JOURNAL OF COMPUTATIONAL PHYSICS 89, 63–79 (1990)

Benchmarking the Propagator Method for Nonlinear Systems: A Burgers–Korteweg–deVries Equation

D. G. CACUCI

Department of Chemical and Nuclear Engineering, University of California, Santa Barbara, California 93106

AND

O. A. KARAKASHIAN

Department of Mathematics, University of Tennessee, Knoxville, Tennessee 37996-1300

Received April 28, 1988; revised July 28, 1989

A new method for solving a generalized Burgers–Korteweg–deVries (BKdV) equation is presented. This new method consists of two symbiotic aspects—analytical and numerical, respectively. The analytical aspect consists of obtaining the propagators—which are the nonlinear analogues of Green's functions—for the BKdV equation, and subsequently using these propagators to obtain the BKdV solution as a fixed-point integral equation. The numerical aspect consists of developing a highly efficient and accurate method for solving this integral equation. Benchmarking comparisons using the Korteweg–deVries and Burgers equations indicate that our new method is efficient, accurate, and robust.

I. INTRODUCTION

Recently, Cacuci, Perez, and Protopopescu [1] have proposed a new formalism for solving scalar nonlinear problems; Cacuci and Protopopescu [2] have subsequently extended it to multicomponent (i.e., matrix) nonlinear systems. This formalism represents a generalization to nonlinear systems of the classical Green's function method in linear theory. In this formalism, the analogues of the Green's functions are forward and backward propagators that are canonically constructed from the original nonlinear system. These propagators are then used to obtain a closed-form expression for the solution of the original system; this procedure effectively recasts the original system—usually a nonlinear initial/boundary value problem—into an equivalent nonlinear integral equation which is, in principle, easier to solve.

In this work, we apply this general canonical formalism [1, 2] to solve a non-

0021-9991/90 $3.00

581/89/1-5

linear equation that generalizes both the Burgers and the Korteweg–deVries (KdV) equations. The analytical aspect of our new method is presented in Section II and consists of deriving the respective propagators and obtaining the closed-form expression for the solution of this generalized Burgers–Korteweg–deVries (BKdV) equation in terms of these propagators. The numerical aspect of our method is presented in Section III and consists of developing an accurate and efficient procedure for solving numerically the closed-form expression obtained in Section II. Section III also presents several numerical experiments, including comparisons for the KdV equation between our new method and the method, based on the inverse scattering transform, presented by Taha and Ablowitz [3], the method of Zabuski and Kruskal [4] as presented in Ref. [3], and a finite element method (FEM) described in Refs. [5–7]. Finally, our conclusions are summarized in Section IV.

II. A Generalized Burgers–Korteweg–deVries Equation

The generalized Burgers–Korteweg–deVries (BKdV) equation introduced in this work is

$$N(u) \equiv u_t + \eta u_x + u^p u_x - \nu u_{xx} + \varepsilon u_{xxx} = 0, \tag{1}$$

where $u(x, t)$ is a real-valued function of the two real variables x and t, the quantities η, ν, and ε are real constants, and the subscripts denote partial differentiations. Equation (1) generalizes both the Burgers equation and the Korteweg–deVries (KdV) equation: the former is obtained from Eq. (1) by setting $\eta = 0$, $p = 1$, $\nu = 1$, and $\varepsilon = 0$, while the latter is obtained from Eq. (1) by setting $\eta = 0$, $p = 1$, $\nu = 0$, and $\varepsilon = 1$.

Throughout this work, we consider that $x \in [0, 1]$ and $t \in [0, T]$, where T is some final-time value; the nonlinear operator $N(u)$ defined by Eq. (1) is the mapping $N(u)$: $D(N) \subset H \to R$, where $D(N)$ is the domain $D(N) = C^3([0, 1] \times [0, T])$, and H is the Hilbert space $H = L^2((0, 1) \times (0, T))$ endowed with the inner product

$$\langle u, v \rangle \equiv \int_0^1 dx \int_0^T u(x, t)\, v(x, t)\, dt, \qquad u \in H, \quad v \in H. \tag{2}$$

The domain $D(N)$ is further specified by considering that $u(x, t)$ satisfies the periodic boundary conditions

$$\{\partial^j u/\partial x^j\}_{x=0} = \{\partial^j u/\partial x^j\}_{x=1}, \qquad j = 0, 1, 2, \tag{3}$$

and the initial-time condition

$$u(x, 0) = u_{\text{in}}(x), \tag{4}$$

where $u_{\text{in}}(x)$ is a known function of x.

III. Solution of the BKdV Equation by the Method of Nonlinear Propagators

III.A. *Nonlinear Propagators for the BKdV Equation*

The concept of forward and backward propagators for nonlinear problems has been introduced by Cacuci *et al.* [1]; they showed that these propagators play a role analogous to that played by the customary Green's functions for linear problems. Applying the procedure presented in Ref. [1] to Eq. (1), we obtain propagators for the BKdV equation as follows:

The Gâteaux (G-)derivative $N'(u_0)$ of $N(u)$ at a point $u_0 \in D(N)$ exists and is explicitly obtained by applying its definition

$$N'(u_0)\, h \equiv \lim_{\gamma \to 0} [N(u_0 + \gamma h) - N(u_0)]/\gamma, \qquad h \in D(N),$$

to Eq. (1); this gives

$$N'(u_0)\, h = (\partial/\partial t + \eta\, \partial/\partial x + p u_0^{p-1}\, \partial u_0/\partial x + u_0^p\, \partial/\partial x - \nu\, \partial^2/\partial x^2 + \varepsilon\, \partial^3/\partial x^3)\, h. \tag{5}$$

Note that $N'(u_0)$ acts linearly on h, but depends nonlinearly on the parameter u_0.

The formal adjoint $[N'(u)]^*$ of $N'(u)$ is obtained from Eq. (5) as

$$[N'(u)]^*\, v = [-\partial/\partial t - \eta\, \partial/\partial x - u^p\, \partial/\partial x - \nu\, \partial^2/\partial x^2 - \varepsilon\, \partial^3\, \partial^3 x]\, v, \qquad v \in H. \tag{6}$$

Next, we define the operators

$$L^*(u)\, v \equiv \int_0^1 [N'(\gamma u)]^*\, v\, d\gamma \tag{7}$$

and

$$L(u)\, h \equiv \int_0^1 N'(\gamma u)\, h\, d\gamma. \tag{8}$$

Using Eqs. (6) and (5) in Eqs. (7) and (8), respectively, yields

$$L^*(u)\, v = [-\partial/\partial t - \eta\, \partial/\partial x - (p+1)^{-1}\, u^p\, \partial/\partial x - \nu\, \partial^2/\partial x^2 - \varepsilon\, \partial^3/\partial x^3]\, v \tag{9}$$

and

$$L(u)\, h = [\partial/\partial t + \eta\, \partial/\partial x + (p+1)^{-1}\, \partial(u^p \cdot)/\partial x - \nu\, \partial^2/\partial x^2 + \varepsilon\, \partial^3/\partial x^3]\, h. \tag{10}$$

Note the important property,

$$L(u)\, u = N(u), \tag{11}$$

and the important relationship

$$\langle L(u)\, h, v\rangle = \langle h, L^*(u)\, v\rangle + \{\Sigma(u)(h, v)\}, \qquad u \in H, \quad v \in H, \tag{12}$$

where the boundary form $\{\Sigma(u)(h, v)\}$, defined as

$$\begin{aligned}\{\Sigma(u)(h, v)\} \equiv & \int_0^1 [h(x, T)\, v(x, T) - h(x, 0)\, v(x, 0)]\, dx \\ & + \int_0^T [\eta h v + (p+1)^{-1} u^p v h - v(v h_x - h v_x) \\ & + \varepsilon(v h_{xx} - v_x h_x + h v_{xx})]_{x=0}^{x=1}\, dt,\end{aligned} \tag{13}$$

is bilinear in h and v while depending nonlinearly on the parameter u. In particular, Eq. (12) holds for $h = u$ and $v = G_u^*(x, t; x_1, t_1)$, where the function $G_u^*(x, t; x_1, t_1)$, henceforth called the *forward* or *advanced propagator*, is chosen to be the solution of the linear system

$$L^*(u)\, G_u^*(x, t; x_1, t_1) = \delta(x - x_1)\, \delta(t - t_1), \tag{14}$$

subject to the time condition

$$G_u^*(x, t; x_1, t_1) = 0 \qquad \text{for} \quad t > t_1, \tag{15}$$

and the (periodic) boundary conditions

$$\{\partial^j G_u^*(x, t; x_1, t_1)/\partial x^j\}_{x=0} = \{\partial^j G_u^*(x, t; x_1, t_1)/\partial x^j\}_{x=1} \qquad \text{for} \quad j = 0, 1, 2. \tag{16}$$

In Eq. (14), $\delta(x)$ is the customary Dirac-delta distribution.

Writing Eq. (12) for $h = u$ and $v \equiv G_u^*$, and using Eqs. (1), (3), (4), (11), (15), and (16) leads to the following expression for the solution $u(x, t)$ of the BKdV equation:

$$u(x, t) = \int_0^1 u_{\text{in}}(y)\, G_u^*(y, 0; x, t)\, dy. \tag{17}$$

The relationship expressed by Eq. (17) underscores the fact that the forward propagator, G_u^*, "propagates" the initial distribution $u_{\text{in}}(x)$ from the phase-space location $(y, 0)$ to the phase-space location (x, t). As noted by Cacuci *et al.* [1], the role played by G_u^* is entirely similar to that played by the customary Green's function encountered in *linear* problems, such as in field theory and quantum mechanics. There are two reasons for choosing the time/boundary conditions for G_u^* as given by Eqs. (15) and (16):

(a) these conditions form, together with Eq. (14), a well-posed problem admitting, for each $u(x, t)$, a unique solution G_u^*, and

(b) replacing these conditions together with the initial/boundary conditions [i.e., Eqs. (3) and (4)] for the BKdV problem in Eq. (13) leads to the cancellation of all but one of the terms in the boundary form $\{\Sigma(u)(u, G_u^*)\}$.

Using the propagator G_u^*, we have succeeded in recasting the original BKdV equation—a nonlinear partial differential equation subject to initial/boundary conditions—into the implicit nonlinear integral equation (17). The implicitness of Eq. (17) stems from the fact that G_u^* depends implicitly and nonlinearly on the solution $u(x, t)$.

To recast the implicit integral equation for $u(x, t)$ given by Eq. (17) into a form that can be solved efficiently and accurately, we introduce the auxiliary forward propagator $G_0^*(x, t; x_1, t_1)$, defined as the solution of the initial/boundary value problem obtained by replacing the *unknown* function $u(x, t)$ in Eqs. (14) through (16) with a *known* function $u_0 \in D(N)$, namely,

$$\begin{aligned} L^*(u_0)\, G_0^*(x, t; x_1, t_1) \equiv & -\partial G_0^*/\partial t - [u_0^p(p+1)^{-1} + \eta]\, \partial G_0^*/\partial x \\ & - \nu \partial^2 G_0^*/\partial x^2 - \varepsilon \partial^3 G_0^*/\partial x^3 \\ = & \delta(x - x_1)\, \delta(t - t_1), \end{aligned} \tag{18}$$

$$G_0^*(x, t; x_1, t_1) = 0 \qquad \text{for} \quad t > t_1, \tag{19}$$

$$\{\partial^j G_0^*/\partial x^j\}_{x=0} = \{\partial^j G_0^*/\partial x^j\}_{x=1} \qquad \text{for} \quad j = 0, 1, 2. \tag{20}$$

At this stage, the implicit nonlinear equation (17) for the solution $u(x, t)$ can be recast into an explicit nonlinear integral equation involving the auxiliary propagator $G_0^*(x, t; x_1, t_1)$ via the following sequence of operations,

$$\begin{aligned} u(x, t) &= \langle u(x_1, t_1), \delta(x_1 - x)\, \delta(t_1 - t) \rangle \\ &= \langle u(x_1, t_1), L^*(u_0(x_1, t_1))\, G_0^*(x_1, t_1; x, t) \rangle \\ &= \langle u(x_1, t_1), [L^*(u_0) - L^*(u)]\, G_0^*(x_1, t_1; x, t) \rangle \\ &\quad + \langle u(x_1, t_1), L^*(u(x_1, t_1))\, G_0^*(x_1, t_1; x, t) \rangle \\ &= \langle u(x_1, t_1), [L^*(u_0) - L^*(u)]\, G_0^*(x_1, t_1; x, t) \rangle \\ &\quad + \langle L(u)\, u(x_1, t_1)), G_0^*(x_1, t_1; x, t) \rangle - \{\Sigma(u)(u, G_0^*)\}, \end{aligned} \tag{21}$$

where the last equality was obtained by using Eq. (12), with $h \to u(x_1, t_1)$ and $v \to G_0^*(x_1, t_1; x, t)$, to replace the term $\langle u, L^*(u)\, G_0^* \rangle$. Next, we (i) note that $\langle L(u)\, u, G_0^* \rangle = \langle N(u), G_0^* \rangle = 0$ in view of Eqs. (11) and (1), (ii) replace the operators $L^*(u_0)$ and $L(u)$ by their respective expressions and perform the subtraction $L^*(u_0) - L^*(u)$ indicated in Eq. (21), and (iii) use the initial and boundary conditions given in Eqs. (3) and (4) for $u(x, t)$, and Eqs. (19) and (20) for G_0^*,

respectively, to simplify the expression of $\{\Sigma(u)(u, G_0^*)\}$ in Eq. (21). Consequently, Eq. (21) reduces to

$$u(x, t) = \int_0^1 u_{\text{in}}(y)\, G_0^*(y, 0; x, t)\, dy - (p+1)^{-1} \int_0^t dt_1 \int_0^1 dx_1 \times \{[u(x_1, t_1)\, u_0^p(x_1, t_1) - u^{p+1}(x_1, t_1)]\, \partial G_0^*(x_1, t_1; x, t)/\partial x_1\}. \quad (22)$$

Equation (22) gives the closed-form expression of the solution $u(x, t)$ of the BKdV equation; it is a nonlinear integral equation whose solution requires the knowledge of the function $G_0^*(x_1, t_1; x, t)$. Solving Eq. (22) would be most easily accomplished if the auxiliary propagator $G_0^*(x_1, t_1; x, t)$ could be obtained by solving Eqs. (18)–(20) exactly, analytically. Since the choice of u_0 is entirely at our disposal, we note that Eqs. (18)–(20) could be solved readily if u_0 were a constant, independent of x and t.

Selecting $u_0 = \text{const}$ and noting that consequently

$$\eta + u_0^p (p+1)^{-1} = \text{const}, \quad (23)$$

the exact expression of $G_0^*(x_1, t_1; x, t)$ can be readily obtained by solving Eqs. (18)–(20) via a Fourier transform in space and a Laplace transform in time. The final result is

$$G_0^*(x_1, t_1; x, t) = H(t - t_1) \left\{ 1 + 2 \sum_{n=1}^{\infty} \exp[-4\pi^2 n^2 \nu (t - t_1)] \times \cos 2\pi n \left[\left(\eta + \frac{u_0^p}{p+1} - 4\pi^2 n^2 \varepsilon \right) (t - t_1) - (x - x_1) \right] \right\}, \quad (24)$$

where

$$H(t) = \begin{cases} 0 & \text{if } t < 0 \\ 1 & \text{if } t \geqslant 0 \end{cases}$$

is the customary Heaviside (unit-step) function.

Note that for the particular value $u_0 \equiv 0$, the operator L^* $(u_0 \equiv 0)$ becomes the formal adjoint of the linearized BKdV operator, i.e., the adjoint of $(\partial/\partial t + \eta \partial/\partial x - \nu \partial^2/\partial x^2 + \varepsilon \partial^3/\partial x^3)$. Consequently, for $u_0 \equiv 0$, the auxiliary propagator $G_0^*(x, t; x_1, t_1)$ becomes the Green's function for the linearized BKdV equation. We also note that for $u_0 \equiv 0$, Eq. (22) reduces to a form that includes, in Fourier space, both the Fornberg and Whitham [8] formulation for the Korteweg–deVries equation and the Canuto *et al.* [9] "integrating-factor" technique for the Burgers equation.

III.B. *Numerical Procedure*

The fixed-point form solution to the BKdV equation given by Eq. (22) lends itself remarkably well to a variety of numerical discretizations: finite difference, finite

element, or spectral decomposition for the spatial variable, and a host of temporal schemes. Note also the important fact that the numerically troublesome u_{xxx} term which appears in the original PDE, cf. Eq. (1), no longer appears in the integral solution (22).

To solve Eq. (22) numerically, we have implemented a spectral method (i.e., a truncated Fourier series) for the spatial variable with a class of implicit Runge–Kutta methods for the temporal integral. This method of solving Eq. (22) will be described in the following.

Setting $p = 1$ and using the representation

$$u(x, t) = \sum_{n=0}^{\infty} \{a_n(t) \cos 2\pi n x + b_n(t) \sin 2\pi n x\}, \tag{25}$$

in Eq. (22), and performing the straightforward integrations and algebra, we obtain the expressions for the coefficients appearing in Eq. (25),

$$a_0(t) = a_0(0), \qquad b_0(t) = 0, \qquad t \geqslant 0, \tag{26}$$

$$\begin{aligned} a_n(t) = {} & a_n(0) \cos 2\pi n \tau_n t - b_n(0) \sin 2\pi n \tau_n t \\ & - \frac{\pi}{2} n \int_0^t \sin 2\pi n \tau_n (t - t_1) \left\{ \left(\sum_{r+s=n} + \sum_{|r-s|=n} \right) a_r(t_1)\, a_s(t_1) \right. \\ & \left. + \left(\sum_{|r-s|=n} - \sum_{r+s=n} \right) b_r(t_1)\, b_s(t_1) \right\} dt_1 - \pi n \int_0^t \cos 2\pi n \tau_n (t - t_1) \\ & \times \left(\sum_{r+s=n} - \sum_{r-s=n} + \sum_{s-r=n} \right) a_r(t_1)\, b_s(t_1)\, dt_1, \qquad n \geqslant 1, \end{aligned} \tag{27}$$

and

$$\begin{aligned} b_n(t) = {} & a_n(0) \sin 2\pi n \tau_n t + b_n(0) \cos 2\pi n \tau_n t \\ & + \frac{\pi}{2} n \int_0^t \cos 2\pi n \tau_n (t - t_1) \left\{ \left(\sum_{r+s=n} + \sum_{|r-s|=n} \right) a_r(t_1)\, a_s(t_1) \right. \\ & \left. + \left(\sum_{|r-s|=n} - \sum_{r+s=n} \right) b_r(t_1)\, b_s(t_1) \right\} dt_1 - \pi n \int_0^t \sin 2\pi n \tau_n (t - t_1) \\ & \times \left(\sum_{r+s=n} - \sum_{r-s=n} + \sum_{s-r=n} \right) a_r(t_1)\, b_s(t_1)\, dt_1, \qquad n \geqslant 1, \end{aligned} \tag{28}$$

where $\tau_n \equiv \eta + u_0^p (p+1)^{-1} - 4\pi^2 n^2 \varepsilon$, and the summation indices are nonnegative. In actual computations, we carry only a finite number, n_f, of Fourier modes.

The initial Fourier coefficients $\{a_n(0), b_n 0\}$ for $n = 0, 1, \ldots, n_f$, are obtained by a discrete least-squares technique using the formulae

$$a_n(0) = \frac{1}{M} \sum_{j=0}^{2M-1} u_{\text{in}} \left(\frac{j}{2M} \right) \cos \frac{\pi n j}{M}, \qquad 1 \leqslant n \leqslant n_f \leqslant M, \tag{29}$$

with

$$a_0(0) = \frac{1}{2M} \sum_{j=0}^{2M-1} u_{\text{in}}\left(\frac{j}{2M}\right)$$

and

$$b_n(0) = \frac{1}{M} \sum_{j=0}^{2M-1} u_{\text{in}}\left(\frac{j}{2M}\right) \sin\frac{\pi n j}{M}, \qquad 0 \leqslant n \leqslant n_f \leqslant M. \tag{30}$$

In the actual computations, we used the value $M = 100$.

To generate the temporal discretizations, we first note that Eqs. (27) and (28) constitute a system of nonlinear integral equations of the form

$$y(t) = g(t) + \int_0^t f(t, t_1, y(t_1))\, dt_1, \tag{31}$$

where, for each n, the quantity

$$g(t) \equiv a_n(0) \cos 2\pi n \tau_n t \pm b_n(0) \sin 2\pi n \tau_n t,$$

is known, while $f(\cdot, \cdot, \cdot)$ contains the integral terms appearing on the right sides of Eqs. (27) and (28).

To solve Eq. (31), we implemented a time-stepping procedure as follows: we select a time-stepsize $k > 0$ and a time-discretization procedure (to be described below) for the selected k; this discretization procedure is then applied repeatedly, until the desired final time T is attained in J steps, for some integer J satisfying $kJ = T$. This approach is feasible because both f in Eq. (31) and G_0^* depend explicitly only on the difference $t - t_1$; furthermore, this approach is both faster and more economical (requiring less storage) than a global method in time.

As the basic time discretization procedure for each k, we use a class of implicit Runge–Kutta (IRK) methods corresponding to numerical quadrature schemes of Gauss–Legendre type (see Ref. [10]). In addition to excellent stability and accuracy properties, this family of methods conserves numerically the first two invariants, i.e., $\int_0^1 u\, dx$ and $\int_0^1 u^2\, dx$, of the BKdV equation.

A q-stage IRK method is characterized by a set of constants given by the tableau

$$\begin{array}{c|c} a_{11} \cdots a_{1q} & t^{(1)} \\ \vdots \quad\quad \vdots & \vdots \\ a_{q1} \cdots a_{qq} & t^{(q)} \\ \hline b_1 \cdots b_q & \end{array}$$

For example, for $q = 1$, this tableau reads

$$\begin{array}{c|c} \frac{1}{2} & \frac{1}{2} \\ \hline & \\ 1 & \end{array}$$

while for $q = 2$, this tableau reads

$$\begin{array}{cc|c} \frac{1}{4} & \frac{1}{4}+\frac{1}{2\sqrt{3}} & \frac{1}{2}+\frac{1}{2\sqrt{3}} \\ \frac{1}{4}-\frac{1}{2\sqrt{3}} & \frac{1}{4} & \frac{1}{2}-\frac{1}{2\sqrt{3}} \\ \hline \frac{1}{2} & \frac{1}{2} & \end{array}$$

Thus, given y^0 as an approximation to $y(0)$, an approximation y^1 to $y(k)$ defined by Eq. (31) is obtained as

$$y(k) = g(k) + k \sum_{j=1}^{q} b_j f(k, kt^{(j)}, y^{(j)}), \tag{32}$$

where the unknown quantities $y^{(1)}, \ldots, y^{(q)}$ are given by the set of equations

$$y^{(i)} = g(t^{(i)}) + k \sum_{j=1}^{q} a_{ij} f(kt^{(i)}, kt^{(j)}, y^{(j)}), \qquad i = 1, \ldots, q. \tag{33}$$

Most of the computational effort is needed for finding or (when efficiency dictates otherwise) approximating the quantities $y^{(1)}, \ldots, y^{(q)}$. For sufficiently regular functions g and f, the procedure described by Eqs. (32) and (33) produces approximations to the solution y of Eq. (31) with temporal errors that decrease at the rate k^{2q}.

For the IRK corresponding to $q = 1$, the time discretization equations are

$$\begin{aligned} a_n^1 &= a_n(0)\cos\tau_n k - b_n(0)\sin\tau_n k \\ &\quad - \frac{k}{2}\pi n \sin\tau_n \frac{k}{2}\left\{\left(\sum_{r+s=n} + \sum_{|r-s|=n}\right) a_r^{(1)} a_s^{(1)} + \left(\sum_{|r-s|=n} - \sum_{r+s=n}\right) b_r^{(1)} b_s^{(1)}\right\} \\ &\quad - k\pi n \cos\tau_n \frac{k}{2}\left(\sum_{r+s=n} - \sum_{r-s=n} + \sum_{s-r=n}\right) a_r^{(1)} b_s^{(1)}, \end{aligned} \tag{34}$$

$$\begin{aligned} b_n^1 &= a_n(0)\sin\tau_n k + b_n(0)\cos\tau_n k \\ &\quad + \frac{k}{2}\pi n \cos\tau_n \frac{k}{2}\left\{\left(\sum_{r+s=n} + \sum_{|r-s|=n}\right) a_r^{(1)} a_s^{(1)} + \left(\sum_{|r-s|=n} - \sum_{r+s=n}\right) b_r^{(1)} b_s^{(1)}\right\} \\ &\quad - k\pi n \sin\tau_n \frac{k}{2}\left(\sum_{r+s=n} - \sum_{r-s=n} + \sum_{s-r=n}\right) a_r^{(1)} b_s^{(1)}, \end{aligned} \tag{35}$$

where, for $n \geqslant 1$, the quantities $a_n^{(1)}$ and $b_n^{(1)}$ are defined as

$$a_n^{(1)} = a_n(0) \cos \tau_n \frac{k}{2} - b_n(0) \sin \tau_n \frac{k}{2} - \frac{k}{2} \pi n \left(\sum_{r+s=n} - \sum_{r-s=n} + \sum_{s-r=n} \right) a_r^{(1)} b_s^{(1)}, \tag{36}$$

and

$$b_n^{(1)} = a_n(0) \cos \tau_n \frac{k}{2} + b_n(0) \cos \tau_n \frac{k}{2} + \frac{k}{4} \pi n \left\{ \left(\sum_{r+s=n} + \sum_{|r-s|=n} \right) a_r^{(1)} a_s^{(1)} + \left(\sum_{|r-s|=n} - \sum_{r+s=n} \right) b_r^{(1)} b_s^{(1)} \right\}. \tag{37}$$

Note that for $n = 0$, we require that $a_0^{(1)} = a_0(0)$ and $b_0^{(1)} = 0$.

For higher order ($q \geqslant 2$) IRK methods, the equations resulting from Eqs. (32) and (33) are similar to, but more complicated than, Eqs. (34) through (37).

The coupled nonlinear system described by Eqs. (36) and (37) was solved by a simple fixed-point-type iteration of the form $Z_{l+1} = F(Z_l)$, $l = 0, 1, ..., l_{\max}$. For each time step, jk, $j = 0, 1, ..., J$, starting points that help minimize the number of fixed-point iterations on l were obtained by using a linear extrapolation of values at the two previous time steps; thus, Z (for $l = 0$, jk) $= \frac{3}{2}\xi_{jk} - \frac{1}{2}\xi_{(j-1)k}$, where the vectors ξ have as components the quantities $a_n(0)$ and $b_n(0)$ as they appear on the right side of Eqs. (36) and (37). For the fourth-order ($q = 2$) IRK method, we used a cubic (involving values at four previous time steps) rather than a linear extrapolation. With these starting points, none of the actual computations required more than one fixed point iteration on l.

IV. Numerical Experiments

To obtain a comparative evaluation of the efficiency of our new (i.e., propagator-spectral-IRK) method, henceforth denoted as CK-S, we used as benchmarks two schemes described in [3], namely a finite difference scheme due to Zabuski and Kruskal [4], henceforth denoted as ZK, and the local-method of Taha and Ablowitz [3], henceforth denoted as TA-L, based on the inverse scattering transform. We chose the TA-L method for comparison to ours because Taha and Ablowitz [3] indicate that the TA-L method was the most efficient of a host of other numerical techniques (reported prior to their work) for solving the KdV equation. On the other hand, we chose the ZK method for comparison purposes not only because it was used by Zabuski and Kruskal [4] in their pioneering work on solitons but also because we repeated their "recurrence time" experiment using our method. In addition, we compared our CK-S method to a finite element (Galerkin) procedure, henceforth denoted by FEM, described in Refs. [5–7]. To our knowledge, this is the only additional technique for solving the KdV equation published since the Taha-Ablowitz work [3]. In this FEM procedure, the spatial

discretization is achieved by using smooth splines on a uniform grid, while the temporal discretizations are obtained by using IRK schemes similar to those described earlier in this work.

We performed the numerical comparisons between the ZK, TA-L, FEM, and CK-S methods in a scalar mode using FORTRAN on the IBM 3090 at the University of Tennessee, Knoxville. The exact benchmark used in these comparisons was the solitary wave solution

$$u(x, t) = A \operatorname{sech}^2\left[\sqrt{\frac{A}{12\varepsilon}}\left(x - \frac{1}{2} - \frac{t}{3}\right)\right] \tag{38}$$

of the KdV equation $u_t + uu_x + \varepsilon u_{xxx} = 0$. Also, to conform to the experiments described in [3], we have taken $\varepsilon = 1.042 \times 10^{-4}$, $T = 0.15$, and values of the amplitude $A = 1$, 2, and 4.

Results of our numerical experiments for the solitary wave solution are summarized in Table I. For the ZK and TA-L methods, we used as the measure of accuracy the quantity $error = \max\{|u(x_i, T) - U_i(T)|\}$, where $U_i(T)$ denotes the value of the approximate solution at the ith spatial node x_i. For the CK-S and FEM methods, we used as the measure of accuracy the quantity

$$error = \max_{0 \leqslant i \leqslant N} \{|u(x_i, T) - U_i(T)|\},$$

where $x_i = i/N$, $i = 0, ..., N$, with $N = 500$ and 1024, respectively. The levels of accuracy used in Table I are chosen to conform to those reported by Taha and Ablowitz [3]. For each method, we experimented to optimize the number of time and spatial intervals in order to achieve the reported level of accuracy using the least amount of computational work. Also, the CPU times reported in Table I concern only the numerical integration of the KdV equation; specifically, we did not lump these CPU times together with the CPU times used to compute the respective errors. For this reason, the ratios of the CPU times between the ZK and TA-L techniques shown in Table I differ from those reported in [3].

Several remarks can be made based on the numerical experiments summarized in Table I:

1. A single iteration of the fixed-point form $Z_{l+1} = F(Z_l)$ of Eqs. (36) and (37) was usually sufficient to preserve the stability of the time-stepping procedure in the CK-S method. We believe this is due to the fact that the term u_{xxx} is completely eliminated in the CK-S method [cf. Eq. (22)]. The increased accuracy given by additional fixed-point iterations was not cost effective; in almost all cases, fixed-point convergence was obtained in less than five iterations, with essentially no change between the fourth and fifth iteration.

2. The two-stage (fourth-order) IRK scheme generally outperformed the one-stage (second-order) IRK scheme. In all experiments, the one-stage IRK scheme proved superior to the ZK method. Increasing the number of temporal/spatial steps

TABLE I

Results from Selected Methods for Solving the KdV Equation

Method	Error ($\times 10^3$)	No. of time steps	No. of spatial subintervals (n_f = Fourier modes)	CPU (seconds)
$A = 1$				
ZK	4.69	500	230	0.134
ZK	1.79	4000	375	1.67
TA-L	3.20	7	250	0.078
TA-L	1.73	8	250	0.084
FEM	4.40	7	128	0.043
FEM	1.61	10	128	0.054
CK-S	4.34	19	$n_f = 22$	0.059
CK-S	1.55	22	$n_f = 25$	0.073
$A = 2$				
ZK	9.3	5264	500	2.97
ZK	3.31	25000	840	22.63
TA-L	9.70	10	370	0.277
TA-L	3.32	10	400	0.300
FEM	9.80	30	128	0.129
FEM	3.105	20	128	0.284
CK-S	9.17	58	$n_f = 32$	0.163
CK-S	2.95	90	$n_f = 36$	0.262
$A = 4$				
ZK	33.54	25000	840	23.36
ZK	94.85	5264	500	2.97
TA-L	17.47	37	800	1.87
TA-L	1.79	75	1524	4.74
FEM	17.65	235	192	1.03[a]
FEM	1.48	73	192	1.67[a]
CK-S	17.3	225	$n_f = 48$	0.9
CK-S	1.7	382	$n_f = 60$	2.18

[a]Different temporal schemes were used in these two runs.

in the ZK method beyond the 25000/840 combination rendered this method unstable.

3. The CK-S method is comparable to the FEM method; the ratio between the CPU times for the CK-S and FEM methods, respectively, varied between 0.87 and 1.37.

4. The CK-S method appears to be more efficient than the TA-L method; the ratio between the CPU times for the TA-L method and the CPU times for the CK-S method varied between 1.15 for $A = 1$ and 2.19 for $A = 4$, the CK-S method becoming more efficient with increasing soliton amplitude.

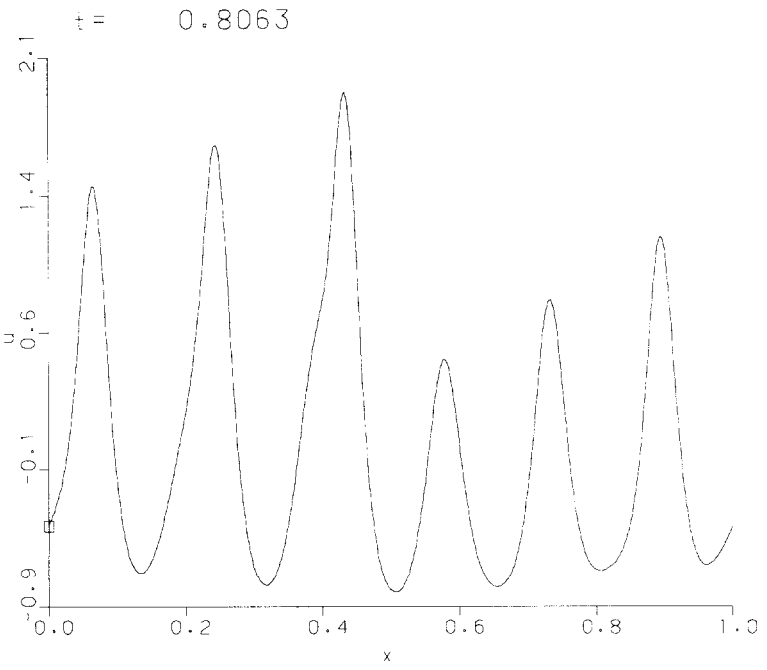

FIG. 1. KdV equation: $u_{in} = \cos 2\pi x$, $\varepsilon = 1.21 \times 10^{-4}$; solution at $t = 0.8063$.

In addition to the experiments summarized in Table I, we have used the CK-S method with initial condition $u_{in}(x) = \cos 2\pi x$, $\varepsilon = 1.21 \times 10^{-4}$, and $\nu = 0$ to repeat the "recurrence time" experiment of Zabuski and Kruskal [4]. Figures 1 through 4 show successive stages of the evolution of the solution $u(x, t)$ up to the "recurrence time," as defined in [4], of $t_R = 30.4 t_B = 4.838$, where $t_B = 1/2\pi$ is the (gradient) blowup time of the equation $v_t + vv_x = 0$. Note that our time scale differs from that in [4] by a factor of 2 due to the different spatial scales. For this experiment, we used $n_f = 35$ and 4200 time steps; the CPU time on the IBM 3090 was 17s. The error $|\int_0^1 u_{\text{exact}}^2\, dx - \int_0^1 u_{\text{approx}}^2\, dx|$ in the second invariant was 0.652×10^{-4}.

To test the robustness of the CK-S method, we decreased ε from 1.21×10^{-4} to 1.21×10^{-6} in the "recurrence time" experiment and calculated the solution $u(x, t)$ up to $t = t_B = 0.159$. With this very small value of ε, the solution produced (as expected) a sharp gradient at $t = t_B$ as seen in Fig. 5, together with a small flurry of Gibbs oscillations. For this experiment, we used 1200 time steps and $n_f = 75$; the CPU time on the IBM 3090 was 20s.

We have also integrated the Burgers equation with $\varepsilon = 0$ and $\nu = 10^{-3}$, using periodic boundary conditions and $u_{in}(x) = \sin 2\pi x$ as initial condition. For this experiment, we used 500 time steps and $n_f = 75$; the CPU time on the IBM 3090 was 8.3s. We have chosen this small value of ν to test once again the robustness of

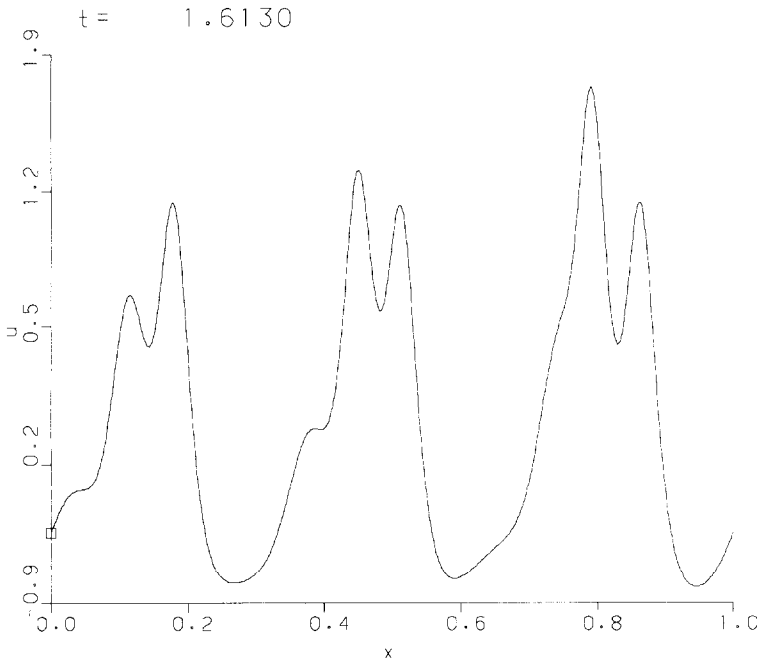

FIG. 2. KdV equation: $u_{in} = \cos 2\pi x$, $\varepsilon = 1.21 \times 10^{-4}$; solution at $t = 1.6130$.

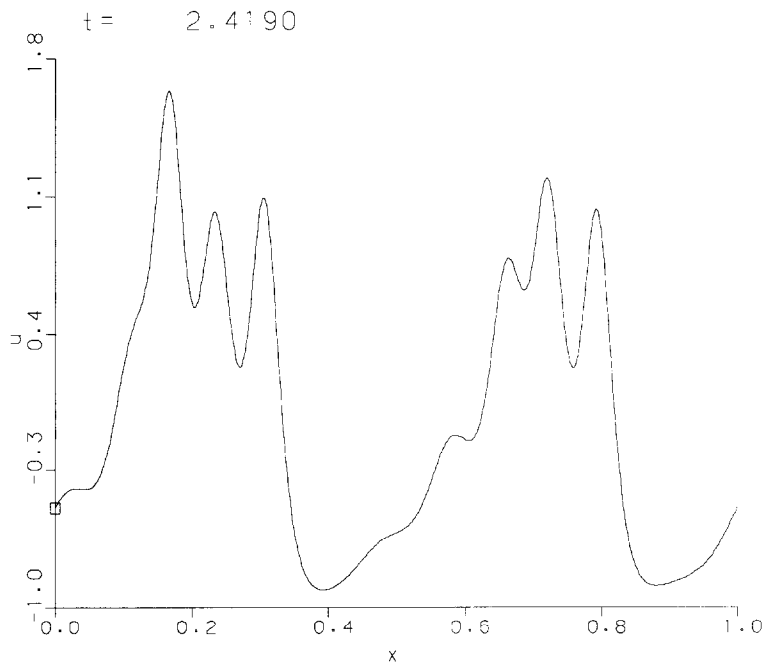

FIG. 3. KdV equation: $u_{in} = \cos 2\pi x$, $\varepsilon = 1.21 \times 10^{-4}$; solution at $t = 2.4190$.

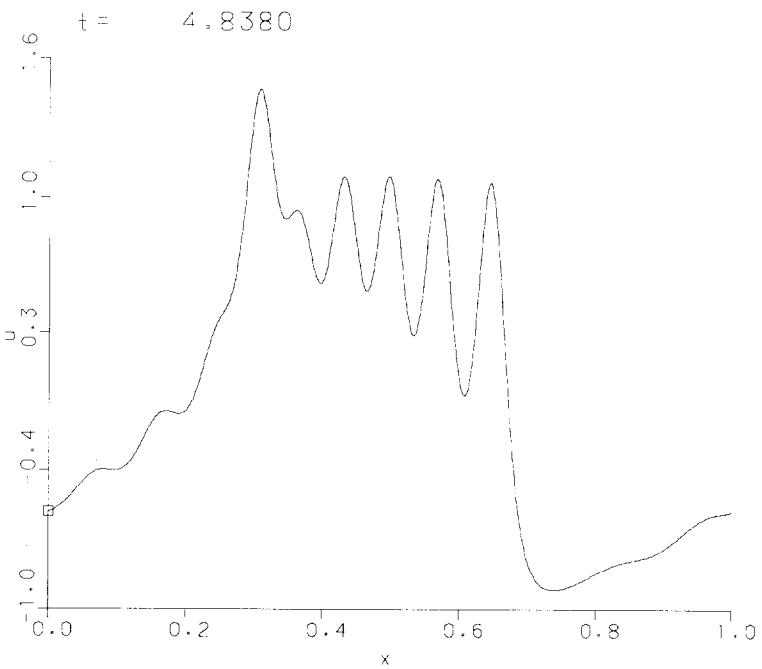

FIG. 4. KdV equation: $u_{in} = \cos 2\pi x$, $\varepsilon = 1.21 \times 10^{-4}$; solution at $t = 4.830$.

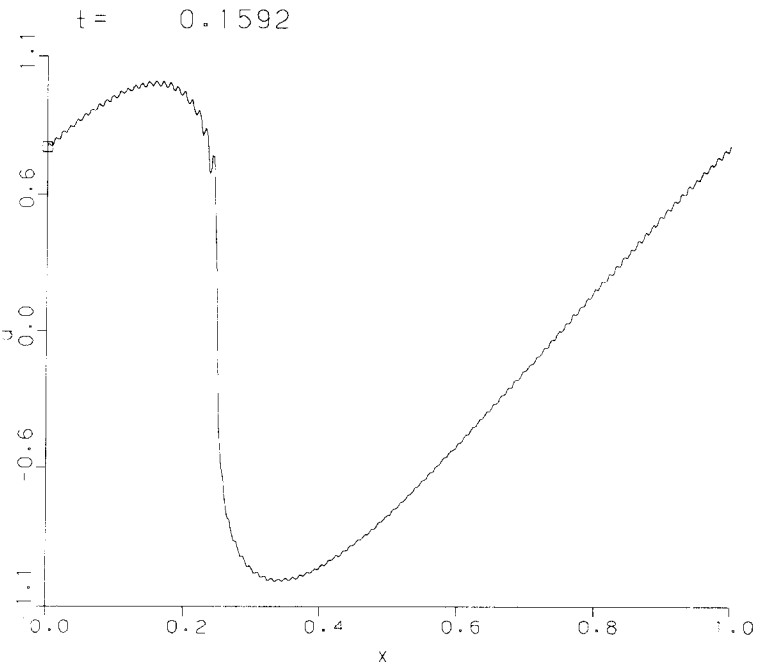

FIG. 5. KdV equation: $u_{in} = \cos 2\pi x$, $\varepsilon = 1.21 \times 10^{-6}$; solution at $t = 0.159$.

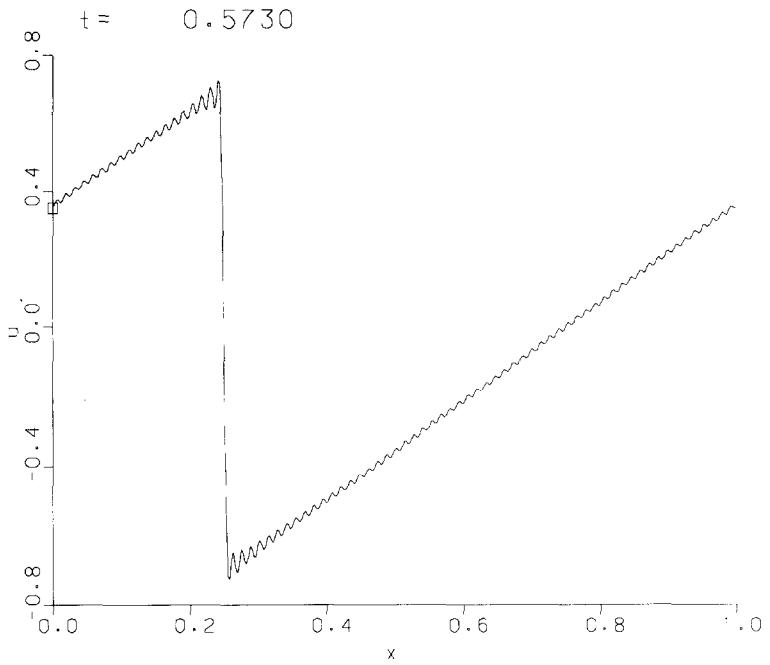

FIG. 6. Burgers equation: $u_{in} = \sin 2\pi x$, $\nu = 10^{-3}$; solution at $t = 0.573$.

our method and its ability to deal with sharp gradients. Figure 6 shows the results calculated by our CK-S method for the solution $u(x, t)$ of the Burgers equation at $t = 0.573$. As expected, $\nu = 10^{-3}$ is too small to produce any significant decay in the solution, which shows the expected sharp gradient accompanied by small, decaying Gibbs oscillations.

VI. Summary and Conclusions

In this work, we have benchmarked a new methodology for nonlinear systems by using the Burgers–Korteweg–deVries (BKdV) equations. This methodology involves the applications of the formalism developed by Cacuci *et al.* [1] to obtain the nonlinear propagators for the BKdV equation. This transforms the BKdV equation from an initial/boundary value problem into a closed-form nonlinear integral equation that is amenable to solution by various numerical methods. We have chosen to solve this integral equation by using spectral (Fourier) decomposition spatially and implicit Runge–Kutta temporally; the spectral decomposition is suggested naturally by the expression of the auxiliary propagator appearing as a kernel in the closed-form BKdV solution.

We have tested the efficiency and accuracy of our new method on a series of benchmarks for the Korteweg–deVries and the Burgers equations, largely following

the accuracy levels used by Taha and Ablowitz [3] when showing that their method (based on the inverse scattering transform) bested a host of other numerical methods. For these benchmarks, the comparisons we have presented in this work indicate that our new method is more efficient for comparable accuracy than the Taha–Ablowitz method and proved to be comparable to a finite element method (FEM), described in Refs. [4–6], which is specifically designed for solving the KdV equation with maximum efficiency. In addition to these accuracy comparisons, we performed several experiments with the KdV and Burgers equations, involving sharp gradients, to test more severely the robustness of our method. These experiments were handled quite well by our method—which produced accurate approximations in reasonable CPU times. These CPU times are expected to decrease considerably by employing fast Fourier transform techniques—as we are currently using in conjunction with application of nonlinear propagators to multidimensional nonlinear systems.

ACKNOWLEDGMENTS

This work was partially supported by the Science Alliance of UTK-ORNL and the U.S. Department of Energy under Contract DE-AC05-84OR21400 with Martin Marietta Energy Systems, Inc. The work of the second author was also supported by the Air Force Office of Scientific Research under Grant AFOSR-88-0019. Special thanks are due to Ms. Barbara Merlo for her expert typing of this manuscript. We also thank Ms. Camille S. Muir and Dr. T. Taha for providing the listing of the Taha–Ablowitz code used in Ref. [3]. Finally, we wish to thank the three anonymous reviewers for their helpful comments.

REFERENCES

1. D. G. CACUCI, R. B. PEREZ, AND V. PROTOPOPESCU, *J. Math. Phys.* **29**, 353 (1987).
2. D. G. CACUCI AND V. PROTOPOPESCU, *J. Phys. A: Math. Gen.* **22**, 2399 (1989)
3. T. R. TAHA AND M. J. ABLOWITZ, *J. Comput. Phys.* **55**, 231 (1984).
4. N. J. ZABUSKI AND M. D. KRUSKAL, *Phys. Rev. Lett.* **15**, 240 (1965).
5. G. A. BAKER, V. A. DOUGALIS, AND O. A. KARAKASHIAN, *Math. Comput.* **40**, 419 (1983).
6. V. A. DOUGALIS AND O. A. KARAKASHIAN, *Math. Comput.* **45**, 329 (1985).
7. J. L. BONA, V. A. DOUGALIS, AND O. A. KARAKASHIAN, Conservative high order schemes for the generalized Korteweg–deVries Equation, in preparation.
8. B. FORNBERG AND G. B. WHITHAM, *Philos. Trans. Roy. Soc.* **289**, 373 (1978).
9. C. CANUTO, M. Y. HUSSAINI, A. QUARTERONI, AND T. A. ZANG, *Spectral Methods in Fluid Dynamics* (Springer-Verlag, New York, 1987).
10. J. C. BUTCHER, *Math. Comput.* **18**, 50 (1964).