A BIVARIATE SPLINE COLLOCATION SOLUTION OF THE KORTEWEG-DE VRIES EQUATION

HELMUT ROTH

A BIVARIATE SPLINE COLLOCATION SOLUTION OF THE KORTEWEG-DE VRIES EQUATION

by

© Helmut Roth

A thesis submitted to the School of Graduate Studies in partial fulfillment of the requirements for the degree of Master of Science

Department of Computer Science
Memorial University of Newfoundland

July 1994

St. John's Newfoundland

ISBN 0-612-06145-0

Canada

Abstract

The Korteweg-deVries (KdV) equation is solved numerically using bivariate spline collocation methods. Our methods permit one or two collocation points in time with an arbitrary number of collocation points in space. The basis functions for the underlying spline spaces are B-splines (in space) and Lagrange polynomials (in time). Numerical experiments show that collocation at two Gauss points in both space and time yields accurate solutions very efficiently. There is numerical evidence for fourth-order convergence in time (at the mesh points), but this is not proved. Using a suite of well known test problems, the methods are compared with the classical method of Zabusky and Kruskal, a convenient and frequently used reference standard for numerical KdV solvers.

Acknowledgements

I would like to take this opportunity to express my gratitude to everyone who has contributed to the completion of this thesis. As space permits me to give special mention to only a few, I hope that the many others do not feel overlooked.

Provision of the necessities of life was made possible by contributions from several sources. I would like to thank the Natural Sciences and Engineering Research Council, the Faculty of Graduate Studies, the Department of Computer Science, and Dr. H. Brunner for unfettered financial support. I would also like to thank Dr. G. Miminis for financial assistance during the summer of 1993, and Dr. R. Greatbatch for interesting part time employment throughout the duration of my graduate studies.

The staff of the Department of Computer Science have assisted in numerous ways. I would like to express special thanks to Elaine Boone and Pat Murphy for assistance with administrative matters, and to Dave Fifield and Nolan White for computer system support.

As a part time employee in the Department of Physics (Physical Oceanography), I often found it convenient to run thesis related computer programs during opportune moments in the workplace. In this regard, I would like to thank Dr. R. Greatbatch for his flexibility and for the use of his computing resources, and Allan Goulding for his system support.

For assistance with text formatting when time was in short supply, or when the text formatter was too mysterious, I offer my thanks to Dave Fifield, Sharon Hynes

and Dennis Wee.

At times, the completion of a thesis can seem impossible. I would like to thank my fellow graduate students and my supervisor for their encouragement and helpful discussions.

This thesis has benefitted considerably from the advice and suggestions of several reviewers. I would like to thank Dr. H. Brunner, Dr. P. Gillard, and especially the anonymous examiners for pointing out several errors in earlier drafts, and for their other helpful comments and criticisms. The responsibility for any remaining errors rests solely with me.

Mostly, however, I would like to sincerely thank my supervisor, Dr. H. Brunner, for his expert and patient assistance, his artful guidance, and his very agreeable style.

Contents

List of Tables

List of Figures

to Mary and Simon Evelio

Chapter 1

Introduction

This work is concerned with the numerical solution of the Korteweg-deVries (KdV) equation. We introduce the reader to this equation and to the remarkable solutions known as solitons by way of a brief historical sketch, relying primarily on the secondary sources cited. We then continue the introduction by showing how the equation can be derived for a physical system.

In Chapter 2 we present some analytic results for the KdV equation that will be useful for testing numerical methods. Chapter 3 gives a short overview of numerical methods for partial differential equations (PDEs) in general, and then surveys some methods that have been used for the KdV equation.

The heart of the present work is contained in Chapter 4. There we construct a numerical solution based on the method of collocation in both space and time. We then demonstrate the accuracy and efficiency of our methods in Chapter 5 and finish

with our conclusions and some directions for further work in Chapter 6.

1.1 Brief Historical Sketch

The first recorded observation of a solitary wave or soliton is undoubtedly that of J. Scott Russell in 1835. Though oft quoted, his description [78] of the event is an enthusiastic one, and I am pleased to include it here. He writes:

> I was observing the motion of a boat which was rapidly drawn along a narrow channel by a pair of horses, when the boat suddenly stopped—not so the mass of water in the channel which it had put in motion: it accumulated round the prow of the vessel in a state of violent agitation, then suddenly leaving it behind, rolled forward with great velocity, assuming the form of a large solitary elevation, a rounded, smooth, and well defined heap of water, which continued its course along the channel without change of form or diminution of speed. I followed it on horse back and overtook it rolling on at a rate of some eight or nine miles an hour, preserving its original figure some thirty feet long and a foot to a foot and a half in height. Its height gradually diminished, and after a chase of one or two miles I lost it in the windings of the channel. Such, in the month of August, 1834, was my first chance interview with that singular and beautiful phenomenon which I have called the Wave of Translation, a name which it now very generally bears.

Here then, according to Russell, was a newly observed phenomenon in search of a theory, for his wave had not yet been shown to be predicted by the equations of fluid mechanics. Such a theory was [78] "...still wanting, a worthy object for the enterprise of a future wave mathematician." The British mathemetician Airy, however, was "not disposed to recognize this wave as deserving the epithets 'great' or 'primary' ..." [4], and disagreed with Russell on the speed of the wave. Furthermore, Airy was of the opinion that " even when friction is neglected long waves in a rectangular canal must necessarily change their form as they advance, becoming steeper in front and less steep behind." [56, footnote, page 422]. Nevertheless Boussinesq in 1872 [10] was sufficiently enterprising to find a partial differential equation admitting a solitary wave solution, with speed of propagation agreeing with Russell, as was Rayleigh, independently, in 1876 [73].

By 1895 the controversy was still alive. Apparently Airy's views were still prominent [56, 14], and it was "the desire to settle this question definitively ..." that led Korteweg and deVries to " ...the somewhat tedious calculations at the end of our paper" [56]. Unlike Boussinesq, Korteweg and deVries begin by considering waves moving in one direction only, and end with a simpler equation. Miles [64] suggests that "the primary contributions of Korteweg and deVries, *vis-à-vis* Boussinesq, were in working directly with unidirectional waves, the simpler form of their evolution equation ...and their direct solution of that equation for both solitary and periodic waves."

In suitably scaled coordinates the KdV equation can be written as

$$u_t + uu_x + u_{xxx} = 0.$$

Kruskal [57] observes that this is "arguably the simplest partial differential equation not covered by classical methods, and by virtue of that property alone deserved attention ...". Nevertheless, it was not until it played a leading role in the birth of soliton theory (some seventy years after its derivation) that the equation obtained a high profile [57, 64, 67, 33, 31].

It was the Fermi-Pasta-Ulam problem that provided the setting. Around 1950, Fermi, Pasta, and Ulam [32] were investigating the evolution of a one dimensional lattice of equal masses with weak non-linear nearest-neighbour coupling. They were expecting that the energy from long wavelength initial conditions would become partioned equitably through the system, but found instead long time near-recurrences of the initial condition. Zabusky and Kruskal, further investigating this phenomena, found that the governing equation was equivalent to a discretised KdV equation [99]. With a numerical model [100], they found that their smooth initial condition soon separated into several well defined solitary waves which would interact with each other, *nonlinearly*, and emerge from the interaction intact, leaving only a small phase shift as evidence of the interaction. Hence in [99] Zabusky writes " The remarkable stability induced us to call them 'solitons' for they seemed to have an intrinsic identity".

There followed a number of analytic results, including an infinite hierarchy of conservation laws [68] and the *inverse scattering method* [39, 40] for the analytic solution of the initial value problem for the KdV equation. See the review article by Miura [67], which also contains some personal anecdotes on his contributions to the theory.

Meanwhile, the equation was being derived virtually everywhere that long wave phenomena existed. In connection with shallow water waves we find derivations for systems with fewer restrictions on geometry and flow [46, 84, 38, 53, 8, 54]. On a planetary scale, the KdV equation is found to model Rossby waves in the atmosphere and ocean [18, 74, 11]. New application areas included ion acoustic waves in plasmas [97] and hydromagnetic waves [41]. Other equations were also being found to have soliton solutions. Among these the nonlinear Schrödinger equation has been applied to solitons in optical fibres [48] and the sine-Gordon equation [26, page 14] has found application in nonlinear field theory. More examples can be found in such works as [37, 52, 67, 83, 26, 33, 58, 59, 61].

In chapter two below we will revisit some of the analytic results mentioned above, and in following chapters we discuss the numerical solution of the initial value problem, taking the Zabusky-Kruskal model [100] as our starting point. Our focus will be a solution by the method of collocation. The remainder of this chapter is devoted to deriving the Korteweg-deVries equation in a physical system.

1.2 The Korteweg-deVries Equation in a Physical System

We derive the Korteweg-deVries equation in a system similar (but neglecting surface tension) to that considered by Korteweg and deVries, namely plane gravity waves in shallow fluid in an infinitely long channel with flat bottom and uniform rectangular cross-section. The derivation is indeed somewhat long and tedious, but it is also instructive to relate the canonical form of the equation to an equation derived from physical considerations. The exposition here follows closely that of Dodd et al. [26], but see also those of Lamb [59] and Whitham [98]. After finding the non-dimensional equations of motion we expand the dependent variables in terms of a small perturbation parameter. Time and space are then rescaled, and the KdV equation results from consistency of kinematic and dynamic conditions on the free surface. This process has been formalized and called *reductive perturbation theory* by Taniuti and Wei [90].

Our lab coordinates will be x_1 along the channel, x_2 across the channel, and z measured vertically from the bottom. If h is the height of the undisturbed surface of the fluid and η is the departure of the surface from its undisturbed position, the free surface can be given by $z = h + \eta(\mathbf{x}, t)$, where $\mathbf{x} = (x_1, x_2)^T$. Our fluid will be inviscid and incompressible, undergoing irrotational flow. To further simplify matters, assume negligible surface tension and constant pressure at the surface.

If $\mathbf{u}$ denotes the velocity of the fluid, then for irrotational flow

$$\mathbf{curl}\,\mathbf{u} = 0 \tag{1.1}$$

which implies

$$\mathbf{u} = \mathbf{grad}\,\phi \tag{1.2}$$

for some velocity potential ϕ . Conservation of mass demands

$$\rho_t + \mathbf{div}\,(\rho\mathbf{u}) = 0.$$

For constant density, this reduces to $\mathbf{div}\,\mathbf{u} = 0$, which combined with equation 1.2 leads to Laplace's equation $\mathbf{div}\,\mathbf{grad}\,\phi = 0$, commonly written as

$$\nabla^2\phi = 0. \tag{1.3}$$

From Newton's second law we have

$$\rho\frac{D\mathbf{u}}{Dt} = -\mathbf{grad}\,p - \rho\mathbf{g}, \tag{1.4}$$

where p is the pressure in the fluid, $\mathbf{g}$ is the acceleration due to gravity, and the material derivative $D/Dt = \partial/\partial t + \mathbf{u}\cdot\mathbf{grad}$ indicates how the time rate of change of a field, following a particle of fluid, is due to both time variation of the field and the positional change of the particle within the the field. Thus equation 1.4 can be written

$$\frac{\partial\mathbf{u}}{\partial t} + (\mathbf{u}\cdot\mathbf{grad})\mathbf{u} = -\frac{1}{\rho}\mathbf{grad}\,p - \mathbf{g}. \tag{1.5}$$

Now use the vector identity

$$\mathbf{grad}\,(\mathbf{u}\cdot\mathbf{u}) = 2[\mathbf{u}\times\mathbf{curl}\,\mathbf{u} + (\mathbf{u}\cdot\mathbf{grad})\mathbf{u}]$$

to obtain

$$(\mathbf{u}\cdot\mathbf{grad})\mathbf{u} = (1/2)\mathbf{grad}(\mathbf{u}\cdot\mathbf{u})$$

when $\mathbf{curl}\,\mathbf{u} = 0$. Then, using (1.2), (1.5) becomes

$$\frac{\partial}{\partial t}(\mathbf{grad}\,\phi) + \frac{1}{2}\mathbf{grad}\,(\mathbf{grad}\,\phi\cdot\mathbf{grad}\,\phi) = -\frac{1}{\rho}\mathbf{grad}\,p - \mathbf{g}.$$

Now write $\mathbf{g} = \mathbf{grad}\,(gz)$ and recall that ρ is constant, **grad** is a linear operator, and $(\mathbf{grad}\,\phi)_t = \mathbf{grad}\,(\phi_t)$ to get

$$\mathbf{grad}\,\left(\phi_t + \frac{1}{2}\mathbf{grad}\,\phi\cdot\mathbf{grad}\,\phi + \frac{1}{\rho}p + gz\right) = 0$$

whence

$$\phi_t + \frac{1}{2}\mathbf{grad}\,\phi\cdot\mathbf{grad}\,\phi + \frac{1}{\rho}p + gz = A(t),$$

where A(t) is an arbitrary function of time only. It is convenient to take $A(t) = p_0/\rho + B(t)$, where p_0 is the pressure at the free surface (assumed constant), and incorporate $B(t)$ into ϕ via the mapping $\phi \mapsto \phi - \int B(t)dt$, so that

$$\phi_t + \frac{1}{2}\mathbf{grad}\,\phi\cdot\mathbf{grad}\,\phi + \frac{1}{\rho}(p - p_0) + gz = 0.$$

Then at the free surface $z = \eta(\mathbf{x}, t)$ we have

$$\phi_t + \frac{1}{2}\nabla\phi\cdot\nabla\phi + g\eta = 0. \tag{1.6}$$

Since the fluid must always be within its own boundaries, the normal fluid velocity at a boundary must be equal to the normal velocity of the boundary itself. The flat, impervious lower boundary is stationary, so at $z = 0$ we find

$$\phi_z = 0, \tag{1.7}$$

while applying the condition at the free surface leads to

$$\phi_{x_1}\eta_{x_1} + \phi_{x_2}\eta_{x_2} - \phi_z = -\eta_t.$$

Using the operator $\nabla_H = \partial/\partial x_1 + \partial/\partial x_2$, this can be written as

$$\phi_z = \eta_t + \nabla_H\phi \cdot \nabla_H\eta.$$

Gathering together the equations of motion and boundary conditions established above, we have

$$\nabla^2\phi = 0, \qquad 0 < z < h + \eta \tag{1.8}$$

$$\phi_t + \tfrac{1}{2}(\nabla\phi \cdot \nabla\phi) = -g\eta, \quad z = h + \eta \tag{1.9}$$

$$\phi_z = \eta_t + \nabla_H\phi \cdot \nabla_H\eta, \quad z = h + \eta \tag{1.10}$$

$$\phi_z = 0, \qquad z = 0. \tag{1.11}$$

To obtain the motion of the free surface η, we now solve Laplace's equation in the fluid, subject to the boundary conditions. To this end, first scale the variables to eliminate physical dimensions. Thus define the new length cordinates

$$\hat{z} = z/h \tag{1.12}$$

$$\hat{\mathbf{x}} = \mathbf{x}/l \tag{1.13}$$

$$\hat{\eta} = \eta/a, \tag{1.14}$$

where a is a typical wave amplitude.

We will use a linearized phase speed to determine an appropriate scaling for time. First linearize the free surface conditions (1.9) and (1.10):

$$\begin{aligned} \phi_t &= -g\eta \\ \phi_z &= \eta_t \end{aligned}$$

on the free surface. Note that $\phi_{tt} = -g\phi_z$. Now apply the linearized conditions on $z = h$:

$$\nabla^2\phi = 0, \quad 0 < z < h \tag{1.15}$$

$$\phi_{tt} = -g\phi_z, \quad z = h \tag{1.16}$$

$$\phi_z = 0, \quad z = 0. \tag{1.17}$$

Seek oscillatory solutions of the form $\phi = A(z)\exp[i(\mathbf{k}\cdot\mathbf{x} - \omega t)]$. From (1.15) we require $A'' - k^2A = 0$, where we have put $k^2 = \mathbf{k}\cdot\mathbf{k}$. Solving with (1.17) yields $A = \beta\cosh(kz)$, where β is an arbitrary constant, so the oscillatory solution is given by

$$\phi = \beta\cosh(kz)\exp[i(\mathbf{k}\cdot\mathbf{x} - \omega t)].$$

Now use (1.16) to obtain the linearized dispersion relation

$$\omega = \sqrt{gk\tanh(kh)} \tag{1.18}$$

and hence linearized phase speed

$$c = \frac{\omega}{k} = \sqrt{gh\frac{\tanh(kh)}{kh}}.$$

Note that

$$\lim_{kh\to 0} \frac{\tanh(kh)}{kh} = 1$$

so for small kh, that is for waves long compared to depth, the linearized phase speed $c \approx \sqrt{gh}$. At this speed a wave of length l has period $T = l/\sqrt{gh}$, so we can now adopt the scaled time

$$\hat{t} = t/T = t\sqrt{gh}/l. \tag{1.19}$$

To scale the velocity potential ϕ, use the above scalings in the dynamic free surface condition (1.9) to get

$$\tfrac{1}{T}\phi_{\hat{t}} + \tfrac{1}{2}\left[\tfrac{1}{l^2}\left(\phi^2_{\hat{x}_1} + \phi^2_{\hat{x}_2}\right) + \tfrac{1}{h^2}\phi^2_{\hat{z}}\right] + ag\hat{\eta} = 0, \quad \hat{z} = 1 + \frac{a}{h}\hat{\eta}.$$

On dividing by ag the coefficient of $\phi_{\hat{t}}$ is $1/(agT) = \sqrt{gh}/(agl) = c/(agl)$. This gives our final scaling

$$\hat{\phi} = \frac{c}{agl}\phi. \tag{1.20}$$

The transformed dimensionless equations are then

$$\hat{\phi}_{\hat{z}\hat{z}} + \delta^2\left(\hat{\phi}_{\hat{x}_1\hat{x}_1} + \hat{\phi}_{\hat{x}_2\hat{x}_2}\right) = 0, \qquad 0 < \hat{z} < 1 + \alpha\eta \tag{1.21}$$

$$\hat{\eta} + \hat{\phi}_{\hat{t}} + \tfrac{\alpha}{2}\left(\tfrac{1}{\delta^2}\hat{\phi}^2_{\hat{z}} + \hat{\phi}^2_{\hat{x}_1} + \hat{\phi}^2_{\hat{x}_2}\right) = 0, \quad \hat{z} = 1 + \alpha\eta \tag{1.22}$$

$$\hat{\phi}_{\hat{z}} = \delta^2\left[\hat{\eta}_{\hat{t}} + \alpha\left(\hat{\phi}_{\hat{x}_1}\hat{\eta}_{\hat{x}_1} + \hat{\phi}_{\hat{x}_2}\hat{\eta}_{\hat{x}_2}\right)\right], \quad \hat{z} = 1 + \alpha\eta \tag{1.23}$$

$$\hat{\phi}_{\hat{z}} = 0, \qquad \hat{z} = 0 \tag{1.24}$$

where $\alpha = a/h$, $\delta = h/l$.

We now simplify the notation by dropping the hats on the non-dimensionalized variables and proceed to seek a series solution of the form

$$\phi(\mathbf{x}, z, t) = \sum_{n=0}^{\infty} z^n \phi_n(\mathbf{x}, t). \tag{1.25}$$

Then (1.21) yields

$$\sum_{n=0}^{\infty} \left\{ n(n-1) z^{n-2} \phi_n + \delta^2 z^n \nabla_H^2 \phi_n \right\} = 0.$$

which can be rewritten as

$$\sum_{n=0}^{\infty} z^n \left\{ (n+2)(n+1)\phi_{n+2} + \delta^2 \nabla_H^2 \phi_n \right\} = 0$$

so that

$$\phi_{n+2} = -\frac{\delta^2}{(n+1)(n+2)} \nabla_H^2 \phi_n. \tag{1.26}$$

Meanwhile (1.24) yields, on $z = 0$

$$\sum_{n=0}^{\infty} \left\{ n z^{n-1} \phi_n(\mathbf{x}, t) \right\} = 0.$$

This can be rewritten as

$$\phi_1 + \sum_{n=2}^{\infty} \left\{ n z^{n-1} \phi_n \right\} = 0.$$

Since this holds on $z = 0$, the terms in the summation are obviously zero, so we have $\phi_1 = 0$ also. Then according to (1.26), $\phi_n = 0$ for all odd n. It is now a straightforward exercise to show that the recurrence (1.26) has the solution

$$\phi_{2m} = (-1)^m \frac{\delta^{2m}}{(2m)!} \nabla_H^{2m} \phi_0. \tag{1.27}$$

Thus

$$\begin{aligned}\phi(\mathbf{x}, z, t) &= \sum_{m=0}^{\infty}(-1)^m \frac{\delta^{2m}}{(2m)!} z^{2m} \nabla_H^{2m} \phi_0 \\ &= f - \frac{1}{2!} z^2 \delta^2 \nabla_H^2 f + \frac{1}{4!} z^4 \delta^4 \nabla_H^4 f - \ldots\end{aligned} \tag{1.28}$$

where we have put $f = \phi_0$. From (1.28) we obtain the expressions

$$\begin{aligned}\phi_z &= -z\delta^2 \nabla_H^2 f + \frac{z^3}{3!}\delta^4 \nabla_H^4 f + O(\delta^6) \\ \phi_t &= f_t - \frac{z^2}{2}\delta^2 \nabla_H^2 f_t + O(\delta^4)\end{aligned}$$

and observe that for any scalar function ψ

$$\nabla_H \phi \cdot \nabla_H \psi = \nabla_H f \cdot \nabla_H \psi - \frac{z^2}{2}\delta^2 \left[\nabla_H \left(\nabla_H^2 f\right) \cdot \nabla_H \psi\right] + O(\delta^4).$$

Using these expressions in the free surface conditions, (1.22) becomes

$$\begin{aligned}&\eta + f_t - \frac{z^2}{2}\delta^2 \nabla_H^2 f_t + \frac{\alpha}{2\delta^2}\left(-z\delta^2 \nabla_H^2 f\right)^2 \\ &\quad + \frac{\alpha}{2}\left[\nabla_H f \cdot \nabla_H f - z^2 \delta^2 \nabla_H \left(\nabla_H^2 f\right) \cdot \nabla_H f\right] + O(\delta^4) = 0,\end{aligned}$$

hence

$$\begin{aligned}&\eta + f_t + \frac{\alpha}{2}\nabla_H f \cdot \nabla_H f \\ &\quad -\frac{1}{2}(1+\alpha\eta)^2 \delta^2 \left[\nabla_H^2 f_t - \alpha\left(\nabla_H^2 f\right)^2 + \alpha \nabla_H \left(\nabla_H^2 f\right) \cdot \nabla_H f\right] \\ &\quad + O(\delta^4) = 0\end{aligned} \tag{1.29}$$

and (1.23), on dividing by δ^2, becomes

$$\begin{aligned}&-(1+\alpha\eta)\nabla_H^2 f + \frac{\delta^2}{6}(1+\alpha\eta)^3 \nabla_H^4 f \\ &\quad = \eta_t + \alpha\left[\nabla_H f \cdot \nabla_H \eta - \frac{1}{2}(1+\alpha\eta)^2 \delta^2 \nabla_H \left(\nabla_H^2 f\right) \cdot \nabla_H \eta\right] + O(\delta^4)\end{aligned}$$

or, on rearranging terms,

$$\eta_t + (1+\alpha\eta)\nabla_H^2 f + \alpha\nabla_H f \cdot \nabla_H \eta$$
$$-\frac{1}{2}(1+\alpha\eta)^2\delta^2\left[\frac{1}{3}(1+\alpha\eta)\nabla_H^4 f + \alpha\nabla_H\left(\nabla_H^2 f\right)\cdot\nabla_H\eta\right]$$
$$+O(\delta^4) = 0. \tag{1.30}$$

At this point it is convenient to recall that we set out to investigate plane waves in the channel, and for these there is no dependence on x_2. Consequently we now drop the x_2 coordinate, and for ease of notation write x for x_1. Then (1.29) and (1.30) become

$$\eta + f_t + \frac{\alpha}{2}f_x^2$$
$$-\frac{1}{2}(1+\alpha\eta)^2\delta^2\left(f_{xxt} - \alpha f_{xx}^2 + \alpha f_x f_{xxx}\right) + O(\delta^4) = 0 \tag{1.31}$$

$$\eta_t + \left[(1+\alpha\eta)f_x\right]_x$$
$$-\frac{1}{2}(1+\alpha\eta)^2\delta^2\left[\frac{1}{3}(1+\alpha\eta)f_{xxxx} + \alpha\eta_x f_{xxx}\right] + O(\delta^4) = 0. \tag{1.32}$$

Differentiating (1.31) with respect to x and then substituting $w = f_x$ leads to

$$\eta_x + w_t + \alpha w w_x$$
$$-\frac{1}{2}(1+\alpha\eta)^2\delta^2\left(w_{xxt} - \alpha w_x w_{xx} + \alpha w w_{xxx}\right)$$
$$-\alpha\eta_x(1+\alpha\eta)\delta^2\left(w_{xt} - \alpha w_x^2 + \alpha w w_{xx}\right) + O(\delta^4) = 0 \tag{1.33}$$

$$\eta_t + \left[(1+\alpha\eta)w\right]_x$$
$$-\frac{1}{2}(1+\alpha\eta)^2\delta^2\left[\frac{1}{3}(1+\alpha\eta)w_{xxx} + \alpha\eta_x w_{xx}\right] + O(\delta^4) = 0. \tag{1.34}$$

Now expand η and w in terms of a small perturbation parameter ϵ, so that

$$\eta = \eta^{(0)} + \epsilon\eta^{(1)} + \epsilon^2\eta^{(2)} + \dots$$

$$w = w^{(0)} + \epsilon w^{(1)} + \epsilon^2 w^{(2)} + \dots$$

Since η and w must both approach zero as $x \to \infty$, we must have $\eta^{(0)} = w^{(0)} = 0$. If we now make the coordinate transformation

$$\xi = \epsilon^P(x - at) \tag{1.35}$$

$$\tau = \epsilon^{3P} t, \tag{1.36}$$

where a, P are constants to be determined, we find

$$\frac{\partial}{\partial x} = \epsilon^P \frac{\partial}{\partial \xi}$$
$$\frac{\partial}{\partial t} = \epsilon^{3P} \frac{\partial}{\partial \tau} - a\epsilon^P \frac{\partial}{\partial \xi}$$

and so equations (1.33) and(1.34) transform to

$$\epsilon^{P+1}\left(\eta_\xi^{(1)} - aw_\xi^{(1)}\right) + \epsilon^{P+2}\left(\alpha w^{(1)} w_\xi^{(1)} + \eta_\xi^{(2)} - aw_\xi^{(2)}\right)$$
$$+\epsilon^{3P+1}\left(w_\tau^{(1)} + \frac{1}{2}\delta^2 a w_{\xi\xi\xi}^{(1)}\right) + \text{higher order terms} = 0 \tag{1.37}$$

$$\epsilon^{P+1}\left(-a\eta_\xi^{(1)} + w_\xi^{(1)}\right) + \epsilon^{P+2}\left(\alpha\eta^{(1)} w_\xi^{(1)} + \alpha\eta_\xi^{(1)} w^{(1)} - a\eta_\xi^{(2)} + w_\xi^{(2)}\right)$$
$$+\epsilon^{3P+1}\left(\eta_\tau^{(1)} - \frac{1}{6}\delta^2 w_{\xi\xi\xi}^{(1)}\right) + \text{higher order terms} = 0. \tag{1.38}$$

For these equations to hold, the coefficients of each power of ϵ must be zero. Consider first the coefficients of ϵ^{P+1}. We have

$$\eta_\xi^{(1)} - aw_\xi^{(1)} = 0$$

from (1.37), and

$$-a\eta_\xi^{(1)} + w_\xi^{(1)} = 0$$

from (1.38). These combine to yield

$$a = 1. \tag{1.39}$$

Observe that the coefficients of ϵ^{P+2} contain no τ derivatives, while those of ϵ^{3P+1} do. To avoid going to terms of order higher than 2, simply set $3P + 1 = P + 2$, thereby obtaining

$$P = \frac{1}{2}. \tag{1.40}$$

With these values for P and a, (1.37) and(1.38) become

$$\begin{aligned}
\epsilon^{3/2}\left(\eta_\xi^{(1)} - w_\xi^{(1)}\right) + \epsilon^{5/2}\left(w_\tau^{(1)} + \alpha w^{(1)} w_\xi^{(1)} + \frac{1}{2}\delta^2 w_{\xi\xi\xi}^{(1)} + \eta_\xi^{(2)} - w_\xi^{(2)}\right) &= 0 \\
\epsilon^{3/2}\left(-\eta_\xi^{(1)} + w_\xi^{(1)}\right) + \epsilon^{5/2}\left[\eta_\tau^{(1)} + \alpha\left(\eta^{(1)} w^{(1)}\right)_\xi - \frac{1}{6}\delta^2 w_{\xi\xi\xi}^{(1)} - \eta_\xi^{(2)} + w_\xi^{(2)}\right] &= 0.
\end{aligned}$$

Setting the term in $\epsilon^{3/2}$ to zero results in $\eta_\xi^{(1)} = w_\xi^{(1)}$. Finally, using $\eta^{(1)} = w^{(1)}$ and adding the terms in $\epsilon^{5/2}$ gives

$$\eta_\tau^{(1)} + \frac{3}{2}\alpha\eta^{(1)}\eta_\xi^{(1)} + \frac{1}{6}\delta^2\eta_{\xi\xi\xi}^{(1)} = 0 \tag{1.41}$$

which is the Korteweg-deVries equation for our physical system. Note that this is in a coordinate system moving to the right with velocity equal to our linearized phase velocity, for we have put $\xi = \epsilon^P(x - at)$ with $a = 1$, which corresponds to $c = \sqrt{gh}$ in the non-dimensional coordinate system.

By the further rescaling

$$u = \frac{3\alpha}{2\delta^2}\eta^{(1)}, \quad t = \frac{\delta^2}{6}\tau, \quad x = \xi$$

we obtain the standard form

$$u_t + 6uu_x + u_{xxx} = 0. \tag{1.42}$$

Chapter 2

The KdV Equation: Some Analytic Results

In this chapter we present some analytic results for the KdV equation, which will prove useful for testing the accuracy of numerical methods. We begin with a discussion of soliton solutions and then move on to conservation laws and some constants of the motion.

2.1 Soliton Solutions

Since a solitary wave, or single *soliton*, is a wave of permanent form (as observed by Russell), we seek a travelling wave solution to

$$u_t + 6uu_x + u_{xxx} = 0 \tag{2.1}$$

of the form

$$u(x,t) = f(\xi), \quad \xi = x - ct,$$

where c is the speed of the wave. Putting this into (2.1) yields

$$-cf' + 6ff' + f''' = 0.$$

Integrating, we get

$$-cf + 3f^2 + f'' = A, \tag{2.2}$$

where A is a constant of integration. For solitary waves, f, f', $f'' \to 0$ as $\xi \to \pm\infty$, so $A = 0$. Using f' as an integrating factor, a second integration then results in

$$-\frac{c}{2}f^2 + f^3 + \frac{1}{2}(f')^2 = B,$$

with B a second constant of integration. Again applying the condition f, $f' \to 0$ as $\xi \to \pm\infty$, we get $B = 0$ and we are left with

$$(f')^2 + f^2(2f - c) = 0.$$

Substituting $(c/2)\,\mathrm{sech}^2\theta$ for f, we obtain

$$\left(\frac{d\theta}{d\xi}\right)^2 + \frac{c}{4} = 0$$

whence

$$\theta = \pm\frac{\sqrt{c}}{2}(\xi - \xi_0) = \pm\frac{\sqrt{c}}{2}(x - x_0 - ct), \quad x_0 = \xi_0$$

and

$$f = \frac{c}{2}\,\mathrm{sech}^2\left[\frac{\sqrt{c}}{2}(x - x_0 - ct)\right]. \tag{2.3}$$

Observe that x_0 determines the position of the peak of the soliton at time $t = 0$, and the amplitude $c/2$ is proportional to the speed c. Also, the width of the soliton pulse is inversely proportional to the square root of the speed, so faster solitons are taller and narrower than slower ones.

Given this state of affairs it is natural to enquire into the evolution of an initial condition consisting of the superposition of two solitary waves, a taller wave to the left of, and well separated from, a shorter one. With sufficient initial separation, the profile near each pulse will not differ significantly from that of a solitary wave, so we may expect both pulses to evolve as solitary waves, at least temporarily. Eventually, however, the faster moving wave will encroach on the slower moving one, and they will undergo *nonlinear* interaction.

The surprising result of this interaction, first observed numerically by Zabusky and Kruskal [100] (but also observed experimentally by Russell [78]), is that the solitons emerge from the interaction without change of speed or form, suffering only a small displacement from the position they would have occupied had no interaction occurred. This two-soliton solution, and more generally the *N-soliton solution* of (2.1), has been determined analytically [95], and can be written as [98, page 583]

$$u = 2(\log D)_{xx} \tag{2.4}$$

where D is the determinant of a matrix with elements

$$D_{m,n} = \delta_{m,n} + 2\frac{\alpha_m}{\alpha_m + \alpha_n} e^{-\theta_m}.$$

Here $\delta_{m,n}$ is the Kronecker delta function, $\alpha_n^2 = c_n$ is the speed of the n-th soliton, $\theta_n = \alpha_n(x - x_n - \alpha_n^2 t)$, and the solitons are considered to be ordered from slowest to fastest so that $\alpha_1 < \alpha_2 < \cdots < \alpha_N$. The total phase shift of the n-th soliton, after interaction with the remaining $N-1$ solitons, is given by

$$\Delta_n = \frac{1}{\alpha_n}\left\{\sum_{m=1}^{n-1} \log\left(\frac{\alpha_n + \alpha_m}{\alpha_n - \alpha_m}\right)^2 - \sum_{m=n+1}^{N} \log\left(\frac{\alpha_m + \alpha_n}{\alpha_m - \alpha_n}\right)^2\right\}. \tag{2.5}$$

For $N = 1$, equation (2.4) yields

$$u = 2\frac{{\alpha_1}^2 e^{-\theta_1}}{(1 + e^{-\theta_1})^2} \tag{2.6}$$

which leads in a straightforward way to equation (2.3) for a solitary wave.

For N=2 the determinant in equation (2.4) is

$$D = \begin{vmatrix} 1 + e^{-\theta_1} & 2\frac{\alpha_1}{\alpha_2+\alpha_1} e^{-\theta_1} \\ 2\frac{\alpha_2}{\alpha_2+\alpha_1} e^{-\theta_2} & 1 + e^{-\theta_2} \end{vmatrix}$$

$$= 1 + e^{-\theta_1} + e^{-\theta_2} + \beta e^{-(\theta_1+\theta_2)},$$

where we have used

$$\beta = \left(\frac{\alpha_2 - \alpha_1}{\alpha_2 + \alpha_1}\right)^2.$$

From D we obtain

$$u = 2\frac{\alpha_1^2 e^{-\theta_1} + \alpha_2^2 e^{-\theta_2} + 2(\alpha_2 - \alpha_1)^2 e^{-(\theta_1+\theta_2)} + \beta(\alpha_2^2 e^{-\theta_1} + \alpha_1^2 e^{-\theta_2}) e^{-(\theta_1+\theta_2)}}{(1 + e^{-\theta_1} + e^{-\theta_2} + \beta e^{-(\theta_1+\theta_2)})^2}. \tag{2.7}$$

The phase shifts of equation (2.5) can be found from this solution with a little asymptotic analysis. First observe that θ_2 may be written

$$\theta_2 = \alpha_2\left[\frac{1}{\alpha_1}\theta_1 - (x_2 - x_1) - (\alpha_2^2 - \alpha_1^2)t\right].$$

Now keeping θ_1 fixed (so that we can follow a wave moving with speed α_1^2 to the right) and letting $t \to -\infty$, we find

$$u \simeq 2\frac{\alpha_1^2 e^{-\theta_1}}{(1+e^{-\theta_1})^2}$$

which, according to (2.6), represents a soliton with speed α_1^2 and position parameter x_1. On the other hand, letting $t \to +\infty$, still keeping θ_1 fixed, we find

$$u \simeq 2\frac{\alpha_1^2 \beta e^{-\theta_1}}{(1+\beta e^{-\theta_1})^2}$$

which can be written

$$u \simeq 2\frac{\alpha_1^2 e^{-\tilde{\theta}_1}}{(1+e^{-\tilde{\theta}_1})^2},$$

where

$$\tilde{\theta}_1 = \theta_1 + \log\left(\frac{1}{\beta}\right) = \theta_1 + \log\left(\frac{\alpha_2+\alpha_1}{\alpha_2-\alpha_1}\right)^2.$$

This last expression for u evidently represents a soliton with speed α_1^2 and position parameter

$$\tilde{x}_1 = x_1 - \frac{1}{\alpha_1}\log\left(\frac{\alpha_2+\alpha_1}{\alpha_2-\alpha_1}\right)^2.$$

Thus the slower soliton with speed α_1^2 undergoes a *backward* phase shift of magnitude $(1/\alpha_1)\log[(\alpha_2+\alpha_1)/(\alpha_2-\alpha_1)]^2$.

Similarly, keeping θ_2 fixed and letting $t \to -\infty$, we find

$$u \simeq 2\frac{\alpha_2^2 \beta e^{-\theta_2}}{(1+\beta e^{-\theta_2})^2} = 2\frac{\alpha_2^2 e^{-\tilde{\theta}_2}}{(1+e^{-\tilde{\theta}_2})^2},$$

where $\tilde{\theta}_2 = \theta_2 + \log[(\alpha_2+\alpha_1)/(\alpha_2-\alpha_1)]^2$, that is u is asymptotic to a soliton with speed α_2^2 and position parameter

$$\tilde{x}_2 = x_2 - \frac{1}{\alpha_2}\log\left(\frac{\alpha_2+\alpha_1}{\alpha_2-\alpha_1}\right)^2,$$

while as $t \to +\infty$, we find

$$u \simeq 2\frac{\alpha_2^2 e^{-\theta_2}}{(1+e^{-\theta_2})^2},$$

a soliton with speed α_2^2 and position parameter x_2. Thus the faster soliton with speed α_2^2 undergoes a *forward* phase shift of magnitude $(1/\alpha_2)\log[(\alpha_2+\alpha_1)/(\alpha_2-\alpha_1)]^2$.

A typical two soliton solution, on the domain $[-20,20]\times[0,2]$ is shown in figure 2.1. The preservation of soliton identity is striking, and the phase shifts are evident. The latter can be seen more clearly in the accompanying contour plot in figure 2.2.

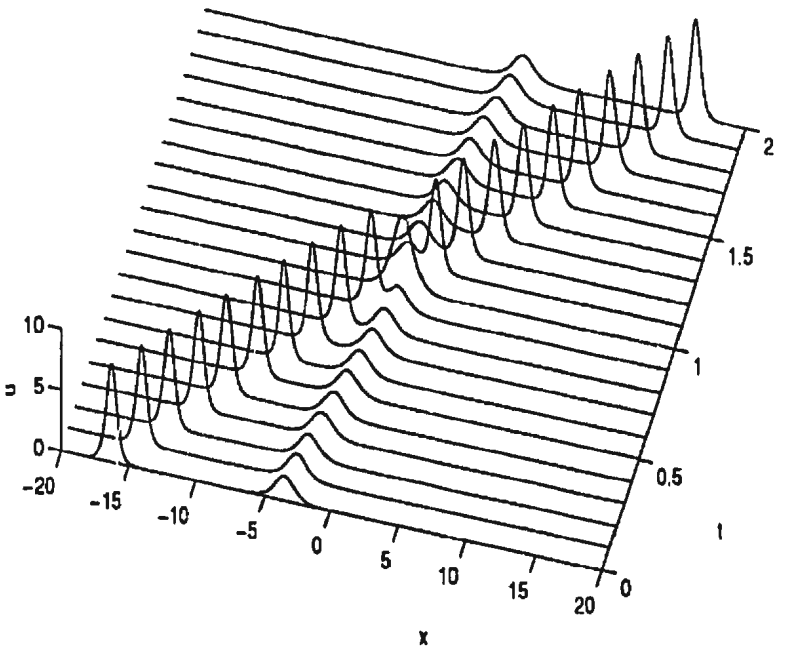

Figure 2.1: Analytic solution of Korteweg-deVries equation on $[-20,20]\times[0,2]$. Initial condition satisfies equation (2.7) with $c_1 = 4$, $c_2 = 16$, $x_1 = -c_1 - \Delta_1/2$, $x_2 = -c2 + \Delta_2/2$.

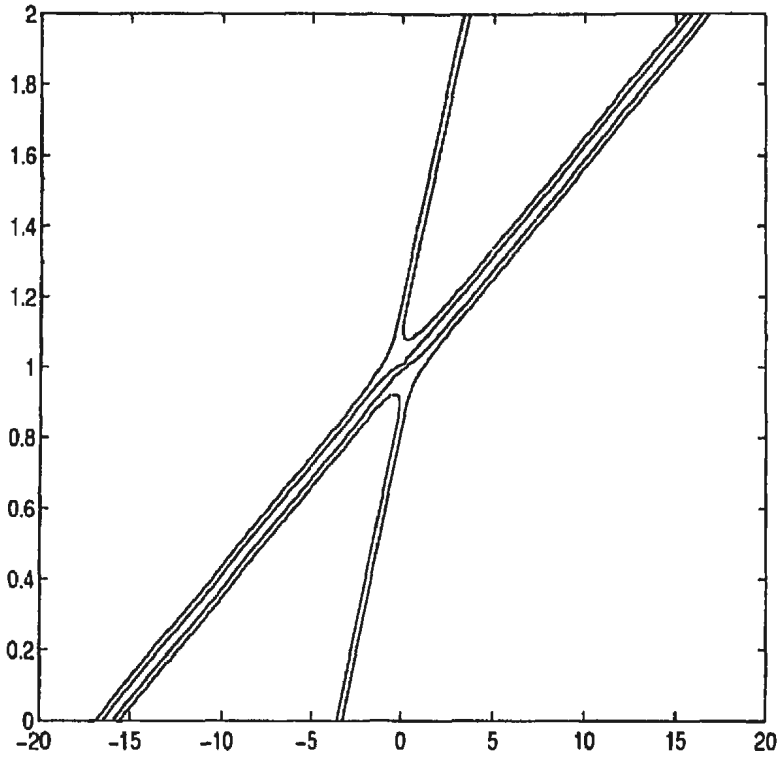

Figure 2.2: Contour plot showing soliton trajectories for analytic solution of Korteweg-deVries equation on $[-20, 20] \times [0, 2]$. Initial condition satisfies equation (2.7) with $c_1 = 4$, $c_2 = 16$, $x_1 = -c_1 - \Delta_1/2$, $x_2 = -c2 + \Delta_2/2$.

2.2 Conservation Laws

A conservation law is an equation of the form

$$T_t + X_x = 0 \tag{2.8}$$

where T, the conserved *density*, and X, the *flux*, are functions of x, t, $u(x,t)$, u_x, $u_{xx}, \ldots$. Integrating (2.8) over all x, we get

$$\frac{\partial}{\partial t} \int_{-\infty}^{\infty} T dx = [X]_{-\infty}^{\infty}$$

(assuming the integrals exist). If now $X \to 0$ as $x \to \pm\infty$, the term on the right vanishes and

$$\int_{-\infty}^{\infty} T dx = \text{constant.} \tag{2.9}$$

The integral in (2.9) is a *constant of the motion* for the evolution of $u(x,t)$.

The KdV equation (2.1) can be written in the conservation law form

$$u_t + (3u^2 + u_{xx})_x = 0$$

and so (at least for soliton solutions)

$$\int_{-\infty}^{\infty} u dx = \text{constant.} \tag{2.10}$$

A second conservation law can be obtained by multiplying the KdV equation (2.1) by u to get

$$uu_t + 6u^2 u_x + uu_{xxx} = 0$$

which can be written

$$\left(\frac{1}{2}u^2\right)_t + \left(2u^3 + uu_{xx} - \frac{1}{2}u_x^2\right)_x = 0$$

leading to

$$\int_{-\infty}^{\infty} u^2 dx = \text{constant.} \tag{2.11}$$

Our third conservation law for the KdV equation can be obtained by multiplying equation (2.1) by $3u^2$ and then subtracting the product of u_x and the x-derivative of (2.1). Thus

$$3u^2(u_t + 6uu_x + u_{xxx}) - u_x(u_t + 6uu_x + u_{xxx})_x = 0.$$

On adding $u_{xx}u_{xxx} - u_{xx}u_{xxx} = 0$ to the above equation, the result can be written

$$\left(u^3 - \frac{1}{2}u_x^2\right)_t + \left(\frac{9}{2}u^4 + 3u^2u_{xx} - 6uu_x^2 - u_xu_{xxx} + \frac{1}{2}u_{xx}^2\right)_x = 0$$

and so

$$\int_{-\infty}^{\infty}\left(u^3 - \frac{1}{2}u_x^2\right)dx = \text{constant}. \tag{2.12}$$

For water waves the three conservation laws 2.10, 2.11, 2.12 correspond to conservation of mass, momentum, and energy, respectively. Miura, Gardner and Kruskal [68] have shown that there are an infinite number of conservation laws for the KdV equation, but it appears that only the first three have physical significance.

Chapter 3

Numerical Methods

3.1 Introduction

Analytic solutions for the KdV equation provide a convenient reference standard for measuring the accuracy of numerical methods designed to solve its initial value problem. We can then reasonably expect that methods accurate for the KdV equation will also be accurate for other wave equations with similarly smooth solutions [24].

Our first problem in solving the KdV equation numerically is to restrict the computation to a finite spatial interval, say $[a, b]$. Then, given an initial condition on $[a, b]$ at time $t = 0$, we aim to find an approximation $U(x, t)$ to the analytic solution $u(x, t)$ for $x \in [a, b]$ and $t \in (0, T]$, for some future time T. The restriction of the space interval introduces boundaries at $x = a$ and $x = b$. In order to mimic the solution on the entire real line, we would like these to be invisible to the solution.

In particular, we don't want the solution on $[a, b]$ to be contaminated by reflections from the boundaries. A simple way to achieve this is to choose $[a, b]$ sufficiently large that the solution remains essentialy zero at the boundaries during the time interval of interest. (Recall that for soliton solutions, u and its space derivatives go to zero as $|x| \to \infty$). Other approaches include the use of periodic boundary conditions [100, 35] or absorbing boundary conditions. These latter may be designed to absorb incoming waves through dissipation or to match the solution at the boundary to that at the far field. Surveys and examples can be found in [9, 76, 77, 15, 92, 44, 55] and references therein. Moving grids that follow the solitons have also been used [80, 36].

Our next problem is obtaining the approximate solution $U(x, t)$. We briefly consider finite difference and spectral methods.

In finite difference methods the approximation is obtained by replacing the partial derivatives of u in the PDE by differences in the approximating function U, and then solving the resulting set of difference equations. Here U is a discrete function, and the solution to the system of difference equations yields the values of U at the mesh points where it is defined. The differences that approximate the derivatives are obtained from truncated Taylor's expansions. For example, the Taylor's formulae

$$u(x + h, t) = u(x, t) + h u_x(x, t) + \frac{h^2}{2!} u_{xx}(x, t) + \mathcal{O}\left(h^3\right) \tag{3.1}$$

$$u(x - h, t) = u(x, t) - h u_x(x, t) + \frac{h^2}{2!} u_{xx}(x, t) - \mathcal{O}\left(h^3\right) \tag{3.2}$$

combine to yield

$$u_x(x,t) = \frac{1}{2h}\left[u(x+h,t) - u(x-h,t)\right] + \mathcal{O}\left(h^2\right)$$

so that if U is defined on the set of spatial grid points

$$X_M = \{x_m : x_m = a + mh,\ m = 0,...,M,\ h = (b-a)/M\},$$

then

$$\frac{1}{2h}\left[U(x_{m+1},t) - U(x_{m-1},t)\right] \approx u_x(x_m,t) \tag{3.3}$$

with error $\mathcal{O}(h^2)$.

The approximation (3.3) is an example of a centred difference. One-sided differences are also available. Thus the series (3.1) yields the forward difference

$$u_x(x,t) = \frac{1}{h}\left[u(x+h,t) - u(x,t)\right] + \mathcal{O}(h) \tag{3.4}$$

while (3.2) gives the backward difference

$$u_x(x,t) = \frac{1}{h}\left[u(x,t) - u(x-h,t)\right] + \mathcal{O}(h)\,. \tag{3.5}$$

Differences for higher order derivatives can be found in a similar way. These will involve function values U at more than two grid points.

Similar differences are also available with respect to time. We discretize $[0,T]$ with the set of points

$$T_N = \{t_n : t_n = n\tau,\ n = 0,...,N,\ \tau = T/N\}$$

and then, given U on X_M at time t_n, we solve for U on X_M at t_{n+1}. Thus if we have an initial condition $U(x_m, 0)$, $m = 0, ..., M$, we can recursively solve for U on $X_M \times T_N$.

A complete finite difference scheme, then, consists of all derivatives in the given PDE replaced by the chosen differences. To illustrate, adopt the notation $U_m^{(n)} = U(x_m, t_n)$ and consider two schemes for the advection equation $u_t + cu_x = 0$:

$$\frac{1}{\tau}\left(U_m^{(n+1)} - U_m^{(n)}\right) = -\frac{c}{2h}\left(U_{m+1}^{(n)} - U_{m-1}^{(n)}\right) \tag{3.6}$$

and

$$\frac{1}{\tau}\left(U_m^{(n+1)} - U_m^{(n)}\right) = -\frac{c}{2h}\left(U_{m+1}^{(n+1)} - U_{m-1}^{(n+1)}\right) \tag{3.7}$$

These two schemes differ only in the time level at which the approximation to u_x is taken. Thus (3.6) yields an *explicit* expression for $U_m^{(n+1)}$ in terms of known values of U at time t_n, while (3.7) is *implicit*, combining three unknown values of U at time level $n+1$. The result is a system of simultaneous equations to be solved for U. Rearranging (3.7), we find the m-th equation in the system is given by

$$-rU_{m-1}^{(n+1)} + U_m^{(n+1)} + rU_{m+1}^{(n+1)} = U_m^{(n)}$$

where we have used $r = c\tau/(2h)$. The time stepping schemes of (3.6) and (3.7) are known respectively as *explicit Euler* and *implicit Euler*, regardless of the particular spatial discretization used.

More details on finite difference schemes and their properties can be found in works such as [6, 63, 75, 85]. Before leaving the subject, however, it seems fitting

to remark here that explicit schemes are subject to restrictions on the size of the timestep τ relative to the space step h. To see this in the present context of the advection equation, note that this equation has analytic solution $f(x-ct)$, which has constant values along the lines $x-ct=$ constant in the x,t plane. These lines, called characteristics, have slope c^{-1} and represent paths for the propagation of information [1]. Figure 3.1 shows the characteristic through the point (x_m, t_n). This characteristic also passes through the point $(\alpha, 0)$ and so knowledge of $u(\alpha, 0)$ is required for later knowledge of $u(x_m, t_n)$. But this knowledge is available to the explicit finite difference scheme (3.6) only if α is inside the interval $[x_{m-n}, x_{m+n}]$, that is only if the slope of the characteristic is at least as large as τ/h, or

$$\tau \leq h/c. \tag{3.8}$$

The interval $[x_{m-n}, x_{m+n}]$ is known as the numerical domain of dependence for $U_m^{(n)}$ with respect to the scheme (3.6). If the analytic domain of dependence is not completely included in the numerical domain of dependence (as in figure 3.1), required information is unavailable and there is consequently no bound on the possible error in $U_m^{(n)}$. Equation (3.8) is known variously as a stability criterion, or a CFL criterion after Courant, Friedrichs, and Lewy [19].

In spectral methods the solution is represented as

$$U(x,t) = \sum_{k=1}^{d} w_k(t)\phi_k(x) \tag{3.9}$$

where the $\phi_k,\ k=1,...,d$ are basis functions for a d-dimensional subspace of a normed

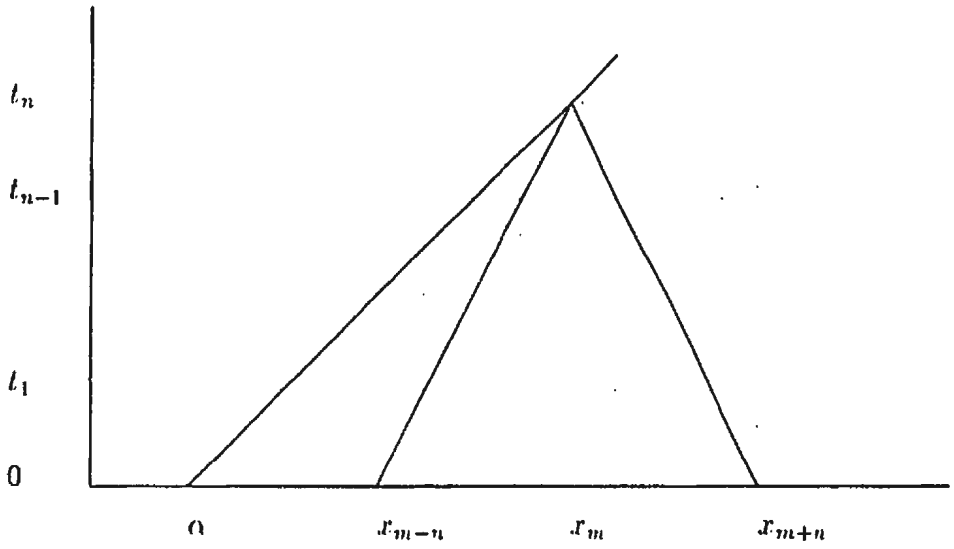

Figure 3.1: Domain of Dependence for $u(x_m, t_n)$. The characteristic through (x_m, t_n) passes through $(a, 0)$, but this point is outside the interval $[x_{m-n}, x_{m+n}]$ of information available to the numerical method. Consequently the error in U at (x_m, t_n) is not bounded.

linear space of functions, such as $L_2[a, b]$, the space of Lebesgue integrable functions on $[a, b]$. The basis functions may be a subset of a complete orthonormal set, as is the case when (3.9) represents a truncated Fourier series, or they might be a basis for a polynomial spline space, such as the B-splines to be introduced in chapter four.

In order to determine the coefficients $w_k(t)$ of the linear combination of basis functions in (3.9), the approximation is made to satisfy some condition that is also satisfied by the analytic solution. Let's consider equations of the form

$$u_t = \mathcal{L}u,$$

where $\mathcal{L}$ is a spatial differential operator. Clearly the KdV equation has this form, with $\mathcal{L}u = -6uu_x - u_{xxx}$.

The Galerkin approximation is obtained from the requirement

$$\frac{d}{dt}(U, \phi_k) = (\mathcal{L}U, \phi_k), \ k = 1, \ldots, d, \tag{3.10}$$

where $(w, v) = \int_a^b wv dx$. Petrov-Galerkin methods are similar, but allow the *test functions* to be different from the basis or *trial functions*:

$$\frac{d}{dt}(U, \psi_k) = (\mathcal{L}U, \psi_k), \ k = 1, \ldots, d. \tag{3.11}$$

A more general and thorough view of trial and test functions, as well as the Galerkin and Petrov-Galerkin approximations, is given by Morton [69]. See also Mitchell and Wait [66].

The collocation approximation is obtained from the requirement that the approximation satisfy the given PDE at some prescribed set of d points, say $\hat{X}$, so that

$$U_t = \mathcal{L}U \ \text{ for all } x \in \hat{X}. \tag{3.12}$$

Note that the collocation method does not require the evaluation of integrals, making it an attractive alternative to the Galerkin methods. Useful introductions to the method of collocation can be found in Brunner [12] or Prenter [72], while Gottlieb and Orszag [43] discuss a variety of spectral methods.

There are computational benefits when the basis functions have compact support, that is when they are identically zero outside a relatively short interval extending only a few gridpoints. In this situation the systems of equations for the coefficients in the linear combinations have a narrow band structure, thereby yielding considerable savings in computing resources.

Equations (3.10), (3.11), and (3.12) all represent systems of ODEs. These must be solved in order to complete the numerical method. The time integration can be implemented by finite differencing, and often is [35, 24, 43], but spectral methods [29] can also be used. In chapter four we will develop a method using collocation in both space and time.

Given an approximation $U(x,t)$ to the exact solution $u(x,t)$ of a partial differential equation, there are a number of important associated properties. Here we simply introduce the ideas and for more details refer the reader to the works of Ames [6], Gottlieb and Orszag [43], Hall and Porsching [47], Richtmyer and Morton [75], Smith [85], and references therein. It will be convenient to write the PDE in the form $\mathcal{L}u = 0$, where $\mathcal{L}$ is an evolutionary partial differential operator, and to represent the numerical method by $\tilde{\mathcal{L}}U = 0$.

The discretization error e at a point (x,t), or simply the error, is the difference between the exact solution and the numerical solution:

$$e(x,t) = u(x,t) - U(x,t).$$

If e goes to zero (at a fixed point) as the space and time steps h, τ go to zero, the numerical method is said to be convergent. Convergence may depend on h, τ obeying some relationship as they are made smaller, as we might expect with our earlier explicit finite difference scheme for the advection equation.

The local truncation error r is the residual obtained when the exact solution u is

substituted for the numerical solution U in the numerical method:

$$r(x,t) = \tilde{\mathcal{K}}u(x,t) \tag{3.13}$$

If r goes to zero as h, τ go to zero, the numerical method is said to be consistent. An estimate of the local truncation error r can often be found by expanding u in a Taylor's series in (3.13). If the method is convergent with a local truncation error $r = \mathcal{O}(h^{p+1}, \tau^{q+1})$, the method is said to be p-th order accurate in space and q-th order accurate in time.

Finally, a numerical method is said to be stable if $U(x_m, t_n)$ remains bounded as n increases. For consistent approximations to linear PDEs, stability is equivalent to convergence. Authors appear to agree that proving stability or convergence for nonlinear problems is often intractable, and so the analysis is confined to linearized versions of the equations. We will see several examples of this in the following survey.

3.2 Survey of Methods Used with the KdV equation

In this section we tour some of the methods that have been used to solve the KdV equation numerically. Our aim is not to be exhaustive, but to visit a representative sample of the variety of methods in use.

Historically, the earliest numerical scheme for the KdV equation was that of

Zabusky and Kruskal [100]. They used the finite difference scheme

$$U_m^{(n+1)} = U_m^{(n-1)} - 6 \cdot 2\tau \left(\frac{U_{m+1}^{(n)} + U_m^{(n)} + U_{m-1}^{(n)}}{3} \right) \left(\frac{U_{m+1}^{(n)} - U_{m-1}^{(n)}}{2h} \right) - 2\tau \left(\frac{U_{m+2}^{(n)} - 2U_{m+1}^{(n)} + 2U_{m-1}^{(n)} - U_{m-2}^{(n)}}{2h^3} \right)$$

which has a local truncation error $\mathcal{O}(h^2, \tau^2)$. The use of the three point average $(U_{m+1}^{(n)} + U_m^{(n)} + U_{m-1}^{(n)})/3$ to approximate $u(x_m, t_n)$ in the non-linear term results in better conservation of (discrete) momentum than does the use of the more simple approximation $U_m^{(n)}$. The scheme conserves discrete mass exactly provided that $U_0^{(n)} = U_{M-1}^{(n)}$ and $U_1^{(n)} = U_M^{(n)}$, as is the case for periodic boundary conditions or for these boundary values set to zero.

The time stepping derives from a centred difference and is known as a *leapfrog* scheme. It combines function values at the current and previous timesteps to generate the function values for the next time step. To get the method started, two initial conditions are required. This isn't physical and can give rise to a separation of the solution into the so-called physical and computational modes. Discussions of this phenomenon and its control can be found in [63, 35, 7]. Given the initial condition $U_m^{(0)}$, $m = 0, \ldots, M$, at $t = t_0$, the obvious way to get the solution at the next time step $t = t_1$ is with the explicit Euler step

$$U_m^{(1)} = U_m^{(0)} - 6\tau \left(\frac{U_{m+1}^{(0)} + U_m^{(0)} + U_{m-1}^{(0)}}{3} \right) \left(\frac{U_{m+1}^{(0)} - U_{m-1}^{(0)}}{2h} \right) - \tau \left(\frac{U_{m+2}^{(0)} - 2U_{m+1}^{(0)} + 2U_{m-1}^{(0)} - U_{m-2}^{(0)}}{2h^3} \right).$$

Vliegenthart [94] obtained the stability criterion

$$\tau \leq \frac{h^3}{4 + 6h^2 u_0}$$

for the linearized equation, where u_0 may be taken as the maximum magnitude of $u(x,t)$. The factor of h^3 is characteristic of explicit methods, and represents a severe restriction on the timestep. If, in the interests of accuracy for example, the space step is reduced from h to $\alpha^{-1}h$, for some $\alpha > 1$, the number of time steps required to integrate on $[0,T]$ increases from T/τ to $\alpha^3 T/\tau$. Then if $W(m)$ is the work required to compute the approximation to $6uu_x + u_{xxx}$ for m gridpoints, the total work required increases by the factor $\alpha^3 W(\alpha M)/W(M)$, where $M = (b-a)/h$ is the original number of mesh intervals. For the Zabusky-Kruskal method, W is a linear function, so the work increases as α^4. The consequences for cpu time can be seen in table 5.4.

Greig and Morris [45] used a Hopscotch method consisting of the equations

$$\begin{aligned} U_m^{(n+1)} &= U_m^{(n)} - 6\tau \left(\frac{U_{m+1}^{(n)} + U_{m-1}^{(n)}}{2} \right) \left(\frac{U_{m+1}^{(n)} - U_{m-1}^{(n)}}{2h} \right) \\ &\quad - \frac{\tau}{2h^3} \left(U_{m+2}^{(n)} - 2U_{m+1}^{(n)} + 2U_{m-1}^{(n)} - U_{m-2}^{(n)} \right) \qquad (3.14) \\ U_m^{(n+1)} &= U_m^{(n)} - 6\tau \left(\frac{U_{m+1}^{(n+1)} + U_{m-1}^{(n+1)}}{2} \right) \left(\frac{U_{m+1}^{(n+1)} - U_{m-1}^{(n+1)}}{2h} \right) \\ &\quad - \frac{\tau}{2h^3} \left(U_{m+2}^{(n+1)} - 2U_{m+1}^{(n+1)} + 2U_{m-1}^{(n+1)} - U_{m-2}^{(n+1)} \right) \qquad (3.15) \end{aligned}$$

Equation (3.14) is explicit and used for those gridpoints for which $m+n$ is even, while equation (3.15) is implicit and used for those gridpoints for which $m+n$ is odd.

Assume the solution is known at a time level for which n is even, as at the beginning of the integration when $n = 0$. Then (3.14) gives the solution at the next time level $n+1$ for points with $m = 0, 2, \cdots, M$; M assumed even. To obtain the solution at points with odd values of m, we rearrange (3.15) as

$$U_m^{(n+1)} + \frac{r}{2h^3}\left(U_{m+2}^{(n+1)} - U_{m-2}^{(n+1)}\right) = U_m^{(n)} - 6\tau\left(\frac{U_{m+1}^{(n+1)}+U_{m-1}^{(n+1)}}{2}\right)\left(\frac{U_{m+1}^{(n+1)}-U_{m-1}^{(n+1)}}{2}\right) + \frac{\tau}{h^3}\left(U_{m+1}^{(n+1)} - U_{m-1}^{(n+1)}\right).$$

All of the terms on the right are known. The solution for time level n has already been found and those terms at time level $n+1$ are all known from (3.14) since that equation is applied at points having $m = 1, 3, \cdots, M-1$. This yields the tridiagonal system

$$\begin{bmatrix} 1 & r & & & \\ -r & 1 & r & & \\ & \cdot & \cdot & \cdot & \\ & & -r & 1 & r \\ & & & -r & 1 \end{bmatrix} \begin{bmatrix} U_1^{(n+1)} \\ U_3^{(n+1)} \\ \\ U_{M-3}^{(n+1)} \\ U_{M-1}^{(n+1)} \end{bmatrix} = \begin{bmatrix} b_1^{(n)} \\ b_3^{(n)} \\ \\ b_{M-3}^{(n)} \\ b_{M-1}^{(n)} \end{bmatrix}$$

where $r = \tau/(2h^3)$ and $b_m^{(n)}$ is the right side of (3.2). Similarly, when n is odd, (3.14) gives the solution for odd m which are then used in the right side of a tridiagonal system for even m.

The scheme is found to have truncation error $\mathcal{O}(\tau^3, \tau h^2)$ under a linearized stability condition $\tau \leq h^3 / \mid 2 - u_{max} h^2 \mid$.

Fornberg and Whitham [35] combine a Fourier transform in space with a leapfrog time step. For convenience, the spatial domain $[a,b]$ is normalized to $[0,2\pi]$ and discretized by the $M+1$ equidistant points $\xi_m = m2\pi/M$, $m = 0,\cdots,M$; M assumed to be even. The KdV equation $u_t + 6uu_x + u_{xxx} = 0$ transforms to

$$v_t + 6\sigma v v_\xi + \sigma^3 v_{\xi\xi\xi} = 0$$

where $\sigma = 2\pi/(b-a)$, $\xi = \sigma(x-a)$, and $v(\xi,t) = u(x,t)$.

Now the vector $\mathbf{v}^{(n)}$ of function values $v(\xi_m, t_n)$, $m = 0,\cdots,M$, is transformed to the vector $\mathbf{w}$ in discrete Fourier space by

$$w(k,t_n) = \frac{1}{\sqrt{M}} \sum_{m=0}^{M-1} v_m^{(n)} e^{-imk2\pi/M},$$
$$k = 0, \pm 1, \cdots, \pm M/2$$

and the inverse transform $\mathbf{v} = \mathcal{F}^{-1}(\mathbf{w})$ is given by

$$v_m^{(n)} = \frac{1}{\sqrt{M}} \sum_{k=-M/2}^{M/2-1} w(k,t_n) e^{imk2\pi/M},$$
$$m = 0,1,\cdots,M.$$

Using the transforms, we have $\mathbf{v}_x = \mathcal{F}^{-1}\{ik\mathcal{F}(\mathbf{v})\}$ and $\mathbf{v}_{xxx} = \mathcal{F}^{-1}\{(ik)^3\mathcal{F}(\mathbf{v})\}$. With a leapfrog timestep and an aproximation $\mathbf{V}$ to $\mathbf{v}$, the numerical scheme to solve (3.2) is then

$$V_m^{n+1} = V_m^{(n-1)} - 2\tau 6\sigma V_m^{(n)} \mathcal{F}^{-1}\{ik\mathcal{F}(\mathbf{V^{(n)}})\} - 2\tau\sigma^3 \mathcal{F}^{-1}\{(ik)^3\mathcal{F}(\mathbf{V^{(n)}})\}.$$

Fornberg and Whitham then modify the last term and use

$$V_m^{(n+1)} = V_m^{(n-1)} - 12i\tau\sigma V_m^{(n)} \mathcal{F}^{-1}\{k\mathcal{F}(\mathbf{V^{(n)}})\} + 2i\mathcal{F}^{-1}\{sin(\sigma^3 k^3 \tau)\mathcal{F}(\mathbf{V^{(n)}})\}$$

in order to better preserve accuracy at high wave numbers.

The method has high spatial accuracy [34], requires 3 fast Fourier transforms per time step, and is subject to the stability criterion

$$\Delta t \leq \frac{3h^3}{2\pi^2}$$

where $h = (b-a)/M$.

Taha and Ablowitz [89] consider the inverse scattering transform to develop a "global" method which in turn leads them to the less complicated "local" method:

$$\begin{aligned} U_m^{(n+1)} &= U_m^{(n)} \\ &+ \tfrac{\tau}{2h^3}(U_{m-1}^{(n+1)} - 3U_m^{(n+1)} + 3U_{m+1}^{(n+1)} - U_{m+2}^{(n+1)}) \\ &+ \tfrac{\tau}{2h^3}(U_{m-2}^{(n)} - 3U_{m-1}^{(n)} + 3U_m^{(n)} - U_{m+1}^{(n)}) \\ &- \tfrac{3\tau}{2h}[(U_m^{(n)})^2 - (U_m^{(n+1)})^2] \\ &- \tfrac{\tau}{2h}[U_{m+1}^{(n+1)}(U_m^{(n+1)} + U_{m+1}^{(n+1)} + U_{m+2}^{(n+1)}) - U_{m-1}^{(n)}(U_m^{(n)} + U_{m-1}^{(n)} + U_{m-2}^{(n)})] \end{aligned}$$

with truncation error $O(h^2, \tau^2)$ [88]. They compare their methods with those of Zabusky and Kruskal (ZK), Greig and Morris (GM), Fornberg and Whitham (FW), and several others [42, 91, 88], finding their local scheme to be the fastest for a given accuracy requirement. On their most difficult test, a simulation of two solitons with amplitudes 0.5 and 2.5, interacting on $[-20, 20] \times [0, 2.4]$, and requiring a solution with error less than 0.02 in max norm, the relative computing times for the local, FW, GM, and ZK methods were 1, 1, 27.5, and 62.6 respectively.

See Nouri and Sloan [71] for a comparison of the Fornberg and Whitham scheme with several other Fourier pseudospectral methods using different time discretizations.

They use the same tests as Taha and Ablowitz, so all of the methods in the two studies are compared to the FW method.

Abe and Inoue [2] use a Fourier expansion method. They put

$$u(x,t) = \sum_{k=-\infty}^{\infty} a_k(t) e^{i\pi kx}$$

into the KdV equation to obtain

$$\frac{\partial}{\partial t}(a_k) = -6i\pi \sum_{m=-\infty}^{\infty} m a_{k-m} a_m + i\pi^3 k^3 a_k.$$

The sum is then truncated at $k = k_{\max}$, and the equation integrated by the (fouth-order) Runge-Kutta-Gill method. The linearized stability condition is

$$\tau \leq \frac{1}{\pi^3 k_{\max}^3}.$$

Note that the shortest wave (with largest wavenumber) that can be resolved on a grid with space step h has length $2h$, or wavenumber π/h. Putting this in the above, the stability condition becomes $\tau \leq h^3/\pi^6$. They obtain accuracy improvements on Zabusky-Kruskal and Greig-Morris in their numerical experiments.

Christov and Bekyarov [17] map the infinite interval into $[-1,1]$ via an algebraic function. They then use the complete orthonormal set of Chebyshev polynomials to represent the solution as

$$u(x,t) = \sum_{k=0}^{\infty} a_k(t) C_k(x)$$

and at each timestep t_n require it to satisfy (cf equation (2.2))

$$-cu + 3u^2 + u_{xx} = 0.$$

The resulting infinite system of (nonlinear) algebraic equations for the unknown coefficients $a_k(t_n)$ is then truncated to some finite number K.

For the case of the solitary wave, they obtain good agreement with analytically computed coefficients and with the soliton shape. Unfortunately, they do not report on phase error or multiple soliton interactions.

Alexander and Morris [5] use a Petrov-Galerkin finite element method proposed by Wahlbin [96]. The approximate solution $u(x,t)$ is represented as

$$U(x,t) = \sum_j a_j(t)\phi_j(x)$$

using smoothest cubic spline basis functions $\phi_j(x)$ having compact support on an interval of length $4h$. The solution is then required to satisfy

$$(U_t + UU_x + U_{xxx}, (\phi + qh^3\phi_{xxx})_j) = 0, \ \forall j$$

where h is the uniform space step, q is an arbitrary dissipation parameter and

$$(f,g) = \int_{-\infty}^{\infty} f(x)g(x)dx.$$

The resulting system of ODEs was solved using an IMSL library routine.

Their analysis for the linearized equation (which has u replaced by a constant in the quadratic term) shows an accuracy of order 6 (more generally, of order 2 p

for splines of degree p). The paper is concerned with the effect of the dissipation parameter and presents results for the case $q = 0$, but does not make comparison with other methods.

Sanz-Serna and Christie [79] use a Petrov-Galerkin method with different trial and test functions. The solution is approximated by

$$U(x,t) = \sum_i U_i(t)\phi_i(x).$$

The "trial" functions $\phi_i(x)$ are piecewise linear with compact support (the well known "hat" functions). The unknown functions $U_i(t)$ are determined from the system of ODEs:

$$(U_t, \psi_j) + 6(UU_x, \psi_j) + (U_x, (\psi_j)_{xx}) = 0,$$

where this last equation has been obtained by multiplying the KdV equation by the twice differentiable function $\psi_j(x)$ and integrating by parts. The "test" functions ψ_j are piecewise cubic with support on an interval of length $4h$.

The scheme is shown to be fourth order accurate in space and is integrated in time by the second order accurate Crank-Nicholson method (trapezoidal rule), using a Newton iteration to solve the resulting nonlinear system of algebraic equations. Numerical experiments with a solitary wave show improved results over Zabusky-Kruskal, Greig-Morris, and Alexander-Morris.

Mitchell and Schoombie [65] modify the method to permit the use of piecewise linear test functions for second order accuracy. Third order accuracy can be obtained

using quadratic splines.

Frutos and Sanz-Serna [24] present an implicit method with fourth order accuracy in time. If $F(\mathbf{U})$ is a spatial discretization of the KdV equation, where U is an N component vector of the approximate solution at the grid points, it is necessary to solve $\mathbf{U}_t = F(\mathbf{U})$. Finding $\mathbf{U}(t_{n+1}) = \mathbf{U}(t_n + \tau)$ from $\mathbf{U}(t_n)$ proceeds in 3 stages, each of which involes the solution of an N dimensional system of equations to compute auxilliary vectors that are meant to approximate the solution u at times $t_n + \beta_1\tau$, $t_n + (\beta_1 + \beta_2)\tau$, and $t_n + (\beta_1 + \beta_2 + \beta_3)\tau = t_n$, where $\beta_1 = (2 + 2^{\frac{1}{3}} + 2^{-\frac{1}{3}})/3$, $\beta_2 = 1 - 2\beta_1$, and $\beta_3 = \beta_1$. Note that $\beta_1 + \beta_2 + \beta_3 = 1$ and also $\beta_1^3 + \beta_2^3 + \beta_3^3 = 0$.

Using the spatial discretization of Sanz-Serna and Christie [79], a space step $h = 0.1$, a time step $\tau = 0.0125$, and an initial condition taken from the one soliton solution $u(x,t) = 2\mathrm{sech}^2(x - 4t)$ at $t = 0$, they integrate from $t = 0$ to $t = 2$ on the interval $[-20, 20]$, obtaining an error of 3.4×10^{-5} in max norm. The simulation required 480 matrix factorizations and the solution of 967 linear systems.

Chapter 4

Collocation Solution of the Korteweg-deVries Equation

We now turn to the development of a bivariate spline collocation approximation to the solution of the KdV equation. Our approach follows that of the well known Method of Lines, first discretizing in space to obtain a system of ODEs with time as the independent variable, and then discretizing in time to obtain a fully discrete numerical scheme. We begin the discussion with a review of some useful results from approximation theory.

4.1 Mathematical Background

Definition 4.1 (Polynomial Spline Space) *Let* $\Pi_M^{(x)} : a = x_0 < x_1 < \cdots < x_M = b$ *be a partition of the interval* $\Omega := [a, b]$, *with* $\Omega_m := [x_m, x_{m+1}]$, $h_m :=$

$x_{m+1} - x_m$, *and* $h := max\{h_m\}$, $m = 0, 1, \ldots, M-1$. *Denote by* π_p *the space of real polynomials with degree at most* p, *and let* $1 \leq \delta \leq p+1$. *Then*

$$S_p^{p-\delta}\left(\Pi_M^{(x)}\right) := \left\{v(x) : v \in C^{p-\delta}; v|_{\Omega_m} \in \pi_p\right\}$$

is the polynomial spline space consisting of polynomials of degree at most p *and having* $p - \delta$ *continuous derivatives on* Ω.

In general the $(p + 1 - \delta)$-th to p-th derivatives of elements of this space are discontinuous at the interior mesh points $x_1, \ldots, x_{M-1}$, while higher derivatives vanish everywhere on Ω (where they are defined). The integer δ, which determines the smoothness at the breakpoints, is known as the defect of the spline space.

Theorem 4.2 *The polynomial spline space* $S_p^{p-\delta}\left(\Pi_M^{(x)}\right)$ *is a linear space with dimension* d *given by*

$$d = M(p+1) - (M-1)(p-\delta+1) = M\delta + p - \delta + 1.$$

Proof: See Schumaker [82, page 110].

Having determined that $S_p^{p-\delta}\left(\Pi_M^{(x)}\right)$ is a linear space, it is natural to seek a set of basis functions.

Definition 4.3 (Extended Partition) *Let* $\Pi_M^{(x)} : a = x_0 < x_1 < \cdots < x_M = b$ *be a partition of* $[a, b]$. *Then the extended partition associated with the polynomial spline*

space $S_p^{p-\delta}\left(\Pi_M^{(x)}\right)$ *is given by*

$$\underline{\Pi_M^{(\bar{x})}} = \Big\{\bar{x}_k : \quad \bar{x}_1 = \cdots = \bar{x}_{p+1} = x_0 = a$$

$$\bar{x}_{p+2+(i-1)\delta} = \cdots = \bar{x}_{p+1+i\delta} = x_i,\ i = 1, \ldots, M-1$$

$$\bar{x}_{p+2+(M-1)\delta} = \cdots = \bar{x}_{2p+2+(M-1)\delta} = x_M = b\Big\}.$$

Before giving our next definition we introduce some convenient notation for divided differences and truncation functions. First, denote by $[\bar{x}_i, \ldots, \bar{x}_{i+m}]\, f$ the m-th divided difference of the function $f(\bar{x})$. Then, for the truncation function, let

$$(x - \bar{x})_+^0 = \begin{cases} 1, & x \geq \bar{x} \\ 0, & x < \bar{x} \end{cases}$$

and, for $p > 0$

$$(x - \bar{x})_+^p = \begin{cases} (x - \bar{x})^p, & x \geq \bar{x} \\ 0, & x < \bar{x}. \end{cases}$$

We can now define our basis functions.

Definition 4.4 (B-splines) *Let* $S_p^{p-\delta}\left(\Pi_M^{(x)}\right)$ *be a given polynomial spline space with associated extended partition* $\underline{\Pi_M^{(\bar{x})}}$. *Then for* $i = 1, 2, \ldots, d$, *the* i*-th B-spline of order* p *for the knot sequence* $\{\bar{x}\}$ *is defined by*

$$B_i^p(x) := (-1)^{p+1}(\bar{x}_{i+p+1} - \bar{x}_i)\,[\bar{x}_i, \ldots, \bar{x}_{i+p+1}]\,(x - \bar{x})_+^p,\ \textit{all } x \in \mathbf{R}.$$

Usually the degree p is easily inferred from context and we simply write B_i for B_i^p. Our next theorem establishes that the B-splines actually form a basis.

Theorem 4.5 (Curry and Schoenberg) *The set of B-splines* $\{B_i^p(x) : 1 \leq i \leq d\}$ *forms a basis for the polynomial spline space* $S_p^{p-\delta}\left(\Pi_M^{(x)}\right)$.

Proof: See Schumaker [82, page 116].

The next few theorems review some important properties of the B-splines associated with $S_p^{p-\delta}\left(\Pi_M^{(x)}\right)$.

Theorem 4.6 *The B-splines have compact support:* $B_i(x) = 0$ *for* $x \notin [\bar{x}_i, \bar{x}_{i+p+1}]$. *Hence only* $p+1$ *B-splines are non-zero on a particular interval* Ω_m.

Proof: See Schumaker [82, page 116] or deBoor [22, page 109].

Theorem 4.7 *The B-splines are positive on their support.*

Proof: See Schumaker [82, page 116] or deBoor [22, page 131].

Theorem 4.8 *The B-splines form a partition of unity, that is* $\sum_{i=1}^{d} B_i(x) = 1$.

Proof: See Schumaker [82, page 125].

Note that in view of the compact support, $\sum_{i=1}^{d} B_i(x) = \sum_{i=j-p}^{j} B_i(x)$ for $\bar{x}_j \leq x < \bar{x}_{j+1}$.

Theorem 4.9 *The B-splines and their derivatives can be evaluated in a stable way by means of a 3-term recurrence relation.*

Proof: See deBoor[22, chapter 10] and Cox [20].

Problem 4.10 (Interpolation) *Given a set of points* x_i, $i = 1, \ldots, d$ *and a set of function values* $f_i := f(x_i)$, $i = 1, \ldots, d$, *the interpolation problem is to find a spline*

function $s(x)$ from a suitable d-dimensional spline space S so that $s(x_i) = f(x_i)$, $i = 1, \ldots, d$.

If the spline space S has basis functions $\varphi_j(x)$, $j = 1, \ldots, d$, then the required spline can be written as a linear combination of these functions. Letting α_j, $j = 1, \ldots, d$ be the coefficients of the linear combination, we can write

$$s(x_i) = \sum_{j=1}^{d} \alpha_j \varphi_j(x_i), \ i = 1, \ldots, d$$

Determining $s(x)$ amounts to solving this linear system for the α_j. The system can be written in matrix notation as

$$\mathbf{Ma} = \sigma,$$

where $\sigma = [s(x_1), \ldots, s(x_d)]^T$, $\mathbf{a} = (\alpha_1, \ldots, \alpha_d)^T$, and the matrix $\mathbf{M} = [\varphi_j(x_i)]_{d \times d}$. Our next theorem tells us under what circumstances the matrix $\mathbf{M}$ is nonsingular when our spline s comes from the space $S_p^{p-\delta}\left(\Pi_M^{(x)}\right)$ and we work with the B-spline basis.

Theorem 4.11 (Schoenberg and Whitney) *Let $\hat{x}_1 < \cdots < \hat{x}_d$, and let $B_i(x)$, $i = 1, \ldots, d$ be the B-spline basis functions for the polynomial spline space $S_p^{p-\delta}\left(\Pi_M^{(x)}\right)$. Then*

$$\mathbf{M} := [B_j(\hat{x}_i)]_{d \times d}$$

is nonsingular if and only if

$$\hat{x}_i \in \{x : B_i(x) \neq 0\},$$

that is, if and only if

$$\bar{x}_i < \hat{x}_i < \bar{x}_{i+p+1}.$$

Proof: See Schumaker [82, page 167].

Problem 4.12 (Collocation) *Given a set of collocation points* $\hat{x}_i$, $i = 1, \ldots, \hat{d}$, *a function* $f(x)$, *an operator* $\mathcal{H}$, *and an equation* $\mathcal{H}u = f$ *for unknown function* $u(x)$, *the collocation problem consists of finding a function* $s(x)$ *from some suitable spline space* S *so that* $\mathcal{H}s = f$ *at the collocation points.*

Just as our previous theorem was useful for solving the interpolation problem, the following theorem will be useful for solving the collocation problem. In particular, it will guide our selection of the collocation points.

Theorem 4.13 (Karlin) *The matrix* $\mathbf{M} = \left[B_j^p(\hat{x}_i)\right]_{d\times d}$ *is totally positive, i.e. for any integers* $1 \le \nu_1 < \cdots < \nu_{\hat{d}} \le d$ *and any points* $\hat{x}_1 < \cdots < \hat{x}_{\hat{d}}$

$$det\left[B_{\nu_j}^p(\hat{x}_i)\right]_{\hat{d}\times\hat{d}} \ge 0.$$

Furthermore, strict positivity holds if and only if

$$\hat{x}_i \in \{x : B_i(x) \neq 0\},$$

that is, if and only if

$$\bar{x}_i < \hat{x}_i < \bar{x}_{i+p+1}.$$

Proof: See Schumaker [82, page 169].

We will see that the Gauss points of a partition satisfy the condition for strict positivity in theorem 4.13. To introduce these we need to look at quadrature formulae. For more details than can be given here, see Brunner and van der Houwen, [13], Davis and Rabinowitz [21], or Stroud [86].

A quadrature formula is a formula of the type

$$\int_a^b w(x)f(x)dx = \sum_{k=1}^{n} a_k f(x_k) + E(f) \tag{4.1}$$

for the approximate evaluation of definite integrals. The x_k, $k = 1, \ldots, n$ are called the nodes, points, or abscissae of the formula, the a_k are the coefficients or weights, and $E(f)$ is the error in the approximation. The function $w(x)$ is called the weight function. For our purposes it will be the constant function $w(x) = 1$. The formula (4.1) is said to have degree p if it is exact for all polynomials of degree $\leq p$, but not exact for some polynomial of degree $p + 1$.

The following sequence of theorems and definitions introduces Gauss quadrature and the Gauss points of a partition $\Pi_M^{(x)}$. It is assumed that on the interval of integration the weight function $w(x)$ is non-negative, and zero at no more than a finite number of points.

Theorem 4.14 *Given any n distinct points $x_1, \ldots, x_n$, we can find constants $a_1, \ldots, a_n$ such that the formula (4.1) is exact, i.e.*

$$\int_a^b w(x)f(x)dx = \sum_{k=1}^{n} a_k f(x_k)$$

whenever $f(x)$ is a polynomial of degree $\leq n-1$.

Proof: See Stroud [86, page 107].

Definition 4.15 (Orthogonal Polynomial) *An n-th degree polynomial $P_n(x)$ is said to be orthogonal on $[a,b]$, with respect to the weight function $w(x)$, to all polynomials of degree $\leq n-1$ if*

$$\int_a^b w(x)P_n(x)Q_{n-1}(x)dx = 0$$

for all polynomials $Q_{n-1}(x)$ of degree $\leq n-1$.

Theorem 4.16 *$P_n(x)$ always exists and is unique.*

Proof: See Stroud [86, page 126].

Theorem 4.17 *The zeros of $P_n(x)$ are real, distinct, and lie in the open interval (a,b).*

Proof: See Stroud [86, page 130].

Theorem 4.18 *If the points in formula (4.1) are the zeros of $P_n(x)$ and $a_1,\ldots,a_n$ are such that the formula has degree $n-1$, then it actualy has degree $2n-1$.*

Proof: See Stroud [86, page 135].

A formula of the type (4.1) for which theorem 4.18 holds is known as a Gauss quadrature formula.

For the weight function $w(x) = 1$ on the interval $[-1,1]$, $P_n(x)$ is the Legendre polynomial of degree n. We will call the zeros of this polynomial the Gauss points

Table 4.1: Gauss points and coefficients on $[-1,1]$.

n	$x_i,\ i=1,\ldots,n$			$c_i,\ i=1,\ldots,n$		
1	0			2		
2	$-\sqrt{3}/3$	$\sqrt{3}/3$		1	1	
3	$-\sqrt{3/5}$	0	$\sqrt{3/5}$	5/9	8/9	5/9

of order n on the standard interval $[-1,1]$. Table 4.1 shows the Gauss points and corresponding coefficients for the first three Gauss formulae on the standard interval. More extensive tables can be found in Stroud and Secrest [87] or in Abramowitz and Stegun [3].

The Gauss points on an arbitrary interval $\Omega_m = [x_m, x_{m+1}]$ are easily obtained from those on the standard interval via the mapping $x \mapsto x_m + h(1+x)/2$, where $h = x_{m+1} - x_m$. We have already spoken of the set of Gauss points of a partition $\Pi_M^{(x)}$. By this we mean the union over all subintervals of the Gauss points on those subintervals, that is

$$\{x_{m,k} : x_{m,k},\ k = 1,\ldots,n \text{ is a Gauss point on } \Omega_m,\ m = 0,1,\ldots,M-1\}.$$

The Gauss quadrature formulae use only points in the interior of the interval $[a,b]$. If the endpoint b is to be included in the abscissae, then the *Radau II* formulae may be used instead. The points for these formulae are also derived from the zeros of Legendre polynomials, and the formulae have degree of precision $2n-2$. See [13, page 60ff] for details. Table 4.2 shows the points and coefficients for the first three Radau II formulae. We will not have occasion to use the Radau I points, which

Table 4.2: Radau II points and coefficients on $[-1, 1]$.

n	$x_i,\ i = 1, \ldots, n$			$c_i,\ i = 1, \ldots, n$		
1	1			2		
2	$-1/3$	1		$3/2$	$1/2$	
3	$(-1-\sqrt{6})/5$	$(-1+\sqrt{6})/5$	0	$(16+\sqrt{6})/18$	$(16-\sqrt{6})/18$	$4/18$

include the left endpoint of the interval $[a, b]$ instead of the right.

4.2 Collocation in Space

We wish to solve the KdV equation in the form

$$u_t + 6uu_x + u_{xxx} = 0, \quad t > 0 \tag{4.2}$$

subject to the initial condition

$$u(x, 0) = \phi(x), \quad x \in \mathbf{R} \tag{4.3}$$

where $\phi^{(\eta)}(x) \to 0$ as $|x| \to \infty$, $\eta = 0, 1, \ldots$.

We seek $U(x, t) \approx u(x, t)$ on $[a, b] \times [0, T]$. The spatial interval $[a, b]$ is chosen sufficiently large so that during the time interval of interest [0,T], the solution and its derivatives remain negligibly small at a and b, that is

$$u^{(\eta)}(x, t) \to 0 \quad \text{as } x \to \left\{ \begin{array}{c} a^+ \\ \\ b^- \end{array} \right\}, \quad \eta = 0, 1, \ldots;\ t \in [0, T]. \tag{4.4}$$

We begin with a semi-discretization of the problem, discretizing in space and thereby obtaining a system of ODEs having time as the independent variable.

Let $\Pi_M^{(x)} : a = x_0 < x_1 < \cdots < x_M = b$ be a partition of the spatial domain $\Omega = [a, b]$. Set $\Omega_m := [x_m, x_{m+1}]$ and $h_m := x_{m+1} - x_m$, $m = 0, 1, \ldots, M-1$.

The approximation $U(x,t)$ will have the form

$$U(x,t) = \sum_{s=1}^{d} B_s(x) w_s(t) \tag{4.5}$$

where w_s, $s = 1, \ldots, d$, are functions to be determined in $C[0,T]$, and $B_s(x)$, $s = 1, \ldots, d$, are basis functions for the polynomial spline space

$$S_p^{p-\delta}\left(\Pi_M^{(x)}\right) := \left\{v(x) : v \in C^{p-\delta}; v|_{\Omega_m} \in \pi_p\right\}$$

of dimension $d = M\delta + (p + 1 - \delta)$.

Now introduce the collocation parameters $0 \leq c_1 < c_2 < \cdots < c_\delta \leq 1$. On each subinterval Ω_m, $m = 0, \ldots, M-1$, set $x_{m,i} := x_m + c_i h_m$, $i = 1, \ldots, \delta$. Relabel $x_{m,i}$ by $\hat{x}_{m\delta+i}$, $i = 1, \ldots, \delta$, $m = 0, \ldots, M-1$, and let $X_M = \{\hat{x}_r : 1 \leq r \leq M\delta\}$. We will require the approximation (4.5) to satisfy (4.2) on the set of lines $X_M \times [0,T]$. This is the (spatial) collocation condition, and the elements of X_M are known as the (spatial) collocation points. Note that we have δ collocation points in each subinterval.

The approximation (4.5) will also be required to satisfy the $p - \delta + 1$ boundary conditions

$$\frac{\partial^\eta U(x,t)}{\partial x^\eta} = 0 \text{ at } x = a, b; \text{ for } \eta = 0, 1, \ldots, \frac{p-\delta-1}{2}. \tag{4.6}$$

Note that this requires $p - \delta$ to be odd. Physical considerations suggest that U_{xxx} be at least piecewise continuous, so we also require $p - \delta \geq 3$.

When the approximation (4.5) is put into (4.2) and (4.4) we obtain a system of d ordinary differential equations:

$$\sum_{s=1}^{d}\left\{B_s(\hat{x}_r)w_s'(t)+6B_s(\hat{x}_r)w_s(t)\sum_{\nu=1}^{d}B_\nu'(\hat{x}_r)w_\nu(t)+B_s'''(\hat{x}_r)w_s(t)\right\}=0,$$

$$r=1,\ldots,M\delta \qquad (4.7)$$

$$\sum_{s=1}^{d}\frac{d^\eta B_s(a)}{dx^\eta}w_s(t)=0,\quad \eta=0,\ldots,\frac{p-\delta-1}{2} \qquad (4.8)$$

$$\sum_{s=1}^{d}\frac{d^\eta B_s(b)}{dx^\eta}w_s(t)=0,\quad \eta=0,\ldots,\frac{p-\delta-1}{2} \qquad (4.9)$$

with the initial condition

$$\sum_{s=1}^{d}B_s(\hat{x}_r)w_s(0)=\phi(\hat{x}_r),\quad r=1,\ldots,M\delta. \qquad (4.10)$$

The above system can be somewhat simplified by working in a suitable subspace of $S_p^{p-\delta}\left(\Pi_M^{(x)}\right)$ whose basis functions all satisfy the given boundary conditions. To this end, consider first the B-spline basis $\mathcal{B}:=\{B_i:1\le i\le d\}$ for $S_p^{p-\delta}\left(\Pi_M^{(x)}\right)$ with its associated extended partition

$$\underline{\Pi_M^{(x)}}=\Big\{\bar{x}_k:\quad \bar{x}_1=\cdots=\bar{x}_{p+1}=x_0=a$$

$$\bar{x}_{p+2+(i-1)\delta}=\cdots=\bar{x}_{p+1+i\delta}=x_i,\ i=1,\ldots,M-1$$

$$\bar{x}_{p+2+(M-1)\delta}=\cdots=\bar{x}_{2p+2+(M-1)\delta}=x_M=b\Big\}.$$

It is easy to show from the definitions of the B-splines that

$$B_i(x_0)=B_i(x_M)=0,\qquad i=2,\ldots,d-1$$

$$B_i'(x_0)=B_i'(x_M)=0,\qquad i=3,\ldots,d-2$$

$$\vdots$$

$$B_i^{(\eta)}(x_0)=B_i^{(\eta)}(x_M)=0,\quad i=\eta+2,\ldots,d-1-\eta;\ \eta=0,\ldots,\tfrac{p-\delta-1}{2}$$

while on the other hand

$$B_i^{(\eta)}(x_0) \neq 0 \text{ and } B_{d+1-i}^{(\eta)}(x_M) \neq 0,\ i = 1,\ldots,\eta+1;\ \eta = 0,\ldots,\frac{p-\delta-1}{2}.$$

Thus the set of functions formed by removing the first and last $(p-\delta+1)/2$ B-splines from the above basis $\mathcal{B}$ will satisfy the given boundary conditions. The span of this reduced basis is the subspace of $S_p^{p-\delta}\left(\Pi_M^{(x)}\right)$ defined by

$$\tilde{S}_p^{p-\delta}\left(\Pi_M^{(x)}\right) = \left\{ v(x):\ v \in S_p^{p-\delta}\left(\Pi_M^{(x)}\right) \text{ and } v^{(\eta)}(x_0) = v^{(\eta)}(x_M) = 0,\ \eta = 0,\ldots,\tfrac{p-\delta-1}{2} \right\}$$

which has dimension reduced from d to $\tilde{d} = d-(p-\delta+1) = M\delta$. Now by renaming according to

$$\tilde{B}_i = B_{i+(p+1-\delta)/2},\ i = 1,\ldots,\tilde{d}$$

our system of o.d.e.'s becomes

$$\sum_{s=1}^{\tilde{d}} \left\{ \tilde{B}_s(\hat{x}_r)\tilde{w}_s'(t) + 6\tilde{B}_s(\hat{x}_r)\tilde{w}_s(t)\sum_{\nu=1}^{\tilde{d}} \tilde{B}_\nu'(\hat{x}_r)\tilde{w}_\nu(t) + \tilde{B}_s'''(\hat{x}_r)\tilde{w}_s(t) \right\} = 0,$$
$$r = 1,\ldots,M\delta \qquad (4.11)$$

where $\tilde{w}_s = w_{s+(p+1-\delta)/2},\ i = 1,\ldots,\tilde{d}$, and primes indicate ordinary derivatives.

It is useful (and somewhat tidier) to write the system (4.11) in matrix notation. Let $\mathrm{w} = (\tilde{w}_1,\ldots,\tilde{w}_{\tilde{d}})^T$, $\mathbf{B} = (\beta_{i,j})_{\tilde{d}\times\tilde{d}}$ where $\beta_{i,j} = \tilde{B}_j(\hat{x}_i)$, and denote by $\otimes$ the elementwise multiplication of vectors, so that $\mathrm{x}\otimes\mathrm{y} = (x_1y_1,\ldots,x_{\tilde{d}}y_{\tilde{d}})^T$. Then, using the convenient subscript notation of partial derivatives to represent ordinary derivatives, (4.11) becomes

$$\mathbf{B}\mathrm{w}_t + 6(\mathbf{B}\mathrm{w})\otimes(\mathbf{B}_x\mathrm{w}) + \mathbf{B}_{xxx}\mathrm{w} = 0. \qquad (4.12)$$

The initial condition (4.4) becomes

$$\mathbf{Bw} = \mathbf{\Phi} \tag{4.13}$$

where $\Phi = (\phi(\hat{x}_1), \ldots, \phi(\hat{x}_{M\delta}))^T$ is the restriction of $\phi(x)$ to the set of collocation points $X_{M\delta}$.

We now have our semidiscretization of the KdV initial value problem in equations (4.12) and (4.13). Our next job is to discretize time and so complete the numerical method.

4.3 Collocation in Time

Our approximation $U(x,t)$ has the form

$$U(x,t) = \sum_{s=1}^{\tilde{d}} \tilde{B}_s(x) w_s(t) \tag{4.14}$$

but now, to complete the discretization of (4.12), our functions $w_s(t)$, $s = 1, \ldots, \tilde{d}$ will be elements of a polynomial spline space, rather than the continuous functions of the previous section. In particular, we will consider the cases where $\mathrm{w} = (\tilde{w}_1, \ldots, \tilde{w}_{\tilde{d}})^T$ is piecewise linear ($w_i \in S_1^0\left(\Pi_N^{(t)}\right)$) and piecewise quadratic ($w_i \in S_2^0\left(\Pi_N^{(t)}\right)$). Here $\Pi_N^{(t)} : 0 = t_0 < \cdots < t_N = T$ is a partition of $[0,T]$, and it will be convenient to have $\tau_n := t_{n+1} - t_n$, $n = 0, \ldots, N-1$.

4.3.1 Piecewise Linear Interpolant

Represent $\mathrm{w}(t)$ on $[t_n, t_{n+1}]$ by

$$\mathrm{w}(t) = \mathrm{a}^{(n)}\mathcal{L}_0^{(n)}(t) + \mathrm{b}^{(n)}\mathcal{L}_1^{(n)}(t) \tag{4.15}$$

where $\mathrm{a}^{(n)}$ and $\mathrm{b}^{(n)}$ are vectors to be determined in $\mathbf{R}^{\tilde{\mathrm{d}}}$, and $\mathcal{L}_0^{(n)}(t)$, $\mathcal{L}_1^{(n)}(t)$ are the fundamental linear Lagrange polynomials

$$\mathcal{L}_0^{(n)}(t) = \begin{cases} \frac{t_{n+1}-t}{t_{n+1}-t_n}, & t \in [t_n, t_{n+1}] \\ 0, & \text{elsewhere} \end{cases}$$

$$\mathcal{L}_1^{(n)}(t) = \begin{cases} \frac{t-t_n}{t_{n+1}-t_n}, & t \in [t_n, t_{n+1}] \\ 0, & \text{elsewhere.} \end{cases}$$

These functions form a local basis (equivalent to the B-spline basis) on the interval $[t_n, t_{n+1}]$ for the polynomial spline space $S_1^0\left(\Pi_N^{(t)}\right)$. Then on (t_n, t_{n+1}) we have

$$\begin{aligned} \frac{d\mathrm{w}}{dt} &= \mathrm{a}^{(n)}\frac{d}{dt}\mathcal{L}_0^{(n)}(t) + \mathrm{b}^{(n)}\frac{d}{dt}\mathcal{L}_1^{(n)}(t) \\ &= -\frac{1}{\tau_n}\mathrm{a}^{(n)} + \frac{1}{\tau_n}\mathrm{b}^{(n)}. \end{aligned}$$

Introduce the set of time collocation points

$$T_N := \left\{\hat{t}_n := t_n + \gamma_1\tau_n : \gamma_1 \in [0,1],\ 0 \le n \le N-1\right\}$$

and, using (4.12), collocate at $\hat{t}_n$ to obtain

$$\begin{aligned} &\mathrm{B}\left(-\frac{1}{\tau_n}\mathrm{a}^{(n)} + \frac{1}{\tau_n}\mathrm{b}^{(n)}\right) \\ &\quad +6\left\{\mathrm{B}\left[\mathrm{a}^{(n)}\mathcal{L}_0^{(n)}\left(\hat{t}_n\right) + \mathrm{b}^{(n)}\mathcal{L}_1^{(n)}\left(\hat{t}_n\right)\right]\right\} \otimes \left\{\mathrm{B}_x\left[\mathrm{a}^{(n)}\mathcal{L}_0^{(n)}\left(\hat{t}_n\right) + \mathrm{b}^{(n)}\mathcal{L}_1^{(n)}\left(\hat{t}_n\right)\right]\right\} \\ &\quad +\mathrm{B}_{xxx}\left[\mathrm{a}^{(n)}\mathcal{L}_0^{(n)}\left(\hat{t}_n\right) + \mathrm{b}^{(n)}\mathcal{L}_1^{(n)}\left(\hat{t}_n\right)\right] = 0. \end{aligned}$$

Using

$$\mathcal{L}_0^{(n)}\left(\hat{t}_n\right) = \mathcal{L}_0^{(n)}(t_n + \gamma_1\tau_n) = 1 - \gamma_1$$

$$\mathcal{L}_1^{(n)}\left(\hat{t}_n\right) = \mathcal{L}_1^{(n)}(t_n + \gamma_1\tau_n) = \gamma_1$$

the collocation equation becomes

$$\begin{aligned}
&\frac{1}{\tau_n}\mathbf{B}\left(\mathrm{b}^{(n)} - \mathrm{a}^{(n)}\right) \\
&\quad +6\left\{\mathbf{B}\left[(1-\gamma_1)\mathbf{a}^{(n)} + \gamma_1\mathbf{b}^{(n)}\right]\right\} \otimes \left\{\mathbf{B}_x\left[(1-\gamma_1)\mathrm{a}^{(n)} + \gamma_1\mathrm{b}^{(n)}\right]\right\} \\
&\quad +\mathbf{B}_{xxx}\left[(1-\gamma_1)\mathbf{a}^{(n)} + \gamma_1\mathbf{b}^{(n)}\right] = 0.
\end{aligned} \tag{4.16}$$

Note that on $[t_n, t_{n+1}]$

$$\begin{aligned}
\mathbf{w}(t_{n+1}) &= \mathrm{a}^{(n)}\mathcal{L}_0^{(n)}(t_{n+1}) + \mathrm{b}^{(n)}\mathcal{L}_1^{(n)}(t_{n+1}) \\
&= \mathrm{a}^{(n)}(0) + \mathrm{b}^{(n)}(1) \\
&= \mathrm{b}^{(n)}
\end{aligned}$$

while on $[t_{n+1}, t_{n+2}]$

$$\begin{aligned}
\mathrm{w}(t_{n+1}) &= \mathbf{a}^{(n+1)}\mathcal{L}_0^{n+1}(t_{n+1}) + \mathrm{b}^{(n+1)}\mathcal{L}_1^{n+1}(t_{n+1}) \\
&= \mathbf{a}^{(n+1)}(1) + \mathrm{b}^{(n+1)}(0) \\
&= \mathbf{a}^{(n+1)}
\end{aligned}$$

which gives the relation

$$\mathrm{a}^{(n+1)} = \mathbf{b}^{(n)} \tag{4.17}$$

Meanwhile, the initial condition $\mathbf{Bw} = \Phi$ yields

$$\mathbf{B}\left[\mathbf{a}^{(0)}\mathcal{L}_{(0)}^{0}(t_0) + \mathbf{b}^{(0)}\mathcal{L}_1^{(0)}(t_0)\right] = \mathbf{B}\mathbf{a}^{(0)} = \Phi$$

or

$$\mathbf{a}^{(0)} = \mathbf{B}^{-1}\Phi \tag{4.18}$$

Assuming that $\mathbf{B}^{-1}$ exists, equations (4.16), (4.17), and (4.18) provide the means for recursively computing $U(x,t)$ on $X_M \times T_N$:

Algorithm 4.19

1. *Solve (4.18) for* $\mathbf{a}^{(0)}$.

2. *Given* $\mathbf{a}^{(n)}$:

   (a) *Solve (4.16) for* $\mathbf{b}^{(n)}$. *Then (4.15) gives* $\mathbf{w}(t)$ *on* $[t_n, t_{n+1}]$.

   (b) *Use (4.14) to obtain* $U(x,t)$ *anywhere on* $\Omega \times [t_n, t_{n+1}]$.

   (c) *Use (4.17) to obtain* $\mathbf{a}^{(n+1)}$ *from* $\mathbf{b}^{(n)}$.

4.3.2 Piecewise Quadratic Interpolant

If $\mathrm{w}(t)$ is a piecewise quadratic function, then $\mathrm{w}_t := d\mathrm{w}/dt$ is piecewise linear. Consequently we can represent w_t on $[t_n, t_{n+1}]$ in a now familiar way with the fundamental linear Lagrange functions, that is

$$\begin{aligned} \mathbf{w}_t(t) &= \mathbf{a}^{(n)}\mathcal{L}_0^{(n)}(t) + \mathbf{b}^{(n)}\mathcal{L}_1^{(n)}(t) \\ &= \frac{\mathbf{a}^{(n)}}{\tau_n}(t_{n+1} - t) + \frac{\mathbf{b}^{(n)}}{\tau_n}(t - t_n) \end{aligned} \tag{4.19}$$

For $t \in [t_n, t_{n+1}]$ we can write $t = t_n + s\tau_n$ for some $s \in [0,1]$. Putting this value in (4.19) results in

$$\mathbf{w}_t(t_n + s\tau_n) = (1-s)\mathbf{a}^{(n)} + s\mathbf{b}^{(n)} \tag{4.20}$$

which, together with

$$\mathbf{w}(t_n + s\tau_n) = \mathbf{w}(t_n) + \int_{t_n}^{t_n + s\tau_n} \mathbf{w}'(z)dz$$

gives

$$\mathbf{w}(t_n + s\tau_n) = \mathbf{w}(t_n) + \frac{1}{\tau_n}\int_{t_n}^{t_n+s\tau_n} \left[\mathbf{a}^{(n)}(t_{n+1} - z) + \mathbf{b}^{(n)}(z - t_n)\right] dz.$$

This evaluates to

$$\mathbf{w}(t_n + s\tau_n) = \mathbf{w}(t_n) + s\tau_n \left[(1 - s/2)\mathbf{a}^{(n)} + (s/2)\mathbf{b}^{(n)}\right]. \tag{4.21}$$

Introducing the collocation parameters $0 \leq \gamma_1 < \gamma_2 \leq 1$ and collocating on $[t_n, t_{n+1}]$ at

$$\hat{t}_{n,s} := t_n + s\tau_n, \ s = \gamma_1, \gamma_2$$

leads from (4.12) to the collocation equations

$$\mathbf{B}\mathbf{w}_t(\hat{t}_{n,s}) + 6\left\{\left[\mathbf{B}\mathbf{w}(\hat{t}_{n,s})\right] \otimes \left[\mathbf{B}_x\mathbf{w}(\hat{t}_{n,s})\right]\right\} + \mathbf{B}_{xxx}\mathbf{w}(\hat{t}_{n,s}) = 0, \ s = \gamma_1, \gamma_2, \tag{4.22}$$

where $\mathbf{w}$ and $\mathbf{w}_t$ are given by (4.21) and (4.20) respectively.

From the initial condition $\mathbf{B}\mathbf{w}(t_0) = \Phi$ we get

$$\mathbf{w}(t_0) = \mathbf{B}^{-1}\Phi, \tag{4.23}$$

and by putting $s = 1$ in (4.21) we obtain

$$\begin{aligned} \mathbf{w}(t_{n+1}) &= \mathbf{w}(t_n + \tau_n) \\ &= \mathbf{w}(t_n) + \frac{\tau_n}{2}\left(\mathbf{a}^{(n)} + \mathbf{b}^{(n)}\right). \end{aligned} \tag{4.24}$$

Letting $T_N = \{\hat{t}_{n,s} : s = \gamma_1, \gamma_2,\ 0 \le n \le N-1\}$ and again assuming $\mathbf{B}^{-1}$ exists, we can now recursively compute $U(x,t)$ on $X_M \times T_N$ as follows:

Algorithm 4.20

1. *Solve (4.23) for* $\mathbf{w}(t_0)$.

2. *Given* $\mathbf{w}(t_n)$:

   (a) *Solve the collocation equations (4.22) for* $\mathbf{a}^{(n)}, \mathbf{b}^{(n)}$. *Then (4.21) gives* $\mathbf{w}(t)$ *on* $[t_n, t_{n+1}]$.

   (b) *Use (4.14) to obtain* $U(x,t)$ *anywhere on* $\Omega \times [t_n, t_{n+1}]$.

   (c) *Use (4.24) to obtain* $\mathbf{w}(t_{n+1})$.

Remark 4.1 The collocation matrix $\mathbf{B}$ will be non-singular whenever the set X_M of collocation points in space is chosen in accordance with the condition for strict positivity in theorem 4.13. This condition is satisfied by the set of Gauss points of the partition $\Pi_M^{(x)}$. Collocation at the Gauss points has been used successfully with initial value problems for ODEs, yielding solutions converging with accuracy $\mathcal{O}\left(h^{2s}\right)$

at the mesh points [23]. For PDEs, similar success has been achieved for elliptic boundary value problems [72, 51] and for parabolic initial value problems [29]. In view of these successes , we choose the Gauss points as our collocation points in space. In the time dimension we will try collocating at the Radau II points as well as at the Gauss points.

Remark 4.2 Recall that the matrix B is order $\tilde{d} = M\delta$ and has entries $\mathcal{B}_{i,j} = \tilde{B}_j(\hat{x}_i) = B_{(p+1-\delta)/2+j}(\hat{x}_i)$. Since only $p+1$ B-splines are supported on any subinterval Ω_m, there are at most $p+1$ non-zeros on any row of $\mathbf{B}$. The semi-bandwidth is equal to $p-(p+1-\delta)/2$. The structure of the matrix $\mathbf{B}$ is shown diagramatically below for $p = 5$ and $\delta = 2$. The asterisks represent the possibly non-zero entries and the zero entries are omitted for clarity.

$$\left[\begin{array}{cccccccccc}
* & * & * & * & & & & & & \\
* & * & * & * & & & & & & \\
\hline
* & * & * & * & * & * & & & & \\
* & * & * & * & * & * & & & & \\
\hline
 & & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & & \\
 & & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & & \\
\hline
 & & & & * & * & * & * & * & * \\
 & & & & * & * & * & * & * & * \\
\hline
 & & & & & & * & * & * & * \\
 & & & & & & * & * & * & *
\end{array}\right]$$

The m-th block row corresponds to the set of collocation points on the mesh subinterval Ω_{m-1}.

Remark 4.3 The collocation equations (4.16) and (4.22) are nonlinear and will need to be solved by an iterative procedure. The systems have order $M\delta$ and $2M\delta$ respectively.

4.4 Implementation Details

Our Fortran implementations of algorithms 4.19 and 4.20 make use of available software for linear algebra and the evaluation of B-splines . All such software is in the public domain and available by electronic mail or file transfer from the NETLIB archive.

The PPPACK software [22] is used to evaluate the B-splines and their derivatives. These are evaluated at the collocation points and stored in compact representations of the matrices B, $\mathbf{B}_x$, and $\mathbf{B}_{xxx}$.

The equations $\mathbf{B}\mathbf{w}(t_0) = \Phi$ are solved for $\mathbf{w}(t_0)$ using LINPACK [27] factor and solve routines for general band matrices. The factorization overwrites $\mathbf{B}$, so we keep a copy for later use.

The collocation equations are solved iteratively for $\mathbf{w}(t_{n+1})$ using a simple Newton iteration

$$\mathbf{w}^{(k+1)} = \mathbf{w}^{(k)} - \mathbf{J}^{-1}\mathbf{F}(\mathbf{w}^{(k)})$$

with $\mathbf{w}(t_n)$ taken as the initial estimate $\mathbf{w}^{(0)}$. Here the vector function $\mathbf{F}(\mathbf{w})$ is the left side of the collocation equation and $\mathbf{J}$ is its Jacobian. The Jacobian is computed afresh in each time step, but then held fixed through the iterations, permitting us to solve the collocation equations with only one matrix factorization. The Jacobian inherits the band structure of the matrix $\mathbf{B}$, and we again use the LINPACK software for the factorization and subsequent solves. An analytic expression for the Jacobian can be obtained from the collocation equation. This expression, as well as the successive

values of F required through the iterations, are computed with the assistance of BLAS routines [60, 28]. Our stopping criteria require that the collocation equation is satisfied within a given tolerance, and that successive iterates differ by less than a (possibly different) given tolerance. The second criterion is included to ensure that the method is converging to a solution when the first criterion is met. Experience showed that there was little to be gained, and often much to be lost, by the use of very small tolerances: see table 5.6.

At preselected times, the solution is computed at the mesh points and written to a file for plotting. At the outset, de Boor's software is again used to evaluate the B-splines , this time at the mesh points. These values are stored in a compact representation of a non-square matrix $\mathrm{M} = [B_j(x_i)]_{(M-1)\times\tilde{d}}$. The product $\mathrm{Mw}(t_n)$ then gives $U(x_m, t_n)$, $m = 1, \ldots, M-1$, while the boundary values $U(x_0, t_n)$ and $U(x_M, t_n)$ are taken to be zero.

If we wish to test the fidelity of the numerical method to our conservation laws, we can compute the B-splines and their first derivative at the quadrature points and store these values in compact representations of matrices Q and Q_x. Then at the same preselected output times we can compute Qw, and $\mathrm{Q}_x\mathrm{w}$ and combine them as required to obtain U, U^2, and $U^3 - U_x/2$ at the quadrature points.

Chapter 5

Numerical Results

In this chapter we examine the performance of the methods developed in chapter four and compare those methods with that of Zabusky and Kruskal. The simplicity of the Zabusky-Kruskal method makes it ideal as a basis for comparison, as it can be implemented quickly and efficiently without specialized or arcane knowledge.

Accurate soliton solutions of the KdV equation present a considerable challenge to numerical methods. The greatest difficulty appears to lie with the speed of the solitons (see for example [50]), so that the numerical soliton increasingly lags the analytic soliton. The growing phase error results in a growing L_∞ (or max) error norm until such time as the numeric and analytic solitons completely separate and the L_∞ error norm saturates at the value of the soliton amplitude. It is this interpretation of error norm as a measure of phase error that we wish to keep in mind when we view the test results below. Recall that faster solitons are taller and narrower than

slower solitons. Consequently their profiles have steeper slope and so the L_∞ error norm becomes an increasingly sensitive measure of phase error with increasing soliton speeds.

It will be convenient to have some abbreviated names for the three methods under comparison, so we will refer to our implementation of algorithm 4.19 as C1, to that of algorithm 4.20 as C2, and to that of Zabusky-Kruskal as ZK. The names Cn, $n = 1, 2$ are intended as mnemonics for collocation at n points in each subinterval of time.

5.1 Testing Procedures

For both of the collocation methods we chose to use a spatial approximation from $S_5^{(5-2)}(\Pi_M^{(x)})$. The quintic splines are the splines of lowest degree that permit two collocation points in each spatial mesh interval. Use of lowest degree splines results in matrices of minimum bandwidth, and convergence results for collocation at Gauss points by other authors (see remark 4.1) suggest that two collocation points will give us fourth-order accuracy at the mesh points. How accuracy and performance vary with different spline space choices remains an open question.

The tests and comparisons are based on a suite of seven initial value problems for the KdV equation. The underlying idea is to choose the parameters of the numerical methods, e.g. space step and time step, in such a way that a prescribed accuracy requirement is met with a minimum use of cpu time. The time used by the various methods can then be compared.

The test problems consist of initial conditions that satisfy either the single soliton solution (2.3) or the two soliton solution (2.7). In each case the problem domain is $[-20, 20] \times [0, T]$, and it is required that the final solution $U(x, T)$ have an absolute error less than a prescribed quantity, as measured by the L_∞ norm. All of the two soliton problems witness a soliton interaction.

The relevant problem parameters are given in table 5.1. The first five problems are equivalent to those used by Taha and Ablowitz in their comparative study [88], and the last two are added to further challenge the methods with faster solitons. The typical two-soliton solution that was shown graphically in chapter 2 (figures 2.1 and 2.2) is the analytic solution to problem 7. This can be shown to be identical to the frequently cited [30, 59, 67] solution

$$u(x,t) = \frac{3 + 4\cosh(2x - 8t) + \cosh(4x - 64t)}{[3\cosh(x - 28t) + \cosh(3x - 36t)]^2}$$

on the domain $[-20, 20] \times [-1, 1]$.

Table 5.1: Initial conditions, domain, and required accuracy for the seven test problems. n: number of solitons, c_i: speed of i-th soliton, x_i: position of i-th soliton at $t = 0$. Solution at time T must have error less than E_{max} in L_∞ norm.

problem	n	c_1	x_1	c_2	x_2	$[0, T]$	E_{max}
1	1	2	0			[0,1]	0.005
2	1	4	0			[0,1]	0.01
3	1	8	0			[0,1]	0.022
4	2	1	0	2	-2	[0,3]	0.002
5	2	1	0	5	-4.8	[0,2.4]	0.02
6	2	3	-2.23965400369905	9	-8.56101403435839	[0,2]	0.02
7	2	4	-3.45069385566594	16	-15.72534692783297	[0,2]	0.08

All timing tests were run on a MIPS R2000A/R3000 33 MHz cpu equipped with a MIPS R2010A/R3010 floating point unit and 64 Kbyte data cache.

We also monitor the evolution of the conserved quantities I_1, I_2, I_3 given by the integrals in equations (2.10), (2.11) and (2.12), respectively. To evaluate the integrals for the C1 and C2 methods we used an eight point Gauss quadrature formula, which has a zero error for polynomials of degree 15 or less. This includes our approximations for U, U^2, and $U^3 - U_x^2/2$. Values of U_x were computed directly from the B-spline derivatives at the quadrature points. For the ZK method we used Simpson's rule (following [88]) on the approximate values at the mesh points, and approximated U_x with finite differences in U.

For problems 1, 2, 3, and 7 we were able to obtain exact values for the integrals using the MAPLE® [16] symbolic computation system. We made use of the fact that the solution to problem 7 at time $t = 1$ reduces to $u(x,1) = 6\mathrm{sech}^2 x$. For the remaining problems we evaluated the integrals using the eight point Gauss quadrature scheme, with analytic values for u and u_x on one thousand intervals of length 0.04 each. Where exact integral values were available, results from the quadrature agreed to nine significant digits or better. The values obtained are shown in table 5.2.

The importance of conservation in numerical schemes is discussed in some detail by de Frutos and Sanz-Serna [25], using the KdV equation for illustration. They observe that schemes conserving I_2 exactly have no error in soliton amplitude and only linear growth in soliton phase error, whereas nonconservative schemes experience

Table 5.2: Values of the conserved quantities I_1, I_2, I_3 for the seven test problems

problem	integral	analytic	quadrature
p1	I_1	$2\sqrt{2}$	2.82842712
	I_2	$4/3\sqrt{2}$	1.88561808
	I_3	$4/5\sqrt{2}$	1.13137085
p2	I_1	4	4.00000000
	I_2	16/3	5.33333333
	I_3	32/5	6.40000000
p3	I_1	$4\sqrt{(2)}$	5.65685425
	I_2	$32/3\sqrt{(2)}$	15.0849446
	I_3	$128/5\sqrt{(2)}$	36.2038672
p4	I_1		4.82842707
	I_2		2.55228474
	I_3		1.33137084
p5	I_1		6.47213593
	I_2		8.12022659
	I_3		11.3803398
p6	I_1		9.46410161
	I_2		21.4641016
	I_3		51.7176914
p7	I_1	12	12.0000000
	I_2	48	48.0000000
	I_3	1056/5	211.200000

linear growth in amplitude error and quadratic growth in phase error.

5.2 Time Collocation at Gauss Points

Collocation at the Gauss points yielded favourable results. Table 5.3 shows data that is consistent with fourth order accuracy in time, as we might expect from collocation at two Gauss points. The fourth column in the table is approximately constant, showing that the final error is proportional to the fourth power of the timestep used.

Table 5.3: Error is proportional to fourth power of timestep for method C2. Data was obtained from solutions of problem 7 using a constant space step h.

h	τ	$\|\|E\|\|_\infty$	$\|\|E\|\|_\infty/\tau^4$
.125	0.75e-2	3.1337e-2	0.9904e+7
	1.00e-2	7.0979e-2	0.7098e+7
	1.25e-2	3.3144e-1	1.3576e+7
	1.50e-2	4.9410e-1	0.9760e+7
	1.75e-2	9.8292e-1	1.0480e+7
	2.00e-2	1.6628e-0	1.0393e+7
	2.25e-2	2.5853e-0	1.0087e+7
	2.50e-2	3.7038e-0	0.9482e+7

Table 5.4 shows the elapsed cpu time and the error achieved at times $t = 2$ and, for problems 1 to 5, also at times $t = T$, where T is the time at the end of the varying intervals $[0, T]$ listed in table 5.1 and used in [88]. Table 5.4 also shows the space and time steps used to satisfy the accuracy constraints. It is clear that in all but the simplest problems, the collocation methods outperform the ZK method, and C2 outperforms C1. The efficiency of C2 is especially noticeable in problems 3, 5, 6, and

7, which have the faster solitons. In these problems the relatively low accuracy of ZK demands the use of a small space step to meet the accuracy requirement, and the resulting short time step imposed by the stability criterion prevents the method from being competitive.

The table gives the error at only one or two times for a given problem-method. A more complete view of the evolution of the error for the three methods for the case of problem 7 is shown in figures 5.1, 5.2, and 5.3. The growing phase error

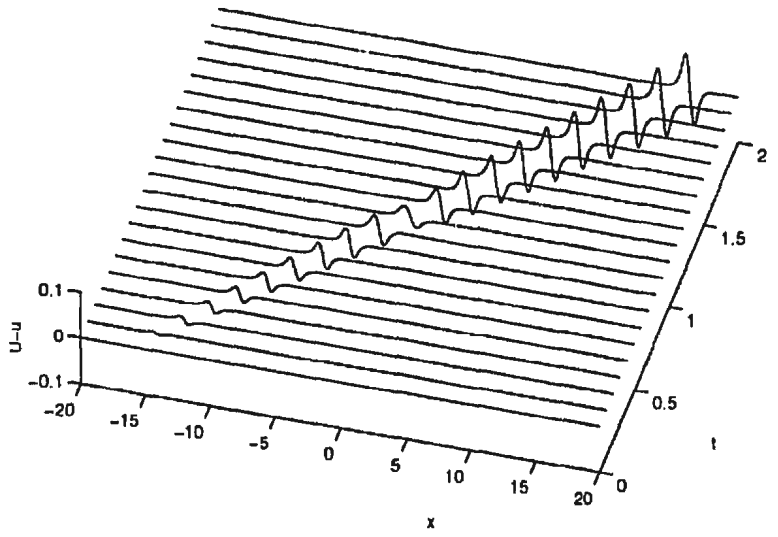

Figure 5.1: Evolution of error U-u for the method ZK for the two soliton initial condition of problem 7.

is clearly visible for all methods. Also visible is a small, nonphysical oscillation for the C2 method. This oscillation appears to be characteristic of high order spline approximations [5, 79, 81]. However, the oscillation is clearly very small and need not be a concern. Figure 5.4 shows the small oscillation in comparison to the solitons. The analytic solution to problem 7 was shown earlier in figure 2.1. At the scale of

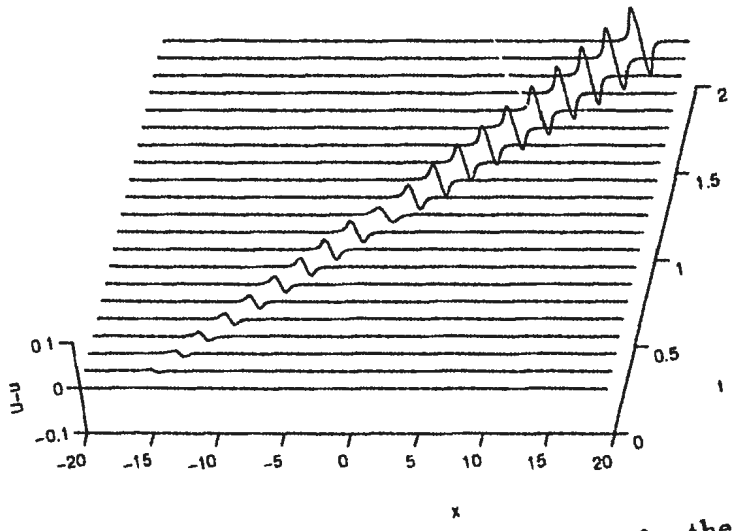

Figure 5.2: Evolution of error U-u for the method C1 for the two soliton initial condition of problem 7.

that figure and the current accuracy constraint, plots of the analytic and numeric solutions are indistinguishable.

Table 5.5 shows the maximum absolute relative errors in the value of the integrals I_1, I_2, I_3, and also the times at which those errors occur. Representative time series of the conserved quantities are shown in figures 5.5 to 5.16. The methods perform better on some of the problems than they do on others. In general the time series show oscillations about a mean trend, the trend deviating only slightly from the analytic values. The third integral is overestimated by the ZK method and suffers a loss during a two soliton interaction.

Table 5.4: Cpu times required to achieve prescribed accuracy, and the actual error norm achieved at $t = T$, where $[0, T]$ is the time domain given in table 5.1, and also at $t = 2$. Also given are the space and time steps that were used, and the total number of Newton iterations required to solve t/τ nonlinear systems.

problem	method	M	h	τ	t	|| error ||	cpu	itns
1	ZK	320	.125	7.0e-4	1	.00243432	4.71	
					2	.00426564	7.19	
	C1	50	.800	8.0e-2	1	.00344671	13.96	39
					2	.00610313	14.54	75
	C2	80	.500	3.4e-1	1	.00386667	14.56	38
					2	.00481102	15.39	76
2	ZK	512	.078125	1.0e-4	1	.00882357	30.19	
					2	.01620527	57.81	
	C1	64	.625	2.5e-2	1	.00887315	15.34	120
					2	.01126806	16.25	240
	C2	80	.500	1.0e-1	1	.00681757	15.39	92
					2	.00770105	16.89	184
3	ZK	1050	.038095	2.0e-5	1	.02144416	299.22	
					2	.04183051	580.50	
	C1	200	.20	4.0e-3	1	.01640828	39.35	750
					2	.03218255	64.46	1500
	C2	120	.333	2.0e-2	1	.01161539	26.53	501
					2	.06561467	39.48	1000
4	ZK	400	.1	2.5e-4	2	.00059953	21.67	
					3	.00133025	28.34	
	C1	100	.4	5.0e-2	2	.00047360	16.40	120
					3	.00127573	17.38	180
	C2	50	.800	3.0e-1	2	.00183198	14.49	57
					3	.00166683	14.68	84
5	ZK	800	.05	4.0e-5	2	.01255676	228.14	
					2.4	.01544735	271.62	
	C1	200	.2	8.0e-3	2	.00907978	39.73	75
					2.4	.01421134	53.15	112
	C2	100	.4	8.0e-2	2	.01197605	19.38	37
					2.4	.01380330	21.12	45
6	ZK	2000	.02	2.0e-6	2	.01633522	9439.94	
	C1	320	.125	2.0e-3	2	.01488418	200.42	3000
	C2	160	.25	2.5e-2	2	.01267173	49.65	1240
7	ZK	2560	.015625	1.0e-6	2	.07626035	27030.62	
	C1	1000	.04	8.0e-4	2	.07409344	1478.94	7500
	C2	320	.125	1.0e-2	2	.07097909	255.28	3564

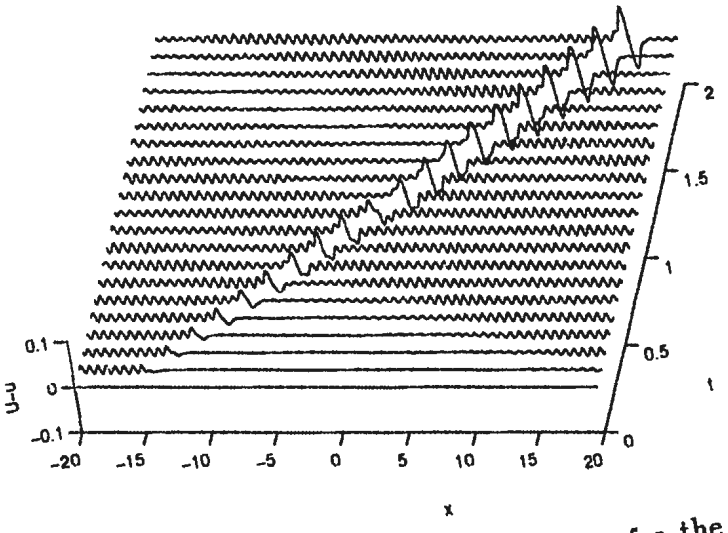

Figure 5.3: Evolution of error U-u for the method C2 for the two soliton initial condition of problem 7.

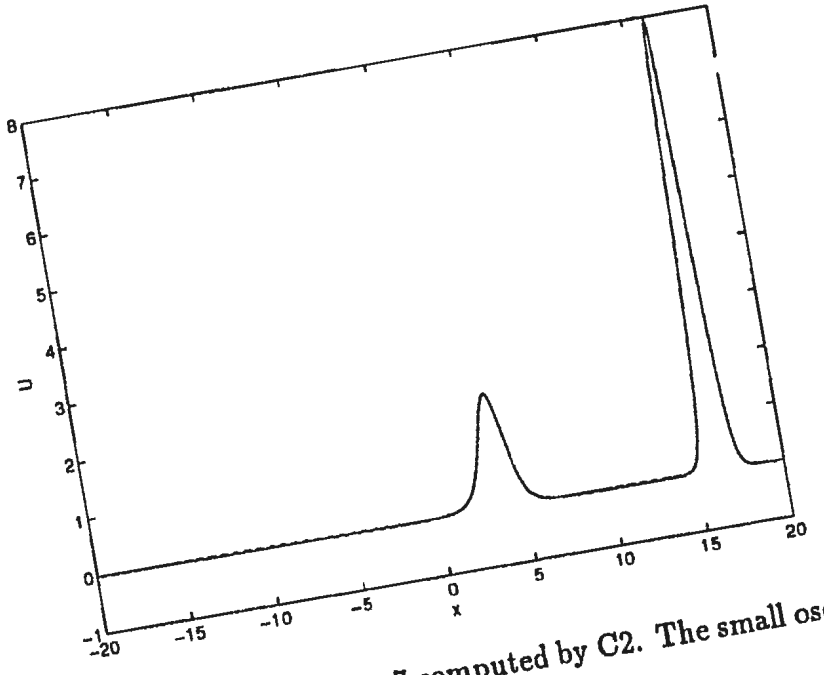

Figure 5.4: Final solution for problem 7 computed by C2. The small oscillation about U=0 is just noticeable.

Table 5.5: Maximum absolute relative errors $\times 10^3$ found in the conserved quantities I_1, I_2, I_3, and the times t_1, t_2, t_3 at which they occur.

problem	method	I_1	t_1	I_2	t_2	I_3	t_3
1	ZK	0.17885	1.50	0.03454	1.80	2.53138	1.80
	C1	0.19429	2.00	1.29910	2.00	1.93585	2.00
	C2	0.07227	2.04	0.51879	1.70	1.25293	1.70
2	ZK	0.12610	1.00	0.02108	1.80	1.96781	1.80
	C1	2.14937	1.85	7.27753	1.85	9.90836	1.85
	C2	0.67938	2.00	1.53639	1.90	3.06201	1.90
3	ZK	0.06104	2.00	0.00737	0.70	0.93079	0.46
	C1	0.02734	2.00	0.04781	1.74	0.07963	1.74
	C2	0.33649	1.92	4.02735	2.00	6.80650	2.00
4	ZK	0.03591	0.84	0.01982	1.20	0.93982	2.40
	C1	0.01128	2.70	0.04475	2.65	0.19283	1.70
	C2	0.12904	2.10	0.53118	2.10	0.88042	2.10
5	ZK	0.03989	0.36	0.01123	0.36	0.98139	2.40
	C1	0.00609	1.06	0.03046	1.04	0.09499	0.80
	C2	0.94744	1.60	1.40225	1.60	4.08813	1.60
6	ZK	0.00695	0.12	0.00247	1.80	0.27735	1.80
	C1	0.00547	1.44	0.01907	1.46	0.04490	1.00
	C2	0.47937	0.68	0.53827	0.90	1.17842	0.65
7	ZK	0.02873	1.10	0.00255	0.02	0.30645	0.02
	C1	0.08659	2.00	0.18796	1.98	0.30919	1.98
	C2	0.14376	1.86	0.45491	1.98	2.47116	1.98

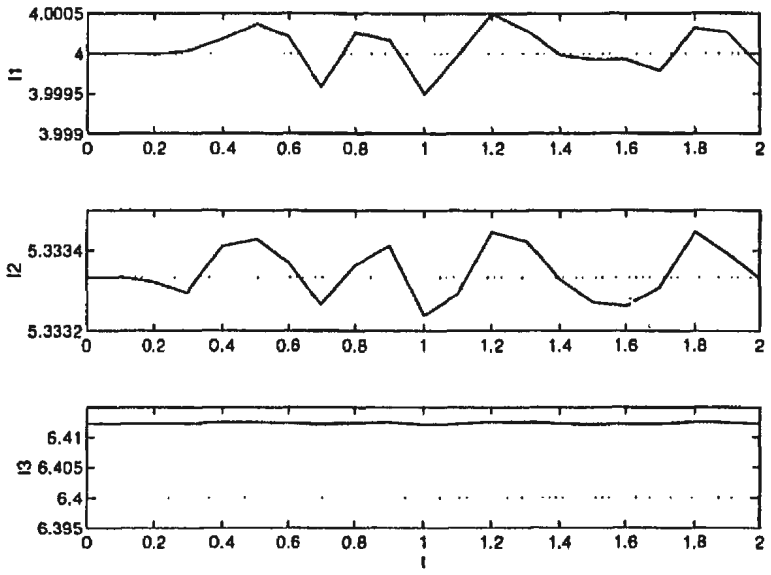

Figure 5.5: Evolution of conserved quantities for problem 2 by ZK method (solid). Top to bottom: I_1, I_2, I_3. Analytic values shown dotted.

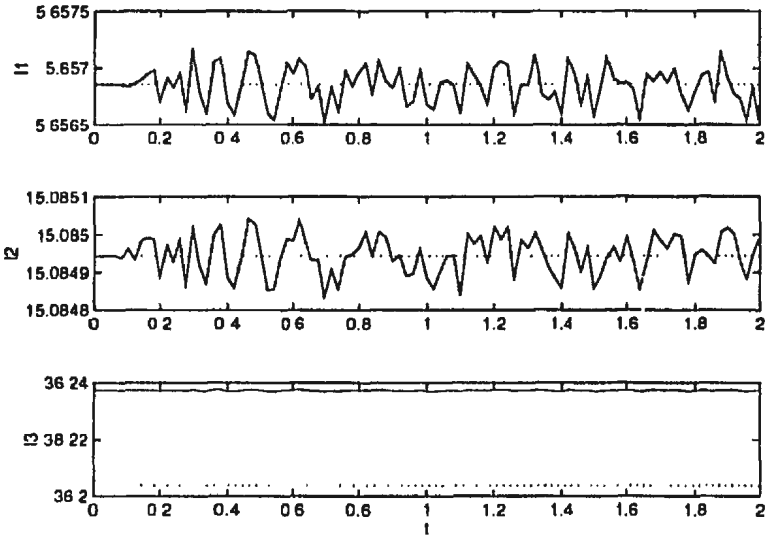

Figure 5.6: Evolution of conserved quantities for problem 3 by ZK method (solid). Top to bottom: I_1, I_2, I_3. Analytic values shown dotted.

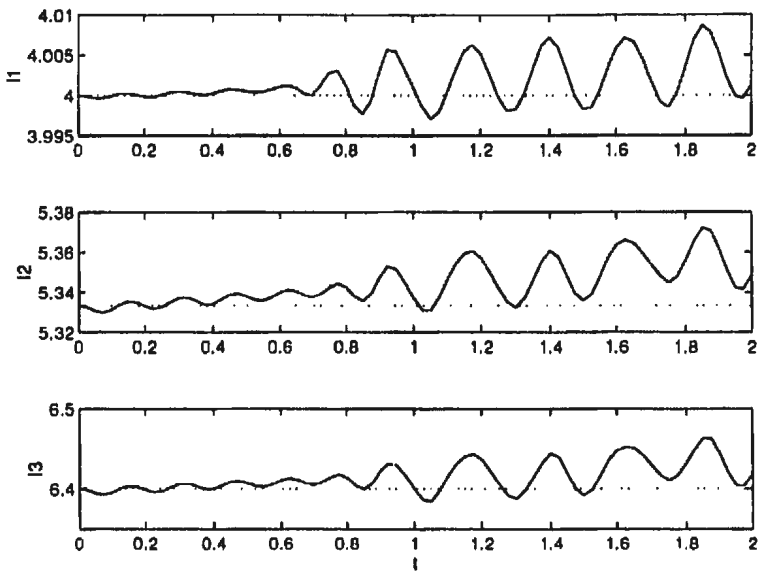

Figure 5.7: Evolution of conserved quantities for problem 2 by C1 method (solid). Top to bottom: I_1, I_2, I_3. Analytic values shown dotted.

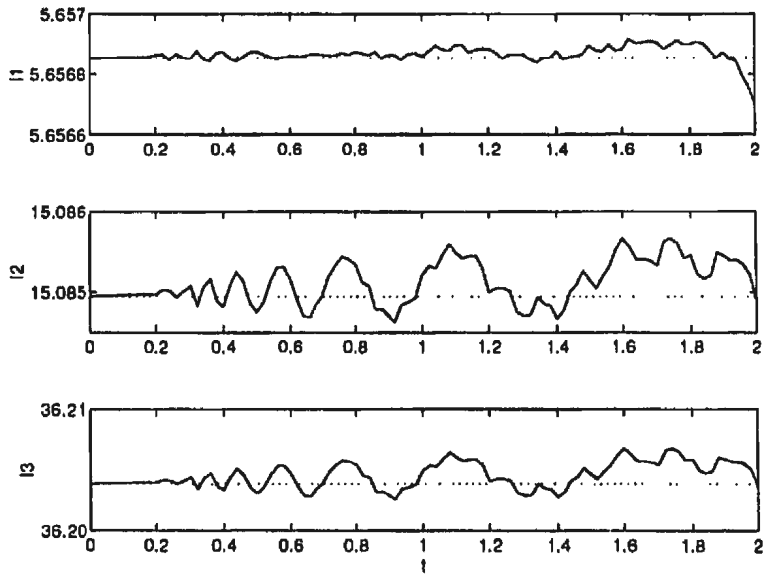

Figure 5.8: Evolution of conserved quantities for problem 3 by C1 method (solid). Top to bottom: I_1, I_2, I_3. Analytic values shown dotted.

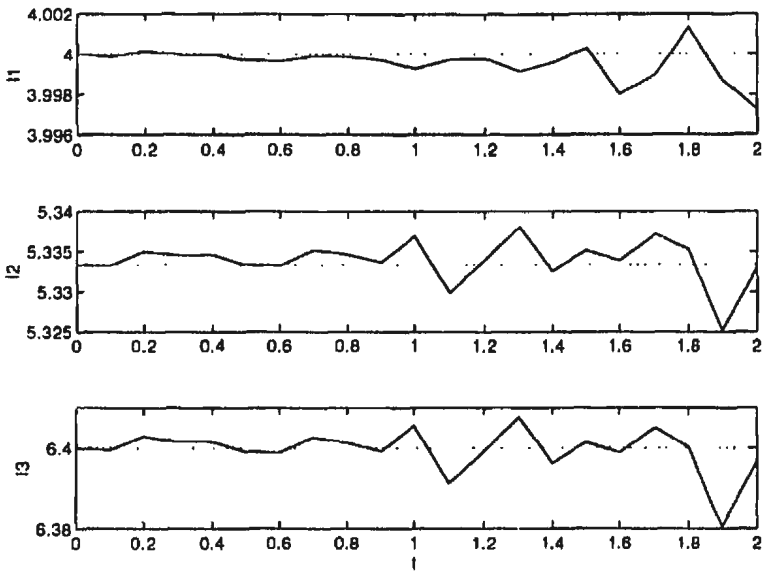

Figure 5.9: Evolution of conserved quantities for problem 2 by C2 method (solid). Top to bottom: I_1, I_2, I_3. Analytic values shown dotted.

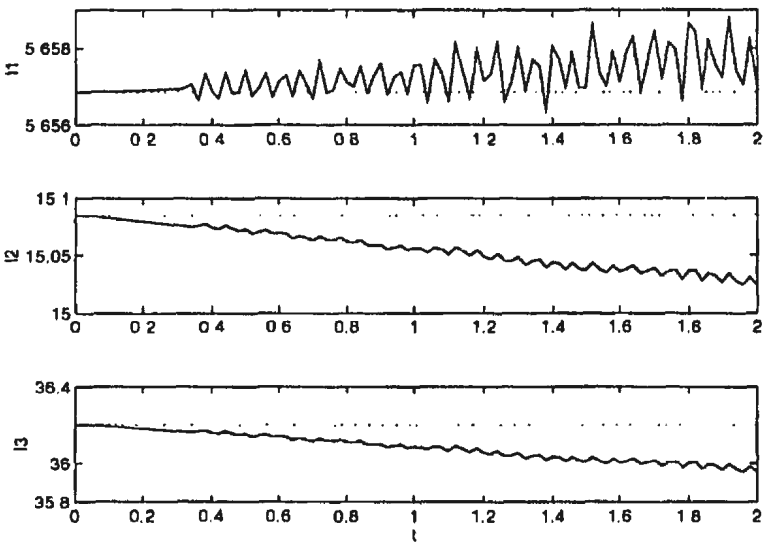

Figure 5.10: Evolution of conserved quantities for problem 3 by C2 method (solid). Top to bottom: I_1, I_2, I_3. Analytic values shown dotted.

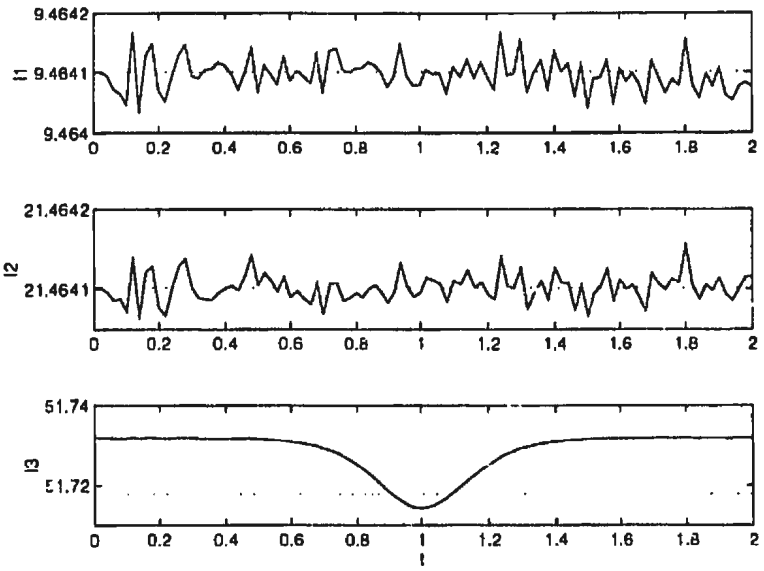

Figure 5.11: Evolution of conserved quantities for problem 6 by ZK method (solid). Top to bottom: I_1, I_2, I_3. Analytic values shown dotted.

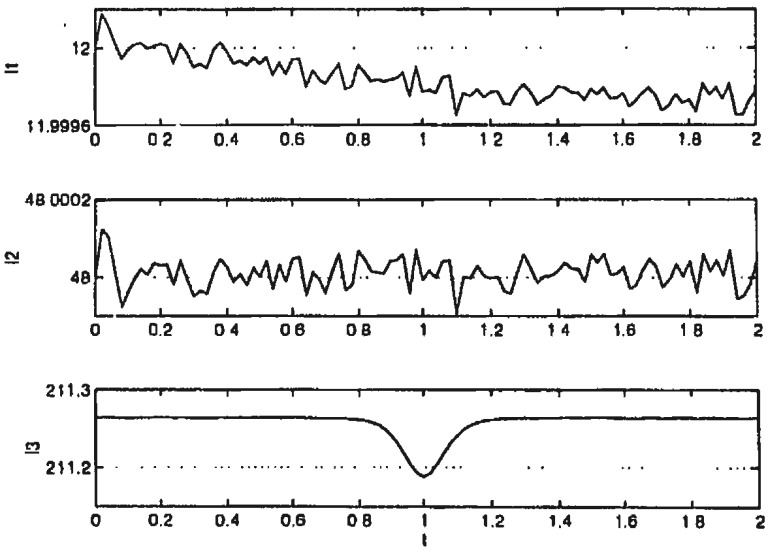

Figure 5.12: Evolution of conserved quantities for problem 7 by ZK method (solid). Top to bottom: I_1, I_2, I_3. Analytic values shown dotted.

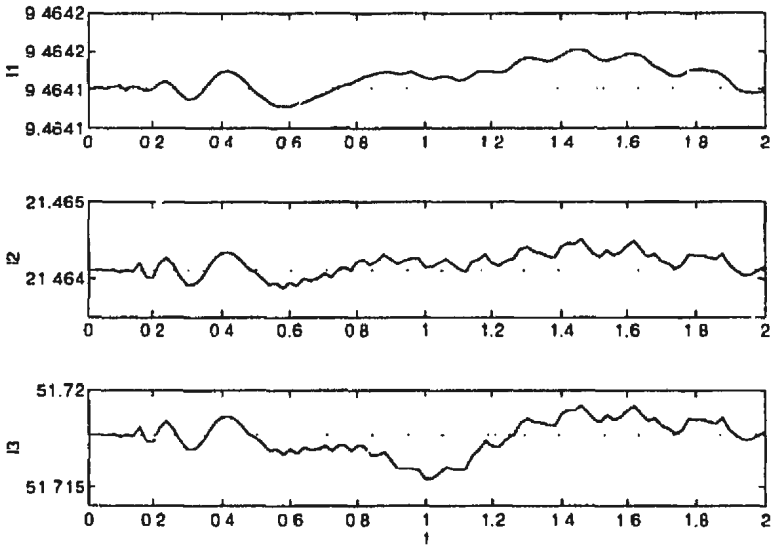

Figure 5.13: Evolution of conserved quantities for problem 6 by C1 method (solid). Top to bottom: I_1, I_2, I_3. Analytic values shown dotted.

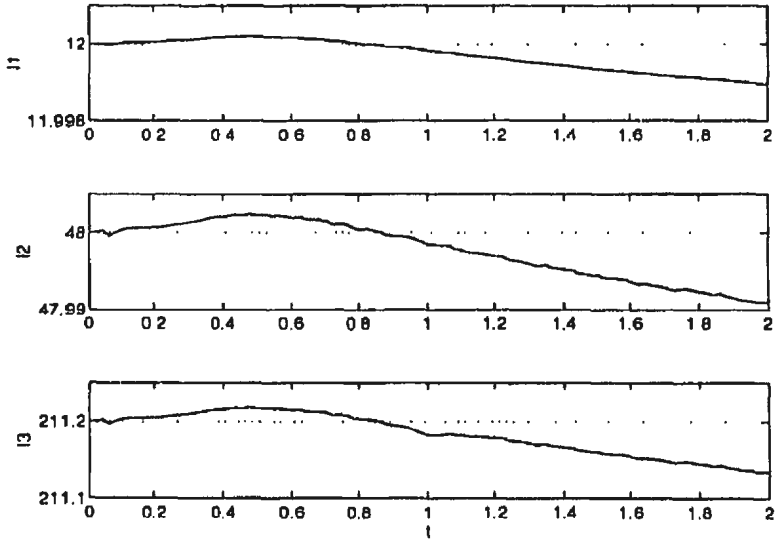

Figure 5.14: Evolution of conserved quantities for problem 7 by C1 method (solid). Top to bottom: I_1, I_2, I_3. Analytic values shown dotted.

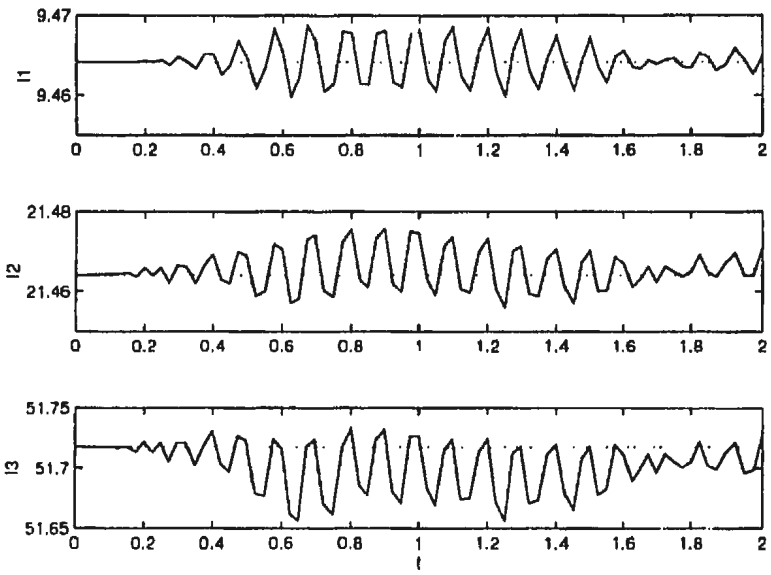

Figure 5.15: Evolution of conserved quantities for problem 6 by C2 method (solid). Top to bottom: I_1, I_2, I_3. Analytic values shown dotted.

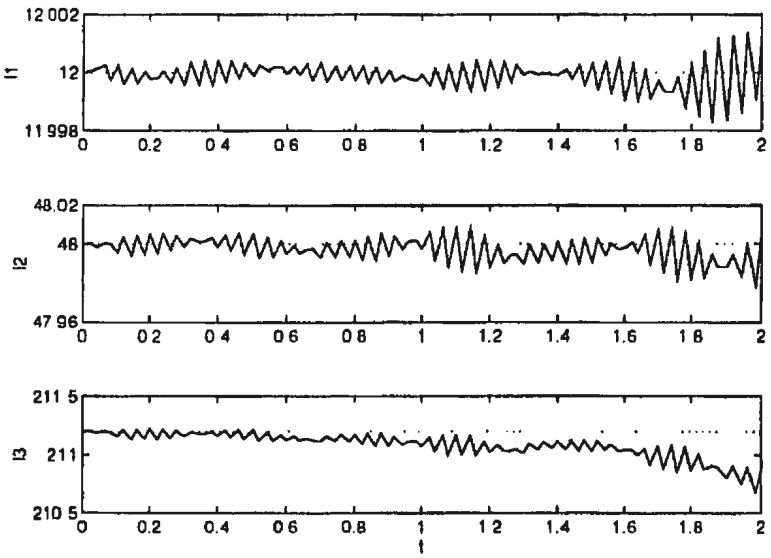

Figure 5.16: Evolution of conserved quantities for problem 7 by C2 method (solid). Top to bottom: I_1, I_2, I_3. Analytic values shown dotted.

5.3 Time Collocation at Radau Points

Collocation at the Radau II points yielded negative results. We report only one experiment with the C1 method on problem 7. Using the Radau II points with one collocation point means collocating at t_{n+1} when the solution is known at t_n. This is equivalent to implicit Euler, which is well known to be dissipative. The result is that the solitons lose amplitude as time progresses, and since soliton velocity is proportional to amplitude, there is an accompanying deceleration. The effect on phase error is naturally greater on the faster soliton, and so the interaction between the two solitons occurs later than it should. All of these effects can be seen in figures 5.17 to 5.19 (compare with the analytic solution in figures 2.1 and 2.2). At the same time, the conservation laws are violated, as shown in figure 5.20.

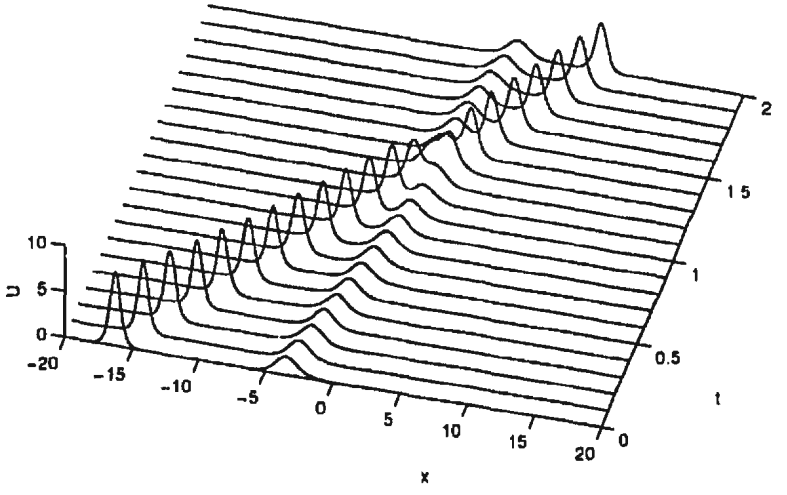

Figure 5.17: Two soliton solution of problem 7 by the C1 method using Radau II points for time collocation. Note tardiness of soliton interaction and decay in amplitude.

We note that our experience with both conservative and nonconservative schemes illustrate the results of de Frutos and Sanz-Serna [25]. Their figure 5.1 for a single soliton evolution under implicit Euler is strikingly similar to our figure 5.19.

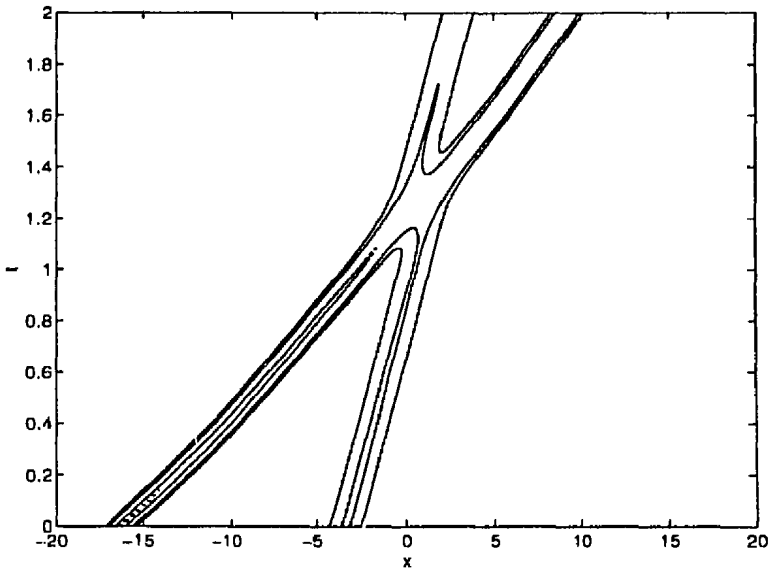

Figure 5.18: Contour plot of two soliton solution of problem 7 by the C1 method using Radau II points for time collocation. Note decay of soliton amplitude. Curved trajectories indicate deceleration.

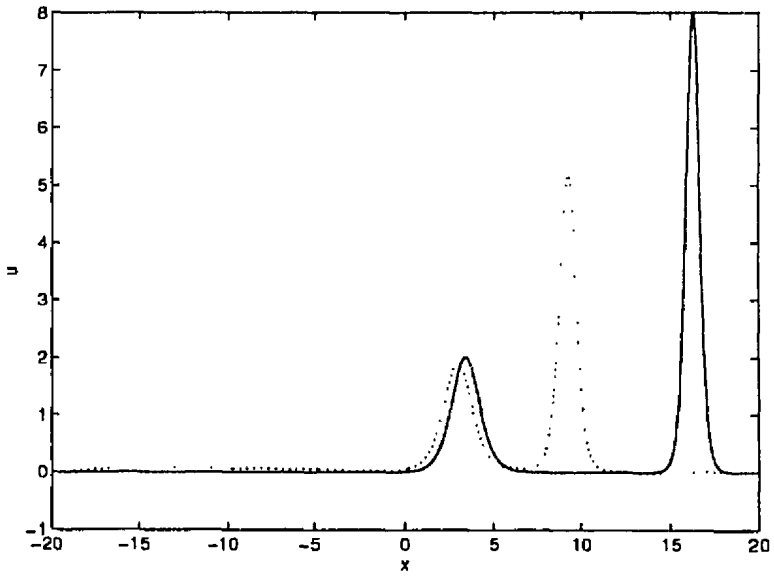

Figure 5.19: Two soliton solution of problem 7 at $t = 2$. Solid: analytic solution. Dotted: C1 method using Radau II points for time collocation. Note phase and amplitude errors in the numerical solution.

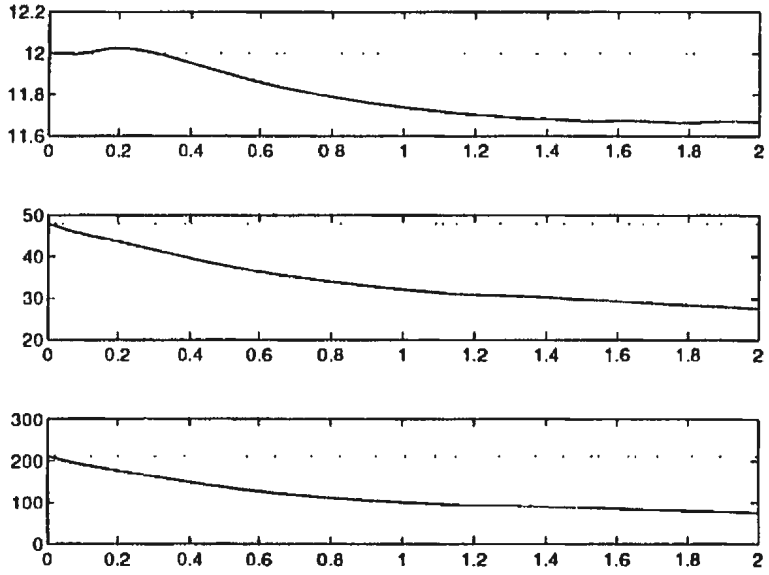

Figure 5.20: Violation of conservation laws using Radau II points. Solid: C1 method using Radau II points for time collocation. Dotted: C1 method using Gauss points for time collocation.

5.4 Some Surprising Results

The search for appropriate parameters to satisfy the accuracy requirements of the tests revealed two interesting phenomena.

The first of these is concerned with the stopping criterion for the Newton iteration. One naturally expects that use of a wide tolerance results in less work (fewer iterations) than a narrow tolerance, but that this should result in a more accurate final solution to the PDE is an unexpected bonus. Table 5.6 shows that in many cases less work leads to higher accuracy. The phenomenon seems to occur mainly when the space and time steps are quite large, when the PDE solution is of relatively low accuracy.

The second phenomenon is also surprising: use of too *small* a timestep can give rise to unbounded growth of the solution just inside the left boundary of the domain. An example of this in its early stages is shown in figure 5.21, where we have plotted the negative of the solution for clarity. We have seen few literature references to difficulties caused by a small timesteps. Maritz and Schoombie [62] indicate that small timesteps suit their purpose in the investigation of parasitic waves caused by discontinuities in the initial data. Schoombie [80] and Hedstrom [49] both report increased error growth at small time steps when using a cubic Hermite Galerkin method. The error is attributed to the presence of a secondary wave induced by the splines, and improvement in accuracy is obtained by filtering. However, the errors there remained small compared to the soliton amplitudes, so the difficulty here seems

Table 5.6: Work done and final L_∞ error norm with various stopping criteria for solution of collocation equations $F = 0$. Note that strict tolerance does not necessarily beget most accuracy. Iteration stops when $||F|| < tol_1$ and successive iterates differ less than tol_2.

h	τ	tol_1	$\|\|e\|\|_{max}\|\|$	iterations
$tol_2 = tol_1 * 10^{-3}$				
0.250	2e-2	1e-3	0.9891	4069
		1e-2	0.9891	3648
		1e-1	0.9896	3126
		1e-0	0.9933	2701
0.125	2e-2	1e-1	1.9001	3091
		1e-0	1.9031	2650
		1e+1	1.9037	2116
	1e-2	1e-9	0.0751	10907
		1e-2	0.0751	5575
		1e-1	0.0753	4962
$tol_2 = tol_1$				
0.250	2e-2	1e-4	0.9891	3368
		1e-3	0.9885	2888
		1e-2	0.9837	2439
		1e-1	0.9145	1980
		1e-0	1.0279	1567
	1e-2	1e-5	0.3652	5619
		1e-4	0.3650	4976
		1e-3	0.3641	4260
		1e-2	0.3718	3584
		1e-1	0.3941	2843
0.125	2e-2	1e-3	1.9015	2817
		1e-2	1.8957	2413
		1e-1	1.8460	1936
		1e-0	1.1592	1568
		1e+1	1.1607	1096
	1e-2	1e-4	0.0753	4965
		1e-3	0.0748	4315
		1e-2	0.0673	3565
		1e-1	0.0839	2784
	8e-3	1e-4	.02285	5718
		1e-3	.02289	4952
		1e-2	.03353	3981

to be of a fundamentally different nature. At time of writing the phenomenon is still poorly understood and awaiting investigation. It is expected that the boundary conditions will be found to be somehow implicated in the problem. Fortunately, one generally wishes to work with as large a timestep as possible, consistent with accuracy requirements, so the phenomenon should not present any practical difficulties.

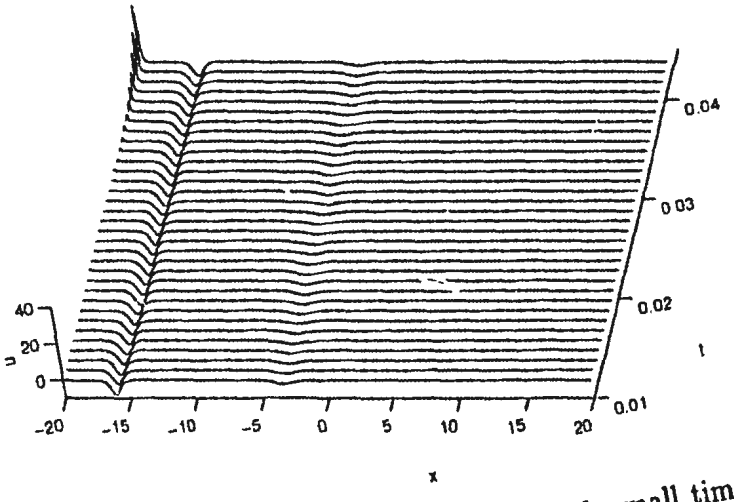

Figure 5.21: Growth of solution near left boundary with small timestep $\tau = 5 \cdot 10^{-4}$. Space step $h = 0.125$. C2 method on problem 7. The negative of the solution is plotted for clarity.

Chapter 6

Conclusions

6.1 Conclusions

We have constructed a bivariate spline collocation method for the numerical solution of the Korteweg-deVries equation. Using two Gauss points for time collocation, we obtain highly accurate solutions very efficiently. There is numerical evidence for fourth-order convergence in time.

All of the methods tested show some variability in their fidelity to the first three laws in the infinite hierarchy of conservation laws for the KdV equation, but do not deviate far from conservation in a mean sense.

Our method, and our code, can be easily adapted to other equations with soliton or soliton-like solutions, for which we can expect the method to show similar performance.

6.2 Work Remaining

There is still much work to be done, however. Until now we have considered only uniform partitions of space, but the method can easily accomodate an adaptive grid. Clearly we do not need extremely accurate solutions in a region where the solution is uniformly constant, so it makes sense to have a larger step in these regions and confine the finer grid to the more interesting neighbourhoods around the solitons. For very large problems, such as long time integrations, grids moving with the solitons will be essential to save computer memory space. If the grid is simply translated, there may be little additional work, but if it is otherwise modified as it moves the matrices will need to be reformed and some (or all) of the B-splines will need to be recomputed at each timestep. How the efficiency of the resulting method compares to other methods using similar devices will need to be studied.

Questions raised by our testing also need to be addressed. We need to confirm our expectation of efficient, accurate performance with other equations. Good candidates include the nonlinear Schrödinger equation and the sine-Gordon equation, both of which have analytically available soliton solutions. Also, recall that we have tested approximations from only one spatial spline space. It is not clear how accuracy and efficiency will be affected by a different choice.

Of considerable interest is the nature of the relationship between the accuracy of the approximate solution of the nonlinear collocation equations and the accuracy of the approximate PDE solution. How does a more accurate approximation to solution

of the collocation equations lead to a less accurate solution to the PDE at the end of the time interval? Is this unique to this equation or to these methods, or is this a wide ranging phenomenon?

Also of great interest is the cause of the apparent instability at short time steps. Is it caused by an excitation of the small oscillation we saw in figure 5.4? If so, what causes the excitation? If we constructed a flow through boundary condition, so that any spurious wave was not reflected back into the domain, or if we used periodic boundary conditions, would the problem disappear? Pending answers to these questions, is it possible to predict the minimum timestep that can be used?.

Arguably the most important work remaining, particularly in view of the apparent instability mentioned above, is a full mathematical analysis of stability and convergence. These problems are difficult, and we confine ourselves to a brief description of a possible course for a convergence analysis.

We propose following the general framework of Verwer and Sanz-Serna [93] for convergence analyses of Method of Lines schemes for nonlinear PDEs. In this approach, the analysis is a two step process, considering first the error resulting from a semidiscretization in space and then the error from the time integration technique.

Let $\tilde{u}(x,t)$ be the approximation to $u(x,t)$ obtained from the discretization in space by collocation at the Gauss points of our partition of the spatial interval $[a,b]$. Although $\tilde{u}(x,t)$ satisfies the KdV equation at the collocation points, it does not do

so everywhere, and so we find

$$\tilde{u}_t + 6\tilde{u}\tilde{u}_x + \tilde{u}_{xxx} = \rho^{(x)}(x,t), \tag{6.1}$$

where the residual $\rho^{(x)}(x,t)$ vanishes whenever x belongs to the set of (spatial) collocation points X_M. Introducing the spatial discretization error

$$\epsilon^{(x)} = u - \tilde{u},$$

equation (6.1) leads to

$$\epsilon_t^{(x)} - 6\epsilon^{(x)}\epsilon_x^{(x)} + \epsilon_{xxx}^{(x)} + 6\left(u\epsilon^{(x)}\right)_x + \rho^{(x)} = 0.$$

Thus the spatial discretization error depends on the exact solution u to the PDE and on the residual $\rho^{(x)}(x,t)$ which depends on the choice of (spatial) collocation points and on the approximating power of the spline space chosen for the spatial approximation.

Now let $U(x,t)$ be an approximation to $\tilde{u}(x,t)$ resulting from discretization in time. Then

$$U_t + 6UU_x + U_{xxx} = \rho(x,t), \tag{6.2}$$

where the residual $\rho(x,t)$ vanishes whenever (x,t) belongs to the set of collocation points $X_M \times T_N$. Introducing the time discretization error

$$\epsilon^{(t)} = \tilde{u} - U,$$

equation (6.2) leads to

$$\epsilon_t^{(t)} - 6\epsilon^{(t)}\epsilon_x^{(t)} + \epsilon_{xxx}^{(t)} + 6\left(\tilde{u}\epsilon^{(t)}\right)_x + \rho^{(t)} = 0, \tag{6.3}$$

where $\rho^{(t)} = \rho - \rho^{(x)}$. Note that $\rho^{(t)}$ vanishes whenever t belongs to the set of time collocation points T_N. From (6.3) we see that the time discretization error depends on the exact solution $\tilde{u}$ to the sytem of ODEs resulting from the semidiscretization in space, and on the residual $\rho^{(t)}$ which depends on the choice of (time) collocation points and on the approximating power of the spline space chosen for the approximation in time.

Finally, we note that the full discretization error ϵ is given by

$$\epsilon = u - U = u - \tilde{u} + \tilde{u} - U = \epsilon^{(x)} + \epsilon^{(t)}.$$

Consequently a norm of ϵ satisfies

$$||\epsilon|| = ||\epsilon^{(x)} + \epsilon^{(t)}|| \leq ||\epsilon^{(x)}|| + ||\epsilon^{(t)}||$$

so we can obtain a bound on the full discretization error from bounds on the separate spatial and temporal contributions, and these contributions depend on both the choice of collocation points and the approximating power of the underlying spline spaces.

Bibliography

[1] M. B. Abbott. *An Introduction to The Method of Characteristics.* American Elsevier Publishing Company, Inc., 1966.

[2] K. Abe and O. Inoue. Fourier expansion solution of the Korteweg-de Vries equation. *Journal of Computational Physics*, 34:202–210, 1980.

[3] M. Abramowitz and I. A. Stegun, editors. *Handbook of Mathematical Functions*, volume 55 of *Applied Mathematics Series.* National Bureau of Standards (U.S.A), 1970.

[4] S. G. B. Airy. Tides and waves. *Encyclopaedia Metropolitana*, 1845.

[5] M. E. Alexander and J. L. Morris. Galerkin methods applied to some model equations for non-linear dispersive waves. *Journal of Computational Physics*, 30:428–451, 1979.

[6] W. F. Ames. *Numerical Methods for Partial Differential Equations.* Acedemic Press, second edition, 1977.

[7] A. Aoyagi and K. Abe. Runge-Kutta smoother for suppression of computational mode instability of leapfrog scheme. *Journal of Computational Physics*, 93:287–296, 1991.

[8] D. J. Benney. Long non-linear waves in fluid flows. *Journal of Mathematics and Physics*, 45:52–63, 1966.

[9] J. G. Blaschak and G. A. Kriegsmann. A comparative study of absorbing boundary conditions. *Journal of Computational Physics*, 77:109–139, 1988.

[10] J. Boussinesq. Theorie des ondes et des remous qui se propagent le long d'un canal rectangulaire horizontal, en communiquant au liquide contenue dans ce canal des vitesses sensiblement pareilles de la surface au fond. *J. Math. Pures et Appliquées*, 17(2):55–108, 1872.

[11] J. P. Boyd. Equatorial solitary waves. Part I: Rossby solitons. *Journal of Physical Oceanography*, 10:1699–1717, 1980.

[12] H. Brunner. On the numerical solution of nonlinear Volterra-Fredholm integral equations by collocation methods. *SIAM Journal on Numerical Analysis*, 27:987–1000, 1990.

[13] H. Brunner and P. J. van der Houwen. *The Numerical Solution of Volterra Equations*. North-Holland, 1986.

[14] R. K. Bullough. "The wave" "par excellence" the solitary progressive great wave of equilibrium of the fluid: An early history of the solitary wave. In M. Lakshmanan, editor, *Solitons. Introduction and Applications*, pages 7–42. Springer-Verlag, 1988.

[15] A. L. Camerlengo and J. J. O'Brien. Open boundary conditions in rotating fluids. *Journal of Computational Physics*, 35:12–35, 1980.

[16] B. W. Char, K. O. Geddes, G. H. Gonnet, B. L. Leong, M. B. Monagan, and S. M. Watt. *Maple V Language Reference Manual*. Springer-Verlag, 1991.

[17] C. I. Christov and K. L. Bekyarov. A Fourier-series method for solving soliton problems. *SIAM Journal on Scientific and Statistical Computing*, 11:631–647, 1990.

[18] R. A. Clarke. Solitary and cnoidal planetary waves. *Geophysical Fluid Dynamics*, 2:343–354, 1971.

[19] R. Courant, K. Friedrichs, and H. Lewy. Über die partiellen Differenzengleichungen der mathematischen Physik. *Math. Annalen*, 100:32–74, 1928.

[20] M. G. Cox. The numerical evaluation of B-splines. *J. Inst. Maths Applics*, 10:134–149, 1972.

[21] P. J. Davis and P. Rabinowitz. *Methods of Numerical Integration*. Acedemic Press, second edition, 1984.

[22] C. de Boor. *A Practical Guide to Splines.* Springer-Verlag, 1978.

[23] C. de Boor and B. Swartz. Collocation at gaussian points. *SIAM Journal on Numerical Analysis*, 10:582–606, 1973.

[24] J. de Frutos and J. M. Sanz-Serna. An easily implementable fourth-order method for the time integration of wave problems. *Journal of Computational Physics*, 103:190–168, 1992.

[25] J. de Frutos and J. M. Sanz-Serna. Erring and being conservative. Technical Report 1993/5, Universidad de Valladolid Departamento de Matemática Aplicada y Computación, May 1993.

[26] R. K. Dodd, J. C. Eilbeck, J. D. Gibbon, and H. C. Morris. *Solitons and Nonlinear Wave Equations.* Acedemic Press Inc., 1982.

[27] J. J. Dongarra, J. R. Bunch, C. B. Moler, and G. W. Stewart. *Linpack Users' Guide.* SIAM, 1979.

[28] J. J. Dongarra, J. DuCroz, S. Hammarling, and R. Hanson. An extended set of Fortran basic linear algebra subprograms. *ACM Transactions on Mathematical Software*, 14:1–32, 1988.

[29] J. Douglas, Jr. and T. Dupont. *Collocation Methods for Parabolic Equations in a Single Space Variable*, volume 385 of *Lecture Notes in Mathematics.* Springer-Verlag, 1974.

[30] P. G. Drazin and R. S. Johnson. *Solitons: an introduction.* Cambridge University Press, 1989.

[31] W. Eckhaus and A. van Harten. *The Inverse Scattering Transformation and the Theory of Solitons*, page 2. North Holland Publishing Company, 1981.

[32] E. Fermi, J. R. Pasta, and S. M. Ulam. Studies on nonlinear problems, I. Technical Report La-1940, Los Alamos, May 1955. See also [70, pages 143-156].

[33] A. P. Fordy. Soliton theory: a brief synopsis. In A. P. Fordy, editor, *Soliton Theory: A Survey of Results*, pages 3–22. Manchester University Press, 1990.

[34] B. Fornberg. On a Fourier method for the integration of hyperbolic equations. *SIAM Journal on Numerical Analysis*, 12:509–528, 1975.

[35] B. Fornberg and G. B. Whitham. A numerical and theoretical study of certain nonlinear wave phenomena. *Philosophical Transactions of the Royal Society of London*, 289:373–404, 1978.

[36] E. S. Fraga and J. L. Morris. A piecewise uniform adaptive grid algorithm for nonlinear dispersive wave equations. Technical Report NA/106, University of Dundee Department of Mathematics, July 1987.

[37] N. C. Freeman. Soliton interactions in two dimensions. *Advances in Applied Mechanics*, 20:1–37, 1980.

[38] N. C. Freeman and R. S. Johnson. Shallow water waves on shear flows. *Journal of Fluid Mechanics*, 42:401–409, 1970.

[39] C. S. Gardner, J. M. Greene, M. D. Kruskal, and R. M. Miura. Method for solving the Korteweg-deVries equation. *Physical Review Letters*, 19:1095–1097, 1967.

[40] C. S. Gardner, J. M. Greene, M. D. Kruskal, and R. M. Miura. Korteweg-deVries equations and generalizations. VI. Methods for exact solution. *Communications in Pure and Applied Math*, 27:97–133, 1974.

[41] C. S. Gardner and G. K. Morikawa. Similarity in the asymptotic behaviour of collision-free hydromagnetic waves and water waves. Technical Report NYO 9082, Courant Institute, 1960.

[42] K. Goda. On stability of some finite difference schemes for the Korteweg-de Vries equation. *Journal of the Physical Society of Japan*, 39:229–236, 1975.

[43] D. Gottlieb and S. A. Orszag. *Numerical Analysis of Spectral Methods: Theory and Applications*. SIAM, 1977.

[44] R. Greatbatch and T. Otterson. On the formulation of open boundary conditions at the mouth of a bay. *Journal of Geophysical Research*, 96(C10):18431–18445, 1991.

[45] I. S. Greig and J. L. Morris. A hopscotch method for the Korteweg-de Vries equation. *Journal of Computational Physics*, 20:64–80, 1976.

[46] R. Grimshaw. The solitary wave in water of variable depth. *Journal of Fluid Mechanics*, 42:639–656, 1970.

[47] C. A. Hall and T. A. Porsching. *Numerical Analysis of Partial Differential Equations.* Prentice Hall, 1990.

[48] A. Hasegawa. Optical solitons in fibers: theoretical review. In J. R. Taylor, editor, *Optical Solitons–Theory and Experiment*, pages 1–29. Cambridge University Press, 1992.

[49] G. W. Hedstrom. The Galerkin method based on Hermite cubics. *SIAM Journal on Numerical Analysis*, 16:385–393, 1979.

[50] R. L. Herman and C. J. Knickerbocker. Numerically induced phase shift in the KdV soliton. *Journal of Computational Physics*, 104:50–55, 1993.

[51] E. N. Houstis, R. E. Lynch, J. R. Rice, and T. S. Papatheodorou. Evaluation of numerical methods for elliptic partial differential equations. *Journal of Computational Physics*, 27:323–350, 1978.

[52] A. Jeffrey and T. Kakutani. Weak nonlinear dispersive waves: A discussion centred around the Korteweg-de Vries equation. *Siam Review*, 14:582–643, 1972.

[53] R. S. Johnson. A non-linear equation incorporating damping and dispersion. *Journal of Fluid Mechanics*, 42:49–60, 1970.

[54] R. S. Johnson. Water waves and Korteweg-de Vries equations. *Journal of Fluid Mechanics*, 97:701–719, 1980.

[55] K. Y. Kim, R. O. Reid, and R. E. Whitaker. On an open radiational boundary condition for weakly dispersive tsunami waves. *Journal of Computational Physics*, 76:327–348, 1988.

[56] D. J. Korteweg and G. de Vries. On the change of form of long waves advancing in a rectangular canal, and on a new type of long stationary waves. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, Series 5*, 39:422–443, 1895.

[57] M. D. Kruskal. The birth of the soliton. In F. Calogero, editor, *International Symposium on Nonlinear Evolution Equations Solvable by the Spectral Transform*, pages 1–8. Pitman, 1977.

[58] M. Lakshmanan, editor. *Solitons. Introduction and Applications.* Springer-Verlag, 1988.

[59] G. L. Lamb. *Elements of Soliton Theory.* John Wiley & Sons, 1980.

[60] C. Lawson, R. Hanson, D. Kincaid, and F. Krogh. Basic linear algebra subprograms for Fortran usage. *ACM Transactions on Mathematical Software*, 5:308–325, 1979.

[61] V. G. Makhankov, V. K. Fedyanin, and O. K. Pashaev, editors. *IVth International Workshop on Solitons and Applications*. World Scientific, 1990.

[62] M. F. Maritz and S. W. Schoombie. Parasitic waves and solitons in the numerical solution of the Korteweg-deVries and modified Korteweg-deVries equation. *Journal of Computational Physics*, 73:244–266, 1987.

[63] F. Mesinger and A. Arakawa. *Numerical Methods Used in Atmospheric Models*, volume 1 of *Series 17*. GARP (Global Atmospheric Research Program) Publications, 1976.

[64] J. W. Miles. The Korteweg-de Vries equation: a historical essay. *Journal of Fluid Mechanics*, 106:131–147, 1981.

[65] A. R. Mitchell and S. W. Schoombie. Finite element studies of solitons. In *Numerical Methods for Coupled Problems: Proceedings of the International Conference*, 1981.

[66] A. R. Mitchell and R. Wait. *The Finite Element Method in Partial Differential Equations*. John Wiley & Sons, 1977.

[67] R. M. Miura. The Korteweg-deVries equation: A survey of results. *Siam Review*, 18(3):412–459, 1976.

[68] R. M. Miura, C. S. Gardner, and M. D. Kruskal. Korteweg-deVries equation and generalizations. II. Existence of conservation laws and constants of motion. *Journal of Mathematical Physics*, 9(8):1204–1209, August 1968.

[69] K. W. Morton. Finite element methods for non-self-adjoint elliptic and for hyperbolic problems: Optimal approximations and recovery techniques. In D. F. Griffiths, editor, *The Mathematical Basis of Finite Element Methods*, pages 91–122. Clarendon-Oxford, 1984.

[70] A. C. Newell, editor. *Nonlinear Wave Motion*, volume 15 of *Lectures in Applied Mathematics*. American Mathematical Society, 1974.

[71] F. Z. Nouri and D. M. Sloan. A comparison of Fourier pseudospectral methods for the solution of the Korteweg-de Vries equation. *Journal of Computational Physics*, 83:324–344, 1989.

[72] P. M. Prenter. *Splines and Variational Methods*. Wiley-Interscience, 1975.

[73] L. Rayleigh. On waves. *Philosophical Magazine*, 1:251–271, 1876.

[74] L. G. Redekopp. On the theory of Rossby solitary waves. *Journal of Fluid Mechanics*, 82:725–745, 1977.

[75] R. D. Richtmyer and K. W. Morton. *Difference Methods for Initial-Value Problems.* Interscience, second edition, 1967.

[76] L. P. Røed and C. K. Cooper. Open boundary conditions in numerical ocean models. In J. J. O'Brien, editor, *Advanced Physical Oceanographic Numerical Modelling*, pages 411–436. D. Reidel Company, 1986.

[77] J. E. Romate. Absorbing boundary conditions for free surface waves. *Journal of Computational Physics*, 99:135–145, 1992.

[78] J. S. Russell. Report on waves. In *Fourteenth Meeting of the British Association for the Advancement of Science*, pages 311–390. John Murray, 1844.

[79] J. M. Sanz-Serna and I. Christie. Petrov-Galerkin methods for nonlinear dispersive waves. *Journal of Computational Physics*, 39:94–102, 1981.

[80] S. W. Schoombie. Finite element methods for the Korteweg-de Vries equation I. Galerkin method with Hermite cubics. Technical Report NA/43, University of Dundee Department of Mathematics, August 1980.

[81] S. W. Schoombie. Spline Petrov-Galerkin methods for the numerical solution of the Korteweg-de Vries equation. *IMA Jornal of Numerical Analysis*, 2:95–109, 1982.

[82] L. L. Schumaker. *Spline Functions: Basic Theory.* John Wiley & Sons, 1981.

[83] A. C. Scott, F. Y. Chu, and D. W. McLaughlin. The soliton: A new concept in applied science. *Proceedings of the IEEE*, 61:1443–1483, 1973.

[84] M. C. Shen and X. C. Zhong. Derivation of K-dV equations for water waves in a channel with variable cross section. *Journal de Mécanique*, 20(4):789–801, 1981.

[85] G. D. Smith. *Numerical Solution of Partial Differential Equations.* Clarendon-Oxford, 1978.

[86] A. H. Stroud. *Numerical Quadrature and Solution of Ordinary Differential Equations.* Springer-Verlag, 1974.

[87] A. H. Stroud and D. H. Secrest. *Gaussian Quadrature Formulas.* Prentice-Hall, 1966.

[88] T. R. Taha and M. J. Ablowitz. Analytical and numerical aspects of certain nonlinear evolution equations. III. Numerical, Korteweg-de Vries equation. *Journal of Computational Physics*, 55:231–253, 1984.

[89] T. R. Taha and M. J. Ablowitz. Analytical and numerical aspects of certain nonlinear evolution equations. I. Analytical. *Journal of Computational Physics*, 55:192–202, 1984.

[90] T. Taniuti and C. C. Wei. Reductive perterbation method in nonlinear wave propagation. I. *Journal of the Physical Society of Japan*, 24:941–946, 1968.

[91] F. Tappert. Numerical solutions of the Korteweg-de Vries equation and its generalizations by the split-step Fourier method. In A. C. Newell, editor, *Nonlinear Wave Motion*, volume 15 of *Lectures in Applied Mathematics*, pages 215–216. American Mathematical Society, 1974.

[92] E. F. G. van Daalen, J. Broeze, and E. van Groesen. Variational principles and conservation laws in the derivation of radiation boundary conditions for wave equations. *Mathematics of Computation*, 58:55–71, 1992.

[93] J. G. Verwer and J. M. Sanz-Serna. Convergence of method of lines approximations to partial differential equations. *Computing*, 33:297–313, 1984.

[94] A. C. Vliegenthart. On finite difference methods for the Korteweg-de Vries equation. *Journal of Engineering Mathematics*, 5:137–155, 1971.

[95] M. Wadati and M. Toda. The exact n-soliton solution of the Korteweg-deVries equation. *Journal of the Physical Society of Japan*, 32:1403–1411, 1972.

[96] L. B. Wahlbin. A dissipative galerkin method for the solution of first order hyperbolic equations. In C. de Boor, editor, *Mathematical Aspects of Finite Elements in Partial Differential Equations*, pages 147–169. Acedemic Press, 1974.

[97] H. Washami and T. Taniuti. Propagation of ion-acoustic solitary waves of small amplitude. *Physical Review Letters*, 17:996–998, 1966.

[98] G. B. Whitham. *Linear and Nonlinear Waves*. John Wiley & Sons, 1974.

[99] N. J. Zabusky. Computational synergetics and mathematical innovation. *Journal of Computational Physics*, 43:195–249, 1981.

[100] N. J. Zabusky and M. D. Kruskal. Interaction of "solitons" in a collisionless plasma and the recurrence of initial states. *Physical Review Letters*, 15(6):240–243, August 1965.