ON NEW AND DIRECT COMPUTATIONAL APPROACHES TO SOME MATHEMATICAL MODELS OF TURBULENCE*

By

R. BELLMAN AND S. P. AZEN (*The RAND Corporation, Santa Monica, Cal.*)
AND J. M. RICHARDSON (*Hughes Research Laboratories, Malibu, Cal.*)

1. Introduction. One of the most tantalizing areas of modern mathematical physics is the theory of turbulence. The complexity of the physical process, combined with the nonlinearity of the hydrodynamical equations, precludes any direct analytic approach and forces us to study various types of approximation techniques. A quite useful technique in many investigations is the analysis of "model equations" which, hopefully, exhibit many of the essential characteristics of the more realistic and correspondingly more recalcitrant equations.

Of these model equations in turbulence, perhaps the best known is the equation of Burgers,

$$u_t + uu_x = \epsilon u_{xx}, \quad u(x, 0) = g(x), \quad (1.1)$$

which, as shown by Hopf and others [1], can be reduced to the linear heat equation by a change of variable and thus can be solved explicitly. Since the analytic solution is available, this equation furnishes a very important test of any proposed analytic approximation method or of any computational procedure for the actual hydrodynamic equations. More general equations of the form

$$u_t + \phi(u)_x = \epsilon u_{xx} \quad (1.2)$$

have been studied by Lax and Kalaba [2].**

In this paper we wish to present the results of two computational methods which have been made feasible only in recent years with the development of the high speed digital computer. With the present greater speeds and larger rapid-access memories, we can expect that routine algorithms can be applied to the study of many important physical processes.

In the first part of this paper, we consider an application of the classical method of converting a partial differential equation into an infinite system of ordinary differential equations. This method was studied by Lichtenstein and Siddiqi (whose work is referred to in reference [3]; in [3] a different approach is used). The reader interested in some of the rigorous aspects may wish to consult references [4] and [5], where justification, under various hypotheses, is given of the "method of sections" ("principe des reduites") that we shall employ.

Fifty years ago, the possibility of using this technique to obtain *numerical* solutions of partial differential equations was slight, since in order to obtain sufficient accuracy one has to think in terms of 50, 100, or 1000 equations. As we shall see below, the number of equations required for accuracy depends upon the value of ϵ in (1.1). With modern computers, the prospects are quite different.

There are several advantages in reducing a numerical process to the solution of

*Received July 6, 1964.

**In the cited paper, references to Lax and others may be found.

ordinary differential equations with initial conditions. First, current computers are ideally designed for this type of repetitive process. Second, we now have available a large stock of computational algorithms of guaranteed accuracy for the solution of equations of this type. This makes the calculation routine, since we have transformed the original problem into that of solving a system of the form

$$\frac{dx_i}{dt} = g_i(x_1, x_2, \dots, x_N), \quad x_i(0) = c_i, \quad i = 1, 2, \dots, N. \quad (1.3)$$

There is, however, much to be gained from effecting this transformation efficiently. This brings us to the subject of "closure," one of the basic problems of mathematical physics. In previous papers [6-8], we have discussed and applied some new techniques. In this paper we further indicate how these techniques may be applied, and show how extrapolation and nonlinear summability methods [9] are used in this case.

We also discuss the use of some new types of difference approximations which have already proved themselves in other connections [10], [11].

Numerical results are given throughout, together with the time required for computing each case.

2. Burgers' equation. Let us consider (1.1) over the region $-\infty < x < \infty$, $t > 0$, and suppose, for the sake of simplicity, that $g(x)$ is periodic, with period 2π . We wish then to find a periodic solution of (1.1).

Write

$$u(x, t) = \sum_{-\infty < k < \infty} u_k(t) e^{ikx} = \sum_{-\infty < k < \infty} u_k(t) (\cos kx + i \sin kx) \quad (2.1)$$

and $u_k(t) = v_k(t) + iw_k(t)$, where v_k and w_k are real. The condition that $u(x, t)$ be real requires that

$$v_k = v_{-k}, \quad w_k = -w_{-k}. \quad (2.2)$$

Substituting the expression in (2.1) into (1.1) and equating coefficients, we obtain the infinite system of ordinary differential equations

$$u'_n(t) + \sum_{-\infty < k < \infty} ik u_k(t) u_{n-k}(t) = -n^2 \epsilon u_n(t). \quad (2.3)$$

Here $\epsilon > 0$. Substituting $u_n = v_n + iw_n$ and equating real and imaginary parts, we obtain the real system

$$\begin{aligned} v'_n - \sum_{-\infty < k < \infty} k(w_k v_{n-k} + v_k w_{n-k}) &= -n^2 \epsilon v_n, \\ w'_n + \sum_{-\infty < k < \infty} k(v_k v_{n-k} - w_k w_{n-k}) &= -n^2 \epsilon w_n. \end{aligned} \quad (2.4)$$

Since this is an infinite system of equations, we must use some closure technique in order to obtain a finite system. At this point we employ the simple device of setting

$$v_k, w_k = 0, \quad |k| \geq N + 1 \quad (2.5)$$

for some value of N . Taking various values of N and ϵ , we explore the degree of accuracy obtained as a function of these variables, by carrying out a computational solution of the resulting finite system of ordinary differential equations. Subsequently, we consider some more sophisticated types of closure along the lines discussed in references [6-8].

The finite system of differential equations obtained from (2.4), using the foregoing closure technique, is

$$\begin{aligned}
 v'_n - \sum_{k=0}^n k(w_k v_{n-k} + z_k w_{n-k}) - \sum_{k=n+1}^N k(w_k v_{k-n} - v_k w_{k-n}) \\
 + \sum_{k=1}^{N-n} k(-w_k v_{n+k} + v_k w_{n+k}) = -n^2 \epsilon v_n, \\
 w'_n + \sum_{k=0}^n k(v_k v_{n-k} - w_k w_{n-k}) + \sum_{k=n+1}^N k(v_k v_{k-n} + w_k w_{k-n}) \\
 - \sum_{k=1}^{N-n} k(v_k v_{n+k} + w_k w_{n+k}) = -n^2 \epsilon w_n,
 \end{aligned} \quad (2.6)$$

for $n = 0, 1, 2, \dots, N$. This is a system of $4N$ simultaneous equations with initial values.

These initial values are obtained from the Fourier expansion

$$g(x) \sim \sum_{-\infty < k < \infty} g_k e^{ikx}. \quad (2.7)$$

That is if $g_k = h_k + ir_k$, we have $v_k(0) = h_k$, $w_k(0) = r_k$.

3. Some numerical examples. A FORTRAN program was written for the IBM 7090 in order to study the effect of varying N and ϵ in (2.6). As mentioned above, the minimum value of N required to give accurate values of, for example, $u_1(t)$, should depend on ϵ . The integration was carried out using either the fixed step Runge-Kutta or the variable step Adams-Moulton technique.

Three cases were considered

Case I:

$$\begin{aligned}
 \epsilon &= 0.01, & (a) \\
 u_1(0) &= 1, u_k(0) = 0, k \neq 1, & (b) \\
 N &= 5, 10, 25, 36, 50. & (c)
 \end{aligned} \quad (3.1)$$

FIG. 1. $v_1(t)$ as a function of t for $N = 5, 10, 25, 36, 50$, for $\epsilon = 0.01$.

Figure 1 shows $v_1(t)$ as a function of t for different values of N . A cutoff at $N = 36$ yields results substantially as good as those for $N = 50$, in half the computation time.

Case II:

$$\begin{aligned} \epsilon &= 0.1, & (a) \\ u_1(0) &= 1, u_k(0) = 0, k \neq 1, & (b) \\ N &= 5, 10, 25. & (c) \end{aligned} \tag{3.2}$$

This example shows the effect of increasing the value of ϵ , since significantly accurate results are obtained using a cutoff of $N = 6$. $N = 10$ and 25 give the same curve (see Fig. 2).

Case III:

$$\begin{aligned} \epsilon &= 0.01, & (a) \\ u_1(0) &= i, u_k(0) = 0 \text{ for } k \neq 1, & (b) \\ N &= 5, 10, 25, 36, 50. & (c) \end{aligned} \tag{3.3}$$

FIG. 2. $v_1(t)$ as a function of t for $N = 10, 25$, for $\epsilon = 0.1$.

Computed values for w_i can be compared in the following table:

TABLE I.

w_1	$N = 5$	$N = 10$	$N = 25$	$N = 36$	$N = 50$
$t = 1.1$	0.634	0.640	0.618	0.612	0.612
$t = 2.2$	0.654	0.568	0.398	0.375	0.370
$t = 3.3$	0.718	0.513	0.306	0.269	0.264
$t = 4.4$	0.614	0.407	0.255	0.210	0.205
$t = 4.95$	0.591	0.367	0.236	0.189	0.184
computer time	1/6 min	1 min	2 min	5 min	12 min

The calculation for $N = 36$ is accurate to ± 0.005 compared with $N = 50$.

Figures 3, 4, and 5 show the dependence of $w_k(t)$ on $t = 1, 2, 3, 4, 5$, as k varies, where $N = 36$.

FIG. 3. The dependence of $w_k(t)$ on k for $t = 1, 2$, with $N = 36$, $\epsilon = 0.01$.

FIG. 4. The dependence of $w_k(t)$ on k for $t = 3, 4$, with $N = 36$, $\epsilon = 0.01$.

4. Two-dimensional case. For the two-dimensional case, we wish to solve the system of partial differential equations,

$$\begin{aligned} u_t + uu_x + vu_y &= \epsilon(u_{xx} + u_{yy}), \\ v_t + uv_x + vv_y &= \epsilon(v_{xx} + v_{yy}), \end{aligned} \tag{4.1}$$

FIG. 5. The dependence of $w_k(t)$ on k for $t = 5$, with $N = 36$, $\epsilon = 0.01$.

obvious analogs of the equation of Burgers. Unfortunately, these equations cannot be solved in closed analytic form for $\epsilon > 0$. For $\epsilon = 0$, the equations can be solved. Once again, we represent the solutions as a Fourier series

$$\begin{aligned} u(x, y, t) &= \sum_k \sum_l u_{kl}(t) e^{i(kx + ly)}, \\ v(x, y, t) &= \sum_k \sum_l v_{kl}(t) e^{i(kx + ly)}, \end{aligned} \quad (4.2)$$

where $u_{kl} = a_{kl} + ib_{kl}$, $v_{kl} = c_{kl} + id_{kl}$.

Substituting in (4.1), we obtain the following infinite systems of differential equations:

$$u'_{m,n}(t) + \sum_k \sum_l iku_{k,l}u_{m-k,n-l} + \sum_k \sum_l ilu_{k,l}v_{m-k,n-l} = -\epsilon(m^2 + n^2)u_{m,n}, \quad (4.3)$$

$$v'_{m,n}(t) + \sum_k \sum_l ikv_{k,l}u_{m-k,n-l} + \sum_k \sum_l ilv_{k,l}v_{m-k,n-l} = -\epsilon(m^2 + n^2)v_{m,n}, \quad (4.4)$$

where $-\infty \leq k, l \leq +\infty$.

Substituting $u_{kl} = a_{kl} + ib_{kl}$ and $v_{kl} = c_{kl} + id_{kl}$ in (4.4), and separating into real and imaginary parts, we obtain an infinite set of real ordinary differential equations. If we assume that $u_{mn} = v_{mn} = 0$ for $|m|, |n| \geq N + 1$ for some integer $N > 0$, we again have a finite system:

$$\begin{aligned} a'_{m,n} - \sum_k \sum_l k(a_{\alpha,\beta}b_{k,l} + a_{k,l}b_{\alpha,\beta}) \\ - \sum_k \sum_l l(b_{k,l}c_{\alpha,\beta} + a_{k,l}d_{\alpha,\beta}) = -\epsilon a_{m,n}(m^2 + n^2), \end{aligned}$$

$$\begin{aligned}
b'_{m,n} + \sum_k \sum_l k(a_{k,l}a_{\alpha,\beta} - b_{k,l}b_{\alpha,\beta}) \\
+ \sum_k \sum_l l(a_{k,l}c_{\alpha,\beta} - b_{k,l}d_{\alpha,\beta}) = -\epsilon b_{m,n}(m^2 + n^2), \\
c'_{m,n} - \sum_k \sum_l k(d_{k,l}a_{\alpha,\beta} + c_{k,l}b_{\alpha,\beta}) \\
- \sum_k \sum_l l(d_{k,l}c_{\alpha,\beta} + c_{k,l}d_{\alpha,\beta}) = -\epsilon c_{m,n}(m^2 + n^2), \\
d'_{m,n} + \sum_k \sum_l k(c_{k,l}a_{\alpha,\beta} - d_{k,l}b_{\alpha,\beta}) \\
+ \sum_k \sum_l l(c_{k,l}c_{\alpha,\beta} - d_{k,l}d_{\alpha,\beta}) = -\epsilon d_{m,n}(m^2 + n^2),
\end{aligned} \tag{4.5}$$

where $\alpha = m - k$, $\beta = n - l$, $-N \leq k$, and $l \leq N$.

We can now solve this system for positive and negative indices, in which there are $(2N + 1)^2$ coefficients, or, we can reduce the number of coefficients by utilizing the symmetry relations:

$$\begin{aligned}
a_{-m,-n} &= a_{m,n}, & b_{-m,-n} &= -b_{m,n}, \\
a_{-m,n} &= a_{m,-n}, & b_{-m,n} &= -b_{m,-n}, \\
c_{-m,-n} &= c_{m,n}, & d_{-m,-n} &= -d_{m,n}, \\
c_{-m,n} &= c_{m,-n}, & d_{-m,n} &= -d_{m,-n}.
\end{aligned}$$

If we follow the latter procedure, each coefficient is written in terms of 12 double summands. For programming purposes, the first method is preferable.

We have carried out some preliminary calculations testing the feasibility of the method. Unfortunately, the times involved make it difficult to carry out the type of experimentation discussed in the foregoing sections in connection with the one-dimensional equation. The situation has greatly improved in the past year; new IBM computers will cut down the required time by at least a factor of 10.

5. Closure techniques. The general question of closure may be phrased in the terms given below. Starting with a system of differential equations,

$$\frac{dx}{dt} = g(x) \tag{5.1}$$

in vector notation, we want to replace it by another system

$$\frac{dy}{dt} = h(y), \tag{5.2}$$

with simpler analytic, conceptual, and computational properties, so that $\|x - y\|$ is small. Here $\|\cdots\|$ is some suitably defined norm; for analytic convenience it is usually the Euclidean norm. In some cases, we are content to have $h(y)$ linear, i.e., $h(y) = Ay + b$ but in other cases we are more interested in having the dimension of y considerably less than the dimension of x .

In earlier papers we have considered the case where x was finite dimensional [6-8]. Here we wish to consider an infinite-dimensional case where (5.1) corresponds to the system of (2.4). The problem of closure is now more difficult since there is no immediate

way of expressing the components corresponding to higher harmonics in terms of the components for smaller k .

We discuss two approaches in Sections 6 and 7 below.

6. A Direct method. Consider the equation

$$u_t + uu_x = \epsilon u_{xx} , \quad (6.1)$$

where we set

$$u(x, t) = \sum_{-\infty < k < \infty} u_k(t) e^{ikx} . \quad (6.2)$$

To obtain a finite set of differential equations, set

$$u = \sum_{|k| \leq N} u_k(t) e^{ikx} , \quad (6.3)$$

substitute in (6.1) and equate coefficients.

7. Extrapolation. We now present another technique. Suppose that we have already employed Carleman linearization [8] to obtain an infinite linear system of the form

$$\frac{du_k}{dt} = \sum_{l=1}^{\infty} a_{kl} u_l , \quad u_k(0) = c_k , \quad k = 1, 2, \dots , \quad (7.1)$$

and suppose further that we wish to obtain an approximating linear system of the form

$$\frac{du_k}{dt} = \sum_{l=1}^N \alpha_{kl} u_l , \quad u_k(0) = c_k . \quad (7.2)$$

Instead of merely choosing $\alpha_{kl} = a_{kl}$, we wish to approximate to the remainder terms by means of linear combinations of the initial terms

$$\sum_{l \geq N+1} a_{kl} u_l \simeq \sum_{l=1}^N b_{kl} u_l \quad (7.3)$$

(see [6-8]).

The coefficients b_{kl} are to be chosen so that

$$\int_0^T \left[\sum_{l \geq N+1} a_{kl} u_l - \sum_{l=1}^N b_{kl} u_l \right]^2 dt \quad (7.4)$$

is a minimum. In this way we obtain linear algebraic equations for the b_{kl} :

$$\sum_{l \geq N+1} a_{kl} \int_0^T u_l u_r dt = \sum_{l=1}^N b_{rl} \int_0^T u_l u_r dt , \quad r = 1, 2, \dots , N. \quad (7.5)$$

The usual difficulty now confronts us. How do we compute the integrals $\int_0^T u_l u_r dt$ involving the unknown solution?

Consider first the case where $1 \leq r, l \leq N$. To obtain these integrals, we use the finite system in (7.2), with $\alpha_{kl} = a_{kl}$. Call the solutions $u_r^{(0)}$. The coefficients of b_{rl} in (7.5) are then $\int_0^T u_l^{(0)} u_r^{(0)} dt$. Observe that these quantities can be computed directly in the course of obtaining the $u_k^{(0)}$ by adjoining to (7.2) the equations

$$\frac{dw_{rl}}{dt} = u_r u_l , \quad w_{rl}^{(0)} = 0 , \quad (7.6)$$

and seeking only the values $w_{rl}(T)$.

The more difficult problem is that of the calculation of $\int_0^T u_r u_l dt$ for $r = 1, 2, \dots, w$, $l \geq N + 1$. Here we use extrapolation techniques. Keep r fixed and let l vary over the integers, $l = 1, 2, \dots$. It is reasonable to expect that $f_{l,r} = \int_0^T u_r^{(0)} u_l^{(0)} dt$ will be a well-behaved sequence. Consequently, if we possess the values $f_{1,r}, f_{2,r}, \dots, f_{N,r}$, we can use any of a number of extrapolation techniques [9] to obtain the values of $f_{l,r}$ for $l \geq N + 1$.

The system in (7.5) then has the form

$$\sum_{M \geq l \geq N+1} a_{kl} f_{l,r} = \sum_{l=1}^N b_{rl} w_{rl}(T). \quad (7.7)$$

Here M is a cutoff number, such as $2N$, which depends upon the size of the coefficients a_{kl} and the rapidity of convergence of the infinite series.

Solving (7.7) numerically, we obtain the coefficients $b_{rl}^{(0)}$. We use the superscript to indicate the fact that these are the first approximations. To obtain higher approximations, we use self-consistency techniques.

In place of (7.2), with $\alpha_{kl} = a_{kl}$ let us now use the system

$$\frac{du_k}{dt} = \sum_{l=1}^N a_{kl} u_l + \sum_{l=1}^N b_{kl}^{(0)} u_l, \quad u_k^{(0)} = c_k. \quad (7.8)$$

Call the solutions of this equation $u_k^{(1)}$, $k = 1, 2, \dots, N$. We now proceed as before to calculate $w_{rl}^{(1)}(T)$, $f_{l,r}^{(1)}$ and $b_{kl}^{(1)}$. With the new coefficients $b_{kl}^{(1)}$, we introduce the equation

$$\frac{du_k}{dt} = \sum_{l=1}^N a_{kl} u_l + \sum_{l=1}^N b_{kl}^{(1)} u_l, \quad u_k^{(0)} = c_k. \quad (7.9)$$

This process is repeated for a fixed number of steps or until the values of the b_{kl} settle down.

8. Difference approximations. An entirely different approach to the numerical solution of partial differential equations is based upon the use of difference equations. In [10], we introduced a new type of difference equation technique and presented some further applications in [11]. Here we wish to apply them to the Burgers' equation.

9. An approximating algorithm. Consider the equation (1.1) over the region $0 \leq x \leq 1$, $t > 0$, and suppose that $g(x)$ is periodic, with period π . Let the approximating algorithm be given by

$$u(x, t + \Delta) = \lambda u(x - au(x, t)\Delta, t) + \frac{(1 - \lambda)}{2} [u(x + b\Delta^{1/2}, t) + u(x - b\Delta^{1/2}, t)], \quad (9.1)$$

where Δ is the integration step size, and λ , a , b are constants which will be determined. To show that (9.1) approximates (1.1) to an error of $O(\Delta^2)$, expand both sides of (9.1) in a Taylor series up to the Δ^2 term, obtaining the equation

$$u_t = -\lambda a u u_x + (1 - \lambda) \frac{b^2}{2} u_{xx}. \quad (9.2)$$

For (9.2) to approximate (1.1), the following relations must hold:

$$a = 1/\lambda, \quad b = \left(\frac{2\epsilon}{1 - \lambda} \right)^{1/2}. \quad (9.3)$$

If ϵ is fixed, then a and b are functions of the parameter λ , and (9.1) becomes

$$u(x, t + \Delta) = \frac{1}{\lambda} u \left[x - u(x, t) \frac{\Delta}{\lambda}, t \right] + \frac{1 - \lambda}{2} \left[u \left(x + \left(\frac{2\epsilon\Delta}{1 - \lambda} \right)^{1/2}, t \right) + u \left(x - \left(\frac{2\epsilon\Delta}{1 - \lambda} \right)^{1/2}, t \right) \right]. \quad (9.4)$$

Let $t = 0, \Delta, 2\Delta, \dots$, and at each stage of the calculation let $u(x, t)$ be stored by means of the finite sum

$$u(x, t) \cong \sum_{n=1}^M u_n(t) \sin n\pi x, \quad (9.5)$$

where the coefficients $u_n(t)$ are obtained by the quadrature scheme

$$u_n(t) = 2 \int_0^1 u(x, t) \sin n\pi x \, dx \cong \frac{2}{R} \sum_{k=1}^{R-1} u(k/R, t) \sin (n\pi k/R). \quad (9.6)$$

Hence, the values $u(k/R, t)$, $k = 1, 2, \dots, R - 1$ store $u(x, t)$ at time t , and by way of (9.4), $u(x, t + \Delta)$ can be obtained.

10. Numerical results. To obtain some numerical results, a FORTRAN program was written for the IBM 7090. The following results were obtained:

$$\text{A.} \quad u_t + uu_x = \epsilon u_{xx}, \quad (10.1)$$

where

$$u(x, 0) = -\sin \pi x, \quad 0 \leq x \leq 1 \\ \epsilon = 0.01, \quad \lambda = 0.5, \quad \Delta = 0.05.$$

In this first example the parameters M and R were varied. As can be seen in the following table, one obtains essentially the same results for $M = R = 10$, compared with $M = R = 15$, in less than half the time.

TABLE II.

x	t	$u(x, t)$	
		$M = R = 10$	$M = R = 15$
0.5	1.00	-0.355	-0.354
0.1	2.00	-0.319	-0.326
0.6	3.00	-0.116	-0.116
0.9	4.00	-0.022	-0.022
0.2	5.00	-0.132	-0.133
Time		40 sec	1 min

Note that in this example and in the following examples the largest differences occur for small values of x .

$$\text{B.} \quad u_t + uu_x = \epsilon u_{xx},$$

where

$$u(x, 0) = -\sin \pi x, \quad 0 \leq x \leq 1 \\ \epsilon = 0.01, \quad \Delta = 0.05, \quad M = R = 10.$$

In the second example the parameter λ was varied. As shown in Fig. 6, varying λ displays large differences in the values of $u(x, t)$ for small x and for small t . Note, how-

FIG. 6. $u(x, t)$ as a function of t and λ , with $x = 0.1$.

FIG. 7. $u(x, t)$ as a function of t and λ , with $x = 0.5$.

ever, that the maximum of each curve occurs at about the same time. For small t and large x , the variation in $u(x, t)$ is much smaller (see Fig. 7); for large values of t , the variation is quite small (see Table III).

TABLE III.

x	t	$\lambda = 2$	$\lambda = 8$	<i>difference</i>
0.1	3	-0.184	-0.223	0.039
0.6	3	-0.115	-0.115	0.000
0.1	4	-0.131	-0.157	0.026
0.6	4	-0.089	-0.089	0.000
0.1	5	-0.098	-0.115	0.017
0.6	5	-0.73	-0.073	0.000

C.

$$u_t + uu_x = \epsilon u_{xx},$$

where

$$u(x, 0) = -\sin \pi x, \quad 0 \leq x \leq 1$$

$$\epsilon = 0.01, \quad \lambda = 0.5, \quad M = R = 10.$$

The parameter which has the greatest effect on the results is the integration step size Δ . The differences can be seen in Table IV.

TABLE IV.

x	t	$u(x, t)$	
		$\Delta = 0.05$	$\Delta = 0.01$
0.5	1.0	-0.355	-0.381
0.1	2.0	-0.319	-0.375
0.6	3.0	-0.116	-0.120
0.9	4.0	-0.022	-0.023
0.2	4.8	-0.139	-0.147
Time		40 sec	180 sec

11. Higher order approximation. The general form for the approximating algorithm is given by

$$u(x, t + \Delta) = \lambda u(x - au(x, t)\Delta, t) + \sum_{i=1}^N a_i [u(x + b_i \Delta^{1/2}, t) + u(x - b_i \Delta^{1/2}, t)] \quad (11.1)$$

where λ , R , a_i , b_i , and a are inputs. It was shown in [12] that higher ordered approximations, as in (11.1), can give more accurate results, provided the polynomial approximation (9.5) is sufficiently accurate.

REFERENCES

1. E. Hopf, *A mathematical example displaying features of turbulence*, Comm. Pure Appl. Math **1** (1948) 303-322

2. R. Kalaba, *On nonlinear differential equations, the maximum Operation and monotone convergence*, J. Math. Mech. **8** (1959) 519–574
3. R. Beliman, *On the existence and boundedness of solutions of nonlinear partial differential equations of parabolic type*, Trans. Amer. Math. Soc. **64** (1948) 21–44
4. R. Bellman, *The boundedness of solutions of infinite systems of linear differential equations*, Duke Math. J. **14** (1947) 695–706
5. F. Riesz, *Les systèmes d'équations linéaires à une infinité d'inconnues*, Gauthier–Villars, Paris, 1913
6. R. Bellman and J. M. Richardson, *Perturbation techniques*, Symposium on Nonlinear Oscillations, Kiev, USSR, 1962
7. R. Bellman and J. M. Richardson, *Renormalization techniques and mean-square averaging—I: Deterministic equations*, Proc. Nat. Acad. Sci. **47** (1961) 1191–1194
8. R. Bellman and J. M. Richardson, *On the asymptotic behavior of solutions of nonlinear differential equations*, J. Math. Anal. Appl. **4** (1962) 470–474
9. R. Bellman and R. Kalaba, *A note on nonlinear summability techniques in invariant imbedding*, J. Math. Anal. Appl. **6** (1963) 465–472
10. R. Bellman, R. Kalaba, and B. Kotkin, *On a new approach to the computational solution of partial differential equations*, Proc. Nat. Acad. Sci. **48** (1962) 1325–1327
11. R. Bellman, I. Cherry, and C. M. Wing, *A note on the numerical integration of a class of nonlinear hyperbolic equations*, Q. Appl. Math. **16** (1958) 181–183
12. S. P. Azen, *Higher Order Approximations to the computational solution partial differential equations*, The RAND Corporation RM-3917-PR, 1963