. The integration of Eq. 3 yields

$$x_\eta - x_\xi = \alpha\Delta x = \int_{t_\xi}^{t_\eta} u(x,t)\,dt \approx \{\phi u_\eta + (1-\phi)u_\xi\}\Delta t \tag{13}$$

where ϕ is a weighting parameter for the trajectory integration, $0 \le \phi \le 1$. If u_η is estimated as u_ξ plus a purely explicit (θ=0) finite-difference approximation of Eq. 11, Eq. 13 becomes:

$$\alpha \frac{\Delta x}{\Delta t} = \phi\{u_\xi + \nu(\Delta t)\frac{u^n_{i+1} - 2u^n_i + u^n_{i-1}}{(\Delta x)^2}\} + (1-\phi)u_\xi$$
$$= u_\xi + \phi\nu(\Delta t)\frac{u^n_{i+1} - 2u^n_i + u^n_{i-1}}{(\Delta x)^2} \qquad (14)$$

Equation 14, with u_ξ given by Eq. 5, is a cubic equation in α. Determination of the appropriate root for α, enables computation of u_ξ and $u_{x\xi}$ for use in Eqs. 11 and 12.

When written for N-2 interior computational points of a reach having N total points, supplemented with two boundary conditions, Eqs. 11 and 12 form two systems of N equations in N unknowns u_i^{n+1} and u_{xi}^{n+1}, i = 1, 2, ...N to be solved at each new time step level, n+1. As Eqs. 11 and 12 have three unknowns, each system of equations is solved implicitly using a tri-diagonal method with appropriate upstream and downstream boundary conditions.

DEMONSTRATION AND EVALUATION OF METHOD

The hybrid method with nonlinear trajectory evaluation described in the previous section has been tested through application to three nonlinear cases having exact solutions.

The first test case (N1) has the exact solution (Lohar and Jain (6)):

$$u(x,t) = \frac{x/t}{1 + \exp(x^2/4\nu t)\,(t/t_o)^{1/2}} \qquad (15)$$

with initial condition

$$u(x,1) = \frac{x}{1 + t_o^{-1/2}\exp(x^2/4\nu)} \qquad (16)$$

where $t_o = \exp(1/8\,\nu)$, and boundary conditions are $u(0,t) = u(\infty,t) = 0$. The tests were performed for $\nu = 0.005$ and 0.0005, and $\Delta t = \Delta x = 0.01$ (all consistent units). For boundary conditions in the restricted model domain, the analytical value was imposed at the upstream (x = 0.0) and downstream (x = 1.2) limits. Physically, this situation may be thought of as an idealized shock propagating downstream subject to the above initial velocity distribution in a compressible fluid of small viscosity. When $\nu = 0.005$, the numerical results are indistinguishable from the corresponding exact solutions for any θ when $\phi = 0$, as seen in Figure 2(a). Quite similar results are obtained for $\phi>0$. For $\nu = 0.0005$, the corresponding solution is quite sensitive to the two parameters θ and ϕ, both of which have an effect on phase error. However, if the parameters are chosen in a suitable, but *ad hoc* way, the computation simulates the corresponding exact solution quite well, as shown in Figure 2(b). It is interesting to note that if Preissmann's four-point finite-difference scheme (Cunge et al (2)) is applied to this strongly nonlinear case, an unstable solution results, even for $\theta = 1$, when $\nu < 0.0015$.

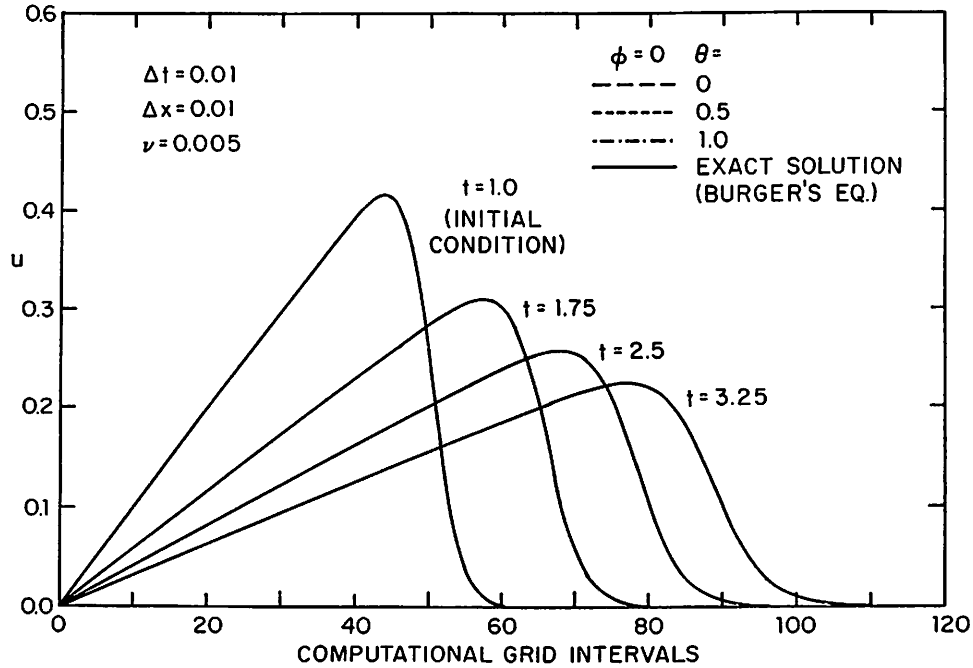

Fig. 2a. Comparison of exact and numerical solutions, nonlinear advection-diffusion (case N)

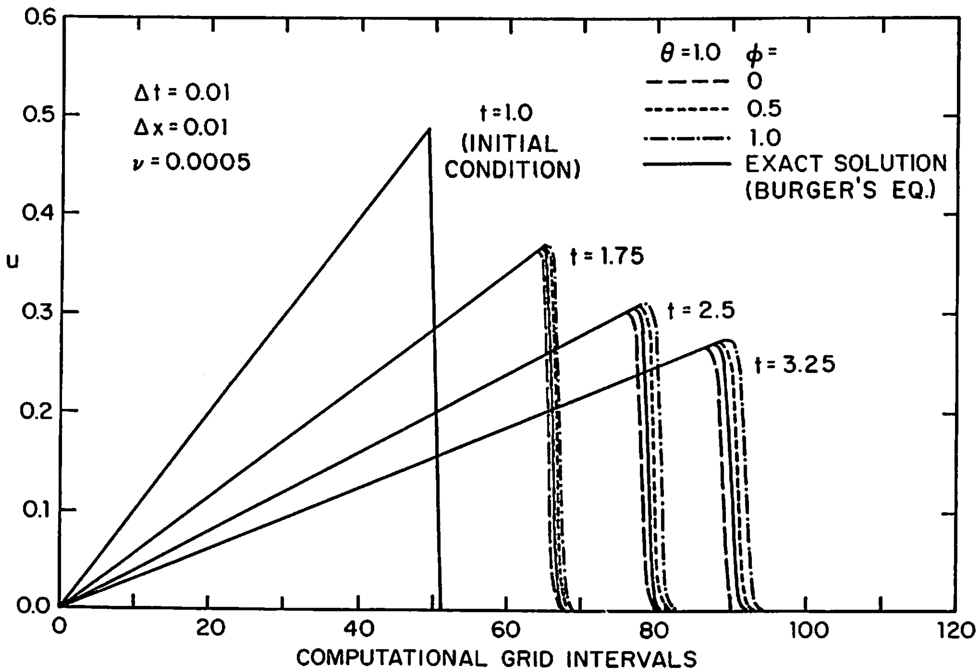

Fig. 2b. Comparison of exact and numerical solutions, nonlinear advection-diffusion (case N1)

The second test case (N2) is a problem of an advancing shock front with $\nu = 0$ (nonlinear pure-advection). The analytical solution is (Lax (5)):

$$u(x,t) = \begin{cases} 1 \text{ for } (x-x_o)/t < 1/2 \\ 0 \text{ for } (x-x_o)/t > 1/2 \end{cases} \qquad (17)$$

with initial condition

$$u(x,0) = 1 \quad \text{for } x-x_o < 0$$
$$u(x,0) = 0 \quad \text{for } x-x_o > 0 \qquad (18)$$

This case may be thought of as an idealized discontinuous shock propagating in a compressible fluid. For the numerical solution, u_x is initialized to zero at the slope discontinuity. In this case, analysis of Eq. 14 reveals that $\Delta t/\Delta x > 8/9$ is required for the front to move at all; otherwise, in the grid interval containing the advancing front, the unique real root $\alpha = 0$ disallows any movement. Also for $\Delta t/\Delta x = 1$, the solution for the cubic quation comprises three real roots, $\alpha = 0$, 0.5, and 1. The corresponding exact solution is perfectly simulated for $\alpha = 0.5$, but the front moves at twice the correct speed for $\alpha = 1$. In order to avoid these problems, a small diffusion coefficient $\nu = 0.001$ is introduced; this has the effect of relaxing the troublesome constraint on α, and letting the cubic equation have only one real root between 0 and 1, at the expense of some artificial smoothing of the numerical solution.

Several numerical experiments have been conducted by changing Δt, θ, and ϕ for a fixed grid, $\Delta x = 0.01$ and $x_o = 2\Delta x$. The corresponding results are rather sensitive to the ratio $\Delta t/\Delta x$, θ, and ϕ. Numerical stability is strongly dependent on the value of $u\Delta t/\Delta x$ (parameter analogous to the Courant number), and θ and ϕ have more effect on phase error than in case N1. Nontheless, by a suitable, but again *ad hoc*, choice of the two parameters, reasonable results can be obtained. Two examples are shown in Figures 3(a) and 3(b), where it is seen that

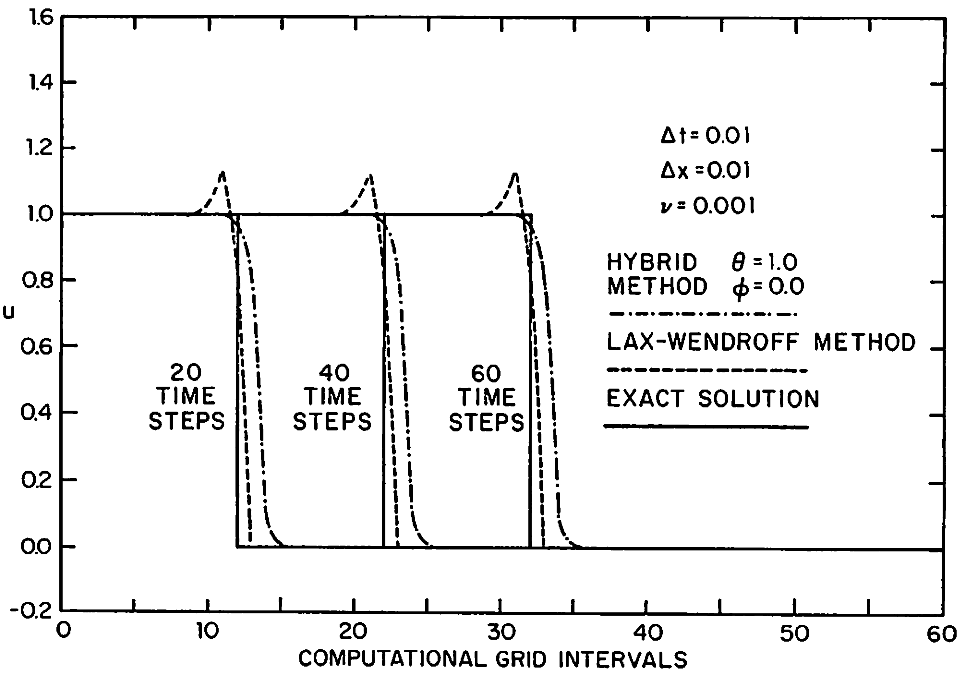

Fig. 3a. Comparison of exact and numerical solutions, nonlinear pure-advection (case N2)

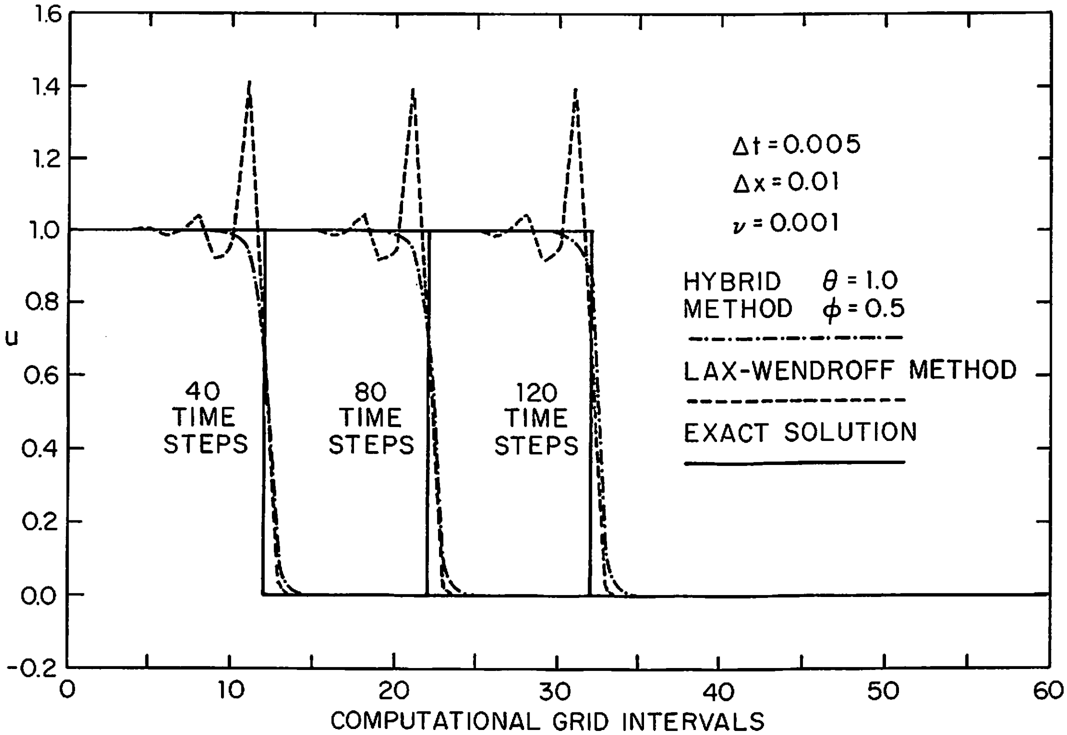

Fig. 3b. Comparison of exact and numerical solutions, nonlinear pure-advection (case N2)

amplitude error is well restrained in spite of some phase error being involved as compared with a two-step Lax-Wendroff explicit method (Richtmyer and Morton (8)). The application of the Preissmann four-point scheme to this case yields unstable results for $\nu \neq 0$ and strong oscillations for $\nu = 0$.

The third case (N3) is another nonlinear pure-advection problem. The analytical solution is (Lax (5)):

$$u(x,t) = \begin{cases} 0 & \text{for } x-x_o<0 \\ (x-x_o)/t & \text{for } 0<x-x_o<t \\ 1 & \text{for } t>x-x_o \end{cases} \qquad (19)$$

with initial condition

$$\begin{aligned} u(x,0) &= 0 \quad \text{for} \quad x-x_o<0 \\ u(x,0) &= 1 \quad \text{for} \quad x-x_o>0 \end{aligned} \qquad (20)$$

In this case, reasonable results are obtained without introducing artificial diffusion. Figure 4 shows the computed results for $\Delta t = \Delta x = 0.01$ and $x_o = 0$. The Lax-Wendroff explicit method is also shown for comparison. The simulated results show that the hybrid method gives almost the same accuracy as the Lax-Wendroff method, although some phase error is seen. The parameter ϕ is irrelevant, since the second term of the right-hand side of Eq. 14 is eliminated. It was also determined that θ has relatively little effect on the computed results for this problem.

Another hybrid method for the full de St. Venant equations of unsteady open-channel flow was developed by treating the momentum-advection term with the compact Holly-Preissmann characteristics approach, using Preissmann's finite-difference method for the remaining terms. In tests of the hybrid method for surge

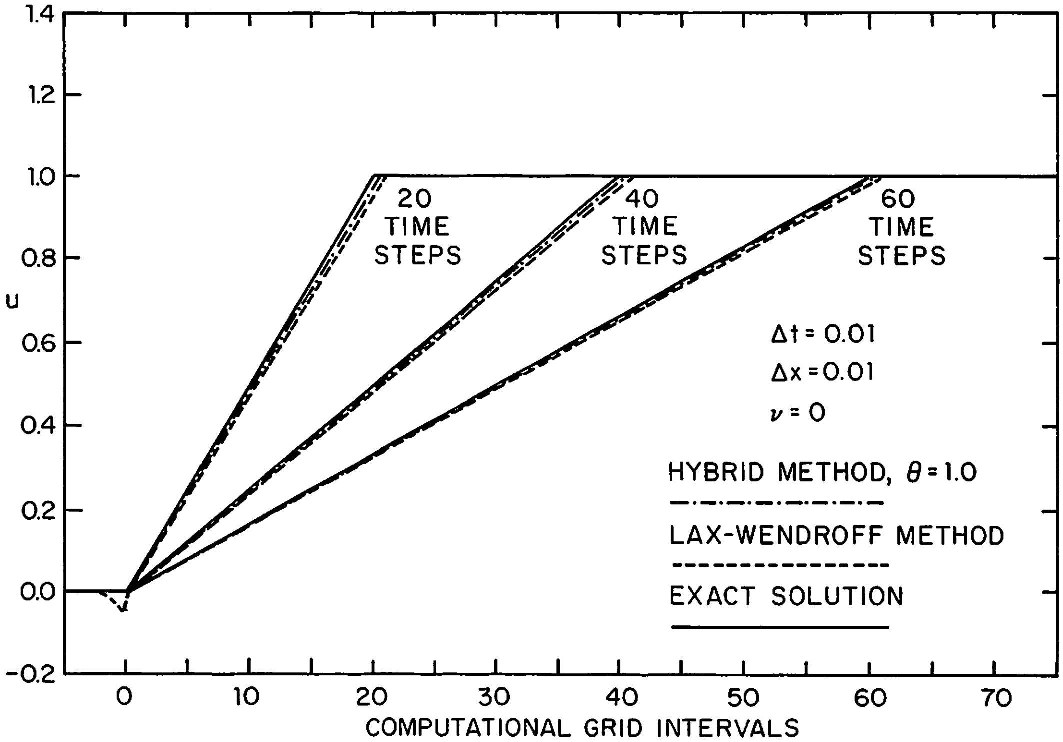

Fig. 4. Comparison of exact and numerical solution, nonlinear pure-advection (case N3)

propagation, it was apparent that the method offered a slight improvement in accuracy over a pure Preissmann-scheme approximation, for an appropriate choice of parameters θ and ϕ. Full details are presented by Toda (10).

DISCUSSION

The hybrid method can simulate the corresponding exact solutions quite well as long as the two parameters θ and ϕ are suitably chosen. However, two concerns have been identified for generalized practical use of this method. First, it has been found difficult to obtain a general rule for choice of the two parameters, as no exact analysis can be performed for the nonlinear situation and the optimum choice differs for each of the three test cases. A second difficulty is in selecting a proper root from the trajectory equation. The value of α is obtained by solving the cubic Eq. 14, with Δt set sufficiently small to ensure $\alpha < 1$. In the above three cases and the de St. Venant applications, examination of roots of Eq. 14 in each computational interval at each time step revealed that when two real roots exist between 0 and 1, one of which is always 0 or very close to 0, adoption of the finite root (i.e., the larger of the two) results in the physically correct trajectory evaluation; selection of the near-zero root disallows any movement of the shock front. Therefore, at least for the above three test cases, no difficulties occurred in root selection. However, even though there is no guarantee that only one real, non-trivial root always exists, no physical guidance has been identified for choosing the proper one of multiple roots with no *a priori* knowledge of the correct solution.

CONCLUSION

A hybrid method comprising the Holly-Preissmann and Crank-Nicholson schemes can succeed in reproducing exact solutions of Burger's equation quite well with suitable θ and ϕ values. However, the method's potential to improve the accuracy of inertia-dominant flow simulation over existing methods is questionable due to the two concerns cited, namely the suitable choice of the two parameters, and the selection of the proper root of the cubic trajectory equation. These concerns should be resolved before extension of the method for use in practical problems.

REFERENCES

1. Abramowitz, M. and I.A. Stegun (Ed.): Handbook of Mathematical Functions, Dover Publications, Inc., New York, NY., 1964.
2. Cunge, J.A., F.M. Holly, Jr., and A. Verwey: Practical Aspects of Computational River Hydraulics, Iowa Institute of Hydraulic Research, Iowa City, Iowa, 1980.
3. Holly Jr., F.M. and A. Preissmann: Accurate calculation of transport in two-dimensions, J. Hydraul. Div., ASCE, Vol. 98, HY11, 1259-1277, 1977.
4. Hughes, T.J.R. (Ed.): Finite Element Method for Convection Dominated Flow, AMD-34, ASME, New York, NY, 1979.
5. Lax, P.D.: Weak solutions of non-linear hyperbolic equations and their numerical applications, Comm. on Pure and Appl. Math., Vol. 7, 159-193, 1954.
6. Lohar, B.L. and P.C. Jain: Variable mesh cubic spline technique for N-wave solution of Burger's equation, J. Comp. Phys., Vol. 39, 433-442, 1981.
7. Lohner,R., K. Morgan, and O.C. Zienkiewicz: The Solution of nonlinear hyperbolic equation system by the finite element method, Int. Num. Meth. Fluids, Vol. 4, 1043-1063, 1984.
8. Richtmyer, R.D. and K.W. Morton: Difference Methods for Initial Value Problems, 2nd edn. Interscience Publishers, Wiley, New York, NY, 1967.
9. Roache, P.J.: Computational Fluid Dynamics, Hermosa Publishers, Albuquerque, NM, 1972.
10. Toda,K.: Numerical modelling of advection phenomena, Ph.D. thesis, Dept. of Civil and Environmental Engineering, University of Iowa, 1986.

APPENDIX-NOTATION

The following symbols are used in this paper:

a_o - a_3	=	coefficients of cubic interpolating ploynomial for u;
c_o - c_3	=	coefficients of cubic trajectory equation;
i	=	computational point index (distance);
n	=	computational point index (time);
t	=	time;
Δt	=	computational time interval;
u	=	dependent variable of Burger's equation, or mean velocity;
u_x	=	spatial derivative of u;
x	=	longitudinal space coordinate;
Δx	=	distance between two computational points in x-direction;
α	=	$(x_\eta - x_\xi)/\Delta x$, or root of cubic interpolating polynomial;
η	=	head of characteristic trajectory;
θ	=	weighting coefficient in two-point integration of diffusion term in Burger's equation;
ν	=	diffusion coefficient;
ξ	=	foot of characteristic trajectory;
ϕ	=	weighting coefficient in two-point integration of trajectory.

APPENDIX-ROOT SELECTION FOR CUBIC TRAJECTORY EQUATION

The solution of the cubic trajectory Eq. 14 is obtained following the analytical procedures described by Abramowitz and Stegan (1). Equation 14 can be rewritten as

$$c_o + c_1\alpha + c_2\alpha^2 + c_3\alpha^3 = 0 \qquad (21)$$

in which $c_o = a_o + \frac{\phi\nu\Delta t}{(\Delta x)^2}(u^n_{i+1} - 2u^n_i + u^n_{i-1})$, $c_1 = a_1 - \frac{\Delta x}{\Delta t}$, $c_2 = a_2$, and $c_3 = \alpha_3$.

Roots of Eq. 21 are obtained as follows:

If $c_3 \neq 0$ is assumed, let

$$q = \frac{1}{3}\left(\frac{c_1}{c_3}\right) - \frac{1}{9}\left(\frac{c_2}{c_3}\right)^2, \quad r = \frac{1}{6}\left\{\left(\frac{c_1}{c_3}\right)\left(\frac{c_2}{c_3}\right) - 3\left(\frac{c_o}{c_3}\right)\right\} - \frac{1}{27}\left(\frac{c_2}{c_3}\right)^3 \qquad (22)$$

If $q^3 + r^2 > 0$, there exist one real root and a pair of complex conjugate roots; if $q^3 + r^2 = 0$, all roots are real and at least two are equal; if $q^3 + r^2 < 0$, all roots are real (irreducible case).

Let $s_1 = [r+(q^3 + r^2)^{1/2}]^{1/3}$ and $s_2 = [r-(q^3 + r^2)^{1/2}]^{1/3}$, then the roots are:

$$\alpha_1 = (s_1 + s_2) - \frac{1}{3}\left(\frac{c_2}{c_3}\right) \qquad (23)$$

$$\alpha_2 = -\frac{1}{2}(s_1 + s_2) - \frac{1}{3}\left(\frac{c_2}{c_3}\right) + \frac{j\sqrt{3}}{2}(s_1 - s_2) \qquad (24)$$

$$\alpha_3 = -\frac{1}{2}(s_1 + s_2) - \frac{1}{3}\left(\frac{c_2}{c_3}\right) - \frac{j\sqrt{3}}{2}(s_1 - s_2) \qquad (25)$$

If $q^3 + r^2 > 0$, the unique real root is α_1, and if $q^3 + r^2 \leq 0$, all three roots (α_1, α_2, and α_3) are real.

During testing of the hybrid method, it was found convenient to reduce the order of the trajectory polynomial whenever the coefficients c_3 and c_2 become very small. Then the following three cases can be considered:

* If $c_2 = c_3 \approx 0$ (linear equation), or $c_3 \neq 0$ (cubic equation) and $q^3 + r^2 > 0$, there exists a unique real root.

* If $c_3 \approx 0$ and $c_2 \neq 0$ (quadratic equation) and $c_1^2 - 4c_oc_2 > 0$, there are two real roots.

* If $c_3 \neq 0$ (cubic equation) and $q^3 + r^2 < 0$, there exist three real roots.

When multiple real roots exist, the root which falls between 0 and 1 (inclusive) is chosen as the appropriate one. If there is more than one real root in that region, the largest one is chosen. The flowchart of Figure 5 shows the algorithm for acquiring the appropriate root as described above.

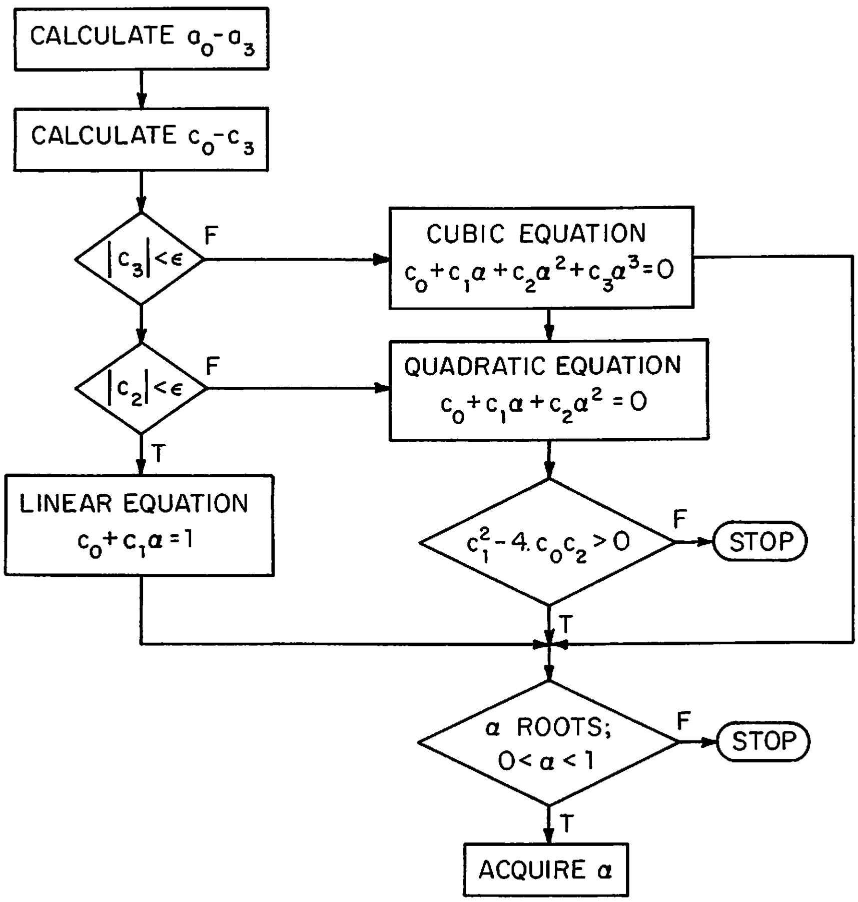

Fig. 5. Flowchart for root selection.