

2002

Chebyshev pseudospectral methods for conservation laws with source terms and applications to multiphase flow

Scott Alan Sarra
West Virginia University

Follow this and additional works at: <https://researchrepository.wvu.edu/etd>

Recommended Citation

Sarra, Scott Alan, "Chebyshev pseudospectral methods for conservation laws with source terms and applications to multiphase flow" (2002). *Graduate Theses, Dissertations, and Problem Reports*. 1642. <https://researchrepository.wvu.edu/etd/1642>

This Dissertation is protected by copyright and/or related rights. It has been brought to you by the The Research Repository @ WVU with permission from the rights-holder(s). You are free to use this Dissertation in any way that is permitted by the copyright and related rights legislation that applies to your use. For other uses you must obtain permission from the rights-holder(s) directly, unless additional rights are indicated by a Creative Commons license in the record and/ or on the work itself. This Dissertation has been accepted for inclusion in WVU Graduate Theses, Dissertations, and Problem Reports collection by an authorized administrator of The Research Repository @ WVU. For more information, please contact researchrepository@mail.wvu.edu.

**Chebyshev Pseudospectral Methods for Conservation
Laws with Source Terms and Applications to Multiphase
Flow**

Scott A. Sarra

Dissertation submitted to the College of Arts and Sciences at West
Virginia University in partial fulfillment of the requirements for
the degree of

Doctor of Philosophy
in
Mathematics

Ian Christie, Ph.D., Chair
Joe Wilder, Ph.D.
Gary Ganser, Ph.D.
Ismail Celik, Ph.D.
Sherman Riemenschneider, Ph.D.

Department of Mathematics

Morgantown, West Virginia
2002

Keywords: Pseudospectral, Collocation, Chebyshev, Gibbs
Phenomenon, Postprocessing, Gegenbauer, Multiphase Flow, Edge
Detection, Fluidized Bed.

Copyright 2002 by Scott Sarra

**Chebyshev Pseudospectral Methods for Conservation
Laws with Source Terms and Applications to Multiphase
Flow**

Scott Sarra

ABSTRACT

Pseudospectral methods are well known to produce superior results for the solution of partial differential equations whose solutions have a certain amount of regularity. Recent advances have made possible the use of spectral methods for the solution of conservation laws whose solutions may contain shocks. We use a recently described Super Spectral Viscosity method to obtain stable approximations of Systems of Nonlinear Hyperbolic Conservation Laws. A recently developed postprocessing method, which is theoretically capable of completely removing the Gibbs phenomenon from the Super Spectral Viscosity approximation, is examined. The postprocessing method has shown great promise when applied in some simple cases. We discuss its application to more complicated problems and examine the possibility of the method being used as a "black box" postprocessing method. Applications to multiphase fluid flow are made.

Contents

Introduction	ix
I Numerical Method	1
1 Chebyshev Pseudospectral Method	2
1.1 Derivative Matrix	3
1.1.1 Parity Matrix Multiplication	5
1.2 Speed Considerations	6
1.3 Spectral Accuracy	7
2 Implementation Issues	9
2.1 Time Integration	9
2.2 Boundary Conditions	10
2.3 Coordinate Transformations	11
2.3.1 Interval [-1,1] to interval [a,b]	12
2.3.2 Kosloff/Tal-Ezer Mapping	12
2.3.3 Center Map	13
2.3.4 Tangent Map	14
2.4 Problem Formulation	14
2.4.1 One-dimensional unsplit	15
2.4.2 One-dimensional Strang split	16
2.4.3 Two-dimensional unsplit	16
2.4.4 Two-dimensional Strang split	17
II Spectral Filters and Spectral Viscosity	18
3 Spectral Filters	20

4	Super Spectral Viscosity	23
4.1	Spatially varying viscosity	27
III	Postprocessing	28
5	Edge Detection	31
5.1	Edge detection in the first derivative	33
5.2	Two Dimensions	33
6	Gegenbauer Reconstruction Procedure	34
6.1	Gegenbauer Reconstruction	34
6.1.1	Reconstruction on a subinterval	36
6.1.2	Truncation and Regularization Errors	37
6.1.3	Convergence Theorem	37
6.1.4	Choice of Reconstruction Parameters	39
6.1.5	The collocation grid	41
6.1.6	Computational Expense	42
6.1.7	Roundoff error	43
6.1.8	A Hybrid Approach	44
6.1.9	Two Dimensions	44
IV	Numerical Examples	45
7	Hyperbolic Heat Transfer	47
7.1	Problem 1	49
7.2	Problem 2	50
8	Reactive Euler Equations	54
9	One-Dimensional Fluidized Bed Equations	57
9.1	Fluidized Bed Equations	58
9.1.1	Vacuums and unphysical particle concentrations	58
9.2	Numerical Results	59
9.2.1	Homogeneous system	59
9.2.2	Slugging Problem	64
9.3	Selection of Edge Detection Parameters	71
9.4	Selection of Postprocessing Parameters	72
9.5	Conclusions	74

10 Two-Dimensional Fluidized Bed Equations	76
10.1 Fluidized Bed Model	77
10.2 Numerical Results	82
10.2.1 Choice of Collocation Grid	84
10.2.2 32 by 32 Grid	87
10.2.3 64 by 64 Grid	88
10.3 Conclusions	88
V Summary	90
11 Conclusions	91
11.1 Summary of Results	91
11.2 Open Questions and Further Work	93
A Chebyshev Polynomials	94
B Gegenbauer Polynomials	96
C Other Postprocessing methods	98
D Software	100
D.1 Environments	100
D.2 Spectral	100
D.3 Postprocessing	101
D.4 Source Code in print	101

List of Tables

1.1	spectral differentiation speed	7
2.1	map (2.7) parameter ranges	13
4.1	SSV parameter limits	24
8.1	local reconstruction parameters	56
9.1	shock-rarefaction, local reconstruction parameters . .	61
9.2	$t = 0.5$, local reconstruction parameters	67
9.3	$t=4$, local reconstruction parameters	69

List of Figures

1.1	convergence rate, fourth-order finite difference	7
1.2	convergence rate, spectral	8
2.1	grid point distribution	14
3.1	exponential filter comparison	22
4.1	step function and Chebyshev approximation	29
6.1	Function and Chebyshev partial sum approximation	40
6.2	Square wave and Chebyshev approximation	41
6.3	Mapped grid error vs. CGL grid error	42
7.1	spectral approximation	48
7.2	edge locations	49
7.3	postprocessed vs. exact	50
7.4	pointwise error, postprocessed vs. exact	51
7.5	Problem 1: MacCormack's vs. exact	52
7.6	spectral approximation	52
7.7	edge locations	53
7.8	postprocessed vs. exact	53
8.1	SSV approximation (left), postprocessed (right)	54
8.2	spectral postprocessed (solid) vs. NT	56
9.1	SSV approximation vs. exact	60
9.2	postprocessed vs. exact	61
9.3	problem 1, without SSV	62
9.4	SSV approximation vs. exact	63
9.5	edge locations, shock/shock	63
9.6	postprocessed vs. exact, shock/shock	64

9.7	initial concentration α , slugging problem	66
9.8	SSV vs. postprocessed, $t=0.5$	67
9.9	edge locations, $t=0.5$	68
9.10	spectral postprocessed vs. reference, $t=0.5$	68
9.11	SSV solution, $t=4$	69
9.12	edge locations, $t=4$	70
9.13	postprocessed, $t=4$	70
9.14	edge series	71
9.15	SSV, dashed; PP, solid; FD, dash-dotted	72
9.16	SSV, dashed; PP, solid; FD, dash-dotted	73
9.17	SSV, dashed; PP, solid; FD, dash-dotted	73
9.18	SSV, dashed; PP, solid; FD, dash-dotted	74
10.1	2d fluidized bed	77
10.2	CGL grid	78
10.3	mapped grid, $\gamma = 0.86$	78
10.4	mapped grid, $\gamma = 0.9999$	79
10.5	32 by 32 grid, $t = 1.5$ to $t = 3.0$	80
10.6	32 by 32 grid, $t = 3.5$ to $t = 5.0$	81
10.7	32 by 32 grid, $t = 5.5$ to $t = 6.0$	82
10.8	32 by 32 grid velocity field, $t = 4.0$	83
10.9	64 by 64 grid, $t = 1.25$ to $t = 2.0$	84
10.10	64 by 64 grid, $t = 2.25$ to $t = 3.0$	85
10.11	64 by 64 grid, $t = 3.25$ to $t = 4.0$	86
10.12	64 by 64 grid stream function, $t = 0.25$ and $t = 4.0$	87

Introduction

In this work we are interested in the initial boundary value problem for systems of nonlinear conservation laws with source terms in one and two space dimensions. The model problem in one space dimension is of the form

$$\begin{aligned}u_t + f(u)_x &= b(u), \\ u(x, 0) &= h(x)\end{aligned}\tag{1}$$

with boundary data prescribed as appropriate.

The superiority of spectral methods for the solution of partial differential equations for problems whose solutions have a certain amount of inherent regularity has been well established, see for example [7, 21, 31]. Until recently, spectral methods were deemed inappropriate for the solution of many hyperbolic problems, in particular nonlinear conservation laws whose solutions may develop shock waves. Spectral methods do not converge to the entropy solution in this case [61]. This non-convergence can be overcome by spectral viscosity methods without sacrificing spectral accuracy. The surge in interest in spectral methods for conservation laws could possibly be attributed to a paper of Lax [43] in 1978 in which he concludes the following about numerical methods for Partial Differential Equations:

...a method with low resolution can not be very accurate; the converse does not follow, i.e., a method with high resolution need not be highly accurate. But at least it furnishes approximations that contain enough information from which a better approximation may be extracted by a postprocessing...even if that is not so, a method with high resolution is more likely to preserve qualitative features of solutions...

In fact, the stable spectral viscosity approximations will be reduced to first order accuracy and will contain oscillations in the neighborhoods of shocks due to the Gibbs-Wilbraham phenomenon. The problem is reduced to one of signal processing in order to recover spectral accuracy.

In this work, the only postprocessing technique that we will discuss is the Gegenbauer Reconstruction Procedure (GRP). One of our goals is to examine if the GRP, which has shown great promise on some simple examples, can be used to successfully postprocess PDE solutions containing a complex and varying structure. The GRP will need to know the exact location of all discontinuities in the solution and in all order derivatives of the solution in order to be successful. The GRP is theoretically capable of recovering spectral accuracy over the entire interval, even at the location of discontinuities.

It should be noted that other, more robust postprocessing methods (see Appendix C) exist which are able to recover spectral accuracy up to within a neighborhood of a discontinuity. Some of the methods may be used without knowing the location of the discontinuities.

The previous applications of the GRP postprocessing method in the literature have been to relatively simple problems: inviscid Burgers Equation [25], the Euler Equations [25], and Shallow Water Equations [23], with Riemann initial conditions. In these simple examples, the parameters in the postprocessing method could be chosen globally with a successful result. With more complicated solutions, this approach fails. Additionally, in previous applications of the GRP postprocessing method, only approximations on the standard pseudospectral grid were used, even when a different grid distribution appeared more appropriate. It will be shown that the postprocessing technique can be applied successfully when the approximations are taken from a grid in which the standard set of collocation points has been mapped to a different distribution. This is an important issue, as the standard Chebyshev grid is rarely the correct choice in applications.

A final topic that is of interest is the presence of significant solution features near computational boundaries. It has been noted that the spectral viscosity operator that we are using is spatially varying and goes to zero on the boundaries [5], but no reports of the

consequences or resolutions of this fact appear in the literature.

Part *I* of this thesis will briefly review the Chebyshev pseudospectral methods. Readers familiar with spectral methods may wish to skip this section. Part *II* will examine the modification, a spectral vanishing viscosity, which must be added to the standard Chebyshev pseudospectral method in order to obtain a numerical approximation which converges to the entropy solution of a system of conservation laws. The approximate solution may still contain non-physical Gibbs oscillations and a postprocessing procedure to recover spectral accuracy will be discussed in part *III*. Additionally, a method to locate the position of discontinuities in the approximate solution will be discussed in part *III*. In part *IV*, numerical examples will be presented.

Part I

Numerical Method

Chapter 1

Chebyshev Pseudospectral Method

The standard collocation points for a Chebyshev Pseudospectral method are usually defined by

$$x_j = -\cos\left(\frac{\pi j}{N}\right), \quad j = 0, 1, \dots, N. \quad (1.1)$$

These points are extrema of the N^{th} order Chebyshev polynomial,

$$T_k(x) = \cos(k \arccos(x)). \quad (1.2)$$

The points are often labelled the Chebyshev-Gauss-Lobatto (CGL) points, a name which alludes to the points role in certain quadrature formulas. The CGL points cluster quadratically around the endpoints and are less densely distributed in the interior of the domain.

The Chebyshev Collocation method is based on assuming that an unknown PDE solution, u , can be represented by a global, interpolating, Chebyshev partial sum,

$$u_N(x) = \sum_{n=0}^N a_n T_n(x). \quad (1.3)$$

The discrete Chebyshev coefficients, a_n , are defined by

$$a_n = \frac{2}{N} \frac{1}{c_n} \sum_{j=0}^N \frac{u(x_j) T_n(x_j)}{c_j} \quad \text{where} \quad c_j = \begin{cases} 2 & \text{when } j = 0, N \\ 1 & \text{otherwise.} \end{cases} \quad (1.4)$$

Derivatives of u at the collocation points are approximated by the derivative of the interpolating polynomial evaluated at the collocation points. The first derivative, for example, is defined by,

$$\frac{du}{dx} = \sum_{n=0}^N a_n^{(1)} T_n(x). \quad (1.5)$$

Since $a_{N+1}^{(1)} = 0$ and $a_N^{(1)} = 0$, the non-zero derivative coefficients can be computed in decreasing order by the recurrence relation:

$$c_n a_n^{(1)} = a_{n+2}^{(1)} + 2(n+1)a_{n+1}, \quad n = N-1, \dots, 1, 0. \quad (1.6)$$

The transform pair given by equations (1.3) or (1.5) and (1.4) can be efficiently computed by a fast cosine transform (FCT). Equivalently, the interpolating polynomial and its derivatives can be computed in physical space using matrix multiplication [7]. Special properties of the Chebyshev basis allow for differentiation via parity matrix multiplication [6] (even-odd decomposition [57]), which can be performed by using slightly more than half as many floating point operations as standard matrix multiplication.

1.1 Derivative Matrix

Rather than expressing the unknown solution in terms of a basis as in (1.3), we could choose to introduce a grid and assume that the solution can be expressed as a global interpolation polynomial

$$I_N u(x, t) = \sum_{j=0}^N u_N(x_j, t) l_j(x) \quad (1.7)$$

where l_j represents the Lagrange interpolation polynomial base on the grid, x_j . Then, spatial derivatives are approximated by

$$\frac{\partial u(x, t)}{\partial x} \cong \frac{\partial I_N u(x, t)}{\partial x} = \sum_{j=0}^N u_N(x_j, t) \frac{dl_j(x)}{dx}. \quad (1.8)$$

The Lagrange interpolation polynomials, $l_j(x)$, are of the form

$$l_j(x) = \frac{q(x)}{(x - x_j)q'(x)}, \quad q(x) = \prod_{j=0}^N (x - x_j). \quad (1.9)$$

Since we are using a collocation approach, we are restricting our attention to the approximation of the derivative of $u(x, t)$, at the grid points x_j , and we have

$$\left. \frac{du}{dx} \right|_{x_i} \cong \left. \frac{dI_N u}{dx} \right|_{x_i} = \sum_{j=0}^N u_N(x_j) \left. \frac{dl_j(x)}{dx} \right|_{x_i} = \sum_{j=0}^N u(x_j) D_{ij}, \quad (1.10)$$

where D is referred to as the *differentiation matrix*. Unlike differentiation matrices which result from finite difference methods, the spectral differentiation matrix is full. This is a consequence of the fact that a global interpolation polynomial has been used. It is well known that high order polynomial interpolation on a finite interval is not feasible on an equidistant grid due to Runge phenomenon [15]. When smooth functions are interpolated by polynomials in $N + 1$ equally spaced points, the approximations not only fail to converge in general as $N \rightarrow \infty$, but they get worse at a rate that may be as great as 2^N . In order to avoid Runge phenomenon, it is necessary to choose a grid that clusters quadratically around the endpoints. The simplest set of points that satisfies this requirement is the CGL grid. If the grid is chosen to be the CGL grid (1.1), whose nodes are the zeros of $(1 - x^2) \frac{d}{dx} T_k(x) = 0$, the polynomial that interpolates the spatial derivative in (1.8) becomes

$$I_N u(x, t) = \sum_{j=0}^N u(x_j) h_j(x) = \sum_{j=0}^N u(x_j) \frac{(-1)^{N+1+j} (1-x^2) T_N'(x)}{c_j (x - x_j)}. \quad (1.11)$$

The Chebyshev Differentiation Matrix on the Chebyshev grid can then be expressed as the following antisymmetric matrix with elements

$$D_{ij} = \begin{cases} -\frac{2N^2+1}{6} & i = j = 0 \\ \frac{c_i (-1)^{i+j}}{c_j x_i - x_j} & i \neq j \\ -\frac{x_i}{2(1-x_1^2)} & 0 < i = j < N \\ \frac{2N^2+1}{6} & i = j = N \end{cases} \quad (1.12)$$

where c_j is defined in (1.4). For large N , forming D is susceptible to round off error due to the term $\frac{1}{x_i - x_j}$ which is hard to evaluate accurately. The smallest distance between points on the CGL grid is $O(N^{-2})$, and taking the reciprocal amplifies the error by a large factor. Trigonometric identities can be used to eliminate the subtraction of similar numbers. A form of D more suited to numerical representation is the matrix with elements

$$D_{ij} = \begin{cases} -\frac{2N^2+1}{6} & i = j = 0 \\ -\frac{1}{2} \frac{c_i (-1)^{i+j}}{c_j \sin[\frac{\pi}{2N}(i+j)] \sin[\frac{\pi}{2N}(i-j)]} & i \neq j \\ -\frac{x_i}{2 \sin^2(\frac{\pi i}{2N})} & 0 < i = j < N \\ \frac{2N^2+1}{6} & i = j = N \end{cases} \quad (1.13)$$

Since

$$D_{ij} = -D_{N-i, N-j} \quad i = \frac{N}{2} + 1, \dots, N \quad (1.14)$$

the bottom half of the matrix can be calculated by using the top half of the matrix without using the formula for the bottom half explicitly. $\sin(x)$ and $\sin(\pi - x)$ are the same number, but when x is small, $\sin(x)$ can be calculated within machine precision but $\sin(\pi - x)$ can not [18]. By forming the bottom half of the matrix with the flipping procedure, the matrix is formed more accurately and with less floating point operations.

When using matrix multiplication to calculate spectral derivatives, all operations take place in physical space. The floating point operation count is of $O(N^2)$ which can be severe for large N .

1.1.1 Parity Matrix Multiplication

If the elements of the basis set are all either symmetric or all antisymmetric with respect to $x = 0$, a spectral derivative may be

evaluated in only half as many floating point operations as a standard matrix multiplication. The Chebyshev Differentiation Matrix is such that $D_{ij} = -D_{N-i, N-j}$ which indicates that multiplication via a Parity Matrix Multiplication (PMM) is possible [57]. The algorithm consists of three stages and is different for an even and odd number of grid points. The following description is for an even number of grid points (thus N is odd). In the first stage, the vector u_i is decomposed into its even e_i and odd o_i parts:

$$\begin{aligned} e_i &= u_i + u_{N-i} \\ o_i &= u_i - u_{N-i} \end{aligned} \quad (1.15)$$

for $i = 0, \dots, (N-1)/2$. Next, the derivative of the even and odd parts is computed as

$$\begin{aligned} e'_i &= D_e e \\ o'_i &= D_o o \end{aligned} \quad (1.16)$$

where the elements of D_e and D_o are given by

$$\begin{aligned} (D_e)_{ij} &= \frac{1}{2} (D_{ij} + D_{i, N-j}) \\ (D_o)_{ij} &= \frac{1}{2} (D_{ij} - D_{i, N-j}) \end{aligned} \quad (1.17)$$

for $i = 0, \dots, (N-1)/2$. In the final step, u' is reconstructed from e' and o' :

$$\begin{aligned} u'_i &= o'_i + e'_i \\ u'_{N-i} &= o'_i - e'_i \end{aligned} \quad (1.18)$$

for $i = 0, \dots, (N-1)/2$. The algorithm is very similar to the first step in the Fast Fourier Transform (FFT) where the problem is broken into smaller parts. However, unlike the FFT, it is not possible to continue the decomposition beyond the first step.

1.2 Speed Considerations

The number m for which the FCT becomes faster than matrix multiplication and parity matrix multiplication is both platform and library dependent. The following results are from using the *jSpecLib* [53] library on a Pentium 550 running Windows 98 SE. The benchmark performs 10,000 spectral differentiations on a randomly generated vector of length m with the results report in seconds.

For the specified library and platform, the FCT is faster than parity matrix multiplication for $m \geq 64$.

m	MM	PMM	FCT
16	0.16	0.11	0.28
32	0.49	0.33	0.44
64	2.04	1.15	0.88
128	7.31	3.96	1.82
256	43.12	15.77	3.68
512	217.34	108.64	7.75

Table 1.1: spectral differentiation speed

1.3 Spectral Accuracy

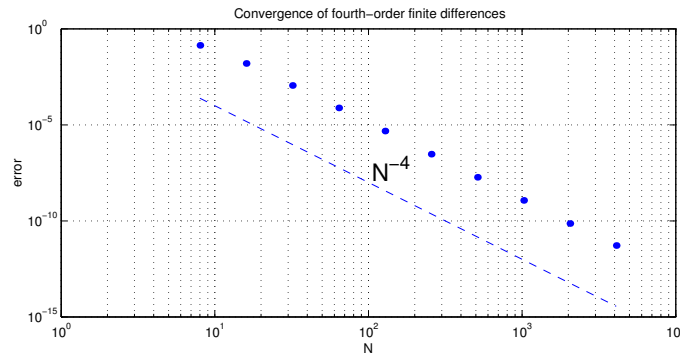


Figure 1.1: convergence rate, fourth-order finite difference

Spectral methods are said to have *spectral accuracy*, or are said to display *exponential convergence*, or are even said to have an *infinite order of accuracy*. Each one of these labels refers to the fact that when $f \in C^\infty$ the error between f and its N^{th} order truncated series decays faster than any power of $\frac{1}{N}$. This is unlike the error in finite difference or finite element methods which typically decrease like $O(N^{-m})$ for some constant m . Figures 1.1 and 1.2 offer a comparison of the convergence rates of a fourth-order finite difference method and a spectral method for differentiating a smooth function. Notice that the spectral errors decrease very rapidly until such high precision is achieved that rounding errors on the computer prevent any further improvement.

In most practical applications the benefit of the spectral method is not the extraordinary accuracy available for large N , but rather the small size of N necessary for a moderately accurate solution.

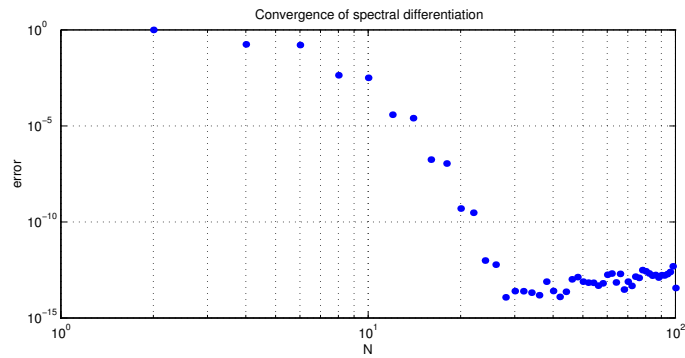


Figure 1.2: convergence rate, spectral

In fact, as will be discussed in a later section, the use of coordinate maps, which may be necessary to reduce the restriction on the stable time step and/or to redistribute the collocation points, may in some cases degrade the accuracy of the spectral method. Even though technically, spectral accuracy is not achieved, the numerical approximation remains much more accurate than alternative methods.

The Gegenbauer reconstruction procedure (chapter 6) recovers pointwise exponential accuracy at all points including the discontinuities themselves of piecewise analytic functions provided that the reconstruction parameters fall within certain bounds. However, it appears that these bounds can be violated, and in fact sometimes must be violated to obtain good results, even though spectral accuracy is not theoretically guaranteed.

Chapter 2

Implementation Issues

2.1 Time Integration

When the spatial discretization is done by spectral methods, the following system of Ordinary Differential Equations (ODEs) results which must be solved to advance the solution in time:

$$\frac{d\mathbf{U}}{dt} = F(\mathbf{U}, t), \quad (2.1)$$

where \mathbf{U} is the vector containing the unknown PDE solution at the collocation points. Any standard method [7] for evaluating systems of ODEs may be used, however, explicit Runge-Kutta (RK) methods are a popular choice and have been used in our numerical examples. In this section, the second, third, and fourth order accurate in time explicit Runge-Kutta methods that we have used are stated with a summary of their properties. The variable Δt represents the fixed time step, t^n is the time level after n steps, and \mathbf{U}^n is the solution at time t^n .

Among the many possible fourth order explicit Runge-Kutta methods, the method (2.2), is often known as the standard fourth order formula.

$$\begin{aligned} \mathbf{k}_1 &= \Delta t F(\mathbf{U}^n, t^n) \\ \mathbf{k}_2 &= \Delta t F(\mathbf{U}^n + 0.5\mathbf{k}_1, t^n + 0.5\Delta t) \\ \mathbf{k}_3 &= \Delta t F(\mathbf{U}^n + 0.5\mathbf{k}_2, t^n + 0.5\Delta t) \\ \mathbf{k}_4 &= \Delta t F(\mathbf{U}^n + \mathbf{k}_3, t^n + \Delta t) \\ \mathbf{U}^{n+1} &= \mathbf{U}^n + \frac{1}{6}(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4) \end{aligned} \quad (2.2)$$

The RK4 scheme requires four levels of storage and four evaluations of F at each time level.

When storage or speed are issues, as is sometimes the case when dealing with systems of PDEs in more than one space dimension, an order of accuracy can be sacrificed to get a less computationally intensive time integration. The low storage, third order Runge-Kutta scheme (2.3) of [56] requires three evaluations of F per time level but only two levels of storage.

$$\begin{cases} \mathbf{U}^1 = \mathbf{U}^n + \Delta t F(\mathbf{U}^n, t^n) \\ \mathbf{U}^2 = \frac{1}{4}(3\mathbf{U}^n + \mathbf{U}^1 + \Delta t F(\mathbf{U}^1, t^n + \Delta t)) \\ \mathbf{U}^{n+1} = \frac{1}{3}(\mathbf{U}^n + 2\mathbf{U}^2 + 2\Delta t F(\mathbf{U}^2, t^n + 0.5\Delta t)) \end{cases} \quad (2.3)$$

When Strang Splitting [59] is used and the source term is dealt with in a step separate from the homogeneous conservation law, the Runge-Kutta (RK2) formula (2.4) is used as the second order accurate time stepping routine:

$$\begin{aligned} \mathbf{k}_1 &= \Delta t F(\mathbf{U}^n, t^n) \\ \mathbf{k}_2 &= \Delta t F(\mathbf{U}^n + \mathbf{k}_1, t^n + \Delta t) \\ \mathbf{U}^{n+1} &= \mathbf{U}^n + \frac{1}{2}(\mathbf{k}_1 + \mathbf{k}_2) \end{aligned} \quad (2.4)$$

Time integration routines above order two are not used with Strang splitting, since the splitting involves a second order splitting error in time which would cause the accuracy of higher order RK methods to be degraded to second order.

The three time integration methods are stable for $\Delta t \approx O(N^{-2})$ when the collocation points are chosen as the CGL points. A coordinate transformation, or change of variable, is commonly used to lessen the restriction on the stable time step (see 2.7).

2.2 Boundary Conditions

The global nature of spectral methods makes them more sensitive to local treatments, such as boundary conditions. The importance of the treatment of boundary conditions is even more crucial than in local methods such as finite differences or finite elements. Incorrect boundary treatment can create strong instabilities and non-periodic hyperbolic problems present the greatest difficulty. This is in contrast to finite difference methods in which instabilities due

to boundary conditions usually appear as relatively weak oscillations. On the other hand, as opposed to high order finite difference methods, spectral methods do not require numerical boundary conditions in addition to physical boundary conditions required by the PDE, as the spectral stencil does not extend beyond the computational domain. Inflow-outflow boundaries need to be applied via characteristic variables. For details see [7].

2.3 Coordinate Transformations

A change of variable is an important tool in the application of spectral methods. A coordinate transformation may be necessary either to map a computational interval to $[a, b]$ from the interval $[-1, 1]$ or to redistribute the collocation points within an interval for the purpose of giving high resolution to regions of very rapid change. There is a great deal of computational evidence that an appropriately chosen mapping can significantly enhance the accuracy (see, [4], [2], [58], [3]) of spectral approximations, particularly those involving regions of localized rapid variation. Additionally, for large N , the stable time step restriction for explicit time stepping algorithms is very severe for Chebyshev spectral methods. Coordinate transformations can greatly reduce this restriction. The mappings that have been used in the numerical examples which follow are described below. The proper choice of a coordinate mapping will be seen to be an important factor in the ability of a spectral method to resolve a solution with a minimum number of degrees of freedom.

The Kosloff/Tal-Ezer (KT) map is the most widely used map in conjunction with Chebyshev spectral methods due to its ability to produce a nearly uniform grid and its ability to allow for a stable explicit time step many times larger than if the standard grid (1.1) were to be used. The properties, both pros and cons, of the KT map are well documented in the literature ([18], [6], [47]). The properties of the final two maps used, referred to here as the Center Map and Tangent Map, seem to be less well known.

Let ξ denote the original variable and $x = g(\xi)$ the new variable. Then differentiation of a function $f(x)$ is accomplished by making use of the chain rule,

$$\frac{df}{dx} = \frac{d\xi}{dx} \frac{df}{d\xi} = \frac{1}{g'(\xi)} \frac{df}{d\xi}. \quad (2.5)$$

As a result, equation (1) becomes

$$u_t + \left[\frac{1}{g'(\xi)} \right] f(u)_x = b(u) \quad (2.6)$$

after a change of variable has been made.

In implementing the spectral method using the coordinate mapping, the first order differentiation matrix D (1.13) is replaced by the matrix $\check{D} = AD$, where A is a diagonal matrix with entries $A_{ii} = 1/g'(\xi)$.

2.3.1 Interval [-1,1] to interval [a,b]

The standard computational interval for Chebyshev Collocation methods is $[-1, 1]$. Computations on other domains may be achieved via the mapping $x(\xi) = \epsilon\xi + \delta$, where $\epsilon = (b - a)/2$ and $\delta = (b + a)/2$, which maps $\xi \in [-1, 1]$ to the interval $[a, b]$.

2.3.2 Kosloff/Tal-Ezer Mapping

In order to relax the $O(N^{-2})$ time-stepping restriction that is present when advancing Chebyshev methods with explicit time-stepping algorithms using the standard grid (1.1), the use of the grid mapping proposed in [42] has become standard practice. The mapping is

$$x = g(\xi, \gamma) = \frac{\arcsin(\gamma\xi)}{\arcsin(\gamma)} \quad (2.7)$$

where $\xi_j = \cos(\pi j/N)$ are the original CGL collocation points and x_j is a new set of interpolation points depending on the parameter $\gamma \in (0, 1)$. As γ approaches one, the grid points become nearly evenly spaced. As γ approaches zero, the CGL grid is approached. It was shown in [42] that if γ is chosen as

$$\gamma = \gamma(N, \varepsilon) = \operatorname{sech}\left(\frac{|\ln \varepsilon|}{N}\right) \quad (2.8)$$

then the approximation error is roughly ε . By choosing ε to be machine zero (on a 32-bit machine using double precision floating

N	$\gamma(N, \varepsilon)$
32	0.5867
64	0.8600
128	0.9602
256	0.9901
512	0.9975

Table 2.1: map (2.7) parameter ranges

point operations, $\varepsilon = 2^{-52}$), it is argued in [42] that the error of the coordinate transformation is essentially guaranteed to be small. However, even though spectral accuracy may theoretically be sacrificed, good results may be obtained by violating the parameter limits listed in table 2.1.

A result of using the mapping is that the spectral radius of the mapped differentiation matrix is asymptotically $O(N)$ as opposed to $O(N^2)$ for the unmapped case. Thus, the implementation of the mapping (2.7) can change the necessary condition for stability from $\Delta t \propto N^{-2}$ to $\Delta t \propto N^{-1}$ in the explicit Runge-Kutta methods described in section (2.1).

It was pointed out in [6] that the argument presented in [42], that the KT map allows the Chebyshev collocation method to maintain spectral accuracy, is flawed. Under certain conditions, even if γ is chosen as (2.8), using the KT map sacrifices spectral accuracy. In [47] and [39], two ways are suggested for choosing γ if spectral accuracy is to be maintained. However, these arguments were seemingly made under the assumption that the solution being approximated was smooth and could be resolved with a small number of basis functions, e.g. $N = 32$, and not considering solutions with rapid variations or shocks.

2.3.3 Center Map

The center map [3] is

$$x = g(\xi, \gamma) = (1.0 - \gamma)\xi^3 + \gamma\xi \quad (2.9)$$

with $\gamma \in (0, 1)$. Smaller values of γ cluster grid points around the center of the computational interval while still maintaining a dense grid point density near boundary points. As $\gamma \rightarrow 1$ the grid

approaches the Chebyshev grid. The map can be used to resolve regions of rapid variation in the center of a computational domain.

2.3.4 Tangent Map

The tangent map [4] is

$$x = g(\xi, \gamma, \beta) = x_0 + \frac{\tan(\delta\xi + \omega)}{\gamma}, \quad (2.10)$$

where $\kappa = \arctan(\gamma(1 - \beta))$, $\mu = \arctan(\gamma(1 + \beta))$, $\delta = 0.5(\kappa + \mu)$, $\omega = 0.5(\kappa - \mu)$, and $x_0 = -1 + 2(\beta - a)/(b - a)$. The map can be used to resolve solutions with either a region of rapid variation in the interior or at boundaries. The map can be used to cluster grid points around the point β in the interval $[a, b]$. The parameter $\gamma > 0$ determines the degree to which the clustering takes place.

Figure 2.1 shows grids which result from using the above maps.

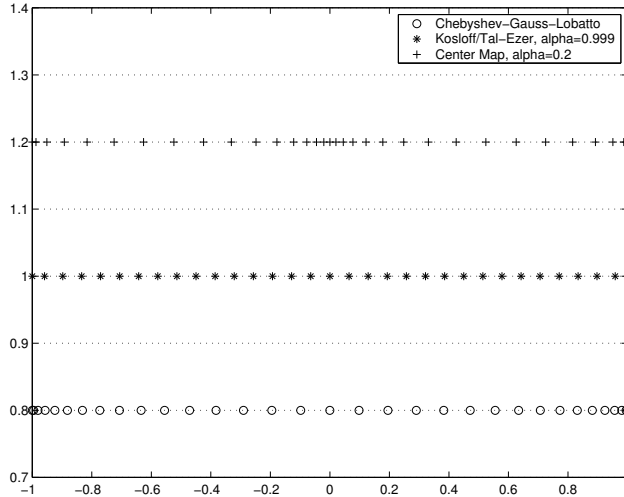


Figure 2.1: grid point distribution

2.4 Problem Formulation

The theoretical results for the spectral viscosity methods are for homogeneous systems of conservation laws. To extend the methods to systems of conservation laws with source terms, operator splitting

can be used to separate the system into two parts, consisting of a homogeneous system of conservation laws and a system of ordinary differential equations. However, from our experience, the spectral viscosity method can be implemented successfully in both an un-split formulation and a split formulation with handles the source term separately. In two dimensions, dimensional splitting may be used to reduce the two dimensional problem to a sequence of one-dimensional problems.

The splitting method used was Strang splitting [59], which allows for second order accuracy in time. The second order accurate Runge-Kutta formula (2.4) was used as the time integration method with the split formulations. The split formulation of the problem is especially convenient when the split steps involving the source term can be evaluated exactly, as is the case in the fluidized bed examples.

By implementing the un-split formulation of the problem, accuracy of order greater than two can be achieved in time, at the expense of a higher operation count. In one dimensional problems, we have used the fourth order Runge-Kutta method (2.2) with the un-split formulation. In two dimensions, a low storage, third order Runge-Kutta (2.3) method has been used.

For a suitably chosen time step, the results of the different problem formulations did not vary significantly in the numerical examples. However, slightly less spectral viscosity was necessary to obtain a stable approximation with the split formulation than with the unsplit formulation. It is speculated that the incremental way in which the spectral viscosity is applied in the split formulation makes this possible.

2.4.1 One-dimensional unsplit

The unsplit formulation of the one-dimensional spectral viscosity approximation (see section 4) of problem (1) is

$$w_t + f(w)_x = b(w) + SSV(\beta, C, N). \quad (2.11)$$

This formulation is implemented as

$$\begin{aligned} w_t + f(w)_x &= b(w) \\ w_t &= SSV(\beta, C, N). \end{aligned} \quad (2.12)$$

The conservation law is advanced in time with a Runge-Kutta method and the SSV equation is evaluated exactly as in (4.5).

2.4.2 One-dimensional Strang split

The Strang split formulation of the one-dimensional spectral viscosity method is

$$w_t = b(w) \quad (2.13)$$

$$w_t + f(w)_x = 0 \quad (2.14)$$

$$w_t = SSV(s, C, N) \quad (2.15)$$

$$w_t = b(w). \quad (2.16)$$

For $t > 0$, the starting value for each equation (2.13) to (2.16) is provided by the solution of the previous equation. Equations (2.14) and (2.15) are solved with a time step Δt while equations (2.13) and (2.16) are solved with a time step of $\Delta t/2$. The second order Runge-Kutta formula (2.4) is used to advance (2.14) in time while the SSV equation is solved exactly as in (4.5). In some cases, it may be possible to evaluate the fractional step involving the source term exactly. Otherwise, it may be advanced with the RK2 formula.

A typical implementation of Strang splitting would have evaluated (2.15) over a time step of size $\Delta t/2$ before and after (2.14). However, the SSV term is just a filter of the same strength and order (assuming a constant Δt) being applied at every time level and the exact location at which it is applied in time is irrelevant and the splitting maintains second order accuracy in time.

2.4.3 Two-dimensional unsplit

The two-dimensional un-split formation is implemented as

$$w_t + f(w)_x + g(w)_y = b(w) \quad (2.17)$$

$$w_t = SSV(\beta, C, N). \quad (2.18)$$

If a third order, three stage Runge-Kutta formula is used for the time stepping, $g(w)_y$ is evaluated 3 times, $f(w)_x$ 3 times, SSV_x is applied once, and SSV_y is applied once, per time step.

2.4.4 Two-dimensional Strang split

$$w_t = b(w) \quad (2.19)$$

$$w_t + g(w)_y = 0 \quad (2.20)$$

$$w_t = SSV(\beta, C, N)_y \quad (2.21)$$

$$w_t + f(w)_x = 0 \quad (2.22)$$

$$w_t = SSV(\beta, C, N)_x \quad (2.23)$$

$$w_t + g(w)_y = 0 \quad (2.24)$$

$$w_t = SSV(\beta, C, N)_y \quad (2.25)$$

$$w_t = b(w) \quad (2.26)$$

Equation (2.22) and (2.23) are solved over a full time step while the other 6 equations are evaluated over a time step of size $\Delta t/2$. The fractional steps involving the source terms, (2.19) and (2.26), may possibly be evaluated in closed form. Otherwise, they may be advanced in time with the Runge-Kutta formula. The SSV split steps (2.21), (2.25), and (2.23) can be evaluated exactly as in (4.5). The remaining equations are advanced in time with the second order Runge-Kutta method (2.4). In this formulation of the problem, $g(w)_y$ is evaluated 4 times, $f(w)_x$ 2 times, the source term 2 times, SSV_x is applied once, and SSV_y is applied twice, per time step.

Part II

Spectral Filters and Spectral Viscosity

The use of unaltered spectral methods on constant coefficient linear hyperbolic problems with discontinuous solutions produces oscillatory numerical results. While stable, these solutions exhibit Gibbs-like oscillations arising directly from the discontinuities. Variable coefficient and nonlinear hyperbolic problems have a mechanism for spreading the discontinuity-induced high frequency oscillations over the entire frequency spectrum as the solution evolves. Some additional dissipative or filtering mechanism may be needed to stabilize the calculation. The spectral approximations to nonlinear hyperbolic conservation laws are known not to converge to the entropy solution [61]. Modifications to the standard spectral method must be made to insure stability and to insure convergence to entropy solutions. Postprocessing methods must be used to remove Gibbs oscillations and recover spectral accuracy.

Chapter 3

Spectral Filters

Before examining spectral vanishing viscosity methods, it is necessary to discuss spectral filters. Spectral space filters can be used to enhance the decay rate of the Chebyshev coefficients without reducing the accuracy. The following is a review of results which first appeared in [63]. The filtered Chebyshev approximation is defined as

$$F_N u(x) = \sum_{n=0}^N \sigma\left(\frac{n}{N}\right) a_n T_n(x) \quad (3.1)$$

where σ is a p^{th} order ($p > 1$) filter function defined as a $C^\infty[-1, 1]$ satisfying

$$\begin{aligned} \sigma(0) &= 1 \\ \sigma(\pm 1) &= 0 \\ \sigma^{(j)}(0) &= 0 \quad j \leq p \\ \sigma^{(j)}(\pm 1) &= 0 \quad j \leq p. \end{aligned} \quad (3.2)$$

The filtering can be implemented through the Fast Cosine algorithm or as a matrix multiplication with a filter matrix, F , containing elements defined as

$$F_{ij} = \frac{2}{N c_j} \sum_{n=0}^N \frac{1}{c_n} \sigma\left(\frac{n}{N}\right) T_n(x_i) T_n(x_j).$$

The convergence rate of the filtered approximation is determined solely by the order of the filter, σ , and the regularity of the func-

tion, $u(x)$, away from the point of discontinuity. In particular, if the function, $u(x)$, is piecewise analytic and the order of the filter increases with N , one recovers an exponentially accurate approximation to the unfiltered function everywhere except very close to the discontinuity.

Many different filter functions are available, but perhaps the most versatile and widely used filter is the exponential filter

$$\sigma(\omega) = \exp(-\alpha|\omega|^\beta). \quad (3.3)$$

The parameter α is taken $\alpha = -\ln \varepsilon$ where ε is defined as machine zero. The parameter β is the order of the filter. The order of the filter is changed simply by specifying a different β . Formally, (3.3) does not satisfy all the conditions necessary for it to be a p^{th} order filter, as $\sigma(1) = \exp(-\alpha)$. However, by specifying that $\alpha = -\ln \varepsilon$ we get that $\sigma(1) = \exp(\ln \varepsilon) = \varepsilon$ so that $\sigma(1) = 0$ as far as a computer is concerned. On a 32-bit machine using double precision floating point operations, $\varepsilon = 2^{-52}$ and $\ln(\varepsilon) \simeq -36.0437$. A smaller value of α indicates a weaker filter as fewer modes are altered and less damping is done on the modes that are altered.

Figure (3.1) compares the exponential filters with three different settings of the parameters α and β . For $\alpha = 0.032$ and $\beta = 4$, $\sigma(1) \neq 0$ and the filter is not a 4th order filter as defined above. However, this is a typical configuration of the filter when it is used to implement the Super Spectral Viscosity method (see section 4) which is used to stabilize spectral approximations to conservation laws. Compared to the strength of an exponential filter, the damping of high order modes is nearly non-existent in the Super Spectral Viscosity method. This type of filtering is what allows the spectral method to retain a high order accuracy in the presence of shocks.

While the use of filters can have a dramatic impact on the quality of the convergence of a global approximation to a discontinuous function, filters fail to improve the quality of the approximation in the neighborhoods of discontinuities and fail to alleviate the non-linear mixing of modes (aliasing). Even more disturbing, filtering generally treats the Gibbs oscillations as a source of noise and attempts to remove them. It has been speculated in the past [43], and recently shown rigorously [36], that the Gibbs oscillations are not noise. Rather, they contain sufficient information from which to extract a global exponentially convergent approximation. The proper

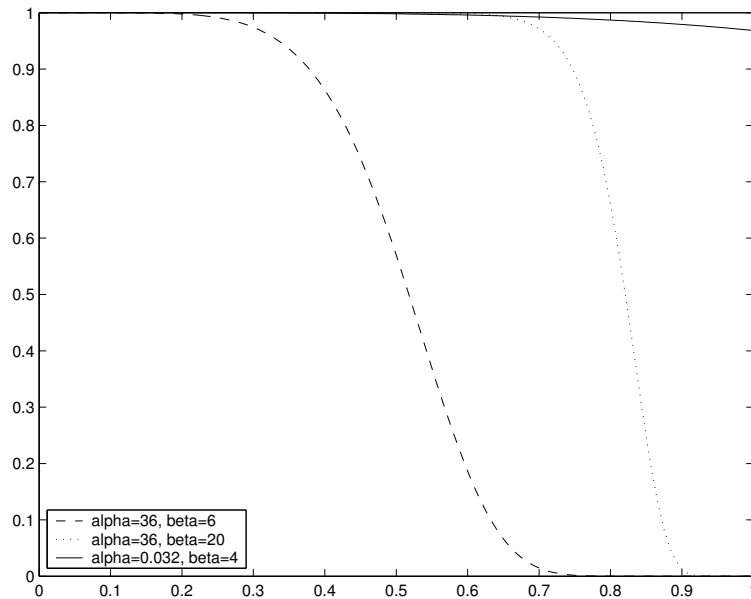


Figure 3.1: exponential filter comparison

use of filtering in the setting of spectral methods is as a stabilization mechanism rather than as a post-processing or reconstruction tool. Exponential filtering has been used, for example in [16], to stabilize spectral approximations. However, the relatively strong filtering applied by a spectral exponential filter provides more damping than is necessary for stability and for guaranteeing convergence to entropy solutions. The use of a strong spectral exponential filter may require the use of a much larger N than necessary to get a desired accuracy.

Chapter 4

Super Spectral Viscosity

When spectral methods are applied to nonlinear hyperbolic equations in conservation form, the problem of obtaining an entropy satisfying solution arises. Unmodified spectral methods do not converge to the correct entropy solution if the solution contains shocks [61]. In fact, there is no artificial dissipation inherent in the method to indicate that it is a limit of a dissipative process. It is clear that some artificial dissipation needs to be added. This added dissipation should be as small as possible, just enough to stabilize the solution and to ensure convergence to the unique entropy satisfying solution, in order not to affect the overall spectral accuracy of the method.

Spectral calculations have been successfully stabilized simply by the application of the exponential filter (3.3) on the conserved variables and/or the flux derivatives at each time step ([16], [17]). However, the straight application of an exponential filter can dampen the high modes more than is necessary for stability, and in the process sacrifice accuracy. A more sophisticated approach is a spectral viscosity (SV) method. A SV method ([61], [10]) applies a second order spectral viscosity, the size of which is kept small by a parameter ε , to all modes $k > k_N$, where k_N is a threshold frequency such that $1 \leq k_N \leq N$. By letting k_N and ε vary with N , spectral accuracy can be maintained. An application of a Chebyshev SV method is given in [1]. A SV method converges to the correct entropy satisfying solution while maintaining spectral accuracy provided that the solution is uniformly bounded, i.e. $|u^N|_{L^\infty} \leq C$ where C is a constant. An improved SV method is the Super Spectral Vis-

N	C	s
32	5.6	3.47
64	8	4.16
128	11.3	4.85
256	16	5.55
512	22.6	6.24
1024	32	6.93

Table 4.1: SSV parameter limits

cosity (SSV) method [45] which guarantees the convergence of the bounded solution of the Chebyshev collocation method while applying less viscosity, and thus preserving more accuracy. The SSV formulation, which is the stabilization method used in our work, will be examined below. The theoretical work justifying the use to the method is complete, but applications of the method have not previously appeared in the literature.

To stabilize the numerical scheme, the PDE is regularized by an additional term,

$$\frac{\partial}{\partial t} u_N + \frac{\partial}{\partial x} f(u_N) = \varepsilon (-1)^{s+1} Q^{2s} u_N = SSV(s, C, N) \quad (4.1)$$

where

$$Q = \sqrt{1-x^2} \frac{\partial}{\partial x} \quad (4.2)$$

is a viscosity operator (Chebyshev Differential Operator). The parameter ε is defined as $\varepsilon = CN^{1-2s}$ where C is a constant chosen large enough to ensure stability and such that $0 \leq C \leq N^{1/2}$. The parameter s may grow with N and should be chosen such that $s \leq \ln(N)$. The parameter s is problem dependent, with its choice being influenced by the strength of the shocks involved. For such s and C , the bounded solutions of (4.1) converge to the correct entropy solution. Table 4.1 lists maximum parameter values for commonly used N .

Unlike the SV method in which the viscosity is only applied to high frequencies, the filtering is applied to all modes (although in reality the viscosity is so weak that the lower modes are not altered within machine precision) and is weaker with its application being

more gradual. It can be seen that $\varepsilon \downarrow 0$ as $N \rightarrow \infty$, thus the term vanishing viscosity is justified.

A direct implementation of (4.1) amounts to adding $2s$ spatial derivatives to the equation. This would introduce additional stiffness which would severely limit the stable time step and increase the computational work involved by requiring the computation of higher order derivatives. Hence, the practical implementation of the SSV method is an important issue. In order to derive an efficient implementation of the SSV method, it is necessary to first examine the viscosity operator Q^2 applied to the Chebyshev polynomial (1.2), $T_k(x)$.

$$\begin{aligned}
Q^2 T_k(x) &= \sqrt{1-x^2} \frac{\partial}{\partial x} \left[\sqrt{1-x^2} \frac{\partial}{\partial x} T_k(x) \right] \\
&= \sqrt{1-x^2} \frac{\partial}{\partial x} \left[\sqrt{1-x^2} \frac{k \sin(k \arccos(x))}{\sqrt{1-x^2}} \right] \\
&= \sqrt{1-x^2} \left[-\frac{k^2 \cos(k \arccos(x))}{\sqrt{1-x^2}} \right] \\
&= -k^2 T_k(x).
\end{aligned}$$

As a result of applying the viscosity operator to the Chebyshev polynomials, it can be noticed that the Chebyshev polynomials are the eigenfunctions of the operator Q^2 with eigenvalues k^2 . Expanding the viscosity term, which is the right side of (4.1), we notice that

$$\begin{aligned}
\varepsilon(-1)^{s+1} Q^{2s} u_N &= \varepsilon(-1)^{s+1} Q^{2s} \sum_{k=0}^N a_k(t) T_k(x) \\
&= -C N^{1-2s} \sum_{k=0}^N k^{2s} a_k(t) T_k(x) \\
&= \frac{-CN}{N^{2s}} \sum_{k=0}^N k^{2s} a_k(t) T_k(x) \\
&= -CN \sum_{k=0}^N \left(\frac{k}{N}\right)^{2s} a_k(t) T_k(x).
\end{aligned}$$

If we implement the SSV method via time splitting where in the first step we solve

$$\frac{\partial}{\partial t} u_N + \frac{\partial}{\partial x} f(u_N) = 0 \quad (4.3)$$

and in the second step we solve

$$\frac{\partial}{\partial t} u_N = \varepsilon(-1)^{s+1} Q^{2s} u_N, \quad (4.4)$$

the second equation, (4.4), in the split step can be written as

$$\frac{\partial}{\partial t} \left[\sum_{k=0}^N a_k(t) T_k(x) \right] = -CN \sum_{k=0}^N \left(\frac{k}{N} \right)^{2s} a_k(t) T_k(x).$$

For every k we have the ODE

$$\frac{d}{dt} [a_k(t)] = -CN \left(\frac{k}{N} \right)^{2s} a_k(t)$$

which can be solved analytically over one time step as

$$a_k(t + \Delta t) = a_k(t) \exp(-CN \Delta t (k/N)^{2s}).$$

Thus, the exact solution of the SSV split step can be written as the filtered partial sum

$$u_N(x) = \sum_{k=0}^N \sigma \left(\frac{k}{N} \right) a_k(t) T_k(x) \quad (4.5)$$

and applying the Chebyshev SSV method is seen to be equivalent to applying the exponential filter (3.3) with $\beta = 2s$, $\alpha = CN \Delta t$. The method can be implemented with little additional cost. It should be stressed that while the SSV is being implemented via the exponential filtering framework, that it is not a β^{th} order filter as defined in (3.2). The amount of damping of the high modes is far smaller with the SSV method than with the application of a β^{th} order exponential filter (see figure 3.1).

After the spectral vanishing viscosity has been added in order to stabilize the computation, the numerical solution may still have strong oscillations near shock fronts and other discontinuities. A postprocessing step which is able to recover spectral accuracy can

now be applied. It should be emphasized that while the spectral viscosity is applied at each time step, that it is only necessary to apply a postprocessing method at the time levels in which a "clean" solution is desired.

4.1 Spatially varying viscosity

The presence of $\sqrt{1-x^2}$ in the viscosity term (4.2) causes the filtering to be spatially varying. It provides less stabilization near the boundaries and is zero at the boundary of the domain. In computations with moving discontinuities or with discontinuities located close to boundaries, this may be a source of problems. The spatially varying viscosity seems to be the major obstacle in implementing a domain decomposition or spectral element method as it is difficult to pass strong shocks through element boundaries.

However, if the viscosity operator did not vanish at the boundaries, it would introduce spurious boundary layers and require additional non-physical boundary conditions [5].

Part III

Postprocessing

The truncation error decays exponentially as N increases when spectral methods are used to approximate smooth functions. However, the situation changes when the function is discontinuous as the spectral approximation no longer converges in the maximum norm. This is known as the Gibbs phenomenon. In the figure 4.1, a step function is approximated by a Chebyshev partial sum (1.3) and the approximation is degraded by the oscillations which characterize the Gibbs phenomenon.

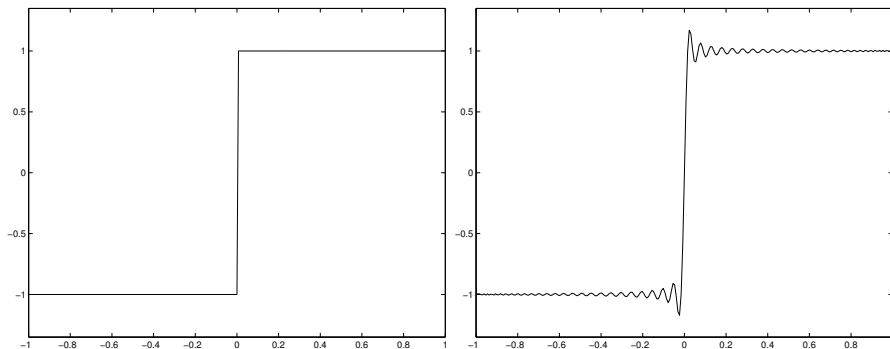


Figure 4.1: step function and Chebyshev approximation

Several methods exist for postprocessing Gibbs polluted spectral approximations (see Appendix C). Most however, such as spectral mollification [30, 62], only recover spectral accuracy up to within a neighborhood of each discontinuity.

To date, the most powerful processing method seems to be the Gegenbauer Reconstruction Procedure (GRP) which is capable of recovering spectral accuracy up to and including at the location of discontinuities. Although the GRP has been shown to produce remarkable results on some simple problem, the method lacks robustness due to the fact that two parameters, for which an optimal choice for is currently not known, must be specified. In this work, the GRP is the only postprocessing technique used. One of our goals is to examine if the GRP can be used to successfully postprocess more complex PDE solutions.

The GRP [32, 36, 35, 33, 34] was developed for the purpose of recovering exponential accuracy at all points, including at the discontinuities themselves, from the knowledge of a spectral partial sum of a discontinuous, but piecewise analytic function. While the SSV solution serves as a highly accurate approximation to the

exact spectral partial sum, only partial theoretical justification can be found concerning using the GRP as a postprocessing method for the SSV solution. However, numerical results indicate that spectral accuracy can be achieved by applying the GRP to the SSV solution of homogeneous systems of conservation laws [23, 25]. The same can be said about the edge detection method described in section 5, as the theoretical results are limited to locating the jump discontinuities of a piecewise smooth function $u(x)$. However, numerical evidence also advocates applying the edge detection method to the SSV solution.

Chapter 5

Edge Detection

The Gegenbauer Reconstruction Procedure recovers spectral accuracy up to the discontinuity points in each smooth subinterval of a piecewise analytic function. Thus, the GRP needs the exact location of discontinuities, or edges, in the function. If a PDE solution is being postprocessed and the solution contains rarefaction waves, discontinuities in the first derivative of the function will exist and need to be located as well. The method used to find the edges originated in [26] for periodic and non-periodic functions. The method is specialized to approximations of functions by Chebyshev methods and is summarized below.

Denote the location of discontinuities as α_j . Let

$$[f](x) := f(x^+) - f(x^-)$$

denote a local jump in the function and define

$$ue(x) = \frac{\pi\sqrt{1-x^2}}{N} \sum_{k=0}^N a_k \frac{d}{dx} T_k(x) \quad (5.1)$$

where

$$\frac{d}{dx} T_k(x) = \frac{k \sin(k \arccos(x))}{\sqrt{1-x^2}}.$$

Essentially, we are looking at the derivative of the spectral projection of the numerical solution to determine the location of the discontinuities. The series $ue(x)$ has the convergence properties

$$ue(x) \rightarrow \begin{cases} O\left(\frac{1}{N}\right) & \text{when } x \neq \alpha_j \\ [f](\alpha_j) & \text{when } x = \alpha_j. \end{cases}$$

The series converges to both the height and direction of the jump at the location of a discontinuity. However, for the GRP, we only need the magnitude of the jumps. While a graphical examination of the series $ue(x)$ verifies that the series does have the desired convergence properties, an additional step is need to numerically pinpoint the location of the discontinuities. For that purpose, make a non-linear enhancement to the edge series as

$$un(x) = N^{\frac{Q}{2}} [ue(x)]^Q \quad (5.2)$$

The values, $un(x)$, will serve to amplify the separation of scales which has taken place in (5.1). The series has the convergence properties

$$un(x) \rightarrow \begin{cases} O\left(N^{-\frac{Q}{2}}\right) & \text{when } x \neq \alpha_j \\ N^{\frac{Q}{2}} [[f](\alpha_j)]^Q & \text{when } x = \alpha_j. \end{cases}$$

By choosing $Q > 1$ we enhance the separation between the $O\left(\left[\frac{1}{N}\right]^{\frac{Q}{2}}\right)$ points of smoothness and the $O\left(N^{\frac{Q}{2}}\right)$ points of discontinuity. The parameter J , whose value will be problem dependent, is a critical threshold value. Finally, redefine $ue(x)$ as

$$ue(x) = \begin{cases} ue(x) & \text{if } un(x) > J \\ 0 & \text{otherwise.} \end{cases}$$

With Q large enough, one ends up with and edge detector $ue(x) = 0$ at all x except at the discontinuities $x = \alpha_j$. Only those edges with amplitude larger than $J^{1/Q} \sqrt{1/N}$ will be detected.

Often the series ue is slow to converge in the area of a discontinuity and the nonlinear enhancement has difficulty pinpointing the exact location of the edge. If an additional parameter, η , is added to the procedure this problem can be overcome in a simple manner. The parameter specifies that only one edge may be located in the interval $(x[i - \eta], x[i + \eta])$, $i = 0, \dots, N$, with appropriate one sided intervals being considered near boundaries. The correct edge will be the maximum of ue in this subinterval. The value of η is problem

dependent and is best chosen after the edge detection procedure has been applied once.

The edge detection parameters J , Q , and η , are all problem dependent. Various combinations of the parameters may be used to successfully locate edges represented by jumps of a magnitude in a certain range.

5.1 Edge detection in the first derivative

If rarefaction waves are present, the first derivative of the solution will also have discontinuities. In this case, the edge detection procedure will have to be used to examine the first derivative of the solution in each piecewise smooth subinterval. After the shock locations are determined, u_N is differentiated in each C^0 smooth interval. Then the shock locations and the rarefaction locations are arranged in increasing order.

5.2 Two Dimensions

Edge detection may be applied in two dimensions by fixing the variable in one dimension and detecting the edges in the other dimension as a function of the fixed variable. Thus, the discontinuities of the function $f(x, y)$ may be found for each fixed \bar{x} and \bar{y} by considering

$$\frac{\pi\sqrt{1-x(\bar{y})^2}}{N} \sum_{k=0}^N \sum_{l=0}^N a_{k,l} \frac{d}{dx} T_k(x(\bar{y})) T_l(\bar{y}) \rightarrow [f](x(\bar{y})) \quad (5.3)$$

to find the edges in the x direction. Likewise, to find the edges in the y direction consider

$$\frac{\pi\sqrt{1-y(\bar{x})^2}}{N} \sum_{k=0}^N \sum_{l=0}^N a_{k,l} T_k(\bar{x}) \frac{d}{dy} T_l(y(\bar{x})) \rightarrow [f](y(\bar{x})). \quad (5.4)$$

Chapter 6

Gegenbauer Reconstruction Procedure

6.1 Gegenbauer Reconstruction

The Gegenbauer Reconstruction procedure works by expanding the function in another basis, the *Gibbs complementary basis*, via knowledge of the known Chebyshev coefficients and the location of discontinuities. The Chebyshev partial sums are projected onto a space spanned by the Gegenbauer polynomials. The associated weight functions increasingly emphasize information away from the discontinuities as the number of included modes grow. The approximation converges exponentially in the new basis even though it only converged very slowly in the original basis due to the Gibbs phenomenon. The choice of a Gibbs complementary basis is the Ultraspherical or Gegenbauer polynomials, C_l^λ (See Appendix B). Whether the Gegenbauer basis is the optimal choice as the Gibbs complementary basis for the Chebyshev basis remains an open question. It is shown in [36] that the Gegenbauer basis is a Gibbs complementary basis for the Chebyshev basis.

If a Gibbs complementary basis exists, it is possible to construct a piecewise exponentially convergent approximation to a piecewise analytic function from the information contained in the original very slowly converging global approximation. In order for a Gibbs complementary basis, $\psi_n^\lambda(\xi)$, to exist we need knowledge of the intervals $[a, b] \subset [-1, 1]$ in which $f(x)$ is analytic and we need that the original spectral approximation is pointwise convergent in all of $[-1, 1]$

except for at a finite number of points. A local variable ξ is defined as $\xi(x) = -1.0 + 2(\frac{x-a}{b-a})$ such that $\xi : [a, b] \rightarrow [-1, 1]$. In order for $\psi_n^\lambda(\xi)$ to be a Gibbs complementary basis we require that:

- $\psi_n^\lambda(\xi)$ is orthogonal in the weighted inner product, $(\cdot, \cdot)_w^\lambda$, where the weight $w(x)$ may depend on λ

$$(\psi_k^\lambda, \psi_n^\lambda)_w^\lambda = \|\psi_n^\lambda\|_{L_w^2[-1,1]}^2 \delta_{kn} = \gamma_n^\lambda \delta_{kn}$$

- $P_\lambda f(\xi) = \sum_{n=0}^\lambda \frac{1}{\gamma_n^\lambda} (f, \psi_n^\lambda)_w^\lambda \psi_n^\lambda(\xi)$ is pointwise exponentially convergent as λ increases

$$\|f - P_\lambda f\|_{L_\infty[-1,1]} \leq C e^{-c\lambda} \quad c > 0$$

- $\psi_n^\lambda(\xi)$ satisfies the *Gibbs condition*, i.e. there exists $\beta < 1$ such that for $\lambda = \beta N$ we have

$$\left| \frac{1}{\gamma_n^\lambda} (T_k(x(\xi)), \psi_n^\lambda(\xi))_w^\lambda \right| \|\psi_n^\lambda\|_{L_\infty[-1,1]} \leq \left(\frac{\alpha N}{k}\right)^\lambda$$

for $k > N$, $n \leq \lambda$, and $\alpha < 1$. The Gibbs condition guarantees that the projection of the high modes of T_k onto the basis ψ_n^λ is exponentially small.

The Gegenbauer expansion of a function $f(x)$, $x \in [-1, 1]$ is defined as

$$f(x) = \sum_{l=0}^{\infty} \widehat{f}_l^\lambda C_l^\lambda(x)$$

where \widehat{f}_l^λ are the Gegenbauer coefficients of $f(x)$ defined as

$$\widehat{f}_l^\lambda = \frac{1}{\widehat{h}_l^\lambda} \int_{-1}^1 (1-x^2)^{\lambda-1/2} C_l^\lambda(x) f(x) dx, \quad (6.1)$$

with

$$\widehat{h}_l^\lambda = \sqrt{\pi} C_l^\lambda(1) \frac{\Gamma(\lambda + 1/2)}{\Gamma(\lambda) \Gamma(l + \lambda)}.$$

Since we don't know the function $f(x)$, implementing the Gegenbauer Reconstruction procedure requires obtaining an exponentially

accurate approximation, \widehat{g}_l^λ , to the first m coefficients \widehat{f}_l^λ in the Gegenbauer expansion from the first $N + 1$ Chebyshev coefficients of $f(x)$. The approximate coefficients are defined as the integral

$$\widehat{g}_l^\lambda = \frac{1}{\widehat{h}_l^\lambda} \int_{-1}^1 (1-x^2)^{\lambda-1/2} C_l^\lambda(x) f_N(x) dx, \quad (6.2)$$

where

$$f_N(x) = \sum_{n=0}^N a_n T_n(x)$$

is a Chebyshev partial sum. The integral should be evaluated by Gauss-Lobatto quadrature in order to insure sufficient accuracy. The coefficients \widehat{g}_l^λ are now used in the partial Gegenbauer sum to approximate the original function as

$$f(x) \approx f_m^\lambda(x) = \sum_{l=0}^m \widehat{g}_l^\lambda C_l^\lambda(x) \quad (6.3)$$

6.1.1 Reconstruction on a subinterval

In practice, there will be discontinuities in the interval $[-1, 1]$ and the reconstruction must be done on each subinterval $[a, b]$ in which the solution remains smooth. To accomplish the reconstruction on each subinterval, define a local variable for each subinterval as $x(\xi) = \epsilon\xi + \delta$ where $\epsilon = (b-a)/2$, $\delta = (b+a)/2$ and $\xi_j = \cos(\pi j/N)$. The reconstruction in each subinterval is then accomplished by

$$f_m^{\lambda,\epsilon}(x) = \sum_{l=0}^m \widehat{g}_\epsilon^\lambda(l) C_l^\lambda(\xi) \quad (6.4)$$

where

$$\widehat{g}_\epsilon^\lambda(l) = \frac{1}{\widehat{h}_l^\lambda} \int_{-1}^1 (1-\xi^2)^{\lambda-1/2} C_l^\lambda(\xi) f_N(\epsilon\xi + \delta) d\xi \quad (6.5)$$

Notice that we have used collocation points on the entire interval $[-1, 1]$ to build the approximation in $[a, b]$. This is referred to as a *global-local* approach [34]. The global-local approach seems to be best when postprocessing PDE solutions where f_N is obtained from the time evolution. The point values $f(x_i)$ may not be accurate, but the global interpolating polynomial $f_N(x)$ is accurate. The

alternative is a *local-local* approach, in which only the collocation points within $[a, b]$, with suitable scaling to expand $[a, b]$ to $[-1, 1]$, are used.

6.1.2 Truncation and Regularization Errors

Two types of errors occur in the reconstruction process. The error between the exact Gegenbauer coefficients and the approximate coefficients obtained from the Chebyshev coefficients is called the *truncation error* (6.6),

$$TE(\lambda, m, N, \epsilon) = \max_{-1 \leq \xi \leq 1} \left| \sum_{l=0}^{\lambda} \left(\hat{f}_{\epsilon}^{\lambda}(l) - \hat{g}_{\epsilon}^{\lambda}(l) \right) C_l^{\lambda}(\xi) \right|. \quad (6.6)$$

The error between the Gegenbauer expansion of $f(x)$ and its approximated partial sum is known as the *regularization error*,

$$RE(\lambda, m, \epsilon) = \max_{-1 \leq \xi \leq 1} \left| f(\epsilon\xi + \delta) - \sum_{l=0}^m \hat{f}_{\epsilon}^{\lambda}(l) C_l^{\lambda}(\xi) \right|. \quad (6.7)$$

If there are more terms in the series to approximate (i.e., larger m) the truncation error will increase, but the regularization error will be smaller. In practice, m needs to be chosen to maintain some balance between the two errors.

6.1.3 Convergence Theorem

Here we will basically quote the main theorems, without proofs, from [33] except that we have specialized it to use a Chebyshev partial sum rather than a more general Gegenbauer partial sum. The theorems guarantee that an exponentially accurate approximation may be obtained at all points (including at the discontinuities themselves), from the knowledge of a spectral partial sum of a discontinuous but piecewise analytic function. Thus, the same order of accuracy as in the smooth case can be recovered.

The assumption that the function we are considering is analytic is used in the regularization error theorem. The assumption allows us to assume that there exists constants $\rho \geq 1$ and $C(\rho)$ such that, for every $k \geq 0$,

$$\max_{a \leq x \leq b} \left| \frac{d^k f}{dx^k}(x) \right| \leq C(\rho) \frac{k!}{p^k} \quad (6.8)$$

If $f(x)$ is not analytic, but only C^k , we would only be able to get an $O(m^{-k})$ estimate for the regularization error.

Theorem 1 (*Truncation Error*) *Let the truncation error be defined as in (6.6). Let $\lambda = \alpha\epsilon N$ and $m = \beta\epsilon N$ with $0 < \alpha, \beta < 1$, then*

$$TE(\lambda, m, N, \epsilon) < A \left(\frac{(\beta + 2\alpha)^{\beta+2\alpha}}{2^\alpha \alpha^\alpha \beta^\beta} \right)^{\epsilon N}$$

where A grows at most as N . In particular, if $\alpha = \beta < 2/27$, then

$$TE(\lambda, m, N, \epsilon) < Aq^{\epsilon N}$$

where

$$q = \left(\frac{27\alpha}{2} \right)^\alpha < 1.$$

Theorem 2 (*Regularization Error*) *Assume $\lambda = \gamma m$ where γ is a positive constant. If $f(x)$ is analytic in $[a, b] \subset [-1, 1]$ satisfying assumption (6.8), then the regularization error (6.7) can be bounded by*

$$RE(\lambda, m, \epsilon) \leq Aq^m$$

where q is given by

$$q = \frac{\epsilon(1+2\gamma)^{1+2\gamma}}{\rho 2^{1+2\gamma} \gamma^\gamma (1+\gamma)^{1+\gamma}}$$

which is always less than 1. In particular, if $\gamma = 1$ and $m = \beta\epsilon N$ where β is a positive constant, then

$$RE(\lambda, m, \epsilon) \leq Aq^{\epsilon N}$$

with

$$q = \left(\frac{27\epsilon}{32p} \right)^\beta.$$

By combining the previous two theorems, we can state:

Theorem 3 (*Removal of the Gibbs Phenomenon*) Consider a L_1 function $f(x)$ on $[-1, 1]$, which is analytic in a sub-interval $[a, b] \subset [-1, 1]$ and satisfies assumption 6.8. Assume that the first $N + 1$ Gegenbauer coefficients, $\widehat{f}_\epsilon^\lambda$, defined in (6.1) are known. Let $\widehat{g}_\epsilon^\lambda$, $0 \leq l \leq m$, be the approximate Gegenbauer expansion coefficients, defined in (6.2), based on the subinterval $[a, b]$, of the Gegenbauer partial sum f_ϵ^λ in (6.4). Then for $\lambda = m = \beta\epsilon N$ with $\beta < 2/27$, we have

$$\max_{-1 \leq \xi \leq 1} \left| f(\epsilon\xi + \delta) - \sum_{l=0}^m \widehat{g}_\epsilon^\lambda(l) C_l^\lambda(\xi) \right| \leq A (q_T^{\epsilon N} + q_R^{\epsilon N})$$

where

$$q_T = \left(\frac{27\beta}{2} \right)^\beta < 1, \quad q_R = \left(\frac{27\epsilon}{32p} \right)^\beta < 1$$

and A grows at most as a fixed-degree polynomial of N .

The above results from [33] apply in the case of Galerkin approximations. In [34], a nearly identical result is stated for collocation approximations. However, the authors were unable to show that the truncation error is exponentially small for the standard collocation approach, and had to collocate the product of $f(x)$ with a weight function, $(1 - \xi(x)^2)^{\lambda-1/2}$, to achieve the result. Experimentally, they were able to show that the standard collocation approach, which we have used, produced similar errors.

6.1.4 Choice of Reconstruction Parameters

As is stated in the theorems of the previous section, if λ and m are chosen as $\lambda = m = \beta\epsilon N$, where $\beta < 2/27$ in each subinterval where the function being reconstructed is assumed to be analytic, the reconstruction will be spectrally accurate. It is not necessary, and usually not advisable, to choose $\lambda = m$. In practice, we are often more concerned with obtaining results for a fixed N , rather than achieving an exponential convergence rate.

If the function to be postprocessed consists of homogeneous features throughout the computational domain, the reconstruction parameters can be successfully chosen as $\lambda = k_\lambda\epsilon N$ and $m = k_m\epsilon N$ for each subinterval where k_λ and k_m are user chosen, globally applied

parameters. We refer to this strategy to selecting reconstruction parameters as the *global approach*. In all previous applications of the GRP in the literature, the method was applied to functions with homogeneous structure and it was possible to chose the parameters in this way.

However, in problems with solutions that vary in structure throughout the computational domain, the reconstruction parameters may need to be chosen independently in each subinterval. We refer to this strategy as the *local approach*. For example, this is the case for the function shown with its Chebyshev approximation in figure (6.1). It is not possible to use globally chosen reconstruction parameters, k_λ and k_m , and obtain a successful postprocessing. In the regions of piecewise constant data, reconstruction parameters of $\lambda = 2$ and $m = 2$ provide good results. In the region $[0, 1]$ which consists of a narrow exponential spike, the data contains small scale structures which will require a large value of m and small value of λ . In the region $[-1, -0.7]$ the function is of moderate detail and reconstruction can be accomplished with moderate values of m and λ .

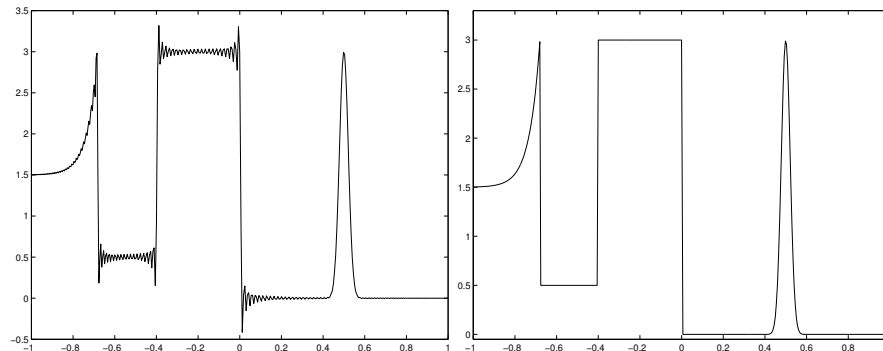


Figure 6.1: Function and Chebyshev partial sum approximation

To date there is no known method to choose optimal values of the reconstruction parameters m and λ . The parameters remain very problem dependent and the best advice that can be offered at this point is the knowledge of the parameters that produced good results in the numerical examples which will follow. Work is under way on choosing optimal parameters and results will be reported in a future paper.

6.1.5 The collocation grid

In previous applications of the GRP in the literature, the method has only been applied to functions known on the CGL grid. For approximations produced by collocation schemes, it was shown in [34] that spectral accuracy could be recovered from point values known at the Chebyshev points. However, numerical evidence indicates the method may also be applied successfully on grids arising from mappings of the CGL grid. The determining factor in the accuracy of the reconstruction is how well the chosen grid can capture the function. For example, consider the piecewise analytic function shown in the figure 6.2 with its discrete Chebyshev approximation.

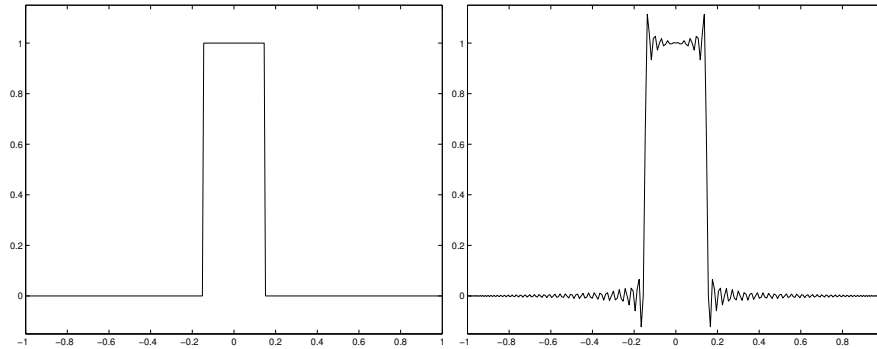


Figure 6.2: Square wave and Chebyshev approximation

The errors of the postprocessed functions in log scale are shown in figure 6.3 for three different resolutions, $N = 80$, $N = 160$, and $N = 320$. In the figure on the left, the grid is formed with map (2.7) with $\gamma = 0.9999$ which results in a near uniform grid. On the right is the function known on the Chebyshev grid (1.1). The reconstruction parameters were taken as $\lambda = 0.2\epsilon N$ and $m = 0.1\epsilon N$. The errors decay at similar rates on both grids but the mapped grid which places more grid points in the center of the domain produces a better approximation in that region than the CGL grid.

The fact that the GRP may be applied on mapped grids is important, as in applications, the CGL grid is seldom the proper grid to use.

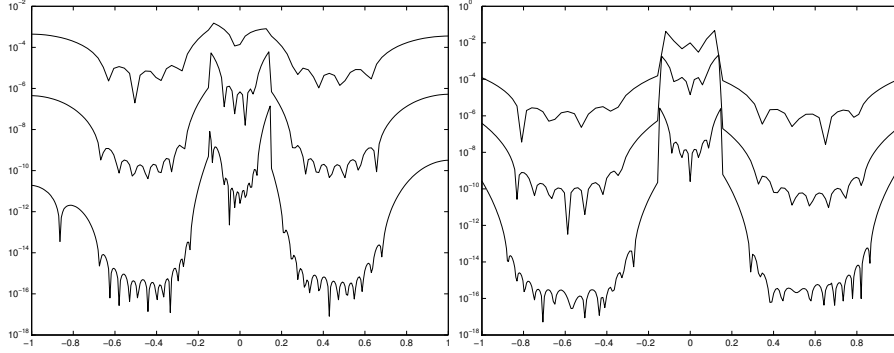


Figure 6.3: Mapped grid error vs. CGL grid error

6.1.6 Computational Expense

Chebyshev Gauss-Lobatto quadrature is used to approximate (6.5) which leads to the Gegenbauer approximation

$$f_m^\lambda(x(\xi)) = \sum_{l=0}^m \left[\frac{1}{\widehat{h}_l^\lambda} \frac{\pi}{N} \sum_{j=0}^N \frac{1}{c_j} (1 - \xi_j^2)^\lambda C_l^\lambda(\xi_j) \sum_{n=0}^N a_n T_n(\epsilon \xi_j + \delta) \right] C_l^\lambda(\xi). \quad (6.9)$$

The approximation at each grid point, x_j , involves a triple summation which is very computationally expensive as N and m grow. In an effort to reduce the computation required, (6.9) can first be written as

$$f_m^\lambda(x(\xi)) = \frac{\pi}{N} \sum_{j=0}^N \frac{1}{c_j} (1 - \xi_j^2)^\lambda f_N(\epsilon \xi_j + \delta) \sum_{l=0}^m \frac{C_l^\lambda(\xi) C_l^\lambda(\xi_j)}{\widehat{h}_l^\lambda}. \quad (6.10)$$

The Christoffel-Darboux formula [15], can be used to get

$$\sum_{l=0}^m \frac{C_l^\lambda(\xi) C_l^\lambda(\xi_j)}{\widehat{h}_l^\lambda} = \frac{k_m}{k_{m+1} \widehat{h}_l^\lambda} \frac{C_{m+1}^\lambda(\xi_j) C_m^\lambda(\xi) - C_m^\lambda(\xi_j) C_{m+1}^\lambda(\xi)}{\xi_j - \xi} \quad (6.11)$$

where

$$k_m = \frac{2^m \Gamma(\lambda + m)}{m! \Gamma(\lambda)} \quad (6.12)$$

to eliminate one of the summations. Thus

$$f_m^\lambda(x(\xi)) = \frac{\pi}{N} \frac{k_m}{k_{m+1} \widehat{h}_l^\lambda} \sum_{j=0}^N \frac{1}{c_j} (1 - \xi_j^2)^\lambda f_N(\epsilon \xi_j + \delta) \frac{C_{m+1}^\lambda(\xi_j) C_m^\lambda(\xi) - C_m^\lambda(\xi_j) C_{m+1}^\lambda(\xi)}{\xi_j - \xi}. \quad (6.13)$$

When $\xi_j = \xi$, L'Hospital's rule can be used as well as the equality $2\lambda C_{m-1}^{\lambda+1}(\xi) = \frac{d}{d\xi} C_m^\lambda(\xi)$, to rewrite (6.13) as

$$f_m^\lambda(x(\xi_j)) = \frac{\pi}{N} \frac{k_m}{k_{m+1} \widehat{h}_l^\lambda} \sum_{j=0}^N \frac{1}{c_j} (1 - \xi_j^2)^\lambda f_N(\epsilon \xi_j + \delta) 2\lambda (C_m^{\lambda+1}(\xi_j) C_m^\lambda(\xi_j) - C_{m+1}^\lambda(\xi_j) C_{m+1}^{\lambda+1}(\xi_j)).$$

$$f_N(x) = \sum_{n=0}^N a_n T_n(x) \quad (6.14)$$

is most efficiently evaluated using Clenshaw's recurrence formula [51].

6.1.7 Roundoff error

The Gegenbauer polynomials grow very rapidly with λ and m which leads to a round off error that may completely ruin the approximation. While the use of the Christoffel-Darboux formula reduces the computational effort, it adds to the round off error problem as two Gegenbauer polynomials are multiplied together. To counteract this [23], the $\frac{\pi}{N} \frac{k_m}{k_{m+1} \widehat{h}_l^\lambda}$ portion of (6.13) can be rearranged and used to our advantage. In each product of two Gegenbauer polynomials, multiply one by $\frac{1}{N}$ and the other by $\frac{\pi k_m}{k_{m+1} \widehat{h}_l^\lambda}$. Both quantities are small and decreasing values with respect to m , λ , and N , and offset the relative large size of the polynomial product and help prevent round off error. The approximation (6.13) now reads

$$f_m^\lambda(x(\xi)) = \sum_{j=0}^N \frac{1}{c_j} (1 - \xi_j^2)^\lambda f_N(\epsilon \xi_j + \delta)$$

$$\frac{\left(\frac{1}{N}C_{m+1}^\lambda(\xi_j)\right)\left(\frac{\pi k_m}{k_{m+1}\widehat{h}_i^\lambda}C_m^\lambda(\xi)\right) - \left(\frac{1}{N}C_m^\lambda(\xi_j)\right)\left(\frac{\pi k_m}{k_{m+1}\widehat{h}_i^\lambda}C_{m+1}^\lambda(\xi)\right)}{\xi_j - \xi} \quad (6.15)$$

6.1.8 A Hybrid Approach

Even with the computational savings made above via the Christoffel-Darboux formula, Gegenbauer Reconstruction may still be very computationally expensive in higher dimensions for large values of N . A hybrid approach was suggested in [23] which uses an exponential filter in smooth regions and the GRP in the neighborhood of discontinuities.

6.1.9 Two Dimensions

The GRP extends in an obvious way to higher dimensions. The two dimensional Gegenbauer approximation is

$$f_{m_1, m_2}^{\lambda_1, \lambda_2}(x(\xi_x), y(\xi_y)) = \sum_{l_1=0}^{m_1} \sum_{l_2=0}^{m_2} \widehat{g}_{l_1, l_2}^{\lambda_1, \lambda_2} C_{l_1}^{\lambda_1}(\xi_x) C_{l_2}^{\lambda_2}(\xi_y)$$

with

$$\begin{aligned} \widehat{g}_{l_1, l_2}^{\lambda_1, \lambda_2}(x(\xi_x), y(\xi_y)) &= \frac{1}{\widehat{h}_{l_1}^{\lambda_1}} \frac{1}{\widehat{h}_{l_2}^{\lambda_2}} \int_1^{-1} \int_1^{-1} (1 - \xi_x^2)^{\lambda_1 - 1/2} (1 - \xi_y^2)^{\lambda_2 - 1/2} \\ &\quad C_{l_1}^{\lambda_1}(\xi_x) C_{l_2}^{\lambda_2}(\xi_y) f_N(\epsilon_x \xi_x + \delta_x, \epsilon_y \xi_y + \delta_y) d\xi_x d\xi_y. \end{aligned}$$

Part IV

Numerical Examples

The examples have been chosen with the idea of examining the effect of a source term as well as examining the consequences of a viscosity operator that is spatially varying and which vanishes on the boundary. Additionally, some of the solutions have homogeneous features throughout their domain, while other solutions have subintervals of varying detail. This will allow for a discussion of strategies to select the GRP parameters in each case.

First, the equations of Hyperbolic Heat Transfer will be solved. The problem is linear, solvable without spectral viscosity, and has an exact solution available which consists of homogeneous features. The GRP is easy to apply to the solutions of this problem and spectacular results are achieved.

The next example is the Reactive Euler Equations, a nonlinear system of conservation laws with a source term. Spectral Viscosity will be needed, but there will not be any significant solution features close to boundaries. Thus the solution will not be affected by the lack of viscosity near the boundaries and the effect of the source term can be isolated.

Next, the numerical solution of a one-dimensional fluidized bed model will be examined. Strong shocks will have to pass through a region of diminished spectral viscosity near the boundary at the bottom of the bed. The solutions consist of varying subintervals of detail. An exact solution exists for the Riemann problem for the one-dimensional fluidized bed model without the source term and will provide another accuracy check on the method.

The final application will be a two-dimensional fluidized bed model. The governing equations are a nonlinear system of hyperbolic conservation laws with a source term coupled with an elliptic equation for determining a stream function. The problem inherits all the difficulties associated with the one-dimensional case, and the additional complications of the coupled elliptic equation and higher dimensionality. However, the shocks are not as strong in the two-dimensional fluidized bed when compared to the one-dimensional model. A high quality solution is produced by the SSV method without postprocessing.

Chapter 7

Hyperbolic Heat Transfer

In situations when the elapsed time during a transient is very small or when temperatures near absolute zero are involved, the classical diffusion (parabolic) theory of heat transfer breaks down since the wave nature of thermal energy transport becomes dominant. The hyperbolic heat equation models this process and results in energy propagating through a medium as a wave with sharp discontinuities at the wave front.

The dimensionless governing equations of Hyperbolic Heat Transfer are

$$T_t + Q_x = S/2 \quad (7.1)$$

$$Q_t + T_x = -2Q \quad (7.2)$$

where $T(x, t)$ is the temperature, $Q(x, t)$ is the heat flux, and $S(x, t)$ is the energy generation rate.

Previously in the literature, the numerical solutions of hyperbolic heat transfer problems have been obtained using the Finite Element Method in [8], and by MacCormack's method in [28], and [27]. In previous numerical investigations, typically 1000 grids points were used and numerical oscillations still remained in the solution. Non-oscillatory finite difference methods [52, 44] which suppress oscillations by using a flux or slope limiter are available. However, the methods are also known to smear the solution at sharp fronts. The spectral method with postprocessing will resolve all fronts sharply.

The first attempt to apply spectral methods to the problem was in [38] where conservative smoothing was used to obtain results with significantly fewer grid points than MacCormack's method while

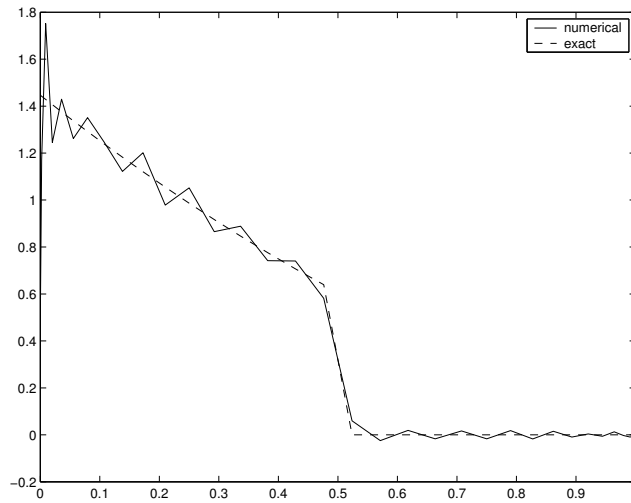


Figure 7.1: spectral approximation

eliminating most of the spurious oscillations. Conservative smoothing amounts to an artificial viscosity, which can be applied selectively in both space and time. However, the global nature of spectral methods causes a spatially localized viscosity to be felt throughout the computational domain and the locally applied viscosity degrades the accuracy of the entire solution, not just around discontinuities. Additionally, conservative smoothing had difficulty controlling oscillations located close to boundaries. The approach used here will allow the problem to be solved with spectral methods without adding any artificial viscosity.

Both examples used the initial conditions $T(x, 0) = 0$ and $Q(x, t) = 0$ for $x \in [0, 1]$. The first example uses boundary conditions of $Q(0, t) = 1$, $Q(1, t) = 0$, $T_t(0, t) = -Q_x(0, t)$, and $T_x(1, t) = 0$ with the energy generation rate, S , set to zero. Exact solutions for problem 1 may be found in [38] and the exact solution to problem 2 is stated in [50]. Two problems with analytical solutions were chosen in order to test the accuracy of the numerical method, however, there are many other cases for which the problem can not be handled analytically.

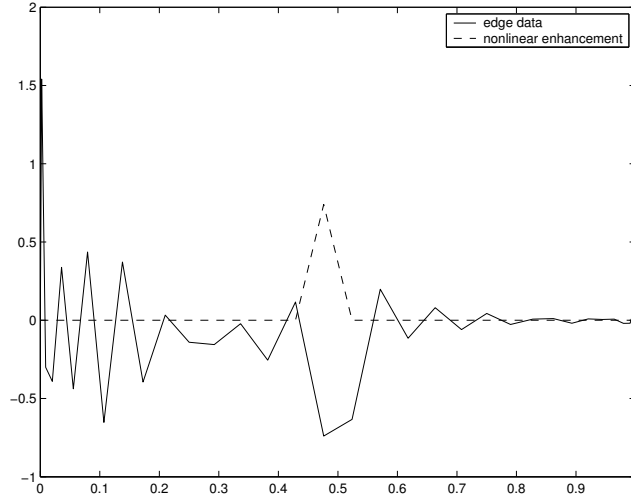


Figure 7.2: edge locations

7.1 Problem 1

In the figure 7.1, the temperature solution, T , is shown at time $t = 0.5$ with $N = 33$ CGL grid points (1.1). The solution was advanced in time with a fourth order Runge-Kutta method and a time step of $\Delta t = 0.001$. Strong oscillations are noticeable at the boundary $x = 0$, due to the jump in the heat flux, Q , which is felt by the temperature.

An edge is found to be at $x = 0.476$ with the parameters $J = 200$, $Q = 4$, and $\eta = 2$. This choice of edge detection parameters results in jumps of 0.65 and larger being located. The exact jump is 0.65 in magnitude. By specifying $\eta = 2$, the oscillation near $x = 0$ is not falsely determined to be a jump in the function. With only 33 grid points, the convergence of the edge series in figure 7.2 is not yet readily apparent. However, if the edge detection parameters are chosen appropriately, the correct edge locations will be found.

After the edges have been located, the GRP is applied in each smooth subinterval with by using global parameters chosen as $k_\lambda = 0.3$ and $k_m = 0.1$ which results in $m = 2$ and $\lambda = 4.7$ in subinterval $(0, 0.476)$ and $m = 2$ and $\lambda = 5.2$ in subinterval $(0.476, 1)$.

After postprocessing (figure 7.3) the numerical and exact solutions are virtually identical. The pointwise error between the exact and postprocessed solution is shown in figure 7.4 where the max-

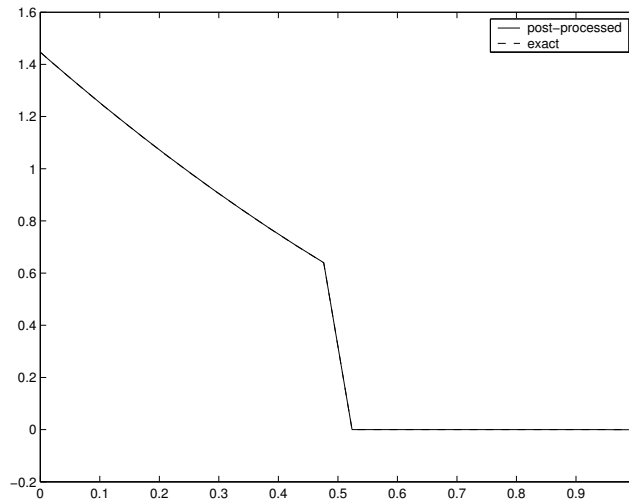


Figure 7.3: postprocessed vs. exact

imum pointwise error is less than 0.00045. Only 1/30 of the grid points are necessary to obtain results superior to those obtained in [28] with MacCormack's method. Figure 7.5 displays the MacCormack's method solution of the first example with $N = 1000$ and a small time step. Despite using substantially more grid points the solution is still oscillatory around the steep front due mainly to phase speed errors. Spectral methods give accurate phase speeds for all modes, while second order finite difference methods typically only resolve first few modes accurately [46]. Unlike the spectral solution, there is no known postprocessing technique which is capable of recovering second order accuracy over the entire computational domain of the finite difference approximation.

7.2 Problem 2

The second example uses boundary conditions of $Q(0, t) = 0$, $Q(1, t) = 0$, $T_x(0, t) = 0$, and $T_x(1, t) = 0$, with the energy generation rate specified as $S(x, t) = \frac{1}{dn}$ if $0 \leq x \leq dn$ and zero otherwise. The energy generation rate, S , represents a pulsed energy source released instantaneously at time $t = 0$. Such an energy source could model the application of a strong laser pulse at the boundary of an absorbing medium encountered in the annealing of semiconductors.

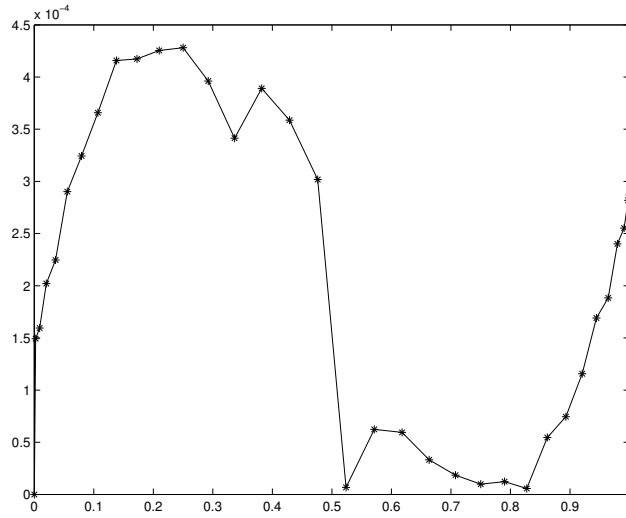


Figure 7.4: pointwise error, postprocessed vs. exact

The temperature solution with $dn = 0.05$ is shown in figure 7.6 at time $t = 0.5$ with $N = 99$ grid points distributed with the map (2.7) with $\gamma = 0.96$. By taking the map parameter as $\gamma = 0.96$, the grid becomes closer to evenly spaced and better resolution is realized in the center of the domain. The solution was advanced in time in an unsplit formulation with a fourth order Runge-Kutta method and a time step of $\Delta t = 0.0005$.

Edges, figure 7.7, are found to be at $x = 0.447$ and $x = 0.541$ with the parameters $J = 5000$, $Q = 3$, and $NE = 1$. With these choices of the edge detection parameters, only jumps of magnitude greater than 1.72 are found. Other combinations of J and Q could work equally as well.

After the edges have been located, the GRP is applied in each smooth subinterval by using the global parameters $k_\lambda = 0.2$ and $k_m = 0.02$. The results are shown in figure 7.8.

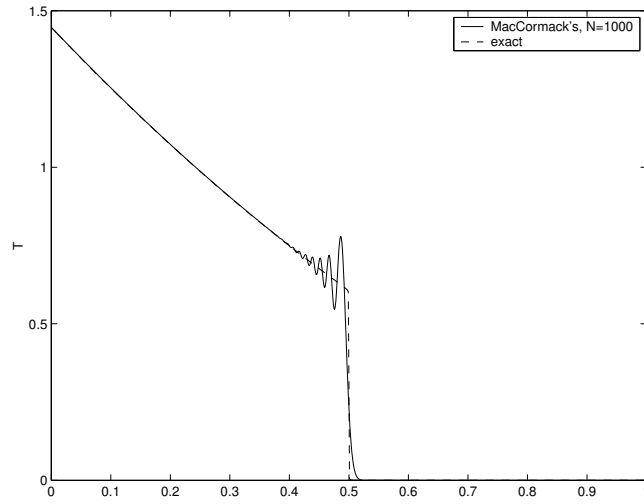


Figure 7.5: Problem 1: MacCormack's vs. exact

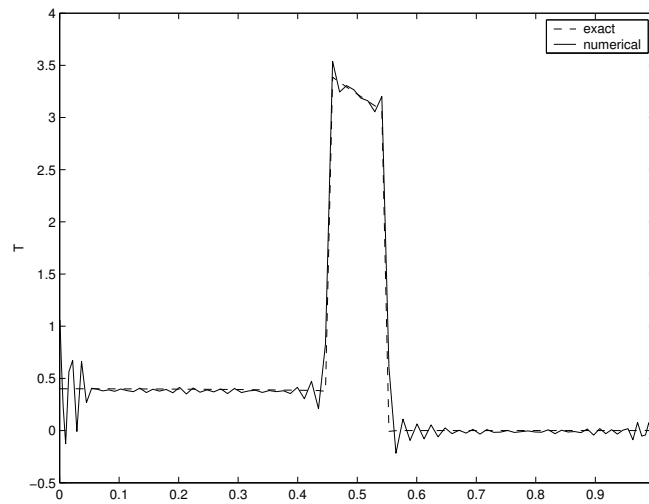


Figure 7.6: spectral approximation

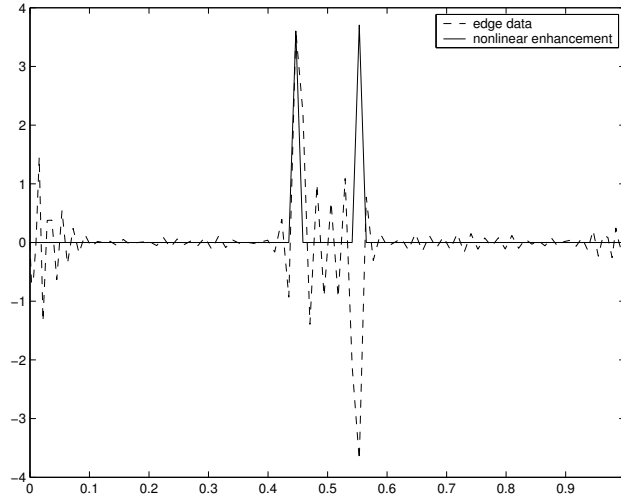


Figure 7.7: edge locations

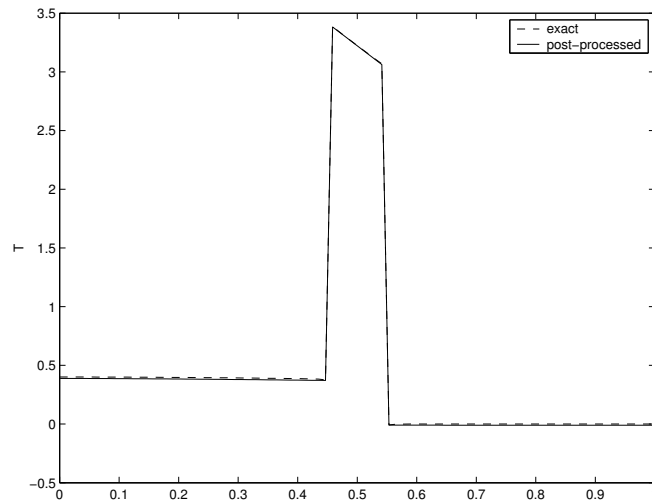


Figure 7.8: postprocessed vs. exact

Chapter 8

Reactive Euler Equations

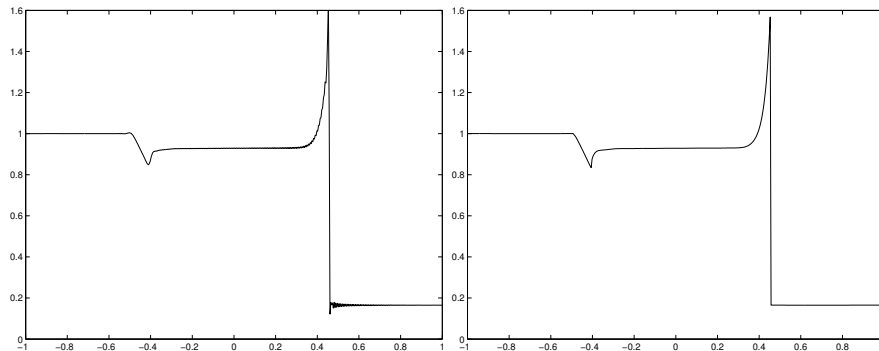


Figure 8.1: SSV approximation (left), postprocessed (right)

The Reactive Euler Equations will be our first test of the method on a nonlinear system of conservation laws with a source term. The boundary conditions are inflow/outflow conditions specified on the characteristic variables. The problem does not have shocks located close to computational boundaries as will be the case in the fluidized bed problems which follow.

The reactive Euler Equations can be expressed in the form of a nonlinear system of hyperbolic conservation laws with a source term as

$$\begin{aligned}\rho_t + (\rho v)_x &= 0 \\ (\rho v)_t + (\rho v^2 + P)_x &= 0 \\ E_t + E_x &= 0 \\ (\rho Z)_t + (\rho v Z)_x &= \rho Z - K(P/\rho)\rho Z\end{aligned}$$

The gas density is $\rho(x, t)$, $v(x, t)$ is the velocity, $E(x, t)$ the total Energy, $P(x, t)$ the pressure of the gas, and $Z(x, t)$ the mass fraction of unburnt gas. The total energy is calculated via the equation of state

$$E = \frac{P}{\gamma - 1} + \frac{1}{2}\rho v^2 + q_0\rho Z$$

where q_0 is the heat release and γ is the ratio of specific heats taken to be 1.4. In the source term, $K(P/\rho)$ represents the reaction rate of the burning process

$$K(P/\rho) = \begin{cases} 1/\tau & \text{if } P/\rho \geq T_0 \\ 0 & \text{if } P/\rho < T_0 \end{cases}$$

where T_0 is the ignition temperature and τ is time scale of the chemical reaction.

In our example, we take $T_0 = 0.22$, $\tau = 0.025$, $q_0 = 1$. The Riemann initial conditions are given as $\rho = 1.5$, $v = 0$, $P = 1.0$, and $Z = 0$ for $x < 0$ and $\rho = 0.9$, $v = -0.5$, $P = 0.15$, and $Z = 1.0$ for $x \geq 0$. The computational domain is the interval $[-1, 1]$.

The SSV method was applied with $C = 3$, $s = 2$, and $N = 512$ on a grid formed with map (2.7) with $\gamma = 0.9999$. The solution was advanced in time in an unsplit formulation (section 2.4.1) with a fourth order Runge-Kutta formula (2.2). The results at $t = 0.5$ are shown in figure 8.1. The edge detection procedure with $J = 10$, $Q = 1$, and $\eta = 1$ finds edges in the solution at $x = 0.4571$. With $J = 50$, $Q = 1$, and $\eta = 4$, edges in the first derivative of the solution in the interval $(-1, -0.4)$ are found at $x = -0.40595$ and $x = -0.49266$. The solution contains subintervals of varying structure which necessitates that the reconstruction parameters must be chosen using the local approach. Table 8.1 lists the locally specified reconstruction parameters that were used to produce the postprocessed solution shown in figure 8.1.

The postprocessed spectral data is compared (see figure 8.2) with the solution by the second order Nessayhu-Tadmor [49] scheme on a very fine grid of 4000 grid points. There is very good agreement between the two solutions. The Spectral SSV method gives a slightly sharper resolution of the shock at $x = 0.5471$ with about one eighth of the grid points than the finite difference method used.

The numerical evidence indicates that the spectral viscosity methods and postprocessing methods developed for systems of homogeneous conservation laws are also applicable in the presence of a

subinterval	m	λ
$(-1, -0.40595)$	2	2
$(-0.40595, -0.49266)$	5	7
$(-0.49266, 0.4571)$	15	2
$(0.4571, 1)$	2	2

Table 8.1: local reconstruction parameters

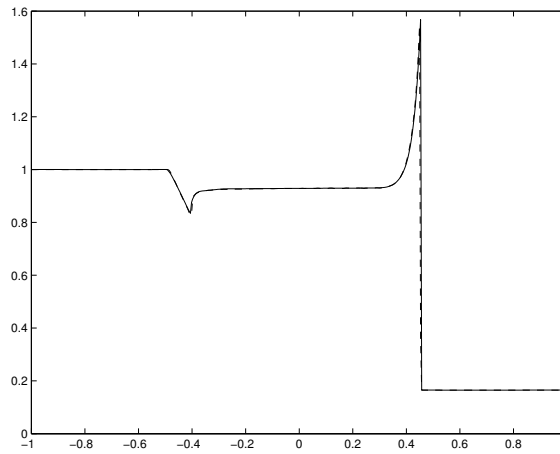


Figure 8.2: spectral postprocessed (solid) vs. NT

source term and that they may be applied on grids other than the standard Chebyshev grid. The unsplit formulation of the problem seems to perform well and there is no need to use a split method which solves a separate equation involving the source term.

Chapter 9

One-Dimensional Fluidized Bed Equations

Fluidized beds are used in the chemical and fossil fuel processing industries to mix particulate solids and fluids (gases or liquids). A typical fluidized bed consists of a vertically oriented chamber, a bed of particulate solids, and a fluid flow distributor at the bottom the chamber. The fluid flows upward through the particles creating a force that counteracts gravity at which time a state of minimum fluidization is reached. Stronger gas inflows (more than is necessary to maintain minimum fluidization) lead to pockets of gas, or equivalently low particle concentrations, resembling bubbles in a liquid travelling upward through the particles. Each rising bubble pushes a large amount of mass in front of it. Particles move downward through and around the rising bubble until it reaches the top of the bed. A settled bed is reestablished, and the cycle repeats. Each set of upward moving particles is referred to as a slug.

In this chapter we consider only one-dimensional flow. Physically, this corresponds to flow in a narrow diameter fluidized bed. The fluidized bed model, which is described by a hyperbolic system of nonlinear conservation laws with a source term, was originally solved numerically in [11]. An exact solution to the homogeneous system with Riemann initial conditions has been developed in [13].

9.1 Fluidized Bed Equations

The variable x denotes the vertical height in the bed. Let $\alpha(x, t)$ denote the concentration of particles by volume, $v(x, t)$ the particle velocity, and $m(x, t) = \alpha(x, t)v(x, t)$ the particle momentum. The parameter α_0 is the concentration of particles at equilibrium (when $v = 0$) and α_p is the packing concentration which sets an upper limit for α where $\alpha \in [0, 1]$. The parameter α_{0u} denotes the particle concentration corresponding to the critical state dividing linearly stable and unstable states (the particle concentration at minimum fluidization). The constant $s = 3.5(1 - \alpha_{0u})^{2.5}(\alpha_p - \alpha_{0u})$ is related to the linear stability of the equilibrium solutions which correspond to states of uniform fluidization.

The model can be put in the form of a system of conservation laws with a source term as

$$\alpha_t + m_x = 0 \quad (9.1)$$

$$m_t + (m^2/\alpha + F(\alpha))_x = b(\alpha, m) \quad (9.2)$$

where

$$F(\alpha) = s^2\alpha + \frac{s^2\alpha_p^2}{\alpha - \alpha_p} + 2s^2\alpha_p \ln(|\alpha - \alpha_p|)$$

The function $b(\alpha, m)$ in the source term is given by

$$b(\alpha, m) = -\alpha + \frac{\alpha J - m}{(1 - \alpha)^{3.5}}$$

where $J = (1 - \alpha_0)^{3.5}$ represents the total volumetric flux through the bed. Increasing J (or decreasing α_0) corresponds to turning up the inflowing gas. Values $\alpha_0 < \alpha_{0u}$ correspond to large gas fluxes and have been shown to produce unstable states corresponding to slug-like solutions. Values $\alpha_0 > \alpha_{0u}$ give rise to stable states. From a mathematical point of view, the non-homogeneous system of conservation laws coincides with the Euler equations for an isentropic gas flow, subject to volumetric forces. The variables α , v , and $F(\alpha)$ play the role of density, velocity, and pressure respectively, in the Euler equations.

9.1.1 Vacuums and unphysical particle concentrations

A vacuum is said to exist at a collocation point if the particle concentration is zero. Numerically, we will assume that a vacuum exists

at a grid point if the concentration is either zero or it is very small ($|\alpha| < thres$). The system becomes meaningless at vacuum points as m^2/α is either undefined ($\alpha = 0$) or produces unrealistic values ($|\alpha| < thres$). At each vacuum point encountered in the numerical method, the corresponding values of v , and therefore m , are set equal to zero at that collocation point rather than using the spurious value ($|\alpha| < thres$) or NaN value ($\alpha = 0$). Values of α such that $|\alpha| < thres$ are retained and not set to zero. Stable approximations by the spectral method always produced $\alpha < \alpha_p$. In the spectral method, α must be allowed to take negative values even though a negative concentration is not physically meaningful, as this information is used in the GRP to postprocess the result. When it was attempted to artificially force the spectral method to work only with $\alpha > 0$, the quality of the postprocessed solution was adversely affected. More importantly, even though the spectral collocation method is conservative, if for $\alpha < 0$, α was redefined as $\alpha = 0$, the conservative properties of the method were destroyed and the method started producing mass. If the values of α were allowed to be negative, the method was conservative and mass was preserved to as many as six decimal places. In all reported results, the parameter $thres$ was taken to be $thres = 0.001$. After postprocessing the solution, all concentrations are such that $\alpha \geq 0$.

9.2 Numerical Results

In the reported results we have used $\alpha_{0u} = 0.55$ and $\alpha_p = 0.6$.

9.2.1 Homogeneous system

The first two problems solve the homogeneous system with Riemann initial data so that the Chebyshev SSV method with GRP postprocessing may be validated against an exact solution.

Our first example consists of a left-moving shock wave and a right moving rarefaction wave. The initial conditions are $v(x, 0) = 0$ for all x in a domain of $[-0.2, 0.2]$ and $\alpha(x, 0) = 0.3$ if $x < 0$ and $\alpha(x, 0) = 0.55$ if $x \geq 0$.

Figure 9.1 shows the solution advanced to time $t = 0.5$ with a fourth order Runge-Kutta method. The grid consists of 64 points distributed by map (2.9) with $\gamma = 0.25$. The use of the coordinate

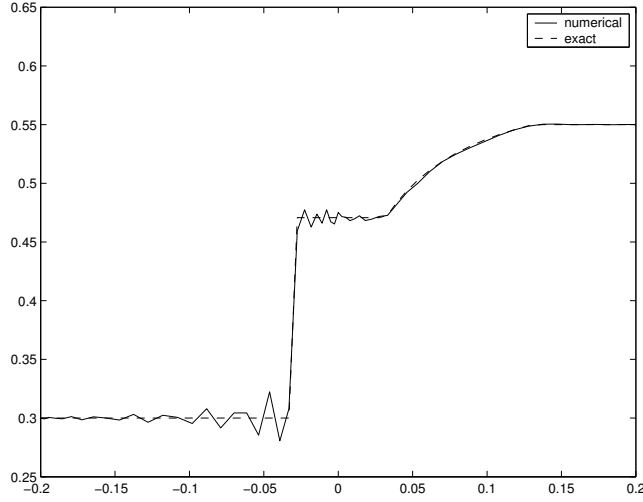


Figure 9.1: SSV approximation vs. exact

map has the effect of placing more points in the center of the domain. The SSV parameters used were $C = 1$ and $s = 4$ which produced a viscosity parameter of $\varepsilon = CN^{1-2s} = 2.27E - 13$ (or $\alpha = CN\Delta t = 0.16$ and $\beta = 8$ in the exponential filter).

The rarefaction wave is characterized by the solution having a discontinuous first derivative, thus edge detection must be applied to the first derivative of the solution in addition to the solution itself. The edge detection procedure with $Q = 1$ and $J = 1$ locates jumps of magnitude greater than 0.125. With these settings, the edge detection procedure locates edges in the function and the first derivative of the function at $x = -0.0331$, $x = 0.0331$, and $x = 0.1374$.

We were unable to get good postprocessed results by specifying the reconstruction parameters globally through the parameters k_λ and k_m . Global parameter specification failed due to the solution containing three intervals of piecewise constant values and a fourth interval, $(0.033, 0.1374)$, consisting of a function requiring different reconstruction parameters. Good results were obtained by specifying the GRP parameters locally in each smooth subinterval as listed in table 9.1. The postprocessed solution in figure 9.2.

With such a small viscosity parameter ε used, it is interesting to note that the numerical calculation is stable up to time $t = 0.5$ with

subinterval	m	λ
$(-0.2, -0.033)$	1	2
$(-0.033, 0.033)$	1	3
$(0.033, 0.1374)$	4	1
$(0.1374, 0.2)$	1	2

Table 9.1: shock-rarefaction, local reconstruction parameters

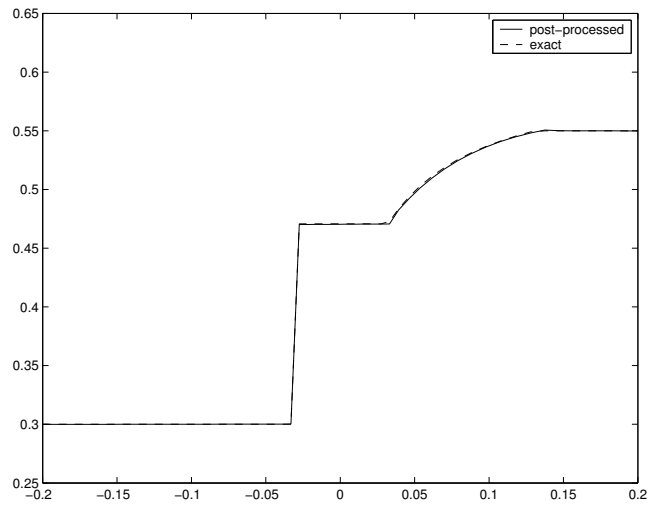


Figure 9.2: postprocessed vs. exact

$\varepsilon = 0$. The result is shown in figure 9.3. The solution is considerably more oscillatory than the solution with the small amount of viscosity added and more significantly, it was impossible to obtain an accurate postprocessed solution from the approximation, indicating that the numerical solution is not converging to the entropy solution without the spectral viscosity being added.

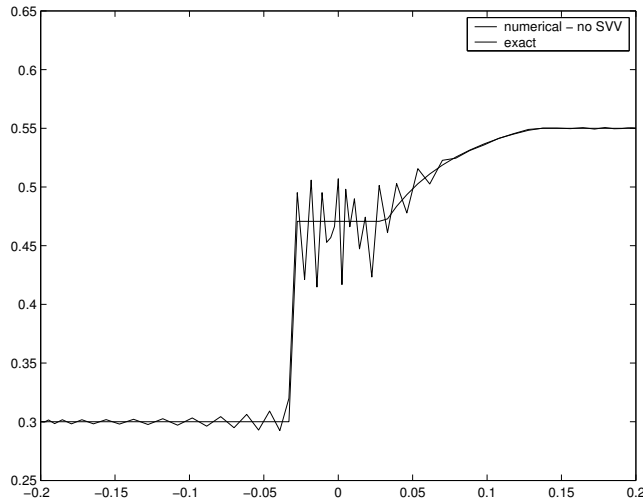


Figure 9.3: problem 1, without SSV

In our second example, the solution contains both left and right moving shocks. The initial conditions used were $v(x, 0) = 0.1$ if $x < 0$, $v(x, 0) = -0.3$ if $x \geq 0$ and $\alpha(x, 0) = 0.3$ if $x < 0$, and $\alpha(x, 0) = 0.4$ if $x \geq 0$ in a computational domain of $[-0.2, 0.2]$.

Figure 9.4 shows the computed solution at $t = 0.2$ on a grid with 64 collocation points. The collocation points were distributed with map (2.7) with $\gamma = 0.999$ which produces a near uniform grid. The shocks are stronger than in shock/rarefaction problem, and a stronger spectral viscosity is required. The SSV parameters used were $C = 25$ and $s = 2$ which produced a viscosity parameter of $\varepsilon = CN^{1-2s} = 9.537E - 5$ (or $\alpha = CN\Delta t = 0.8$ and $\beta = 4$ in the exponential filter).

The edge detection procedure with $J = 1$, $Q = 1$, and $\eta = 1$ located edges at $x = -0.07711$ and $x = 0.04499$ as shown in figure 9.5.

The homogeneous features of the solution throughout the compu-

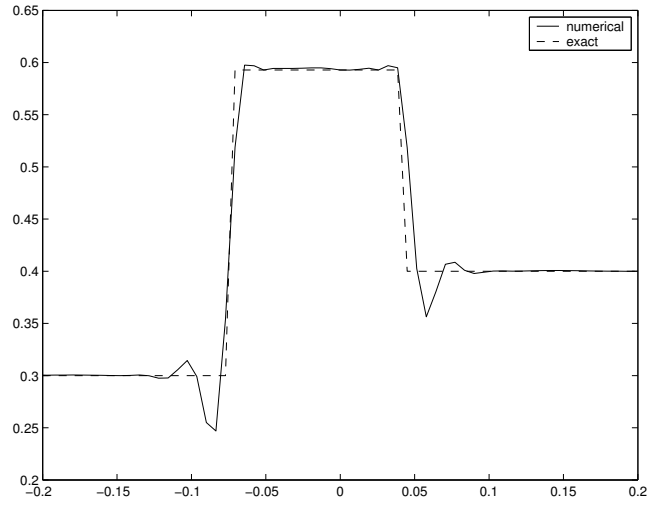


Figure 9.4: SSV approximation vs. exact

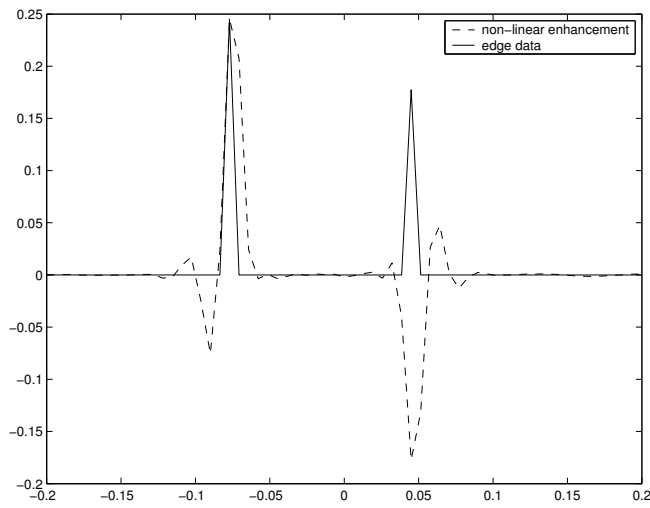


Figure 9.5: edge locations, shock/shock

tational domain allowed the reconstruction parameters to be chosen globally. The parameters were specified by setting $k_\lambda = 0.3$ and $k_m = 0.03$. The postprocessed solution is shown in figure 9.6.

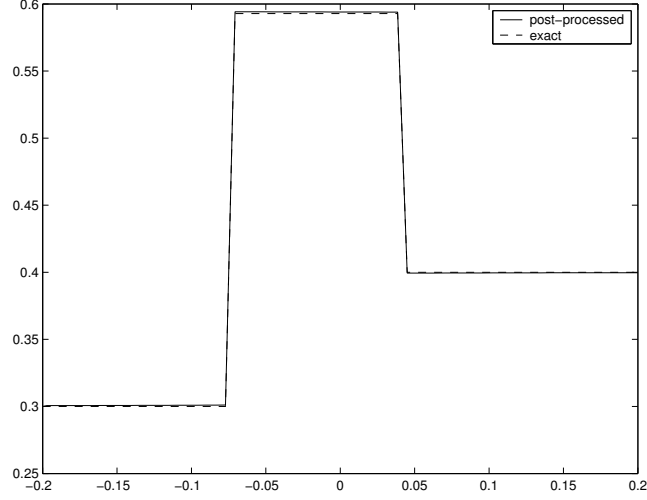


Figure 9.6: postprocessed vs. exact, shock/shock

The postprocessed spectral solution, in comparison with the second order Godunov methods used in [11] and [13], used less grid points to produce solutions which contained neither smears nor overshoots at shock locations in both examples.

9.2.2 Slugging Problem

Now we consider the numerical solution of the fluidized bed model and produce numerical solutions which replicate the slugging behavior observed in fluidized beds. To advance the system of conservation laws with the source term in time, a split method was used to take advantage of exact evaluation of the fractional steps involving the source term and the SSV term. The splitting used was

$$u_t = b(u) \quad (9.3)$$

$$u_t + f(u)_x = 0 \quad (9.4)$$

$$u_t = SVV(s, C, N) \quad (9.5)$$

$$u_t = b(u). \quad (9.6)$$

For $t > 0$, the starting value for each equation (9.3) through (9.6) is provided by the solution of the previous equation. Equation (9.4) is solved with a time step Δt and advanced in time with an explicit second order Runge-Kutta method. Equation (9.5) is evaluated exactly over a time step of size Δt according to (4.5). The fractional steps involving the source terms, (9.3) and (9.6), may be evaluated exactly over time steps of size $\Delta t/2$ by solving the linear ODE

$$\frac{dm}{dt} = b(\alpha, m) = -\alpha + \frac{\alpha J - m}{(1 - \alpha)^{3.5}}$$

in closed form. Thus, in the split steps involving the source term, m can be updated as

$$m = \exp\left(\frac{-\Delta t}{2(1 - \alpha)^{3.5}}\right) [a(1 - \alpha)^{3.5} - \alpha J + m] - a(1 - \alpha)^{3.5} + \alpha J.$$

The described fractional steps amount to Strang splitting [59] and maintains second order accuracy in time. A typical implementation of Strang splitting would have evaluated (9.5) over a time step of size $\Delta t/2$ before and after (9.4). However, the SSV term is just a filter at every time level and the exact location at which it is applied in time is irrelevant to temporal accuracy. An unsplit formulation of the problem was also solved. The full system was advanced in time with an explicit fourth order Runge-Kutta method with no noticeable differences in the solution being observed.

Boundary conditions are imposed on v which physically correspond to perforated plates preventing the flow of particles. At both endpoints of the computational interval the value of the velocity is set to $v = 0$. All numerical simulations began from a state of uniform fluidization, in which $\alpha = \alpha(x)$ is found by letting $v = 0$ and $\alpha_0 = \alpha_{0u}$ in the system. This results in the ODE (9.7),

$$[F(\alpha)]_x = \psi(\alpha, m) \quad (9.7)$$

with an initial condition of $\alpha(0) = \alpha_{0u}$, to determine α . The ODE (9.7) may be expressed in the form (9.8) which is more suitable for numerical evaluation by an ODE solver.

$$\alpha_x = \frac{(\alpha - \alpha_p)^2}{s^2 \alpha} \left[\frac{(1 - \alpha_0)^{3.5}}{(1 - \alpha)^{3.5}} - 1 \right]. \quad (9.8)$$

A typical initial concentration is pictured in figure 9.7. After the initial condition for α is found, α_0 is set equal to 0.4 and a steady state no longer exists and instabilities in the form of slugs are expected in the solution.

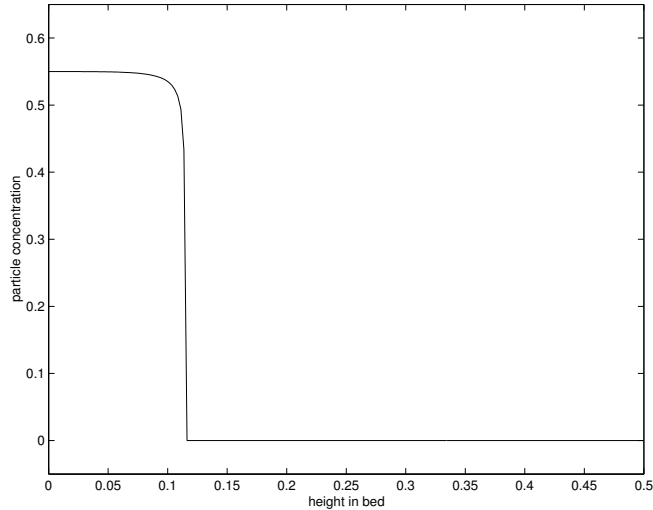
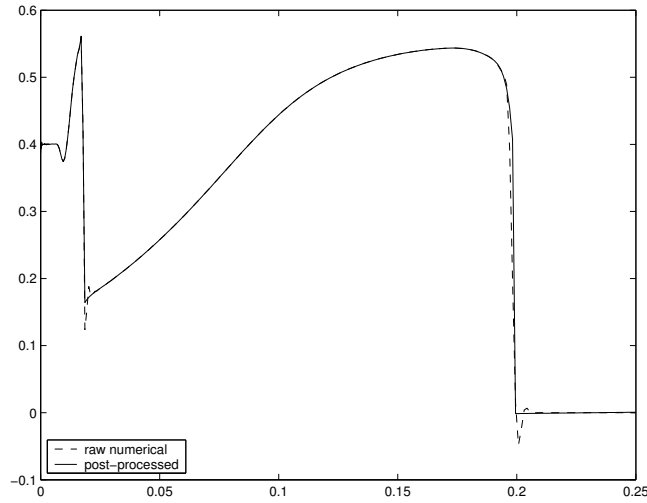


Figure 9.7: initial concentration α , slugging problem

Figure 9.8 shows the SSV solution at $t = 0.5$, at which time the slugging behavior is becoming evident. We have taken $N = 256$ and the CGL grid (1.1) was used. The SSV parameters were $C = 5$ and $s = 2$ which produced a viscosity parameter of $\varepsilon = CN^{1-2s} = 0.00000012$ (or $\alpha = CN\Delta t = 0.0128$ and $\beta = 4$ in the exponential filter). After postprocessing, the physically unrealistic concentrations, $\alpha < 0$, have all been replaced by $\alpha \geq 0$.

The edge detection procedure, figure 9.9, with $J = 1$, $Q = 1$, and $\eta = 2$ located shocks at $x = 0.01778$ and $x = 0.19822$. The postprocessed solution (figure 9.8) was obtained by locally specifying the reconstruction parameters in each smooth subinterval as listed in table 9.2.

In order to validate the method against a method with that has a much longer track record in solving nonlinear conservation laws, the postprocessed solution is compared with a solution by Roe's method [52]. In figure 9.10, the Roe's method solution with $N = 1024$ is shown with the postprocessed spectral solution from figure 9.8. There is a good agreement between the two solutions. The

Figure 9.8: SSV vs. postprocessed, $t=0.5$

subinterval	m	λ
(0,0.01778)	15	2
(0.01778,0.19822)	14	4
(0.19822,0.25)	1	1

Table 9.2: $t = 0.5$, local reconstruction parameters

only slight variance is towards the top of the bed where the area of zero concentration begins. This is largely due to the fact that the Godunov method is calculated on a uniform grid while the spectral method uses a nonuniform grid, which led to the height in the bed at which particles existed at $t = 0$ being slightly different. The system of equations exhibits a chaotic-like, sensitive dependence on initial conditions. The slightest variation of the initial condition results in a noticeably different concentration profile at later times.

By time $t = 4.0$, the slugging has become well developed throughout the bed. The solution is calculated by the spectral method with $N = 256$ on a grid formed with map (2.10) with $\gamma = 1$ and $\mu = 0$. The map causes the grid points to cluster densely around the gas inflow at $x = 0$ while lessening the density of grid points towards the other end of the interval where the particle concentration remains constant at zero. This grid distribution allows the problem to be resolved on the larger interval $[0, 0.4]$ without increasing the number

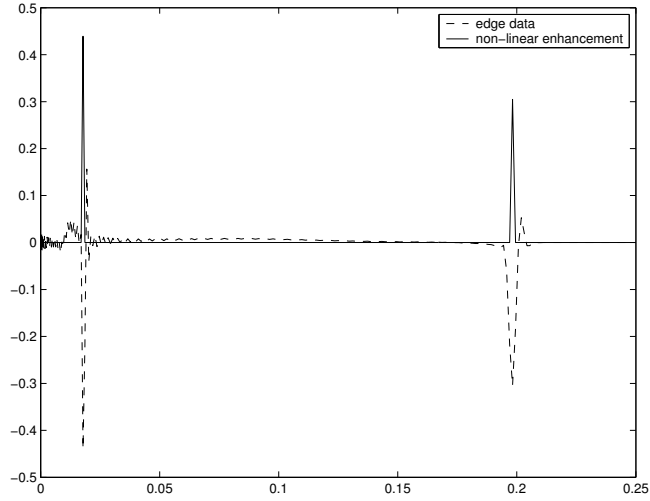


Figure 9.9: edge locations, $t=0.5$

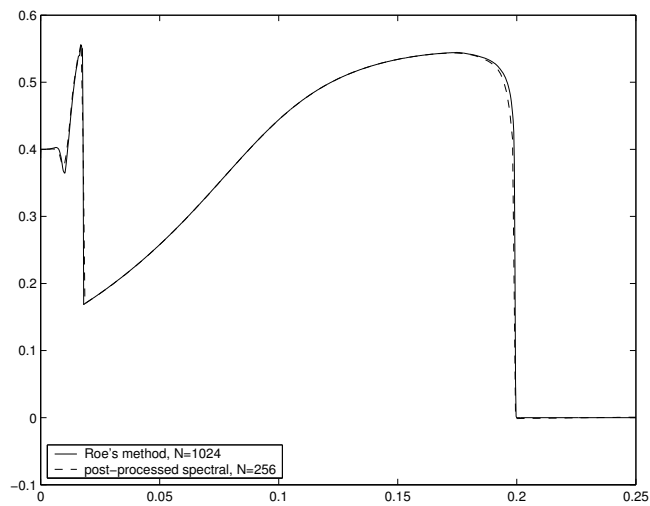
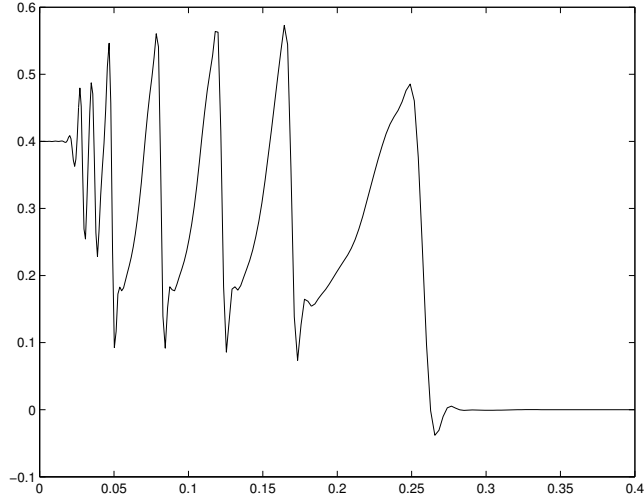


Figure 9.10: spectral postprocessed vs. reference, $t=0.5$

Figure 9.11: SSV solution, $t=4$

subinterval	m	λ
(0,0.028)	18	1
(0.028,0.037)	10	2
(0.037,0.049)	7	1.5
(0.049,0.081)	6	4
(0.081,0.122)	6	1.7
(0.122,0.169)	6	2.2
(0.169,0.257)	6	3.25
(0.257,0.4)	1	1

Table 9.3: $t=4$, local reconstruction parameters

of collocation points used. If the grid distribution (1.1) or (2.7) with $\gamma = 0.995$ are used, it is necessary to have $N = 512$ to get a well resolved solution. The SSV solution is shown in figure 9.11. The SSV parameters used were $C = 9$ and $s = 2$ which produced a viscosity parameter of $\varepsilon = CN^{1-2s} = 5.4E - 07$ (or $\alpha = CN\Delta t = 0.0576$ and $\beta = 4$ in the exponential filter). The edge detection parameters $J = 2$, $Q = 1$, and $\eta = 2$, produces the edge information in figure 9.12 and located shocks at $x = 0.028$, $x = 0.037$, $x = 0.049$, $x = 0.081$, $x = 0.122$, $x = 0.169$, and $x = 0.257$. The postprocessed solution in figure 9.13 was obtained by specifying the GRP parameters locally in each smooth subinterval as listed in table 9.3.

Chebyshev collocation methods are known to be conservative. It

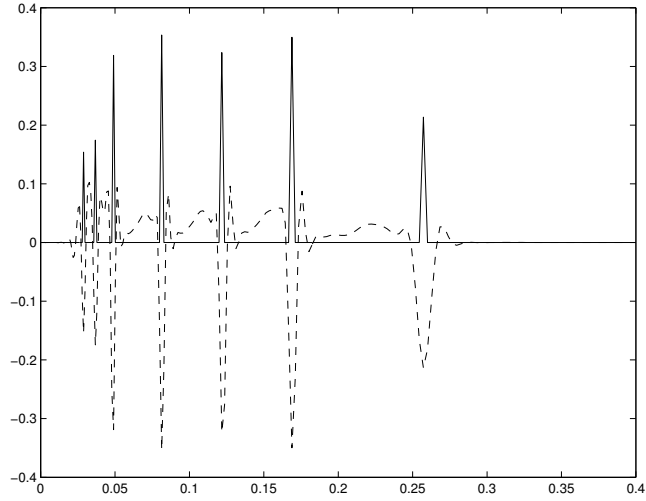


Figure 9.12: edge locations, $t=4$

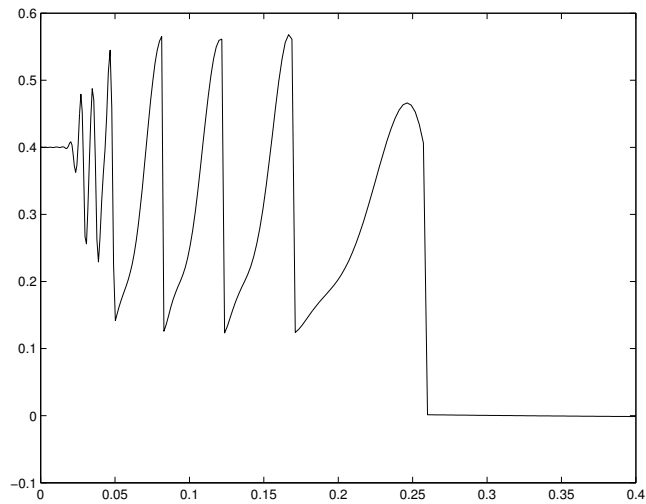


Figure 9.13: postprocessed, $t=4$

was recently shown in [9] that the addition of a spectral vanishing viscosity term does not affect the conservative properties of Chebyshev collocation methods. This fact was confirmed by our numerical experiments. Additionally, the postprocessing method was observed to preserve the conservative properties of the approximation. In all the numerical results presented, the amount of mass present initially never varied more than 1 percent in either the raw numerical data or in the post processed solution. For the approximations shown in figure 9.11 and 9.13, the initial total mass at $t = 0$ was 0.08303, the total mass in the SSV approximation at $t = 4.0$ was 0.08308, and the total mass of the postprocessed solution was 0.08302.

9.3 Selection of Edge Detection Parameters

The edges series (see section 5) for the spectral approximation represented in figure 9.8 is shown in figure 9.14. To pinpoint the location of the shocks represented in the SSV solution as jump discontinuities of magnitude 0.44 and 0.31, we can choose values of J and Q which result in $E = J^{1/Q} \sqrt{1/N} > 0.158$. For example, if we set $J = 3$ and $Q = 1$, we have $E = 0.1875$ and the desired edge locations with be found.

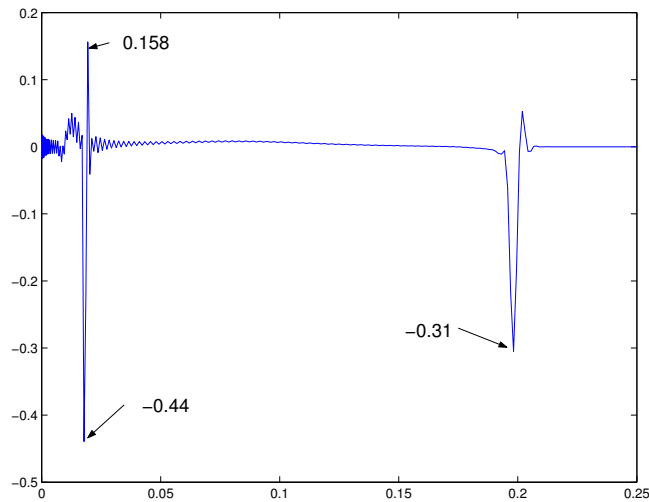


Figure 9.14: edge series

9.4 Selection of Postprocessing Parameters

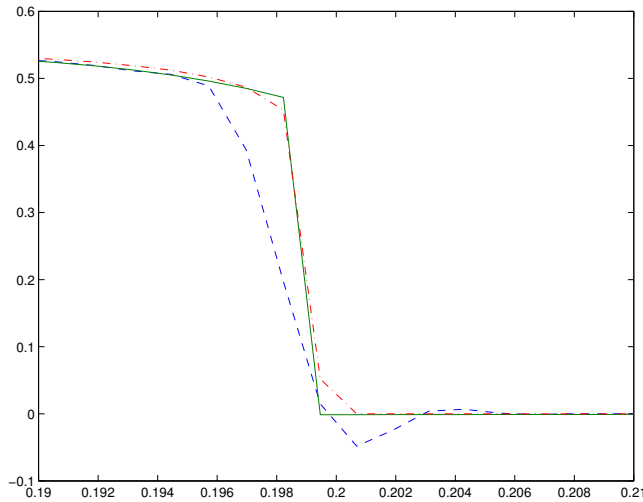


Figure 9.15: SSV, dashed; PP, solid; FD, dash-dotted

Figure 9.15 zooms in on the shock front in the neighborhood of $x = 0.198$ of the approximation from figures 9.8 and 9.10. The GRP parameters were $m = 14$ and $\lambda = 4$ in this region. The postprocessed spectral solution (PP) is in good agreement with the Roe's method reference solution (FD). The postprocessed spectral solution provides a sharper resolution of the shock front. The parameters which produced the good results were chosen by trial and error. They were adjusted until there was a good agreement with the reference solution.

The main drawback of the method is that there is no way to specify the parameters in advance. A method to optimize the parameters has not been developed. So while we can always select GRP parameters which result in the postprocessed solution comparing favorably with an exact or reference solution, the GRP is not ready to be used as a "black box" postprocessing method until a way to choose the optimal parameters in advance is available.

To illustrate the variations in the postprocessed solution that could result with a slightly different choice of the GRP parameters, consider the same SSV approximation, but postprocessed using GRP parameters $m = 25$ and $\lambda = 4$. The result is pictured in figure 9.16. Compared to the postprocessed solution in figure 9.15,

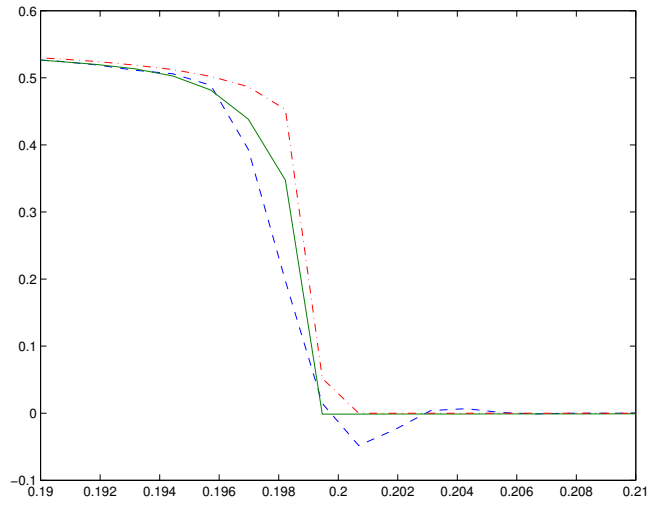


Figure 9.16: SSV, dashed; PP, solid; FD, dash-dotted

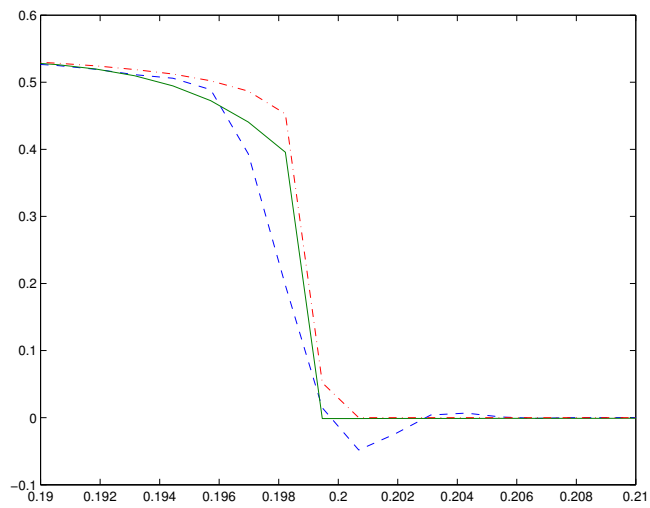


Figure 9.17: SSV, dashed; PP, solid; FD, dash-dotted

the shock front is smeared. However, if we were not comparing to a reference solution, our experience with the systems of PDEs might lead us to take this as a good result.

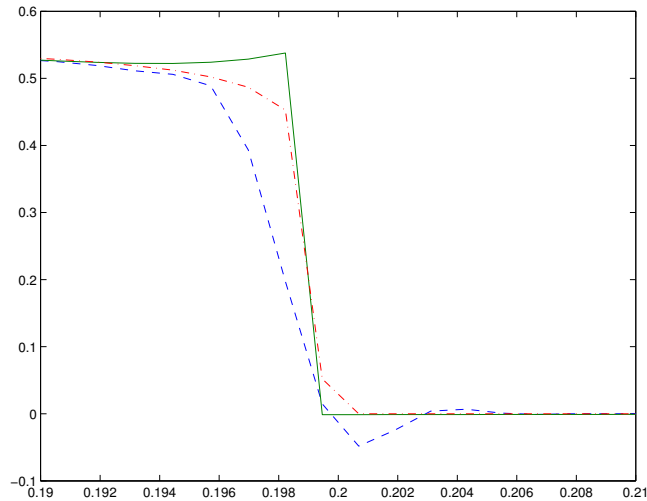


Figure 9.18: SSV, dashed; PP, solid; FD, dash-dotted

A similar result (figure 9.17) is obtained by taking the GRP parameters as $m = 14$ and $\lambda = 2$. Again, similar results to those displayed in figure 9.16 are obtained, but the agreement with the reference solution is not as good.

Finally, the GRP results using $m = 18$ and $\lambda = 9$ are shown in figure 9.18. The postprocessed results are similar to the those in figures 9.15, 9.16, and 9.17, but the overshoot at the shock front would cause us to disregard this result.

9.5 Conclusions

The Chebyshev super spectral viscosity method and Gegenbauer reconstruction procedure were shown to accurately resolve the solution of a nonlinear system of conservation laws with a source term. The problem could be formulated in a split manner in which the source was evaluated exactly in a separate step or in a more general unsplit way which achieved higher accuracy in time. The results of the two formulations are practically identical. Not surprisingly, the spectral method was able to resolve the numerical examples more

accurately and with less grid points than the second order Godunov methods used in [11] and [13].

The solution to the slugging problem has its significant features located near the bottom of the bed ($x = 0$). The standard CGL collocation point distribution (1.1) worked satisfactorily due to its clustering of grid points around $x = 0$, but the clustering of grid points around the boundary at the top of the bed is unnecessary. A more appropriate grid distribution can be achieved via the map (2.10) which clusters collocation points around a specified point in the physical domain.

It is necessary to retain the unphysical, negative particle concentrations throughout the calculation. Setting negative concentrations to zero results in the spectral scheme losing its conservative properties. When the GRP is used to postprocess the final solution, all concentrations became positive.

In order to get results for the slugging problems with the GRP, it was necessary to take a local approach to specifying the reconstruction parameters and to set different values of the parameters m and λ in each smooth subinterval depending on the nature of the solution. This is in contrast to previous applications of the GRP to non-periodic problems in the literature where the parameters were able to be specified by a global approach as $m = k_1\epsilon N$ and $\lambda = k_2\epsilon N$ where k_1 and k_2 are constants, and ϵ is the length of the subinterval. The previous applications of the GRP where the parameters were chosen in this way were to problems with homogeneous features throughout the computational domain: Burgers Equation [25], and the Shallow Water Equations with Riemann initial data [23]. The solution with the most varying detail in which the GRP had been applied to previously was the Euler equations with a Mach 3 shock interacting with an entropy wave. However, the authors did not state the values of the GRP parameters that were used [29]. The need to specify different parameters for each subinterval for varying solutions could make the reconstruction of solutions with varying subintervals of detail in two-dimensions very difficult. Additionally, all previous applications of the GRP had been to functions known on the CGL grid. It was demonstrated that the reconstruction procedure can also be applied to functions known on mapped grids which are often necessary in applications.

Chapter 10

Two-Dimensional Fluidized Bed Equations

In this chapter, a two-dimensional fluidized bed model in the form of a hyperbolic system of conservation laws with a source term (10.1), coupled with an elliptic equation (10.2) for determining a stream function, is solved numerically.

$$w_t + f(w)_x + g(w)_z = b(w, \psi_x, \psi_z) \quad (10.1)$$

$$-(\psi_{xx} + \psi_{zz}) + p(x, z)\psi_x + q(x, z)\psi_z = r(x, z) \quad (10.2)$$

The origins of the model can be found in [19] where a general set of equations modelling dispersed two-phase flow is derived. In [22] a stream function is introduced into the model which corresponds to the total volumetric flux. In the paper [12], the authors state the model for the case of heavy particles dispersed in a gas and with the gas inertia being neglected. It is in this form that we consider the model. A distinguishing feature of the model is that it neglects particle viscosity. Mathematical models of fluidized beds may or may not include a particle viscosity term in an attempt to model the property of the fluidized particles that resists the force tending to cause them to flow. It has been speculated by some authors [48] that particle viscosity, no matter how small, is essential for the behavior corresponding to slugging to occur. However, it has been demonstrated numerically for a one-dimensional model [11, 13] and for the considered two-dimensional model [12], that a model without particle viscosity is capable of reproducing oscillatory slugging behavior.

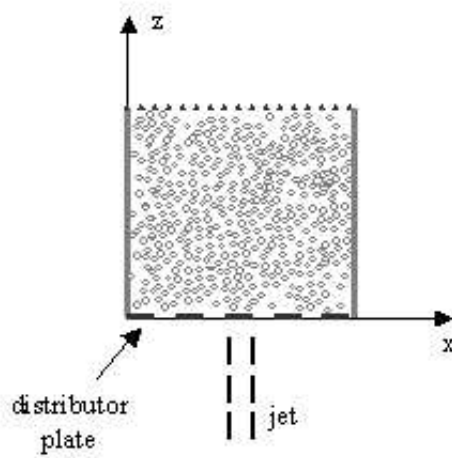


Figure 10.1: 2d fluidized bed

Much of the early numerical work with fluidized bed models produced results, particularly bubble shape, which did not agree with experimental observations (see [60] and references within). Often, the models which were used included particle viscosity. Recently, the particle viscosity free model, was solved numerically by a second order Godunov method which produced a numerical solution which included the physically observed kidney-shaped bubble [12].

Our interest in using the Chebyshev super spectral viscosity method is to see if a realistic bubble shape can be realized while using coarser grids than second order finite difference methods required. Also, it is of interest to see if the spectral method can reveal any small scale structures in the flow that the finite difference methods could not. The standard Chebyshev Collocation method will not converge to the entropy solution [61] since the solution contains shocks, thus, the addition of spectral viscosity will necessary.

10.1 Fluidized Bed Model

Let $\alpha(x, z, t)$ denote the particle concentration, $m(x, z, t) = \alpha u$ the horizontal momentum, $n(x, z, t) = \alpha v$ the vertical momentum, $u(x, z, t)$ the horizontal velocity, and $v(x, z, t)$ the vertical velocity. The variable x describes the variation along the distributor plate at the bottom of the bed and the variable z describes the vertical di-

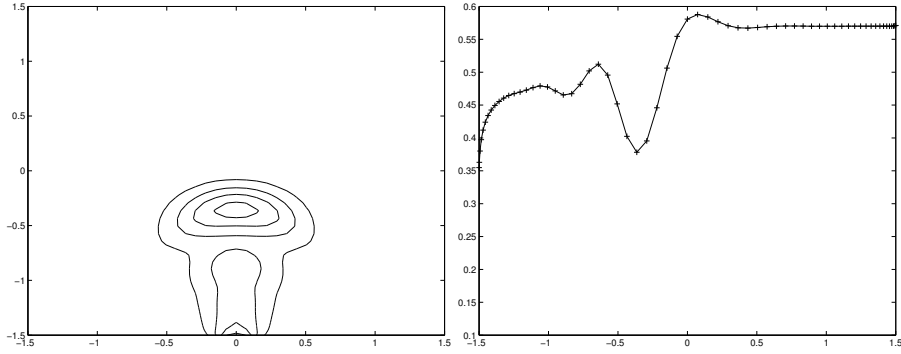
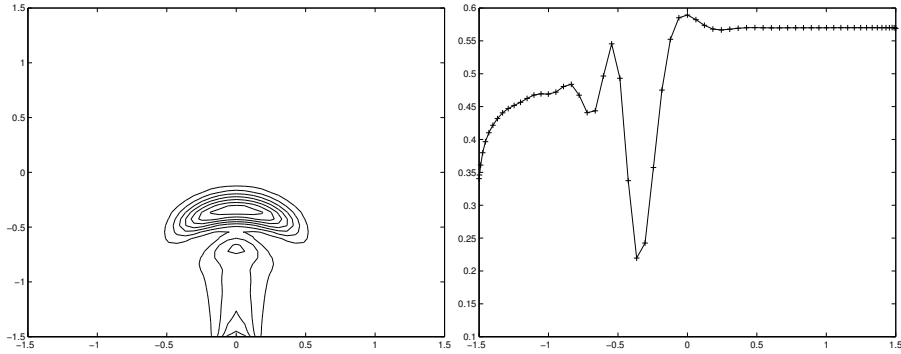


Figure 10.2: CGL grid


 Figure 10.3: mapped grid, $\gamma = 0.86$

rection from the bottom to the top of the bed. The two-dimensional fluidized bed can be described by a system of conservation laws with a source term as

$$\alpha_t + m_x + n_z = 0 \quad (10.3)$$

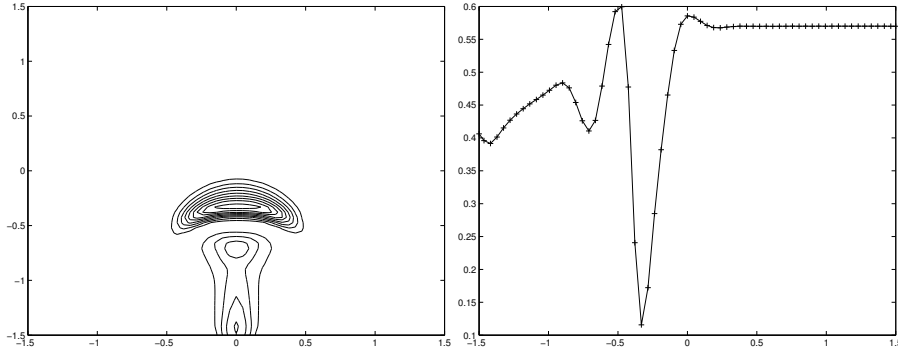
$$m_t + (mu + F(\alpha))_x + (nu)_z = (1 - \alpha)^{-3.5}(\alpha\psi_z - m) \quad (10.4)$$

$$n_t + (mv)_x + (nv + F(\alpha))_z = -(1 - \alpha)^{-3.5}(\alpha\psi_x + n) \quad (10.5)$$

$F(\alpha)$ is specified as

$$F(\alpha) = s^2\alpha + \frac{s^2\alpha_p^2}{\alpha - \alpha_p} + 2s^2\alpha_p \ln(|\alpha - \alpha_p|). \quad (10.6)$$

These equations have been non-dimensionalized using v_t , the terminal velocity of an isolated particle as the velocity scale, and v_t^2/g and v_t/g as the length and time scales, respectively, where g is the acceleration due to gravity [12].


 Figure 10.4: mapped grid, $\gamma = 0.9999$

The parameter α_0 is the concentration of particles at equilibrium and α_p is the packing concentration which sets an upper limit for α where $0 < \alpha < 1$. The parameter α_{0u} denotes the particle concentration corresponding to the critical state dividing linearly stable and unstable states (the particle concentration at minimum fluidization). The constant $s = 3.5(1 - \alpha_{0u})^{2.5}(\alpha_p - \alpha_{0u})$ is related to the linear stability of the equilibrium solutions which correspond to states of uniform fluidization. The stream function $\psi(x, z, t)$ is defined by the elliptic equation of the form (10.2) with the functions $p(x, z)$, $q(x, z)$, and $r(x, z)$ specified as

$$\begin{aligned}
 p(x, z) &= -\frac{\alpha_x}{\alpha} \left(1 + \frac{3.5\alpha}{1 - \alpha}\right) \\
 q(x, z) &= -\frac{\alpha_z}{\alpha} \left(1 + \frac{3.5\alpha}{1 - \alpha}\right) \\
 r(x, z) &= \frac{1}{\alpha} (n_x - m_z + \frac{3.5}{1 - \alpha} [\alpha_x n - \alpha_z m])
 \end{aligned}$$

The scale for the stream function is v_t^3/g .

The computational domain is taken as $(x, z) \in [-x_R, x_R] \times [-z_R, z_R]$. Zero particle momenta in the directions normal to physical boundaries for particles colliding with a wall are applied giving $u = 0$ at $x = -x_R$ and x_R and $v = 0$ at $z = -z_R$. The boundaries for the elliptic equation at $x = \pm x_R$ are streamlines with constant ψ . At the top of the bed, a somewhat artificial boundary is assumed to exist, where the total volumetric flux is taken to be evenly dispersed. At the bottom of the bed (see figure 10.1), a jet of gas of width $2x_b$ is

centrally located at the point $(x = 0, z = -z_R)$ with the background fluidizing gas entering outside of the jet being $j_M = (1 - \alpha_{0u})^{3.5}$. The flux of gas entering through the jet is $j > j_M$ which is specified through the variable α_0 as $j = (1 - \alpha_0)^{3.5}$. The described boundary conditions on ψ can be written as $\psi(-x_R, z, t) = 0$, $\psi(x_R, z, t) = -2x_R j_M + 2x_b(j_M - j)$, $\psi(x, z_R, t) = (-j_M + x_b(j_M - j)/x_R)(x + x_R)$ and

$$\psi(x, -z_R, t) = \begin{cases} -j_M(x + x_R) & -x_R \leq x \leq -x_b \\ -j(x + x_b) - j_M(x_R - x_b) & -x_b < x < x_b \\ -j_M(x + x_R) + 2x_b(j_M - j) & x_b \leq x \leq x_R. \end{cases} \quad (10.7)$$

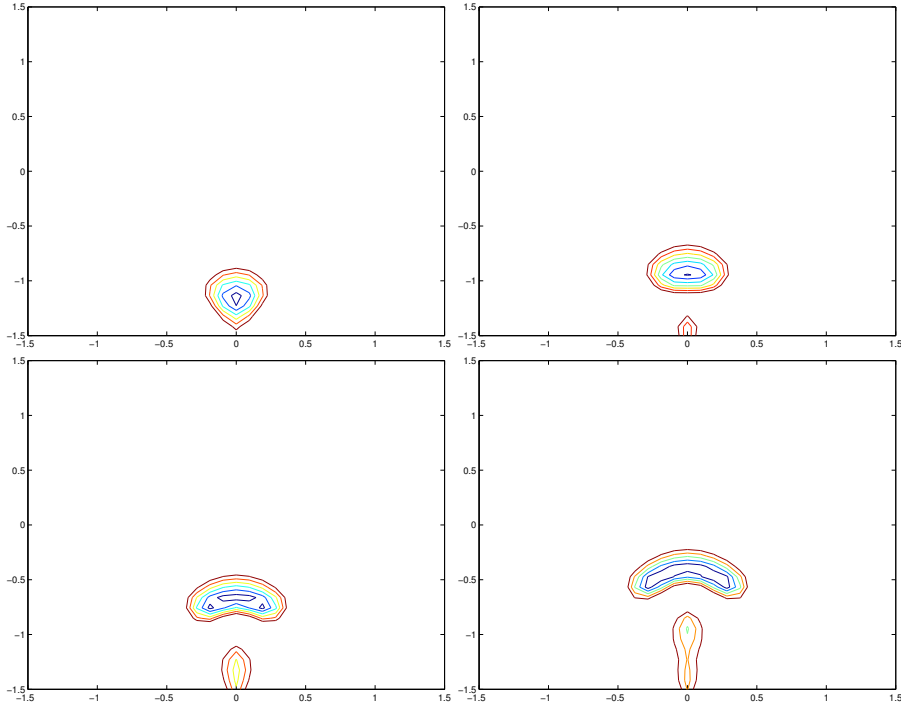
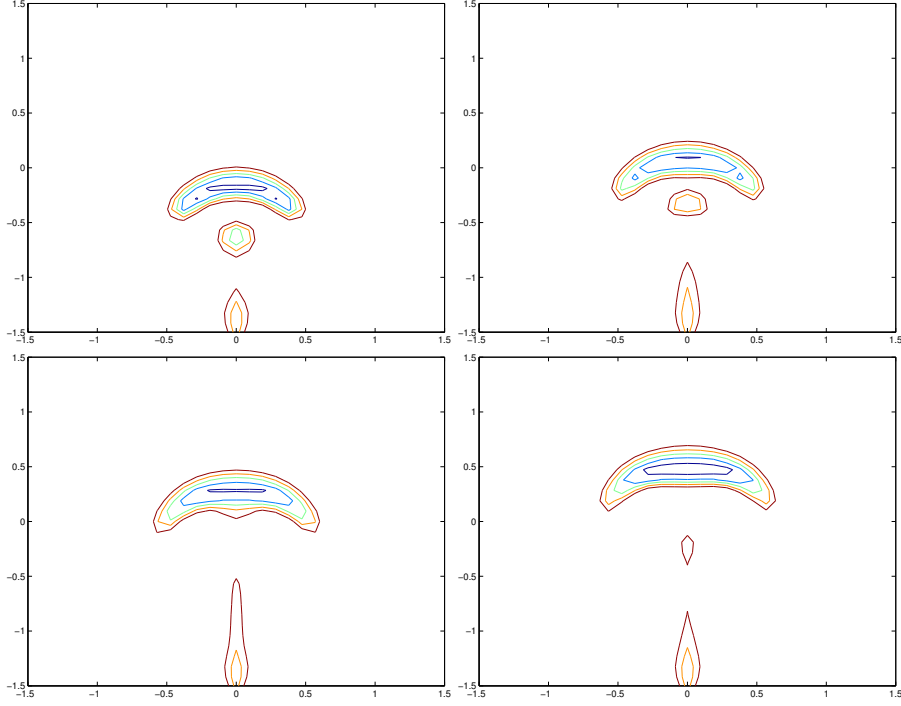


Figure 10.5: 32 by 32 grid, $t = 1.5$ to $t = 3.0$

To extend the Chebyshev SSV method to two-dimensions we have used Strang's second-order splitting [59] to reduce the two-dimensional problem to a sequence of one-dimensional problems. We have also used Strang splitting to separate the contribution of the source term. The splitting is as follows:


 Figure 10.6: 32 by 32 grid, $t = 3.5$ to $t = 5.0$

$$w_t = b(w, \psi_x, \psi_z) \quad (10.8)$$

$$w_t + g(w)_z = 0 \quad (10.9)$$

$$w_t = SSV(\beta, C, N)_z \quad (10.10)$$

$$w_t + f(w)_x = 0 \quad (10.11)$$

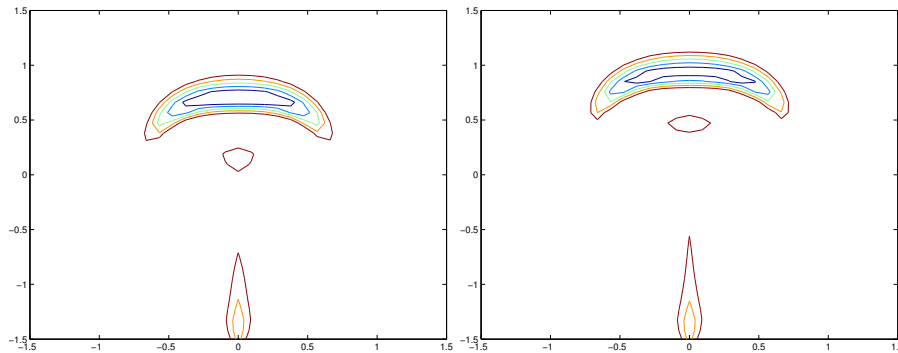
$$w_t = SSV(\beta, C, N)_x \quad (10.12)$$

$$w_t + g(w)_z = 0 \quad (10.13)$$

$$w_t = SSV(\beta, C, N)_z \quad (10.14)$$

$$w_t = b(w, \psi_x, \psi_z) \quad (10.15)$$

Equation (10.11) and (10.12) are solved over a full time step while the other 6 equations are evaluated over a time step of size $\Delta t/2$. The fractional steps involving the source terms, (10.8) and (10.15), may possibly be evaluated in closed form. Otherwise, they may be advanced in time with an ODE integrator. The SSV split steps (10.10), (10.14), and (10.12) can be evaluated exactly as in (4.5).

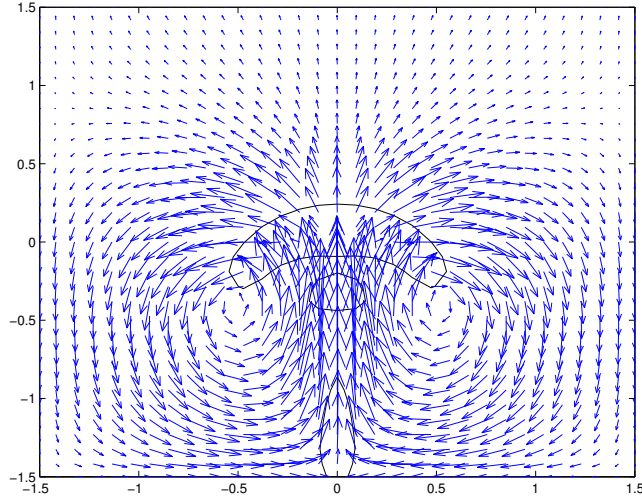
Figure 10.7: 32 by 32 grid, $t = 5.5$ to $t = 6.0$

The remaining equations are advanced in time with a second-order ODE integrator. We have used an explicit second order Runge-Kutta method in the numerical examples. In this formulation of the problem, $g(w)_z$ is evaluated 4 times, $f(w)_x$ 2 times, the source term 2 times, SSV_x is applied once, and SSV_z is applied twice, per time step.

From our experience, the spectral viscosity method can also be implemented successfully in an unsplit, fully two-dimensional formulation, without source term splitting. For a suitably chosen time step, the results of the different problem formulations did not noticeably vary in the numerical examples. However, slightly less spectral viscosity was necessary to obtain a stable approximation with the split formulation than with the unsplit formulation. It is speculated that the incremental way in which the spectral viscosity is applied in the split formulation makes this possible.

10.2 Numerical Results

Since the solution of the equation for the stream may be needed thousands of times during a numerical run, we have not implemented a spectral solution of the elliptic equation. For efficiency we have used a finite difference method. The solution of the elliptic equation for the stream function is based on fitting a parabola to the data at points, x_{i-1} , x_i , and x_{i+1} and then computing the first and second derivatives at x_i . On a uniform grid, the approximation reduces to the standard second order central differences approximation. The

Figure 10.8: 32 by 32 grid velocity field, $t = 4.0$

resulting system of algebraic equations is solved by Gauss-Seidel iteration. The streamfunction ψ appears in terms of its first partial derivatives only in equations (10.8) and (10.15). Therefore, equation (10.2) is solved initially and then immediately before and after solving equation (10.15) at each time step. The derivatives of ψ required in equations (10.8) and (10.15) are found fitting a parabola to the data at the points x_{i-1} , x_i , and x_{i+1} , and then computing the first derivative at x_i . The approximation is second order accurate on any grid.

The fractional steps involving the source term may be evaluated in closed form if the first partial derivatives of the stream function are assumed to remain constant across the fractional steps. By evaluating (10.8) and (10.15) exactly, m and n can be updated as

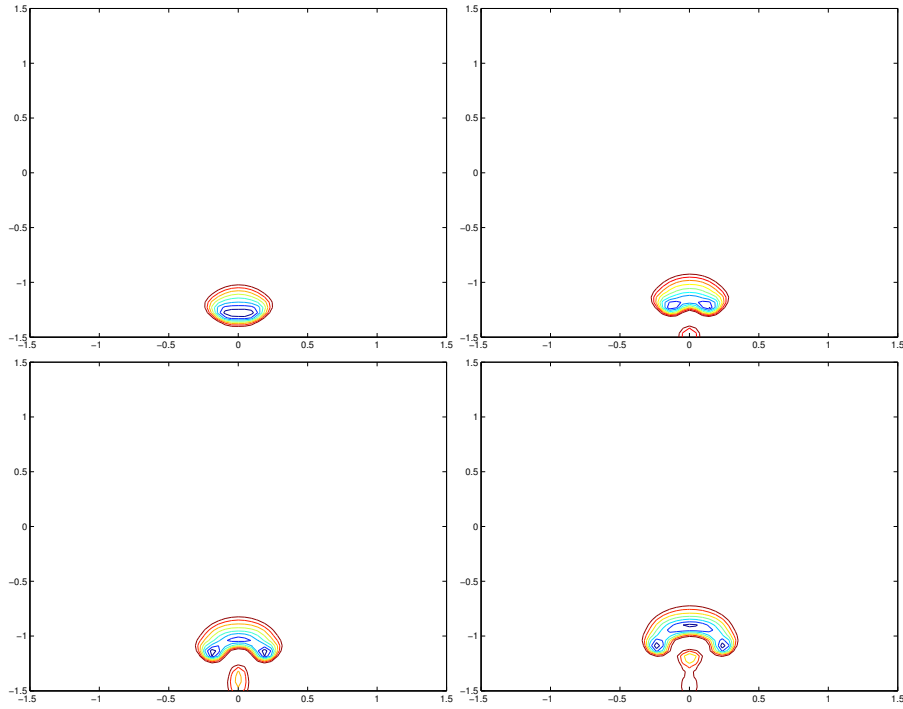
$$m = \alpha\psi_z(1 - E) + mE \quad (10.16)$$

and

$$n = \alpha(1 - \alpha)^{3.5} [1 + (1 - \alpha)^{3.5} \psi_x] (E - 1) + En \quad (10.17)$$

where

$$E = \exp \left[\frac{-\Delta t}{2(1 - \alpha)^{3.5}} \right]. \quad (10.18)$$

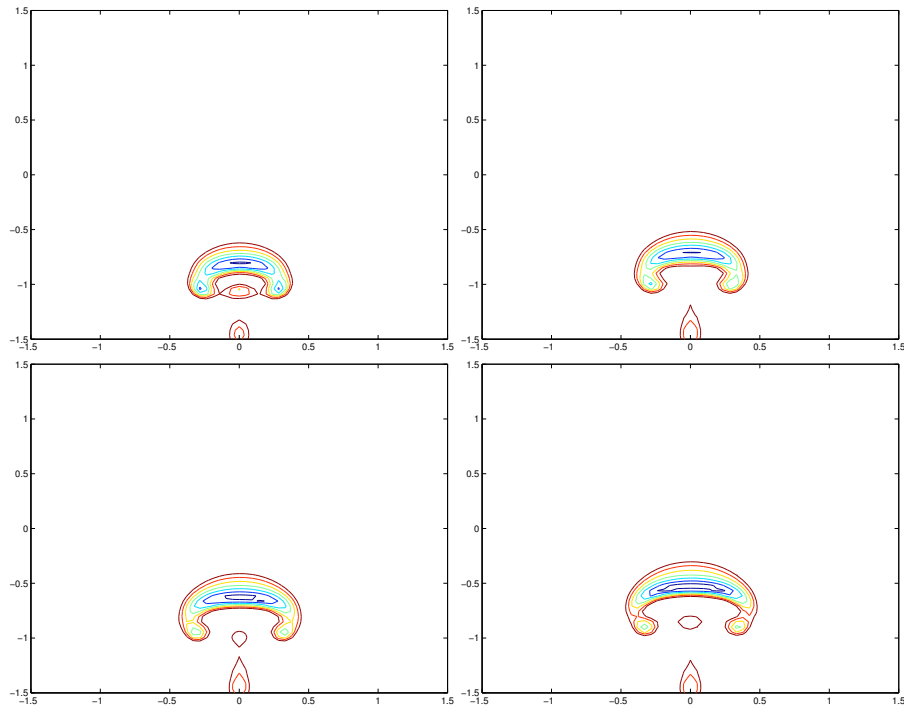
Figure 10.9: 64 by 64 grid, $t = 1.25$ to $t = 2.0$

A fluidized bed of height and width 3 ($z_R = x_R = 1.5$) units is considered. The initial concentration of particles is taken as $\alpha = \alpha_{0u} = 0.57$ and the initial velocities are $u = v = 0$. Equation (10.2) is solved to find the initial value of ψ . At time $t > 0$, a centered jet of gas with a total width of 0.2 units ($x_b = 0.1$), enters from the bottom of the bed.

10.2.1 Choice of Collocation Grid

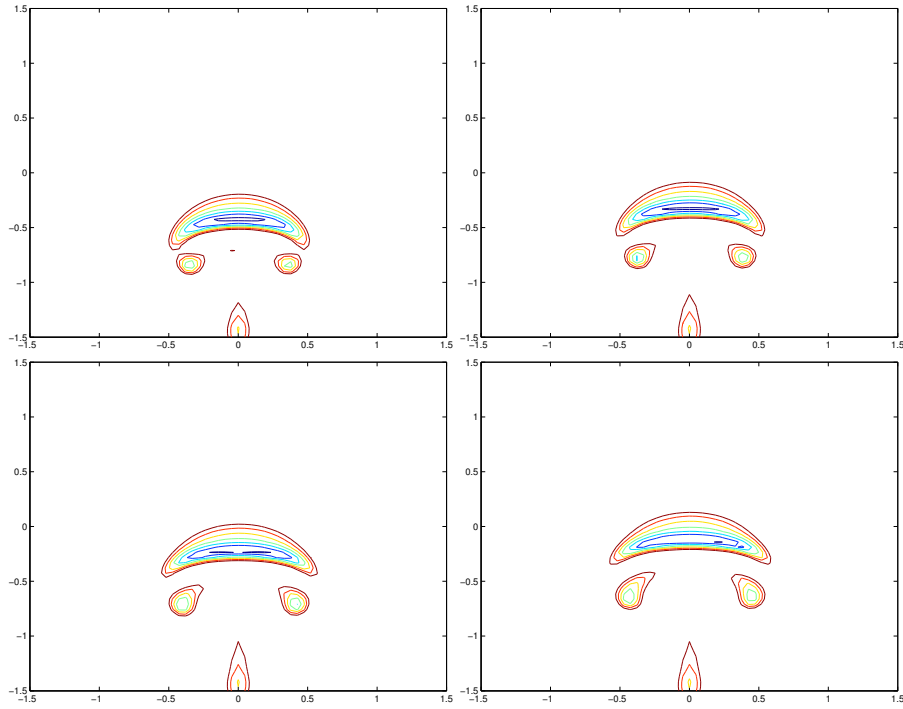
In the first numerical experiment, the gas inflow is specified by setting $\alpha_0 = 0.35$. A 64 by 64 grid is used and the distribution of collocation points is specified three different ways. Three different solutions are obtained, each with a different computational grid. The contours ($\alpha = 0$ to $\alpha = 0.6$) and center line ($x = 0$) plots are compared at time $t = 3.0$. The goal is to determine which grid best resolves the solution.

The first run uses the CGL grid (1.1) which clusters points densely around the boundaries and provides poor interior resolution. In or-

Figure 10.10: 64 by 64 grid, $t = 2.25$ to $t = 3.0$

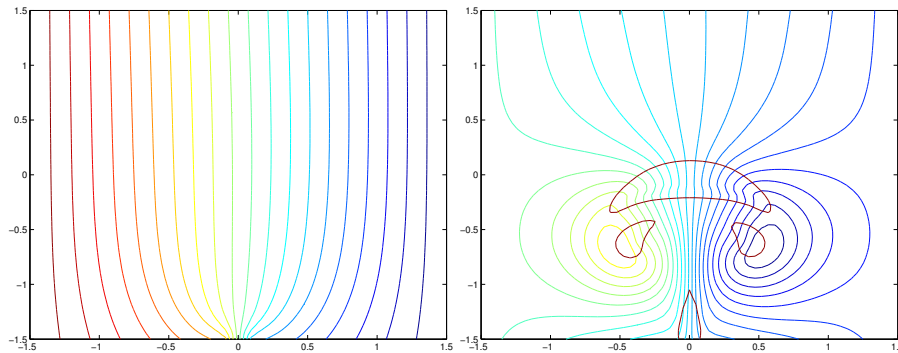
der to obtain stable results with the explicit time stepping, it was necessary to take $\Delta t = 0.000025$ and take the SSV parameters as $C = 6$ and $s = 2$. The small time step is typical due to the $O(N^{-2})$ stable time step restriction imposed by the CGL grid. The lack of resolution towards the interior of the domain is apparent from the wide spread contour lines and the center line plot (figure 10.2).

To relax the $O(N^{-2})$ time stepping restriction, a mapped grid specified by the map (2.7) can be used. By taking the map parameter to be $\gamma = 0.86$ in both the x and z directions we end up with a grid with less clustering around the boundaries and with better interior resolution. This setting of the map parameter is theoretically the upper limit of the parameter range that can be used with $N = 64$ in order to maintain a spectral convergence rate [42]. Taking $\Delta t = 0.0001$, and $C = 3$ and $s = 2$, as the SSV parameters, produces stable results. A marked improvement in results can be observed (figure 10.3) when compared with the CGL grid results. Increasing the mapping parameter closer to one produces even bet-

Figure 10.11: 64 by 64 grid, $t = 3.25$ to $t = 4.0$

ter results.

The third run again uses map (2.7) to specify the grid. The map parameter was chosen as $\gamma = 0.9999$ in both the x and z directions. Choosing the map parameter so close to one results in a near uniform grid. Taking $\Delta t = 0.0005$, and $C = 3$ and $s = 2$ as the SSV parameters produces stable results. Even though taking γ so large could introduce a mapping error and theoretically sacrifice the spectral convergence rate of the method, this is not an issue in this case as we are implementing a mixed spectral/finite difference method in which the the overall accuracy of the solution will not be spectral. The increased resolution in the interior provided by the near uniform grid is evident in figure 10.4. It is concluded that this is the grid that best resolves the problem. The grid allows for the largest stable explicit time step and values of the SSV parameters which result in the smallest amount of spectral viscosity being applied. Compared with the CGL grid results from the first run, the third run used a time step 20 times larger and a spectral viscosity

Figure 10.12: 64 by 64 grid stream function, $t = 0.25$ and $t = 4.0$

that was only half as strong.

10.2.2 32 by 32 Grid

The map (2.7) is used to form the grid with $\gamma = 0.9999$ in both the x and z directions. The map produces a near uniform grid and allows for good resolution in the center of the domain as well as permitting a relatively large stable time step to be taken. The parameter α_0 is set to $\alpha_0 = 0.2$ which corresponds to a strong gas inflow. The flow is stronger than that which is required to maintain a state of minimum fluidization and slugging in the bed is expected. In figures 10.5 through 10.7, contour plots show α ranging from 0.05 to 0.4 in 0.05 unit increments from time $t = 1.5$ to $t = 6.0$. The setup for this problem is similar to experiments run in [12] where the numerical solution was by Roe's method which required a 100 by 100 or a 150 by 150 grid to resolve the flow.

The spectral method resolves the flow well on this very coarse grid and a physically correct bubble shape is obtained. The simulation exhibits other features observed in fluidized beds such as coalescence, which can be observed in the α plots in figure 10.6 from $t = 3.5$ to $t = 5.0$, as a smaller bubble catches up to and is absorbed by the main bubble. The SSV parameters in both the x and z directions were $s = 2$ and $C = 1$, which applies only a very weak high pass filter to the spectral solution.

The fact that the flow is well resolved on the 32 by 32 grid indicates that the second order finite difference approximation of the stream function is adequate. It seems as if the accuracy in which

the flux derivatives are evaluated in the system of conservation laws is the most important factor in obtaining a resolved solution.

The 32 by 32 spectral solution produced solutions of similar quality as the finite difference solutions in [12], but at a fraction of the computational effort, and used significantly less storage space.

In figure 10.8, counter-rotating convective rolls behind the main bubble are very evident in the velocity field of the coarse grid SSV solution.

10.2.3 64 by 64 Grid

The same experiment that was run in the previous section with a 32 by 32 grid is repeated. The results (figures 10.9 through 10.11) on the finer grid are similar to the results obtained on the coarse grid, but some small scale details in the flow were revealed that were not present in the coarse grid spectral solution or in the finite difference solutions in [12]. The SSV parameters in both the x and z directions were $s = 2$ and $C = 3$.

In figure 10.9, the particle concentration contours at $t = 1.25$ are pictured when the bubble entered the bed. A smaller disturbance below the main bubble is noticeable by $t = 1.5$ (figure 10.9) which coalesces with the main bubble by $t = 2.5$ (figure 10.10) at which time the formation of two small satellite bubbles is noticeable. By time $t = 3.25$ (figure 10.11) the main bubble has shed the two satellite bubbles.

10.3 Conclusions

A mixed Chebyshev SSV/finite difference method has been implemented for the fluidized bed model. The flux derivatives in the conservation laws are evaluated with spectral accuracy while the elliptic equation for the stream function is solved by a second order finite difference method. The approximation is second order accurate in time. The SSV method has been shown to produce quality numerical solutions of a complicated multiphase flow problem. The method was able to use fewer degrees of freedom and still resolve fine scale features in the solution better than in previous reported results in [12] using Roe's method with a Superbee flux limiter. A grid which was nearly uniform best resolved the problem, allowed

the smallest amount of spectral viscosity to be applied, and allowed the largest stable time step to be used.

Although the development of bubbles in fluidized beds has been shown to mathematically correspond to the development of shocks, spurious oscillations are not visibly evident in the numerical solution. This is in contrast to the Chebyshev SSV solution of the one-dimensional fluidized bed model where a postprocessing method was used to remove the effects of the Gibbs phenomenon from the approximation. The shocks seem much weaker in the two-dimensional model and the mild filtering of the SSV methods seems to keep any spurious oscillations under control. If an oscillatory solution was obtained, postprocessing could be attempted using the GRP or other postprocessing method (see Appendix C). However, theoretically, the results of their application to the two-dimensional fluidized bed solution would be less certain due to the second order finite difference solution of the equation for the stream.

Part V

Summary

Chapter 11

Conclusions

11.1 Summary of Results

- Functions with homogeneous features in each smooth subinterval may be postprocessed by globally specifying the GRP parameters. Functions with varying subintervals of detail may require that the GRP parameters be specified locally.
- The main drawback of the Gegenbauer reconstruction procedure is that there is no way to specify the parameters in advance. A method to optimize the parameters has not been developed. So while we can always select GRP parameters which result in the postprocessed solution comparing favorably with an exact or reference solution, the GRP is not ready to be used as a "black box" postprocessing method until a way to choose the optimal parameters in advance is available. Work is under way on choosing optimal reconstruction parameters and results will be reported in a future paper. When this is complete, it is hoped that the method will be able to be implemented as a "black box" postprocessing method for Spectral Collocation methods for the solution of Partial Differential Equations.
- Numerical Evidence supports using the SSV method and GRP for solving and postprocessing systems of conservation laws with source terms. The SSV method may be implemented in a split or unsplit method. Slightly less spectral viscosity was need to stabilize problems implemented in split formulations.
- Numerical evidence indicates that the GRP may be applied to

functions known on mapped CGL grids.

- The spatially varying viscosity term of the Chebyshev SSV method, which goes to zero at the boundaries, did not prevent the method from being used on problems with shocks located near boundaries. However, a slightly stronger viscosity term was necessary in this case.
- Edge detection and postprocessing software [55] was made publicly available. The algorithms are time consuming to code and the software may be used as a starting point for other researchers in the field.
- All numerical examples have been postprocessed using the Gegenbauer Reconstruction Procedure. One of our goals was to examine if the GRP, which has shown great promise on some simple examples, could be used to successfully postprocess PDE solutions with varying subintervals of detail. The numerical examples show that while using the GRP to postprocess such problems is possible, that it is necessary to use a local approach to specify the reconstruction parameters m and λ . Even then, since there is not yet a method for choose optimal reconstruction parameters, it is possible to choose several different combinations of the GRP parameters which might result in different, but "reasonable" postprocessed solutions. The conclusion is that until there is a way to choose the optimal parameters in advance, that the GRP will not be a very robust method on such problems. We can always choose the reconstruction parameters so that the postprocessed solution compares favorably with an exact or reference solution, but without such comparisons, the results of the postprocessing are less certain.

It should be noted that other, more robust postprocessing methods (see Appendix C) exist which are able to recover spectral accuracy up to within a neighborhood of a discontinuity. Some of the methods may be used without knowing the location of the discontinuities.

11.2 Open Questions and Further Work

- Optimal selection of the Gegenbauer reconstruction parameters, λ and m .
- The Gegenbauer basis has been shown to be a Gibbs complementary basis for the Chebyshev basis [36]. However, it has not been shown to be the optimal Gibbs complementary basis in this case. It is possible that the Chebyshev approximations could be projected onto another Gibbs complementary basis with optimal postprocessing being the result.
- The Spectral Signal Processing Suite will be updated periodically to reflect improvements to the algorithms. Implementing the methods for two-dimensional functions is planned for a future release. The implementation of the methods for the edge detection and reconstruction procedures for periodic functions approximated by Fourier methods is also planned. Additionally, the Spectral Mollification and Padé-based algorithms will be implemented so that comparisons among the available post-processing methods may more easily be made.

Appendix A

Chebyshev Polynomials

The Chebyshev Polynomials, $T_k(x)$, are orthogonal under the weight function $(1 - x^2)^{-\frac{1}{2}}$ with

$$\int_{-1}^1 \frac{T_k(x)T_n(x)}{\sqrt{1-x^2}} dx = \delta_{k,n} c_k \frac{\pi}{2}$$

where

$$c_k = \begin{cases} 2 & k = 0 \\ 1 & k \geq 1 \end{cases} .$$

Three Term Recurrence Formula

$$\begin{aligned} T_0(x) &= 1 \\ T_1(x) &= x \\ T_k(x) &= 2xT_{k-1}(x) - T_{k-2}(x) \end{aligned}$$

Differentiation Formula

$$\frac{d}{dx} T_k(x) = \frac{k}{1-x^2} [T_{k-1}(x) - xT_k(x)]$$

Christoffel-Darboux Formula

$$\sum_{k=0}^p T_k(x)T_k(y) = \frac{k_p}{k_{p+1}h_p} \frac{T_{p+1}(x)T_p(y) - T_p(x)T_{p+1}(y)}{x-y}$$

where

$$k_p = 2^{p-1}$$

and

$$\begin{aligned}h_0 &= \pi \\h_n &= \frac{\pi}{2}\end{aligned}$$

Christoffel-Darboux Formula for $x = y$

$$\sum_{k=0}^p T_k(x)T_k(x) = \frac{k_p}{k_{p+1}h_p} [T_{p+1}(x)\frac{d}{dx}T_p(x) - T_p(x)\frac{d}{dx}T_{p+1}(x)]$$

where

$$\frac{d}{dx}T_k(x) = \frac{k \sin(k \arccos(x))}{\sqrt{1-x^2}}$$

Relationship to Gegenbauer Polynomials

$$T_k(x) = \frac{k}{2}C_k^0(x)$$

Appendix B

Gegenbauer Polynomials

The Gegenbauer Polynomials, $C_n^\lambda(x)$, $\lambda \geq 0$, are orthogonal under the weight function $(1 - x^2)^{\lambda - \frac{1}{2}}$ with

$$\int_{-1}^1 (1 - x^2)^{\lambda - \frac{1}{2}} C_k^\lambda(x) C_n^\lambda(x) dx = \delta_{k,n} h_n^\lambda$$

and

$$h_n^\lambda = \pi^{\frac{1}{2}} C_n^\lambda(1) \frac{\Gamma(\lambda + \frac{1}{2})}{\Gamma(\lambda)(n + \lambda)}$$

with

$$C_n^\lambda(1) = \frac{\Gamma(n + 2\lambda)}{n! \Gamma(2\lambda)}$$

Three Term Recurrence formula

$$C_0^\lambda(x) = 1$$

$$C_1^\lambda(x) = 2\lambda x$$

$$C_m^\lambda(x) = \frac{1}{m} [2(m - \lambda - 1)x C_{m-1}^\lambda(x) - (m + 2\lambda - 2) C_{m-2}^\lambda(x)]$$

The Gegenbauer polynomials achieve their maximum at the boundary

$$|C_m^\lambda(x)| \leq C_m^\lambda(1)$$

Christoffel-Darboux Formula

$$\sum_{l=0}^m \frac{C_m^\lambda(x) C_m^\lambda(y)}{h_l^\lambda} = \frac{k_m}{k_{m+1} h_m^\lambda} \frac{C_{m+1}^\lambda(x) C_m^\lambda(y) - C_m^\lambda(x) C_{m+1}^\lambda(y)}{x - y}$$

with

$$k_m = \frac{2^m \Gamma(\lambda + m)}{m! \Gamma(\lambda)}$$

When $x = y$, use L'Hospital's rule and the equality $2\lambda C_{m-1}^{\lambda+1}(x) = \frac{d}{d\xi} C_m^\lambda(x)$ and the Christoffel-Darboux Formula becomes

$$\sum_{l=0}^m \frac{C_m^\lambda(x) C_m^\lambda(x)}{h_l^\lambda} = \frac{k_m}{k_{m+1} h_m^\lambda} 2\lambda (C_m^{\lambda+1}(x) C_m^\lambda(x) - C_{m+1}^\lambda(x) C_{m+1}^{\lambda+1}(x))$$

Appendix C

Other Postprocessing methods

In this work, all numerical examples have been postprocessed using the Gegenbauer Reconstruction Procedure. One of our goals was to examine if the GRP, which has shown great promise on some simple examples, could be used to successfully postprocess PDE solutions which were more detailed than piecewise linear or solutions which contained varying subintervals of detail. The numerical examples show that while using the GRP to postprocess such problems is possible, that it is necessary to use a local approach to specify the reconstruction parameters m and λ . Even then, since there is not yet a method to choose optimal reconstruction parameters, it is possible to choose several different combinations of the GRP parameters which might result in different, but "reasonable" postprocessed solutions. The conclusion is that until there is a way to choose the optimal parameters in advance, that the GRP will not be a very robust method on such problems. We can always choose the reconstruction parameters so that the postprocessed solution compares favorably with an exact or reference solution, but without such comparisons, the results of the postprocessing are less certain.

The GRP is capable of recovering spectral accuracy at every point, even at the locations of the discontinuities. However, more robust postprocessing methods exist which are able to recover spectral accuracy at every x which is not too close to a discontinuity, but which can not recover spectral accuracy in the neighborhood of the discontinuity. Some of the methods may be used without knowing

the location of the discontinuities, but may produce a more accurate postprocessing if the locations of the discontinuities are known.

The first class of postprocessing methods consists of variations of the spectral mollification idea which was originally developed in [37]. Spectral mollification involves applying a two parameter family of filters. The two parameters are somewhat problem dependent, but are much easier to select than the GRP parameters. The method can recover spectral accuracy up to within a neighborhood of each discontinuity. The Gibbs phenomenon can be removed, but some smearing at the discontinuity locations will occur. This idea is discussed in more detail in [41] and examples of using the method to postprocess PDE solutions are contained in [40].

The method of [37] can be improved upon if the locations of the edges are known [24, 62]. This allows one of the two parameters to be optimized which leads to increased accuracy away from the discontinuities and less smearing at the discontinuities. Further optimization of the method is considered in [62].

A Padé-based algorithm for removing the Gibbs phenomenon from Fourier approximations is developed in [20]. The authors present an example in which the Padé-based algorithm exhibits a spectral convergence rate, but the GRP does not. This is despite the fact that theoretically, the GRP should produce a spectral rate of convergence. However, the authors did not state the values of the GRP parameters that were used in the example.

Appendix D

Software

Before spectral methods and postprocessing can be implemented, there is a substantial amount of software, in comparison with finite difference methods, which must be in place. It is not necessary to start from scratch as several free open source libraries are available.

D.1 Environments

Simulation Environment for Numerical Ordinary and Partial Equations, SENOPDE [54], provides both a web based and application based object oriented programming environment in the Java language. SENOPDE contains a large number of standard test problems which may be extended by the user. Two and three-dimensional graphics provide both animated and static plots. Several numerical libraries exist for use in SENOPDE and more are under development.

D.2 Spectral

Freely available packages for implementing spectral methods:

- Matlab Differentiation Suite [64]
- Pseudopack 2000 [14].
- JSpecLib [53], a Java library for implementing Spectral Methods. Designed for use with SENOPDE.

D.3 Postprocessing

The only available software for edge detection and postprocessing of spectral data is the Spectral Signal Processing Suite [55]. Many examples are provided.

D.4 Source Code in print

Source code in print, in FORTRAN, for fast transforms, matrix differentiation, etc. may be found in [21] and [7].

Bibliography

- [1] Ø. Andreassen, I. Lie, and C. E. Wasberg. The spectral viscosity method applied to simulation of waves in a stratified atmosphere. *Journal of Computational Physics*, 110:257–273, 1994.
- [2] J. Augenbaum. An adaptive pseudospectral method for discontinuous problems. *Applied Numerical Mathematics*, 5:459–480, 1989.
- [3] C. Basdevant, M. Deville, P. Haldenvang, J.M. Lacroix, D. Orlando, A. Patera, R. Peyret, and J. Quazzani. Spectral and finite difference solutions of Burgers’ equation. *Computer and Fluids*, 14(1):23–41, 1986.
- [4] A. Bayliss and E. Turkel. Mappings and accuracy for Chebyshev pseudospectral approximations. *Journal of Computational Physics*, 101:349–359, 1992.
- [5] John P. Boyd. Two comments on filtering (artificial viscosity) for Chebyshev and Legendre spectral and spectral element methods: Preserving boundary conditions and interpretation of the filter as a diffusion. *Journal of Computational Physics*, 143:283–288, 1998.
- [6] John P. Boyd. *Chebyshev and Fourier Spectral Methods*. Dover Publications, Inc, New York, second edition, 2000.
- [7] Claudio Canuto, M. Y. Hussaini, Alfio Quarteroni, and Thomas A. Zang. *Spectral Methods for Fluid Dynamics*. Springer-Verlag, New York, 1988.
- [8] G.F. Carey and M. Tsai. Hyperbolic heat transfer with reflection. *Numerical Heat Transfer*, 5, 1982.

- [9] M. Carpenter, D. Gottlieb, and C. Shu. On the Conservation and Convergence to weak solutions of Global Schemes. *ICASE report* - www.icase.edu, 2001.
- [10] G. Q. Chen, Q. Du, and E. Tadmor. Spectral viscosity approximations to multidimensional scalar conservation laws. *Mathematics of Computation*, 61:629–643, 1993.
- [11] I. Christie, G.H. Ganser, and J.M. Sanz-Serna. Numerical solution of a hyperbolic system of conservation laws with source term arising in a fluidized bed model. *Journal of Computational Physics*, 93(2):297–311, 1991.
- [12] I. Christie, G.H. Ganser, and J.W. Wilder. Numerical solution of a two-dimensional fluidized bed model. *International Journal for Numerical Methods in Fluids*, 28:381–394, 1998.
- [13] I. Christie and C. Palencia. An exact Riemann solver for a fluidized bed model. *IMA Journal of Numerical Analysis*, 11:493–508, 1991.
- [14] B. Costa and W.S. Don. Pseudopack. <http://www.cfm.brown.edu/people/wsdon/home.html>, 2000.
- [15] P. J. Davis. *Interpolation and Approximation*. Dover Publications, New York, 1975.
- [16] W. S. Don. Numerical study of pseudospectral methods in shock wave applications. *Journal of Computational Physics*, 110:103–111, 1994.
- [17] W. S. Don and C. Quillen. Numerical simulation of shock-cylinder interactions. *Journal of Computational Physics*, 122:244–265, 1995.
- [18] W. S. Don and A. Solomonoff. Accuracy and speed in computing the Chebyshev collocation derivative. *SIAM Journal of Scientific Computing*, 16(6):1253–1268, 1995.
- [19] D. A. Drew. Mathematical modeling of two-phase flow. *Ann. Rev. Fluid. Mech.*, 15:261–291, 1983.
- [20] T. Driscoll and B. Fornberg. A Padé-based algorithm for overcoming the Gibbs phenomenon. *Numerical Algorithms*, 2000.

- [21] Bengt Fornberg. *A Practical Guide to Pseudospectral Methods*. Cambridge University Press, New York, 1996.
- [22] G. Ganser, X. Hu, and D. Li. Solutions for a two-dimensional hyperbolic-elliptic coupled system. *SIAM Journal of Mathematical Analysis*, 27(4):1024–1037, 1996.
- [23] Anne Gelb. A hybrid approach to spectral reconstruction of piecewise smooth functions. *Preprint*, 2000.
- [24] Anne Gelb and James Gleeson. Enhanced spectral viscosity method for shallow water equations. *Preprint*, 2000.
- [25] Anne Gelb and Eitan Tadmor. Enhanced spectral viscosity approximations for conservation laws. *Applied Numerical Mathematics*, 33:3–21, 2000.
- [26] Anne Gelb and Eitan Tadmor. Detection of edges in spectral data II: Nonlinear enhancement. *SIAM Journal of Numerical Analysis - To appear*, 2001.
- [27] D.E. Glass, M.N. Ozisik, and D.S. McRae. Hyperbolic heat conduction with radiation in an absorbing and emitting medium. *Numerical Heat Transfer*, 12:321–333, 1987.
- [28] D.E. Glass, M.N. Ozisik, D.S. McRae, and Brain Vick. On the numerical solution of hyperbolic heat conduction. *Numerical Heat Transfer*, 8:497–504, 1985.
- [29] D. Gottlieb and J.S. Hestaven. Spectral methods for hyperbolic problems. *Journal of Computational and Applied Mathematics*, 128:83–131, 2001.
- [30] D. Gottlieb and Eitan Tadmor. Recovering pointwise values of discontinuous data within spectral accuracy. In *Progress in Scientific Computing*, number 6 in Proceedings of U.S.-Israel Workshop, 1984, pages 357–375. Birkhauser Boston, 1985.
- [31] David Gottlieb and Steven A. Orszag. *Numerical Analysis of Spectral Methods*. SIAM, Philadelphia, PA, 1977.
- [32] David Gottlieb, C.-W. Shu, A. Solomonoff, and H. Vandeven. On the Gibbs phenomenon I: recovering exponential accuracy from the Fourier partial sum of a nonperiodic analytic function.

- Journal of Computational and Applied Mathematics*, 43:81–98, 1992.
- [33] David Gottlieb and Chi-Wang Shu. On the Gibbs phenomenon IV: Recovering exponential accuracy in a subinterval from a Gegenbauer partial sum of a piecewise analytic function. *Mathematics of Computation*, 64:1081–1095, 1995.
- [34] David Gottlieb and Chi-Wang Shu. On the Gibbs phenomenon V: recovering exponential accuracy from collocation point values of a piecewise analytic function. *Numerische Mathematik*, 71:511–526, 1995.
- [35] David Gottlieb and Chi-Wang Shu. On the Gibbs phenomenon III: Recovering exponential accuracy in a subinterval from a partial sum of a piecewise analytic function. *SIAM Journal of Numerical Analysis*, 33:280–290, 1996.
- [36] David Gottlieb and Chi-Wang Shu. On the Gibbs phenomenon and its resolution. *SIAM Review*, 39(4):644–668, 1997.
- [37] David Gottlieb and E. Tadmor. Recovering pointwise values of discontinuous data within spectral accuracy. In E. M. Murman and S. S. Abarbanel, editors, *Progress and Supercomputing in Computational Fluid Dynamics*, pages 357–375. Birkhäuser, Boston, 1985.
- [38] Chris Guenther. *Pseudospectral Techniques for Non-Smooth Evolutionary Problems*. PhD thesis, West Virginia University, 1998.
- [39] J. Hesthaven, P. Dinesen, and J. Lynov. Spectral collocation time-domain modelling of diffractive optical elements. *Journal of Computational Physics*, 155:287–306, 1999.
- [40] O. Kaber. A Legendre pseudospectral viscosity method. *Journal of Computational Physics*, 128:165–180, 1996.
- [41] O. Kaber and Sidi Mahmoud. Filtering non-periodic function. *Computer Methods in Applied Mechanics and Engineering*, 116:123–130, 1994.
- [42] R. Kosloff and H. Tal-Ezer. A modified Chebyshev pseudospectral method with an $O(1/n)$ time step restriction. *Journal of Computational Physics*, 104:457–469, 1993.

- [43] P.D. Lax. Accuracy and resolution in the computation of solutions of linear and nonlinear equations. In *In Proc. of Recent Advances in Numerical Analysis, Univ. Wisconsin*, pages 107–117. Academic Press, 1978.
- [44] B. P. Leonard. The ULTIMATE conservative difference scheme applied to unsteady one-dimensional advection. *Computer Methods in Applied Mechanics and Engineering*, 88:17–74, 1991.
- [45] H. Ma. Chebyshev-Legendre super spectral viscosity method for nonlinear conservation laws. *SIAM J. Numer. Anal.*, 35:893–908, 1998.
- [46] B. E. McDonald. Flux-corrected pseudospectral method for scalar hyperbolic conservation laws. *Journal of Computational Physics*, 82:413–428, 1989.
- [47] J. Mead and R. Renaut. Accuracy, resolution, and stability properties of a modified Chebyshev method. *Journal of Computational Physics*, 4:71–118, 2000.
- [48] D. Needham and J. Merkin. The propagation of a voidage disturbance in a uniformly fluidized bed. *Journal of Fluid Mechanics*, 131:427–454, 1983.
- [49] H. Nessyahu and E. Tadmor. Non-oscillatory central differencing for hyperbolic conservation laws. *Journal of Computational Physics*, 87:408–463, 1990.
- [50] M.N. Ozisik and Brian Vick. Propagation and reflection of thermal waves in a finite medium. *International Journal of Heat and Mass Transfer*, 27(10):1845–1854, 1984.
- [51] William H. Press, B. H. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, New York, second edition, 1992.
- [52] P.L. Roe. Approximate Riemann solvers, parameter vectors, and difference schemes. *Journal of Computational Physics*, 43:357–372, 1981.
- [53] Scott A. Sarra. jSpecLib: a Java library for implementing Pseudospectral Methods. www.scottsarra.org/senopde/jSpecLib/speclib.html, 2002.

- [54] Scott A. Sarra. SENOPDE (Simulation Environment for Numerical Ordinary and Partial Differential Equations). www.scottsarra.org/senopde/senopde.html, 2002.
- [55] Scott A. Sarra. Spectral Signal Processing. *submitted: ACM Transactions on Mathematical Software*, 2002. www.scottsarra.org/signal/signal.html.
- [56] C.W. Shu and S. Osher. Efficient implementation of essentially nonoscillatory shock-capturing schemes, II. *Journal of Computational Physics*, 83:32–78, 1989.
- [57] A. Solomonoff. A fast algorithm for spectral differentiation. *Journal of Computational Physics*, 98:174–177, 1992.
- [58] A. Solomonoff and E. Turkel. On global collocation. *Journal of Computational Physics*, 81:239–276, 1989.
- [59] Gilbert Strang. On the construction and comparison of difference schemes. *SIAM Journal of Numerical Analysis*, 5(3):506–517, 1968.
- [60] Madhava Syamlal. High order discretization methods for the numerical simulation of fluidized beds. *AICHE Symposium Series*, 94(318), 1998.
- [61] E. Tadmor. Convergence of spectral methods for nonlinear conservation laws. *SIAM Journal of Numerical Analysis*, 26:30–44, 1989.
- [62] E. Tadmor and J. Tanner. Adaptive mollifiers - high resolution recovery of piecewise smooth data from its spectral information. *Preprint*, March 2001.
- [63] H. Vandeven. Family of spectral filters for discontinuous problems. *SIAM Journal of Scientific Computing*, 6:159–192, 1991.
- [64] J.A.C. Weideman and S. Reddy. A MATLAB differentiation matrix suite. *to appear: ACM Transactions on Mathematical Software*, 2001.