

University of Reading

School of Mathematics, Meteorology & Physics

The computation of spectral
representations for evolution PDE

By
Sanita Vetra

August 2007

This dissertation is submitted to the Department of Mathematics in partial fulfillment of the requirements for the degree of Master of Science.

Acknowledgments

First, I would like to thank my supervisor Beatrice Pelloni for all the help she has provided, the patience answering all my questions and the endless enthusiasm and interest in the project. I also would like to thank the rest of the teaching staff in the maths department for all their support throughout the year. And many thanks to Sue Davis who has been like a second mum to all of us this year.

Secondly, thanks to all my course mates for the interesting chats we have had in the room 107, the great times out and for the support they have given to me. It has been very enjoyable to go through this year with them alongside. I would also like to thank my friends and family for the support and understanding through this hectic year.

Lastly but most importantly, I would like to thank the EPSRC for allowing me the opportunity to study this course and fund this research.

Declaration

I confirm that this is my own work and the use of all material from other sources has been properly and fully acknowledged.

Sanita Vetra.

Abstract

In this dissertation we evaluate numerically the solution representations obtained from a recently developed Fokas integral method for solving boundary value problems for linear evolution PDEs. In particular, we consider the case of the linear KdV equation. The Fokas method is quite general and it is therefore of wider interest to assess its competitiveness for numerical purposes. Until now pseudospectral methods have been known to be the most accurate numerical scheme for smooth functions. In the work following, the linear KdV equation will be computed numerically using both a pseudospectral method and the direct evaluation of the integral representation, and comparisons will be made between these two methods for accuracy and speed of the numerical computation. The nonlinear KdV equation will be looked at using pseudospectral methods and a motivation for a possible hybrid method which would use both the Fokas and pseudospectral methods together will be given.

Contents

1	Introduction	1
1.1	Korteweg-de Vries (KdV) equation	3
1.2	Preliminary results	4
2	The Fokas Spectral Transform Method for Linear Evolution PDEs on Half Line	6
2.1	General Solution	7
2.2	Boundary conditions	12
2.2.1	Dirichlet boundary condition	13
2.2.2	Neumann boundary condition	17
3	Pseudospectral Methods	21
3.1	Non-periodic problems	23
3.1.1	Chebyshev differentiation matrices	24
3.1.2	Polynomial trick	25
3.2	Time-dependent problems	29
4	Linear Numerical Results	30
4.1	Numerical integration of the Fokas spectral transform method	30
4.1.1	Mapping from complex to real contour	31
4.1.2	Composite Trapezoidal rule	37
4.1.3	Results for the linear KdV problems for $t \in [0, 2\pi]$	38
4.1.4	Results for the linear KdV problems for $t \in [0, 15\pi]$	40
4.2	Numerical solution using pseudospectral method	41

4.2.1	Results for the linear KdV problems for $t \in [0, 2\pi]$	43
4.2.2	Results for the linear KdV problems for $t \in [0, 24\pi]$	44
4.3	Comparison between the methods	45
5	Nonlinear Numerical Results	48
5.1	Pseudospectral method using the split-step method	48
5.1.1	Non-linear step	49
5.1.2	Linear step	51
5.1.3	Numerical results from pseudospectral method	53
5.2	Pseudospectral method and Fokas integral method using the split-step method	53
6	Conclusions and Further Work	56
	Bibliography	59

List of Figures

1.1	The contour C corresponding to Jordan's Lemma.	5
2.1	The regions D^+, D_1^-, D_2^- where $Re[i(k-k^3)] < 0$	10
3.1	Chebyshev points, projections onto x axis of equally spaced points on the unit circle with $N = 27$	24
3.2	Interpolation of $u(x) = \frac{4}{5+128x^2}$ for equispaced and Chebyshev points.	24
4.1	Complex k plane, with contour of integration ∂D_+ (blue) for the linear KdV equation.	31
4.2	Mapping (red) of real θ line onto $k(\theta)$ contour along $\pi/12$ ray	33
4.3	Deformation of the integration contour (blue) to red for the first map.	33
4.4	Numerical solution for the linear KdV by simple mapping along $\pi/12$ ray	34
4.5	Mapping (red) of real θ line onto $k(\theta)$ contour by shift of the imaginary axis	34
4.6	Deformation of integration contour (blue) to red for the second map.	35
4.7	Introduction of spurious periodic solution for large x by using the shift of the imaginary axis	35
4.8	Evaluation of the two exponentials: $\sum_{x=0}^{x=40} e^{ik(\theta)x}$ and $\sum_{t=0}^{t=2\pi} e^{-i(k(\theta)-k^3(\theta))t}$ against θ	36
4.9	Numerical solution of the linear KdV with $q(x,0) = 0$ and $q(0,t) = \sin(\omega t)$ for $t \in [0, 2\pi]$, where $\omega = 1$ using the new Fokas transform method, with $\alpha = \pi/12, \gamma = 1$	38
4.10	Numerical solution of the linear KdV with $q(x,0) = xe^{-ax}$ and $q(0,t) = \sin(\omega t)$ for $t \in [0, 2\pi]$, where $\omega = 1$ using the new Fokas transform method, with $\alpha = \pi/12, \gamma = 1$	39
4.11	Numerical solution of the linear KdV with $q(x,0) = xe^{-ax}$ and $q_x(0,t) = \sin(\omega t)$ for $t \in [0, 2\pi]$, where $\omega = 1$ using the new Fokas transform method, with $\alpha = \pi/12, \gamma = 1$	39

4.12	Numerical solution of the linear KdV with $q(x,0) = 0$ and $q(0,t) = \sin(\omega t)$ for $t \in [0, 10\pi]$, where $\omega = 1$ using the new Fokas transform method, with $\alpha = \pi/12$, $\gamma = 0.53$	40
4.13	Numerical solution of the linear KdV with $q(x,0) = xe^{-ax}$ and $q(0,t) = \sin(\omega t)$ for $t \in [0, 10\pi]$, where $\omega = 1$ using the new Fokas transform method, with $\alpha = \pi/12$, $\gamma = 0.53$	40
4.14	Numerical solution of the linear KdV with $q(x,0) = xe^{-ax}$ and $q_x(0,t) = \sin(\omega t)$ for $t \in [0, 10\pi]$, where $\omega = 1$ using the new Fokas transform method, with $\alpha = \pi/12$, $\gamma = 0.53$	41
4.15	Numerical solution of the linear KdV with $q(x,0) = 0$ and $q(0,t) = \sin(\omega t)$ for $t \in [0, 2\pi]$, with $\omega = 1$	43
4.16	Numerical solution of the linear KdV with $q(x,0) = xe^{-ax}$ and $q(0,t) = \sin(\omega t)$ for $t \in [0, 2\pi]$, with $\omega = 1$	43
4.17	Numerical solution of the linear KdV with $q(x,0) = 0$ and $q(0,t) = \sin(\omega t)$ for $t \in [0, 24\pi]$, with $\omega = 1$	44
4.18	Numerical solution of the linear KdV with $q(x,0) = xe^{-ax}$ and $q(0,t) = \sin(\omega t)$ for $t \in [0, 24\pi]$, with $\omega = 1$	44
4.19	Diffusion of the soliton in numerical solution of the linear KdV with $q(x,0) = 0$ and $q(0,t) = \sin(\omega t)$ for $t \in [0, 24\pi]$, with $\omega = 1$	47
5.1	Solution to the nonlinear KdV with $u(x,0) = xe^{-x}$ and $u(0,t) = \sin(t)$, for $x \in [0, 40]$, $t \in [0, 2\pi]$ with $\Delta t = 0.01$	53
5.2	Comparison between linear and nonlinear KdV solution with $u(x,0) = xe^{-x}$ and $u(0,t) = \sin(t)$, for $x \in [0, 40]$, $t \in [0, 8\pi]$	54
5.3	Comparison between linear and nonlinear KdV solution with $u(x,0) = xe^{-x}$ and $u(0,t) = \sin(t)$, for $x \in [0, 40]$, $t \in [0, 8\pi]$	54
5.4	Comparison between linear and nonlinear KdV solution with $u(x,0) = x^2e^{-x}$ and $u(0,t) = \sin(t)$, for $x \in [0, 40]$, $t \in [0, 8\pi]$	55

Chapter 1

Introduction

In this project we consider the analytical and numerical solutions of initial boundary value problems for evolution partial differential equations (PDEs) with constant coefficients, posed on a half line. These equations are of the form

$$q_t(x,t) + Tq(x,t) = 0, \quad t > 0, \quad x \in [0, \infty) \quad (1.1)$$

where T is an x -differential operator. We prescribe initial data

$$q(x,0) = q_0(x), \quad x \in [0, \infty),$$

where $q_0(x)$ is a given smooth function, such that $q(x,t) \rightarrow 0$ as $x \rightarrow \infty$ and an appropriate number of boundary conditions are prescribed.

Equations of type (1.1) describe processes that are evolving in time from a given initial state. The simplest example is the heat or diffusion equation, $q_t = q_{xx}$, which models how heat diffuses starting from a given initial temperature.

The classical way of solving this type of problem is using a Fourier type transforms on the real line. For example, the heat equations on the half line with $q(0,t)$ given is solved by the sine transform, while the same equation but with $q_x(0,t)$ given is solved by the cosine transform. Thus, the type of transform needed for a given initial boundary value problem is specified by the PDE, by the domain, and by the given boundary conditions. For simple boundary value problems there exists an algorithmic procedure for deriving the associated transform [9]. However, for many problems the

classical transform method fails. For example, as we will see in *Chapter 2*, there does not exist a real Fourier type transform for solving third order equations on the half line.

A new transform method was introduced by Fokas in 1997 [5], where the solution was found by rewriting the PDE as a Lax pair and then performing a simultaneous spectral analysis of it. Using this approach one can express the solution in the form of an integral for all linear and integrable non-linear PDEs with spatial derivatives of arbitrary order. Moreover, the solution obtained by this new method is uniformly convergent at the boundaries and spectrally decomposed. The characterization of the solution relies on the analysis of the global relation (relation between all boundary and initial data). Later Fokas found an alternative simpler derivation of this result (only for linear PDEs), not involving spectral analysis. We will be using this simpler approach and leave it to the interested reader to look up the spectral analysis method.

This new method will be introduced in *Chapter 2* and we will illustrate it via Korteweg-de Vries (KdV) equation. In particular, we aim to solve the following four problems,

1. For the linear KdV,

$$q_t(x,t) + q_x(x,t) + q_{xxx}(x,t) = 0. \quad (1.2)$$

we consider the following combinations of initial and boundary data:

- (a) Dirichlet boundary conditions and zero initial conditions,

$$\begin{cases} q(x,0) = 0 \\ q(0,t) = \sin(wt) \end{cases} \quad w \in \mathbb{R} \quad (1.3)$$

- (b) Dirichlet boundary conditions and non-zero initial conditions,

$$\begin{cases} q(x,0) = xe^{-ax} & a \in (0,1] \\ q(0,t) = \sin(wt) & w \in \mathbb{R}. \end{cases} \quad (1.4)$$

- (c) Neumann boundary conditions and non-zero initial conditions,

$$\begin{cases} q(x,0) = x^2 e^{-ax} & a \in (0,1] \\ q_x(0,t) = \sin(wt) & w \in \mathbb{R}. \end{cases} \quad (1.5)$$

2. For the non linear KdV

$$q_t(x,t) + q_x(x,t) + q(x,t)q_x(x,t) + q_{xxx}(x,t) = 0. \quad (1.6)$$

we consider the following given data,

$$\begin{cases} q(x,0) = xe^{-ax} & a \in (0,1] \\ q(0,t) = \sin(wt) & w \in \mathbb{R}. \end{cases} \quad (1.7)$$

For the three linear cases a solution will be computed numerically both via the new Fokas method and by spectral methods, which will enable us to compare the two methods. In the case of non-linear KdV, the solution will be obtained using split step method where one step will compute solution using Fokas method and other using spectral method. This will be compared to the results obtained from purely spectral code. In *Chapter 3* an overview of the pseudospectral methods will be given. It will be shown in *Chapter 4* that the novel integral representations given by the new Fokas method are suitable for the numerical evaluation of the solution. This is possible, as using simple contour deformations in the complex k -plane, to obtain integrals involving integrands with strong decay for large k . [1] In *Chapter 4* we look at the numerical results for the problems (1.3) - (1.5) and compare the results obtained from both methods for each of the problems. Lastly, in *Chapter 5* we solve the problem (1.7) by pseudospectral methods using the Fourier split-step and give a motivation for combining the Fokas method for the linear part with pseudospectral method for the nonlinear part. All of the numerical schemes presented in this work are performed using Matlab. Codes, developed in this project, for the new Fokas integral method have been adopted from [11] and codes for pseudospectral methods have been adopted from [3]. It is possible that the codes' performance can be enhanced but this is not the focus of the present project.

1.1 Korteweg-de Vries (KdV) equation

The KdV equation is defined by,

$$q_t + q_x + qq_x + q_{xxx} = 0,$$

and it was derived in 1895 by Korteweg and de Vries to describe long wave propagation on shallow water. It is an integrable, dispersive nonlinear evolution equation and it relates the amplitude of the wave and its change in space, with the change of the amplitude in time [3]. The spatial variable x is usually assumed to be real (so there is some decay at infinity) or on torus (so the data is periodic)

[17].

The intriguing property of the KdV equation is that under certain circumstances the dispersion and nonlinearity balance each other out, thus allowing the special solutions that travel without changing their shape. Korteweg and de Vries showed that the equation posed on the real line possesses a soliton solution, which takes the form

$$q(x, t) = 3c^2 \operatorname{sech}^2 \left(\frac{cx - c^3 t}{2} \right),$$

where c is the velocity and can take any positive value. Moreover, the solitons are important because they characterize the long time behaviour of evolution equations in one space dimension. The solitons travel without changing the shape and interact linearly.

Since its discovery the KdV equation has been applied to many problems in different areas of physics, as in fluid dynamics, aerodynamics, acoustics and others as a model for shock wave formation, solitons, turbulence, boundary layer behavior, and mass transport [18].

It has been studied for decades and many numerical methods have been developed over the years, starting with explicit methods in the 1960s to pseudospectral methods. Recently some simple linear problems have been implemented numerically by exploiting the integral representation of the solution obtained from the new Fokas transform method. Here, we use this numerical approach and extend it for the linear KdV and for the nonlinear problem as well.

1.2 Preliminary results

In this section we state the definitions and the theorems used throughout this work. We have omitted the proofs as they are easily obtainable in undergraduate textbooks.

Theorem 1.2.1 (Cauchy)

If a function $f(k)$ is analytic and bounded in a simply connected domain D , then along any simple closed contour C in D

$$\oint_C f(k) dk = 0.$$

Next we present a lemma which is important for evaluating exponential integrals on open unbounded domains.

Lemma 1.2.1 (Jordan)

Let C be in circular arc, obtained by considering the intersection of the circle of radius R with the upper half complex plane \mathbb{C}^+ . Suppose that on C we have $f(k) \rightarrow 0$ uniformly as $R \rightarrow \infty$. Then

$$\lim_{R \rightarrow \infty} \int_C e^{i\lambda k} f(k) dk = 0, \quad (\lambda > 0).$$

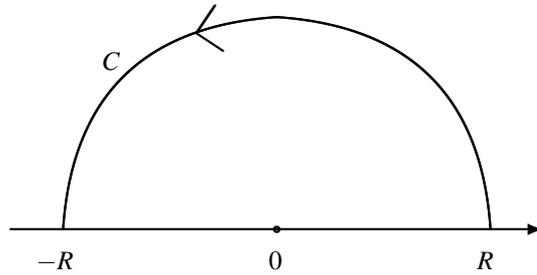


Figure 1.1: The contour C corresponding to Jordan's Lemma.

Definition 1.2.1 (The Fourier transform pair)

For a given function $f(x)$ which is continuous and infinitely differentiable, i.e. $f \in S[0, \infty)$ ¹, defined on the half line, $x \in [0, \infty)$, the Fourier transform pair is defined as follows

$$\begin{aligned} \hat{f}(k) &= \int_0^{\infty} e^{-ikx} f(x) dx \\ f(x) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{ikx} \hat{f}(k) dx \end{aligned}$$

¹Where S is a Schwarz space

Chapter 2

The Fokas Spectral Transform Method for Linear Evolution PDEs on Half Line

In this chapter we will introduce a transform method for solving initial boundary value problems for linear PDEs and integrable nonlinear PDEs with constant coefficients in one space dimension. This new method has been introduced by Fokas [5], and further developed by Fokas and Sung [2]. Although this method can only be applied in general to PDEs with constant coefficients, many physically relevant problems fall into this category. Among the equations considered we list Schrödinger equation with zero potential

$$iq_t + q_{xx} = 0, \tag{2.1}$$

the heat equation

$$q_t = q_{xx}, \tag{2.2}$$

first Stokes equation

$$q_t + q_{xxx} = 0, \tag{2.3}$$

and the linear KdV or Stokes equation

$$q_t + q_x + q_{xxx} = 0. \tag{2.4}$$

In this project we study the linear KdV equation (2.4). Hence, we shall now use this as an example to describe the general method. The heat equation and Stokes first equations follow similarly and for Schrödinger equation and/or more detail on these and other problems the reader is referred to [1] - [3], [5], [6].

2.1 General Solution

Consider the linear KdV equation (2.4) posed on the half line,

$$q_t + q_x + q_{xxx} = 0, \quad x \in [0, \infty).$$

To model an initial and boundary value problem, we need to supplement the equation with initial conditions and one boundary condition at $x = 0$. Hence, we always assume

$$q(x, 0) = q_0(x) \tag{2.5}$$

and we shall specify boundary conditions below.

Using Fourier Transform we have

$$\hat{q}(k, t) = \int_0^\infty e^{-ikx} q(x, t) dx,$$

then by integration by parts we obtain

$$\widehat{q_x}(k, t) = -q(0, t) + ik\hat{q}(k, t) \tag{2.6}$$

and

$$\widehat{q_{xxx}}(k, t) = -q_{xx}(0, t) - ikq_x(0, t) + k^2q(0, t) - ik^3\hat{q}(k, t) \tag{2.7}$$

Now substituting (2.7) and (2.6) into (2.4) gives

$$\hat{q}_t(k, t) + (ik - ik^3)\hat{q}(k, t) = q_{xx}(0, t) + ikq_x(0, t) + (1 - k^2)q(0, t). \tag{2.8}$$

Multiplying through by $e^{(ik-ik^3)t}$ and integrating with respect to t , we obtain

$$e^{(ik-ik^3)t}\hat{q}(k, t) = \int_0^t e^{(ik-ik^3)s} [q_{xx}(0, s) + ikq_x(0, s) + (1 - k^2)q(0, s)] ds + \hat{q}_0(k). \tag{2.9}$$

We can define t transforms on the left boundary by

$$\tilde{g}_j(k, t) = \int_0^t e^{(ik-ik^3)s} \partial_x^j q(0, s) ds, \quad t > 0 \text{ for } j = 0, 1, 2 \quad (2.10)$$

and

$$\tilde{g}(k, t) = \tilde{g}_2(k, t) + ik\tilde{g}_1(k, t) + (1-k^2)\tilde{g}_0(k, t), \quad k \in \mathbb{R}. \quad (2.11)$$

Thus, rearranging (2.8) we get expression for x transform of $q(x, t)$,

$$\hat{q}(k, t) = e^{-(ik-ik^3)t} \tilde{g}(k, t) + e^{-(ik-ik^3)t} \hat{q}_0(k). \quad (2.12)$$

Inverse Fourier Transform

$$q(x, t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{ikx} \hat{q}(k, t) dk$$

gives a formal integral representation of the solution of (2.4) with some initial data (2.5) on real line ($k \in \mathbb{R}$) given by

$$q(x, t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{ikx-(ik-ik^3)t} \hat{q}_0(k) dk + \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{ikx-(ik-ik^3)t} \tilde{g}(k, t) dk. \quad (2.13)$$

Recall that one of the three boundary functions can be prescribed as a boundary condition. Hence, this expression involves two unknown functions and thus it is not a representation of the solution of practical significance.

This problem arises already for second order problems such as the heat equation (2.2), if one uses the Fourier transform. However, for second order problems one could instead use sine or cosine transforms. For example, for heat equation with prescribed initial condition $q(0, t)$, it is possible to find a solution representation involving only given functions by using the transform pair

$$\begin{aligned} \hat{q}(k, t) &= \int_0^{\infty} \sin(kx) q(x, t) dx, \quad k \in \mathbb{R}, \\ q(x, t) &= \frac{2}{\pi} \int_0^{\infty} \sin(kx) \hat{q}(k, t) dk \end{aligned}$$

also (2.2) and integration by parts yields

$$\hat{q}_t(k, t) + k^2 \hat{g}(k, t) = kq(0, t).$$

Hence, the solution is therefore

$$q(x, t) = \frac{2}{\pi} \int_0^{\infty} e^{-k^2 t} \left(\hat{q}(k, 0) + \int_0^t e^{k^2 s} kq(0, s) ds \right) \sin(kx) dk \quad (2.14)$$

where $\hat{q}(k, 0) = \int_0^\infty \sin(kx)q(x, 0) dx$.

Thus, the integral representation for the solution obtained via the sine transform in this case is fully defined in terms of the data of the problem. Note however, that (2.14) is not uniformly convergent as $x \rightarrow \infty$, so we cannot compute $q(0, t)$ by just letting $x = 0$.

However, for the first Stokes equation (2.3) or the linear KdV this approach fails and it has been shown that for an odd number of spatial derivatives there are no x -transforms which could give a solution involving only known functions [7]. This is confirmed by the linear KdV for which the x -transform has the given solution (2.13) involving unknown functions.

The main difference between the new approach and the Fourier transform approach just obtained is to integrate along complex contours instead of the real line. i.e. k is no longer real but rather a complex variable. This allows the use of complex analytic techniques.

Using analyticity considerations, we obtain the integral representation in the form

$$q(x, t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{ikx - (ik - ik^3)t} \hat{q}_0(k) dk + \frac{1}{2\pi} \int_{\partial D_+} e^{ikx - (ik - ik^3)t} \tilde{g}(k, t) dk, \quad (2.15)$$

where now $k \in \mathbb{C}$ and where ∂D_+ is the contour in the upper half complex k -plane where the exponential $e^{i(k-k^3)}$ is purely oscillatory. The domain D is defined by

$$D = \{k \in \mathbb{C} : \text{Re}(ik - ik^3) \leq 0\}, \quad D^\pm = D \cap \mathbb{C}^\pm$$

and D has boundaries given by ∂D_\pm , where the interior of the domain is always to the left, see Fig. (2.1).

We now sketch the proof of the identity between (2.13) and the Fourier representation (2.15). We note that the first term on the right hand side of equation (2.13) is identical to the first term in (2.15). The equality of the second terms is due to analyticity considerations and Cauchy's theorem. Indeed the term

$$e^{ikx} \int_0^t e^{-i(k-k^3)(t-s)} [q_{xx}(0, t) + ikq_x(0, t) + (1-k^2)q(0, t)] ds \quad (2.16)$$

is analytic and bounded in the region of complex k -plane, where $\text{Im}(k) > 0$ and $\text{Re}(ik - ik^3) > 0$ and $t - s > 0$. Hence, the integral of this term along the real axis can be deformed to the integral along the

contour ∂D_+ , i.e. the integral along the rays $(-\infty, -1/\sqrt{3}] \cup [1/\sqrt{3}, \infty)$ can be deformed along the curves (in blue) shown in Fig.(2.1). These curves are the two branches of the curve $Re(ik - ik^3) = 0$, $Im(k) > 0$.

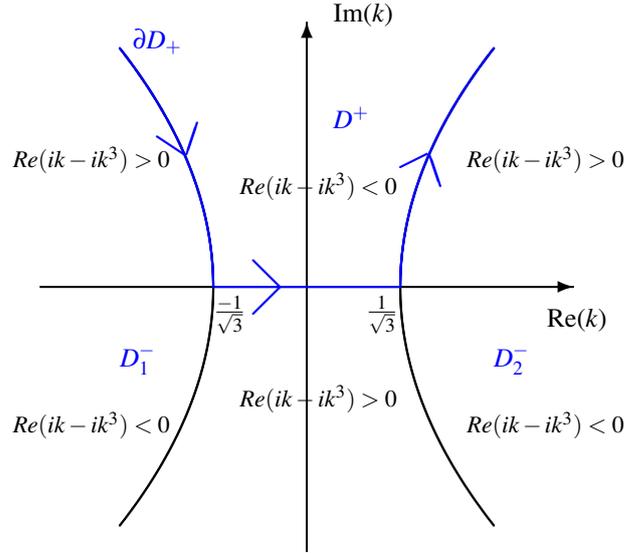


Figure 2.1: The regions D^+ , D_1^- , D_2^- where $Re[i(k - k^3)] < 0$.

However, equation (2.15) still involves the unknown boundary data, but now we are integrating over the complex plane along the contour of D^+ , ∂D_+ , see Fig.(2.1). This gives us much more freedom and we shall show that in this wider setting it is always possible to express unknown functions in terms of the known one(s) for all linear and integrable nonlinear PDEs.

The important relation that yields the characterisation of the unknown boundary function is,

$$e^{(ik-ik^3)t} \hat{q}(k,t) = \tilde{g}_2(k,t) + ik\tilde{g}_1(k,t) + (1-k^2)\tilde{g}_0(k,t) + \hat{q}_0(k) \quad Im(k) \leq 0 \quad (2.17)$$

This is called by Fokas the global relation, and it is a constraint linking all initial and boundary values. Analyticity considerations show that the contribution of the term involving $\hat{q}(k,t)$ to the final integral representation of the solution is zero. Hence, we can neglect the contribution of this term

and consider a simplified global relation,

$$(k^2 - 1)\tilde{g}_0(k, t) = \tilde{g}_2(k, t) + ik\tilde{g}_1(k, t) + \hat{q}_0(k) \quad \text{Im}(k) \leq 0. \quad (2.18)$$

We consider this relation for $k \in \partial D_+$, as this is the contour at which we need to characterise the unknown functions. Note that this contour is the boundary of the domain D_+ . Also, that the global relation is only well defined if $\text{Im}(k) \leq 0$. This is due to the fact that the term $\hat{q}_0(k)$ contains the exponential e^{-ikx} , $x > 0$, which is unbounded as x grows if $\text{Im}(k) > 0$. Indeed,

$$\begin{aligned} e^{-ikx} &= e^{-i[\text{Re}(k) + i\text{Im}(k)]x} \\ &= e^{-i\text{Re}(k)x} e^{\text{Im}(k)x} \end{aligned}$$

and since $|e^{-i\text{Re}(k)x}| = 1$, we have that

$$|e^{-ikx}| = |e^{\text{Im}(k)x}|. \quad (2.19)$$

Thus,

$$|e^{-ikx}| \rightarrow \begin{cases} \infty, & \text{Im}(k) > 0 \\ 0, & \text{Im}(k) < 0 \end{cases} \quad (2.20)$$

and is therefore bounded in the lower half of the complex- k plane. Thus, the global relations (2.17) and (2.18) are also defined for $\text{Im}(k) \leq 0$. This is crucial for what follows.

We still need to show that the representation (2.15) can be expressed only in terms of the known data of the problem. For this we need to show that it is possible to compute the term $\tilde{g}(k, t)$ for k on the contour ∂D_+ in the upper half complex k -plane, in terms of the given initial and boundary conditions. It would seem then that the global relation is not useful, firstly because it is defined in the lower half complex k -plane, and secondly as it is one equation involving two unknown function. However, we can transform (2.18) from lower complex k -plane to the domain D_+ . This is easily done by observing that $\tilde{g}_j(k, t)$, $j = 0, 1, 2$, depend on k only via the $w(k) = ik - ik^3$ expression. Hence, $\tilde{g}_j(k, t)$, $j = 0, 1, 2$, are invariant by those transformations $k \rightarrow v(k)$, which preserve $w(k)$. For the linear KdV equation we have $w(k) = ik - ik^3 = w(v(k))$, which gives

$$v - v^3 = k - k^3.$$

This has a trivial root $v(k) = k$, hence we have

$$(v(k) - k)(-v^2 - kv - k^2 + 1) = 0$$

and since we don't want the trivial root, solve $(v^2 + kv + k^2 - 1) = 0$, which gives

$$\begin{aligned} v_{1,2} &= \frac{-k \pm \sqrt{k^2 - 4(k^2 - 1)}}{2} \\ &= \frac{-k \pm \sqrt{-3k^2 + 4}}{2} \\ &= \frac{-k \pm i\sqrt{3k^2 - 4}}{2}. \end{aligned}$$

If $v_1 \in D_1^-$ and $v_2 \in D_2^-$ then $k \in D_+$. Thus, replacing k by $v_j(k)$ in equation (2.18) we find two algebraic equations

$$\begin{aligned} (v_1^2 - 1)\tilde{g}_0(v_1, t) &= \tilde{g}_2(v_1, t) + iv_1\tilde{g}_1(v_1, t) + \hat{q}_0(v_1) \\ (v_2^2 - 1)\tilde{g}_0(v_2, t) &= \tilde{g}_2(v_2, t) + iv_2\tilde{g}_1(v_2, t) + \hat{q}_0(v_2) \end{aligned}$$

and since $\tilde{g}_j(k, t)$ are invariant to $v_i(k)$ transformation, then we have

$$(v_1^2 - 1)\tilde{g}_0(k, t) = \tilde{g}_2(k, t) + iv_1\tilde{g}_1(k, t) + \hat{q}_0(v_1) \quad (2.21)$$

$$(v_2^2 - 1)\tilde{g}_0(k, t) = \tilde{g}_2(k, t) + iv_2\tilde{g}_1(k, t) + \hat{q}_0(v_2) \quad (2.22)$$

which are valid in D_+ . Thus, we are left with two equations and two unknowns which are simple to solve for given boundary data.

2.2 Boundary conditions

Two types of boundary data will be considered here: Dirichlet and Neumann for which general solutions will be derived.

2.2.1 Dirichlet boundary condition

Let,

$$q(0, t) = f(t) \quad (2.23)$$

for which our Fourier Transform is $\tilde{g}_0(k, t) = \int_0^t e^{(ik-ik^3)s} q(0, s) ds = \int_0^t e^{(ik-ik^3)s} f(s) ds$.

Here, we can express terms $\tilde{g}_1(k, t)$ and $\tilde{g}_2(k, t)$ in terms of $\tilde{g}_0(k, t)$ using equations (2.21) and (2.22).

So (2.21) - (2.22) gives

$$(\mathbf{v}_1^2 - \mathbf{v}_2^2)\tilde{g}_0(k, t) = i(\mathbf{v}_1 - \mathbf{v}_2)\tilde{g}_1(k, t) + \hat{q}_0(\mathbf{v}_1) - \hat{q}_0(\mathbf{v}_2)$$

and so,

$$\begin{aligned} \tilde{g}_1(k, t) &= \frac{\mathbf{v}_1^2 - \mathbf{v}_2^2}{i(\mathbf{v}_1 - \mathbf{v}_2)} \tilde{g}_0(k, t) + \frac{\hat{q}_0(\mathbf{v}_2) - \hat{q}_0(\mathbf{v}_1)}{i(\mathbf{v}_1 - \mathbf{v}_2)} \\ &= \frac{\mathbf{v}_1 + \mathbf{v}_2}{i} \tilde{g}_0(k, t) + \frac{\hat{q}_0(\mathbf{v}_2) - \hat{q}_0(\mathbf{v}_1)}{i(\mathbf{v}_1 - \mathbf{v}_2)} \\ &= \frac{-k}{i} \tilde{g}_0(k, t) + \frac{\hat{q}_0(\mathbf{v}_2) - \hat{q}_0(\mathbf{v}_1)}{i(\mathbf{v}_1 - \mathbf{v}_2)} \end{aligned} \quad (2.24)$$

and (2.21) - $\frac{\mathbf{v}_1}{\mathbf{v}_2}$ (2.22) gives

$$\left(\mathbf{v}_1^2 - 1 - \mathbf{v}_1\mathbf{v}_2 + \frac{\mathbf{v}_1}{\mathbf{v}_2}\right) \tilde{g}_0(k, t) = \left(1 - \frac{\mathbf{v}_1}{\mathbf{v}_2}\right) \tilde{g}_2(k, t) + \hat{q}_0(\mathbf{v}_1) - \frac{\mathbf{v}_1}{\mathbf{v}_2} \hat{q}_0(\mathbf{v}_2)$$

and so,

$$\begin{aligned} (\mathbf{v}_1^2\mathbf{v}_2 - \mathbf{v}_2 - \mathbf{v}_1\mathbf{v}_2^2 + \mathbf{v}_1) \tilde{g}_0(k, t) &= (\mathbf{v}_2 - \mathbf{v}_1) \tilde{g}_2(k, t) + \mathbf{v}_2\hat{q}_0(\mathbf{v}_1) - \mathbf{v}_1\hat{q}_0(\mathbf{v}_2) \\ (\mathbf{v}_1(\mathbf{v}_2\mathbf{v}_1 + 1) - \mathbf{v}_2(\mathbf{v}_2\mathbf{v}_1 + 1)) \tilde{g}_0(k, t) &= (\mathbf{v}_2 - \mathbf{v}_1) \tilde{g}_2(k, t) + \mathbf{v}_2\hat{q}_0(\mathbf{v}_1) - \mathbf{v}_1\hat{q}_0(\mathbf{v}_2) \\ (\mathbf{v}_2\mathbf{v}_1 + 1)(\mathbf{v}_1 - \mathbf{v}_2) \tilde{g}_0(k, t) &= (\mathbf{v}_2 - \mathbf{v}_1) \tilde{g}_2(k, t) + \mathbf{v}_2\hat{q}_0(\mathbf{v}_1) - \mathbf{v}_1\hat{q}_0(\mathbf{v}_2) \end{aligned}$$

hence,

$$\begin{aligned} \tilde{g}_2(k, t) &= -(\mathbf{v}_1\mathbf{v}_2 + 1)\tilde{g}_0(k, t) + \frac{\mathbf{v}_1\hat{q}_0(\mathbf{v}_2) - \mathbf{v}_2\hat{q}_0(\mathbf{v}_1)}{\mathbf{v}_2 - \mathbf{v}_1} \\ &= -\left(\frac{1}{4}(k^2 + 3k^2 - 4) + 1\right) \tilde{g}_0(k, t) + \frac{\mathbf{v}_1\hat{q}_0(\mathbf{v}_2) - \mathbf{v}_2\hat{q}_0(\mathbf{v}_1)}{\mathbf{v}_2 - \mathbf{v}_1} \\ &= -k^2\tilde{g}_0(k, t) + \frac{\mathbf{v}_1\hat{q}_0(\mathbf{v}_2) - \mathbf{v}_2\hat{q}_0(\mathbf{v}_1)}{\mathbf{v}_2 - \mathbf{v}_1} \end{aligned} \quad (2.25)$$

Substituting (2.24) and (2.25) into (2.11) we have

$$\begin{aligned} \tilde{g}(k,t) = & -k^2 \tilde{g}_0(k,t) + \frac{v_1 \hat{q}_0(v_2) - v_2 \hat{q}_0(v_1)}{v_2 - v_1} + \\ & + ik \left(\frac{-k}{i} \tilde{g}_0(k,t) + \frac{\hat{q}_0(v_2) - \hat{q}_0(v_1)}{i(v_1 - v_2)} \right) + (1 - k^2) \tilde{g}_0(k,t) \end{aligned}$$

which simplifies to

$$\tilde{g}(k,t) = (1 - 3k^2) \tilde{g}_0(k,t) + \hat{q}_0(v_1) \frac{k - v_2}{v_2 - v_1} + \hat{q}_0(v_2) \frac{v_1 - k}{v_2 - v_1} \quad (2.26)$$

and substituting (2.26) into (2.15) gives

$$\begin{aligned} q(x,t) = & \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{ikx - (ik - ik^3)t} \hat{q}_0(k) dk + \\ & + \frac{1}{2\pi} \int_{\partial D_+} e^{ikx - (ik - ik^3)t} (1 - 3k^2) \tilde{g}_0(k,t) + \\ & + \frac{1}{2\pi} \int_{\partial D_+} e^{ikx - (ik - ik^3)t} \left(\hat{q}_0(v_1) \frac{k - v_2}{v_2 - v_1} + \hat{q}_0(v_2) \frac{v_1 - k}{v_2 - v_1} \right) dk \end{aligned} \quad (2.27)$$

which is the solution to the linear KdV (2.4) subject to initial conditions (2.5) and Dirichlet boundary conditions (2.23).

Example 1: Linear KdV with Dirichlet BC and zero IC

Now we find a solution to the particular linear KdV problem with initial and boundary data defined as follows,

$$q(x,0) = 0 \quad (2.28)$$

$$q(0,t) = \sin(wt). \quad (2.29)$$

Fourier Transforms of the initial condition is

$$\begin{aligned} \hat{q}_0(k) &= \int_0^{\infty} e^{-ikx} q(x,0) dx \\ &= 0 \end{aligned} \quad (2.30)$$

and of boundary condition is,

$$\tilde{g}_0(k,t) = \int_0^t e^{(ik - ik^3)s} q(0,s) ds$$

$$\begin{aligned}
 &= \int_0^t e^{(ik-ik^3)s} \sin(ws) ds \\
 &= \left[-\frac{1}{w} \cos(ws) e^{(ik-ik^3)s} \right]_0^t + \frac{ik-ik^3}{w} \int_0^t e^{(ik-ik^3)s} \cos(ws) ds \\
 &= -\frac{1}{w} \cos(wt) e^{(ik-ik^3)t} + \frac{1}{w} + \frac{ik-ik^3}{w} \left(\left[\frac{1}{w} \sin(ws) e^{(ik-ik^3)s} \right]_0^t - \frac{ik-ik^3}{w} \tilde{g}_0(k,t) \right)
 \end{aligned}$$

so,

$$\tilde{g}_0(k,t) \left(\frac{w^2 + (ik-ik^3)^2}{w^2} \right) = -\frac{1}{w} \cos(wt) e^{(ik-ik^3)t} + \frac{1}{w} + \frac{ik-ik^3}{w^2} \sin(wt) e^{(ik-ik^3)t}$$

and

$$\begin{aligned}
 \tilde{g}_0(k,t) &= \left(\frac{w^2}{w^2 + (ik-ik^3)^2} \right) \left(-\frac{1}{w} \cos(wt) e^{(ik-ik^3)t} + \frac{1}{w} + \frac{ik-ik^3}{w^2} \sin(wt) e^{(ik-ik^3)t} \right) \\
 &= e^{(ik-ik^3)t} \left(\frac{e^{-(ik-ik^3)t} w - w \cos(wt) + (ik-ik^3) \sin(wt)}{w^2 - (k-k^3)^2} \right).
 \end{aligned}$$

Using identities $\sin(wt) = \frac{e^{iwt} - e^{-iwt}}{2i}$ and $\cos(wt) = \frac{e^{iwt} + e^{-iwt}}{2}$, adding and subtracting extra term and rearranging yields

$$\begin{aligned}
 \tilde{g}_0(k,t) &= \frac{e^{(ik-ik^3)t}}{2} \left(\frac{2we^{-(ik-ik^3)t} - w(e^{iwt} + e^{-iwt}) + (k-k^3)(e^{iwt} - e^{-iwt})}{(w - (k-k^3))(w + (k-k^3))} \right) \pm \\
 &\quad \pm \frac{e^{(ik-ik^3)t}}{2} \left(\frac{(k-k^3)e^{-(ik-ik^3)t}}{(w - (k-k^3))(w + (k-k^3))} \right) \\
 &= \frac{e^{(ik-ik^3)t}}{2} \left(\frac{e^{-(ik-ik^3)t}(w + (k-k^3)) - e^{-iwt}(w + (k-k^3))}{(w - (k-k^3))(w + (k-k^3))} \right) + \\
 &\quad + \left(\frac{e^{-(ik-ik^3)t}(w - (k-k^3)) - e^{iwt}(w - (k-k^3))}{(w - (k-k^3))(w + (k-k^3))} \right)
 \end{aligned}$$

and so

$$\tilde{g}_0(k,t) = \frac{e^{(ik-ik^3)t}}{2} \left(\frac{e^{-(ik-ik^3)t} - e^{-iwt}}{(w - (k-k^3))} + \frac{e^{-(ik-ik^3)t} - e^{iwt}}{(w + (k-k^3))} \right) \quad (2.31)$$

Hence, using general Dirichlet solution (2.27) obtained above and substituting values of initial (2.30)

and boundary (2.31) data we have

$$q(x,t) = \frac{1}{4\pi} \int_{\partial D_+} e^{ikx} (1-3k^2) \left(\frac{e^{-(ik-ik^3)t} - e^{-iwt}}{(w - (k - k^3))} + \frac{e^{-(ik-ik^3)t} - e^{iwt}}{(w + (k - k^3))} \right) dk \quad (2.32)$$

which is an integral representation of the solution of (2.4) subject to (2.28) and 2.29).

Example 2: Linear KdV with Dirichlet BC and non-zero IC

Here, consider Dirichlet problem with non-zero initial conditions, defined as follows,

$$q(x,0) = xe^{-ax}, \quad a \in \mathbb{R}^+ \quad (2.33)$$

and the same boundary data as for Example 1.

Fourier Transforms of the initial condition is

$$\begin{aligned} \hat{q}_0(k) &= \int_0^\infty e^{-ikx} q(x,0) dx = \int_0^\infty xe^{-(ik+a)x} dx \\ &= \left[\frac{e^{-(ik+a)x}}{-(ik+a)} x \right]_0^\infty + \frac{1}{(ik+a)} \int_0^\infty e^{-(ik+a)x} dx \\ &= \frac{1}{(ik+a)} \left[\frac{e^{-(ik+a)x}}{-(ik+a)} \right]_0^\infty \\ &= \frac{1}{(ik+a)^2} \end{aligned} \quad (2.34)$$

and of the boundary condition is the same as for Example 1,

$$\tilde{g}_0(k,t) = \frac{e^{(ik-ik^3)t}}{2} \left(\frac{e^{-(ik-ik^3)t} - e^{-iwt}}{(w - (k - k^3))} + \frac{e^{-(ik-ik^3)t} - e^{iwt}}{(w + (k - k^3))} \right).$$

Hence, using the general Dirichlet solution (2.41) obtained above and substituting values of initial (2.34) and boundary (2.31) data we have

$$\begin{aligned} q(x,t) &= \frac{1}{2\pi} \int_{-\infty}^\infty e^{ikx - (ik-ik^3)t} \frac{1}{(ik+a)^2} dk + \\ &+ \frac{1}{4\pi} \int_{\partial D_+} e^{ikx} (1-3k^2) \left(\frac{e^{-(ik-ik^3)t} - e^{-iwt}}{(w - (k - k^3))} + \frac{e^{-(ik-ik^3)t} - e^{iwt}}{(w + (k - k^3))} \right) dk + \\ &+ \frac{1}{2\pi} \int_{\partial D_+} e^{ikx - (ik-ik^3)t} \left(\frac{k - v_2}{(iv_1 + a)^2 (v_2 - v_1)} + \frac{v_1 - k}{(iv_2 + a)^2 (v_2 - v_1)} \right) dk \end{aligned} \quad (2.35)$$

Lastly, by expanding and simplifying the algebraic expression in the third part of the (2.35), yields the integral representation to the linear KdV equation with initial and boundary data given by (2.33) and (2.29), respectively,

$$\begin{aligned}
 q(x,t) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{ikx-(ik-ik^3)t} \frac{1}{(ik+a)^2} dk + \\
 &+ \frac{1}{4\pi} \int_{\partial D_+} e^{ikx} (1-3k^2) \left(\frac{e^{-(ik-ik^3)t} - e^{-iwt}}{(w-(k-k^3))} + \frac{e^{-(ik-ik^3)t} - e^{iwt}}{(w+(k-k^3))} \right) dk + \\
 &+ \frac{1}{2\pi} \int_{\partial D_+} e^{ikx-(ik-ik^3)t} \frac{k^2 + 4ika - a^2 + 1}{(-k^2 - ika + a^2 + 1)^2} dk
 \end{aligned} \tag{2.36}$$

2.2.2 Neumann boundary condition

Let,

$$q_x(0,t) = h(t) \tag{2.37}$$

for which our Fourier Transform is

$$\begin{aligned}
 \tilde{g}_1(k,t) &= \int_0^t e^{(ik-ik^3)s} q_x(0,s) ds \\
 &= \int_0^t e^{(ik-ik^3)s} h(s) ds.
 \end{aligned}$$

As for Dirichlet case we can express terms $\tilde{g}_0(k,t)$ and $\tilde{g}_2(k,t)$ in terms of $\tilde{g}_1(k,t)$ using equations (2.21) and (2.22). As before (2.21) - (2.22) gives

$$(\mathbf{v}_1^2 - \mathbf{v}_2^2) \tilde{g}_0(k,t) = i(\mathbf{v}_1 - \mathbf{v}_2) \tilde{g}_1(k,t) + \hat{q}_0(\mathbf{v}_1) - \hat{q}_0(\mathbf{v}_2)$$

and this yields,

$$\begin{aligned}
 \tilde{g}_0(k,t) &= \frac{i(\mathbf{v}_1 - \mathbf{v}_2)}{\mathbf{v}_1^2 - \mathbf{v}_2^2} \tilde{g}_1(k,t) + \frac{\hat{q}_0(\mathbf{v}_1) - \hat{q}_0(\mathbf{v}_2)}{\mathbf{v}_1^2 - \mathbf{v}_2^2} \\
 &= \frac{i}{\mathbf{v}_1 + \mathbf{v}_2} \tilde{g}_1(k,t) + \frac{\hat{q}_0(\mathbf{v}_1) - \hat{q}_0(\mathbf{v}_2)}{\mathbf{v}_1^2 - \mathbf{v}_2^2} \\
 &= \frac{i}{-k} \tilde{g}_1(k,t) + \frac{\hat{q}_0(\mathbf{v}_1) - \hat{q}_0(\mathbf{v}_2)}{\mathbf{v}_1^2 - \mathbf{v}_2^2}.
 \end{aligned} \tag{2.38}$$

Here, (2.21) - $\frac{v_1^2-1}{v_2^2-1}$ (2.22) gives

$$\begin{aligned} 0 &= \left(1 - \frac{v_1^2-1}{v_2^2-1}\right) \tilde{g}_2(k,t) + i \left(v_1 - v_2 \frac{v_1^2-1}{v_2^2-1}\right) \tilde{g}_1(k,t) + \hat{q}_0(v_1) - \frac{v_1^2-1}{v_2^2-1} \hat{q}_0(v_2) \\ 0 &= ((v_2^2-1) - (v_1^2-1)) \tilde{g}_2(k,t) + i(v_1(v_2^2-1) - v_2(v_1^2-1)) \tilde{g}_1(k,t) - \\ &\quad + (v_2^2-1)\hat{q}_0(v_1) - (v_1^2-1)\hat{q}_0(v_2) \end{aligned}$$

from which we obtain,

$$\tilde{g}_2(k,t) = ik\tilde{g}_1(k,t) + \frac{(v_2^2-1)\hat{q}_0(v_1) - (v_1^2-1)\hat{q}_0(v_2)}{(v_1^2-v_2^2)}. \quad (2.39)$$

Hence, substituting (2.38) and (2.39) into (2.11) we have

$$\begin{aligned} \tilde{g}(k,t) &= ik\tilde{g}_1(k,t) + \frac{(v_2^2-1)\hat{q}_0(v_1) - (v_1^2-1)\hat{q}_0(v_2)}{(v_1^2-v_2^2)} + ik\tilde{g}_1(k,t) + \\ &\quad + (1-k^2) \left(\frac{i}{-k} \tilde{g}_1(k,t) + \frac{\hat{q}_0(v_1) - \hat{q}_0(v_2)}{v_1^2 - v_2^2} \right) \end{aligned}$$

which simplifies to

$$\tilde{g}(k,t) = \left(3ik - \frac{i}{k}\right) \tilde{g}_1(k,t) + \hat{q}_0(v_1) \frac{v_2^2 - k^2}{v_1^2 - v_2^2} + \hat{q}_0(v_2) \frac{k^2 - v_1^2}{v_1^2 - v_2^2} \quad (2.40)$$

and substituting (2.40) into (2.15) yields

$$\begin{aligned} q(x,t) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{ikx - (ik - ik^3)t} \hat{q}_0(k) dk + \\ &\quad + \frac{1}{2\pi} \int_{\partial D_+} e^{ikx - (ik - ik^3)t} \left(3ik - \frac{i}{k}\right) \tilde{g}_1(k,t) + \\ &\quad + \frac{1}{2\pi} \int_{\partial D_+} e^{ikx - (ik - ik^3)t} \left(\hat{q}_0(v_1) \frac{v_2^2 - k^2}{v_1^2 - v_2^2} + \hat{q}_0(v_2) \frac{k^2 - v_1^2}{v_1^2 - v_2^2} \right) dk \end{aligned} \quad (2.41)$$

which is the solution to the linear KdV (2.4) subject to initial conditions (2.5) and Dirichlet boundary conditions (2.37).

Example 3: Linear KdV with Neumann BC and non-zero IC

Here, we consider a Neumann problem with non-zero initial conditions. The case with zero initial conditions is obvious as seen from the two Dirichlet problems above. Let,

$$q(x, 0) = x^2 e^{-ax}, \quad a \in \mathbb{R}^+ \quad (2.42)$$

$$q_x(0, t) = \sin(\omega t). \quad (2.43)$$

Fourier Transforms of the initial condition is

$$\begin{aligned} \hat{q}_0(k) &= \int_0^\infty e^{-ikx} q(x, 0) dx = \int_0^\infty x^2 e^{-(ik+a)x} dx \\ &= \left[\frac{e^{-(ik+a)x}}{-(ik+a)} x^2 \right]_0^\infty + \frac{2}{(ik+a)} \int_0^\infty x e^{-(ik+a)x} dx \\ &= \left[\frac{e^{-(ik+a)x}}{-(ik+a)} x \right]_0^\infty + \frac{2}{(ik+a)^2} \int_0^\infty e^{-(ik+a)x} dx \\ &= \frac{2}{(ik+a)^2} \left[\frac{e^{-(ik+a)x}}{-(ik+a)} \right]_0^\infty = \frac{2}{(ik+a)^3} \end{aligned} \quad (2.44)$$

and of the boundary condition is the same as for the case 1, but here it is $\tilde{g}_1(k, t)$ which is provided by given data,

$$\begin{aligned} \tilde{g}_1(k, t) &= \int_0^t e^{(ik-ik^3)s} q_x(0, s) ds \\ &= \dots \\ &= \frac{e^{(ik-ik^3)t}}{2} \left(\frac{e^{-(ik-ik^3)t} - e^{-i\omega t}}{(w - (k - k^3))} + \frac{e^{-(ik-ik^3)t} - e^{i\omega t}}{(w + (k - k^3))} \right). \end{aligned} \quad (2.45)$$

Thus, again substituting the equations for initial and boundary data (2.45) and (2.44) into the general solution (2.41) yields,

$$\begin{aligned} q(x, t) &= \frac{1}{\pi} \int_{-\infty}^\infty e^{ikx - (ik - ik^3)t} \frac{1}{(ik+a)^3} dk + \\ &+ \frac{1}{4\pi} \int_{\partial D_+} e^{ikx} \left(3ik - \frac{i}{k} \right) \left(\frac{e^{-(ik-ik^3)t} - e^{-i\omega t}}{(w - (k - k^3))} + \frac{e^{-(ik-ik^3)t} - e^{i\omega t}}{(w + (k - k^3))} \right) dk + \\ &+ \frac{1}{2\pi} \int_{\partial D_+} e^{ikx - (ik - ik^3)t} \left(\hat{q}_0(v_1) \frac{v_2^2 - k^2}{v_1^2 - v_2^2} + \hat{q}_0(v_2) \frac{k^2 - v_1^2}{v_1^2 - v_2^2} \right) dk \end{aligned} \quad (2.46)$$

and, by expanding and simplifying the algebraic expression in the third part of the (2.46), gives the

integral representation to the linear KdV equation with Neumann boundary data,

$$\begin{aligned}
 q(x,t) = & \frac{1}{\pi} \int_{-\infty}^{\infty} e^{ikx-(ik-ik^3)t} \frac{1}{(ik+a)^3} dk + \\
 & + \frac{1}{4\pi} \int_{\partial D_+} e^{ikx} \left(3ik - \frac{i}{k} \right) \left(\frac{e^{-(ik-ik^3)t} - e^{-iwt}}{(w-(k-k^3))} + \frac{e^{-(ik-ik^3)t} - e^{iwt}}{(w+(k-k^3))} \right) dk + \\
 & + \frac{1}{\pi} \int_{\partial D_+} e^{ikx-(ik-ik^3)t} \left(\frac{ik^4 + 3ia^2 - ka^3 - 6k^3a + 6ka - 3ik^2a^2 - i}{k(-k^2 - ika + a^2 + 1)^3} \right) dk +
 \end{aligned}$$

Chapter 3

Pseudospectral Methods

Spectral methods were introduced in the 1970s and they have become widely popular for their accuracy and convergence speed when applied to smooth functions in comparison to the finite difference (FD) and finite element (FE) methods.

In this chapter we give an overview of pseudospectral methods, concentrating on the case of non-periodic problems, in particular for third order equations.

The essential difference between spectral as opposed to finite difference/finite element approximations is that the latter approximates functions locally, using only the information available in some neighborhood of the point at which the approximation is computed. In contrast, spectral methods use all the information on the domain and thus approximate a solution globally.

For finite differences the most accurate schemes are those where the interpolating polynomial is centered on the grid point of interest, e.g. three and five point quadratic formulas with errors of magnitude $O(h^2)$ and $O(h^4)$, respectively. In contrast pseudospectral methods can be viewed as the limit of N -point formulas, with error of order $O(h^N)$. Thus, as N is increased error in pseudospectral methods is rapidly decreased as h becomes smaller and unlike FD and FE methods, the order in pseudospectral methods is not fixed. Now since h is $O(1/N)$, then

$$\text{Pseudospectral error} = O\left(\frac{1}{N^N}\right).$$

Moreover, not only does the error in pseudospectral methods decreases faster than any finite power

of N thus making it much more accurate than finite elements or finite difference methods, but also because of the high accuracy spectral methods are memory minimizing, too. Thus, if high accuracy is required spectral methods is the ideal tool, however problems arise for spectral methods when complex geometries are used and finite element methods should be considered instead.

Spectral methods are usually described as expansions based on global functions. The idea is to approximate a solution $u(x)$ by a finite sum of $N + 1$ basis functions

$$u(x) \approx u_N(x) = \sum_{k=0}^N a_k \phi_k(x),$$

where a_k are weights or expansion coefficients and $\phi_k(x)$ are basis functions such as trigonometric functions or Chebyshev polynomials, for example. Substituting this series into the equation

$$Lu = f(x),$$

where L is a differential operator, we obtain the residual function defined by

$$R(x; a_0, a_1, \dots, a_N) = Lu_N - f,$$

and since the residual function is zero for the exact solution, then the goal is to choose the series coefficients $\{a_n\}$ so that the residual is minimized. The different spectral methods and pseudospectral methods differ mainly in their minimization strategies [12].

Pseudospectral methods require that the differential equation is exactly satisfied at interpolation or collocation points. Thus, as $R(x, ; a_n)$ is forced to vanish at increasingly more number of points, it will become smaller and smaller in between the interpolation points. Thus, $u_N(x)$ will converge to $u(x)$ as N increases.

It is important to choose the right set of basis functions which must satisfy three requirements listed in [14], namely:

1. The approximations $\sum_{k=0}^N a_k \phi_k(x)$ of $u(x)$ must converge rapidly (at least for reasonably smooth functions);
2. Given coefficients a_k , it should be easy to determine b_k such that

$$\frac{d}{dx} \left(\sum_{k=0}^N a_k \phi_k(x) \right) = \sum_{k=0}^N b_k \phi_k(x),$$

3. It should be fast to convert between coefficients a_k , $k = 0, \dots, N$, and the values for the sum $u(x_i)$ at some set of nodes x_i , $i = 0, \dots, N$. It turns out that $u_N = I_N u$ is precisely the interpolant of u on the given nodes. Thus, spectral methods can also be described in terms of interpolation properties.

That is basis sets need to be easy to compute, have fast convergence and completeness. The choice of the basis functions is determined by the type of problem at hand, for periodic functions trigonometric functions satisfy the above criteria and for non-periodic functions one resorts to using Chebyshev or Legendre polynomials; we will concentrate on Chebyshev polynomials referring the reader to [12] or [14] for background on other types of polynomials.

3.1 Non-periodic problems

Consider a non-periodic smooth function defined on $[-1, 1]$. In general, when a smooth function is extended it becomes non-smooth and if trigonometric interpolation in equispaced points is used, then the above given criteria for $\phi_k(x)$ will not be satisfied. Thus, spectral accuracy will be lost and hence, the main reason for using spectral methods.

Instead, we must replace trigonometric polynomials with algebraic polynomials,

$$p(x) = a_0 + a_1(x) + \dots + a_N(x)$$

on unevenly spaced points. We will use Chebyshev points, which are defined by

$$x_j = \cos\left(\frac{j\pi}{N}\right), \quad j = 0, 1, \dots, N, \quad (3.1)$$

and are the projections onto the interval $[-1, 1]$, of equispaced points along the unit circle in the complex plane, see Fig. 3.1. Chebyshev polynomials are defined as

$$T_n(\cos\theta) \equiv \cos(n\theta). \quad (3.2)$$

We can see how the use of these points increases the accuracy of polynomial interpolant. For example, if we interpolate $u(x) = \frac{4}{5+128x^2}$ on a 21 point grid, then the maximum error on an equispaced grid is 72.4399, but using Chebyshev points the maximum error is only 0.014837. The difference in

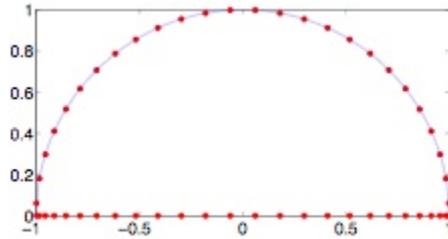


Figure 3.1: Chebyshev points, projections onto x axis of equally spaced points on the unit circle with $N = 27$.

easily seen in Fig. (3.2), where the interpolant on the left is oscillating near the ends of the interval (Gibbs phenomena see [10]). However, Chebyshev points approximate function $u(x)$ very well. By increasing the number of grid points accuracy for equispaced points decreases exponentially with oscillations escalating, whereas for Chebyshev points accuracy increases exponentially.

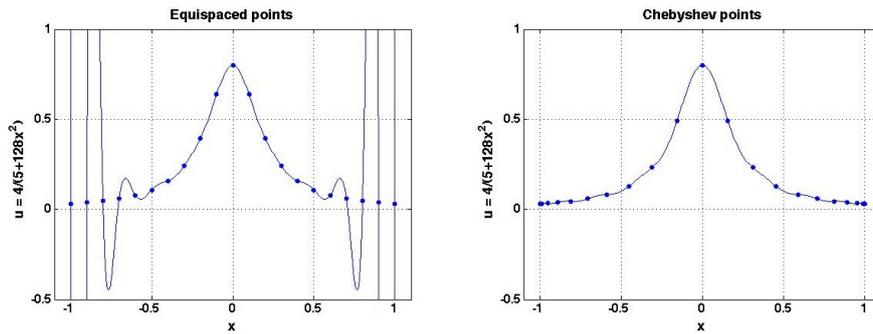


Figure 3.2: Interpolation of $u(x) = \frac{4}{5+128x^2}$ for equispaced and Chebyshev points.

3.1.1 Chebyshev differentiation matrices

Chebyshev points $x_j = \cos(j\pi/N)$ can be used to construct Chebyshev differentiation matrices, which then can be used to differentiate functions defined on these points. Now given a function u_j defined on Chebyshev points we obtain a discrete derivative w_j in two steps:

- Let $p(x)$ be the unique polynomial of degree $\leq N$ with $p(x_j) = u_j, 0 \leq j \leq N$.
- Set $w_j = p'(x_j)$.

The differentiation operator is linear so it can be represented by multiplication by an $(N + 1) \times (N + 1)$ matrix, which is denoted by D_N . Hence, we have

$$w = D_N u,$$

where N is the number of grid points and can be odd or even positive integer.

Theorem 3.1.1 For each $N \geq 1$, let the rows and columns of the $(N + 1) \times (N + 1)$ Chebyshev spectral differentiation matrix D_N be indexed from 0 to N . The entries of this matrix are

$$(D_N)_{00} = \frac{2N^2 + 1}{6}, \quad (D_N)_{NN} = -\frac{2N^2 + 1}{6} \quad (3.3)$$

$$(D_N)_{jj} = \frac{-x_j}{2(1 - x_j^2)}, \quad j = 1, \dots, N - 1 \quad (3.4)$$

$$(D_N)_{ij} = \frac{c_i (-1)^{i+j}}{c_j x_i - x_j}, \quad i \neq j, \quad i, j = 0, \dots, N \quad (3.5)$$

where

$$c_i = \begin{cases} 2, & i = 0 \text{ or } N \\ 1, & \text{otherwise.} \end{cases}$$

Using explicit formulas in the above theorem, it is simple to compute Chebyshev differentiation matrix D_N , this has been done in [13]. In [13] a program called *cheb.m* computes D_N using eight line Matlab code. This code is used for all the spectral numerical computations in *Chapters 4,5*. The program, given number of grid points N , returns Chebyshev points in vector y and Chebyshev differentiation matrix D_N .

3.1.2 Polynomial trick

So far the main components needed to solve the linear and nonlinear KdV equation have been given. However, as described above Chebyshev differentiation matrices can be used only on functions defined on Chebyshev points, that is $x \in [-1, 1]$, where as problems solved in this project are defined on the half-line $x \in [0, \infty)$. This is easily overcome by using a polynomial trick described here for the linear KdV case.

We can truncate the spatial, x , domain, say at $x = 40$ as the solution decays rapidly and will be zero at this point. Hence, the problem now is defined on a finite, closed domain by,

$$u_t + u_x + u_{xxx} = 0 \quad x \in [0, 40] \quad (3.6)$$

with boundary conditions

$$\begin{aligned} u(0) &= \sin(t) \\ u(40) &= 0 \\ u_x(40) &= 0. \end{aligned} \quad (3.7)$$

Note, that since now we have a bounded domain, we require three boundary conditions for the problem to be well-defined. That is, we have forced the solution and it's first derivative to be zero at the right boundary.

Now to make use of the Chebyshev differentiation matrix we need to transform spatial variable $x \in [0, 40]$ to $y \in [-1, 1]$. We use simple map for this

$$y = \frac{1}{20}x - 1.$$

With the change in variable our problem (3.6) has changed too,

$$\begin{aligned} 0 &= u_t + y_x u_y + y_x^3 u_{yyy} \\ &= u_t + \frac{1}{20}u_y + \frac{1}{8000}u_{yyy}. \end{aligned}$$

So we now solve the transformed problem

$$u_t + \frac{1}{20}u_y + \frac{1}{8000}u_{yyy} = 0 \quad y \in [-1, 1]$$

with boundary conditions

$$\begin{aligned} u(-1) &= \sin(t) \quad (= a) \\ u(1) &= 0 \quad (= b) \\ u_y(1) &= 0 \quad (= c). \end{aligned}$$

To impose the boundary conditions we use the polynomial trick, that is rewrite $u(y, t)$ in terms of some function $q(y)$, which satisfy Dirichlet boundary conditions and 'something else'. We chose

this 'something else' to be polynomials as they are the easiest to compute.

So let

$$u(y,t) = g(y)q(y) + h(y), \quad (3.8)$$

where $q(y)$ is a polynomial such that $q(\pm 1) = 0$ and $g(y)$ and $h(y)$ are smooth functions and $h(y)$ satisfies the same boundary conditions as u .

Now at $y = \pm 1$ we have that $u(\pm 1) = h(\pm 1)$, but

$$u_y(1) = g(1)q_y(1) + h_y(1)$$

since $u(y,t)$ and $h(y)$ satisfy the same boundary conditions, then we require that $g(1) = 0$. The simplest function giving this is $g(y) = y - 1$. Lastly we need to find $h(y)$ such that $h(1) = h_y(1) = 0$ and $h(-1) = \sin(t)$,

$$\begin{aligned} h(y) &= \frac{a+2c-b}{4}y^2 + \frac{b-a}{2}y + \frac{3b+a-2c}{4} \\ &= \frac{\sin(t)}{4}y^2 + \frac{-\sin(t)}{2}y + \frac{\sin(t)}{4} \\ &= \sin(t) \left(\frac{y^2}{4} - \frac{y}{2} + \frac{1}{4} \right). \end{aligned} \quad (3.9)$$

Hence, substituting above into (3.8)

$$u(y) = (y-1)q(y) + \sin(t) \left(\frac{y^2}{4} - \frac{y}{2} + \frac{1}{4} \right), \quad (3.10)$$

and so first three derivatives of $u(y,t)$ are

$$\begin{aligned} u_y &= (y-1)q_y + q + \sin(t) \left(\frac{y}{2} - \frac{1}{2} \right) \\ u_{yy} &= (y-1)q_{yy} + 2q_y + \sin(t) \frac{1}{2} \\ u_{yyy} &= (y-1)q_{yyy} + 3q_{yy}. \end{aligned}$$

Now, backward Euler formula is

$$\begin{aligned} \frac{u(t + \Delta t) - u(t)}{\Delta t} &= -\frac{u_y}{20} - \frac{u_{yyy}}{8000} \\ &= -\frac{(y-1)q_y + q + \sin(t) \left(\frac{y}{2} - \frac{1}{2} \right)}{20} - \frac{(y-1)q_{yyy} + 3q_{yy}}{8000} \end{aligned}$$

so

$$\begin{aligned} \frac{u(t+\Delta t)}{\Delta t} &= \frac{u(t)}{\Delta t} - \frac{(y-1)q_y + q + \sin(t)\left(\frac{y}{2} - \frac{1}{2}\right)}{20} - \frac{(y-1)q_{yyy} + 3q_{yy}}{8000} \\ \frac{(y-1)q + \sin(t+\Delta t)\left(\frac{y^2}{4} - \frac{y}{2} + \frac{1}{4}\right)}{\Delta t} &= \frac{u(t)}{\Delta t} - \frac{(y-1)q_y + q + \sin(t)\left(\frac{y}{2} - \frac{1}{2}\right)}{20} - \frac{(y-1)q_{yyy} + 3q_{yy}}{8000} \end{aligned}$$

Using the Chebyshev differential matrix D_N gives,

$$\frac{(y-1)D_N^0 q + \sin(t+\Delta t)\left(\frac{y^2}{4} - \frac{y}{2} + \frac{1}{4}\right)}{\Delta t} = \frac{u(t)}{\Delta t} - \frac{(y-1)D_N^1 q + D_N^0 q + \sin(t)\left(\frac{y}{2} - \frac{1}{2}\right)}{20} - \frac{(y-1)D_N^3 q + 3D_N^2 q}{8000}$$

and we have

$$\begin{aligned} \left(\frac{(y-1)}{\Delta t} D_N^0 + \frac{(y-1)}{20} D_N^1 + \frac{D_N^0}{20} + \frac{(y-1)}{8000} D_N^3 + \frac{3D_N^2}{8000} \right) q(y) &= \frac{u(t)}{\Delta t} - \frac{1}{\Delta t} \sin(t+\Delta t) \left(\frac{y^2}{4} - \frac{y}{2} + \frac{1}{4} \right) - \\ &\quad - \frac{\sin(t)}{20} \left(\frac{y}{2} - \frac{1}{2} \right), \end{aligned}$$

Thus, if we let

$$L = \frac{(y-1)}{\Delta t} D_N^0 + \frac{(y-1)}{20} D_N^1 + \frac{D_N^0}{20} + \frac{(y-1)}{8000} D_N^3 + \frac{3D_N^2}{8000} \quad (3.11)$$

and

$$f(y) = \frac{u(t)}{\Delta t} - \frac{1}{\Delta t} \sin(t+\Delta t) \left(\frac{y^2}{4} - \frac{y}{2} + \frac{1}{4} \right) - \frac{\sin(t)}{20} \left(\frac{y}{2} - \frac{1}{2} \right), \quad (3.12)$$

then we are left to solve a the following system

$$Lq(y) = f(y) \quad (3.13)$$

which gives $q(y)$ at a given time step t^* , then from (3.10) we find $u(y, t^*)$ and the process is repeated until the final time has been reached. Lastly, a simple mapping back into x space via $x = 20y + 20$ will give the solution $u(x, t)$.

3.2 Time-dependent problems

In general, the time coordinate is not treated spectrally. Discretising the spatial coordinate by a pseudospectral algorithm leaves us with a system of ODEs of the form $u_t = f(u, x, t)$, with u and f being vectors which can be marched forwards in time using some time stepping scheme like Backward Euler or Runge-Kutta. In principle, one sacrifices spectral accuracy in doing so, but in practice, small time steps with formulas of order two or higher often leave the global accuracy quite satisfactory. Marching with small time steps is much cheaper than computing the solution simultaneously over all space-time [12, 13].

We use backward Euler scheme in above polynomial trick method, which is applied to all linear problems in this project. Backward Euler scheme is of order $O(\Delta t)$ and thus for it not to undermine the spectral accuracy we need to use small time steps. The fourth order Runge-Kutta scheme (RK4) has been used to obtain nonlinear numerical results in *Chapter 5*. The benefits of using RK4, a rather expensive explicit method, is that it is stable with a large time step compared to other explicit methods (i.e. backward Euler), it is self-starting, that is it requires initial data only at one time level and lastly but importantly it has an error decreasing as $O(\Delta t^4)$ where Δt is the step size in t .

It is important that an accurate time stepping scheme is used, as otherwise the high spectral accuracy would be compromised. Backward Euler is much easier (and possibly cheaper) to implement, and if smaller time steps is not a problem, then this approach is preferred over the RK4 scheme for its simplicity.

Chapter 4

Linear Numerical Results

In this chapter we consider the numerical solution of the linear KdV equation. In particular, we compute solutions to the three linear problems defined earlier (1.3) - (1.5). As set out at the beginning of this project, the aim is to compute the three problems using the Fokas integral representation and using pseudospectral methods. Hence, this chapter is divided into two parts. In the first part we will discuss the numerical techniques required for integrating the solution obtained using the new Fokas transform method and present the obtained results. In the second part, we will look at the results obtained for each of the three problems by using the pseudospectral method.

4.1 Numerical integration of the Fokas spectral transform method

In *Chapter 2* we obtained the general solutions to Dirichlet and Neumann problems for the linear KdV equation,

$$q_t(x,t) + q_x(x,t) + q_{xxx}(x,t) = 0.$$

These solutions are given by the integral around the contour ∂D_+ in the complex k -plane, see Fig. (4.1).

To integrate the linear KdV equation numerically we need to map contour ∂D_+ onto the real line, which then can be integrated by the simple Trapezoidal rule. For simplicity we will use the solution

(2.32) to the problem (1.3), given by

$$q(x,t) = \frac{1}{4\pi} \int_{\partial D_+} e^{ikx} (1 - 3k^2) \underbrace{\left(\frac{e^{-(ik-ik^3)t} - e^{-iwt}}{(w - (k - k^3))} + \frac{e^{-(ik-ik^3)t} - e^{iwt}}{(w + (k - k^3))} \right)}_{(*)} dk.$$

Computation of the solution for the other problems will use exactly the same method, but involve more terms.

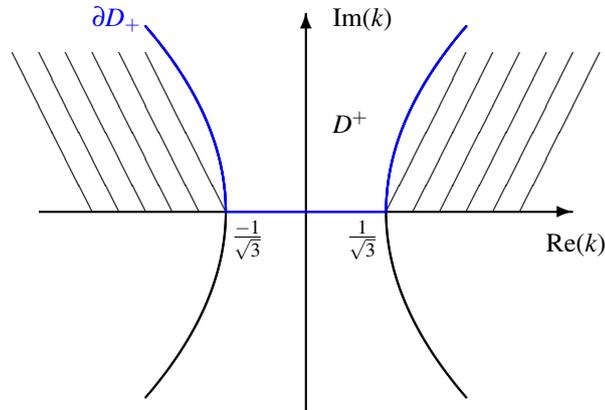


Figure 4.1: Complex k plane, with contour of integration ∂D_+ (blue) for the linear KdV equation.

4.1.1 Mapping from complex to real contour

Since ∂D_+ is the boundary of the domain of boundedness (at infinity), then appealing to Cauchy's theorem we can deduce that $q(x,t)$, under mild assumptions, is independent of ∂D_+ , even if ∂D_+ is deformed inside the shaded regions to widen out to a shape such as parabola or hyperbola for large $|Re(k)|$ [4]. The integral of $(*)$ is exactly in the form covered by Trefethen et al. Numerical methods for the quadrature of integrals of type $(*)$ have recently been derived by Weideman which are based on applying N point trapezoid or midpoint rule to $(*)$ after transformation of k to a real variable $\theta \in \mathbb{R}$. Trefethen et al showed that if the integration contour is a hyperbola in the upper half of the complex k -plane, then the trapezoidal rule has a convergence rate of order $O(3.20^{-N})$.

Several analytic functions that map a contour in the complex plane onto the real line are given in [4].

Here we use the analytic function,

$$k(\theta) = i\gamma \sin(\alpha - i\theta) \quad (4.1)$$

to map the points θ onto the real line to a hyperbola type contours in complex k -plane [11]. Thus, once mapping is applied, the solution becomes

$$q(x, t) = \frac{1}{4\pi} \int_R e^{ik(\theta)x} (1 - 3k(\theta)^2) \left(\underbrace{\frac{e^{-(ik(\theta) - ik(\theta)^3)t} - e^{-iwt}}{(w - (k(\theta) - k(\theta)^3))} + \frac{e^{-(ik(\theta) - ik(\theta)^3)t} - e^{iwt}}{(w + (k(\theta) - k(\theta)^3))}}_{(\star\star)} \right) \gamma \cos(\alpha - i\theta) d\theta \quad (4.2)$$

Hence, we want to deform ∂D_+ to a contour for which we can expect Trefethen's results. For this we need the solution to be bounded and analytic in the shaded regions in the Figures (4.3), (4.6). Now e^{ikx} is analytic and decaying in \mathbb{C}^+ , $e^{-i(k-k^3)t}$ is analytic and decaying in the shaded regions in the Figures (4.3), (4.6) and e^{iwt} does not depend on k and so is analytic and decaying for fixed t but any k values. Lastly, we need to check if there are any singularities in the denominators of $(\star\star)$. For simplicity we take $w = 1$, then roots of $k^3 - k + 1 = 0$ are

$$k_0 \approx -1.32472, \quad k_1 \approx 0.662 + 0.562i, \quad k_2 \approx 0.662 - 0.562i \quad (4.3)$$

and roots of $k^3 - k - 1 = 0$ are,

$$k'_0 \approx -k_0, \quad k'_1 \approx -0.662 + 0.562i, \quad k'_2 \approx -0.662 - 0.562i. \quad (4.4)$$

Thus, we can deform ∂D_+ to any contour inside the shaded regions, if we avoid zeros of the denominators of $(\star\star)$.

Keeping this in mind we consider the following two possibilities as integration contour:

- **Hyperbola along ray $\pi/12$**

The simplest case is to define ∂D_+ on a hyperbola through the origin asymptotic to the rays $\pi/12$ and $11\pi/12$, as using these rays we escape the singularities. We can visualize this as follows in the complex k plane.

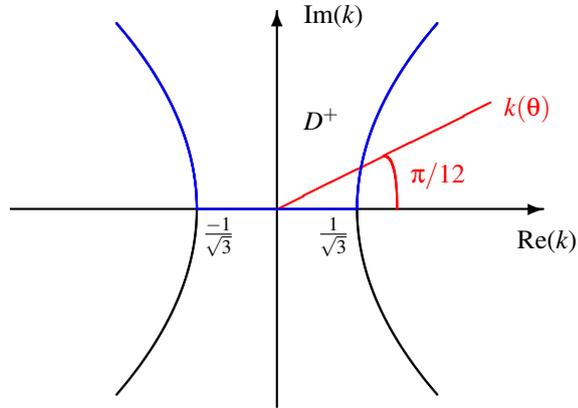


Figure 4.2: Mapping (red) of real θ line onto $k(\theta)$ contour along $\pi/12$ ray

We can see from Fig. (4.3) that roots k_0 and k_1 are close but not in the shaded regions and the rest of the roots of the two equations k_2, k'_0, k'_1 and k'_2 are well away from the shaded regions. Thus, using this mapping we can expect the convergence results in [4].

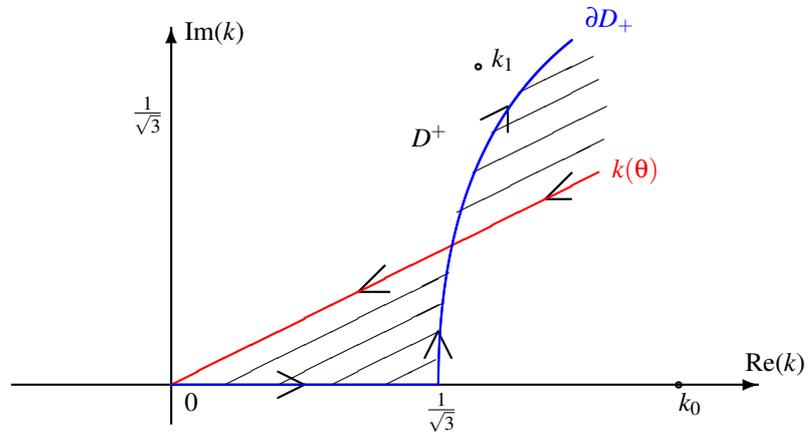


Figure 4.3: Deformation of the integration contour (blue) to red for the first map.

This mapping gives the following solution to the problem (1.3) on $x \in [0, 200], t \in [0, 2\pi]$,

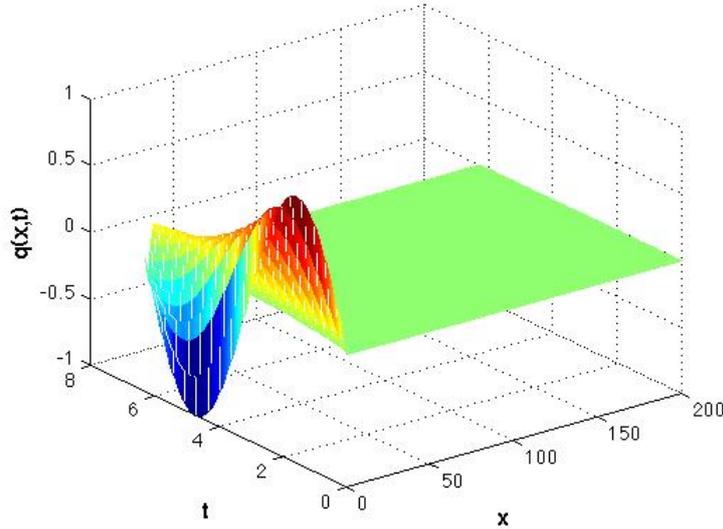


Figure 4.4: Numerical solution for the linear KdV by simple mapping along $\pi/12$ ray

- **Shifted hyperbola along the ray $\pi/6$**

A more involved but equally simple case is to take $k(\theta) = \theta$ for $0 \leq \text{Re}(k) \leq 1/\sqrt{3}$ and for $\text{Re}(k) > 1/\sqrt{3}$ apply mapping $k(\theta) = i\gamma \sin(\alpha - i\theta) - i\text{Im}(k(1/\sqrt{3}))$, see Fig. (4.5).

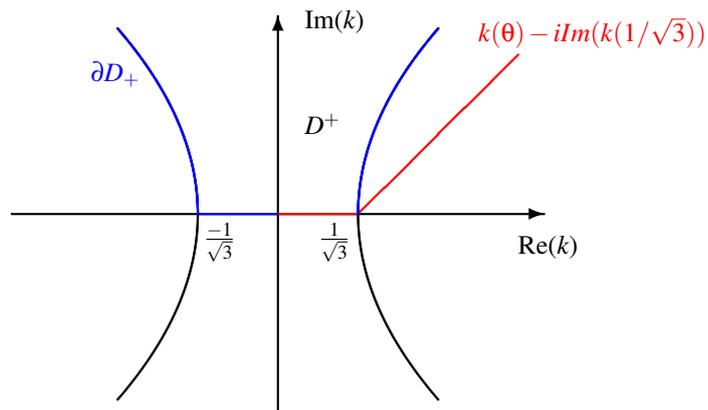


Figure 4.5: Mapping (red) of real θ line onto $k(\theta)$ contour by shift of the imaginary axis

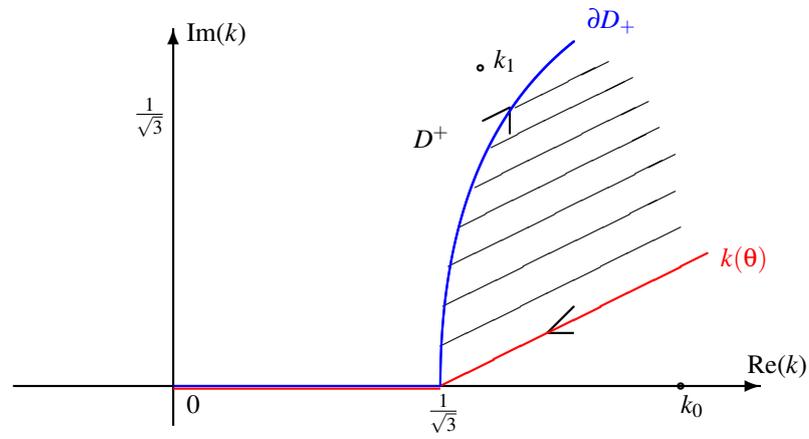


Figure 4.6: Deformation of integration contour (blue) to red for the second map.

Using this mapping we obtain the solution to the problem (1.3) in $x \in [0, 200], t \in [0, 2\pi]$,

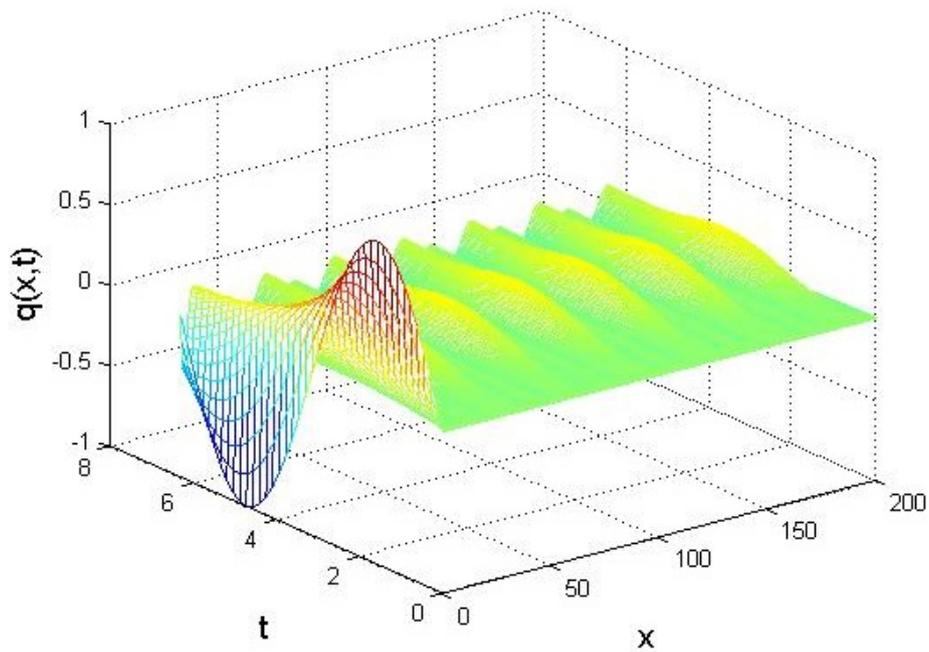


Figure 4.7: Introduction of spurious periodic solution for large x by using the shift of the imaginary axis

Both of these methods approximate the solution near the boundary extremely well as neither of them

have singularities. However, the shift of the imaginary part in the second case introduces numerical instability in the solution and as a result the solution possesses periodic waves, see Fig. (4.7). This can be solved by finer discretisation in θ , but that means more computation time. Hence, we give the upper hand to the first method, and will use it for all of the numerical computations here.

Next is the question: how far along the θ do we need to integrate? This we find simply evaluating the two exponentials that depend on θ . From Fig. (4.8) we see that both exponentials have reached zero at $\theta = 7$. Thus, it is enough to integrate over the short interval $\theta \in [0, 7]$ to capture the full solution.

It should be noted that we integrate over the $\theta \in [-7, 0]$ interval too, but because of the symmetry we can just multiply the solution obtained from integrating over $\theta \in [0, 7]$ by two. Note, that solution obtained from $\theta = 0$ we use only once.

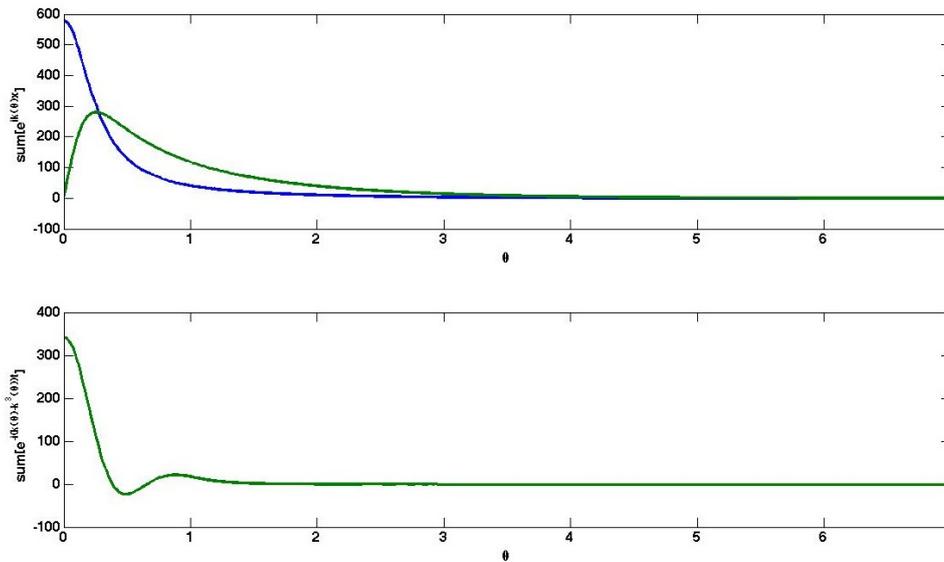


Figure 4.8: Evaluation of the two exponentials: $\sum_{x=0}^{40} e^{ik(\theta)x}$ and $\sum_{t=0}^{2\pi} e^{-i(k(\theta)-k^3(\theta))t}$ against θ .

4.1.2 Composite Trapezoidal rule

Since the integrand in (4.2) is smooth and continuous we can use a composite trapezium rule to integrate it. The trapezoidal rule is one of the Newton-Cotes formulas. These formulas are based on interpolatory polynomials that use equally spaced nodes.

Definition 4.1.1 (Composite trapezoidal rule)

Let $f \in C^2[a, b]$ of the real line and let the interval $[a, b]$ be subdivided into m subintervals $[x_k, x_{k+1}]$ of width $h = (b - a)/m$ by using the equally spaced nodes $x = a + kh$, for $k = 0, 1, \dots, m$. The composite trapezium rule then is given by

$$T(f, h) = h \left[\frac{f(x_0) + f(x_m)}{2} + \sum_{k=1}^{m-1} f(x_k) \right] \quad (4.5)$$

with error given by

$$E_T(f, h) = -\frac{(b-a)}{12} f''(\xi) h^2 = O(h^2) \quad (4.6)$$

where $\xi \in [a, b]$. Hence,

$$\int_a^b f(x) dx = T(f, h) + E_T(f, h).$$

This is a very simple integration method and is preferred because of its simplicity and ease of implementation.

The error introduced by the trapezoidal method, in general, is of order $O(h^2)$ as stated above. However, the trapezoidal rule is exponentially accurate when applied to an analytic integrand on a periodic or unbounded domain [4]. We compute precisely an integral of this kind, thus the approximation given here of the solution theoretically has an error of order $O(3^{-N})$, introduced by the trapezoidal approximation of the integral; there is no error of time stepping and spectral discretisation. In practice, of course we need to take into account the roundoff errors introduced by machine precision (here Matlab double precision). If the roundoff is treaded carefully, however, this method can in principle achieve machine precision.

Hence, to integrate (1.3) using above the mapping and composite trapezoidal rule we have the following code

```

for kk = 0:N
    theta = kk*h;
    k = i*ga*sin(alpha - i*theta);
    q = 1/(4*pi) exp(ikx)(1-3k^2).*...
        ((exp(-i(k-k^3)t) - exp(-iwt))/(w-(k-k^3))+...
         (exp(-i(k-k^3)t) - exp(iwt))/(w+(k-k^3)));
    term = real(q*gamma*cos(alpha-i*theta));
    sol = sol+term;
    if k == 0; sol = 0.5 sol; end
end
sol = (2h) sol;

```

Exponentials e^{ikt} and e^{-ikt} can be computed once outside the loop as their value does not depend on k and terms can be grouped for $e^{-i(k-k^3)t}$, for faster computations.

4.1.3 Results for the linear KdV problems for $t \in [0, 2\pi]$

Here we present the numerical results of the three linear KdV problems as set out in *Chapter 1* and for which integral representations were found in *Chapter 2*.

Numerical solution of Example 1

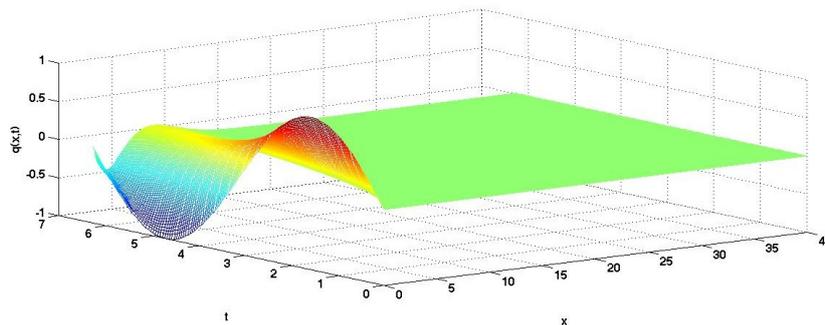


Figure 4.9: Numerical solution of the linear KdV with $q(x, 0) = 0$ and $q(0, t) = \sin(\omega t)$ for $t \in [0, 2\pi]$, where $\omega = 1$ using the new Fokas transform method, with $\alpha = \pi/12$, $\gamma = 1$

Numerical solution of Example 2

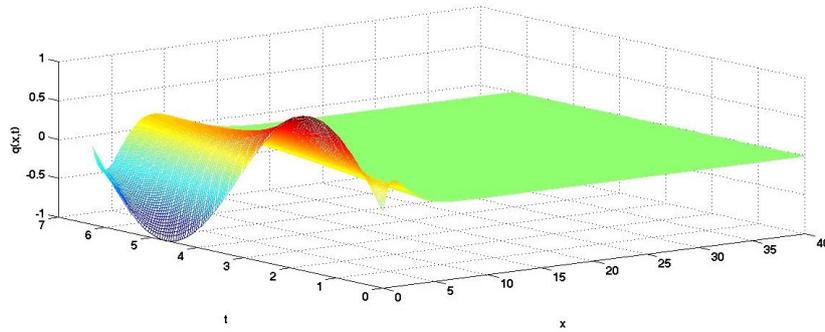


Figure 4.10: Numerical solution of the linear KdV with $q(x,0) = xe^{-ax}$ and $q(0,t) = \sin(wt)$ for $t \in [0, 2\pi]$, where $w = 1$ using the new Fokas transform method, with $\alpha = \pi/12$, $\gamma = 1$

Numerical solution of Example 3

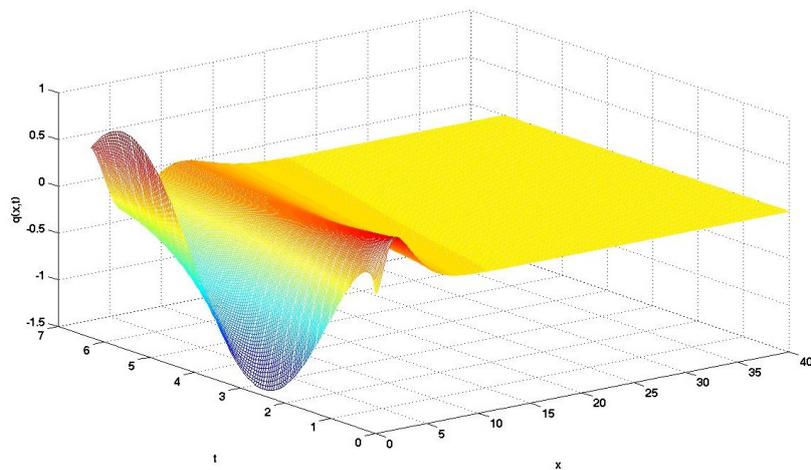


Figure 4.11: Numerical solution of the linear KdV with $q(x,0) = xe^{-ax}$ and $q_x(0,t) = \sin(wt)$ for $t \in [0, 2\pi]$, where $w = 1$ using the new Fokas transform method, with $\alpha = \pi/12$, $\gamma = 1$

4.1.4 Results for the linear KdV problems for $t \in [0, 15\pi]$

To see how the initial wave is dispersing we need to compute the solution for larger t .

Numerical solution of Example 1

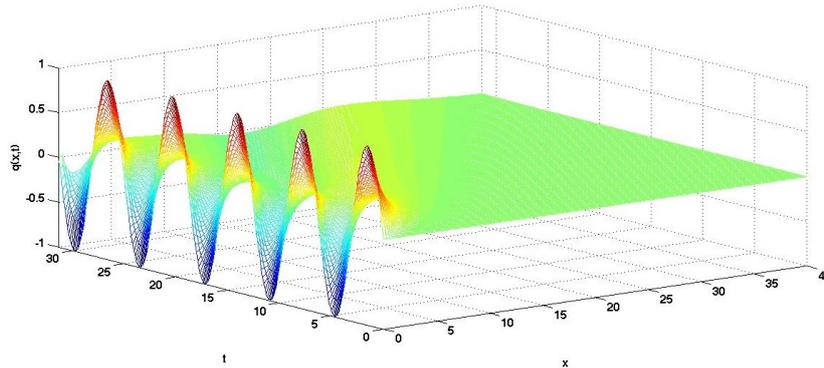


Figure 4.12: Numerical solution of the linear KdV with $q(x,0) = 0$ and $q(0,t) = \sin(\omega t)$ for $t \in [0, 10\pi]$, where $\omega = 1$ using the new Fokas transform method, with $\alpha = \pi/12$, $\gamma = 0.53$

Numerical solution of Example 2

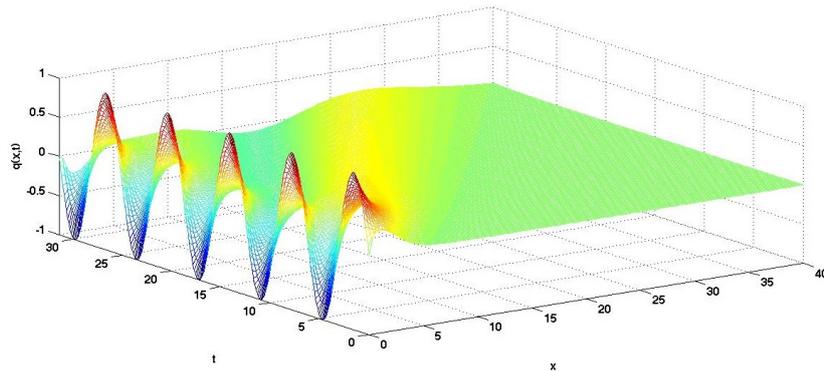


Figure 4.13: Numerical solution of the linear KdV with $q(x,0) = xe^{-ax}$ and $q(0,t) = \sin(\omega t)$ for $t \in [0, 10\pi]$, where $\omega = 1$ using the new Fokas transform method, with $\alpha = \pi/12$, $\gamma = 0.53$

Numerical solution of Example 3

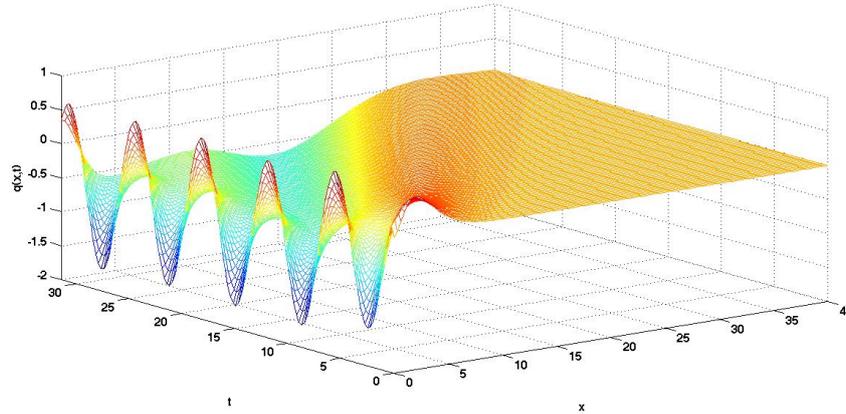


Figure 4.14: Numerical solution of the linear KdV with $q(x, 0) = xe^{-ax}$ and $q_x(0, t) = \sin(\omega t)$ for $t \in [0, 10\pi]$, where $\omega = 1$ using the new Fokas transform method, with $\alpha = \pi/12$, $\gamma = 0.53$

4.2 Numerical solution using pseudospectral method

Numerical results for the linear KdV are computed using the polynomial trick described in *Chapter 3*. Here we truncated the spatial domain far enough so that the solution and its derivative there naturally would be zero (not forced to be). Thus now we are solving the linear KdV on the finite domain with three boundary conditions (for the problem to be well posed). Next we transformed the problem from $x \in [0, 40]$ to $y \in [-1, 1]$ so that Chebyshev points and the corresponding differentiation matrix can be used. Then we impose that the solution consists of two parts. The first part is a solution to $(y - 1)q(y)$, which satisfies only the Dirichlet boundary conditions. The second part is a polynomial, $h(y)$, which satisfies the same three boundary conditions as the original problem and can easily be found. The only unknown function now is $q(y)$. Using backward Euler for time stepping and Chebyshev differential matrices for spatial differentials we arrive at an expression where all the

terms involving the unknown function $q(y)$ can be grouped together (all matrices) as in (3.11),

$$L = \frac{(y-1)}{\Delta t} D_N^0 + \frac{(y-1)}{20} D_N^1 + \frac{D_N^0}{20} + \frac{(y-1)D_N^3 + 3D_N^2}{8000}$$

and all the left over terms are known (all vectors), as in (3.12)

$$f(y) = \frac{u(t)}{\Delta t} - \frac{1}{\Delta t} \sin(t + \Delta t) \left(\frac{y^2}{4} - \frac{y}{2} + \frac{1}{4} \right) - \frac{\sin(t)}{20} \left(\frac{y}{2} - \frac{1}{2} \right).$$

Hence, we have a simple matrix problem to solve (3.13), given by

$$Lq(y) = f(y).$$

To do this, we exploit the simple Matlab matrix solver $q(y) = L \setminus f(y)$.

The error for spectral methods is exponentially decreasing, i.e. it is of order $O(h^N)$ or $O(1/N^N)$, as we use $N = 128$ in all the computations, then for all of the spectral methods the error introduced by spectral discretisation is

$$\text{pseudospectral error} \approx 1.89288 \cdot 10^{-270}.$$

However, the backward Euler method is of order $O(\Delta t)$ and hence we need Δt to be very small for good approximations to the solution. As backward Euler is implicit scheme, then it does not pose a stability issue and code is stable for all values of Δt even if it is not very accurate. We also need to take into account roundoff errors introduced by numerical arithmetics. Hence, spectral methods is much more accurate than finite difference or finite element methods for smooth functions. However, time discretisation introduces new errors in the solution and even though this still would give a method that is more accurate than FD or FE, the numerical integration of Fokas integrals is much more accurate than pseudospectral methods with time stepping.

4.2.1 Results for the linear KdV problems for $t \in [0, 2\pi]$

Here we present the numerical results of the three linear KdV problems as set out in *Chapter 1* and for which integral representations were found in *Chapter 2*.

Numerical solution of Example 1

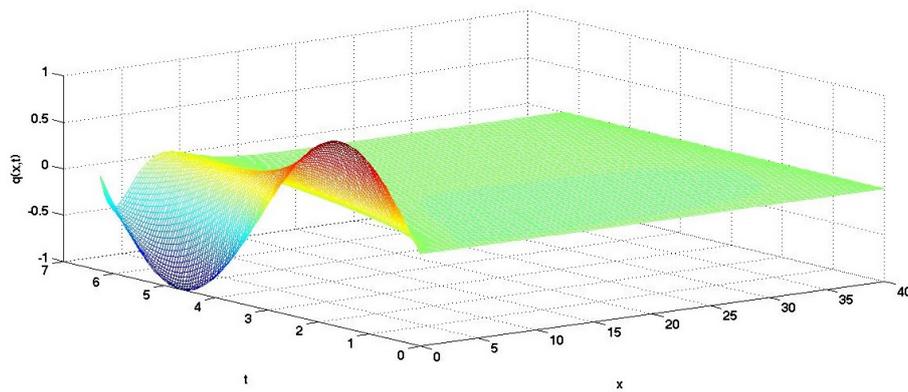


Figure 4.15: Numerical solution of the linear KdV with $q(x,0) = 0$ and $q(0,t) = \sin(wt)$ for $t \in [0, 2\pi]$, with $w = 1$.

Numerical solution of Example 2

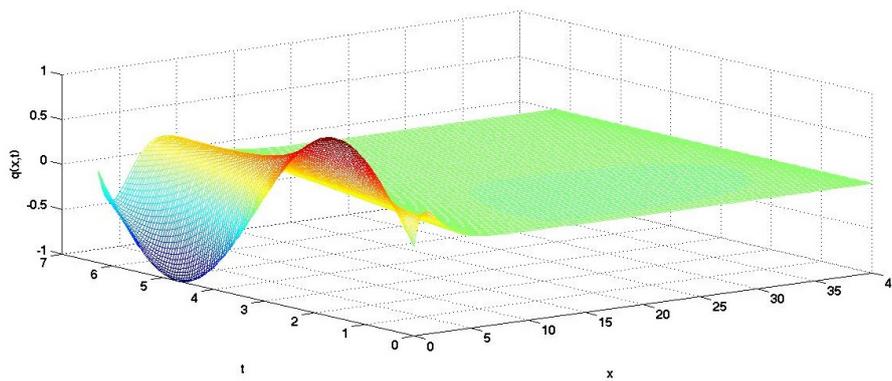


Figure 4.16: Numerical solution of the linear KdV with $q(x,0) = xe^{-ax}$ and $q(0,t) = \sin(wt)$ for $t \in [0, 2\pi]$, with $w = 1$

4.2.2 Results for the linear KdV problems for $t \in [0, 24\pi]$

To see how the initial wave is dispersing we need to compute the solution for larger t .

Numerical solution of Example 1

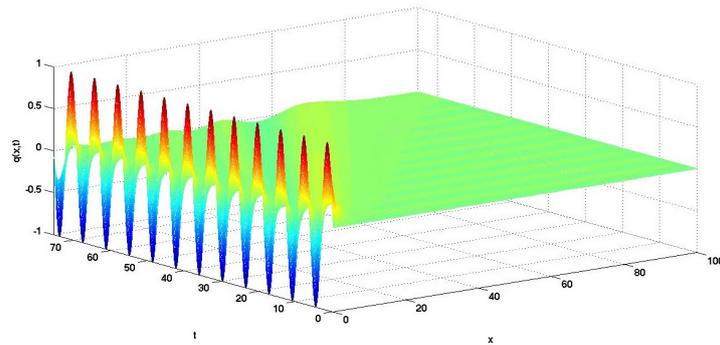


Figure 4.17: Numerical solution of the linear KdV with $q(x, 0) = 0$ and $q(0, t) = \sin(\omega t)$ for $t \in [0, 24\pi]$, with $\omega = 1$

Numerical solution of Example 2

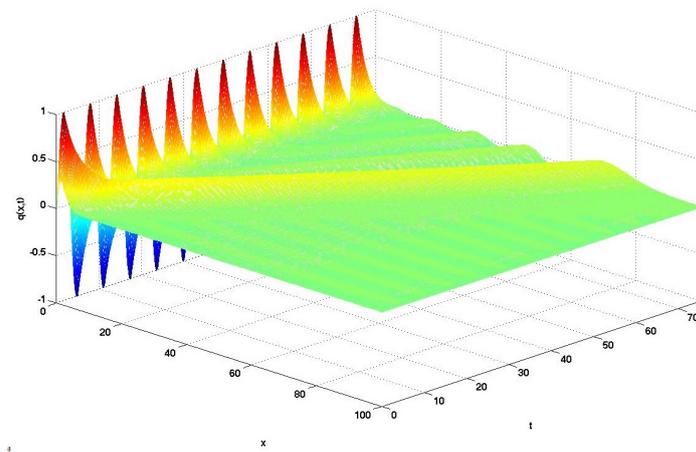


Figure 4.18: Numerical solution of the linear KdV with $q(x, 0) = xe^{-ax}$ and $q(0, t) = \sin(\omega t)$ for $t \in [0, 24\pi]$, with $\omega = 1$

4.3 Comparison between the methods

To compare the two methods, since the pseudospectral method can only be used in conjunction with Chebyshev points, then we can either run the code for the Fokas method on Chebyshev points, which are mapped into x space, i.e. $y \in [-1, 1] \rightarrow x \in [0, 40]$, or on an equally spaced x grid and with small Δx which then leaves us to match the nearest points. We measure the absolute difference between the two solution matrices to gain the error estimate. The two methods at most differed by 0.04, and that could be lowered provided the Fokas integral is evaluated on much finer grid. Hence, the two methods are approaching the same solution. Even though this does not provide an estimate of the error in either of the methods or codes, it still gives a verification that methods are computing correctly the solution.

We have not computed the Example 3, defined in (1.5), using pseudospectral methods as it requires a method other than the polynomial trick for us to apply the Chebyshev differentiation matrix and it is out of the scope of this project. However, this illustrates the benefits of using the Fokas integral solutions as not only the method is general to all linear PDEs of type (1.1), but also the numerical method is exactly the same for all of these problems, one only needs to change the integral obtained (as in *Chapter 2*) and put it in the code. When using other methods we are forced to use different methods depending on the problem for computing the solution.

Max t value	200×127	200×200	400×400	$\Delta\theta$
2π	1.23s	1.27s	2.48s	0.2
6π	1.57s	1.7s	3.46s	0.1
10π	8.28s	9.32s	20.76s	0.01
15π	1m 18s	1m 35s	3m 17s	0.001

Table 4.1: CPU time taken computing the solution to (1.3) using the Fokas integral. 200×127 means that the solution is computed for 200 points in x direction and 127 points in t directions and s -seconds, m -minutes.

In regards to the speed of the computation, we present the following observations of the two methods.¹ Experiments running each code for a different length of time and from rough to fine discretisation in time and space, suggests the following. The code integrating the new method is much faster than pseudospectral code for all possible discretisations (up to $N = 3333$ in space and time, and where the solution domain is $x \in [0, 40]$ and $\max(t) \leq 6\pi$), see Tables (4.1) and (4.2). For example, the new method computed solution on domain $x \in [0, 40], t \in [0, 2\pi]$ for 1000 points in x and t domains in just 21 seconds, where as pseudospectral methods took too long to complete the computation.

However, as we increase the time domain, not necessarily increasing the number of points, when maximum of $t = 10\pi$ both methods seem to compute at similar speeds. After this point pseudospectral methods become faster. The integration method loses its upper hand because there is a correlation between $\max(t)$ and $\max(x)$ and the size of $\Delta\theta$, i.e. for domain $x \in [0, 40], t \in [0, 2\pi]$ we have $\Delta\theta = 0.2$, but for domain $x \in [0, 40], t \in [0, 15\pi]$ we need $\Delta\theta = 0.001$ for the solution not to blow up. Hence, as $\Delta\theta$ is linked to the number of quadrature points needed (remember we need to compute to $\theta = 7$) then as $\Delta\theta$ gets smaller the number of quadrature points increases and the more time it takes. It is possible that this is just a numerical problem, which once resolved would give the superiority (in computation time) to the Fokas integral method for all time values.

Max t value	200×127	Δt	200×200	Δt	400×400	Δt
2π	9.01s	0.0314	12.53s	0.0495	2m 34s	0.0157
6π	9.58s	0.094	14.5s	0.1484	2m 30s	0.0471
10π	9s	0.1571	13.5s	0.2474	2. 31s	0.0785
15π	9s	0.2356	13.64s	0.3711	2m 34s	0.1178

Table 4.2: CPU time taken computing solution to (1.3) using pseudospectral methods. 200×127 means that solution is computed for 200 points in x direction and 127 points in t directions and s -seconds, m -minutes.

¹Note, for both codes all the variables that can have been evaluated outside the loops and advantage where possible have been taken of Matlab vectorization.

Solutions from both of methods, as expected show decrease in hight of the main solitary wave. This process is slow, especially if energy in initial condition is large. We can see this especially well in Fig. (4.19).

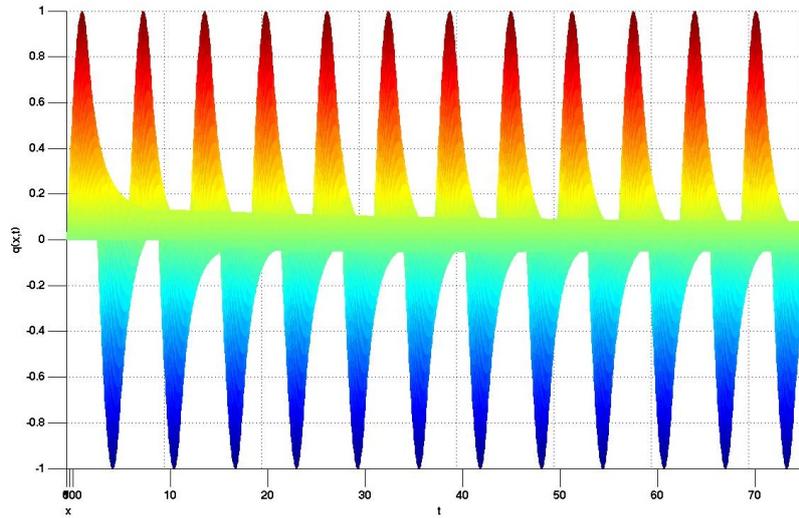


Figure 4.19: Diffusion of the soliton in numerical solution of the linear KdV with $q(x, 0) = 0$ and $q(0, t) = \sin(\omega t)$ for $t \in [0, 24\pi]$, with $\omega = 1$

Now, we should note that above computation time comparisons were made on the speed of computation for the same number of points in x and t directions, not the accuracy. As a matter of fact all solutions from pseudospectral methods for speed comparisons were less accurate than the numerical integral method of Fokas solution. Hence, as discussed above, for pseudospectral methods to be as accurate as the new Fokas integral method we require that $\Delta t = 10^{-9}$ (for second order scheme in time). Thus, as for pseudospectral methods computation time increases drastically with the increase of the problem domain (see Tables (4.1),(4.2)), it will not be practical to compute pseudospectral methods to the same accuracy as numerically integrating the Fokas integral. This is because the code using Chebyshev differentiation matrices will become very slow. For example, to compute solution for Example 1 on domain $x \in [0, 40], t \in [0, 2\pi]$ with $\Delta t = 10^{-9}$ took 42 minutes. Hence, the Fokas integral method wins hands down for all domain sizes and maximum times in the case when we want the same accuracy from both methods.

Chapter 5

Nonlinear Numerical Results

As the numerical computation has proved to be so easy to implement and to such high accuracy, it is just natural to try to compute the nonlinear KdV next. In this chapter we are interested in looking at the nonlinear KdV equation defined in (1.6),

$$q_t(x,t) + q_x(x,t) + q(x,t)q_x(x,t) + q_{xxx}(x,t) = 0 \quad x \in [0, \infty), t > 0.$$

In particular we consider (1.6) with the following initial and boundary data,

$$\begin{cases} q(x,0) = xe^{-ax} & a \in (0, 1] \\ q(0,t) = \sin(wt) & w \in \mathbb{R}, \end{cases}$$

as defined in (1.7). First we will apply the split step Fourier method using the Chebyshev grid to find the pseudospectral solution. Next we aim to adapt the split step method to solve this non-periodic, nonlinear problem, by taking advantage of the exact formula for solving the linearised problem. We start by solving the problem by a classical split step spectral method, see Chilton. We then indicate what modifications are necessary in order to incorporate the Fokas integral formula.

5.1 Pseudospectral method using the split-step method

In this section, a split-step Fourier method is presented using the nonlinear KdV problem subject to (1.7). The idea of this method is to split the problem into a nonlinear part

$$u_t(x,t) + u(x,t)u_x(x,t) = 0$$

and a linear part

$$u_t(x,t) + u_x(x,t) + u_{xxx}(x,t) = 0$$

and alternate between them.

Since we have non-homogeneous time-dependent uncoupled boundary conditions, i.e. the problem is non-periodic, we solve the problem on the Chebyshev grid and apply the polynomial trick (see *Chapter 3*) to impose the boundary conditions. For this we need to truncate the spatial domain to say $x \in [0, 40]$ ¹, then the nonlinear KdV mapped on the Chebyshev grid is

$$u_t(y,t) + \frac{1}{20}u_y(y,t) + \frac{1}{20}u(y,t)u_y(y,t) + \frac{1}{8000}u_{yyy}(y,t) = 0 \quad y \in [-1, 1], t > 0. \quad (5.1)$$

subject to initial and boundary data

$$\left\{ \begin{array}{l} u(y,0) = 20(y+1)e^{-a20(y+1)} \quad a \in (0, 1], y \in [-1, 1] \\ u(-1,t) = \sin(wt) \quad w \in \mathbb{R} \\ u(1,t) = 0 \\ u_y(1,t) = 0, \end{array} \right. \quad (5.2)$$

5.1.1 Non-linear step

In this first step we advance the nonlinear part of the solution,

$$u_t(y,t) + \frac{1}{20}u(y,t)u_y(y,t) = 0 \quad (5.3)$$

by half a time step, $\Delta t/2$, using RK4 method.

Boundary conditions are imposed using the polynomial trick,

$$u(y,t) = (1-y)q(y,t) \quad y \in [-1, 1], t > 0 \quad (5.4)$$

¹This is far enough for the solution and its first derivative to be zero there as in the original problem we have that $u(x,t) \rightarrow 0, u_x(x,t) \rightarrow 0$ as $x \rightarrow \infty$. For some of the plots, to illustrate the behaviour for large t , in calculations we have truncated the domain at $x = 100$.

and hence, the boundary data for this part is

$$q(\pm 1, t) = 0.$$

Hence, we have

$$\begin{aligned} u_t(y, t) &= (1-y)q(y, t)_t \\ u_y(y, t) &= -q(y, t) + (1-y)q_y(y, t), \end{aligned}$$

and in terms of $q(y, t)$, (5.3) is given by

$$(1-y)q_t(y, t) + \frac{1}{20}(1-y)q(y, t)[-q(y, t) + (1-y)q_y(y, t)] = 0$$

Rearranging above we obtain,

$$\begin{aligned} q_t(y, t) &= -\frac{1}{20}q^2(y, t) + \frac{1}{20}q(y, t)q_y(y, t)(1-y) \\ &= -\frac{1}{20}q^2(y, t) + \frac{1}{40}(q^2(y, t))_y(1-y). \end{aligned} \quad (5.5)$$

Given an initial condition $u(y_j, 0)$, $0 \leq j \leq N$, then in terms of $q(y, t)$ from (5.4), initial condition becomes

$$q(y_j, 0) = \frac{u(y_j, 0)}{(1-y_j)}. \quad (5.6)$$

We use the fourth order Runge Kutta scheme, given by,

$$q_{n+1} = q_n + \frac{1}{6}(d_1 + 2(d_2 + d_3) + d_4) \quad (5.7)$$

where

$$\begin{aligned} d_1 &= \frac{\Delta t}{2} \left(-\frac{1}{20}q_n^2(y, t) + \frac{1}{40}[q_n(y, t)]_y^2(1-y) \right) \\ d_2 &= \frac{\Delta t}{2} \left(-\frac{1}{20} \left[q_n(y, t) + \frac{1}{2}d_1 \right]^2 + \frac{1}{40} \left[q_n(y, t) + \frac{1}{2}d_1 \right]_y^2(1-y) \right) \\ d_3 &= \frac{\Delta t}{2} \left(-\frac{1}{20} \left[q_n(y, t) + \frac{1}{2}d_2 \right]^2 + \frac{1}{40} \left[q_n(y, t) + \frac{1}{2}d_2 \right]_y^2(1-y) \right) \\ d_4 &= \frac{\Delta t}{2} \left(-\frac{1}{20} [q_n(y, t) + d_3]^2 + \frac{1}{40} [q_n(y, t) + d_3]_y^2(1-y) \right) \end{aligned}$$

to advance the solution half a time step. Since we are solving the Dirichlet problem, the above is only applied to interior points of the spatial domain. The Chebyshev differential matrix D_N is used to differentiate with respect to y . Hence, above becomes

$$\begin{aligned} d_1 &= \frac{\Delta t}{2} \left(-\frac{1}{20} q_n^2(y,t) + \frac{1}{40} (1-y) q_n^2(y,t) \cdot D_N \right) \\ d_2 &= \frac{\Delta t}{2} \left(-\frac{1}{20} \left[q_n(y,t) + \frac{1}{2} d_1 \right]^2 + \frac{1}{40} (1-y) \left[q_n(y,t) + \frac{1}{2} d_1 \right]^2 \cdot D_N \right) \\ d_3 &= \frac{\Delta t}{2} \left(-\frac{1}{20} \left[q_n(y,t) + \frac{1}{2} d_2 \right]^2 + \frac{1}{40} (1-y) \left[q_n(y,t) + \frac{1}{2} d_2 \right]^2 \cdot D_N \right) \\ d_4 &= \frac{\Delta t}{2} \left(-\frac{1}{20} [q_n(y,t) + d_3]^2 + \frac{1}{40} (1-y) [q_n(y,t) + d_3]^2 \cdot D_N \right). \end{aligned}$$

The full solution in $q(y,t)$ for half a time step with initial and boundary data is then given by

$$q \left(y_j, t + \frac{\Delta t}{2} \right) = \left[\sin(\omega t); q \left(y_i, t + \frac{\Delta t}{2} \right); 0 \right], \quad i = 1, \dots, N-1, j = 0, \{i\}, N \quad (5.8)$$

and the full solution in $u(y,t)$ for the nonlinear part is,

$$u \left(y_j, t + \frac{\Delta t}{2} \right) = (1-y_j) q \left(y_i, t + \frac{\Delta t}{2} \right), \quad i = 1, \dots, N-1, j = 0, \{i\}, N. \quad (5.9)$$

5.1.2 Linear step

Now the linear part,

$$u_t(y,t) + \frac{1}{20} u_y(y,t) + \frac{1}{8000} u_{yyy}(y,t) = 0 \quad (5.10)$$

with boundary data

$$u(-1,t) = \sin(\omega t), \quad u(1,t) = 0, \quad u_y(1,t) = 0,$$

where the initial data is now the solution from the nonlinear step. Now, this is exactly Example 3, solved in *Chapter 4*, only now we advance the solution half a time step at a time. Hence, we apply

the same polynomial method as in *Chapter 3* and it is readily seen that we obtain the same linear system as in (3.13) but with half a time step, i.e. now instead of (3.11) we have

$$L = \frac{2}{\Delta t}(y-1)D_N^0 + \frac{(y-1)}{20}D_N^1 + \frac{D^0}{20} + \frac{(y-1)}{8000}D_N^3 + \frac{3D_N^2}{8000} \quad (5.11)$$

and instead of (3.12) we have

$$f(y) = \frac{2}{\Delta t}u(t) - \frac{2}{\Delta t} \sin(t + \Delta t) \left(\frac{y^2}{4} - \frac{y}{2} + \frac{1}{4} \right) - \frac{\sin(t + \Delta t/2)}{20} \left(\frac{y}{2} - \frac{1}{2} \right). \quad (5.12)$$

Hence, the method can be summarized as follows,

- set initial data u_0
 set initial time $t = 0$
 set final time $tMax$
 set Δt
 compute number of plots $nplots = tMax/dt$
 - for $i = 1 : nplots$
 1. Advance the nonlinear part $u_t + uu_x = 0$ by $t + \Delta t/2$ from time t to time $t + \Delta t/2$.
 If $i = 1$ use given initial data u_0 , else use data from linear part obtained at $i - 1$.
 2. Advance the linear part $u_t + u_x + u_{xxx} = 0$ by $t + \Delta t/2$ from time $t + \Delta t/2$ to time $t + \Delta t$.
 Initial data is always the solution from the nonlinear part.
- end

5.1.3 Numerical results from pseudospectral method

Using the above description, the nonlinear KdV with data as defined in (1.7) has the following solution

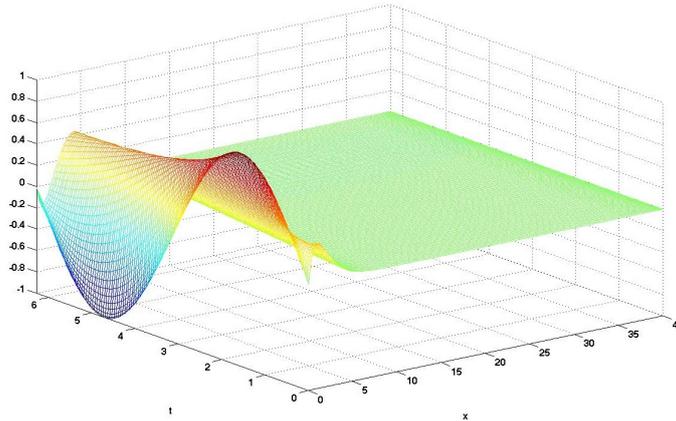


Figure 5.1: Solution to the nonlinear KdV with $u(x,0) = xe^{-x}$ and $u(0,t) = \sin(t)$, for $x \in [0,40]$, $t \in [0, 2\pi]$ with $\Delta t = 0.01$.

We can see the difference between the linear and the nonlinear KdV from Figures (5.2) and (5.3) below.

The figures above verify that the linear solution is dispersive and loses energy whereas the nonlinear KdV is indeed not dispersive and the peak of the soliton is kept at the same height. This can be seen especially well if more energy is put in by initial data as in Fig. (5.4).

5.2 Pseudospectral method and Fokas integral method using the split-step method

A possible future development would be to use the exact computation for the linear step, achieved by the Fokas method in *Chapter 4*, to propose an alternative way to compute numerically the solution of non-periodic boundary value problem for the nonlinear KdV. Indeed, since it is so easy to compute the linear KdV using the Fokas integral method and because it can be applied to any linear

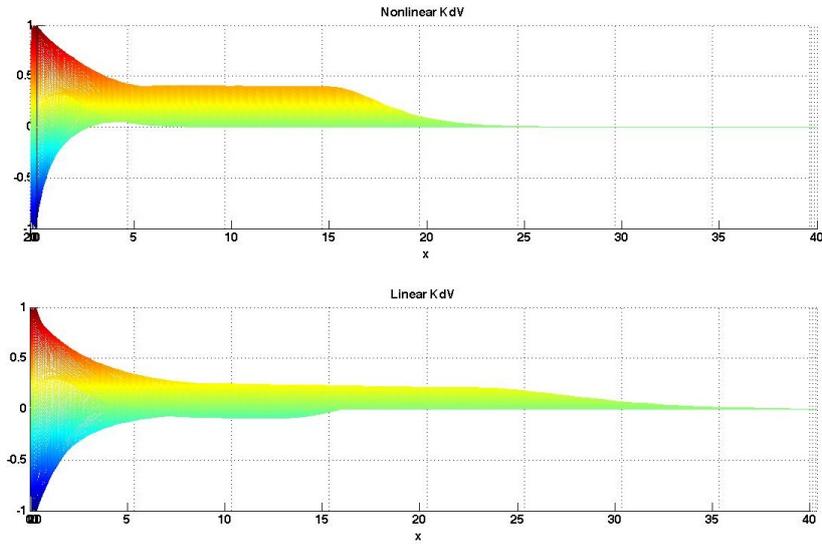


Figure 5.2: Comparison between linear and nonlinear KdV solution with $u(x, 0) = xe^{-x}$ and $u(0, t) = \sin(t)$, for $x \in [0, 40]$, $t \in [0, 8\pi]$.

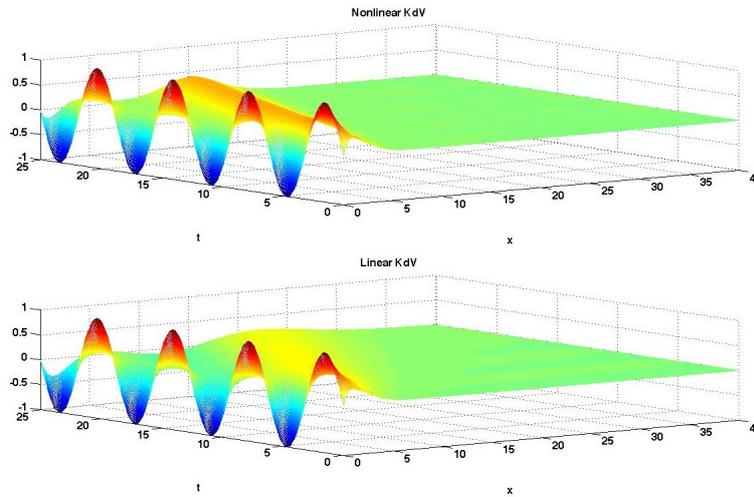


Figure 5.3: Comparison between linear and nonlinear KdV solution with $u(x, 0) = xe^{-x}$ and $u(0, t) = \sin(t)$, for $x \in [0, 40]$, $t \in [0, 8\pi]$.

PDE of the form (1.1) with no restriction on boundary conditions, we suggest the combinations of the pseudospectral method with the Fokas integral method for calculating the nonlinear KdV using

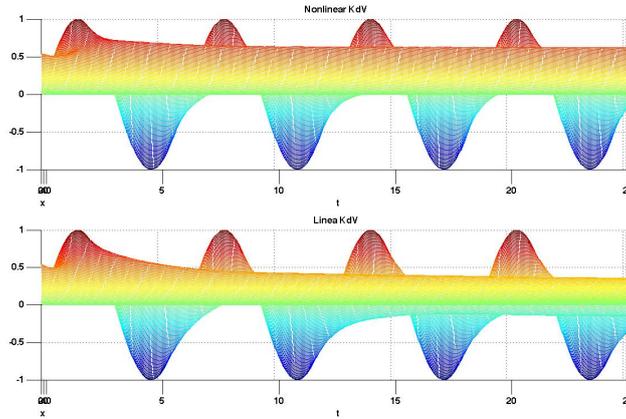


Figure 5.4: Comparison between linear and nonlinear KdV solution with $u(x,0) = x^2 e^{-x}$ and $u(0,t) = \sin(t)$, for $x \in [0, 40]$, $t \in [0, 8\pi]$.

the split step method.

This would involve solving the nonlinear part, $u_t + uu_x = 0$, as above using the Chebyshev differentiation matrix and a time stepping scheme like RK4. Then, the linear part, $u_t + u_x + u_{xxx} = 0$, would use the solution obtained from the Fokas integral method. Since, both linear and nonlinear parts have to "communicate" with each other, that is the linear part needs to use the solution from the nonlinear part and vice versa, we cannot use the code developed for the Fokas method as is. If we would use the numerical integration on the linear part as we have done in *Chapter 4* then the "communication" would be only one way and the linear part would never know what the nonlinear part does. What we require is to take the solution from the nonlinear part as an initial data for the linear part and evaluate it at k , μ_1 and μ_2 (e.g. for Dirichlet case \hat{q}_0 in (2.27)). Note, that since we cannot solve now for the initial data analytically we compute it using the FFT (Fast Fourier Transform) and the IFFT (Inverse Fast Fourier Transform) pair.

The difficulty is in the computation via FFT of the transform of the new initial condition not only for k real, but also at the other points needed in order to evaluate the integral representation. This work is in progress, as it presents a more serious challenge than we originally envisaged. This method should produce more accurate results than the pure pseudospectral method and its computation speed should also improve, especially for larger domains.

Chapter 6

Conclusions and Further Work

In this chapter we summarize the work carried out in this dissertation. We will discuss the main results of the work carried out here and refer the reader to the relevant chapters. We will also indicate the areas for possible further research in the topic.

The aim of this dissertation was to compare the numerical results of the recently developed Fokas integral method for solving boundary value problems for linear and integrable nonlinear PDEs in two variables and the well known pseudospectral methods. The Fokas integral method produces the exact solution in the integral form. This only leaves us to do the numerical integration, which is much easier to do than use any other methods developed until now for the numerical computation of linear evolutionary PDEs on the half line. Moreover, it also proves to be much more accurate.

The theory of the Fokas integral method was developed in *Chapter 2* and we used the linear KdV equation to illustrate the method. Even though the method was explained only in terms of the linear KdV, exactly the same method applies to ALL linear evolutionary PDEs. Because of this, the method is of great importance as it provides a generalized way of solving such equations, rather than needing to resort to a specific transform (and in many cases not being able to solve the problem at all). Moreover, in comparison to solutions obtained using classical Fourier transforms, the integral representation obtained from the Fokas method is uniformly convergent at the boundaries and it is spectrally decomposed. In *Chapter 2*, we found the general solutions for both Dirichlet and Neumann cases, as well as giving integral representation of the solution to the three particular sets of

initial and boundary data.

To compute these integrals numerically we used the Trapezoidal rule which we introduced in *Chapter 4*. It is very easy to implement and is cheap to use. Moreover, applying the Trapezoidal method to integrals on the infinite domain produces error of order $O(3^{-N})$, where N is the number of quadrature points used. Thus, the numerical solution obtained by numerically integrating the solution from the Fokas method, in theory is more accurate than the machine precision. However, of course in practice we can only achieve the machine precision but with this method it is readily obtained. Hence, this gives an extremely easy way of solving linear evolution PDEs analytically and numerically.

We also discussed the pseudospectral methods in *Chapter 3*. These methods have been very popular since the 1970s as they, until now, have proved to be by far more accurate than any finite differences or finite element methods when applied to smooth functions. In general, one only approximates spatial variables spectrally, using finite differences for time stepping. This is also what we have done, using the Backward Euler method for the linear KdV and the fourth order Runge Kutta method for the nonlinear KdV. Time stepping is computationally much cheaper than discretising the time domain spectrally. However, even though pseudospectral methods have errors that decrease exponentially for spectrally discretised domains, the time stepping introduces new errors. For Backward Euler errors are of order $O(\Delta t)$, for RK4 of order $O(\Delta t^4)$.

In *Chapter 4* we showed numerical results of the linear KdV for both of the methods. We verified the two solutions for both of the methods approach the same solution and thus concluded that the methods have been implemented correctly. We also tested out the computation time taken for different grid sizes for the two methods. We concluded that numerical integration of the Fokas method as implemented here is much faster when it comes to domains where $0 \leq \max T \leq 6\pi$. This was because of some correlation occurring between the size of the domain and θ . This, correlation is not understood at the moment and most probably is a numerical issue rather than an issue with the method. In future it would be very useful to understand why this correlation occurs as if it does prove to be purely numerical, then implementing the method differently would provide us with a numerical method which is far superior in accuracy and speed for any domain. However, even with this correlation, which forced us to use more quadrature points for integration, the Fokas method is still numerically superior when it come to matching the accuracy of the solution. Pseudospectral methods were much faster for time domains $t \in [0, 15\pi]$ for a fixed number of grid points, but also far less accurate than the numerical code for the Fokas integral method.

Thus, the conclusion is that the new Fokas integral method is a first general method which can be applied to all the linear evolutionary PDEs. It produces an analytic integral representation of the solution around a contour in the complex plane. Using simple numerical integration method, it is extremely easy to implement the method numerically. It is more accurate than the pseudospectral methods, and if accuracy is an issue, the Fokas method is also faster too (as currently implemented).

Because of the ease of implementation of the linear method it would be very useful to apply this approach for the nonlinear case too. Thus, in *Chapter 5* we first described the split step method for solving the nonlinear KdV equation using pseudospectral methods as currently one of the most accurate schemes for this equation. Then we give an outlay of how the solution from the Fokas method for the linear KdV could be used in the second step to give us a hybrid method, one which is half pseudospectral and half numerical integration of the solutions from the Fokas method. We have not been able to develop or implement this idea in this project because of the time constraints. However, it would be an interesting path to explore, as if implemented successfully, it would provide a more accurate solution than the purely pseudospectral method and the speed of computation should improve too, as the step size in time would not need to be so small.

Bibliography

- [1] Flyer N., Fokas A.S. (2007), "*A new Transform Method for the Numerical Integration of Evolution PDEs*", Proc. Roy. Soc. A, (preprint)
- [2] Fokas A.S. (2002), "*A new Transform Method for Evolution Partial Differential Equations*", IMA Journal of Applied Mathematics, 67:559-590.
- [3] Chilton, D. (Jul. 2006), "*An Alternative Approach to the Analysis of Two-Point Boundary Value Problems for Linear Evolution PDEs and Applications*", Ph.D. Thesis, Reading University: U.K.
- [4] Trefethen L.N., Weideman J.A.C., Schmelzer T. (Aug. 2006), "*Talbot Quadratures and Rational Approximations*", BIT Numerical Mathematics 46:653-670.
- [5] Fokas A.S. (Jul. 1997), "*A Unified Method for Solving Linear and Certain Nonlinear PDEs*", Mathematical, Physical and Engineering Sciences, 453:1411-1443.
- [6] Pelloni B. (Oct. 2006), "*Linear and Nonlinear generalized Fourier Transforms*", Phil. Trans. R. Soc. A, 364:3231-3249.
- [7] Fokas A.S., Pelloni B. (2000), "*Integral Transforms, Spectral Representation and the d -bar Problem*", Proceedings of the Royal Society of London Series A, 456:805-833.
- [8] Fokas A.S., Sung L.Y., "*Initial-Boundary Value Problem for Linear Dispersive Evolution Equations on Half-Line*", (preprint).
- [9] Fokas A. (Jul. 2001), "*Two Dimensional Linear PDE's in a Convex Polygon*", Proceedings of the Royal Society of London Series A, 457:371-393

- [10] Gottlieb D., Shu C. W. (1997), "*The Gibbs Phenomenon and its Resolution*", SIAM Review, 39:644668.
- [11] Flyer N., Personal communication
- [12] Boyd J.P. (2001), "*Chebyshev and Fourier Spectral Methods*", 2nd edition, Dover Publications Inc. (UK)
- [13] Trefethen L.N. (2002), "*Spectral Methods in Matlab*", SIAM (USA)
- [14] Fornberg B. (1998), "*A Practical Guide to Pseudospectral Methods*", Cambridge University Press (UK)
- [15] Iserles A. (1996), "*A First Course in the Numerical Analysis of Differential Equations*", Cambridge Texts in Applied Mathematics, Cambridge University Press
- [16] Wolfram Research Inc. (2007), "*Numerical Solution of Partial Differential Equations*", Online Mathematica Tutorial,
URL: <http://reference.wolfram.com/mathematica/tutorial/NDSolvePDE.html>
- [17] Colliander J., Keel M., Staffilani G., Takaoka H., Tao T., "*Equations of Korteweg de Vries type*", URL: <http://www.math.ucla.edu/tao/Dispersive/kdv.html>, accessed on 05/07/2007
- [18] Davenport C. M. (2006), "*The General Analytical Solution for the Korteweg- de Vries Equation*", URL: <http://home.usit.net/cmdaven/korteweg.htm>, accessed on 11/08/2007