IMA Journal of Numerical Analysis (1986) **6**, 25–42

Conservative and Nonconservative Schemes for the Solution of the Nonlinear Schrödinger Equation

J. M. SANZ-SERNA

Dpto. Ecuaciones Funcionales, Facultad de Ciencias, Universidad de Valladolid, Valladolid, Spain

AND

J. G. VERWER

Centre for Mathematics and Computer Science, Amsterdam

[Received 20 June 1984 and in revised form 29 January 1985]

Five methods for the integration in time of a semidiscretization of the nonlinear Schrödinger equation are extensively tested. Three of them (a partly explicit scheme and two splitting procedures) are found to perform poorly. The reasons for their failure, including the so-called nonlinear blow-up, are analysed. We draw general conclusions on the advantages and drawbacks associated with the use of time-integrators which exactly conserve energy.

1. Introduction

THE NUMERICAL treatment of the initial-value problem for the nonlinear Schrödinger equation

$$iu_t + u_{xx} + q|u|^2 u = 0 \qquad (-\infty < x < \infty,\ t \geq 0) \tag{1.1}$$

$$u(x, 0) = g(x) \qquad (-\infty < x < \infty) \tag{1.2}$$

(u complex, $i^2 = -1$, q a positive constant) has received much attention recently: Delfour *et al.* (1981), Griffiths *et al.* (1984), Herbst & Mitchell (1983), Herbst *et al.* (1984), Sanz-Serna & Manoranjan (1983), Sanz-Serna (1984b), Verwer & Sanz-Serna (1984). This paper is devoted to a study of schemes for the integration *in time* of space-discretizations of (1.1). To this end (1.1) is first discretized in space by standard central differences and then five methods for the time-integration of the resulting system of ODEs are considered. The methods studied are the implicit midpoint rule, the pseudolinear midpoint rule (Verwer & Dekker, 1984), the partly explicit scheme of Griffiths *et al.* (1984) and two splitting (fractional step) procedures. The emphasis lies in the investigation of the potential advantages to be gained by the use of schemes, which *conserve energy exactly* (Morton, 1977). In this connection our experiments and those by Herbst *et al.*

(1984) and Sanz-Serna & Christie (1985) will enable us to make a definite assessment of the merits of exact conservation.

An overview of the paper is as follows. Section 2 is devoted to a survey of the analytic properties of (1.1)–(1.2) which are essential in the understanding of the article. (References concerning the nonlinear Schrödinger equation can be seen in any of the papers quoted above.) The numerical methods are described in Section 3. The methods are then tested in Section 4 by means of three increasingly difficult problems. The splitting methods turn out to perform badly and the reasons for this failure are analysed. The scheme suggested by Griffiths *et al.* (1984) may lead to nonlinear blow-up in agreement with these authors' findings and the mechanism of that undesirable phenomenon is investigated. The last section is devoted to conclusions. We have added an appendix on the energy growth in Runge–Kutta schemes.

2. The nonlinear Schrödinger equation

2.1 *Some Analytic Properties*

(a) *Dispersion and Nonlinearity* The *linear* Schrödinger equation

$$iu_t + u_{xx} = 0 \tag{2.1}$$

provides a model for the propagation of *dispersive* waves. In fact, (2.1) possesses Fourier solutions

$$u(x, t) = \exp[i(kx - W(k)t)], \qquad W(k) = k^2,$$

corresponding to the translation of the initial profile $\exp(ikx)$ with a speed $W(k)/k$ which obviously depends on the wave number k. Let us assume that the initial condition represents a disturbance confined to a small interval of the x-axis. Such an initial condition is a superposition of modes $\exp(ikx)$ and (each mode travelling at a different speed), the disturbance evolves spreading over the whole x-axis. It can be shown that, for the pure initial-value problem, the solutions of (2.1) have an amplitude which decays like $t^{-\frac{1}{2}}$ for $t, x \to \infty$ with x/t fixed (Whitham, 1974, p. 371).

The cubic term in (1.1) opposes dispersion, and thus it is possible for the *nonlinear* Schrödinger equation to possess solutions where the competing forces of dispersion and nonlinearity balance each other exactly. These 'balanced' solutions include the soliton, the interaction of solitons and the bound state of solitons, which will all be discussed later.

(b) *x-Independent Solutions* These satisfy the ODE

$$iu_t + q|u|^2 u = 0, \tag{2.2}$$

with general solution $b \exp(iq|b|^2 t)$ with b a complex constant. It is clear that in order that a numerical method for (1.1) be useful, it is necessary that it integrates accurately these simple x-independent solutions.

It is also interesting to point out that the linearization of (1.1) around an x-independent solution with $b \neq 0$ exhibits *growing* Fourier modes (*instability* with respect to long-wave perturbations) (Yuen & Ferguson, 1978, Herbst & Mitchell, 1983).

(c) *Conservation Laws* The pure initial-value problem for (1.1) possesses an infinite set of conservation laws (Zakharov & Shabat, 1972). The conservation in time of the 'energy' or squared L^2-norm

$$E(u) = \int_{-\infty}^{\infty} |u(x, t)|^2 \, dx, \tag{2.3}$$

is of particular significance in the present work. Conservation of energy implies L^2-boundedness of the solutions and also plays an important role in the dynamics of the equation (1.2): the growth of the Fourier modes predicted above by the *linear* theory cannot take place indefinitely if (2.3) is to be conserved. What happens is that the initially unstable Fourier modes draw energy from the stable modes, but due to (2.3) this process must come to an end and in fact it is possible for the energy to return to its initial distribution among the modes (the so-called Fermi–Pasta–Ulam recurrence, see Yuen & Ferguson, 1978).

2.2 *Test Solutions*

(a) *Single Soliton* The single soliton solution is given by

$$u(x, t) = \sqrt{(2\alpha/q)} \exp\{i[\tfrac{1}{2}cx - (\tfrac{1}{4}c^2 - \alpha)t]\} \operatorname{sech}[\sqrt{\alpha}(x - ct)] \tag{2.4}$$

and, for fixed t, decays exponentially as $|x| \to \infty$. The soliton represents a disturbance which travels with speed c and whose amplitude is governed by the real parameter α. Obviously, the initial condition corresponding to (2.4) is

$$g(x) = \sqrt{(2\alpha/q)} \exp(\tfrac{1}{2}icx) \operatorname{sech}(\sqrt{\alpha}x). \tag{2.5}$$

(b) *Collision of Two Solitons* Assume that the initial condition is the superposition of two solitons, a slower one ahead of a faster one in such a way that they are well separated. As time progresses the faster wave catches the slower wave and passes through it in such a manner that the shape and velocity of both waves remain unchanged after the collision, while their phases are shifted.

(c) *Bound States of More Than One Soliton* The initial condition

$$g(x) = \operatorname{sech}(x), \tag{2.6}$$

which according to (2.5) gives rise to a stationary ($c = 0$) soliton with $\alpha = 1$ provided that $q = 2$, may originate more complex phenomena for other values of q. For $q = 2N^2$ ($N = 2, 3, \ldots$) Miles (1981) has shown that (2.6) corresponds to a bound state of N solitons.

3. Numerical methods

3.1 *Space discretization*

In the time interval $0 \leqslant t \leqslant T$ under consideration the solutions of (1.1)–(1.2) we are interested in are negligibly small outside an interval $x_L \leqslant x \leqslant x_R$. Therefore in our numerical study we replace (1.1)–(1.2) by

$$iu_t + u_{xx} + q\,|u|^2 u = 0 \qquad (x_L \leqslant x \leqslant x_R,\ 0 \leqslant t \leqslant T), \tag{3.1}$$

$$u(x,0) = g(x) \quad (x_L \leqslant x \leqslant x_R), \qquad u_x = 0 \text{ at } x = x_L, x_R \quad (0 < t \leqslant T). \tag{3.2}$$

The same replacement has been used by Griffiths, Mitchell, & Morris (1984) and Herbst, Morris, & Mitchell (1984).

For numerical work we decompose the complex function u into its real and imaginary parts v and w, respectively. For the real functions v and w we have the problem

$$\begin{aligned} v_t + w_{xx} + q(v^2+w^2)w = 0 \qquad & (x_L \leqslant x \leqslant x_R,\ 0 \leqslant t \leqslant T), \\ w_t - v_{xx} - q(v^2+w^2)v = 0 \qquad & (x_L \leqslant x \leqslant x_R,\ 0 \leqslant t \leqslant T), \end{aligned} \tag{3.3}$$

$$v(x,0) = g_R(x), \quad w(x,0) = g_I(x) \qquad (x_L \leqslant x \leqslant x_R) \tag{3.4}$$

$$v_x = w_x = 0 \text{ at } x = x_L, x_R \quad (0 < t \leqslant T), \tag{3.5}$$

where g_R and g_I are respectively the real and imaginary parts of g.

Since our interest is in the time-integration, we confine ourselves to a simple finite-difference scheme for the discretization in space. We introduce an equidistant grid $(x_j = x_L + jh : j = 1\,(1)\,N)$ with $h = (x_R - x_L)/(N-1)$, and approximate $v(x_j, t)$ and $w(x_j, t)$ by the solutions $V_j(t)$ and $W_j(t)$ of the semidiscrete system

$$\begin{aligned} \dot{V}_j + h^{-2}(W_{j+1} - 2W_j + W_{j-1}) + q(V_j^2 + W_j^2)W_j = 0 \quad & \text{for } j = 1\,(1)\,N, \\ \dot{W}_j - h^{-2}(V_{j+1} - 2V_j + V_{j-1}) - q(V_j^2 + W_j^2)V_j = 0 \quad & \text{for } j = 1\,(1)\,N, \end{aligned} \tag{3.6}$$

where, in view of (3.5), $V_0 = V_2$, $W_0 = W_2$, $V_{N+1} = V_{N-1}$, $W_{N+1} = W_{N-1}$, and a dot denotes differentiation with respect to time.

Upon introducing the vectors $\boldsymbol{u}_j = [V_j, W_j]^T$ for $j = 1\,(1)\,N$, $\boldsymbol{u} = [\boldsymbol{u}_1^T, \ldots, \boldsymbol{u}_N^T]^T$, the system of ordinary differential equations (3.6) can be written as

$$\dot{\boldsymbol{u}} = \boldsymbol{P}(\boldsymbol{u})\boldsymbol{u} = [\boldsymbol{S} + \boldsymbol{B}(\boldsymbol{u})]\boldsymbol{u}, \tag{3.7a}$$

where $\boldsymbol{S}$ is the block-tridiagonal matrix

$$\boldsymbol{S} = -h^{-2}\begin{bmatrix} -2\mathbf{A} & 2\mathbf{A} & & & \\ \mathbf{A} & -2\mathbf{A} & \mathbf{A} & & 0 \\ & \ddots & \ddots & \ddots & \\ & 0 & \mathbf{A} & -2\mathbf{A} & \mathbf{A} \\ & & & 2\mathbf{A} & -2\mathbf{A} \end{bmatrix} \tag{3.7b}$$

with

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix},$$

and $\boldsymbol{B}(\boldsymbol{u})$ is block-diagonal:

$$\boldsymbol{B}(\boldsymbol{u}) = -\mathrm{diag}\,[\boldsymbol{B}_1(\boldsymbol{u}_1), \ldots, \boldsymbol{B}_N(\boldsymbol{u}_N)],$$
$$\boldsymbol{B}_j(\boldsymbol{u}_j) = \begin{bmatrix} 0 & V_j^2 + W_j^2 \\ -V_j^2 - W_j^2 & 0 \end{bmatrix} \quad \text{for } j = 1\,(1)\,N.$$

From the definitions of S and $B(\boldsymbol{u})$ we conclude that for any $\boldsymbol{u}, \tilde{\boldsymbol{u}} \in \mathbb{R}^{2N}$

$$\langle \boldsymbol{S}\boldsymbol{u}, \boldsymbol{u}\rangle = 0, \qquad \langle \boldsymbol{B}(\tilde{\boldsymbol{u}})\boldsymbol{u}, \boldsymbol{u}\rangle = 0, \tag{3.8}$$

where $\langle \bullet, \bullet \rangle$ denotes the inner product

$$\langle \boldsymbol{u}, \tilde{\boldsymbol{u}}\rangle = h\left(\tfrac{1}{2}\boldsymbol{u}_1^{\mathsf{T}}\tilde{\boldsymbol{u}}_1 + \sum_{j=2}^{N-1} \boldsymbol{u}_j^{\mathsf{T}}\tilde{\boldsymbol{u}}_j + \tfrac{1}{2}\boldsymbol{u}_N^{\mathsf{T}}\tilde{\boldsymbol{u}}_N\right).$$

The skew-symmetry relations (3.8) imply in turn that for solutions $\boldsymbol{u}(t)$ of (3.7) the quantity $\|\boldsymbol{u}\|^2 = \langle \boldsymbol{u}, \boldsymbol{u}\rangle$ is conserved in the evolution in time. This is, of course, the discrete analogue of the conservation of the energy (2.3).

One of the referees has pointed out that it is possible to divide the equations in the first and last blocks of (3.7) so as to render the matrix $\boldsymbol{S}$ symmetric thus enhancing computational efficiency. The experiments reported here correspond to implementations using the unsymmetric matrix given in (3.7b).

3.2 *Integration in Time*

We now consider several methods for the integration in time of the semi-discrete system (3.7). All the methods studied are *second-order accurate* (in time) and of the one-step type $\boldsymbol{u}^n \to \boldsymbol{u}^{n+1}$ ($t_n \to t_{n+1} = t_n + \tau$) with τ the step-size in time and $\boldsymbol{u}^n$ the fully discrete approximation at $t = t_n$. The methods considered are the implicit midpoint rule and four modifications of it. Due to the aims of the paper, we were interested in comparing *conservative* and *nonconservative* methods which were not widely different from each other, so that any observed advantages may be related to the differences in treating the conservation issue. By this reason we have not tested the explicit conservative scheme of Sanz-Serna (Sanz-Serna, 1982; Sanz-Serna & Manoranjan, 1983; cf. Herbst *et al.* 1984).

METHOD 0 This is the implicit midpoint rule

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}^n + \tau \boldsymbol{P}\left(\frac{\boldsymbol{u}^n + \boldsymbol{u}^{n+1}}{2}\right)\frac{\boldsymbol{u}^n + \boldsymbol{u}^{n+1}}{2}. \tag{3.9}$$

It is well known (Sanz-Serna, 1982, Verwer & Dekker, 1984) that solutions of (3.9) possess the conservation property $\|\boldsymbol{u}^n\| = \|\boldsymbol{u}^{n+1}\|$, which mimics the analogous conservation properties of the original PDE and the semidiscrete system of ODEs. This property ensures the boundedness, as $n \to \infty$, of the approximations $\boldsymbol{u}^n$, thus ruling out the occurrence of nonlinear blow-up (Sanz-Serna, 1982; Sanz-Serna & Manoranjan, 1983; Verwer & Dekker, 1984; cf. Morton, 1977). Methods for which $\|\boldsymbol{u}^n\| = \|\boldsymbol{u}^{n+1}\|$ are called *conservative.*

An efficient implementation of the implicit method (3.9) is discussed after

method 1. The convergence of method (3.9) is investigated in Verwer & Sanz-Serna, 1984. Further properties will be mentioned when required.

METHOD 1 The previous method demands that at each time level a system of $2N$ nonlinear algebraic equations be solved. In order to save computational effort one may consider treating the nonlinear part $\boldsymbol{B}(\boldsymbol{u})\boldsymbol{u}$ of (3.7) in an explicit way. Note that this is reasonable from the stability point of view, since $\boldsymbol{B}(\boldsymbol{u})\boldsymbol{u}$ does not contain the space mesh-size h and therefore does not contribute to the stiffness of (3.7). Griffiths *et al.* (1984) suggest the method

$$\boldsymbol{u}^* = \boldsymbol{u}^n + \tau \boldsymbol{P}(\boldsymbol{u}^n)\boldsymbol{u}^n, \tag{3.10a}$$

$$\left(\boldsymbol{I} - \frac{\tau}{2}\boldsymbol{S}\right)\boldsymbol{u}^{n+1} = \left(\boldsymbol{I} + \frac{\tau}{2}\boldsymbol{S}\right)\boldsymbol{u}^n + \frac{\tau}{2}\boldsymbol{B}\left(\frac{\boldsymbol{u}^n + \boldsymbol{u}^*}{2}\right)(\boldsymbol{u}^n + \boldsymbol{u}^*). \tag{3.10b}$$

Now $\boldsymbol{u}^{n+1}$ is obtained by solving a system of *linear* equations, whose matrix does not change with n and therefore can be factorized once and for all. Thus at each time level only a forward and a backward solve are required. For efficiency it is advantageous to implement (3.10b) in the form

$$\left(\boldsymbol{I} - \frac{\tau}{2}\boldsymbol{S}\right)\hat{\boldsymbol{u}} = \boldsymbol{u}^n + \frac{\tau}{4}\boldsymbol{B}\left(\frac{\boldsymbol{u}^n + \boldsymbol{u}^*}{2}\right)(\boldsymbol{u}^n + \boldsymbol{u}^*), \qquad \boldsymbol{u}^{n+1} = 2\hat{\boldsymbol{u}} - \boldsymbol{u}^n. \tag{3.10c}$$

The Method 1 defined by (3.10) is *not* conservative (cf. Section 4).

IMPLEMENTATION OF METHOD 0 We now leave Method 1 and mention that iteration to convergence of the predictor–corrector (3.10) (Griffiths *et al.* 1984) provides an efficient technique for the implementation of method 0. (This was the implementation used in our numerical experiments.) Namely, when $\boldsymbol{u}^n$ has been obtained, we compute $\boldsymbol{u}^*$ according to (3.10a) and then employ the corrector stages ($\boldsymbol{u}_{[0]} = \boldsymbol{u}^*$) for $r = 0, 1, 2, \ldots$:

$$\left(\boldsymbol{I} - \frac{\tau}{2}\boldsymbol{S}\right)\boldsymbol{u}_{[r+1]} = \left(\boldsymbol{I} + \frac{\tau}{2}\boldsymbol{S}\right)\boldsymbol{u}^n + \frac{\tau}{2}\boldsymbol{B}\left(\frac{\boldsymbol{u}^n + \boldsymbol{u}_{[r]}}{2}\right)(\boldsymbol{u}^n + \boldsymbol{u}_{[r]}),$$

until two consecutive iterants $\boldsymbol{u}_{[r]}$ and $\boldsymbol{u}_{[r+1]}$ are found which differ under $\|\cdot\|$ less than a prescribed tolerance (This was chosen to be 10^{-4} in our experiments.) Then $\boldsymbol{u}_{[r+1]}$ is taken to be $\boldsymbol{u}^{n+1}$. In this implementation, method 0 only requires the factorization of the matrix $\boldsymbol{I} - \frac{1}{2}\tau\boldsymbol{S}$ at the beginning of the computation and then an unspecified number of back and forward solves per time step. Note that the corrector stages above can be regarded as a modified Newton iteration for (3.9) where the true Jacobian has been replaced by $\boldsymbol{I} - \frac{1}{2}\tau\boldsymbol{S}$, thus disregarding the contribution of the nonlinear terms. This contribution is expected to be small since $\boldsymbol{B}(\boldsymbol{u})\boldsymbol{u}$ does not include negative powers of h. The corrector stages are best implemented in the efficient form (3.10c).

METHOD 2 This is an attempt to achieve the property of conservation enjoyed by method 0 under the requirement that only a linear system be solved per step. Following Verwer & Dekker (1984) we consider the pseudolinear midpoint rule

$$\boldsymbol{u}^* = \boldsymbol{u}^n + \tfrac{1}{2}\tau\boldsymbol{P}(\boldsymbol{u}^n)\boldsymbol{u}^n, \qquad \boldsymbol{u}^{n+1} = \boldsymbol{u}^n + \tfrac{1}{2}\tau\boldsymbol{P}(\boldsymbol{u}^*)(\boldsymbol{u}^n + \boldsymbol{u}^{n+1}). \tag{3.11a,b}$$

The conservation property is easily shown to hold in view of (3.8). Note that while only one linear system appears per step, the corresponding matrix now *changes* with n. Again (3.11b) can be rewritten in the more efficient form

$$\boldsymbol{I} - \tfrac{1}{2}\tau \boldsymbol{P}(\boldsymbol{u}^*)\hat{\boldsymbol{u}} = \boldsymbol{u}^n, \qquad \boldsymbol{u}^{n+1} = 2\hat{\boldsymbol{u}} - \boldsymbol{u}^n. \tag{3.11c}$$

METHOD 3 We examine this method in order to show that the advantages of methods 0–1 (no LU-decomposition except in the first step), 0–2 (conservation), and 1–2 (only one $2N$-dimensional linear system per step) can be brought together. Consider, with $\boldsymbol{u}^*$ given by (3.11a)

$$\mathbf{y}_{[1]} = \boldsymbol{u}^n + \tfrac{1}{4}\tau \boldsymbol{B}(\boldsymbol{u}^*)(\boldsymbol{u}^n + \mathbf{y}_{[1]}), \qquad \mathbf{y}_{[2]} = \mathbf{y}_{[1]} + \tfrac{1}{2}\tau \boldsymbol{S}(\mathbf{y}_{[1]} + \mathbf{y}_{[2]}), \tag{3.12a,b}$$

$$\boldsymbol{u}^{n+1} = \mathbf{y}_{[2]} + \tfrac{1}{4}\tau \boldsymbol{B}(\boldsymbol{u}^*)(\mathbf{y}_{[2]} + \boldsymbol{u}^{n+1}). \tag{3.12c}$$

The complete step $\boldsymbol{u}^n \to \boldsymbol{u}^{n+1}$ consists of three so-called fractional steps. The first, $\boldsymbol{u}^n \to \mathbf{y}_{[1]}$, is just a step with the pseudo-linear midpoint rule applied to $\dot{\boldsymbol{u}} = \boldsymbol{B}(\boldsymbol{u})\boldsymbol{u}$ using a step-size $\frac{1}{2}\tau$. The fractional step $\mathbf{y}_{[1]} \to \mathbf{y}_{[2]}$ is a midpoint rule step applied to $\dot{\boldsymbol{u}} = \boldsymbol{S}\boldsymbol{u}$ with step-size τ. Finally the third fractional step $\mathbf{y}_{[2]} \to \boldsymbol{u}^{n+1}$ is similar to the first. In the literature methods of the present type are called fractional step (Yanenko, 1971) or splitting methods (e.g. Verwer, 1984). Due to the relations (3.8) the complete step is conservative. More precisely $\|\boldsymbol{u}^{n+1}\| = \|\mathbf{y}_{[2]}\| = \|\mathbf{y}_{[1]}\| = \|\boldsymbol{u}^n\|$.

The method is accurate to the second order due to the symmetry of the splitting employed and the second order of each the fractional formulas when they are considered as methods on their own (Strang, 1968). As before each fractional step can be written in a more efficient form. Note that the first and third fractional steps are very cheap in view of the *block-diagonal* structure of the matrix involved. The only system with $2N$ unknowns to be solved is that of the second fractional step and the corresponding matrix is independent of n. The computational cost of Method 3 is nearly equal to that of Method 1.

Splitting schemes appear to be particularly attractive in the study of the generalization of (1.1) to several space variables (replacing u_{xx} by the Laplacian Δu).

METHOD 4 In Method 3 we twice use the nonlinear part $\boldsymbol{B}(\boldsymbol{u})\boldsymbol{u}$ and only once the linear part $\boldsymbol{S}\boldsymbol{u}$. An obvious alternative reads as follows:

$$\mathbf{y}_{[1]} = \boldsymbol{u}^n + \tfrac{1}{4}\tau \boldsymbol{S}(\boldsymbol{u}^n + \mathbf{y}_{[1]}), \qquad \mathbf{y}_{[2]} = \mathbf{y}_{[1]} + \tfrac{1}{2}\tau \boldsymbol{B}(\boldsymbol{u}^*)(\mathbf{y}_{[1]} + \mathbf{y}_{[2]}), \tag{3.13a,b}$$

$$\boldsymbol{u}^{n+1} = \mathbf{y}_{[2]} + \tfrac{1}{4}\tau \boldsymbol{S}(\mathbf{y}_{[2]} + \boldsymbol{u}^{n+1}), \tag{3.13c}$$

where $\boldsymbol{u}^*$ is defined as previously. This scheme is slightly more expensive than Method 3. Of course the conservation property remains unchanged.

4. Numerical tests

The methods described in the previous section were tested on a set of increasingly difficult problems as follows.

4.1 *Single Soliton*

Here the initial profile given by (2.5) was tested for several values of α, q, and c. For each choice of these parameters the semidiscrete system (3.6) was numerically integrated with high accuracy by means of a Runge–Kutta–Fehlberg ODE package for decreasing values of h until a value was found for which the semidiscrete solution was a good approximation to the PDE solution. The quality of the approximation was investigated both by producing tables of errors and by drawing plots of the semidiscrete solutions, the latter being more apt to show undesirable features such as spurious oscillations, phase errors, etc... Once a suitable value of h was found (for the single soliton problem this value of h turned out to be 0·5), the five methods were tried with a variety of values of the time step τ and again tables and plots were produced. In order not to render this paper unduly long, we avoid presenting in detail this part of our experiments. The general conclusions were that Methods 0 and 2–4 performed well. In fact when accuracy and efficiency were taken into account there was little difference between them. Method 0 was the most accurate and expensive, followed by Methods 2, 4, and 3. Method 3 was 2·5 times faster than Method 2, and 1·5 times faster than Method 4. The computational cost of Method 0 was highly problem-dependent due to the unspecified number of linear systems to be solved per step. For small values of τ the predictor provides a good initial guess for the solution of the nonlinear system (3.9) and Method 0 was only marginally more expensive than Method 2. For larger τ Method 0 becomes more expensive (cf. Section 4.2 below). In Methods 0, 2, 3, and 4 the growth of the time integration error obtained by comparison with the highly accurate RKF solution and measured in the previously introduced L^2 norm in $\mathbb{R}^{2N}$ was approximately *linear* in t.

The performance of Method 1 was unsatisfactory: very often the computation led to machine overflow. For instance, when $\alpha = 1{\cdot}00$, $c = 1{\cdot}00$, $q = 1{\cdot}00$, $h = 0{\cdot}50$, $x_L = -30$, $x_R = 70$, and $\tau = 0{\cdot}250$ the computation blew up at $t \simeq 5$. Reduction of τ to 0·125 deferred the explosion until $t \simeq 28$, but did not avoid it. We emphasize that this form of instability—sometimes called nonlinear blow-up (Sanz-Serna, 1984a)—only becomes apparent after many steps of the computation have been successfully performed; see Fig. 1 corresponding to $\tau = 0{\cdot}125$ and the set of parameters quoted above. The figures display the modulus $|u| = \sqrt{(v^2 + w^2)}$ as a function of x and t. The real and imaginary parts are oscillatory; see (2.4). It should also be observed that, as pointed out by Griffiths *et al.* (1984) and predicted by the analysis below, the amplitude is crucial for the blow-up time of Method 1. For example, for $\alpha = 0{\cdot}5$ the blow-up times are at least $t = 50$ and 70 for $\tau = 0{\cdot}250$ and 0·125, respectively. This is consistent with the experiences reported by Griffiths, Mitchell, and Morris.

In order to gain insight into the mechanism of nonlinear blow-up, we note that when Method 1 is used to integrate the x-independent solutions (Section 2.1.b) it reduces to the RK procedure defined by the array

$$\begin{array}{c|cc} 0 & & \\ \frac{1}{2} & 0 & \\ \hline 0 & 1 & \end{array} \tag{4.1}$$

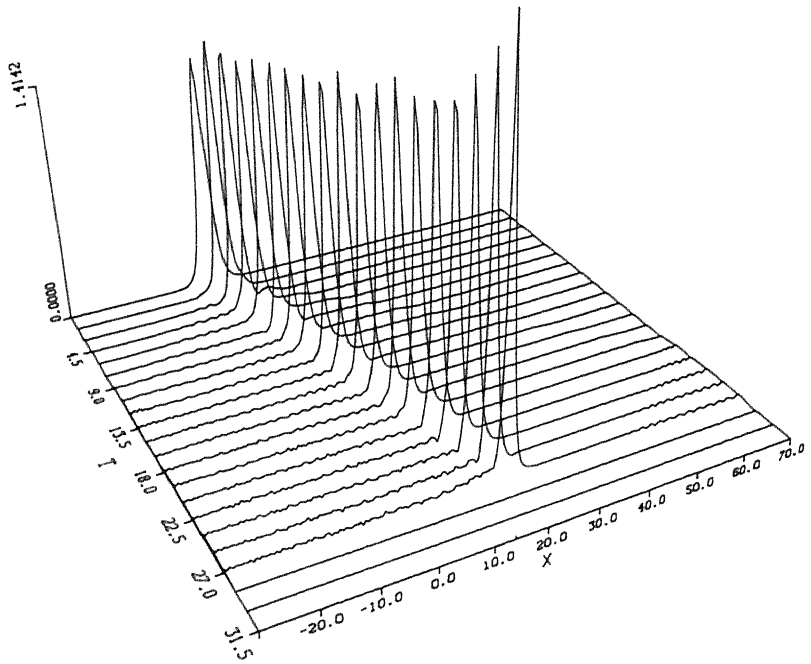

FIG. 1.

applied to the complex ODE (2.2). (Lambert, 1973, p. 118 refers to (4.1) as the improved polygon method.) Note that this reduction to an RK procedure is independent of the spatial discretization used in Method 1. Therefore what follows is also valid if the discretization in space had been carried out by means of a Galerkin method as those considered by Griffiths *et al.* (1984).

The equation (2.2) can be put in real form

$$\dot{\xi} = -q(\xi^T\xi)\mathbf{A}\xi, \tag{4.2}$$

where $\xi \in \mathbb{R}^2$ is the vector $[\text{Re } u, \text{Im } u]^T$ and the matrix $\mathbf{A}$ is as in Section 3.1. The system (4.2) conserves the energy $\xi^T\xi = \|\xi\|_2^2$. Upon introducing the energy $e_n = \|\xi_n\|_2^2$ of the approximations generated by the method (4.1), the following recursion can be found

$$e_{n+1} = e_n + q^4\frac{\tau^4}{2}e_n^5\left(1 + q^2\frac{\tau^2}{4}e_n^2\right)\left(1 + q^2\frac{\tau^2}{8}e_n^2\right). \tag{4.3}$$

(The derivation of (4.3) can be seen in Griffiths *et al.* (1984), but there is an error in their final formula.) We conclude that the *increase* in energy per step is small: $O(\tau^4)$. However note that (4.3) can be seen as a one-step method with step-length τ^4, for the integration of the ODE

$$\dot{e} = \frac{q^4}{2}e^5 \tag{4.4}$$

with initial value $e(0)=e_0$. The solution of this problem is given by

$$-\frac{1}{4e^4}=\frac{q^4}{2}t-\frac{1}{4e_0^4}$$

and therefore becomes infinite when $t=(2q^4e_0^4)^{-1}$. Hence we conclude that in the integration of x-independent solutions, Method 1 blows up after approximately $n^*=(2q^4e_0^4)^{-1}/\tau^4$ steps. This prediction was found to agree very well with numerical experiments *concerning x-independent solutions*. Note that n^* decreases as the initial energy increases.

Essentially, the mechanism leading to the blow-up is as follows. The small term $\frac{1}{2}q^4\tau^4e_n^5+O(\tau^6)$ added to e_n renders $e_{n+1}>e_n$ and this results in a larger increment $\frac{1}{2}q^4\tau^4e_{n+1}^5+O(\tau^6)$ at the next time step. Since the feedback is proportional to a high power of e_n rather than to e_n itself, the growth is more violent than exponential. We remark that if (4.2) is replaced by the *linear* problem

$$\dot{\xi}=-q\mathbf{A}\xi, \tag{4.5}$$

then (4.3) and (4.4) become, respectively

$$e_{n+1}=e_n+q^4\frac{\tau^4}{4}e_n, \qquad \dot{e}=\frac{q^4}{4}e,$$

and the solutions of this ODE, although increasing with t, do not blow up at finite time.

It is perhaps useful to point out that, for (4.5), the increase in $\|\xi_n\|$ can be predicted from the fact that the region of absolute stability of (4.1) does not intersect the imaginary axis, which contains the eigenvalues of $\mathbf{A}$.

The techniques in this subsection have been expanded by Sanz-Serna & Verwer (1984). An appendix is devoted to a study of energy growth in Runge–Kutta methods.

4.2 *Collision of Two Solitons*

Now $x_L=-20$, $x_R=80$, $T=44$, and the initial condition was taken to be

$$g(x)=\sqrt{(2\alpha/q)}\{\exp(\tfrac{1}{2}ic_1x)\operatorname{sech}(x\sqrt{\alpha})+\exp[\tfrac{1}{2}ic_2(x-\delta)]\operatorname{sech}[(x-\delta)\sqrt{\alpha}]\}.$$

We chose $\alpha=0{\cdot}5$, $q=1{\cdot}0$, $c_1=1{\cdot}0$, and $c_2=0{\cdot}1$, while the parameter δ governing the initial location of the slower soliton was given the value $\delta=25$. Again we employed a RKF package to find a value of h for which the semidiscrete solution provided a satisfactory description of the interaction. This value of h turned out to be $h=0{\cdot}25$. Then we applied methods 0, 2, 3, and 4 and measured the errors with respect to the semidiscrete solution. (Method 1 was discarded due to its failure in the preceding problem.) The results for $\tau=0{\cdot}25$ and $\tau=0{\cdot}125$ are given in Table 1 and correspond to the L^2 norm in $\mathbb{R}^{2N}$. Recall that, in practice, Method 0 does not conserve energy exactly because system (3.9) is not exactly solved for $\boldsymbol{u}^{n+1}$ (see also §4). By way of illustration, Table 1 also shows the energy behaviour of method 0 which is quite acceptable. At $t=0$ the computed energy is 2·378414.

TABLE 1

t	Method 0	2	3	4	Energy in Method 0
		($\tau = 0{\cdot}25$)			
8·0	0·101	0·141	0·283	3·306	2·378497
16·0	0·191	0·261	0·523	3·048	2·378579
24·0	0·439	0·536	1·503	3·436	2·378631
32·0	0·715	0·803	2·572	3·555	2·378638
40·0	1·222	1·197	3·578	3·266	2·378717
		($\tau = 0{\cdot}125$)			
8·0	0·021	0·034	0·064	0·038	2·378355
16·0	0·034	0·063	0·116	0·069	2·378297
24·0	0·095	0·133	0·280	0·155	2·378286
32·0	0·164	0·204	0·706	3·082	2·378254
40·0	0·216	0·282	1·434	3·640	2·378194

Methods 0 and 2 performed well. When $\tau = 0{\cdot}25$ the CPU times for 176 steps were 20·9 and 12·3 seconds, respectively. For the smaller value $\tau = 0{\cdot}125$ Method 0 becomes more competitive for the reason outlined before and those times become respectively 25·3 and 24·9 for 352 steps. Note that the cost in Method 2 is almost proportional to the number of steps. When $\tau = 0{\cdot}25$ method 0 required 934 inner iterations (applications of the corrector) to complete 176 time steps yielding an average of 5·3 linear systems per step. When $\tau = 0{\cdot}125$ the average was 3·17 systems per time step. The time steps corresponding to the actual collision required more inner iterations per step than those preceding or following the collision. Figure 2 depicts the interaction as integrated by Method 0 with $\tau = 0{\cdot}125$.

The performance of the splitting methods 3 and 4 was poor as can be seen in the table. Figure 3 corresponds to method 4 with $\tau = 0{\cdot}125$. From this plot we see that the large errors in the splitting methods arise from the fact that they break the balance between nonlinearity and dispersion. In fact in a splitting method the dispersive and nonlinear forces act *successively* rather than *simultaneously*. The linear fractional steps act dispersively and tend to 'spread' the solution over the x-axis. This spreading cannot be eliminated by the nonlinear fractional steps since in the latter there is no coupling between adjacent space grid points x_j and x_{j+1}.

From an analytic point of view we note that in the step $t \to t+\tau$ the evolution given by (3.7) is *replaced* by *successive* evolutions according to the equations

$$\dot{\boldsymbol{u}} = \boldsymbol{B}(\boldsymbol{u})\boldsymbol{u}, \qquad \dot{\boldsymbol{u}} = \boldsymbol{S}\boldsymbol{u}, \tag{4.6}$$

and the argument above shows that there is an error E_{split} associated with this *replacement*. The local error of a splitting numerical method consists of the splitting error E_{split} plus the local errors E_1 and E_2 associated with the replacement of (4.6) for their numerical counterparts (i.e. the individual local errors of the fractional steps). In our situation E_1 and E_2 are not too large due to the small

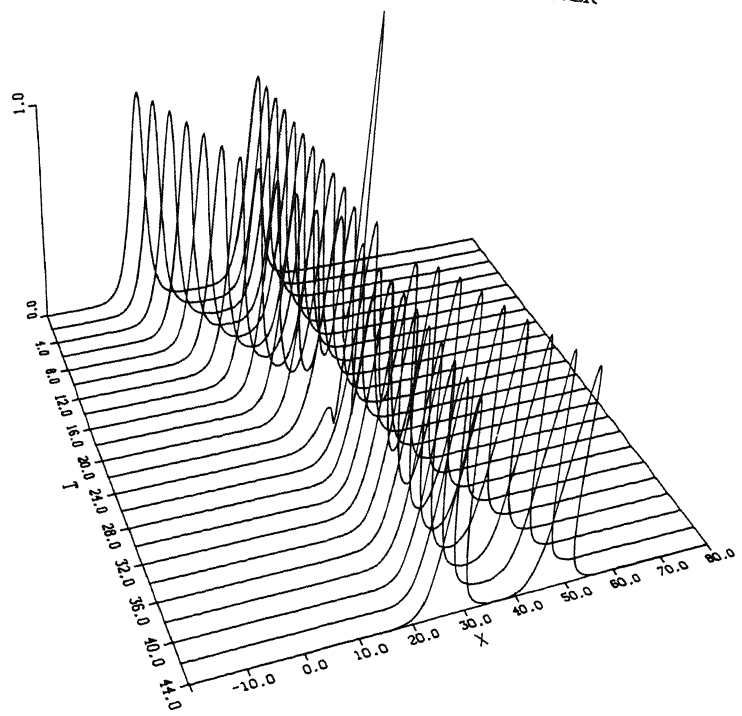
1.0
0.0
T
0.0
4.0
8.0
12.0
16.0
20.0
24.0
28.0
32.0
36.0
40.0
44.0
X
-10.0
0.0
10.0
20.0
30.0
40.0
50.0
60.0
70.0
80.0

FIG. 2.

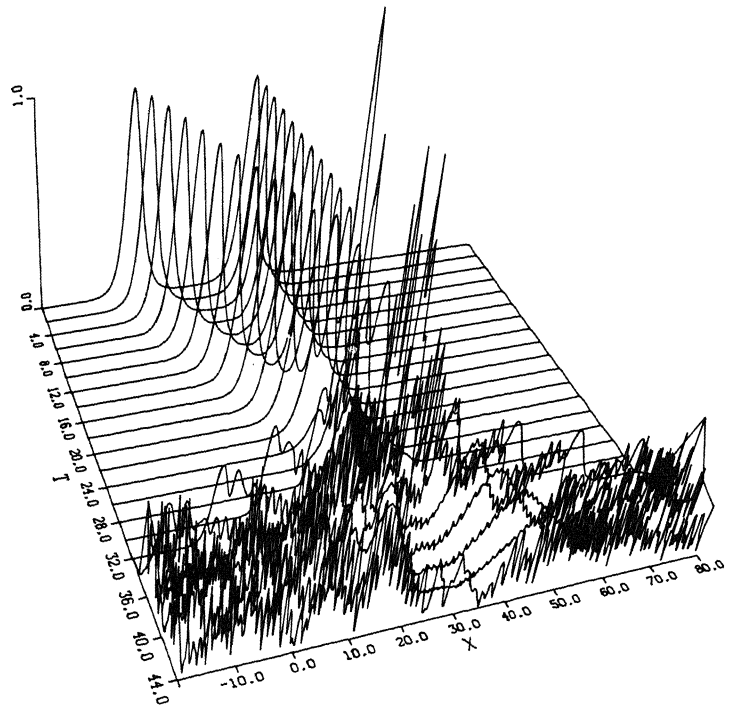
1.0
0.0
T
0.0
4.0
8.0
12.0
16.0
20.0
24.0
28.0
32.0
36.0
40.0
44.0
X
-10.0
0.0
10.0
20.0
30.0
40.0
50.0
60.0
70.0
80.0

FIG. 3.

values employed for h and τ. Further treatment of this point is outside the scope of this paper and the reader is referred to Leveque & Oliger (1983) for a rigorous analysis of a similar situation..

4.3 *Bound States of More Than One Soliton*

Now $q = 18$, $x_L = -20$, $x_R = 20$, $T = 2{\cdot}5$, and the initial condition is given by (2.6). Herbst *et al.* (1983) have shown this problem to be a difficult test, since the solutions develop steep spatial and temporal gradients. Upon using the RKF code it was found that h had to be reduced to $h = 1/32 = 0{\cdot}03125$ in order that the semidiscrete solution provided a good description of the phenomenon studied. Only Methods 0 and 2 were tested, since the other schemes had failed in easier problems. Figures 4 and 5 correspond to $\tau = 1/160 = 0{\cdot}00625$ and show the superiority of method 0. The CPU times for Methods 0 and 2 were 133 and 92 seconds, respectively. In the midpoint rule an average of five applications of the corrector per step was required. However some steps needed as many as 10 corrections. When τ was halved ($\tau = 1/320$), Method 2 was able to integrate the problem successfully. The CPU times were then 186 and 178, respectively.

From the experiments in this and the preceding paragraphs, we conclude that for the present Schrödinger equation Method 2 is less accurate than Method 0 and therefore would require a smaller value of τ. But for small values of τ

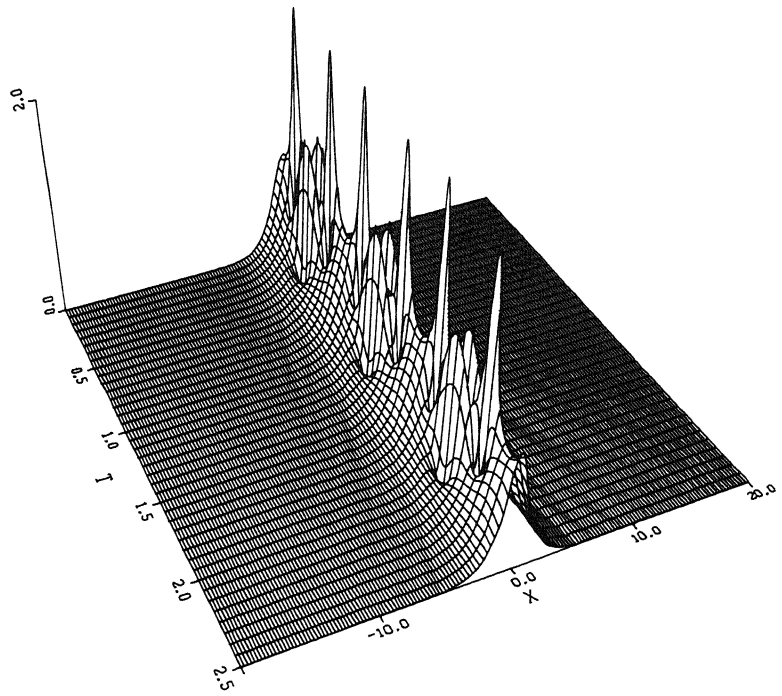

FIG. 4.

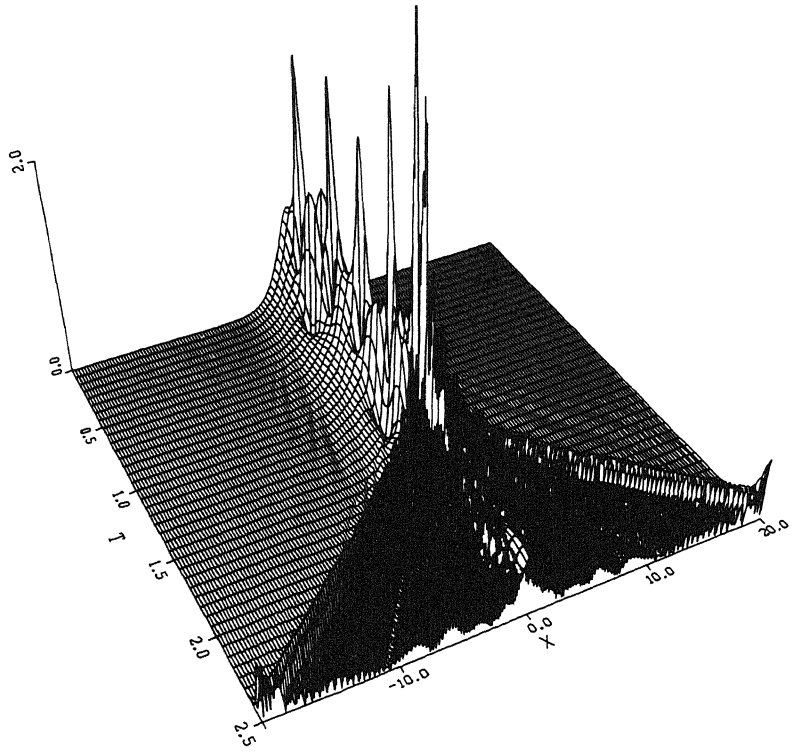

FIG. 5.

Method 2 is no longer advantageous in computational effort. Thus, in the present authors' opinion Method 0 is the best among those tested in this paper, although the difference in performance of Methods 0 and 2 is not large.

An explanation of the difference in accuracy between Methods 0 and 2 will now be provided. For simplicity we consider the scalar ODE

$$\dot{y} = P(y)y, \tag{4.7}$$

although what follows is easily extended to systems in $\mathbb{R}^d$, by replacing ordinary derivatives by Fréchet derivatives. Differentiation of (4.7) yields

$$\ddot{y} = p'(y)\dot{y}y + p(y)\dot{y}, \quad \dddot{y} = p''(y)(\dot{y})^2 y + p'(y)\ddot{y}y + 2p'(y)(\dot{y})^2 + p(y)\ddot{y}.$$

For the midpoint rule the residual or truncation error

$$\varepsilon_T = y(t+\tau) - y(\tau) - \tau p(\tfrac{1}{2}[y(t+\tau) + y(t)])\tfrac{1}{2}[y(t+\tau) + y(t)]$$

is easily seen to have the expansion

$$\varepsilon_T = \tau^3[\tfrac{1}{24}p''(\dot{y})^2 y - \tfrac{1}{12}p'\ddot{y}y + \tfrac{1}{12}p'(\dot{y})^2 - \tfrac{1}{12}p\ddot{y}] + O(\tau^4), \tag{4.8}$$

where y is evaluated at t and p, p', p'' are evaluated at $y(t)$. For the pseudolinear midpoint rule the residual is given by

$$\varepsilon_T^* = \tau^3[\tfrac{1}{24}p''(\dot{y})^2 y + \tfrac{1}{6}p'\ddot{y}y + \tfrac{1}{12}p'(\dot{y})^2 - \tfrac{1}{12}p\ddot{y}] + O(\tau^4). \tag{4.9}$$

We see that the leading terms in ε_T and ε_T^* are composed of the same elementary differentials. In fact they are equal except for the coefficient of $p'\ddot{y}y$ which is larger in ε_T^*. We owe the somewhat higher accuracy of Method 0 for our Schrödinger equation to this difference. On the other hand, in general some cancellation between terms may occur and thus it is possible for ε_T^* to be in some instances smaller than ε_T. We found experimentally this to be the case for the ODE (2.2), where Method 2 was more accurate than Method 0.

Before we close this section a comment should be made on the possibility of blow up in Method 0. It was noted above that if $\boldsymbol{u}^n$ and $\boldsymbol{u}^{n+1}$ satisfy (3.9) then $\|\boldsymbol{u}^n\| = \|\boldsymbol{u}^{n+1}\|$ so that no blow-up can take place if the nonlinear system (3.9) is exactly solved for $\boldsymbol{u}^{n+1}$. Also note that the existence of solutions $\boldsymbol{u}^{n+1}$ of (3.9) is only guaranteed for τ suitably small (Sanz-Serna, 1984b). If τ is large or if the stopping criterion used in the iterative solution of the nonlinear system is not very demanding, it is possible that the vector returned by the code at the end of the step possesses a norm much larger than $\|\boldsymbol{u}^n\|$ and this growth may lead to machine overflow. We experienced such an overflow in the integration of the bound state with $h = 0{\cdot}125$ and $\tau = 0{\cdot}0125$. For this value of h the semidiscrete system does not approximate accurately the theoretical solution and, in fact, the semidiscrete solution presents huge spatial and temporal gradients. In the fourteenth time step the maximum allowed number of corrections (twenty) was reached before the criterion of convergence of the iteration was met (the norm of the difference between the 19th and 20th iterant was $5{\cdot}0_{10}-4$). The iterants in the fifteenth time-step showed no convergence whatsoever, so that the vector returned by the machine as $\boldsymbol{u}^{15}$ has to be regarded as suspect. During the sixteenth time step, overflow took place. Similar overflows have been reported by Herbst *et al.* (1984).

It is clear that in writing codes, failures in the convergence of the inner iteration should be regarded as suggestions that the current value of τ is too large for the problem at hand and that a smaller τ should be attempted. Also note that Method 0, implemented in the predictor–corrector manner described here, renders itself easily to variable-step control. Variable steps would no doubt be essential in the integration of realistic problems.

5. Conclusions

Five methods for the integration in time of a semidiscretization of the nonlinear Schrödinger equation have been extensively tested. Three of them (a partly explicit scheme and two splitting procedures) have been found to perform poorly and the reasons for their failures have been analysed. Our analysis has included a detailed investigation of an instance of the so-called nonlinear blow-up.

From a more general point of view, the experiments in this paper throw light into the advantages and drawbacks associated with the use of time-integrators which *conserve energy exactly* (cf. Morton, 1977; Sanz-Serna, 1982; Sanz-Serna, 1984b; Sanz-Serna & Manoranjan, 1983; Verwer & Dekker, 1983; Delfour *et al.* 1981). From the experience gained in this paper and those by Herbst *et al.* (1984) and Sanz-Serna & Christie (1985), the following conclusions appear to emerge.

(i) Exact conservation does not necessarily guarantee the success of a method

as exemplified by the splitting schemes considered in this paper. These are conservative and '*a priori*' could have been regarded as 'sound' and yet in practice performed poorly. Likewise, Sanz-Serna & Christie (1985) report that a modification of the Crank–Nicolson scheme so as to render it conservative resulted in a decrease in the accuracy.

(ii) Lack of exact conservation may lead to the undesirable *nonlinear* blow-up (Morton, 1977) as shown by Method 1 in this paper. However the energy growth in this method could have been forecast by an analysis of the usual (*linear*) *region of stability* of the method (Section 4.1).

(iii) There are useful numerical schemes which perform in a very stable way and yet do not conserve energy exactly. See, among others, the experiments in Herbst *et al.* (1984) and Sanz-Serna & Christie (1985).

Our conclusions agree with those of Schamel & Elsässer (1976).

Acknowledgements

The authors are extremely thankful to Mrs. J. Blom for her expert programming. J.M.S. has received financial support from 'Comision Asesora'.

References

Dekker, K., & Verwer, J. G. 1984 *Stability of Runge-Kutta Methods for Stiff Nonlinear Differential Equations*. Amsterdam: North-Holland.

Delfour, M., Fortin, M., & Payne, G. 1981 Finite-difference solutions of a nonlinear Schroedinger equation. *J. Comp. Phys.* **44,** 277–288.

Griffiths, D. F., Mitchell, A. R., Morris, J. Li. 1984 A numerical study of the nonlinear Schroedinger equation, *Comp. Meth. Appl. Mech. Eng.* **45,** 177–215.

Herbst, B. M., & Mitchell, A. R. 1983 The numerical stability of the nonlinear Schroedinger equation. *Report NA/66*, University of Dundee (revised version).

Herbst, B. M., Morris, J. Li., & Mitchell, A. R. 1984 Numerical experience with the nonlinear Schroedinger equation. *J. Comp. Phys.* (to appear).

Lambert, J. D. 1973 *Computational Methods in Ordinary Differential Equations*, London: John Wiley.

Leveque, R. J., & Oliger, J. 1983 Numerical methods based on additive splittings for hyperbolic partial differential equations. *Math. Comp.* **40,** 469–497.

Miles, J. W. 1981 An envelope soliton problem, *SIAM J. Appl. Math.* **41,** 227–230.

Morton, K. W. 1977 Initial value problems by finite difference and other methods. In: *The State of the Art in Numerical Analysis* (D. A. H. Jacobs, Ed.), p. 699–756. Academic Press: London.

Sanz-Serna, J. M. 1982 An explicit finite difference scheme with exact conservation properties. *J. Comp. Phys.* **52,** 273–289.

Sanz-Serna, J. M. 1984 Nonlinear instability of leap frog schemes. In: *Numerical methods for nonlinear problems* (C. Taylor, E. Hinton, D. R. J. Owen, & E. Onate, Eds.), p. 77–86, Swansea: Pineridge Press.

Sanz-Serna, J. M. 1984b. Methods for the numerical solution of the nonlinear Schroedinger equation. *Math. Comp.* **43,** 21–27.

Sanz-Serna, J. M., & Christie, I. 1985 Finite elements for nonlinear integro-differential equations and their integration in time. In *Proceedings MAFELAP 84*, J. Whiteman (ed.) London: Academic Press.

Sanz-Serna, J. M., & Manoranjan, V. S. 1983 A method for the integration in time of certain partial differential equations. *J. Comp. Phys.* **52,** 273–289.

SANZ-SERNA, J. M., & VERWER, J. G. 1984 A study of the recursion $y_{n+1} = y_n + \tau y_n^m$. *Report NM-R*8403, Centre for Math. and Comp. Sc., Amsterdam.

SCHAMEL, H., & ELSÄSSER, K. 1976 The application of the spectral method to nonlinear wave propagation. *J. Comp. Phys.* **22,** 501–516.

STETTER, H. J. 1973 *Analysis of Discretization Methods for Ordinary Differential Equations.* Berlin: Springer.

STRANG, G. 1968 On the construction and comparison of difference schemes. *SIAM J. Numer. Anal.* **5,** 506–517.

VERWER, J. G. 1984 Contractivity of locally one-dimensional splitting methods. *Numer. Math.* **44,** 247–259.

VERWER, J. G., & DEKKER, K. 1983 Step-by-step stability in the numerical solution of partial differential equations. *Report NW* 161–83, Centre for Math. and Comp. Sc., Amsterdam.

VERWER, J. G., & SANZ-SERNA, J. M. 1984 Convergence of method of lines approximations to partial differential equations. *Computing* **33,** 297–313.

WHITHAM, G. B. 1974 *Linear and Nonlinear Waves.* New York: Wiley, Interscience.

YANENKO, N. N. 1971 *The Method of Fractional Steps.* Berlin: Springer.

YUEN, H. C., & FERGUSON, W. E. 1978, Relationship between Benjamin-Feir instability and recurrence in the nonlinear Schroedinger equation. *Phys. Fluids* **21,** 1275–1278.

ZAKHAROV, V. E., & SHABAT, A. B. 1972 Exact theory of two dimensional self-focusing and one-dimensional self-modulation of waves in nonlinear media. *Sov. Phys. JETP* **34,** 62–69.

Appendix

Consider the system of ODEs

$$\mathbf{y}' = \mathbf{f}(\mathbf{y}), \qquad \mathbf{y}(0) = \boldsymbol{\eta}, \tag{A1}$$

where $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^n$, is such that for all $\mathbf{v} \in \mathbb{R}^n$

$$\mathbf{v}^T \mathbf{f}(\mathbf{v}) = 0, \tag{A2}$$

leading to the conservation $\|\mathbf{y}(t)\|_2 = \text{constant}$ for the solutions of (A1). Assume that a step of length τ of a second-order Runge–Kutta method is applied to (A1) yielding a vector $\mathbf{y}_1$. Then, it is well known that the (local) error $\mathbf{y}_1 - \mathbf{y}(\tau)$ has an expansion

$$\mathbf{y}_1 - \mathbf{y}(\tau) = \tau^3\{\beta \boldsymbol{J}\boldsymbol{J}\mathbf{f}(\boldsymbol{\eta}) + \gamma \boldsymbol{H}[\mathbf{f}(\boldsymbol{\eta}), \mathbf{f}(\boldsymbol{\eta})]\} + O(\tau^4), \tag{A3}$$

where β and γ depend only on the coefficients of the method, $\boldsymbol{J}$ is the Jacobian matrix of $\mathbf{f}$ evaluated at $\boldsymbol{\eta}$, and $\boldsymbol{H}(\mathbf{v}; \mathbf{v})$ denotes the second Fréchet derivative of $\mathbf{f}$, evaluated at $\boldsymbol{\eta}$ and acting on $\mathbf{v}$. Upon transferring $\mathbf{y}(\tau)$ to the r.h.s. and squaring both sides (3) becomes

$$\|\mathbf{y}_1\|^2 = \|\mathbf{y}(0)\|^2 + 2\tau^3\{\beta \boldsymbol{\eta}^T \boldsymbol{J}\boldsymbol{J}\mathbf{f}(\boldsymbol{\eta}) + \gamma \boldsymbol{\eta}^T \boldsymbol{H}[\mathbf{f}(\boldsymbol{\eta}), \mathbf{f}(\boldsymbol{\eta})]\} + O(\tau^4) \tag{A4}$$

where we have taken into account that $\|\mathbf{y}(\tau)\|^2 = \|\mathbf{y}(0)\|^2$ and that $\mathbf{y}(\tau) = \mathbf{y}(0) + O(\tau)$. Two differentiations of (A2) show, after some manipulation, that (A4) can be rewritten as

$$\|\mathbf{y}_1\|^2 - \|\mathbf{y}(0)\|^2 = -2\tau^3(\beta + 2\gamma)[\dot{\mathbf{y}}(0)^T \ddot{\mathbf{y}}(0)] + O(\tau^4).$$

We conclude that second order RK-methods yield an $O(\tau^3)$ increase in energy per step unless $\beta + 2\gamma = 0$.

Among the explicit, two-stage procedures, this relation is satisfied only for the method with array

$$\begin{array}{c|cc} 0 & & \\ 1 & 1 & 0 \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

(the improved Euler method in the terminology of Lambert, 1973 p. 119.) The system (4.2) is special, in that its solutions satisfy $\dot{\mathbf{y}}^T\ddot{\mathbf{y}} \equiv 0$. Thus the method (4.1) exhibits a $O(\tau^4)$ growth per step when applied to (4.2), but a larger $O(\tau^3)$ growth when applied to more general conservative problems.

Dekker & Verwer (1983) point out that for a general RK-method with array

$$\begin{array}{c|ccc} a_{11} & \cdots & a_{1s} \\ \vdots & & \vdots \\ a_{s1} & \cdots & a_{ss} \\ \hline b_1 & \cdots & b_s \end{array}$$

the increase in energy is given by

$$\|\mathbf{y}_1\|^2 - \|\mathbf{y}(0)\|^2 = -\sum_{i,j=1}^{s} m_{ij}\mathbf{k}_i^T\mathbf{k}_j, \qquad m_{ij} = b_i a_{ij} + b_j a_{ji} - b_i b_j$$

where the k_i are the 'slopes'

$$\mathbf{k}_i = \mathbf{f}\left(\eta + \tau \sum_{j=1}^{s} a_{ij}\mathbf{k}_j\right) \quad \text{for } i = 1\,(1)\,s.$$

Thus the method is conservative if and only if the matrix $\boldsymbol{M}$ with entries m_{ij} reduces to the null matrix. This condition is satisfied for Gaussian methods including the midpoint rule (Dekker & Verwer, 1984). More generally the conservation properties of the method depend only on $\boldsymbol{M}$.