



National Library
of Canada

Acquisitions and
Bibliographic Services Branch

395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale
du Canada

Direction des acquisitions et
des services bibliographiques

395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Votre titre - Votre référence

Our file - Notre référence

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

**THE PSEUDOSPECTRAL CHEBYSHEV METHOD
FOR TWO-POINT BOUNDARY VALUE PROBLEMS**

by

Yunyun Liu

B.Sc. Shanghai University of Science and Technology 1982

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE

in the Department of Mathematics and Statistics

of

Simon Fraser University

© Yunyun Liu 1992

SIMON FRASER UNIVERSITY

April, 1992

All rights reserved. This work may not be
reproduced in whole or in part, by photocopy
or other means, without the permission of the author.



National Library
of Canada

Bibliothèque nationale
du Canada

Acquisitions and
Bibliographic Services Branch

Direction des acquisitions et
des services bibliographiques

395 Wellington Street
Ottawa, Ontario
K1A 0N4

395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file *Votre référence*

Our file *Notre référence*

The author has granted an irrevocable non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.

L'auteur a accordé une licence irrévocable et non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without his/her permission.

L'auteur conserve la propriété du droit d'auteur qui protège sa thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

ISBN 0-315-83694-6

Canada

APPROVAL

Name: Yunyun Liu

Degree: Master of Science

Title of thesis: **THE PSEUDOSPECTRAL CHEBYSHEV METHOD
FOR TWO-POINT BOUNDARY VALUE PROBLEMS**

Examining Committee: Dr. G. A. C. Graham
Chairman

Dr. Manfred Trummer, Senior Supervisor

Dr. Robert D. Russell

Dr. George Bojadziev

Dr. Tao Tang, External Examiner

Date Approved:

April 2, 1992

PARTIAL COPYRIGHT LICENSE

I hereby grant to Simon Fraser University the right to lend my thesis, project or extended essay (the title of which is shown below) to users of the Simon Fraser University Library, and to make partial or single copies only for such users or in response to a request from the library of any other university, or other educational institution, on its own behalf or for one of its users. I further agree that permission for multiple copying of this work for scholarly purposes may be granted by me or the Doan of Graduate Studies. It is understood that copying or publication of this work for financial gain shall not be allowed without my written permission.

Title of Thesis/Project/Extended Essay

THE PSEUDOSPECTRAL CHEBYSHEV METHOD

FOR TWO-POINT BOUNDARY VALUE PROBLEM

Author:

(signature)

YUNYUN LIU

(name)

MAY 20, 1992

(date)

Abstract

Although spectral methods have attracted much of the attention in current research of numerical methods for solving differential equations, little experience is available in applying spectral methods to solve (two - point) boundary value problems. In this thesis, two spectral methods (the pseudospectral Chebyshev method and the method of transformation to the circle) have been studied to solve second order boundary value problems (BVPs).

Emphasis is given to singularly perturbed BVPs that are controlled by a singularity parameter ϵ . Three kinds of singularities are investigated: mild ($\epsilon = 10^{-2}$), stiff ($\epsilon = 10^{-4}$) and very stiff ($\epsilon = 10^{-6}$), and problems with interior or boundary layers are considered.

Numerical performances of the above mentioned two spectral methods are judged by condition numbers, average errors and cpu times. It has been found that the pseudospectral Chebyshev method is more suitable to solve non-stiff problems than to stiff problems; that the method of transformation to the circle can decrease the condition numbers of the relevant matrices when applied to solve non-stiff problems. However, for stiff problems, it is nevertheless seldom successful. A brief comparison of the above two methods to the Chebyshev collocation method is also presented in this thesis. It is found that once sufficient resolution is achieved, numerical solutions are accurate despite the large condition numbers.

Acknowledgements

First of all, I want to thank my senior supervisor, Dr. Manfred R. Trummer for his guidance, encouragement and patience during the preparation of this thesis.

My thanks also go to Dr. Robert D. Russell, Dr. Margorzata Dubiel and Dr. Kathy Heinrich for their support.

Finally, I want to thank Graduate Secretary Mrs. Sylvia Holmes and other department staff. It was their kindness and help that made my stay at SFU comfortable.

A special thank is to the Department of Mathematics and Statistics of Simon Fraser University for its financial support.

Dedication

To My Parents

To the Memory of My Grandparents

To Great Chen

Contents

Abstract	iii
Acknowledgements	iv
Dedication	v
Contents	vii
List of Tables	vii
List of Figures	vii
1 Introduction	1
1.1 Pseudospectral Chebyshev Methods	1
1.2 Collocation Method	4
1.3 Thesis Plan	7
2 A Pseudospectral Chebyshev Method	9
2.1 Introduction	9
2.2 Chebyshev Polynomial Expansions	10
2.3 A Pseudospectral Chebyshev Method	12
2.4 Linear System Solver	15

2.5	Numerical Results	17
3	Preliminary Transform To The Circle	39
3.1	Introduction	39
3.2	The Method of Transformation to the Circle	40
3.3	Iterative Methods	44
3.3.1	CGNR (The Conjugate Gradient Iteration Applied to the Normal Equations)	44
3.3.2	CGS (The Conjugate Gradient Squared)	45
3.4	Numerical Results	47
4	Chebyshev Collocation Method	63
4.1	Chebyshev Collocation Method	63
4.2	Numerical Results	66
5	Discussion	74
	Bibliography	85

List of Tables

2.1	Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-2}$	25
2.2	Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-4}$	25
2.3	Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-6}$	25
2.4	Example 2, pseudospectral Chebyshev method, $\epsilon = 10^{-2}$	30
2.5	Example 2, pseudospectral Chebyshev method, $\epsilon = 10^{-4}$	30
2.6	Example 2, pseudospectral Chebyshev method, $\epsilon = 10^{-6}$	31
2.7	Example 3, pseudospectral Chebyshev method, $\epsilon = 10^{-2}$	35
2.8	Example 3, pseudospectral Chebyshev method, $\epsilon = 10^{-4}$	35
2.9	Example 3, pseudospectral Chebyshev method, $\epsilon = 10^{-6}$	35
3.1	Example 1, transformation to the circle method, $\epsilon = 10^{-2}$	54
3.2	Example 1, transformation to the circle method, $\epsilon = 10^{-4}$	55
3.3	Example 1, transformation to the circle method, $\epsilon = 10^{-6}$	55
3.4	Example 2, transformation to the circle method, $\epsilon = 10^{-2}$	60
3.5	Example 2, transformation to the circle method, $\epsilon = 10^{-4}$	60
3.6	Example 2, transformation to the circle method, $\epsilon = 10^{-6}$	60
4.1	Example 2, Chebyshev collocation method, $\epsilon = 10^{-2}$	66
4.2	Example 2, Chebyshev collocation method, $\epsilon = 10^{-4}$	66
4.3	Example 2, Chebyshev collocation method, $\epsilon = 10^{-6}$	66

5.1	Example 1, the condition number for two methods, $\epsilon = 10^{-2}$	79
5.2	Example 1, the condition number for two methods, $\epsilon = 10^{-4}$	79
5.3	Example 1, the condition number for two methods, $\epsilon = 10^{-6}$	79
5.4	Example 2, the condition number for two methods, $\epsilon = 10^{-2}$	79
5.5	Example 2, the condition number for two methods, $\epsilon = 10^{-4}$	80
5.6	Example 2, the condition number for two methods, $\epsilon = 10^{-6}$	80

List of Figures

1.1	Cubic B-spline functions	6
2.1	Distribution of Chebyshev collocation points, $N = 32$	13
2.2	Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-2}$, $N = 32$	19
2.3	Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-2}$, $N = 64$	20
2.4	Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-2}$, $N = 96$	20
2.5	Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-4}$, $N = 32$	21
2.6	Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-4}$, $N = 64$	21
2.7	Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-4}$, $N = 96$	22
2.8	Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-4}$, $N = 128$	22
2.9	Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-6}$, $N = 32$	23
2.10	Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-6}$, $N = 64$	23
2.11	Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-6}$, $N = 96$	24
2.12	Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-6}$, $N = 128$	24
2.13	Example 2, pseudospectral Chebyshev method, $\epsilon = 10^{-2}$, $N = 32$	27
2.14	Example 2, pseudospectral Chebyshev method, $\epsilon = 10^{-2}$, $N = 64$	27
2.15	Example 2, pseudospectral Chebyshev method, $\epsilon = 10^{-4}$, $N = 128$	28
2.16	Example 2, pseudospectral Chebyshev method, $\epsilon = 10^{-4}$, $N = 256$	28
2.17	Example 2, pseudospectral Chebyshev method, $\epsilon = 10^{-4}$, $N = 512$	29
2.18	Example 2, pseudospectral Chebyshev method, $\epsilon = 10^{-6}$, $N = 128$	29

2.19	Example 2, pseudospectral Chebyshev method, $\epsilon = 10^{-6}$, $N = 256$	30
2.20	Example 3, pseudospectral Chebyshev method, $\epsilon = 10^{-2}$, $N = 32$	32
2.21	Example 3, pseudospectral Chebyshev method, $\epsilon = 10^{-2}$, $N = 64$	32
2.22	Example 3, pseudospectral Chebyshev method, $\epsilon = 10^{-4}$, $N = 128$	33
2.23	Example 3, pseudospectral Chebyshev method, $\epsilon = 10^{-4}$, $N = 256$	33
2.24	Example 3, pseudospectral Chebyshev method, $\epsilon = 10^{-6}$, $N = 256$	34
2.25	Example 3, pseudospectral Chebyshev method, $\epsilon = 10^{-6}$, $N = 512$	34
2.26	Example 1, average error ($\cdot 10^{12}$) for $\epsilon = 10^{-2}$, $N = 32, 64, 96, 128$	36
2.27	Example 1, average error ($\cdot 10^2$) for $\epsilon = 10^{-4}$, $N = 32, 64, 96, 128$	36
2.28	Example 1, average error ($\cdot 10^2$) for $\epsilon = 10^{-6}$, $N = 32, 64, 96, 128$	37
2.29	Example 2, average error ($\cdot 10^3$) for $\epsilon = 10^{-2}$, $N = 32, 64, 96, 128$	37
2.30	Example 2, average error for $\epsilon = 10^{-4}$, $N = 128, 256, 512$	38
2.31	Example 2, average error for $\epsilon = 10^{-6}$, $N = 128, 256, 512$	38
3.1	Example 1, transformation to the circle method, $\epsilon = 10^{-2}$, $N = 32$	49
3.2	Example 1, transformation to the circle method, $\epsilon = 10^{-2}$, $N = 64$	49
3.3	Example 1, transformation to the circle method, $\epsilon = 10^{-2}$, $N = 96$	50
3.4	Example 1, transformation to the circle method, $\epsilon = 10^{-4}$, $N = 32$	50
3.5	Example 1, transformation to the circle method, $\epsilon = 10^{-4}$, $N = 64$	51
3.6	Example 1, transformation to the circle method, $\epsilon = 10^{-4}$, $N = 96$	51
3.7	Example 1, transformation to the circle method, $\epsilon = 10^{-4}$, $N = 128$	52
3.8	Example 1, transformation to the circle method, $\epsilon = 10^{-6}$, $N = 32$	52
3.9	Example 1, transformation to the circle method, $\epsilon = 10^{-6}$, $N = 64$	53
3.10	Example 1, transformation to the circle method, $\epsilon = 10^{-6}$, $N = 96$	53
3.11	Example 1, transformation to the circle method, $\epsilon = 10^{-6}$, $N = 128$	54
3.12	Example 2, transformation to the circle method, $\epsilon = 10^{-2}$, $N = 64$	57
3.13	Example 2, transformation to the circle method, $\epsilon = 10^{-4}$, $N = 128$	57

3.14	Example 2, transformation to the circle method, $\epsilon = 10^{-4}$, $N = 256$	58
3.15	Example 2, transformation to the circle method, $\epsilon = 10^{-6}$, $N = 128$	58
3.16	Example 2, transformation to the circle method, $\epsilon = 10^{-6}$, $N = 256$	59
3.17	Example 1, the pseudospectral method, $N = 96$, $\epsilon = 10^{-4}$	61
3.18	Example 1, the pseudospectral method, $N = 96$, $\epsilon = 10^{-4}$	61
3.19	Example 1, the circle technique, $N = 96$, $\epsilon = 10^{-4}$	62
3.20	Example 1, the circle technique, $N = 96$, $\epsilon = 10^{-4}$	62
4.1	Example 2, Chebyshev collocation method, $\epsilon = 10^{-2}$, $N = 32$	68
4.2	Example 2, Chebyshev collocation method, $\epsilon = 10^{-2}$, $N = 64$	68
4.3	Example 2, Chebyshev collocation method, $\epsilon = 10^{-2}$, $N = 96$	69
4.4	Example 2, Chebyshev collocation method, $\epsilon = 10^{-2}$, $N = 128$	69
4.5	Example 2, Chebyshev collocation method, $\epsilon = 10^{-2}$, $N = 256$	70
4.6	Example 2, Chebyshev collocation method, $\epsilon = 10^{-2}$, $N = 512$	70
4.7	Example 2, Chebyshev collocation method, $\epsilon = 10^{-4}$, $N = 96$	71
4.8	Example 2, Chebyshev collocation method, $\epsilon = 10^{-4}$, $N = 128$	71
4.9	Example 2, Chebyshev collocation method, $\epsilon = 10^{-4}$, $N = 256$	72
4.10	Example 2, Chebyshev collocation method, $\epsilon = 10^{-6}$, $N = 96$	72
4.11	Example 2, Chebyshev collocation method, $\epsilon = 10^{-6}$, $N = 128$	73
4.12	Example 2, Chebyshev collocation method, $\epsilon = 10^{-6}$, $N = 256$	73
5.1	Example 1, singular vector (u_1) versus Chebyshev points, $\epsilon = 10^{-2}$, $N = 96$	80
5.2	Example 1, singular vector (u_2) versus Chebyshev points, $\epsilon = 10^{-2}$, $N = 96$	81
5.3	Example 1, singular vector (u_3) versus Chebyshev points, $\epsilon = 10^{-2}$, $N = 96$	81
5.4	Example 1, singular vector (u_{93}) versus Chebyshev points, $\epsilon = 10^{-2}$, $N = 96$	82
5.5	Example 1, singular vector (u_{94}) versus Chebyshev points, $\epsilon = 10^{-2}$, $N = 96$	82
5.6	Example 1, singular vector (u_{95}) versus Chebyshev points, $\epsilon = 10^{-2}$, $N = 96$	83
5.7	Example 1, singular values $\epsilon = 10^{-2}$, $N = 96$	83

5.8 Example 1, coefficients $\{b_i\}$ of the equation (5.1) $\epsilon = 10^{-2}$, $N = 96$ 84

Chapter 1

Introduction

Numerical methods for solving ordinary and partial differential equations have always been important in scientific investigations. With the advent of computers, the use of numerical methods has been popularized, and more importantly, people are now able to attack those problems which are fundamental to our understanding of scientific phenomenon, but were so much more difficult to study in the past.

There are a large number of numerical methods available for use now, and among them, spectral methods have attracted much of the attention in current research on numerical methods. In this thesis, we plan to study two specific spectral methods for solving boundary value problems. In this Chapter, we first introduce the pseudospectral Chebyshev method, and then briefly mention a quite popular numerical method for solving boundary value problems, viz., the collocation method.

1.1 Pseudospectral Chebyshev Methods

Spectral methods are today not as widely used in the numerical solution of (two-point) boundary value problems (BVPs), even though they are very popular for time-dependent PDE's. However, for a variety of BVPs, they are competitive with the classical finite

difference, shooting, and collocation methods (see [15]). In spectral methods, the solution is assumed to be a finite linear combination of some set of global analytic basis functions, for example, Chebyshev polynomials. The differential equation yields then a system of equations for the coefficients.

These schemes can be very efficient because the rate of convergence or the order of accuracy as the number of modes increases depends only on the smoothness of the solution.

In particular, for an analytic solution of the differential equation, the error decays exponentially (see [15]). By contrast, for example, in finite difference methods, the order of accuracy is fixed by the scheme.

For the pseudospectral Chebyshev methods, the solution is discretized at the Chebyshev collocation points. The approximate solution is forced to satisfy the equation only at the Chebyshev collocation points. This method has the advantage of being able to deal more easily with nonlinear terms than the spectral method. The basic idea of pseudospectral Chebyshev methods consists of replacing exact derivatives by derivatives of interpolating polynomials at the Chebyshev points. Unfortunately, the pseudospectral Chebyshev methods lead to very ill-conditioned matrices ([4], [17]). The condition number of the first derivative matrix is proportional to N^2 , while the condition number of the second derivative related matrix increases like N^4 , where N is the number of collocation points.

In order to overcome the ill-conditioning an efficient preconditioner was suggested by Orszag [17]. The resulting condition number is nearly constant with respect to N [8]. Canuto and Quarteroni [6] proposed two classes of preconditioning matrices: one arising from finite difference, the other from finite elements. Further, Orszag [17] chose a different mesh for the preconditioner. An interesting alternative procedure was proposed in [8]: the preconditioning operator computes the first derivative at the intermediate grid points, then shifts the value to the original grid points.

Berrut [4] transfers the problem into an equivalent one on the circle, and then solves the problem on the circle with the pseudospectral Fourier method so that the matrix has

condition $O(N^2)$ for second order ordinary differential equations. We will study this “circle technique” in Chapter 3.

In this section, we will present the pseudospectral Chebyshev method. We consider a smooth function $u(x)$ in the domain $x \in [-1, 1]$. The Chebyshev collocation points are

$$x_i = \cos\left(\frac{\pi i}{N}\right), \quad i = 0, \dots, N,$$

which are the extrema of the N th order Chebyshev polynomial

$$T_N(x) = \cos(N \cos^{-1} x).$$

The function $u(x)$ is interpolated by a polynomial $P(x)$, ($P(x_i) = u(x_i) = u_i$), of degree $\leq N$,

$$P(x) = \sum_{j=0}^N u_j L_j(x),$$

where L_j is the polynomial of degree N with

$$L_j(x_k) = \delta_{jk}.$$

It can be shown [14] that

$$L_j(x) = \frac{(-1)^{j+1}(1-x^2)T'_N(x)}{c_j N^2(x-x_j)}, \quad j = 0, \dots, N,$$

where

$$c_j = \begin{cases} 2, & j = 0, N, \\ 1, & j = 1, \dots, N-1. \end{cases}$$

The derivative of $u(x)$ at the collocation points x_j can be approximated in many different ways. The most obvious way to compute the derivative is via matrix-vector multiplication [7]. The entries of the Chebyshev derivative matrix D_x are computed by taking the analytical derivative of $L_j(x)$ and evaluating it at the collocation points x_k for $j, k = 0, \dots, N$, i.e., $d_{kj} = L'_j(x_k)$. Then the entries of the matrix are

$$d_{kj} = \frac{c_k (-1)^{j+k}}{c_j (x_k - x_j)}, \quad j \neq k,$$

$$d_{kk} = -\frac{1}{2} \frac{x_k}{(1-x_k^2)}, \quad k \neq 0, N,$$

$$d_{00} = -d_{NN} = \frac{2N^2 + 1}{6},$$

and now the derivative of $u(x_i)$ becomes

$$u'_i = \sum_{j=0}^N d_{ij} u_j, \quad i = 0, \dots, N.$$

Some researchers have used other methods [21]. Variation of this matrix - vector multiplication algorithm are described in [21]; the other popular method uses the FFT (Fast Fourier Transform), and is asymptotically faster ($O(N \log N)$ operations) than a matrix vector multiply ($O(N^2)$).

1.2 Collocation Method

In this section, we will discuss a collocation method which in some sense is equivalent to the Runge-Kutta method; for an extensive treatment (see [1]). The important difference between this method and the pseudospectral method is that it uses basis functions with local support and limited differentiability.

We are given the linear equation

$$Lx = y, \tag{1.2.1}$$

where y is a given function. Approximating the solution $x(t)$ of (1.2.1) by the method of collocation consists of finding a function $x_N(t) \in \text{span}\{\phi_1, \dots, \phi_N\}$.

$$x_N(t) = a_1 \phi_1(t) + a_2 \phi_2(t) + \dots + a_N \phi_N(t)$$

by solving the $N \times N$ system of linear equations

$$Lx_N(t_i) = \sum_{j=1}^N a_j L\phi_j(t_i) = y(t_i) \quad 1 \leq i \leq N. \tag{1.2.2}$$

Here t_1, t_2, \dots, t_N are N distinct points of the domain at which all the terms of (1.2.2) are defined. The function $x_N(t)$, if it exists, is said to collocate $x(t)$ at the points t_1, \dots, t_N .

Any function $x_N(t)$ so obtained is referred to as an approximate solution obtained by the method of collocation.

For example, let us consider the linear boundary value problem

$$Lx(t) = x''(t) + p(t)x'(t) + q(t)x(t) = f(t), \quad 0 \leq t \leq 1$$

$$x(0) = x(1) = 0,$$

and we assume that this problem has a unique solution $x(t)$, and that $p(t)$ and $q(t)$ are continuous on $[0, 1]$. To approximate $x(t)$ by collocation using the cubic B-splines $\{B_i\}_{i=-1}^{n+1}$, let $\pi := 0 = t_0 < t_1 < \dots < t_n = 1$. Let $x_N \in \text{span}\{B_{-1}, B_0, \dots, B_{n+1}\}$. Using collocation with these approximating functions, we seek

$$x_N = a_{-1}B_{-1}(t) + a_0B_0(t) + \dots + a_{n+1}B_{n+1}(t) \quad (1.2.3)$$

such that

$$\begin{cases} x_N(0) = 0, \\ Lx_N(t_i) = f(t_i), & 0 \leq i \leq n \\ x_N(1) = 0. \end{cases}$$

Notice that we are collocating at $n+1$ knots and we force $x_N(t)$ to satisfy the same boundary data as $x(t)$. However, since we have $n+3$ basis functions, we have the same number of unknowns as equations.

To compute $x_N(t)$, we first use the linearity of L .

Thus for $0 \leq i \leq n$,

$$Lx_N(t_i) = \sum_{j=-1}^{n+1} a_j LB_j(t_i) = f(t_i), \quad (1.2.4)$$

where

$$LB_j(t_i) = B_j''(t_i) + p(t_i)B_j'(t_i) + q(t_i)B_j(t_i), \quad (1.2.5)$$

while

$$x_N(0) = a_{-1} + 4a_0 + a_1$$

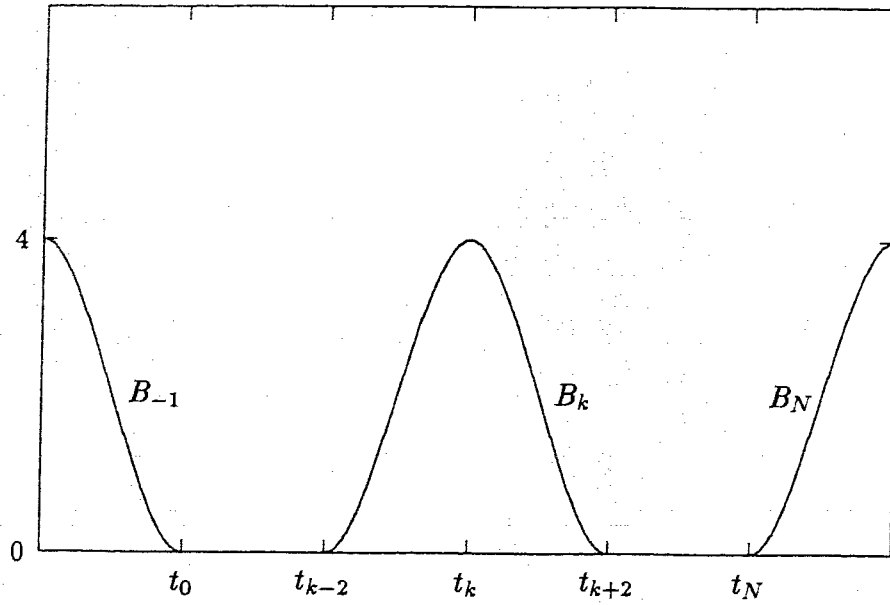


Figure 1.1: Cubic B-spline functions

and

$$x_N(1) = a_{n-1} + 4a_n + a_{n+1}.$$

Recalling that $B_i(t) = B'_i(t) = B''_i(t) = 0$ when $t \geq t_{i+2}$ and $t \leq t_{i-2}$ for each i , we see that (1.2.4) reduces to the linear system

$$C_n a = b, \tag{1.2.6}$$

where the coefficient matrix C_n is

iterative methods. The first iterative method is the conjugate gradient iteration applied to the normal equations, and the second iterative method is the biorthogonalization algorithm adapted from the biconjugate gradient iteration. We then apply the transformation method and the iterative methods to solve the three problems studied in Chapter 2.

In Chapter 4, we briefly discuss the Chebyshev collocation method and compare with the pseudospectral Chebyshev method discussed in Chapter 2.

In Chapter 5, we will discuss the numerical results obtained in the previous three Chapters comprehensively. Both merits and shortcomings with each method will be considered, together with some explanations and a mention of future research.

Emphasis in this thesis has been given to singularly perturbed BVPs controlled by a singularity parameter ϵ . Three kinds of singularities have been investigated: mild ($\epsilon = 10^{-2}$), stiff ($\epsilon = 10^{-4}$) and very stiff ($\epsilon = 10^{-6}$), and three types of layers in exact solutions have been considered: one layer at the center of the domain; two layers on the boundaries of the domain; and one layer on the left boundary of the domain. Numerical performance of the above mentioned methods have been judged in terms of condition numbers, average errors and cpu times. It is shown that numerical solutions are accurate despite the large condition number.

Chapter 2

A Pseudospectral Chebyshev Method

2.1 Introduction

There are many different numerical methods for solving differential equations, and spectral methods are now widely used. However, for boundary value problems (BVPs), the use of spectral method has not been studied in great detail. In fact, standard textbooks usually do not cover the spectral method or its variants for BVPs, although in the sixties polynomial collocation methods had received some attention for BVPs (see for example [23]). However, spline collocation methods became more popular, and renewed interest in spectral methods is more recent. In this Chapter, we study a specific variant of spectral methods, namely, the pseudospectral Chebyshev method.

The basic idea of pseudospectral Chebyshev methods consists of replacing exact derivatives by derivatives of interpolating polynomials at the Chebyshev points in the domain. Improving computational efficiency of spectral methods is one of the main purposes of the current investigation into these methods. Discrete Chebyshev differentiation can be represented by matrices, and in practice, these matrices are severely ill-conditioned and full.

Thus, it is important to find efficient techniques for solving the algebraic systems. The condition number of the matrix related to the N -point Chebyshev pseudospectral approximation of the first derivative operator is proportional to N^2 , while for the matrix related to the second derivative operator, the condition number increases like N^4 . Iterative methods and preconditioning techniques have been proposed to solve these spectral algebraic systems by Orszag [17], Canuto and Quarteroni [6] and other authors. We will discuss iterative methods in Chapter 3. In this Chapter, we consider the direct method, namely, the Gaussian elimination method.

2.2 Chebyshev Polynomial Expansions

First, we review several results from approximation theory. The Chebyshev polynomial of degree k ($k = 0, 1, \dots$) on $[-1, 1]$ is defined by the formula

$$T_k = \cos(k \cos^{-1} x). \quad (2.2.1)$$

Clearly, $|T_k(x)| \leq 1$ for $x \in [-1, 1]$. The T_k are indeed polynomials in x . For example,

$$T_0(x) = 1, \quad T_1(x) = x, \quad (2.2.2)$$

by definition, and using elementary trigonometric identities, we can obtain the recursion

$$T_{k+1}(x) = 2xT_k(x) - T_{k-1}(x) \quad k \geq 1. \quad (2.2.3)$$

Let $L^2_{[-1,1]}$ be the space of square integrable functions defined on $[-1, 1]$. Then the functions T_k constitute an orthonormal basis with respect to the inner product

$$(f, g) = \int_{-1}^1 f(x)g(x)(1-x^2)^{-1/2} dx \quad (2.2.4)$$

in $L^2_{[-1,1]}$. The convergence theory of Chebyshev polynomial expansions is very similar to that of Fourier cosine series. In fact, suppose for $f(x) \in L^2_{[-1,1]}$, we write formally

$$g(x) = \sum_{k=0}^{\infty} a_k T_k(x), \quad (2.2.5)$$

the so-called the Chebyshev series associated with $f(x)$, where a_k is to be determined, then $G(\theta) := g(\cos \theta)$ is the Fourier cosine series of $F(\theta) := f(\cos \theta)$ for $0 \leq \theta \leq \pi$. This result follows from the definition of T_k , because $T_k(\cos \theta) = \cos(k\theta)$, $G(\theta) = g(\cos \theta) = \sum_{k=0}^{\infty} a_k \cos(k\theta)$. Thus,

$$a_k = \frac{2}{\pi c_k} \int_0^{\pi} f(\cos \theta) \cos(k\theta) d\theta = \frac{2}{\pi c_k} \int_{-1}^1 f(x) T_k(x) (1-x^2)^{-1/2} dx, \quad (2.2.6)$$

where $c_0 = 2$, $c_k = 1$ ($k > 0$).

It follows from this close relation between Chebyshev series and Fourier cosine series that if $f(x)$ is piecewise continuous and if $f(x)$ is of bounded total variation for $-1 \leq x \leq 1$ then $g(x) = \frac{1}{2}[f(x+) + f(x-)]$ for each x ($-1 < x < 1$) and $g(1) = f(1-)$, $g(-1) = f(-1+)$. Also, if $f^{(p)}(x)$ is continuous for all $|x| \leq 1$ for $p = 0, 1, \dots, n-1$, and $f^{(n)}(x)$ is integrable, then

$$a_k = O(k^{-n}). \quad (2.2.7)$$

Since $|T_k(x)| \leq 1$ for $|x| \leq 1$, it follows that the remainder after k terms of the Chebyshev series (2.2.5) is asymptotically much smaller than $k^{-(n-1)}$ as $k \rightarrow \infty$. If $f(x)$ is infinitely differentiable for $|x| \leq 1$, the error in the Chebyshev series goes to zero more rapidly than any finite power of k^{-1} as $k \rightarrow \infty$ [15].

The most important feature of Chebyshev series is that their convergence properties are not affected by the values of $f(x)$ or its derivatives at the boundaries $x = \pm 1$, but only by the smoothness of $f(x)$ and its derivatives throughout $-1 < x < 1$. In contrast, the Gibbs phenomenon shows that the rate of convergence of Fourier series depends on the value of $f(x)$ and its derivatives at the boundaries in addition to the smoothness of $f(x)$ and its derivative in the interior of the interval. The reason for the absence of a Gibbs phenomenon for the Chebyshev series of $f(x)$ at $x = \pm 1$ is due to the fact that $F(\theta) = f(\cos \theta)$ satisfies $F_{2p+1}(0) = F_{2p+1}(\pi) = 0$ provided only that all derivatives of $f(x)$ of order at most $2p+1$ exist at $x = \pm 1$.

An important consequence of the rapid convergence of Chebyshev polynomial expansions

of smooth functions is that Chebyshev expansions may normally be differentiated termwise.

Since

$$\frac{d^p}{dx^p} T_k(x) = O(k^{2p}), \quad k \rightarrow \infty, \quad (2.2.8)$$

uniformly for $|x| \leq 1$, if $a_k \rightarrow 0$ faster than any finite power of k^{-1} as $k \rightarrow \infty$, then (2.2.5) may be differentiated formally [15]:

$$\frac{d^p g(x)}{dx^p} = \sum_{k=0}^{\infty} a_k \frac{d^p T_k(x)}{dx^p}. \quad (2.2.9)$$

2.3 A Pseudospectral Chebyshev Method

In this section, we consider a pseudospectral Chebyshev method for the linear second order two-point boundary value problem,

$$\begin{cases} \epsilon U''(x) + p(x)U' + q(x)U(x) = f(x), \\ U(1) = U_0, \quad U(-1) = U_N, \end{cases} \quad (2.3.10)$$

where ϵ is a parameter that controls the singularity of the problem. Our discussion will follow Berrut [4], who formulated the algorithm for $\epsilon = 1$. Suppose that the problem is well-posed and in particular that it has a unique solution. The Chebyshev interpolation polynomial can be written as

$$\bar{U}(x) = \sum_{j=0}^N \bar{U}_j L_j(x), \quad (2.3.11)$$

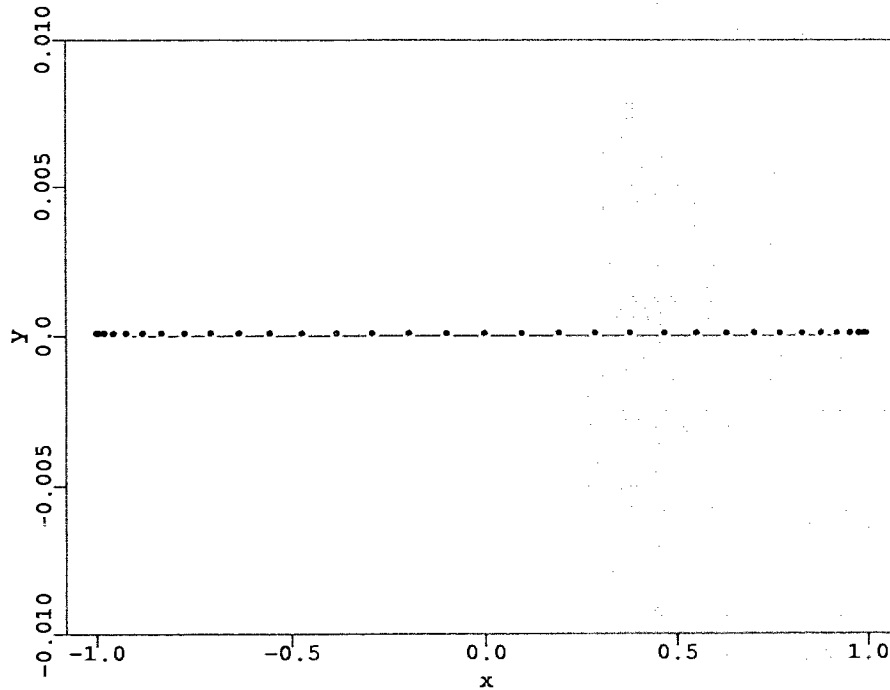
where

$$x_j = \cos \frac{j\pi}{N}, \quad j = 0, 1, \dots, N$$

are the interpolating points (also called collocation points), $\{\bar{U}_j\}_{j=1}^{N-1}$ are the unknown coefficients to be determined, and $L_j(x)$ is the Lagrange interpolation polynomial associated with x_j , that is,

$$L_j(x) = \prod_{i=0, i \neq j}^N \frac{(x - x_i)}{(x_j - x_i)}. \quad (2.3.12)$$

Since $L_j(x_k) = \delta_{jk}$ (δ_{jk} is the Kronecker delta), it follows that $\bar{U}(x_j) = \bar{U}_j$.

Figure 2.1: Distribution of Chebyshev collocation points, $N = 32$

Here we analyze the relation between the number of the collocation points and the width $O(\epsilon)$ of the boundary layer.

Since

$$\begin{aligned}\Delta x_1 &= |x_1 - x_0| = |\cos(\pi/N) - 1| \\ &= |1 - \sin(0)(\pi/N) - \frac{1}{2}\cos(0)(\pi/N)^2 + \dots - 1| \\ &\approx \frac{1}{2}\frac{\pi^2}{N^2} \approx \frac{5}{N^2}, \\ \Delta \bar{x} &= |\cos(\frac{\pi}{N} + \frac{\pi}{2}) - 0| \approx \frac{\pi}{N}.\end{aligned}$$

Therefore we can see that the spacing between the collocation points near the boundary is $\Delta x_1 \approx O(\frac{1}{N^2})$ and the spacing near the center of the interval is $\Delta \bar{x} \approx O(\frac{1}{N})$, while the width of the boundary layer is $O(\epsilon)$. For good resolution of our numerical solutions at least one of the collocation points should lie in the boundary layer (or any transition layer). Therefore we need $5/N^2 < \epsilon$, i.e., $N > \sqrt{5}/\sqrt{\epsilon}$. If the width of the boundary layer was exactly ϵ , we should use $N \approx 2200, 220$ and 22 for $\epsilon = 10^{-6}, 10^{-4}$ and 10^{-2} respectively.

In our numerical computations we never used N larger than 1024, and therefore, were not able to resolve the boundary layers for $\epsilon = 10^{-6}$. See [24] for a method which achieves resolution for smaller values of N in the case of small ϵ .

The Chebyshev method substitutes the interpolation polynomial (2.3.11) into (2.3.10), and then replaces $\bar{U}(x_0)$ by U_0 and $\bar{U}(x_N)$ by U_N . Doing this, we have

$$\begin{cases} \epsilon \bar{U}''(x) + p(x)\bar{U}'(x) + q(x)\bar{U}(x) = f(x), \\ \bar{U}(1) = U_0, \quad \bar{U}(-1) = U_N. \end{cases} \quad (2.3.13)$$

Using collocation points $\{x_j\}_{j=1}^{N-1}$, we have collocation equations

$$\begin{cases} \epsilon \bar{U}''(x_j) + p(x_j)\bar{U}'(x_j) + q(x_j)\bar{U}(x_j) = f(x_j), \\ \bar{U}(1) = U_0, \quad \bar{U}(-1) = U_N. \end{cases} \quad (2.3.14)$$

Denoting by D_x the matrix of the first derivatives $L'_i(x_j)$ ($i, j = 0, 1, \dots, N$), and letting $\bar{U}'(x_j) = \bar{U}'_j$ ($j = 0, 1, \dots, N$), $\bar{U}(x_j) = \bar{U}_j$ ($j = 0, 1, \dots, N$) gives

$$\begin{bmatrix} \bar{U}'_0 \\ \vdots \\ \bar{U}'_N \end{bmatrix} = D_x \begin{bmatrix} \bar{U}_0 \\ \vdots \\ \bar{U}_N \end{bmatrix},$$

where

$$\begin{aligned} D_x &= (L'_i(x_j)) = (d_{ij}), \\ d_{ij} &= \frac{(-1)^{j+i}}{x_i - x_j} \cdot \frac{c_i}{c_j}, \quad i \neq j \\ d_{00} &= \frac{2N^2 + 1}{6} = -d_{NN}, \\ d_{ii} &= \frac{-x_i}{2(1 - x_i^2)}, \quad 1 \leq i \leq N - 1 \\ c_0 &= c_N = 2, \\ c_i &= 1, \quad 1 \leq i \leq N - 1. \end{aligned}$$

In view of the fact that the matrix of the second derivatives $L_j''(x_i)$ is the square of D_x (see [4]), we obtain the system of linear equations

$$\sum_{j=1}^{N-1} [\epsilon (D_x^2)_{ij} + p_i (D_x)_{ij} + q_i \delta_{ij}] \bar{U}(x_i) = f(x_i) - [\epsilon (D_x^2)_{i0} + p_i (D_x)_{i0}] U_0 - [\epsilon (D_x^2)_{iN} + p_i (D_x)_{iN}] U_N \quad (2.3.15)$$

for the $\bar{U}(x_i)$, $i = 1, 2, \dots, N-1$, with $p_i := p(x_i)$, $q_i := q(x_i)$. The largest eigenvalue modulus of D_x^2 is known to grow with N like $O(N^4)$ (see [4]), and in the example we consider in section 2.5, the numerical value of that modulus grows about that fast.

For simplicity, we write equation (2.3.15) in matrix form

$$A\bar{U} = b \quad (2.3.16)$$

where

$$A = (a_{ij}) \in R^{(N-1) \times (N-1)},$$

with

$$a_{ij} = \epsilon (D_x^2)_{ij} + p_i (D_x)_{ij} + q_i \delta_{ij}.$$

$$\bar{U} = (\bar{U}_1, \dots, \bar{U}_{N-1})^T \in R^{N-1}.$$

$$b = (b_1, b_2, \dots, b_{N-1})^T \in R^{N-1},$$

with

$$b_i = f(x_i) - [\epsilon (D_x^2)_{i0} + p_i (D_x)_{i0}] U_0 - [\epsilon (D_x^2)_{iN} + p_i (D_x)_{iN}] U_N.$$

For a fixed integer N , solving system (2.3.16) for \bar{U} and substituting \bar{U} , U_0 and U_N in to (2.3.11) will lead to our numerical solution $\bar{U}(x)$ to the BVP (2.3.10).

2.4 Linear System Solver

The LINPACK [7] subroutines operate on general square nonsymmetric matrices. The operations performed include the triangular factorization of matrices, the estimation of the

matrix condition number, the solution of simultaneous linear equations and the calculation of determinants and inverses. Let A be a real or complex square matrix of order n . There is an upper triangular matrix U and a matrix L which is the product of elementary lower triangular and permutation matrices such that $A = LU$. This factorization can be used to solve linear equations $Ax = b$ by solving successively $L(Ux) = b$ to compute the inverse of A as $A^{-1} = U^{-1}L^{-1}$.

The condition number $k(A)$ is a quantity which measures the sensitivity of the solution x to errors in the matrix A and the right hand side b . If the relative error in A is of size ϵ , then the resulting relative error in x can be as large as $k(A)\epsilon$. Error in A can arise in many ways. In particular, the effect of the roundoff error introduced by the subroutines in this Chapter can usually be assessed by taking ϵ to be a small multiple of the rounding unit.

It is possible to efficiently compute a quantity $RCOND$ which is an estimate of the reciprocal condition, $1/k(A)$. If exact singularity is detected, $RCOND$ may be set to 0.

If A is badly scaled, then the interpretation of $RCOND$ is more delicate.

We use the double precision, general subroutines $DGECO$, $DGEFA$ to solve linear system $Ax = b$. $DGECO$ is usually called first to factor the matrix and estimate its condition. The actual factorization is done by $DGEFA$ which can be called in place of $DGECO$ if the condition estimate is not needed. The time required by $DGECO$ is roughly $(1 + 9/N)$ times the time required by $DGEFA$. Thus when $N = 9$, $DGECO$ costs twice as much as $DGEFA$, but when $N = 90$, $DGECO$ costs only 10 percent more.

Roundoff errors in floating point arithmetic operations usually cause the quantities computed by the subroutines in this Chapter to be somewhat inaccurate. The following fairly vague statements give a general idea of the extent of these inaccuracies.

$DGEFA$ produces matrices L and U for which the product LU is almost always within roundoff error of A , no matter how close A is to being singular.

$DGECO$ has experimentally been shown to produce an estimate $RCOND$ for which $1/RCOND$ is usually the same order of magnitude as the actual condition number.

To make these statements more precise, we introduce the following notation. Let a_j be the columns of A . For $x \in R^n$, define

$$\|x\|_1 = \sum_{j=1}^n |x_j|,$$

$$\|Ax\|_1 = \sum_{i=1}^n \left| \sum_{j=1}^n a_{ij} x_j \right| \leq \sum_{i=1}^n \max_j |a_{ij}| \sum_{j=1}^n |x_j| \leq \left(\sum_{i=1}^n \max_j |a_{ij}| \right) \|x\|_1$$

$$\|A\|_1 = \max_x \frac{\|Ax\|_1}{\|x\|_1} = \max_j \|a_j\|_1.$$

Then the condition number of A with respect to $\|\cdot\|_1$ is

$$k_1(A) = \|A\|_1 \|A^{-1}\|_1.$$

If errors are measured in the usual Euclidean norm,

$$\|x\|_2 = \left(\sum_{j=1}^n x_j^2 \right)^{1/2},$$

then the condition number is

$$k_2(A) = \|A\|_2 \|A^{-1}\|_2 = \frac{\sigma_1(A)}{\sigma_n(A)}.$$

Here $\sigma_1(A)$ and $\sigma_n(A)$ are the largest and smallest singular value of A . $k_1(A)$ and $k_2(A)$ are different, but are usually of the same order of magnitude.

2.5 Numerical Results

The pseudospectral Chebyshev method discussed in section 2.3 was seldom applied to solve singularly perturbed BVPs. For such problems, we need to consider two regions: narrow regions of very fast variation (so-called boundary or interior layers), and wider regions of slow variation. The concept of singularly perturbed BVP relates to the concept of stiff BVP in numerical analysis. All the computations in this thesis were done on a Sparc station. In our Tables, the average errors are computed by

$$\text{Average Error} = \frac{1}{N} \sum_{j=1}^N |u(x_j) - u_{exact}(x_j)|,$$

where N is the number of terms used in the Chebyshev series, $u(x)$ is the numerical solution and u_{exact} is an exact solution. The condition number (RCOND) of A is obtained by LINPACK [7]

$$k_1(A) = \|A\|_1 \|A^{-1}\|_1.$$

Example 1.

Consider

$$\begin{cases} \epsilon U''(x) + xU'(x) = -\epsilon\pi^2 \cos(\pi x) \sin(\pi x), & -1 \leq x \leq 1 \\ U(-1) = -2, \quad U(1) = 0, \end{cases} \quad (2.5.17)$$

whose solution is

$$U(x) = \cos(\pi x) + \operatorname{erf}(x/\sqrt{2\epsilon})/\operatorname{erf}(1/\sqrt{2\epsilon}), \quad (2.5.18)$$

where

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x \exp(-t^2) dt$$

is the error function. For $0 \leq \epsilon \ll 1$ this solution has a rapid transition layer at $x = 0$.

COLSYS [1] has no problem for $\epsilon = 10^{-4}$ to reach error $O(10^{-9})$ with mesh points $N = 132$, but if the problem is run using no mesh selection, comparable accuracy is not achieved until $N = 1280$ [1]. The interior layer gets resolved well, because one of the collocation points is exactly at the location of the layer. Usually interior layers pose more difficulty.

We first tested $\epsilon = 10^{-2}$. For $N = 32, 64$ and 96 , we plotted our numerical solutions (the solid-dotted lines) and the exact solution (the dotted line) in Figure 2.1 to Figure 2.3. As can be seen, our numerical solutions captured all the properties of the exact solution for N as small as 64 . See Table 2.1 for a summary of the related condition numbers, average errors and cpu times. It is interesting that for $\epsilon = 10^{-2}$ the condition numbers of the matrix are much smaller than the theoretical $O(N^4)$.

Next, we tested $\epsilon = 10^{-4}$. In this case, the rapid transition of the exact solution at $x = 0$ showed up moderately. We plotted our numerical solutions in Figure 2.4 to Figure 2.7 for $N = 32, 64, 96$ and 128 , respectively. The approximation to the exact solution improves as

N increases. However, even when $N = 128$, there is noticeable oscillation near $x = 0$. Table 2.2 contains the relevant numerical summary. Still, the condition numbers are smaller than $O(N^4)$.

Finally, we tested $\epsilon = 10^{-6}$. In this case, the exact solution has a sharp change at $x = 0$, and it took N as large as 128 to reach a fairly good numerical solution. For this stiff problem (i.e. $\epsilon = 10^{-6}$), see Figure 2.8 to Figure 2.11 and Table 2.3.

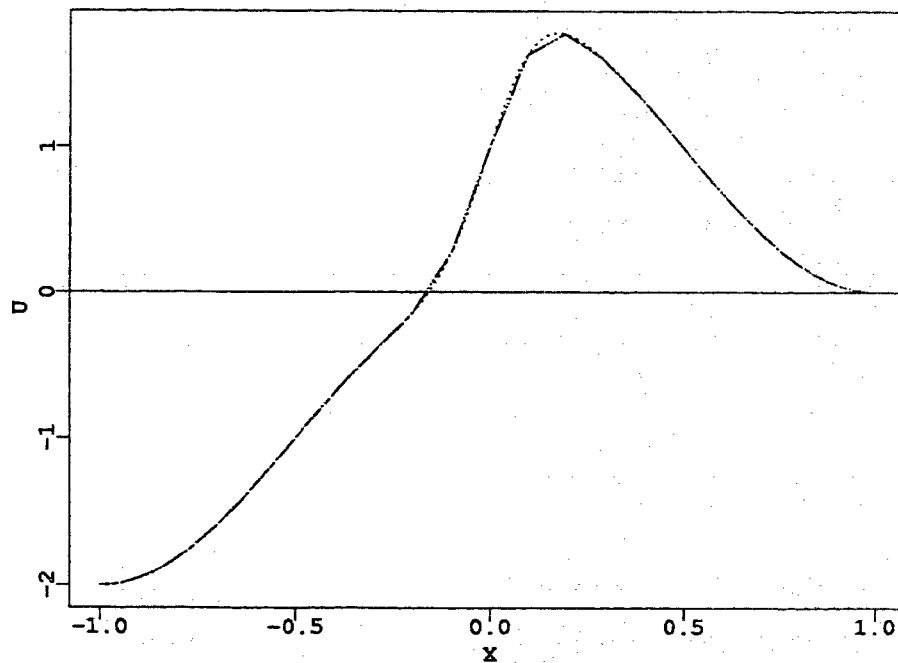


Figure 2.2: Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-2}$, $N = 32$

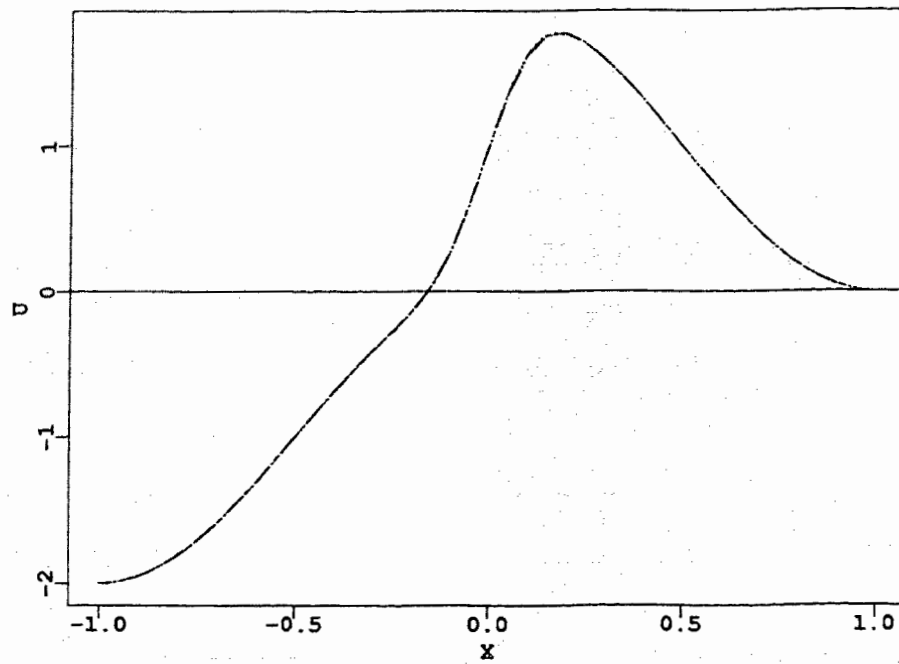


Figure 2.3: Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-2}$, $N = 64$

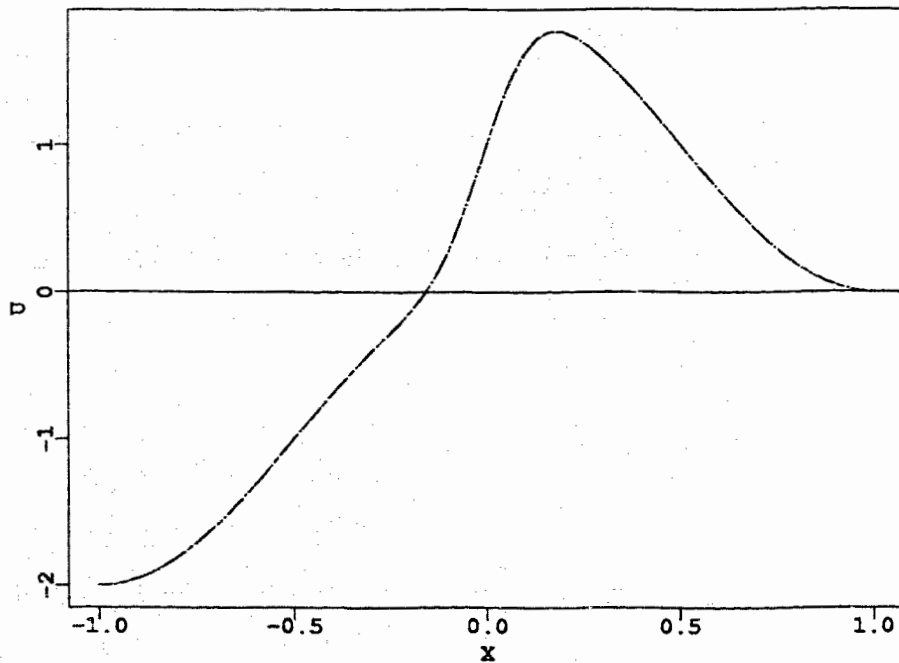


Figure 2.4: Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-2}$, $N = 96$

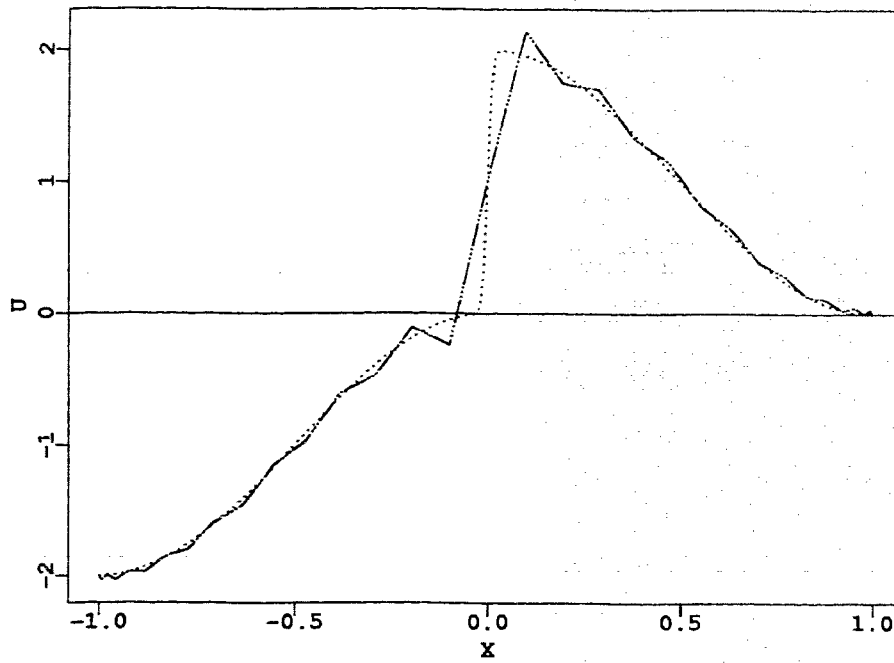


Figure 2.5: Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-4}$, $N = 32$

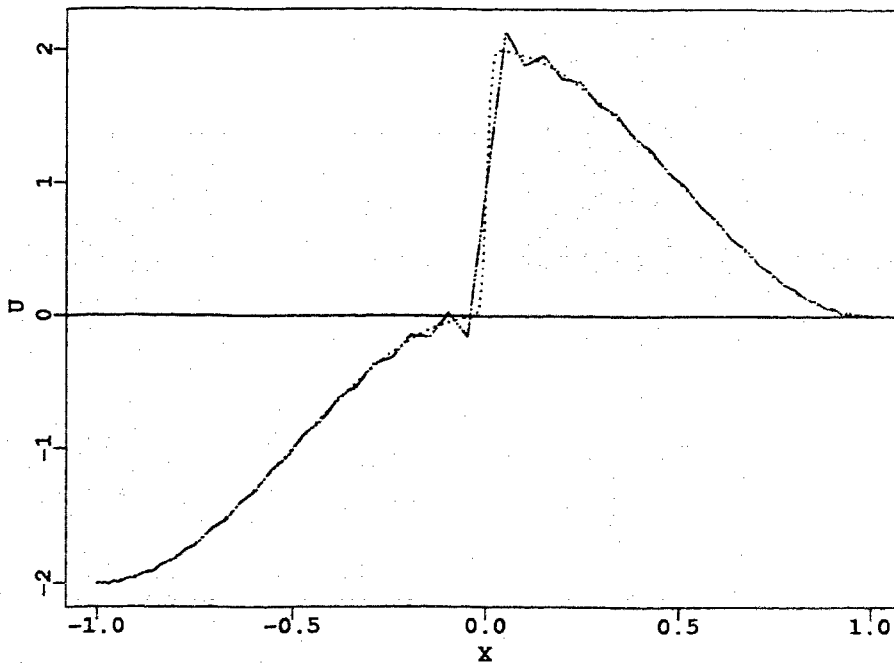


Figure 2.6: Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-4}$, $N = 64$

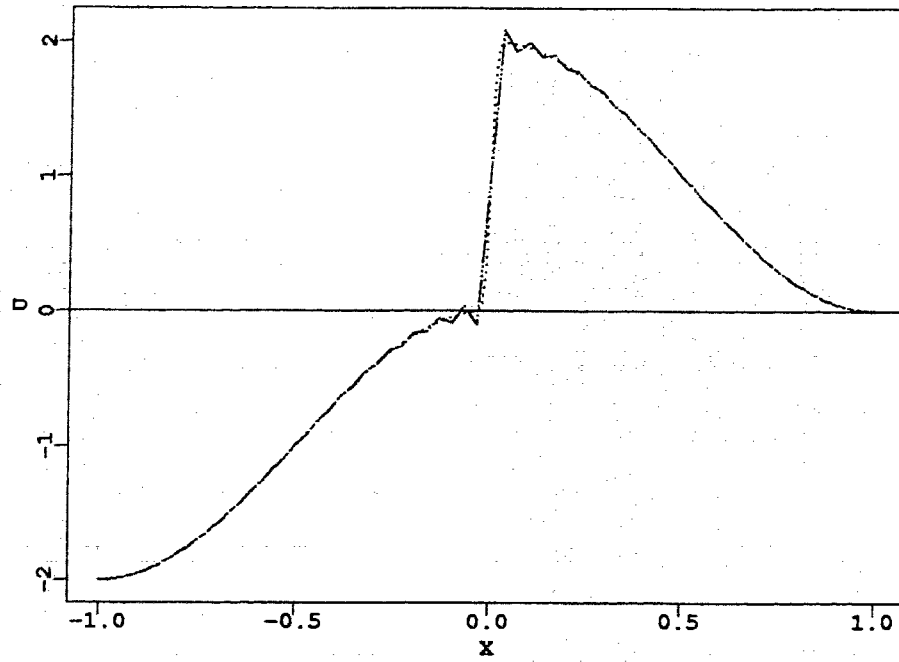


Figure 2.7: Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-4}$, $N = 96$

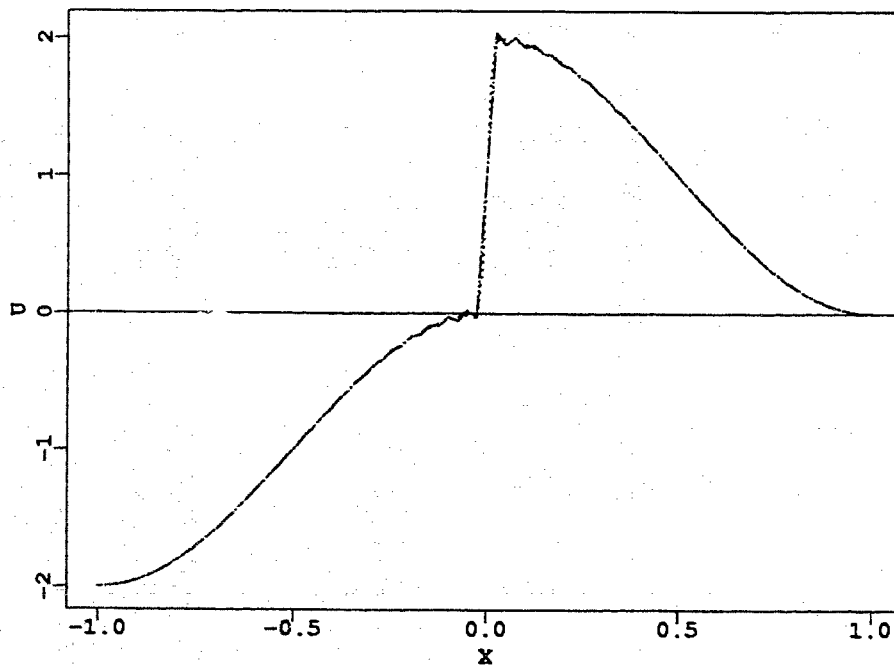


Figure 2.8: Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-4}$, $N = 128$

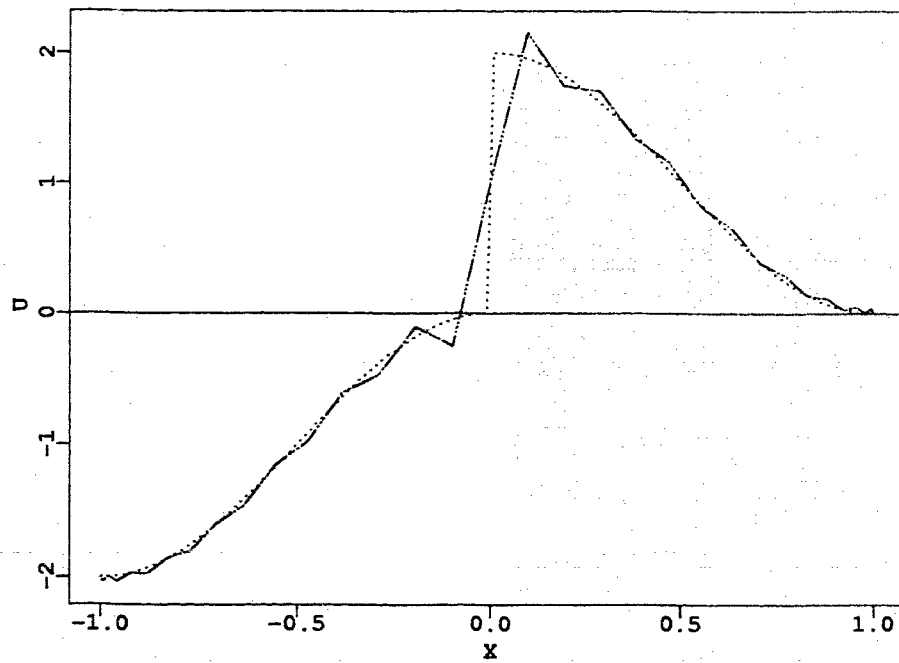


Figure 2.9: Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-6}$, $N = 32$

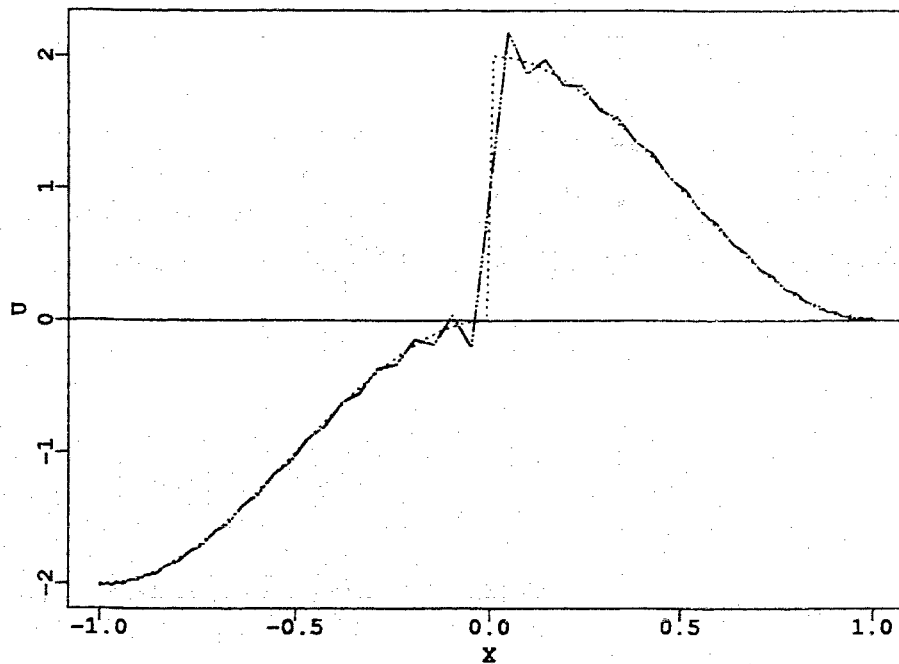


Figure 2.10: Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-6}$, $N = 64$

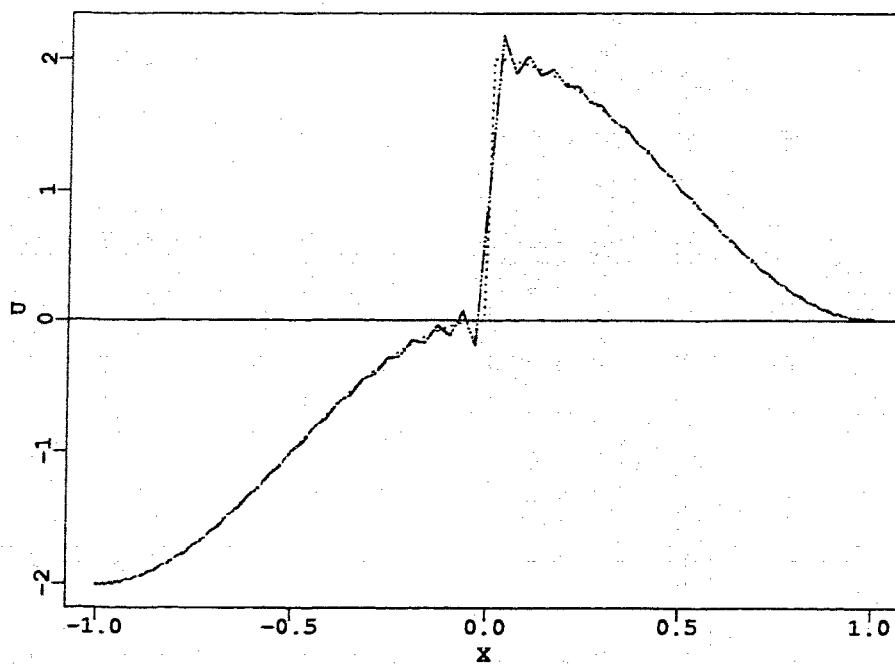


Figure 2.11: Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-6}$, $N = 96$

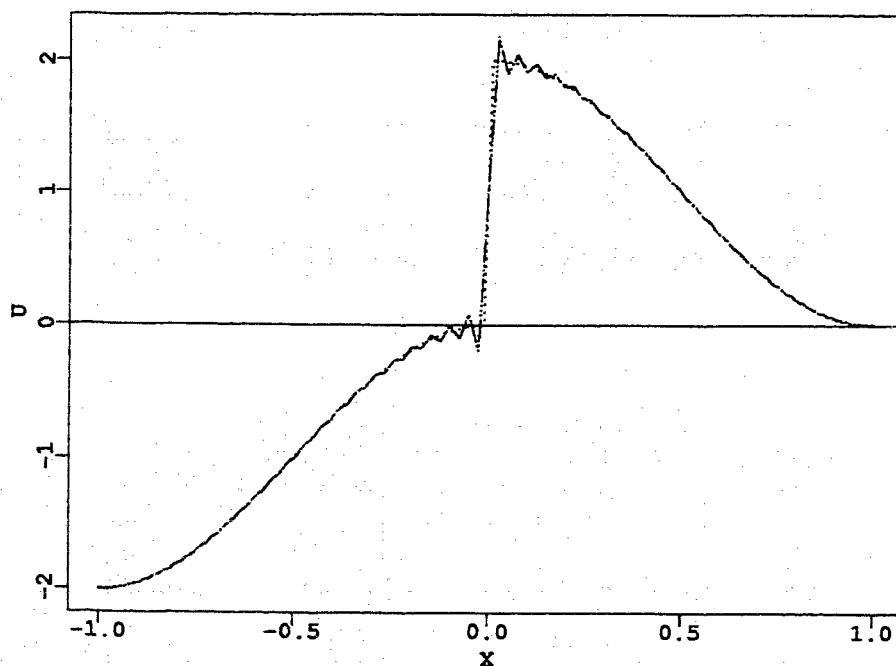


Figure 2.12: Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-6}$, $N = 128$

N	Condition Number	Average Error	Cpu-Time
32	6.4×10^2	1.23×10^{-4}	0.3
64	8.3×10^3	4.18×10^{-11}	1.58
96	1.4×10^5	4.59×10^{-14}	5.62
128	1.2×10^5	6.42×10^{-14}	12.53

Table 2.1: Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-2}$

N	Condition Number	Average Error	Cpu-Time
32	4.5×10^4	4.38×10^{-2}	0.28
64	7.7×10^4	2.15×10^{-2}	1.58
96	1.3×10^5	1.17×10^{-2}	5.68
128	1.4×10^5	6.19×10^{-3}	12.66

Table 2.2: Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-4}$

N	Condition Number	Average Error	Cpu-Time
32	5.8×10^6	4.65×10^{-2}	0.41
64	1.2×10^7	2.69×10^{-2}	1.88
96	1.8×10^7	1.95×10^{-2}	5.68
128	2.4×10^7	1.54×10^{-2}	12.64

Table 2.3: Example 1, pseudospectral Chebyshev method, $\epsilon = 10^{-6}$

Example 2.

As the second example, we considered the BVP

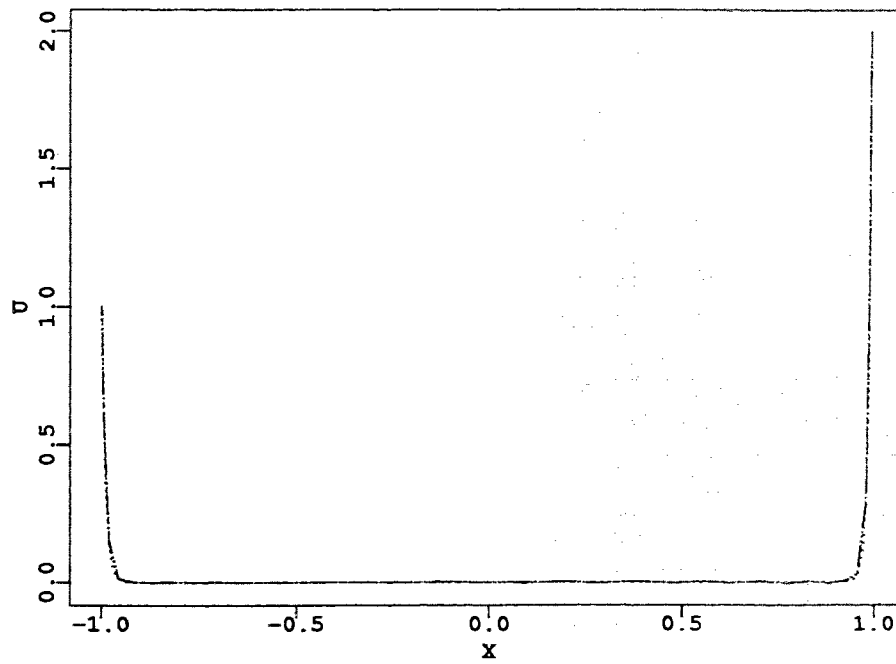
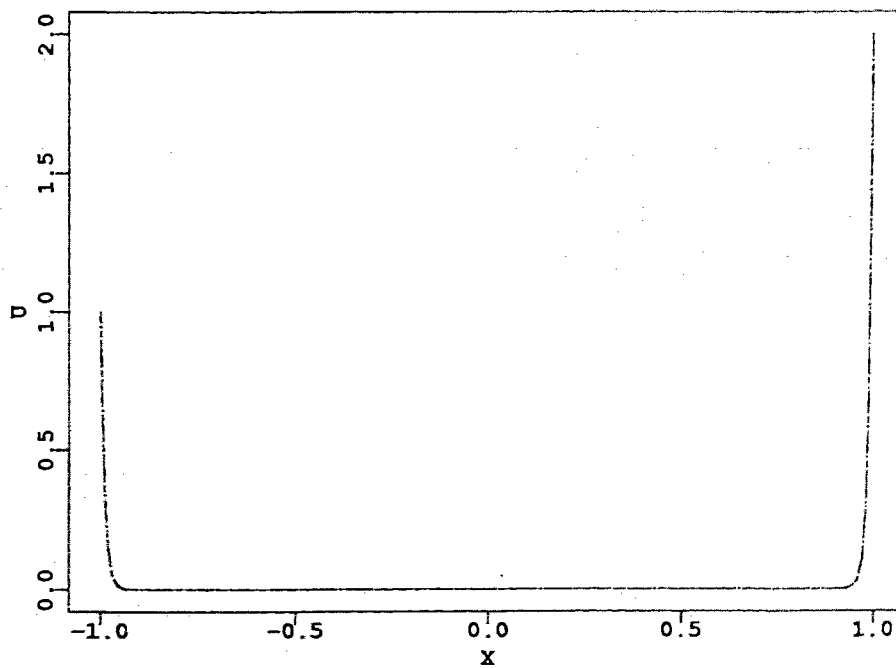
$$\begin{cases} \epsilon U''(x) = xU'(x) + \frac{1}{2}U(x), & -1 \leq x \leq 1, \\ U(-1) = 1, \quad U(1) = 2, \end{cases} \quad (2.5.19)$$

which has the exact solution

$$u(x) = \epsilon^{-(x+1)/\epsilon} + 2e^{(x-1)/\epsilon} + O(\epsilon), \quad (2.5.20)$$

where $O(\epsilon)$ denotes an quantity which can be bounded by a constant times ϵ . This solution has two boundary layers at the ends, and is smooth near the “turning point” $x = 0$. The numerical error propagates from the boundaries to the middle of the interval $[-1, 1]$. This equation has tendency for an interior layer. Maybe oscillations in interior are related to this fact. COLSYS succeeded in solving this problem for $\epsilon = 10^{-6}$ although the error is not too small (See [1]).

Again, we considered three different cases corresponding to $\epsilon = 10^{-2}$, $\epsilon = 10^{-4}$, and $\epsilon = 10^{-6}$, respectively. When $\epsilon = 10^{-2}$, accurate numerical approximation to the true solution was easy to obtain; see Figure 2.12, Figure 2.13 and Table 2.4. When $\epsilon = 10^{-4}$, it required a relatively large N to obtain a reasonable numerical solution; see Figure 1.14 to Figure 2.16 for $n = 128, 256, 512$, respectively and Table 2.5. However, for $N = 512$, a fairly good numerical solution was achieved. When $\epsilon = 10^{-6}$, it was difficult to reach good numerical solutions even for large N . It is shown that the pseudospectral Chebyshev method fails to solve this two boundary layers problem with small ϵ (e.g. 10^{-6}), while it succeeds in solving it with $\epsilon = 10^{-4}$. See Figures 2.17 and 2.18. and Table 2.6. However, it appears reasonable to expect highly accurate solution for sufficiently large N , even for small ϵ ($\epsilon = 10^{-6}$).

Figure 2.13: Example 2, pseudospectral Chebyshev method, $\epsilon = 10^{-2}$, $N = 32$ Figure 2.14: Example 2, pseudospectral Chebyshev method, $\epsilon = 10^{-2}$, $N = 64$

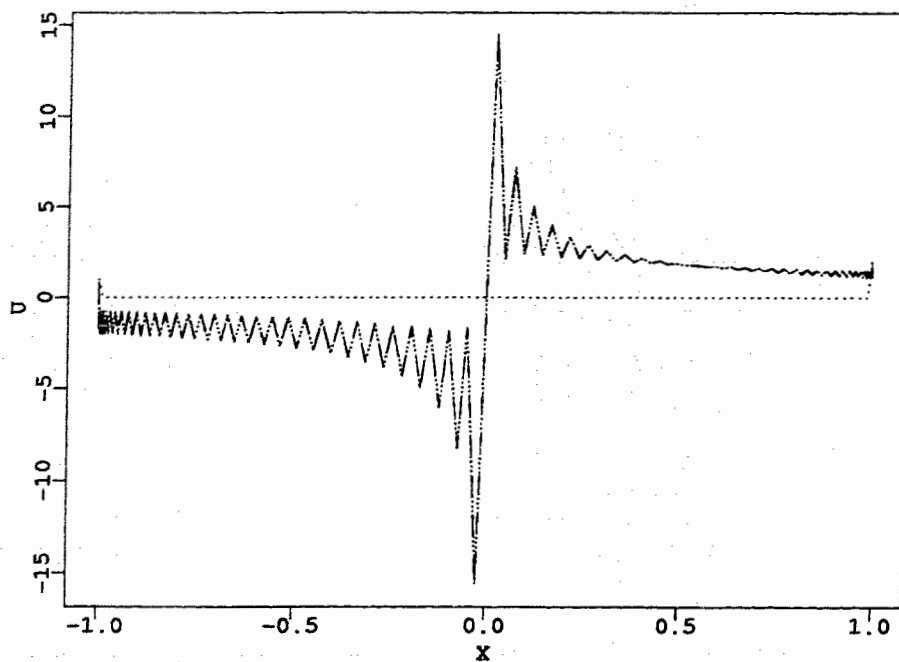


Figure 2.15: Example 2, pseudospectral Chebyshev method, $\epsilon = 10^{-4}$, $N = 128$

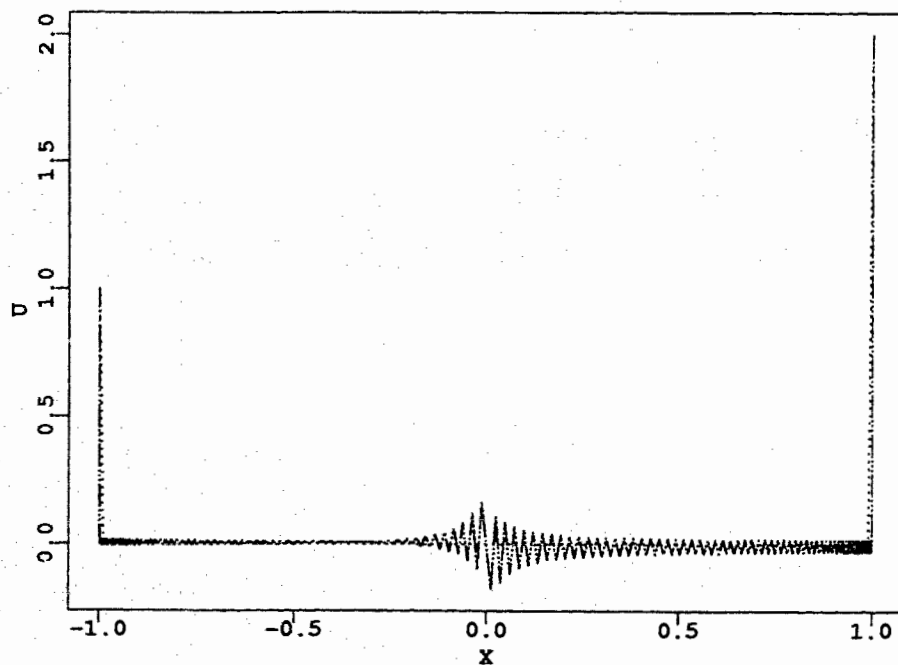


Figure 2.16: Example 2, pseudospectral Chebyshev method, $\epsilon = 10^{-4}$, $N = 256$

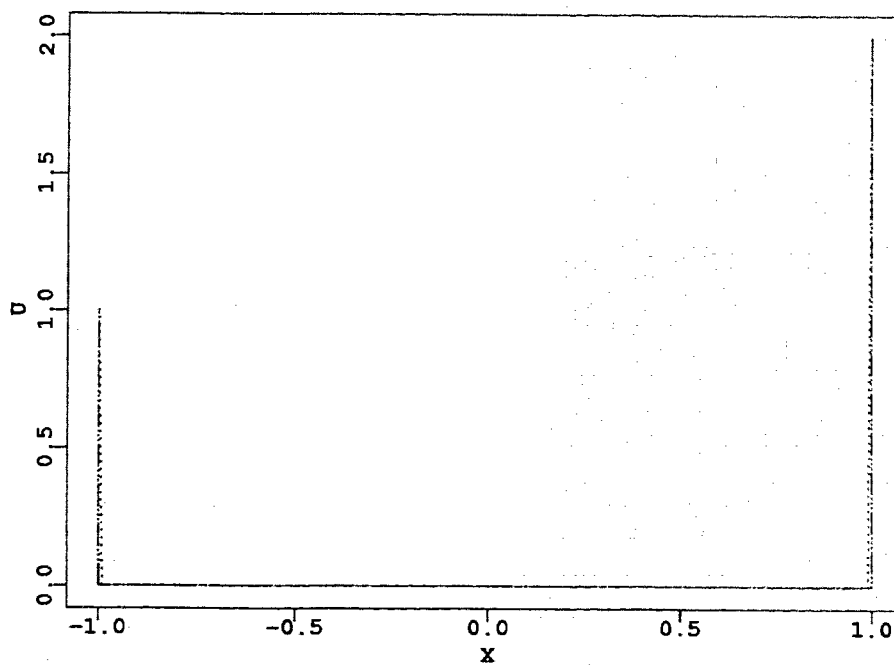


Figure 2.17: Example 2, pseudospectral Chebyshev method, $\epsilon = 10^{-4}$, $N = 512$

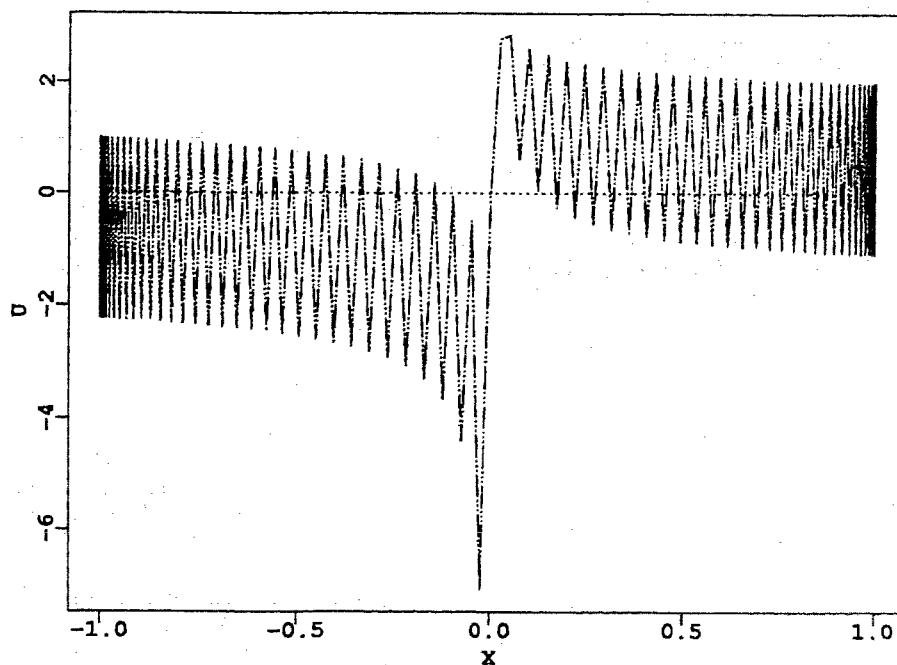
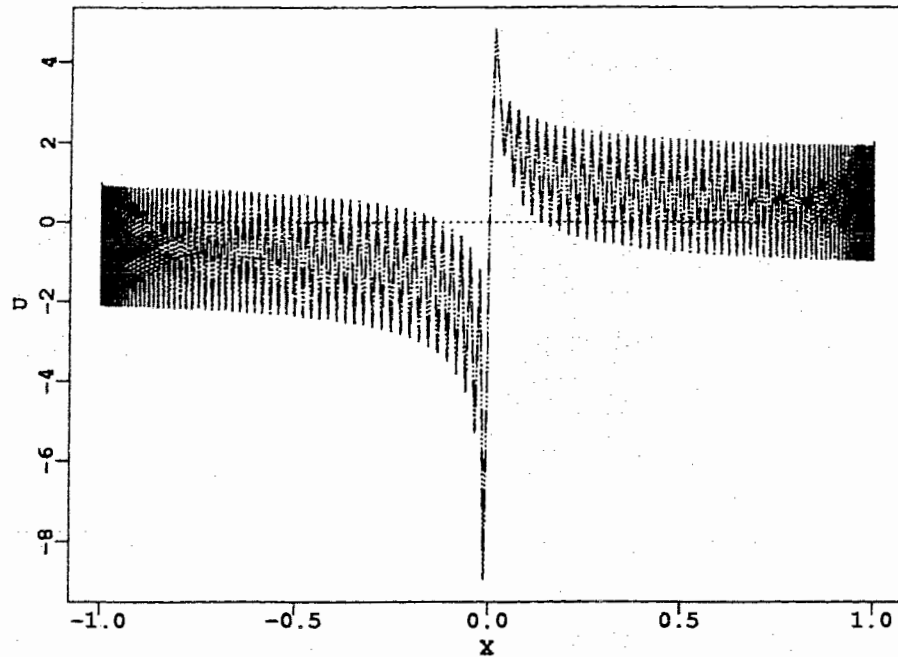


Figure 2.18: Example 2, pseudospectral Chebyshev method, $\epsilon = 10^{-6}$, $N = 128$

Figure 2.19: Example 2, pseudospectral Chebyshev method, $\epsilon = 10^{-6}$, $N = 256$

N	Condition Number	Average Error	Cpu-Time
32	3.0×10^3	2.10×10^{-3}	0.39
64	5.2×10^4	7.89×10^{-4}	1.89
96	2.7×10^5	7.85×10^{-4}	5.64
128	8.8×10^5	7.83×10^{-4}	12.77

Table 2.4: Example 2, pseudospectral Chebyshev method, $\epsilon = 10^{-2}$

N	Condition Number	Average Error	Cpu-Time
128	3.3×10^4	2.10×10^0	14.57
256	2.5×10^5	2.43×10^{-2}	102.58
512	3.9×10^6	1.30×10^{-6}	804.09

Table 2.5: Example 2, pseudospectral Chebyshev method, $\epsilon = 10^{-4}$

N	Condition Number	Average Error	Cpu-Time
128	7.1×10^3	1.62×10^0	12.71
256	2.3×10^4	2.43×10^0	93.08
512	1.1×10^5	1.32×10^0	730.58

Table 2.6: Example 2, pseudospectral Chebyshev method, $\epsilon = 10^{-6}$ **Example 3.**

As the third example, we considered a problem with one boundary layer

$$\begin{cases} 2\epsilon U''(x) + U'(x) = 0, & -1 \leq x \leq 1, \\ U(-1) = 0, & U(1) = 1 \end{cases} \quad (2.5.21)$$

which has the exact solution

$$U(x) = \left\{ 1 - \exp\left(-\frac{x+1}{2\epsilon}\right) \right\} / \left\{ 1 - \exp\left(-\frac{1}{\epsilon}\right) \right\}.$$

Similar to the previous two examples, when $\epsilon = 10^{-2}$, our numerical solutions became very accurate for N as small as 64; see Figure 2.19, Figure 2.20 and Table 2.7. When $\epsilon = 10^{-4}$, a large N was needed to produce a fairly good numerical solution; see Figure 2.21, Figure 2.22 and Table 2.8. When $\epsilon = 10^{-6}$, our numerical solutions were not close to the exact solution for N as large as 512; see Figure 2.23, Figure 2.24 and Table 2.9. For this problem, the condition numbers are much smaller than $O(N^4)$, because there is a small ϵ multiplying the matrix D_x^2 .

Finally, we plot some pictures of the error versus N for Example 1 and 2, as shown in Figure 2.25 to Figure 2.30. The errors become smaller as N is larger.

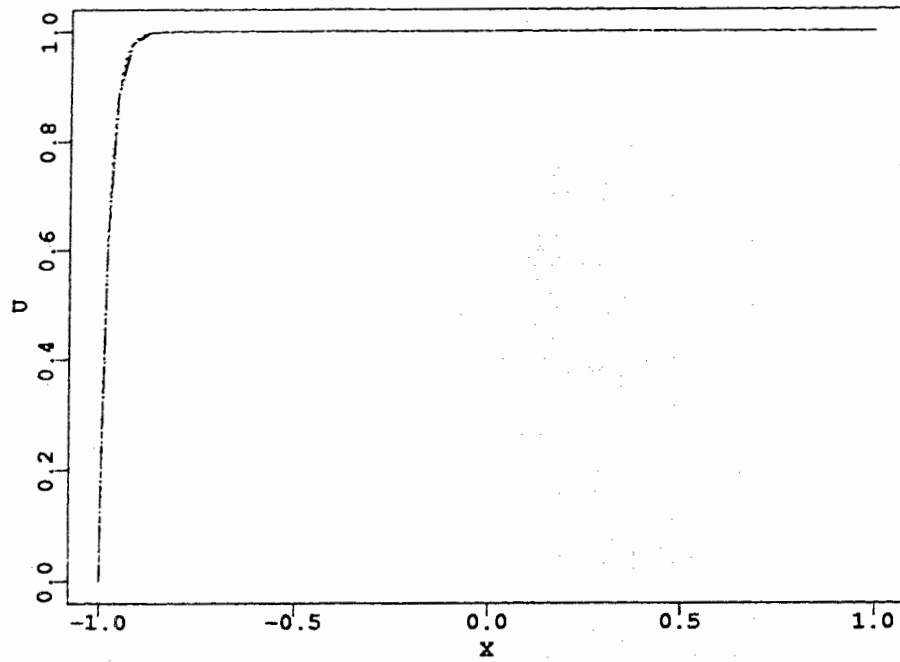


Figure 2.20: Example 3, pseudospectral Chebyshev method, $\epsilon = 10^{-2}$, $N = 32$

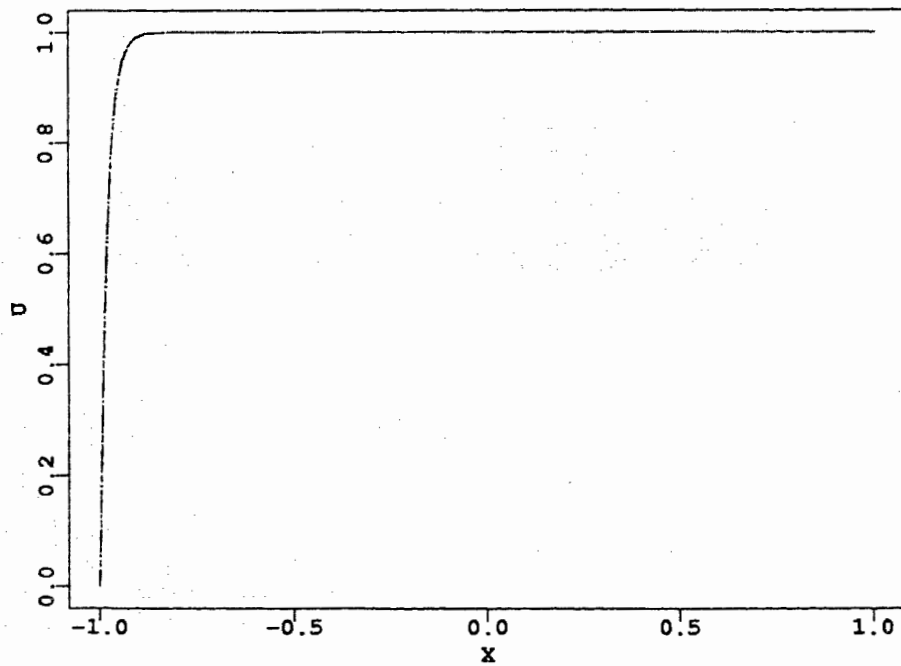


Figure 2.21: Example 3, pseudospectral Chebyshev method, $\epsilon = 10^{-2}$, $N = 64$

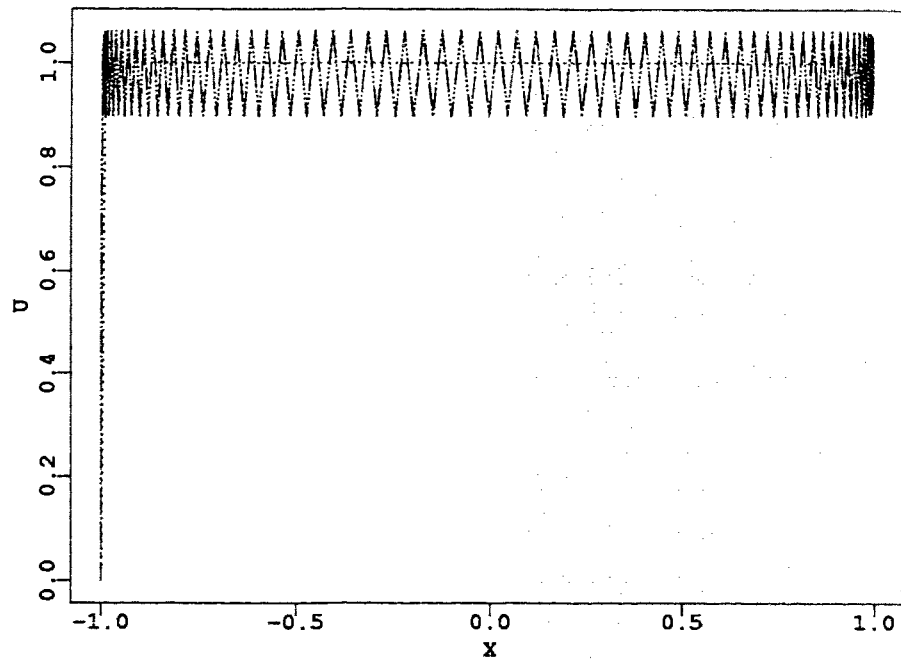


Figure 2.22: Example 3, pseudospectral Chebyshev method, $\epsilon = 10^{-4}$, $N = 128$

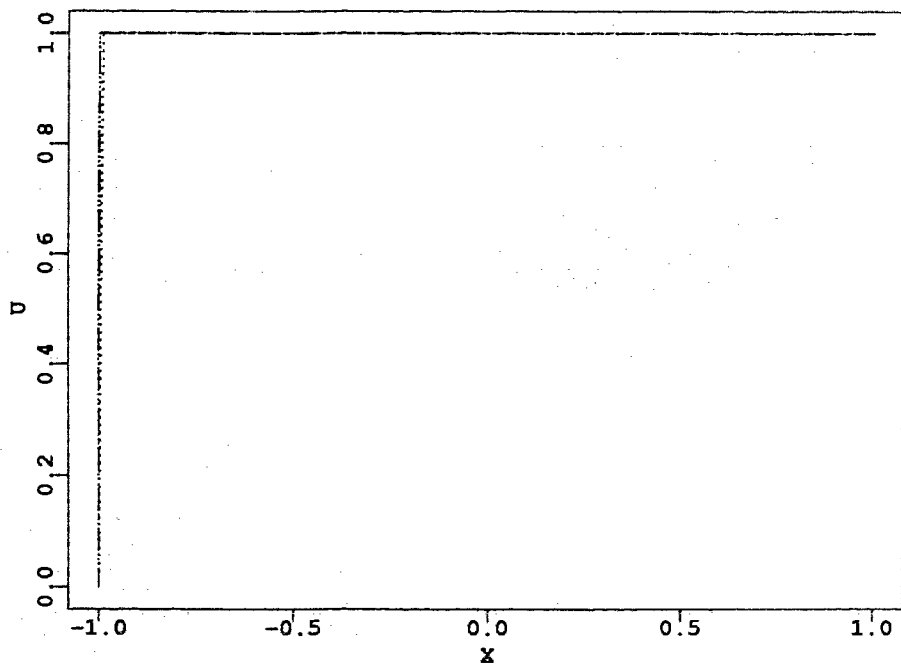


Figure 2.23: Example 3, pseudospectral Chebyshev method, $\epsilon = 10^{-4}$, $N = 256$

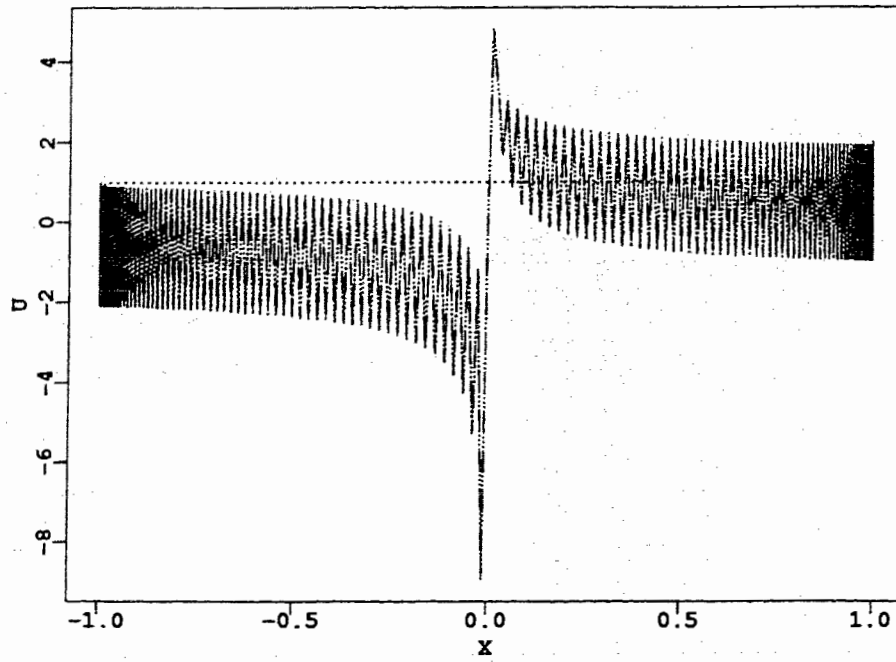


Figure 2.24: Example 3, pseudospectral Chebyshev method, $\epsilon = 10^{-6}$, $N = 256$

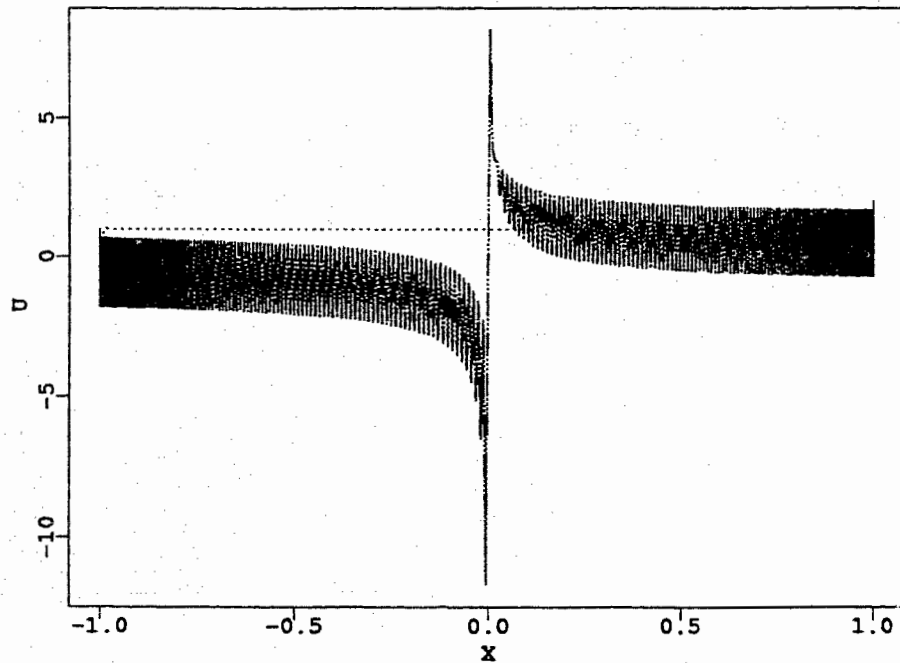


Figure 2.25: Example 3, pseudospectral Chebyshev method, $\epsilon = 10^{-6}$, $N = 512$

N	Condition Number	Average Error	Cpu-Time
32	1.7×10^3	4.52×10^{-6}	0.36
64	2.4×10^4	4.36×10^{-15}	1.90
96	1.2×10^5	3.25×10^{-14}	5.59
128	3.7×10^5	4.96×10^{-14}	12.69

Table 2.7: Example 3, pseudospectral Chebyshev method, $\epsilon = 10^{-2}$

N	Condition Number	Average Error	Cpu-Time
32	1.4×10^3	0.25×10^1	0.39
64	2.0×10^3	0.59×10^0	1.86
96	3.7×10^3	0.22×10^0	5.58
128	7.5×10^3	0.81×10^{-1}	12.47
256	7.0×10^4	0.30×10^{-3}	93.52

Table 2.8: Example 3, pseudospectral Chebyshev method, $\epsilon = 10^{-4}$

N	Condition Number	Average Error	Cpu-Time
256	2.3×10^4	1.54×10^0	94.07
512	1.3×10^5	1.32×10^0	749.28

Table 2.9: Example 3, pseudospectral Chebyshev method, $\epsilon = 10^{-6}$

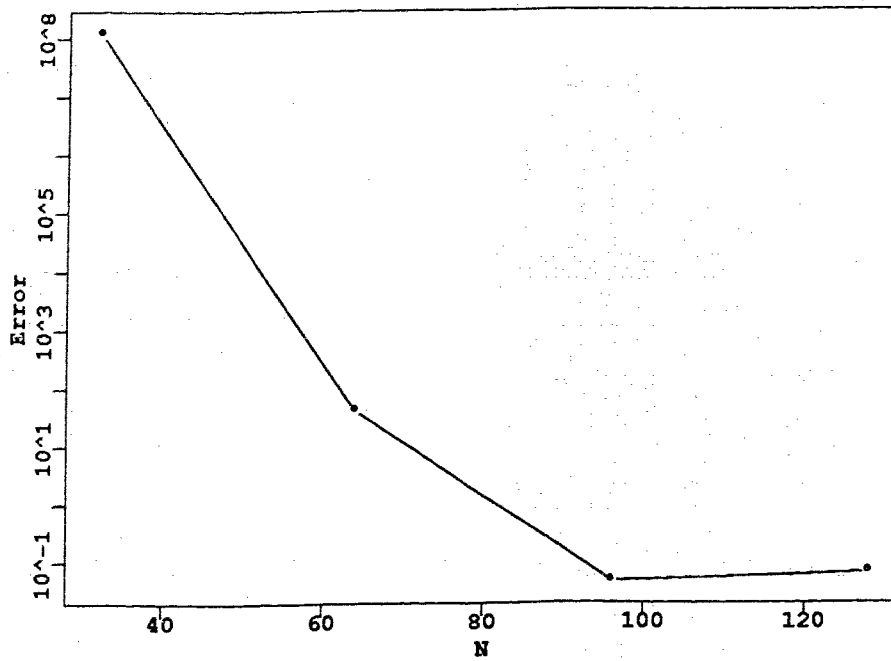


Figure 2.26: Example 1, average error ($\times 10^{12}$) for $\epsilon = 10^{-2}$, $N = 32, 64, 96, 128$

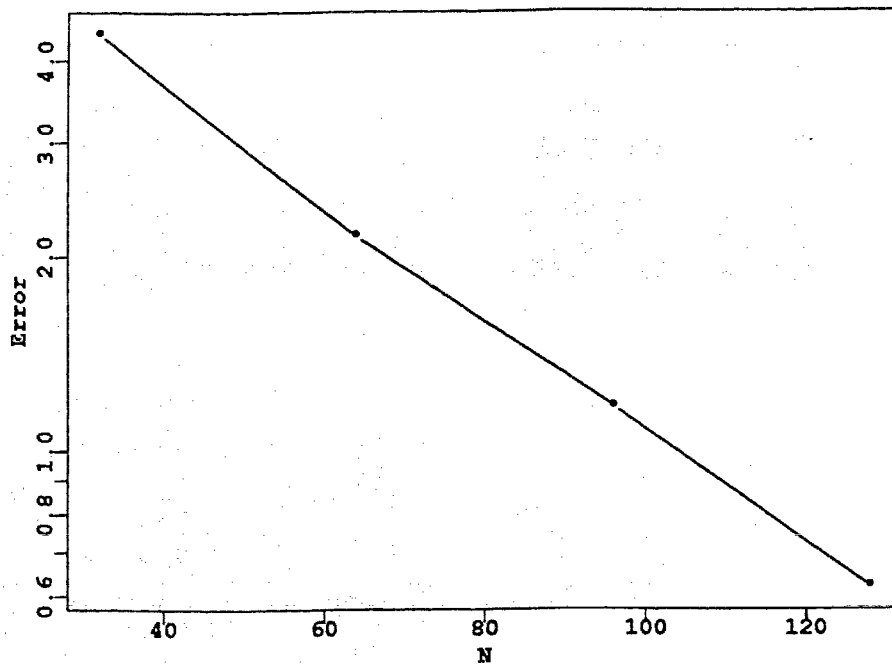


Figure 2.27: Example 1, average error ($\times 10^2$) for $\epsilon = 10^{-4}$, $N = 32, 64, 96, 128$

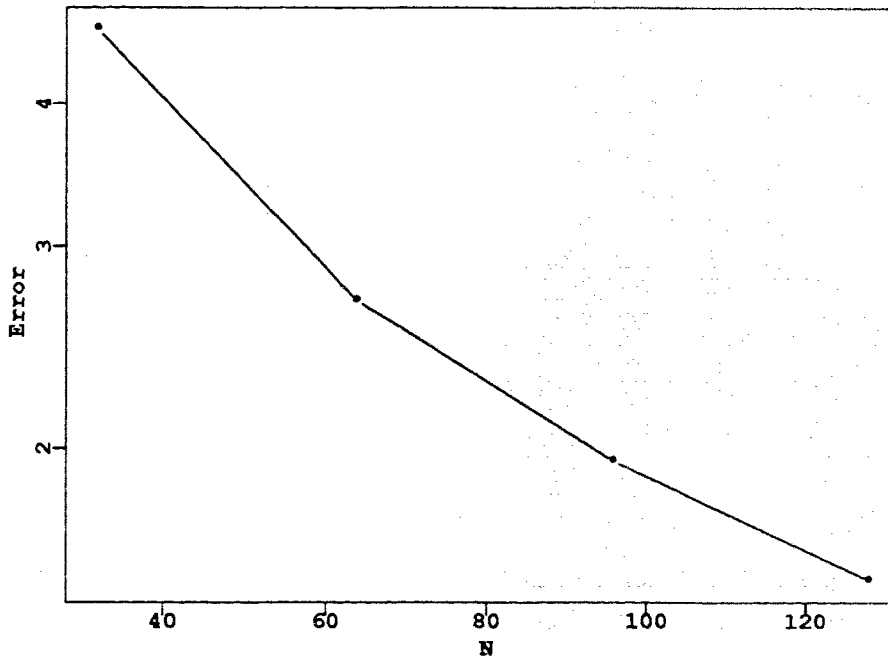


Figure 2.28: Example 1, average error ($\times 10^2$) for $\epsilon = 10^{-6}$, $N = 32, 64, 96, 128$

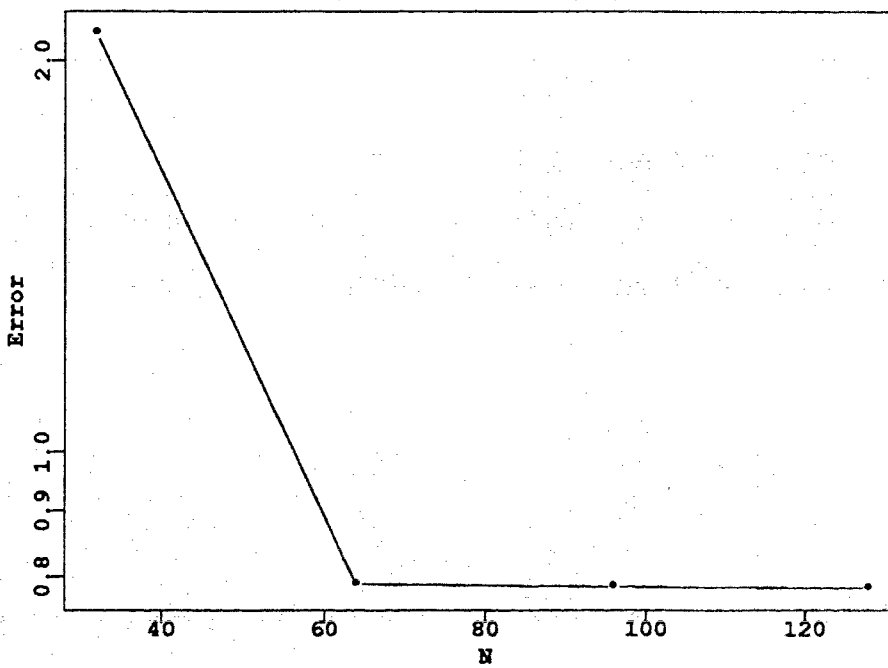


Figure 2.29: Example 2, average error ($\times 10^3$) for $\epsilon = 10^{-2}$, $N = 32, 64, 96, 128$

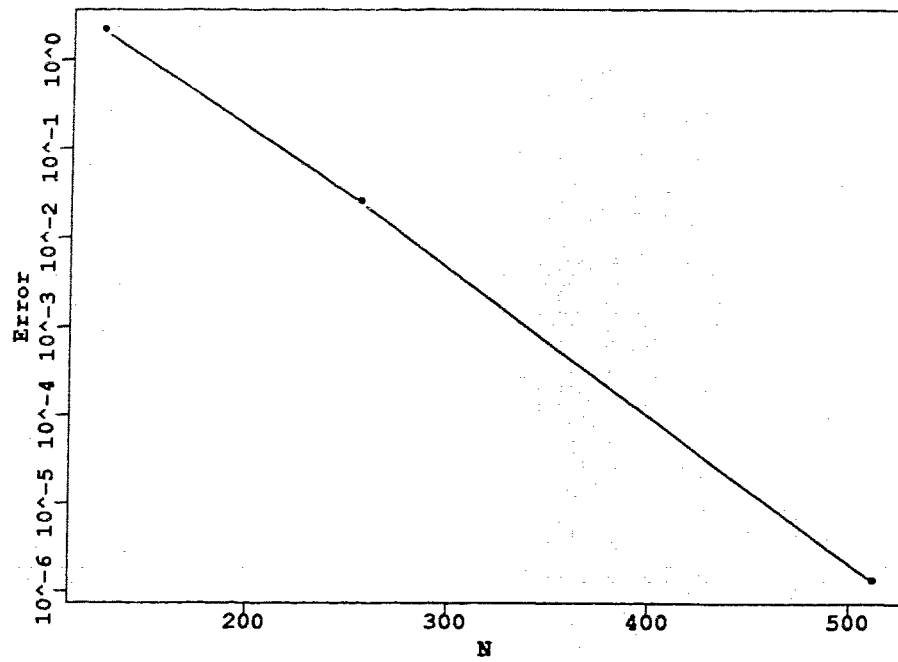


Figure 2.30: Example 2, average error for $\epsilon = 10^{-4}$, $N = 128, 256, 512$

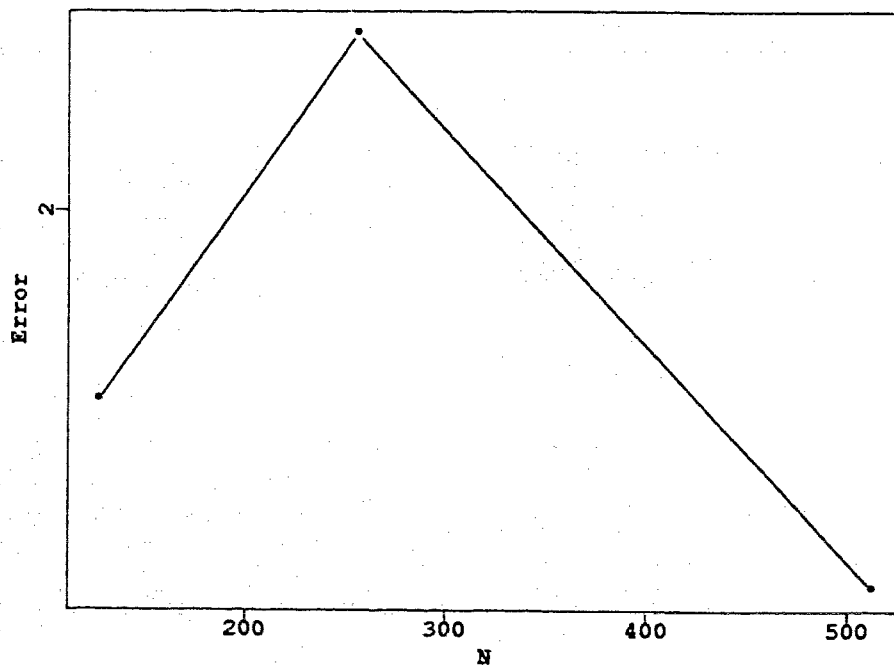


Figure 2.31: Example 2, average error for $\epsilon = 10^{-6}$, $N = 128, 256, 512$

Chapter 3

Preliminary Transform To The Circle

3.1 Introduction

In our study of the pseudospectral Chebyshev method for linear ODEs of order 2 in Chapter 2, we must solve a system of linear equations whose condition number grows with the number of collocation points N like $O(N^4)$. This property arises from the differentiation of the Chebyshev interpolating polynomials. When a relatively large N is needed to produce a fairly good numerical solution, the related condition number is large, indicating relatively low precision in the numerical solution. Therefore, various efforts have been made to reduce the condition number whenever possible.

In this Chapter, we will first discuss a variant of the pseudospectral Chebyshev method studied in Chapter 2, which was suggested by Berrut [4]. The idea is simple: first, the problem is replaced by an equivalent one on the circle using a transformation; secondly, the transformed problem on the circle is solved with the pseudospectral Fourier method. It is hoped that the related matrices resulted from doing this will have smaller condition numbers like $O(N^2)$; see [4]. Following this, we will discuss two iterative methods briefly.

3.2 The Method of Transformation to the Circle

For simplicity, we consider a first order problem. Consider the initial value problem

$$\begin{cases} U'(x) = F(x), \\ U(1) = 0. \end{cases} \quad (3.2.1)$$

If the pseudospectral Chebyshev method is used to find the Chebyshev interpolation polynomials

$$\bar{U}(x) = \sum_{j=0}^N \bar{U}(x_j) L_j(x),$$

and if the Chebyshev points

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

are used as collocation points, one will reach a system of linear equations

$$A\bar{U} = b, \quad (3.2.2)$$

where A is D_x defined in section 2.3, $b = (F(x_1), \dots, F(x_N))^T$, $\bar{U} = (\bar{U}(x_1), \dots, \bar{U}(x_N))^T$.

The direct solution to (2.2.2) requires $O(N^3)$ operations. The condition number of the matrix A grows with N like $O(N^2)$ (see [2]).

We now define the process of transformation to the circle with ϕ denoting the angle; we have

$$x = \cos \phi,$$

$$u(\phi) := U(\cos \phi),$$

$$f(\phi) := F(\cos \phi).$$

Thus $u(\phi)$ and $f(\phi)$ are 2π -periodic even functions on the full unit circle by defining $f(2\pi - \phi) = f(\phi)$.

With these definitions, problem (3.2.1) now becomes

$$\begin{cases} -u'(\phi)/\sin(\phi) = f(\phi), \\ u(0) = 0. \end{cases} \quad (3.2.3)$$

or, multiplying both sides by $\sin \phi$,

$$\begin{cases} u'(\phi) = -f(\phi) \sin \phi =: h(\phi), \\ u(0) = 0. \end{cases} \quad (3.2.4)$$

It is clear from the definitions that all functions appearing in (3.2.4) are 2π -periodic and inherit the differentiability properties of the corresponding functions in the original problem. More importantly, problem (3.2.4) can be solved by the pseudospectral Fourier method as described below.

Interpolating the trigonometric polynomials of degree N between the equidistant points

$$\phi_j = j \frac{2\pi}{2N}, \quad j = 0, 1, \dots, 2N - 1,$$

and collocating at the ϕ_j yield the linear system

$$\sum_{n=0}^{2N-1} (D_\phi)_{jn} \bar{u}(\phi_n) = h(\phi_j) \quad (3.2.5)$$

for the approximate values $\bar{u}(\phi_0), \dots, \bar{u}(\phi_{2N-1})$. The coefficients $(D_\phi)_{jn}$ are the values at ϕ_j of the derivative of the coefficient [4]

$$l_n(\phi) = \frac{1}{2N} \sin[N(\phi - \phi_n)] \cot \frac{\phi - \phi_n}{2}.$$

This result comes from the general Lagrangian form

$$\sum_{n=0}^{2N-1} f(\phi_n) l_n(\phi) \quad (3.2.6)$$

of the trigonometric polynomial interpolating an arbitrary function $f(\phi)$ between the ϕ_j .

From the definition of the derivative and Bernoulli rule, one can easily compute [4]

$$(D_\phi)_{jn} = l'_n(\phi_j) = \begin{cases} \frac{1}{2}(-1)^{j+n} \cot \frac{\phi_j - \phi_n}{2} = \frac{1}{2}(-1)^{j-n} \cot \frac{(j-n)\pi}{2N}, & j \neq n, \\ 0, & j = n. \end{cases}$$

The matrix D_ϕ of the system (3.2.5) is therefore circulant (and skew symmetric). The ϕ_j are the images of the Chebyshev points under the transformation $\phi = \arccos(x)$; therefore,

the values of $F(\phi)$ used in the computation of $h(\phi_j)$ are the same as those used in $A\bar{U} = b$ of (3.2.2).

The transformation to the circle technique can be applied to higher order problems. We now consider the (singularly perturbed) second order linear two-point boundary value problem

$$\begin{cases} \epsilon U''(x) + P(x)U'(x) + Q(x)U(x) = F(x), \\ U(1) = u_0, \quad U(-1) = u_N. \end{cases} \quad (3.2.7)$$

Now, instead of solving (2.3.15) directly, we propose to apply the circle transformation to obtain an equivalent problem which is defined on the circle. To do this, let $\phi(x) = \arccos(x)$ be the inverse of $x = \cos \phi$ and define $u(\phi) := U(\cos \phi)$, $p(\phi) := P(\cos \phi)$, $q(\phi) := Q(\cos \phi)$, and $f(\phi) := F(\cos \phi)$; then

$$\begin{aligned} \phi'(x) &= -(\sin \phi(x))^{-1}, \\ \phi''(x) &= [\sin \phi(x)]^{-2} \cdot \cos \phi(x) \cdot \phi'(x) = -\frac{\cos \phi(x)}{\sin^3 \phi(x)}, \\ \frac{d}{dx}U(x) &= \frac{d}{d\phi}U(\cos \phi)\phi'(x) = u'(\phi)\phi'(x), \\ \frac{d^2}{dx^2}U(x) &= \frac{d}{dx}[u'(\phi)\phi'(x)], \\ &= u''(\phi)\phi'(x)^2 + u'(\phi)\phi''(x). \end{aligned}$$

Substituting the above expressions into (3.2.7) leads to

$$\epsilon \left[u''(\phi) \frac{1}{\sin^2 \phi} + u'(\phi) \left(-\frac{\cos \phi}{\sin^3 \phi} \right) \right] + p(\phi) \left(-\frac{1}{\sin \phi} \right) u'(\phi) + q(\phi)u(\phi) = f(\phi).$$

Multiplying both sides by $\sin^3 \phi$, we get

$$\epsilon u''(\phi) \sin(\phi) - \epsilon u'(\phi) \cos(\phi) - p(\phi) \sin^2(\phi)u'(\phi) + q(\phi) \sin^3(\phi)u(\phi) = f(\phi) \sin^3(\phi). \quad (3.2.8)$$

The boundary conditions in (3.2.6) are transformed into $u(\phi_0) = u_0$ and $u(\phi_N) = u_N$.

The pseudospectral Fourier method applied to the above equation then leads to the linear system ($j = 0, 1, \dots, 2N - 1$)

$$\sum_{n=0}^{2N-1} a_{jn} u(\phi_n) = b_j, \quad (3.2.9)$$

with

$$\begin{aligned} a_{jn} &:= \epsilon \sin(\phi_j)(D_\phi^2)_{jn} - [\epsilon \cos(\phi_j) + p(\phi_j) \sin^2(\phi_j)](D_\phi)_{jn} + q(\phi_j) \sin^3(\phi_j) \delta_{jn} \\ b_j &:= f(\phi_j) \sin^3(\phi_j). \end{aligned}$$

Similar to the pseudospectral Chebyshev method, the matrix of the second derivatives $l''_n(\phi_j)$ is the square of the matrix D_ϕ of the first derivatives. And there even exists a simple formula [4]:

$$(D_\phi^2)_{jn} = \begin{cases} \frac{1}{2}(-1)^{j+n+1} \csc^2 \frac{\phi_j - \phi_n}{2} = \frac{1}{2}(-1)^{j-n+1} \csc^2 \frac{(j-n)\pi}{2N}, & j \neq n, \\ -\frac{2N^2+1}{6}, & j = n. \end{cases}$$

Since $p(\phi)$, $q(\phi)$, $f(\phi)$ and $\bar{u}(\phi)$ are even functions of ϕ , and moreover, since D_ϕ is antisymmetric and D_ϕ^2 is symmetric, only $N - 1$ of the $2N$ equations (3.2.10) are linearly independent: equations number $j = 0$ and N reduce to 0 ($\sin \phi_j = 0$ and $(D_\phi)_{jn} \bar{u}(\phi_n) = 0$), and for $n = 1, \dots, N - 1$, equation number $2N - n$ is the negative of equation number n . If we substitute $\bar{u}(\phi_{2N-n}) = \bar{u}(\phi_n)$ into (3.2.9), we only need to solve the system composed of the equations numbered 1 to $N - 1$ of (3.2.9), namely

$$\sum_{n=1}^{N-1} (a_{jn} + a_{j,2N-n}) u(\phi_n) = b_j - a_{j0} u_0 - a_{jN} u_N, \quad (3.2.10)$$

by Gaussian elimination. Our computations later show that the condition number of (3.2.10) grows like $O(N^2)$ for non-stiff problems (3.2.7)[4]. But for stiff problems, this is not true. The principle is the transformation of the original problem to the circle and the consequent elimination of the endpoint singularities of the factor of the derivatives ($\csc(\phi)$ or an odd power of it). The Fourier method is then applied to the regularized equation; it makes use of the same values of the functions appearing in the differential problem as the classical Chebyshev method.

3.3 Iterative Methods

It is well-known that algebraic systems resulting from spectral methods are full and rather ill-conditioned. For problem (3.2.6), the condition number of the matrix D_x^2 grows like $O(N^4)$. Considerable attention has been devoted to the use of iterative methods. In this section, we briefly discuss two iterative methods [16].

3.3.1 CGNR (The Conjugate Gradient Iteration Applied to the Normal Equations)

CGNR is the name for implementation of the conjugate gradient iteration method applied to the normal equations [16]. We apply the conjugate gradient iteration to the nonsymmetric system

$$A\bar{U} = b.$$

The normal equations are

$$A^T A U = A^T b.$$

This algorithm constructs a sequence

$$r_0, A^T r_0, (A^T A) A^T r_0, \dots, (A^T A)^{n-1} A^T r_0.$$

We want to find the unique sequence of vectors

$$u_n \in \text{span}\{A^T r_0, (A^T A) A^T r_0, \dots, (A^T A)^{n-1} A^T r_0\},$$

with minimal residual at each step:

$$\|r_n\| = \text{minimum}$$

where $r_n = b - Au_n$. This is equivalent to the orthogonality condition:

$$r_n \perp \text{span}\{AA^T r_0, (AA^T)^2 r_0, \dots, (AA^T)^n r_0\}.$$

Now, u_n can be formed by a three-term recurrence relation in the conjugate gradient iteration.

The algorithm of CGNR:

1. Given an initial guess u_0 and initial residual

$$r_0 = b - Au_0,$$

2. Given $\beta_0 := 0; p_0 := 0,$

3. Do $n := 1, 2, \dots,$

$$p_n := A^T r_{n-1} + \beta_{n-1} p_{n-1},$$

$$\alpha_n := \|A^T r_{n-1}\|_2^2 / \|Ap_n\|_2^2,$$

$$u_n := u_{n-1} + \alpha_n p_n,$$

$$r_n := r_{n-1} - \alpha_n Ap_n,$$

$$\beta_n := \|A^T r_n\|_2^2 / \|A^T r_{n-1}\|_2^2.$$

The convergence of CGNR is determined solely by the singular values of A . Any two matrices with the same singular values have identical worst-case convergence rates. If A is normal, the moduli of the eigenvalues are equal to the singular values. For details see [16].

3.3.2 CGS (The Conjugate Gradient Squared)

CGS is a biorthogonalization algorithm adapted from the biconjugate gradient iteration [16].

First, we introduce biconjugate gradient iteration (BCG) which constructs non-optimal approximations.

BCG constructs a sequence of vectors

$$u_n \in \text{span}\{r_0, Ar_0, \dots, A_{n-1}r_0\},$$

which implies

$$e_n = p_n(A)e_0, \quad r_n = p_n(A)r_0 \tag{3.3.11}$$

for some polynomial p_n of degree n , where $e_0 = A^{-1}b - u_0$, and $r_0 = b - Au_0$. The value p_n is now determined by the orthogonality condition

$$r_n \perp \text{span}\{\tilde{r}_0, A^T \tilde{r}_0, \dots, A_{n-1}^T \tilde{r}_0\}$$

where $\tilde{r}_0 \in R^{n-1}$ is a vector often taken equal to r_0 . BCG computes its choice of u_n by three-term recurrence relations.

CGS, which stands for “CG-squared”, is a modification of BCG by Sonneveld [20] who replaces equation (3.3.11) by

$$e_n = p_n^2(A)e_0, \quad r_n = p_n^2(A)r_0. \quad (3.3.12)$$

The polynomial p_n remains the same and there is no increase in the amount of work per step.

Furthermore, whereas BCG requires vector multiplications by both A and A^T , CGS only requires multiplications by A .

CGS algorithm:

1. Given an initial guess u_0 and initial residual

$$r_0 = b - Au_0,$$

2. Given $q_0 := p_0 := 0; \rho_0 := 1; \tilde{r}_0 = r_0$,

3. For $n = 1, 2, \dots$,

$$\rho_n := \tilde{r}_0^T r_{n-1},$$

$$\beta_n := \rho_n / \rho_{n-1},$$

$$w_n := r_{n-1} + \beta_n q_{n-1},$$

$$p_n := w_n + \beta_n (q_{n-1} + \beta_n p_{n-1}),$$

$$v_n := Ap_n,$$

$$\sigma_n := \tilde{r}_0^T v_n,$$

$$\begin{aligned}\alpha_n &:= \rho_n / \sigma_n, \\ q_n &:= w_n - \alpha_n v_n, \\ r_n &:= r_{n-1} - \alpha_n A(w_n + q_n), \\ u_n &:= u_{n-1} + \alpha_n (w_n + q_n).\end{aligned}$$

Although the convergence of CGS is governed by eigenvalues of A [16], its convergence properties are less well understood than CGNR. It is still unknown how frequently CGS turns out to be effective.

3.4 Numerical Results

We look again at the three examples studied in Chapter 2, using the method of transformation to the circle of section 3.2. We emphasize the comparison of the methods of this Chapter and the pseudospectral Chebyshev method of Chapter 2. The reader is reminded that in all the figures to follow, the exact solutions are plotted as dotted lines, and the numerical solutions are plotted as solid-dotted lines.

Example 1 Revisited.

For the three different values of ϵ (10^{-2} , 10^{-4} , and 10^{-6}), the method of transformation to the circle was applied to solve the BVP of Example 1. Figure 3.1 to Figure 3.11 and Table 3.1 to Table 3.3 show the results of the same kind of numerical work that was done in Example 1 of Chapter 2.

The pictures here and the pictures in Chapter 2 look similar, however, the condition numbers here are much smaller than those of Chapter 2 (except for the case where $N = 128$, $\epsilon = 10^{-4}$), and the average errors here tend to be smaller for $\epsilon = 10^{-2}$, and slightly larger for $\epsilon = 10^{-4}$ and $\epsilon = 10^{-6}$. The cpu times here tend to be larger than those of Chapter 2 (except for the case where $N = 128$, $\epsilon = 10^{-6}$). When we applied the iterative methods above, we found that CGNR converged reasonably fast, but CGS did not. Also CGNR appears to work better when applied to the original problems versus the transformation to

the circle. This came as a surprise, since the matrices associated with the circle technique are better conditioned. Figure 3.18 also shows that quite a few iterations of CGNR are required, making this rather expensive. Further study, especially on the use of efficient preconditioners is needed. We have chosen $N = 96$ and computed $(2 \times N)$ iterations of the CGNR method. The results for the pseudospectral method are shown in Figure 3.17 and Figure 3.18. The Figure 3.19 and Figure 3.20 are the results for the transformation to the circle technique.

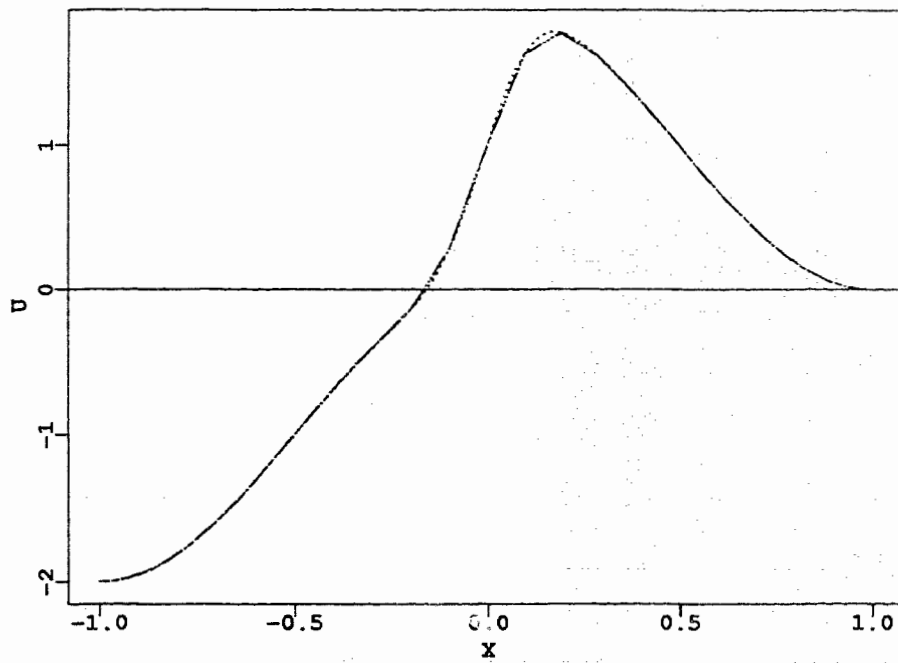


Figure 3.1: Example 1, transformation to the circle method, $\epsilon = 10^{-2}$, $N = 32$

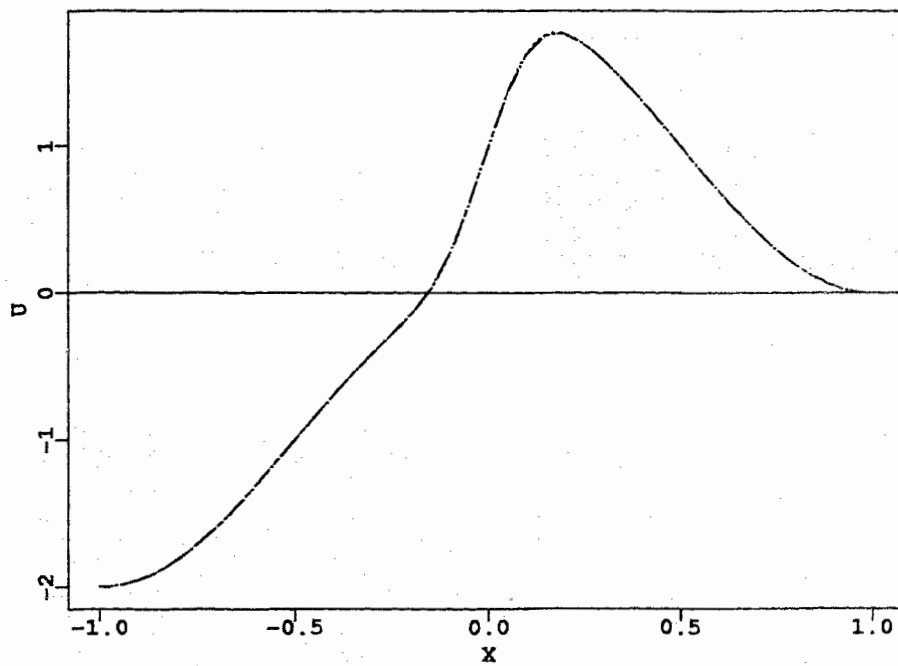


Figure 3.2: Example 1, transformation to the circle method, $\epsilon = 10^{-2}$, $N = 64$

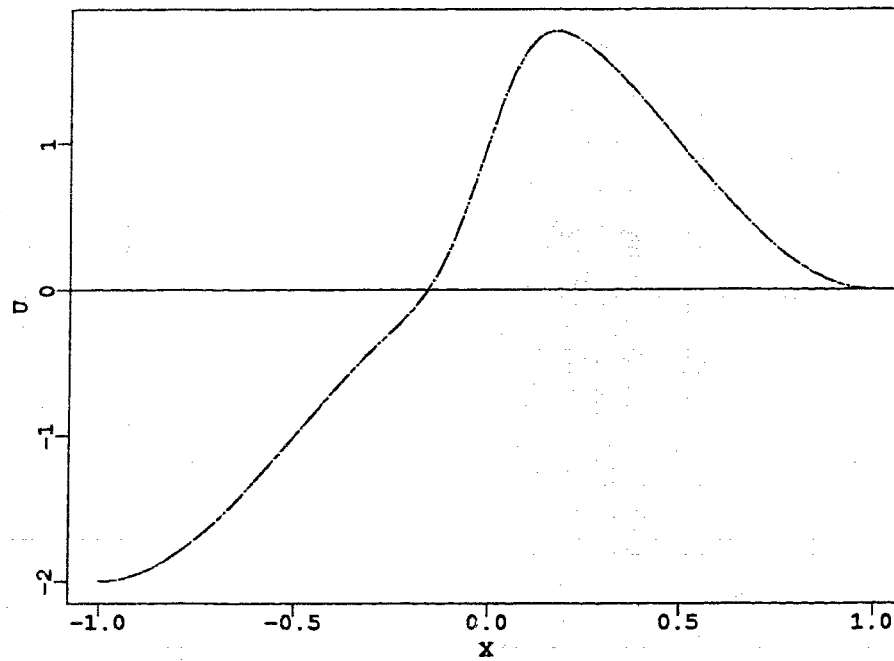


Figure 3.3: Example 1, transformation to the circle method, $\epsilon = 10^{-2}$, $N = 96$

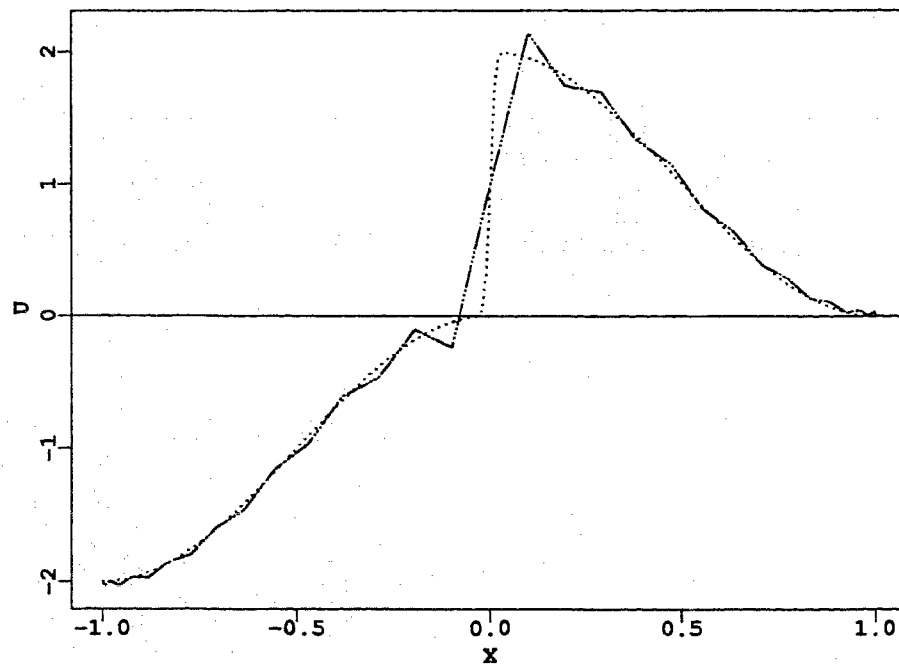


Figure 3.4: Example 1, transformation to the circle method, $\epsilon = 10^{-4}$, $N = 32$

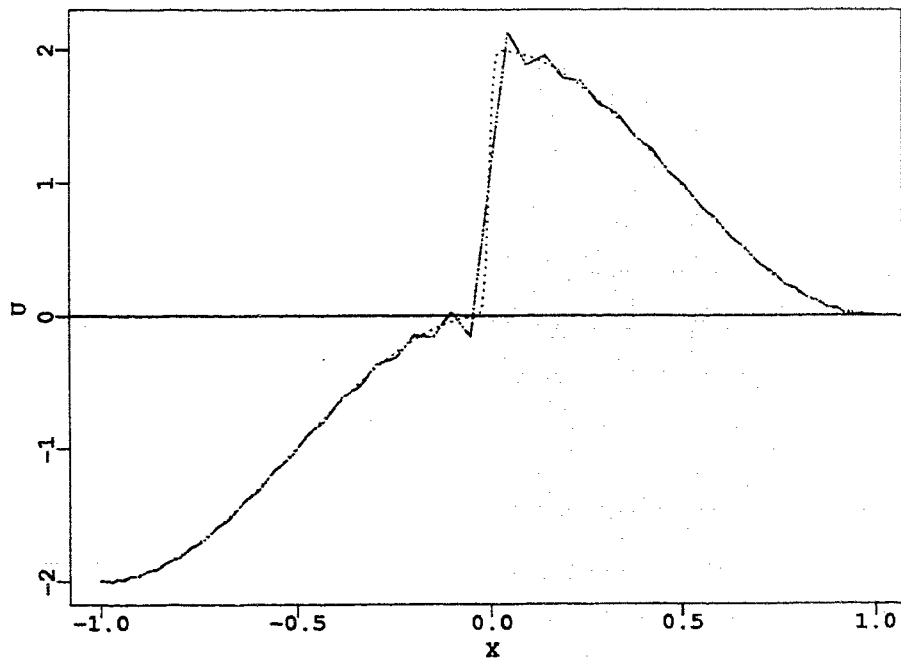


Figure 3.5: Example 1, transformation to the circle method, $\epsilon = 10^{-4}$, $N = 64$

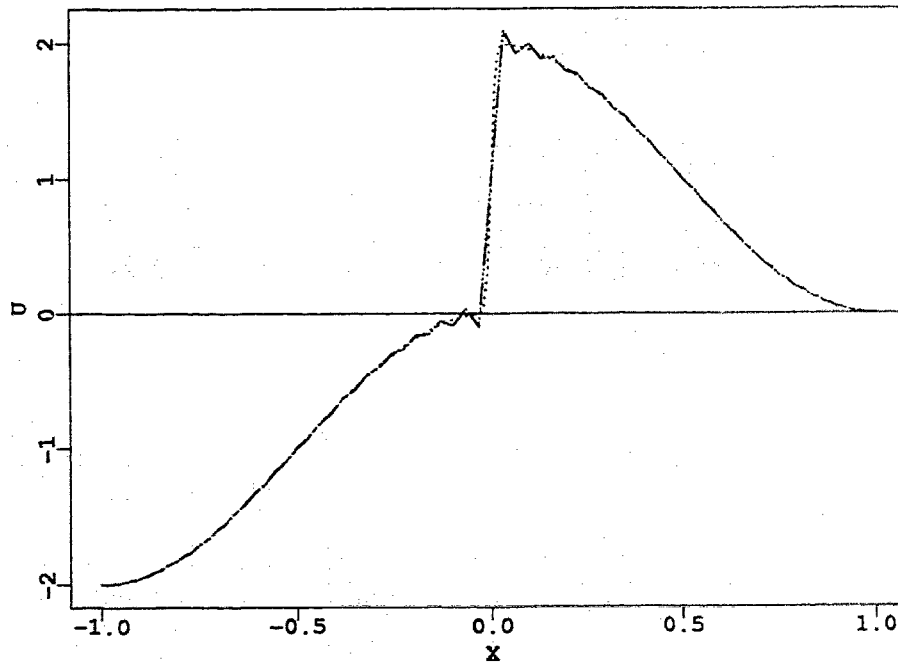


Figure 3.6: Example 1, transformation to the circle method, $\epsilon = 10^{-4}$, $N = 96$

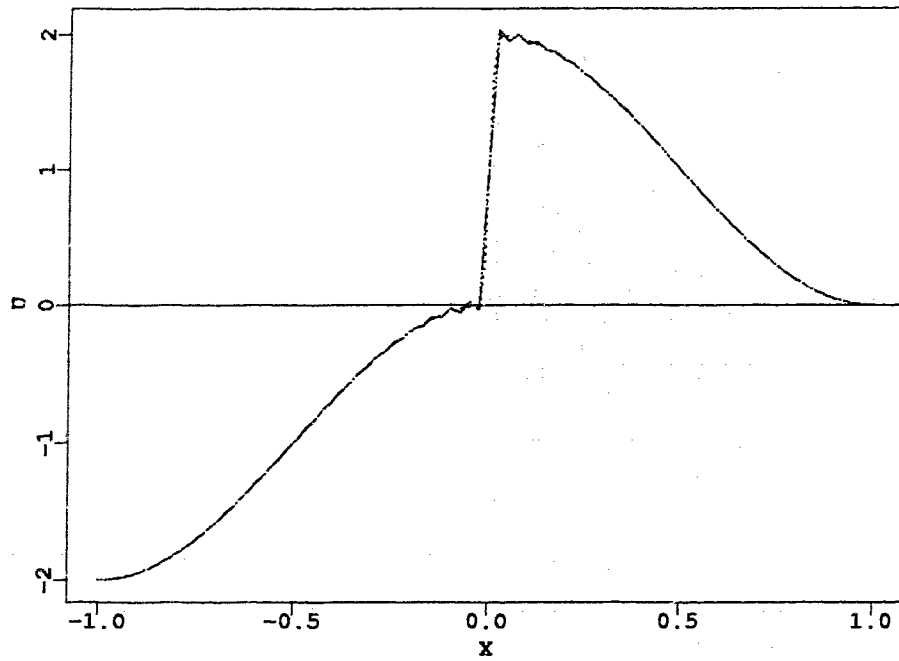


Figure 3.7: Example 1, transformation to the circle method, $\epsilon = 10^{-4}$, $N = 128$

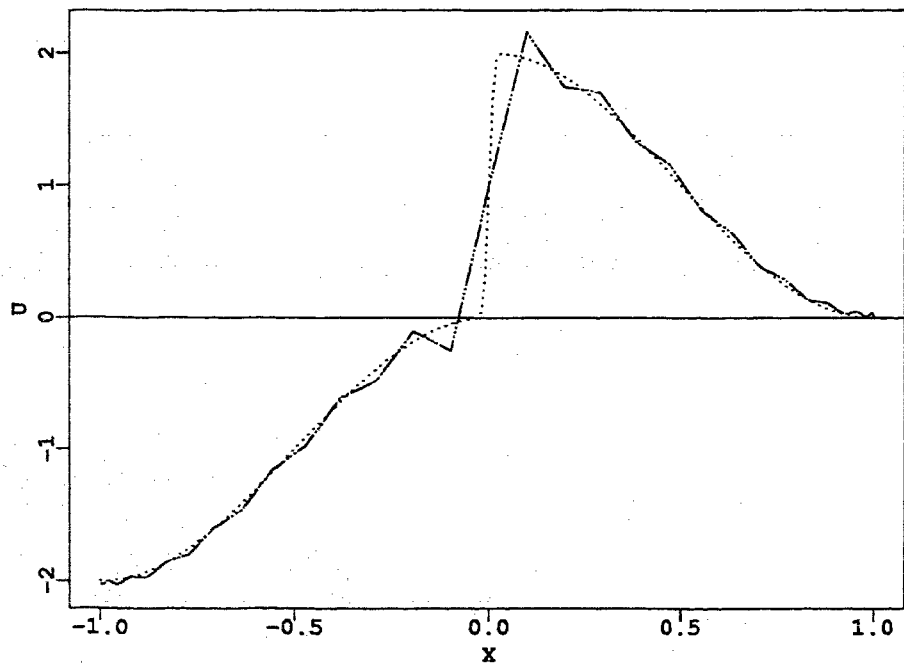


Figure 3.8: Example 1, transformation to the circle method, $\epsilon = 10^{-6}$, $N = 32$

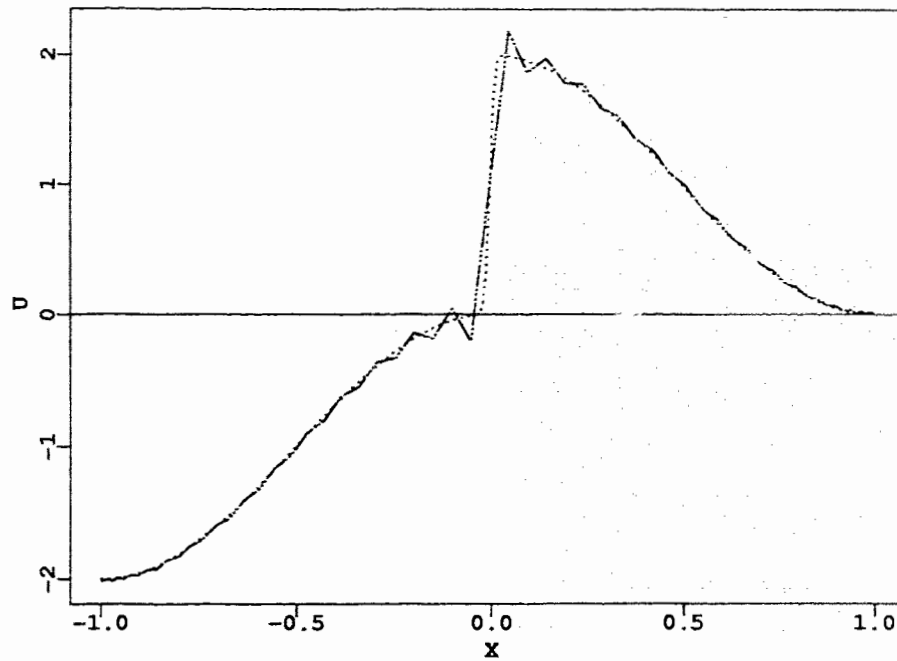


Figure 3.9: Example 1, transformation to the circle method, $\epsilon = 10^{-6}$, $N = 64$.

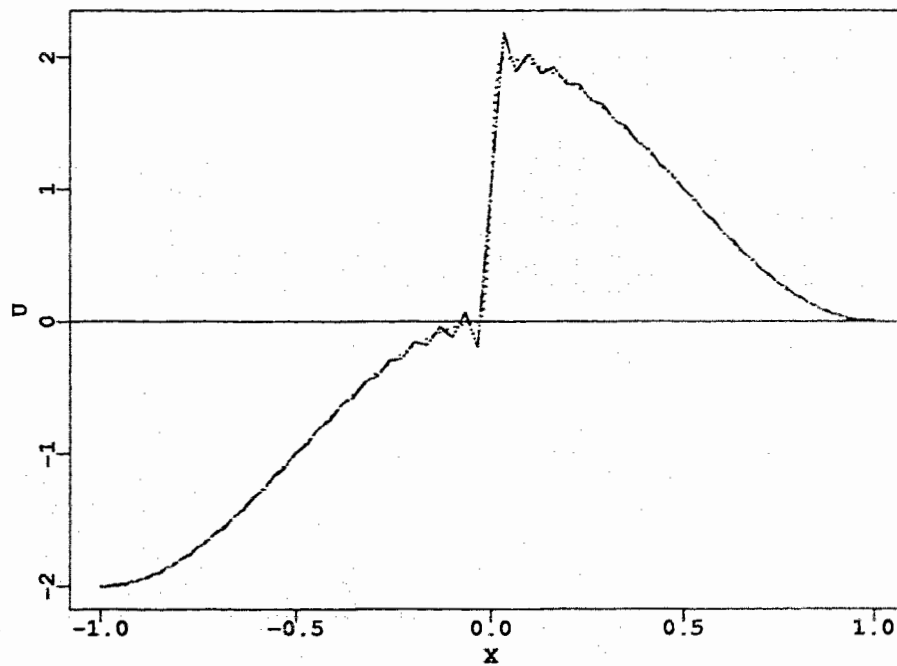
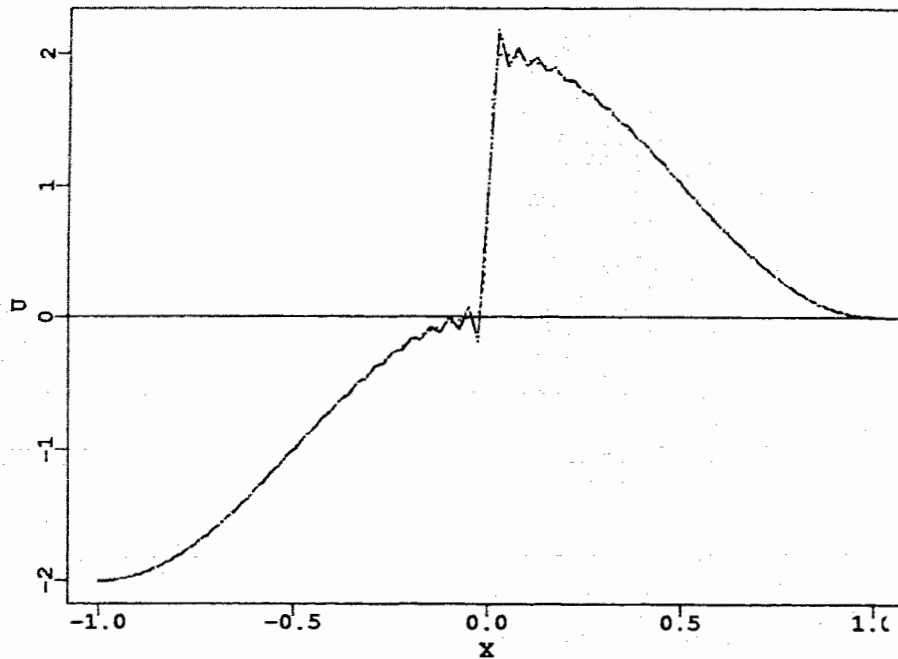


Figure 3.10: Example 1, transformation to the circle method, $\epsilon = 10^{-6}$, $N = 96$.

Figure 3.11: Example 1, transformation to the circle method, $\epsilon = 10^{-6}$, $N = 128$

N	Condition Number	Average Error	Cpu-Time
32	7.1×10^2	1.23×10^{-4}	0.97
64	2.3×10^3	4.18×10^{-11}	3.26
96	4.7×10^3	1.92×10^{-14}	7.17
128	7.9×10^3	2.10×10^{-14}	13.66

Table 3.1: Example 1, transformation to the circle method, $\epsilon = 10^{-2}$

N	Condition Number	Average Error	Cpu-Time
32	3.2×10^3	4.39×10^{-2}	0.92
64	2.2×10^4	2.15×10^{-2}	3.29
96	7.7×10^4	1.17×10^{-2}	7.17
128	1.7×10^5	6.20×10^{-3}	12.74

Table 3.2: Example 1, transformation to the circle method, $\epsilon = 10^{-4}$

N	Condition Number	Average Error	Cpu-Time
32	5.3×10^5	4.65×10^{-2}	0.91
64	6.3×10^5	2.97×10^{-2}	2.82
96	5.8×10^5	1.95×10^{-2}	6.06
128	5.3×10^5	1.54×10^{-2}	11.02

Table 3.3: Example 1, transformation to the circle method, $\epsilon = 10^{-6}$

Example 2 Revisited.

The same kind of numerical work was carried out as we did in Example 2 of Chapter 2. The results are displayed in Figure 3.15 to Figure 3.20 and Table 3.4 to Table 3.6.

The pictures here look close to the pictures in Chapter 2 again, but the condition numbers and the average errors show a crossed pattern. For $\epsilon = 10^{-2}$, the method of transformation to the circle produced much smaller condition numbers to reach the same average errors when compared with the pseudospectral Chebyshev method, however, for $\epsilon = 10^{-4}$ and $\epsilon = 10^{-6}$, this is no longer true. For $\epsilon = 10^{-4}$, the average errors are still the same for the two methods, but the condition numbers are larger for the first method when N is small, and then become smaller when N is large. For $\epsilon = 10^{-6}$, the condition numbers with the first method are always larger, while the average errors with the first method are always smaller. The cup times also show a crossed pattern.

We wonder why the condition numbers related with circle technique for smaller ϵ (10^{-4} and 10^{-6}) are much larger than the condition numbers obtained by pseudospectral Chebyshev method. It seems that the circle technique is only successful in solving non-singularly perturbed boundary value problems, but not so successful for solving singularly perturbed boundary value problems.

BVPs. We

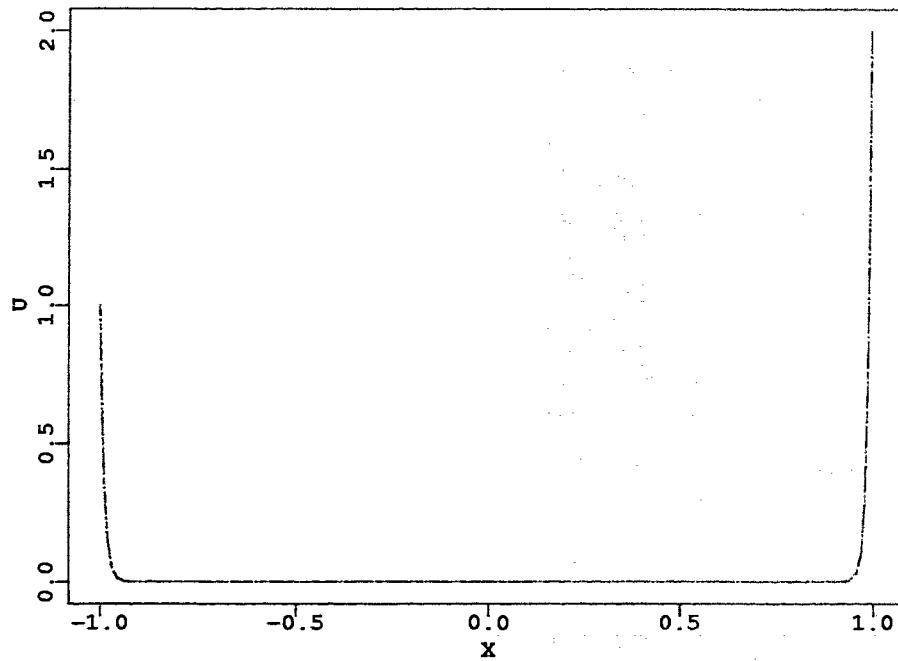


Figure 3.12: Example 2, transformation to the circle method, $\epsilon = 10^{-2}$, $N = 64$

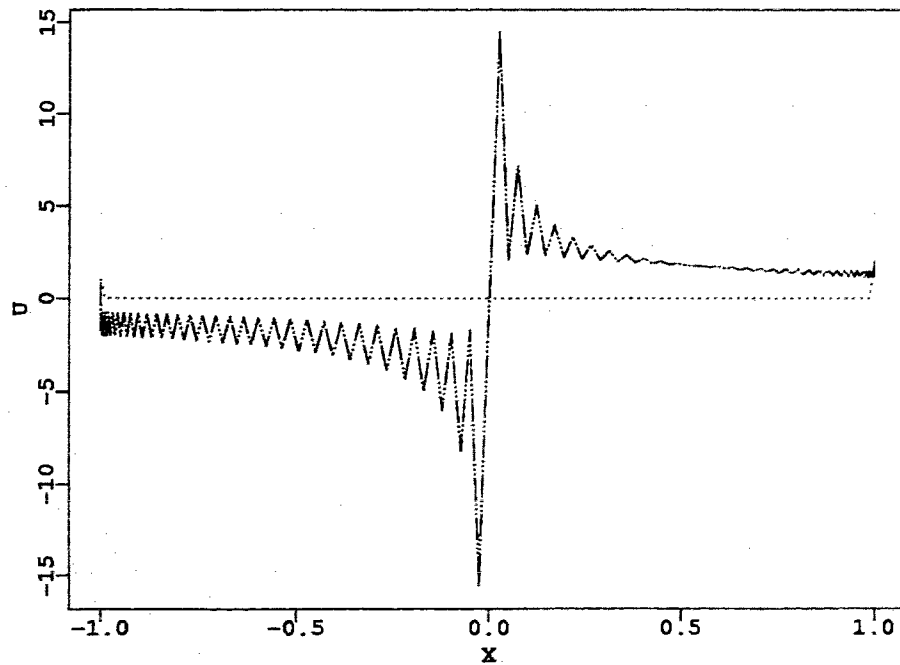


Figure 3.13: Example 2, transformation to the circle method, $\epsilon = 10^{-4}$, $N = 128$

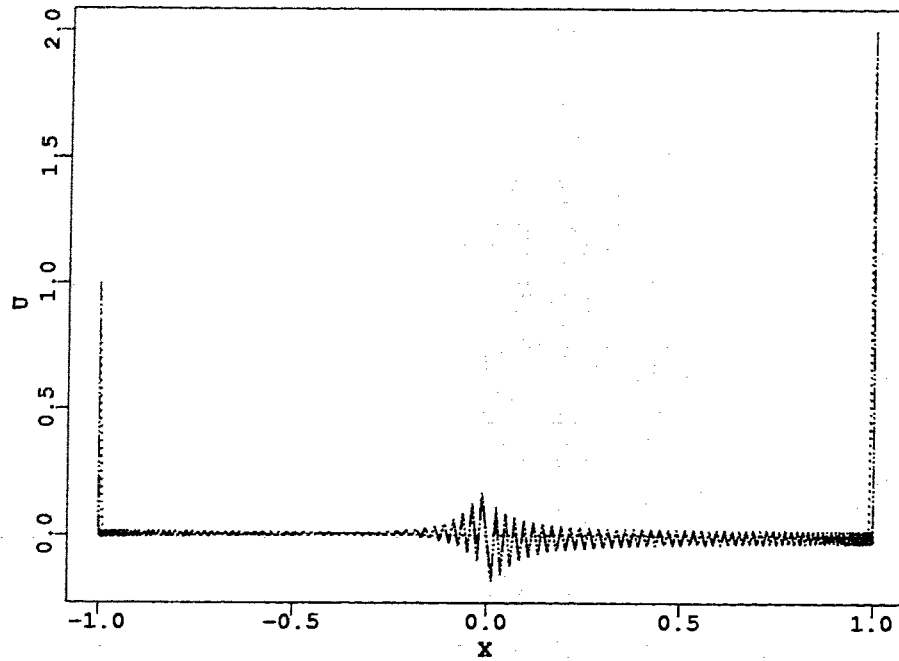


Figure 3.14: Example 2, transformation to the circle method, $\epsilon = 10^{-4}$, $N = 256$

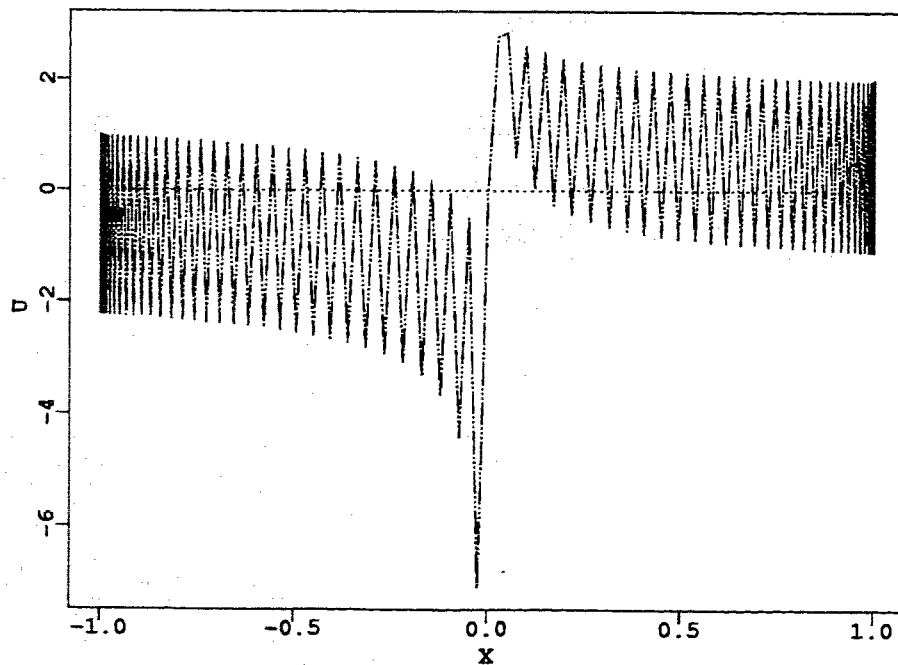


Figure 3.15: Example 2, transformation to the circle method, $\epsilon = 10^{-6}$, $N = 128$

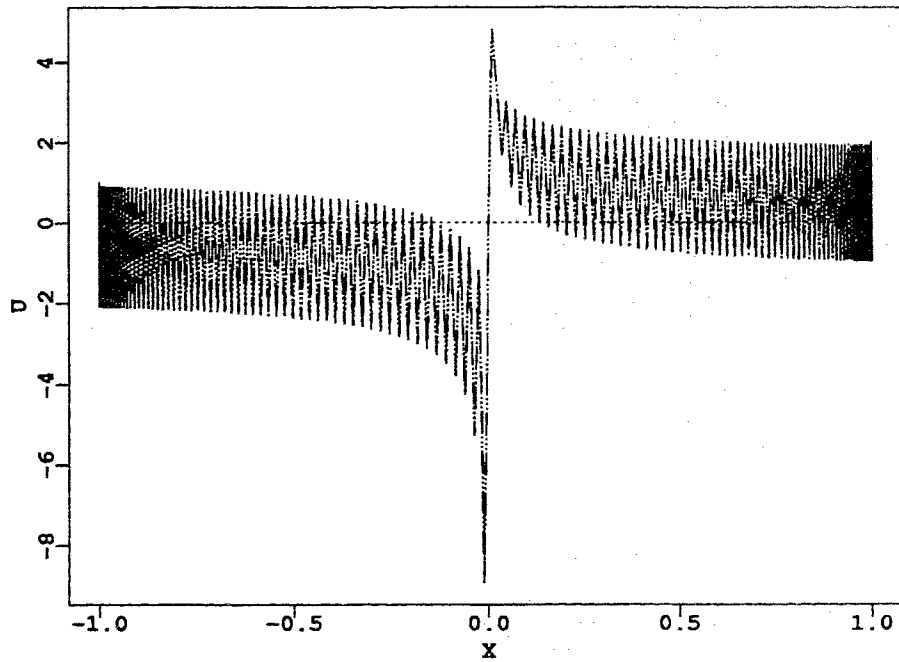


Figure 3.16: Example 2, transformation to the circle method, $\epsilon = 10^{-6}$, $N = 256$

N	Condition Number	Average Error	Cpu-Time
32	9.2×10^1	2.10×10^{-3}	0.91
64	3.3×10^2	7.89×10^{-4}	3.2
96	6.5×10^2	7.85×10^{-4}	7.18
128	1.1×10^3	7.83×10^{-4}	12.87

Table 3.4: Example 2, transformation to the circle method, $\epsilon = 10^{-2}$

N	Condition Number	Average Error	Cpu-Time
128	1.5×10^6	0.21×10^1	12.78
256	3.7×10^5	0.24×10^{-1}	49.12
512	1.2×10^5	0.13×10^{-5}	343.65

Table 3.5: Example 2, transformation to the circle method, $\epsilon = 10^{-4}$

N	Condition Number	Average Error	Cpu-Time
128	4.0×10^5	1.62×10^{-1}	12.91
256	3.9×10^6	1.54×10^{-1}	54.63

Table 3.6: Example 2, transformation to the circle method, $\epsilon = 10^{-6}$

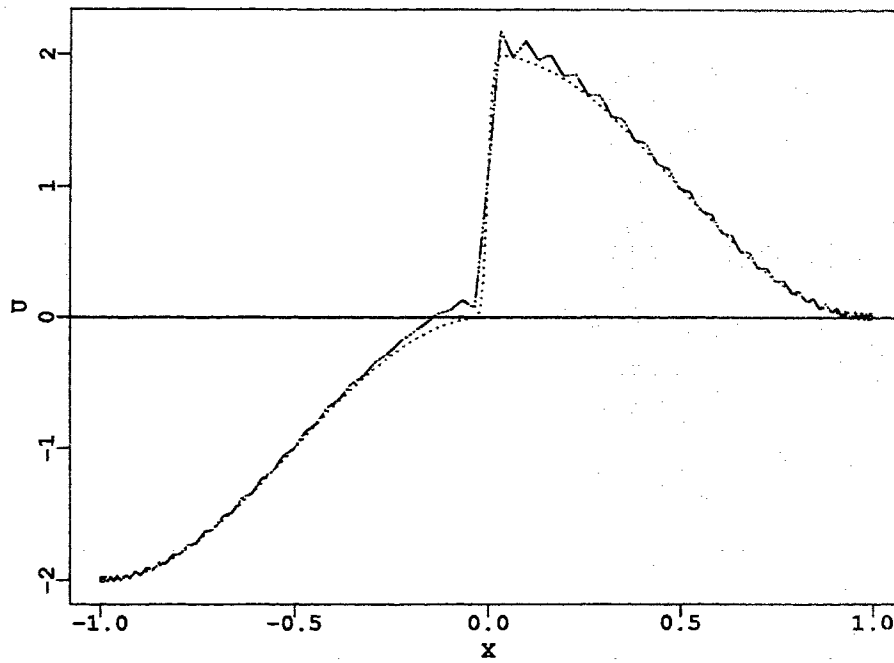


Figure 3.17: Example 1, the pseudospectral method, $N = 96$, $\epsilon = 10^{-4}$

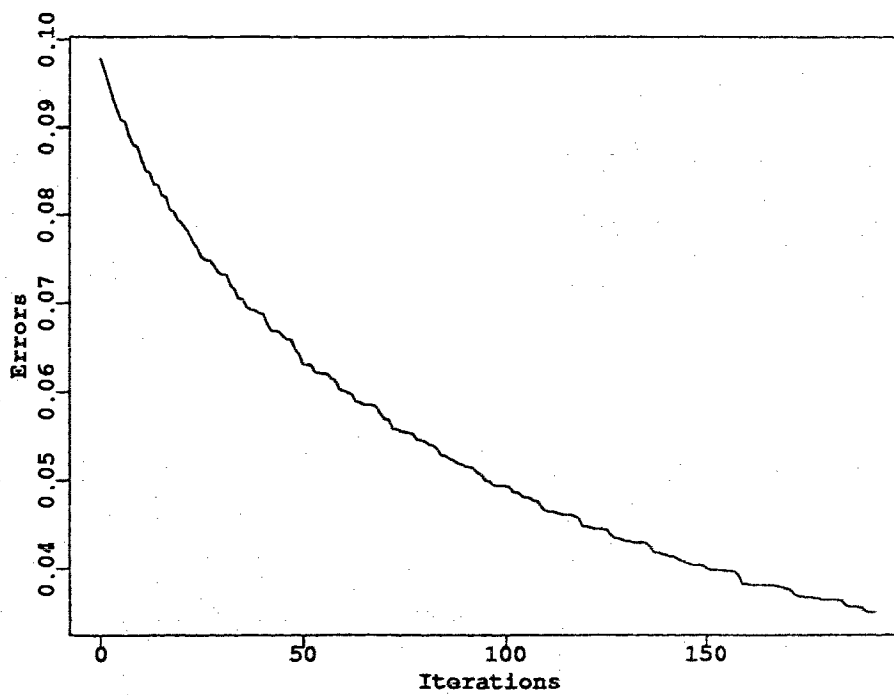


Figure 3.18: Example 1, the pseudospectral method, $N = 96$, $\epsilon = 10^{-4}$

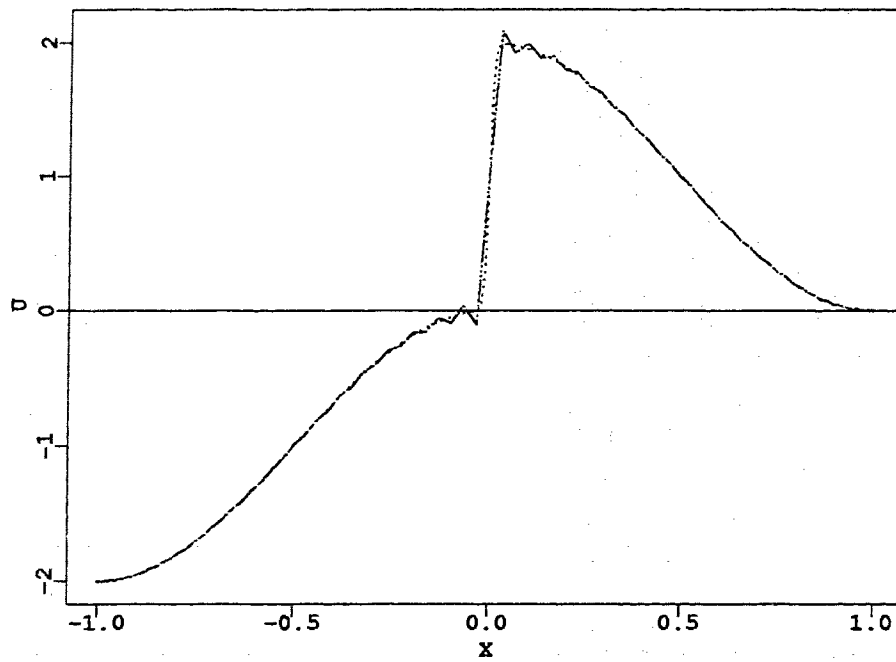


Figure 3.19: Example 1, the circle technique, $N = 96$, $\epsilon = 10^{-4}$

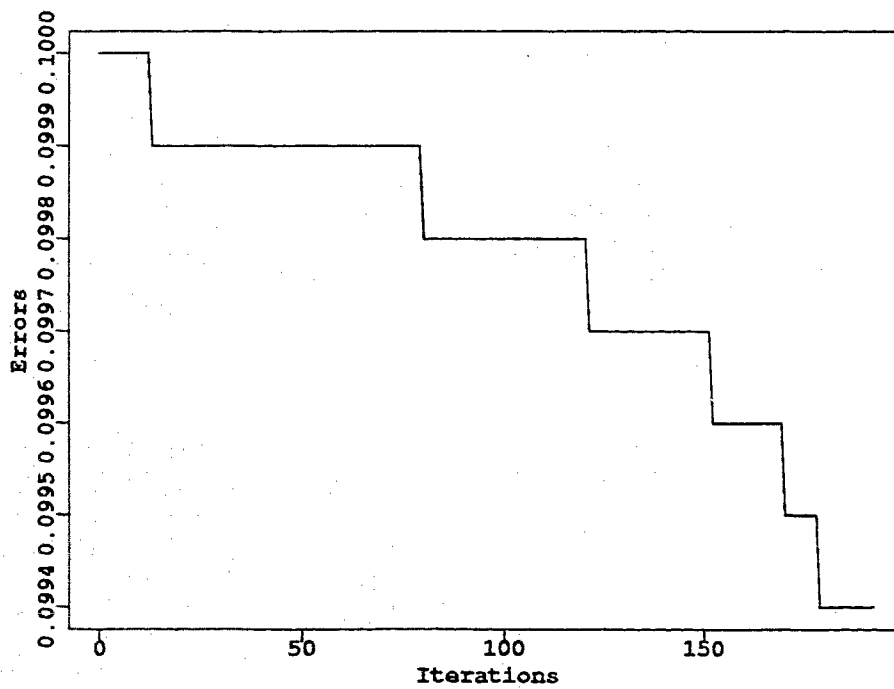


Figure 3.20: Example 1, the circle technique, $N = 96$, $\epsilon = 10^{-4}$

Chapter 4

Chebyshev Collocation Method

4.1 Chebyshev Collocation Method

In this Chapter, we will consider the collocation method with Chebyshev polynomials to solve two point boundary value problems. This method is discussed by Boyd [5] and it is a “nontraditional” spectral method, in the sense that we formulate the problem in terms of the (Chebyshev) expansion coefficients, but still use collocation. Usually these coefficients are determined by a Galerkin approach. We will compare the collocation method with the pseudospectral Chebyshev method discussed in Chapter 2. The pseudospectral Chebyshev method has a feature that is very attractive for certain types of problems, but leads to difficulties with others. This feature is very high resolution near the boundaries. For example, the collocation points for Chebyshev polynomial pseudospectral methods for problems on $-1 \leq x \leq 1$ are usually chosen such that $x_j = \cos(\pi j/N)$ ($j = 0, 1, \dots, N$). The collocation points x_1 and x_{N-1} are within approximately $\pi^2/2N^2$ of the boundary points x_0 and x_N , respectively, so that the boundary resolution is $\Delta x = O(1/N^2)$.

This leads to extremely good resolution properties of spectral methods for boundary-layer problems. While resolution of a problem with a boundary layer of thickness $\epsilon \ll 1$ would require $O(1/\epsilon)$ uniformly spaced grid points, it requires only $O(1/\epsilon^{1/2})$ terms in the

Chebyshev spectral series. Nonuniform grids could be used in many of these problems, as they can be implemented fairly efficiently in spectral methods using coordinate transformation. We will consider linear two point boundary value problems of the form:

$$\begin{cases} \epsilon u''(x) + p(x)u'(x) + q(x)u(x) = f(x) \\ u(-1) = u_N, \quad u(1) = u_0. \end{cases} \quad (4.1.1)$$

Using the Chebyshev collocation method, we seek a solution

$$u(x) = \sum_{k=0}^N a_k T_k(x) \quad (4.1.2)$$

where $\{T_k\}$ is the Chebyshev polynomial of degree k on $[-1, 1]$ defined by formula

$$T_k(\cos \theta) = \cos(k\theta).$$

Clearly,

$$|T_k(x)| \leq 1 \quad \text{for } x \in [-1, 1],$$

$$T_0(x) = 1, \quad T_1(x) = x,$$

and, using elementary trigonometric identities

$$T_{k+1}(x) = 2xT_k(x) - T_{k-1}(x) \quad \text{for } k \geq 1.$$

We introduce $N + 1$ suitable collocation points which are the extreme of the Chebyshev polynomial of order N . i.e., $x_j = \cos(\pi j/N)$ for $j = 0, 1, \dots, N$. Then, the approximate solution (4.1.2) must satisfy the ODE (4.1.1) and its boundary conditions at the collocation points $\{x_j\}$. Moreover, the following three steps are done:

(1) Determination of $N + 1$ coefficients $a_k (k = 0, 1, \dots, N)$, so that

$$u(x_j) = \sum_{k=0}^N a_k T_k(x_j) \quad (j = 0, 1, \dots, N). \quad (4.1.3)$$

and; (2) Evaluation of $u'(x_j)$ by

$$u'(x_j) = \sum_{k=0}^N a_k T'_k(x_j) \quad (j = 0, 1, \dots, N). \quad (4.1.4)$$

Finally, (3) we evaluate $u''(x_j)$ by

$$u''(x_j) = \sum_{k=0}^N a_k T_k''(x_j) \quad (j = 0, 1, \dots, N). \quad (4.1.5)$$

Here, the collocation equations are

$$\begin{cases} \epsilon u''(x_j) + p(x)u'(x_j) + q(x)u(x_j) = f(x_j) & (j = 1, \dots, N-1) \\ u(-1) = u_N, \quad u(1) = u_0 \end{cases} \quad (4.1.6)$$

and substituting (4.1.3) (4.1.4) and (4.1.5) into the collocation equations, we have a linear system for determining coefficients $\{a_k\}_{k=0}^N$, i.e.,

$$\begin{cases} \sum_{k=0}^N a_k T_k(1) = u_0 \\ \sum_{k=0}^N [\epsilon T_k''(x_j) + p(x_j)T_k'(x_j) + q(x_j)T_k(x_j)] a_k = f(x_j) & (j = 1, \dots, N-1) \\ \sum_{k=1}^N a_k T_k(-1) = u_N. \end{cases}$$

We write it in matrix form

$$Ax = b \quad (4.1.7)$$

where $A \in R^{(N+1) \times (N+1)}$, $x, b \in R^{N+1}$,

$$x = (a_0, a_1, \dots, a_N)^T, \quad b = (u_0, f(x_1), \dots, f(x_{N-1}), u_N)^T,$$

$$A = (a_{ij}),$$

with

$$a_{1k} = T_k(1) \quad (k = 0, \dots, N),$$

$$a_{jk} = \epsilon T_k''(x_j) + p(x_j)T_k'(x_j) + q(x_j)T_k(x_j) \quad (k = 0, 1, \dots, N, \text{ and } j = 1, \dots, N-1),$$

$$a_{Nk} = T_k(-1) \quad (k = 0, 1, \dots, N).$$

Chapter 2 indicates that the matrix A is a very ill-condition matrix. Its condition number is $O(N^4)$. Some researchers consider a preconditioning scheme to solve this system(4.1.7) [6]. However, we use Linpack subroutines (Gaussian elimination) to solve (4.1.7).

$$Ax = b. \quad (4.1.8)$$

4.2 Numerical Results

The method described in section 4.1 is applied to Example 2. Table 4.1 to Table 4.3 contain the summaries of the related results.

N	Condition Number	Average Error	Cpu-Time
32	1.6×10^3	2.30×10^{-2}	0.73
64	1.7×10^4	6.31×10^{-3}	2.12
96	6.2×10^4	5.16×10^{-3}	4.63
128	1.8×10^5	4.95×10^{-3}	8.43
256	1.9×10^6	4.76×10^{-3}	40.42
512	2.5×10^7	4.76×10^{-3}	206.77

Table 4.1: Example 2, Chebyshev collocation method, $\epsilon = 10^{-2}$.

N	Condition Number	Average Error	Cpu-Time
96	6.1×10^4	1.27×10^0	4.59
128	2.2×10^5	2.02×10^0	8.38
256	6.0×10^5	8.31×10^{-2}	38.98
512	2.9×10^6	6.43×10^{-3}	207.17

Table 4.2: Example 2, Chebyshev collocation method, $\epsilon = 10^{-4}$.

N	Condition Number	Average Error	Cpu-Time
96	5.0×10^4	1.28×10^0	4.59
128	1.0×10^5	1.29×10^0	8.38
256	6.0×10^5	1.21×10^0	39.92
512	2.9×10^6	1.14×10^0	208.74

Table 4.3: Example 2, Chebyshev collocation method, $\epsilon = 10^{-6}$.

For the large parameter ($\epsilon = 10^{-2}$), the numerical results are gradually better with increasing N . See Figure 4.1 to Figure 4.6 for $N = 32, 64, 96, 128, 256, 512$ respectively.

For stiff problem, $\epsilon = 10^{-4}$, we first try $N = 96$ and $N = 128$, respectively. Figure 4.7 and Figure 4.8 show that there are large oscillations. A large N is needed to produce a fairly good numerical solution, see Figure 4.9 and Figure 4.10 and Table 4.2.

For the very stiff problem (viz., $\epsilon = 10^{-6}$), the method becomes difficult to solve it with the large average errors. See Figure 4.10, Figure 4.11, Figure 4.12 and Table 4.3.

We compare the present method with pseudospectral Chebyshev method we studied in Chapter 2. The Chebyshev collocation method for solving the non-stiff problem ($\epsilon = 10^{-2}$) is less efficient than the pseudospectral Chebyshev method. For the stiff problem $\epsilon = 10^{-4}$, the Chebyshev collocation method can compete with the pseudospectral Chebyshev method. For the very stiff problem ($\epsilon = 10^{-6}$), the Chebyshev collocation method is not quite as bad as the pseudospectral Chebyshev method. It is interesting the oscillation that occurred in the Chebyshev collocation method is still reasonable, while the oscillation that occurred in the pseudospectral method is unacceptable in some sense. However, our basic conclusion is that the pseudospectral is better than the collocation approach. Also, the condition numbers resulting from the collocation approach appear to be worse.

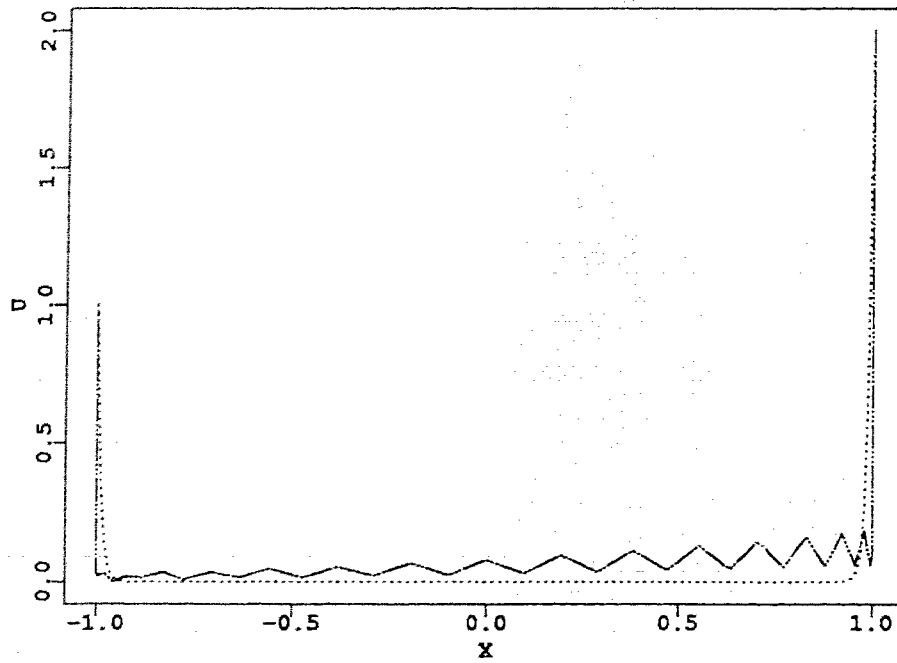


Figure 4.1: Example 2, Chebyshev collocation method, $\epsilon = 10^{-2}$, $N = 32$

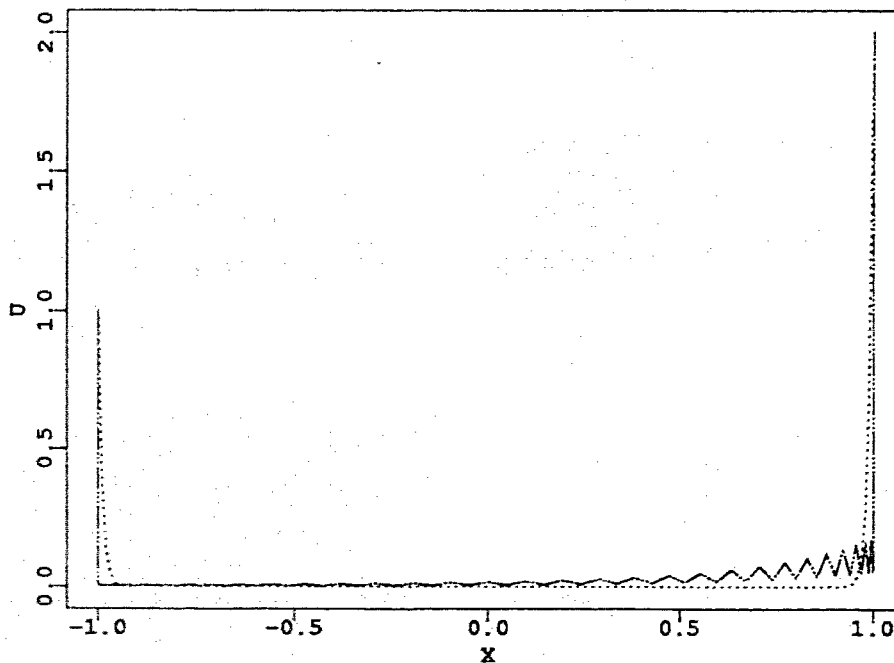


Figure 4.2: Example 2, Chebyshev collocation method, $\epsilon = 10^{-2}$, $N = 64$

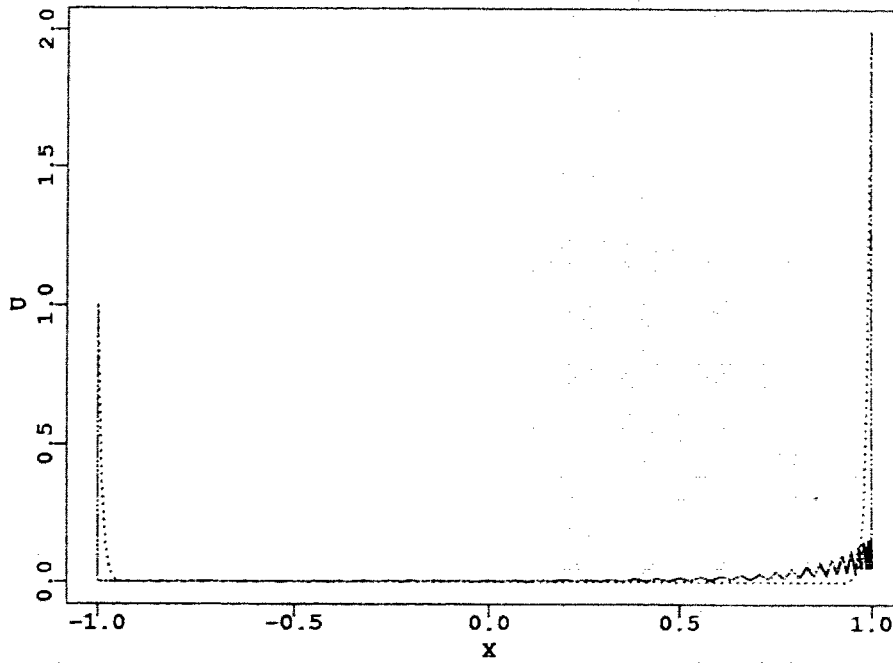


Figure 4.3: Example 2, Chebyshev collocation method, $\epsilon = 10^{-2}$, $N = 96$

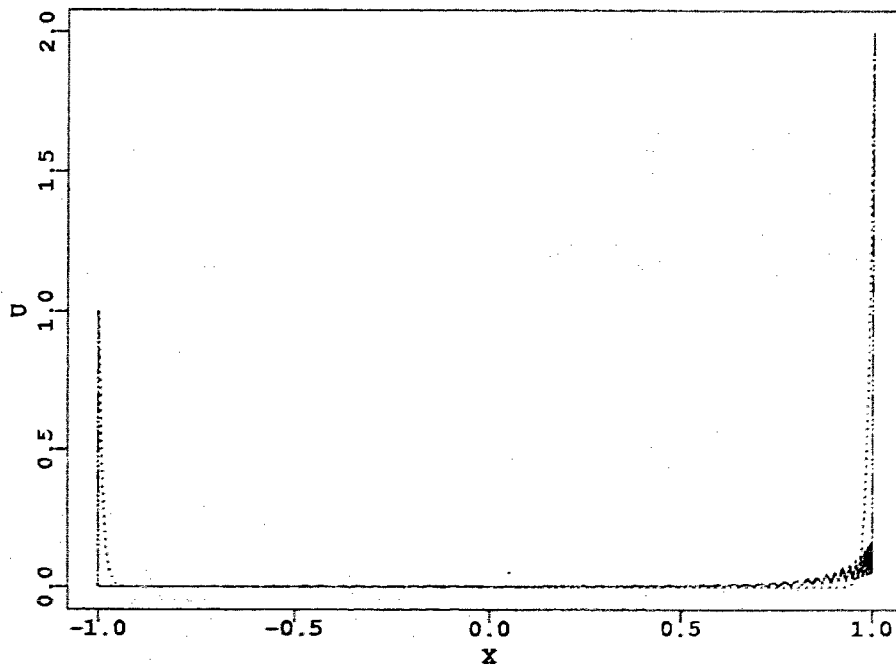
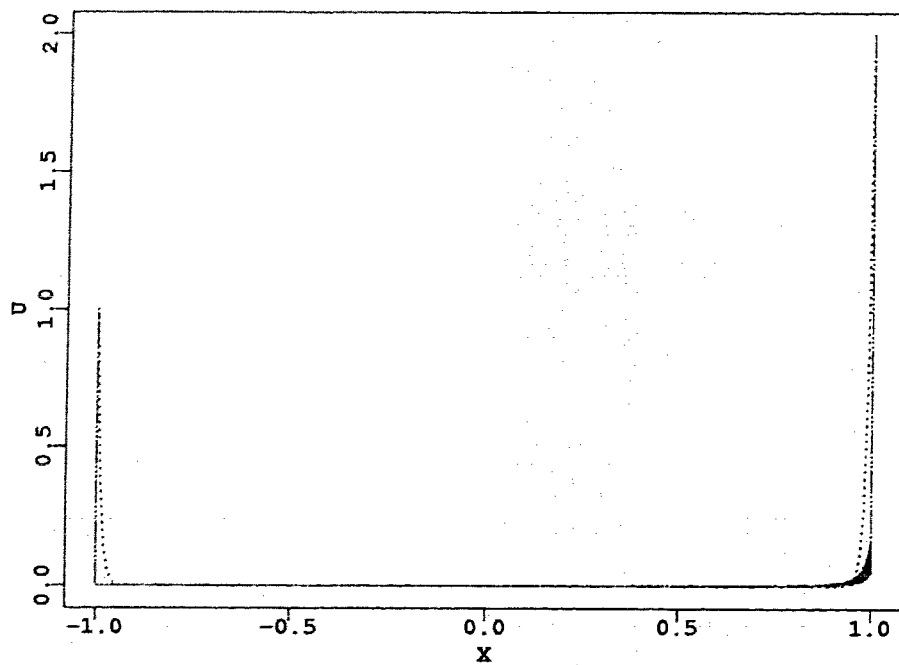
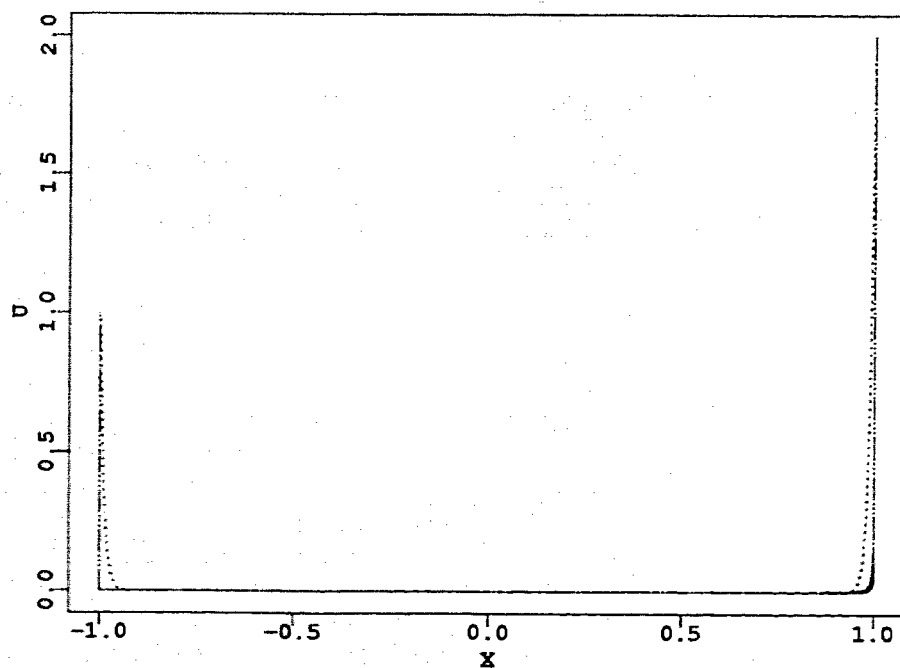
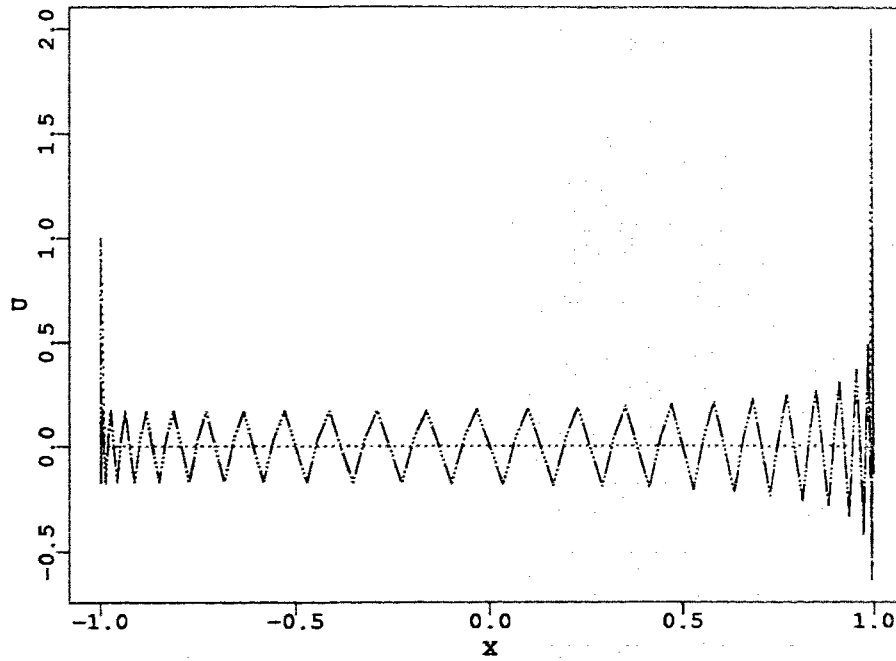
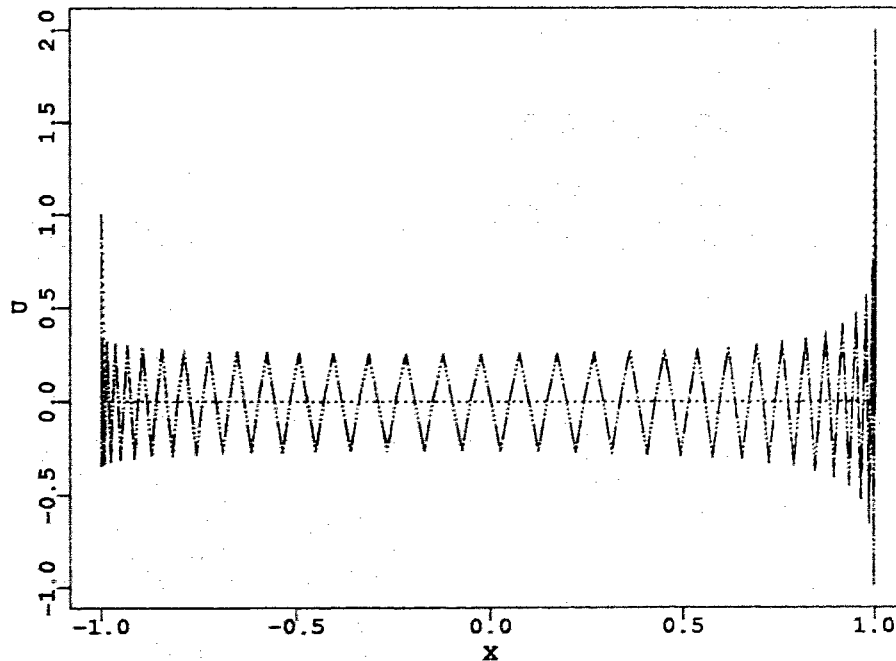


Figure 4.4: Example 2, Chebyshev collocation method, $\epsilon = 10^{-2}$, $N = 128$

Figure 4.5: Example 2, Chebyshev collocation method, $\epsilon = 10^{-2}$, $N = 256$ Figure 4.6: Example 2, Chebyshev collocation method, $\epsilon = 10^{-2}$, $N = 512$

Figure 4.7: Example 2, Chebyshev collocation method, $\epsilon = 10^{-4}$, $N = 96$ Figure 4.8: Example 2, Chebyshev collocation method, $\epsilon = 10^{-4}$, $N = 128$

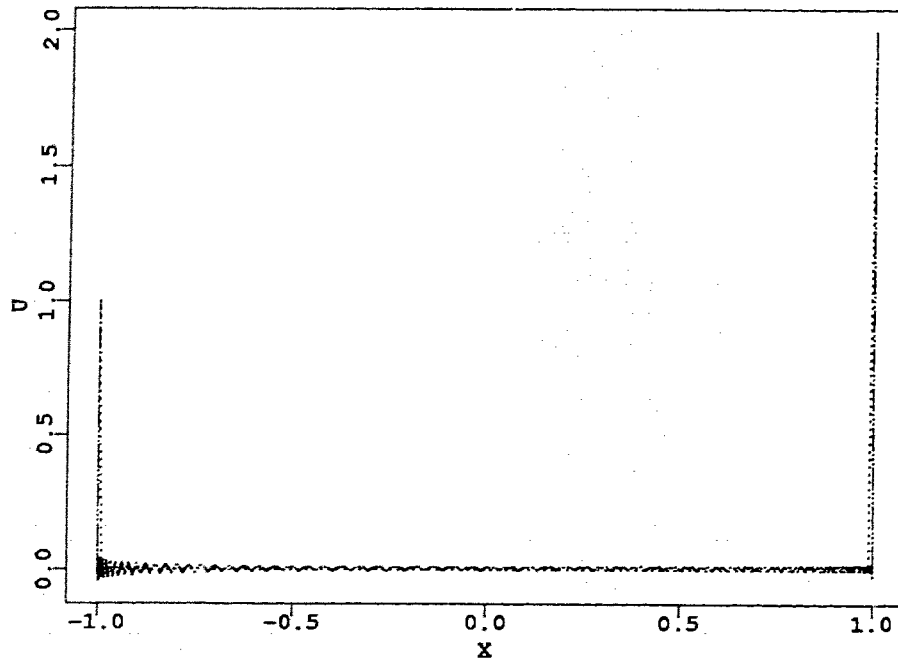


Figure 4.9: Example 2, Chebyshev collocation method, $\epsilon = 10^{-4}$, $N = 256$

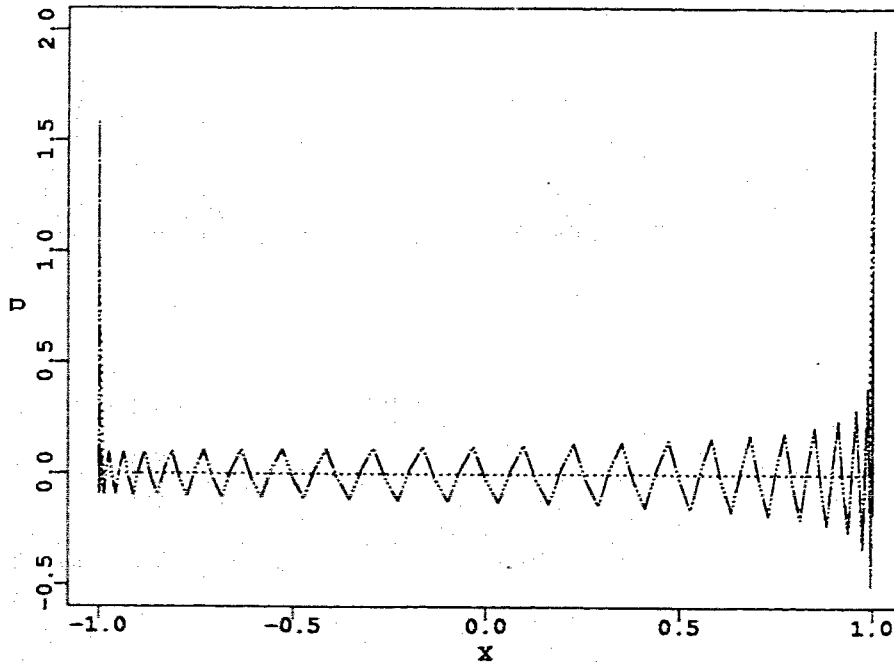
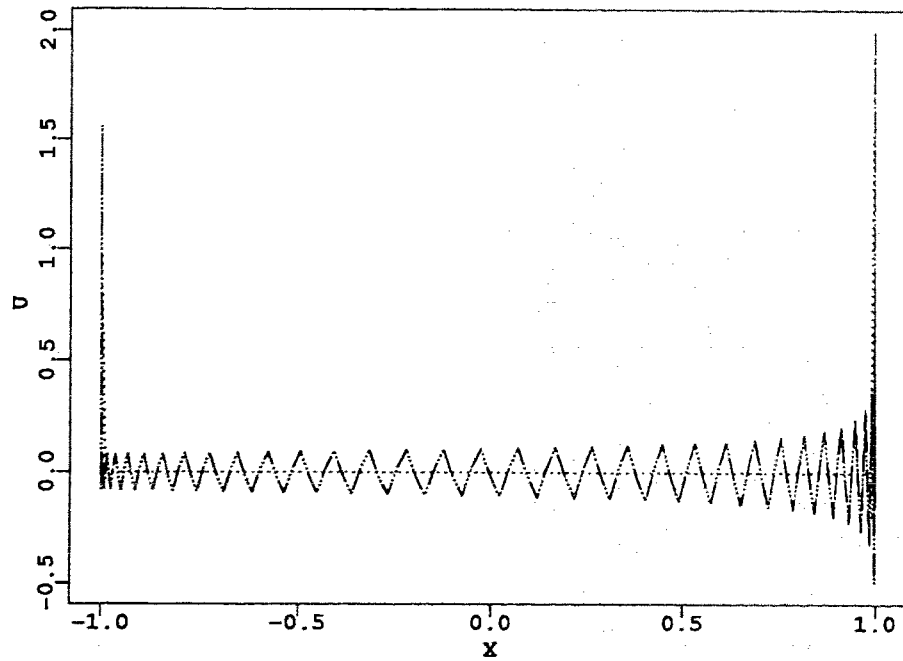
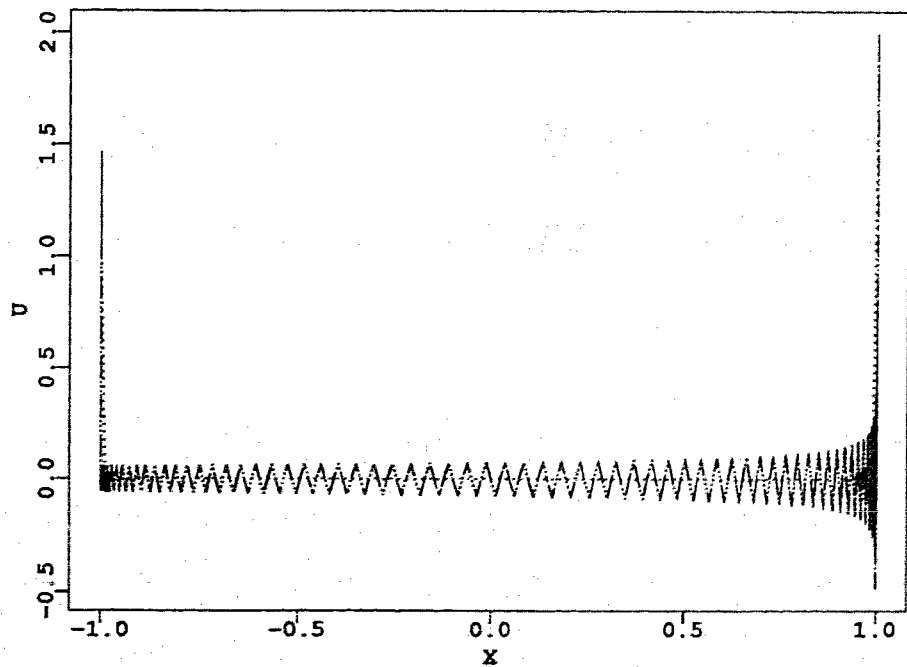


Figure 4.10: Example 2, Chebyshev collocation method, $\epsilon = 10^{-6}$, $N = 96$

Figure 4.11: Example 2, Chebyshev collocation method, $\epsilon = 10^{-6}$, $N = 128$ Figure 4.12: Example 2, Chebyshev collocation method, $\epsilon = 10^{-6}$, $N = 256$

Chapter 5

Discussion

In this thesis we have studied three numerical methods for solving singularly perturbed boundary value problems of order 2. The pseudospectral Chebyshev method was little applied to solve singularly perturbed boundary value problems, therefore, three problems have been chosen for our study—the first problem has a layer in the middle of the interval $[-1, 1]$; the second problem has two boundary layers at $x = -1$ and $x = 1$; and the third problem has one boundary layer at $x = -1$. With each problem and with each method, we have considered three cases of singularity—mild ($\epsilon = 10^{-2}$), stiff ($\epsilon = 10^{-4}$), and very stiff ($\epsilon = 10^{-6}$). Our attention in the study has been given to three areas: condition numbers, average errors and cpu times.

The pseudospectral Chebyshev method was studied in Chapter 2. The numerical results in this Chapter show that the condition number of the matrix related to the N -point pseudospectral Chebyshev approximation is smaller than N^4 , but it depends on the singularity parameter ϵ . In general, the smaller the ϵ , the larger the condition number.

For $\epsilon = 10^{-2}$ and $\epsilon = 10^{-4}$, the pseudospectral Chebyshev method can be successfully applied to solve the three singularly perturbed BVPs, even though for $\epsilon = 10^{-4}$, a relatively large N ($N = 256$ or 512) is needed to reach a fairly good solution. For $\epsilon = 10^{-6}$, the pseudospectral Chebyshev method succeeds in solving the first problem. When applied to

solve the second and the third problems, the method fails to produce converging solutions for N as large as 512. The failure in these cases is most serious in the middle of the interval $[-1, 1]$, where the exact solutions are flat and smooth, while the numerical solutions have large oscillations.

The method of transformation to the circle was studied in Chapter 3. By transforming the problem defined on $x \in [-1, 1]$ into an equivalent problem defined on the unit circle through $x = \cos \phi$, this method proceeds to apply the Fourier spectral method. The numerical results in this chapter also show that the condition number of the N -point approximation increases as the singularity parameter ϵ decreases.

The numerical solutions produced with the method of transformation to the circle are of similar quality when compared with those of the numerical solutions produced with the pseudospectral Chebyshev method. However, the transformation method does outperform the pseudospectral Chebyshev method for the mildly singularly perturbed problems ($\epsilon = 10^{-2}$). For the stiff ($\epsilon = 10^{-4}$) and very stiff ($\epsilon = 10^{-6}$) problems, there is no dominance in performance; sometimes the transformation method does better; sometimes the pseudospectral Chebyshev method does better. We compare the condition number for above two methods as shown in Table 5.1 to Table 5.6.

It is noticed that when $\epsilon = 10^{-6}$, both the pseudospectral Chebyshev method and the transformation method failed to produce acceptable numerical solutions in all the cases related to the second and the third examples, for N as large as 512. The failure is in the middle of the interval $[-1, 1]$, where large oscillations occur in the numerical solutions even when $N = 512$. One possible reason for this failure is that there are always fewer collocation points near $x = 0$ than near $x = -1$ and $x = 1$ in the above two methods. (For the transformation method, think in terms of the inverse transformation $\phi = \arccos(x)$.)

The two iterative methods, the conjugate gradient iteration applied to the normal equations (CGNR) and the biorthogonalization algorithm adapted from the biconjugate gradient iteration (CGS) were also studied in Chapter 3. We had hoped to see a good performance

of these two iterative methods, but only the CGNR method converged.

Other effort has also been made to study the possibility of improving the quality of our numerical solutions. For instance, in the study of the pseudospectral Chebyshev method, we also tried to use Chebyshev polynomials to construct approximations (known as the Chebyshev collocation method) in addition to the standard use of Lagrange polynomials. The condition number appears to be worse.

Finally, we will briefly analyze the numerical errors. We consider the linear system $Ax = b$, then the relative error in x [10] is

$$\frac{\|\tilde{x} - x\|}{\|x\|} \leq k_2(A) \left(\frac{\|\tilde{b} - b\|}{\|b\|} \right)$$

where \tilde{x} and \tilde{b} are the numerical results, i.e. $A\tilde{x} = \tilde{b}$. Thus, the relative error in x can be the condition number $k_2(A)$ times the relative error in A and b . In this sense, the condition number $k_2(A)$ quantifies the sensitivity of the $Ax = b$ problem. On the other hand, by a singular value decomposition (SVD) analysis, the SVD of A is

$$A = U\Sigma V^T = \sum_{i=1}^n \sigma_i u_i v_i^T,$$

where $U = (u_1, u_2, \dots, u_n)$, $V = (v_1, v_2, \dots, v_n)$ and $\Sigma = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_n)$ with $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n \geq 0$. Then

$$x = A^{-1}b = (U\Sigma V^T)^{-1}b = \sum_{i=1}^n \frac{u_i^T b}{\sigma_i} v_i,$$

$$\text{and } \tilde{x} = A^{-1}\tilde{b} = \sum_{i=1}^n \frac{u_i^T \tilde{b}}{\sigma_i} v_i.$$

The absolute error in x is

$$x - \tilde{x} = \sum_{i=1}^n \frac{u_i^T (b - \tilde{b})}{\sigma_i} v_i.$$

Let

$$b = \sum_{i=1}^n b_i u_i \tag{5.1}$$

$$\delta b = b - \tilde{b} = \sum_{i=1}^n \epsilon_i b_i u_i \quad (\text{where } \epsilon_i = \pm \epsilon),$$

Here we assume that every component in \tilde{b} has the same relative error

$$\delta x = x - \tilde{x} = \sum_{i=1}^n \frac{u_i^T (b - \tilde{b})}{\sigma_i} v_i.$$

Therefore, we have

$$\begin{aligned} \|x\|_2^2 &\approx \sum_{i=1}^n \frac{b_i^2}{\sigma_i^2}, \\ \|b\|_2^2 &\approx \sum_{i=1}^n b_i^2, \\ \|\delta x\|_2^2 &\approx \epsilon^2 \sum_{i=1}^n \frac{b_i^2}{\sigma_i^2}, \\ \|\delta b\|_2^2 &\approx \epsilon^2 \sum_{i=1}^n b_i^2. \end{aligned}$$

The relative errors in x and b have the same order,

$$\frac{\|\delta x\|_2^2 / \|x\|_2^2}{\|\delta b\|_2^2 / \|b\|_2^2} \approx \frac{\epsilon}{\epsilon} = 1. \quad (5.2)$$

We have plotted the singular vectors versus Chebyshev points for Example 1 (with $\epsilon = 10^{-2}$ and $N = 96$) as shown in Figure 5.1 to Figure 5.6. The behaviour of the singular values and the coefficients $\{b_i\}$ of the equation (5.1) are shown in Figure 5.7 and Figure 5.8. Based on our numerical experiments, we can see that the numerical solutions are accurate despite the large condition numbers. This needs to be contrasted with the worst case scenario, namely that $b = b_1 u_1$ and $\delta b = \beta_n u_n$, i.e. the right hand side is the direction of the singular vector associated with the largest singular value σ_1 , but the perturbation is only in the direction of u_n associated with the smallest singular value σ_n . Our investigation shows that the singular vectors associated with large singular values are highly oscillatory, whereas the ones associated with small σ_k 's are smooth. This is expected since A describes a differentiation process. Now, although the right hand side of our BVPs may be smooth, the boundary conditions are incorporated in the vector b , making it a nonsmooth vector (at least for problems which generate interior or boundary layers), as is illustrated by Figure 5.8. However, the full effect of the large condition number $k_2(A) = \sigma_1/\sigma_n$ could only be felt, if the perturbation in b were very smooth (associated with the small singular values), which

is extremely unlikely for perturbation generated by finite precision arithmetic. Thus it is plausible that (5.2) holds. This also explains why the transformation to the circle, although decreasing the condition numbers, does not promote higher accuracy.

N	Direct Method	Circle Technique
32	6.4×10^2	7.1×10^2
64	8.3×10^3	2.3×10^3
96	1.4×10^5	4.7×10^3
128	1.2×10^5	7.9×10^3

Table 5.1: Example 1, the condition number for two methods, $\epsilon = 10^{-2}$

N	Direct Method	Circle Technique
32	4.5×10^4	3.2×10^3
64	7.7×10^4	2.2×10^4
96	1.3×10^5	7.7×10^4
128	1.4×10^5	1.7×10^5

Table 5.2: Example 1, the condition number for two methods, $\epsilon = 10^{-4}$

N	Direct Method	Circle Technique
32	5.8×10^6	5.3×10^5
64	1.2×10^7	6.3×10^5
96	1.8×10^7	5.8×10^5
128	2.4×10^7	5.3×10^5

Table 5.3: Example 1, the condition number for two methods, $\epsilon = 10^{-6}$

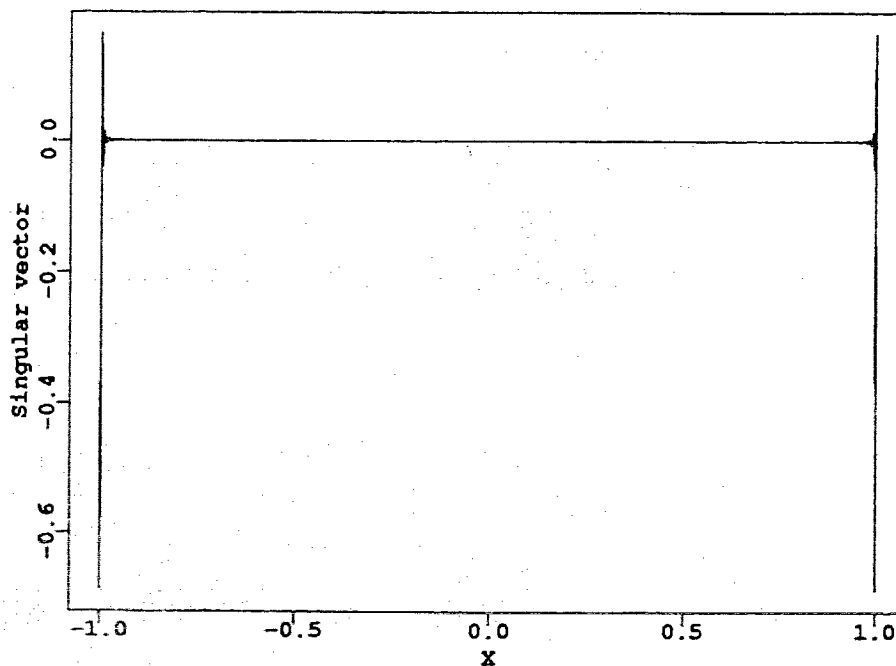
N	Direct Method	Circle Technique
32	3.0×10^3	9.2×10^1
64	5.2×10^4	3.3×10^2
96	2.7×10^5	6.5×10^2
128	8.8×10^5	1.1×10^3

Table 5.4: Example 2, the condition number for two methods, $\epsilon = 10^{-2}$

N	Direct Method	Circle Technique
128	3.3×10^4	1.5×10^6
256	2.5×10^5	3.7×10^5
512	3.9×10^6	1.2×10^5

Table 5.5: Example 2, the condition number for two methods, $\epsilon = 10^{-4}$

N	Direct Method	Circle Technique
128	7.1×10^3	4.0×10^5
256	2.3×10^4	3.9×10^6

Table 5.6: Example 2, the condition number for two methods, $\epsilon = 10^{-6}$ Figure 5.1: Example 1, singular vector (u_1) versus Chebyshev points, $\epsilon = 10^{-2}$, $N = 96$

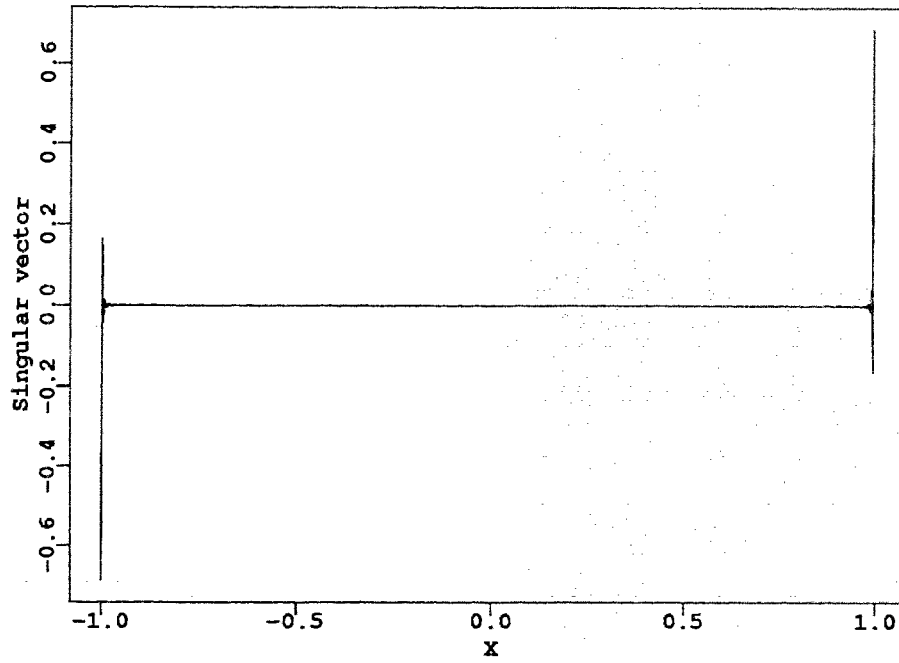


Figure 5.2: Example 1, singular vector (u_2) versus Chebyshev points, $\epsilon = 10^{-2}$, $N = 96$

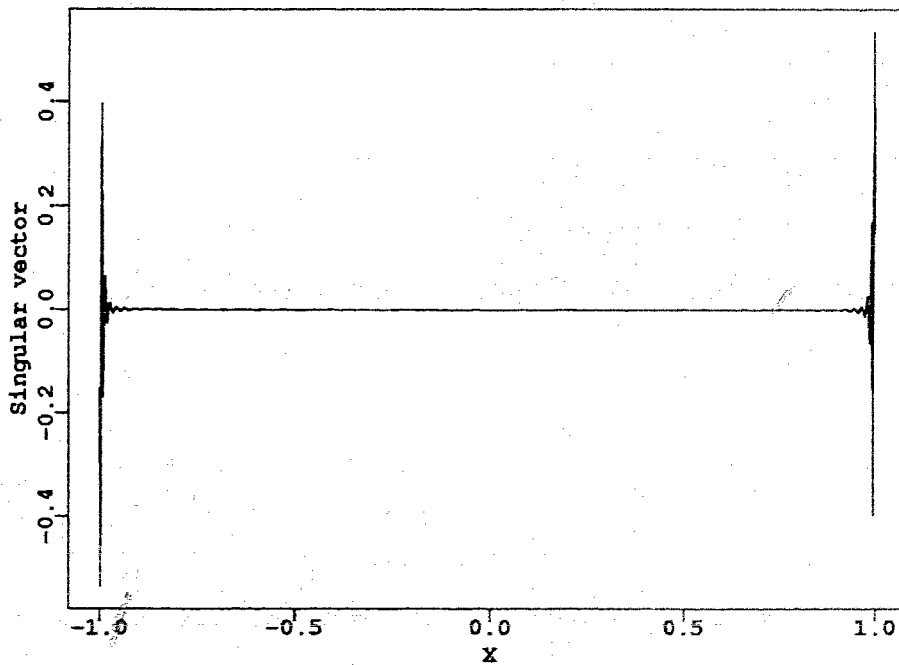


Figure 5.3: Example 1, singular vector (u_3) versus Chebyshev points, $\epsilon = 10^{-2}$, $N = 96$

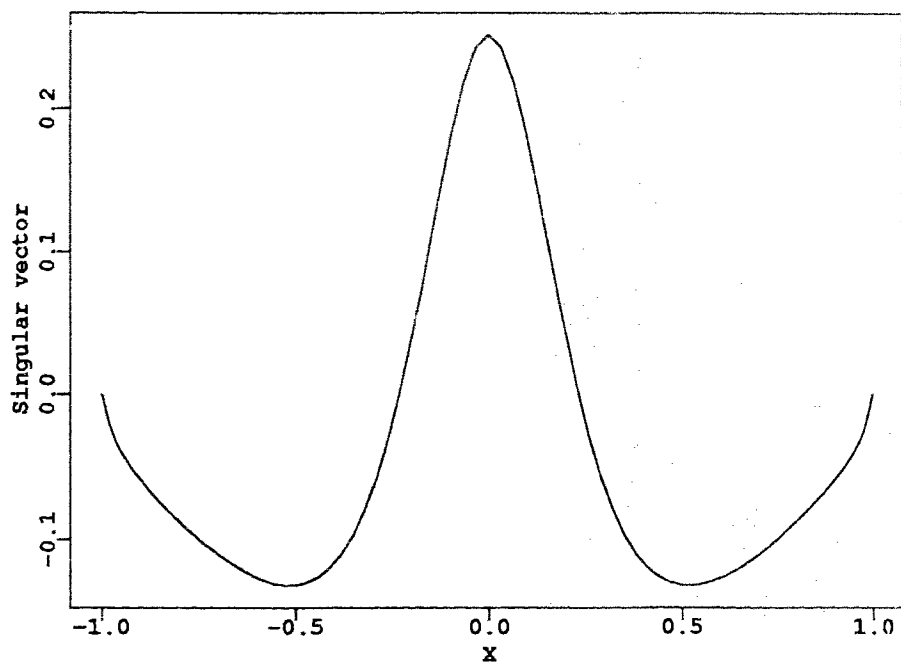


Figure 5.4: Example 1, singular vector (u_{93}) versus Chebyshev points, $\epsilon = 10^{-2}$, $N = 96$

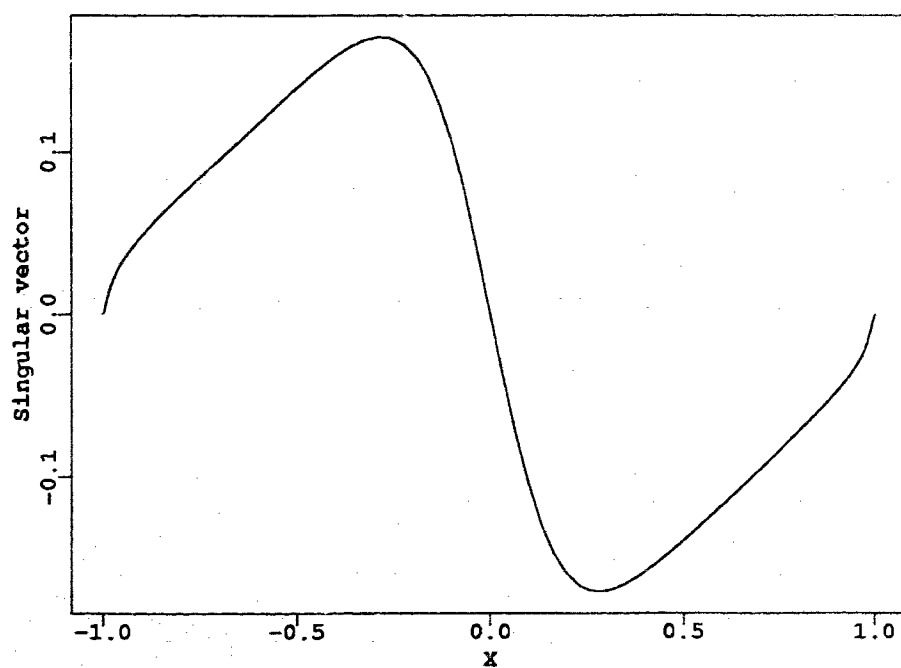


Figure 5.5: Example 1, singular vector (u_{94}) versus Chebyshev points, $\epsilon = 10^{-2}$, $N = 96$

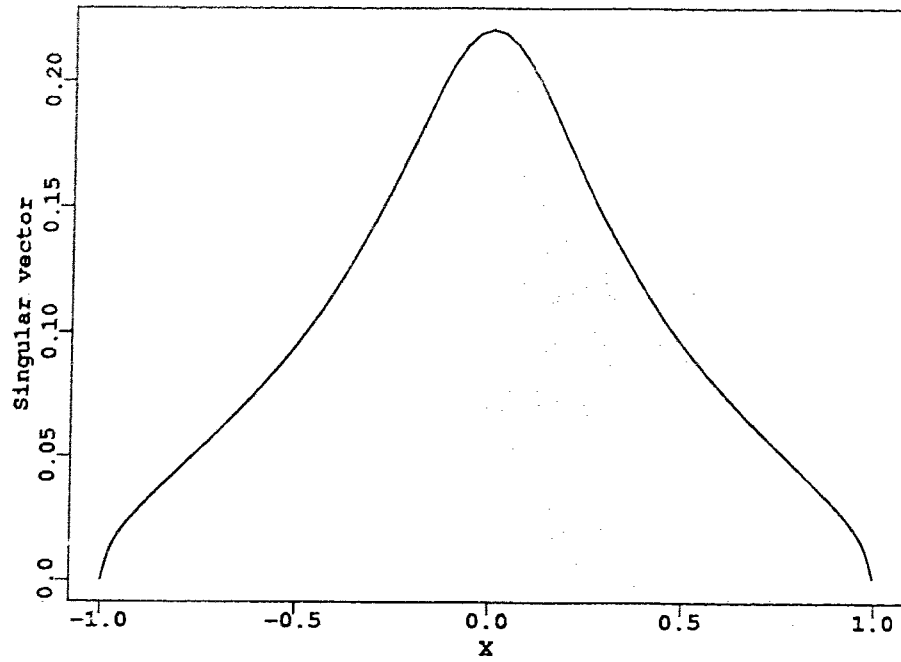


Figure 5.6: Example 1, singular vector (u_{95}) versus Chebyshev points, $\epsilon = 10^{-2}$, $N = 96$

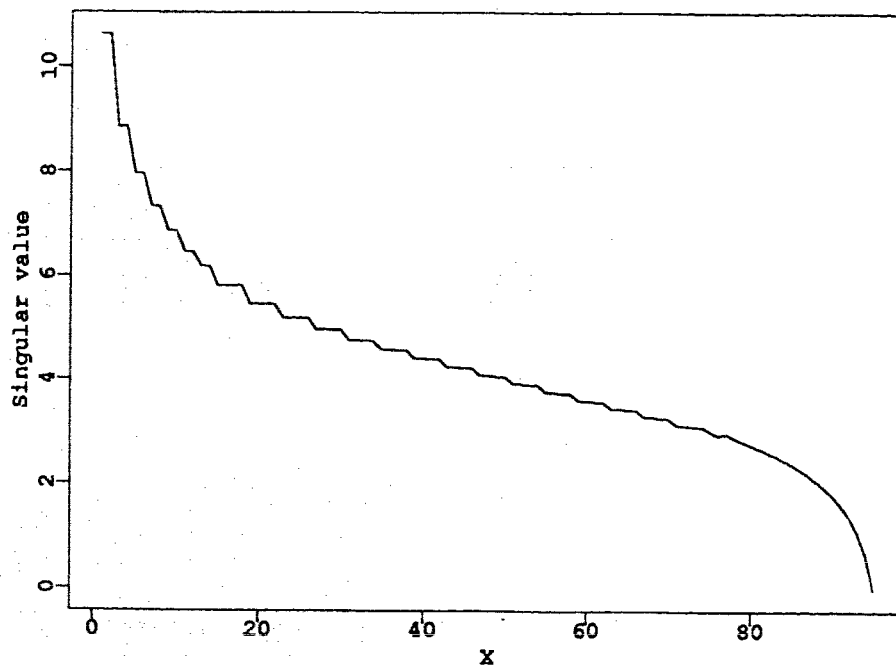


Figure 5.7: Example 1, singular values $\epsilon = 10^{-2}$, $N = 96$

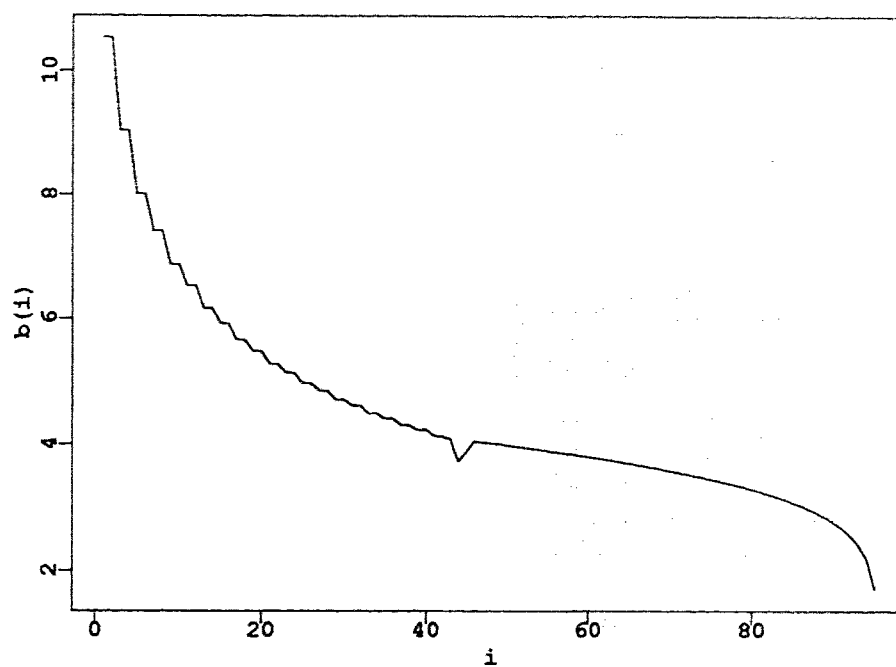


Figure 5.8: Example 1, coefficients $\{b_i\}$ of the equation (5.1) $\epsilon = 10^{-2}$, $N = 96$

Bibliography

- [1] Ascher, U. M., R. M. M. Mattheij, R. M. M. and Russell, R. D., *Numerical Solution of Boundary Value Problems for Ordinary Differential Equations*, Prentice-Hall Inc., 1988.
- [2] Atkinson, K. E., *A Survey of Numerical Methods for the Solution of Fredholm Integral Equations of the Second Kind*, Society for Industrial and Applied Mathematics. 1976.
- [3] Becker, R. A., Chambers, J. M. and Wilks, A. R., *The New S Language: A Programming Environment for Data Analysis and Graphics*, Wadsworth & Brooks/Cole Advanced Books & Software, Pacific Grove, California, 1988.
- [4] Berrut, J. P., A Pseudospectral Chebyshev Method with Preliminary Transform to the Circle: O. D. E., Manuscript, 1990.
- [5] Boyd, J. P., *Chebyshev & Fourier Spectral Methods*, Lecture Notes in Engineering, **49**, Brebbia, C. A. and Orszag, S. A. ed., Springer-Verlag, 1989.
- [6] Canuto, C. and Quarteroni, A., Preconditioned Minimal Residual Methods for Chebyshev Spectral Calculations, *J. Comput. Phys.*, **60**, 315-337, 1985.
- [7] Dongarra, J. J., Moler, C. B. and Bunch, J. R., Stewart, G. W., *Lapack users' Guide*, SIAM Philadelphia, 1979.

- [8] Funaro, D., A Preconditioning Matrix for The Chebyshev Differencing Operator, *SIAM J. Numer. Anal.*, **24**, 1024-1031, 1987.
- [9] Funaro, D., Some Results about the Spectrum of Chebyshev Differencing Operator, *In the proceedings of the International Symposium of Numerical Analysis* (Madrid, Sept. 1985).
- [10] Golub, G. H. and Van Loan, C. F., *Matrix Computations*, second edition, The Johns Hopkins University Press, Baltimore and London, 1989.
- [11] Greengard, L., Spectral Integration and Two - Point Boundary Value, *SIAM J. Numer. Anal.*, **28**, 1071-1080, 1991.
- [12] Gottlieb, D., Hussaini, M. T., and Orszag, S. A., Theory and Applications of Spectral Methods, Report No. 83-66, 1983.
- [13] Gutknecht, M. H., Variants of BICGSTAB for Matrices with Complex Spectrum, IPS Research report No. 91-14, ETH-Zentrum CH-8092 Zurich, 1991.
- [14] Hussaini, M. Y., and Zang, T. A., Iterative Spectral Method and Spectral Solution to Compressible Flows, *Spectral Methods for Partial Differential Equations* (R. Voigt, D. Gottlieb, Y. Hussaini Eds.), SIAM, Philadelphia, 1984.
- [15] Gottlieb, D., and Orszag, S. A., *Numerical Analysis of Spectral Methods: Theory and Applications*, CBMS regional Conference Series in Applied Math., **26**, SIAM Philadelphia, 1977.
- [16] Nachtigal, N. M., Reddy, S. C., and Trefethen, L. N., How Fast Are Nonsymmetric Matrix Iterations?, Manuscript, Massachusetts Institute of Technology, Cambridge, 1991.
- [17] Orszag, S. A., Spectral Methods for Problems in Complex Geometries, *J. Comput. Phys.*, **37**, 70-92, 1980.

- [18] Prenter, P. M., *Splines and Variational Methods*, John Wiley & Sons, New York, 1975.
- [19] Smith, R. C., Bogar, G. A., Bower, K. L., and Lund, J., The Sinc-Galerkin Method for Fourth-Order Differential Equations, *SIAM J. Numer. Anal.*, **28**, 760-788, 1991.
- [20] Sonneveld, P., CGS, A Fast Lanczos - Type Solver for Nonsymmetric Linear Systems, *SIAM J. Sci. Stat. Comp.*, **10**, 36-52, 1989.
- [21] Don, W. S. and Solomonoff, A., *Accuracy and Speed in Computing the Chebyshev Collocation Derivative*, Division of Applied Mathematics, Brown University, Providence, 1991.
- [22] Yamasaki, A., *New Preconditioners Based on Low-Rank Elimination*, Numerical Analysis Report 89 - 10, Massachusetts Institute of Technology, Cambridge, 1989.
- [23] Wright, K., Chebyshev Collocation methods for Ordinary Differential Equations, *J. Comput.*, **16**, 358-365, 1964.
- [24] Huang, W., and Sloan, D. M., A New Pseudospectral Method with Upwind Features. Manuscript, 1992.