AFRL-VA-WP-TR-1998-3060

HIGH-ORDER SCHEMES FOR NAVIER-STOKES EQUATIONS: ALGORITHM AND IMPLEMENTATION INTO FDL3DI

DATTA V. GAITONDE
MIGUEL R. VISBAL

Air Vehicles Directorate
Air Force Research Laboratory
Wright-Patterson Air Force Base, Ohio 45433-7913

AUGUST 1998

FINAL REPORT FOR PERIOD 1 AUGUST 1997 – 31 JULY 1998

Approved for public release; distribution unlimited

19990614 010

AIR VEHICLES DIRECTORATE
AIR FORCE RESEARCH LABORATORY
AIR FORCE MATERIEL COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OH 45433-7542

DTIC QUALITY INSPECTED 4

NOTICE

USING GOVERNMENT DRAWINGS, SPECIFICATIONS, OR OTHER DATA INCLUDED IN THIS DOCUMENT FOR ANY PURPOSE OTHER THAN GOVERNMENT PROCUREMENT DOES NOT IN ANY WAY OBLIGATE THE US GOVERNMENT. THE FACT THAT THE GOVERNMENT FORMULATED OR SUPPLIED THE DRAWINGS, SPECIFICATIONS, OR OTHER DATA DOES NOT LICENSE THE HOLDER OR ANY OTHER PERSON OR CORPORATION; OR CONVEY ANY RIGHTS OR PERMISSION TO MANUFACTURE, USE, OR SELL ANY PATENTED INVENTION THAT MAY RELATE TO THEM.

THIS REPORT IS RELEASABLE TO THE NATIONAL TECHNICAL INFORMATION SERVICE (NTIS). AT NTIS, IT WILL BE AVAILABLE TO THE GENERAL PUBLIC, INCLUDING FOREIGN NATIONS.

THIS TECHNICAL REPORT HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION.

Datta V. Gaitonde
Research Aerospace Engineer
Computational Sciences Branch

Joseph J.S. Shang
Senior Scientist
Air Vehicles Directorate

James Rudd
Chief
Aeronautical Sciences Division

Do not return copies of this report unless contractual obligations or notice on a specific document require its return.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	AUG 98	FINAL REPORT 1 Aug 97-31 July 98

4. TITLE AND SUBTITLE
HIGH-ORDER SCHEMES FOR NAVIER-STOKES EQUATIONS: ALGORITHM AND IMPLEMENTATION INTO FDL3DI

5. FUNDING NUMBERS
C
PE 61102F
PR 2307
TA N4
WU: 02

6. AUTHOR(S)
DATTA V. GAITONDE and MIGUEL R. VISBAL

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
AIR VEHICLES DIRECTORATE
AIR FORCE RESEARCH LABORATORY
WRIGHT-PATTERSON AFB OH 45433-7913

8. PERFORMING ORGANIZATION REPORT NUMBER
AFRL-VA-WP-TR-1998-3060

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AIR VEHICLES DIRECTORATE
AIR FORCE RESEARCH LABORATORY
AIR FORCE MATERIEL COMMAND
WRIGHT-PATTERSON AFB OH 45433-7542
POC: J.S.Shang, AFRL/VAAC,(937)255-6157, M.Visbal,(937)255-2551

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
AFRL-VA-WP-TR-1998-3060

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE, DISTRIBUTION IS UNLIMITED

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

A spectrum of higher-order schemes is developed to solve the Navier-Stokes equations in finite-difference formulations. Pade type formulas of up to sixth order with a five-point stencil are developed for the difference scheme. Viscous terms are treated by successive applications of the first derivative operator. However, formulas are also derived for use in a mid-point interpolation-differentiation strategy. For numerical stability, up to tenth-order filtering schemes are developed. The spectral properties of the differentiation and filtering schemes are examined and guidelines are provided to choose proper filter coefficients. Special high-order formulas are obtained for differentiation and filtering in the vicinity of boundaries. The coefficients required for systematic implementation of Neumann-type boundary conditions are also presented. A brief description is provided of the manner in which the FDL3DI code is enhanced by coupling the approximately-factored procedure with these compact-difference based algorithms and by incorporating an explicit fourth-order Runge-Kutta scheme.

14. SUBJECT TERMS
HIGH-ORDER SCHEMES, COMPACT-DIFFERENCING, NAVIER-STOKES EQUATIONS, FILTERING

15. NUMBER OF PAGES
49

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAR

Standard Form 298 (Rev. 2-89) (EG)
Prescribed by ANSI Std. 239.18
Designed using Perform Pro, WHS/DIOR, Oct 94

Contents

List of Figures

List of Tables

Acknowledgements

This work was supported in part by a grant of HPC time from the DoD HPC Centers at CEWES (Cray C-90), NAVOCEANO (Cray C-90) and ASC (SGI).

Chapter 1

Introduction

The large computational requirement for the simulation of a wide range of turbulence [1], aeroacoustic [2] and electromagnetic [3] phenomena motivates the development and implementation of highly accurate schemes. In direct and large-eddy simulations of turbulence for example, high-order - for this purpose fourth-order accurate or higher - numerical schemes bring some previously intractable problems within the reach of modern supercomputers. An overview of recent efforts may be derived from a study of Refs. [4, 5, 6].

Although the complexity of problems addressed with higher-order schemes has increased over the past several years, remarkably few studies have focused on wall-bounded flows around geometrically complex configurations. Indeed, higher-order spatial schemes are usually coupled with explicit time-integration techniques. This is usually a good choice in situations where the limiting time-step size is dictated by physical rather than numerical constraints. For the large set of problems constituted by wall-bounded flows however, the stringent mesh resolution requirements near walls incur a severe numerical time-step-size limitation which can be alleviated by implicit methods.

The objective of this work is to describe the incorporation of a high-order accurate spatial differencing scheme into an existing implicit flow solver. The platform chosen for implementation is the FDL3DI code which is the primary research tool of the CFD group at Wright-Patterson AFB (AFRL/VAAC). The basic algorithm employs the Beam-Warming approximate factorization technique [7]. Several enhancements have been added as options including a subiteration technique and an efficient diagonalized procedure [8]. Since the code is formulated with finite-differences, all quantities are assumed to be *pointwise* in nature. Consequently, the formal difficulties encountered in extending finite-volume approaches to higher-order are avoided. However, flux and metric conservation issues are of some concern and are addressed elsewhere [9].

Several choices arise in the development of higher-order schemes. At the most basic level, the formula can be either centered or upwinded. While each has its advantages, we choose centered formulas because of their nondiffusive semidiscrete error. This is a particularly appropriate choice

for our present interests which encompass relatively low-speed (*i.e.*, subsonic) and thus shock-free flows. Extensions to include shock-capturing techniques are presently under development and will be described elsewhere. For a fixed order of accuracy, centered schemes have smaller stencil than upwind schemes. Even within centered schemes, additional advantage is obtained through the use of "compact" (or Padé type) formulas which require that the derivatives be computed in a coupled fashion along an entire line [4, 10]. With this approach, greater accuracy is obtained with fewer boundary schemes [10].

Compact schemes do however incur a moderate increase in computational expense over their non-compact counterparts. In order to limit this extra effort, in this entire work formulas are restricted to tridiagonal systems. With this simplification, a particular stencil size yields two orders higher accuracy than an explicit equivalent.

The formulas required to treat various aspects of the solution of the Navier-Stokes equations are presented in Chapter 2. In developing these schemes, the approach adopts the techniques discussed by Lele [4]. The principal mathematical tools required are Fourier analysis and Taylor series approximations. For the inviscid terms, formulas are required to evaluate first derivatives of the metric quantities and the fluxes. The maximum stencil size for this element of the algorithm is chosen to be five points, the highest scheme so obtained being of sixth-order accuracy. The coefficients of the five point scheme can be adjusted to yield lower order schemes as well. Section 2.1 lists the various formulas employed in this work, some of which have been derived previously elsewhere [10] but are reproduced and classified within the framework of the optimized schemes developed in Ref. [11]. The interior scheme cannot be applied at points near the boundary where the five point stencil protrudes the domain. These special formulas have not been standardized to the same extent as the interior schemes and are also presented in Section 2.1.

The numerical formation of the viscous terms in the Navier-Stokes equations requires calculation of derivatives of the components of the shear stress tensor. The straightforward approach followed here is to apply the formulas of Section 2.1 twice in succession. However, stability considerations suggest that the nonconservative form be employed together with formulas which compute the second derivatives directly (see *e.g.*, Ref. [4]). For compact schemes, this approach is expensive either in storage or in number of operations depending upon implementation. An alternate strategy is to utilize a midpoint interpolation and differentiate sequence to evaluate certain viscous terms. These formulas are presented in Sections 2.2 and 2.3 respectively. The dual formulas to compute nodal derivatives with known midpoint values are described in Section 2.4.

One of the principal problems encountered in the solution of the Navier-Stokes equations with centered schemes is the appearance of numerical instabilities, typically arising near boundaries and in regions of mesh nonuniformities. If left unchecked, these spurious waves contaminate the solution and destroy the fidelity of the solution. A common method to suppress such instabilities is through artificial dissipation in the form of a (small) additive damping term to the governing equations (*e.g.*, Refs. [12, 13]). A technique of similar vintage is to filter the solution at appropriate intervals in its temporal advancement (*e.g.*, Refs. [14, 15]). The distinction between damping and filtering is

relatively subtle. Ref. [16] notes that filtering is a more general approach not restricted to hyperbolic equations. In Section 2.5, tridiagonal based filters of up to tenth-order are presented. In each formula, control is exercised through a free parameter whose range and impact on stability are investigated.

The implementation of the various formulas into the FDL3DI code (to yield the new version FDL3D_ICE) is described in Chapter 3. An additional option of time-integration has also been added in the form of the classical fourth-order Runge-Kutta method (see *e.g.*., Ref. [17]). Finally, the Appendix presents a brief description of the input parameters to the FDL3D_ICE code with particular emphasis on those which are either new or whose meaning has been modified from the original code. Results on a range of fluid dynamic problems utilizing simple and complex mesh systems can be found in Ref. [9].

Chapter 2

Compact Formulas

Consider a 1-D mesh, consisting of N points (or nodes), $1, 2, \ldots, i-2, i-1, i, i+1, i+2, \ldots, N-2, N-1(=M), N$ as shown in Fig. 2.1 (a). Let $\phi = \phi(x)$ be a scalar variable whose pointwise values, ϕ_i are known at these nodes. We assume that $x_{i+1} - x_i = 1$ *i.e.*, the mesh step-size is normalized to unity. For body conforming meshes, a curvilinear transformation $\xi = \xi(x)$ is introduced and the same formulas are then employed in the transformed (ξ) plane. As noted earlier, the high-order method requires several types of quantities, formulas for which are now obtained in succession: a) first derivatives at nodes, b) interpolation of quantities from nodes to midpoints, c) first derivatives at midpoints in terms of known quantities at nodes, d) second derivatives at nodes from known first derivatives at midpoints and e) filtering formulas.

2.1 First Derivative at Nodes

The problem is to utilize the known ϕ_i to estimate the derivative, $\phi'_i = \partial\phi/\partial x|_i$ at each point in the mesh.

2.1.1 Interior Scheme

At interior points, a centered formula is employed:

$$\alpha\phi'_{i-1} + \phi'_i + \alpha\phi'_{i+1} = b\frac{\phi_{i+2} - \phi_{i-2}}{4} + a\frac{\phi_{i+1} - \phi_{i-1}}{2} \tag{2.1}$$

where α, a and b are constants which determine the spatial properties of the algorithm. Note that the stencil consists of five points as shown in Fig. 2.1(b). Up to sixth order accurate schemes can be obtained through proper choice of coefficients. To aid in this procedure, Taylor series approximations about point i are inserted in Eqn. 2.1 and terms of various orders are set equal to zero. This gives

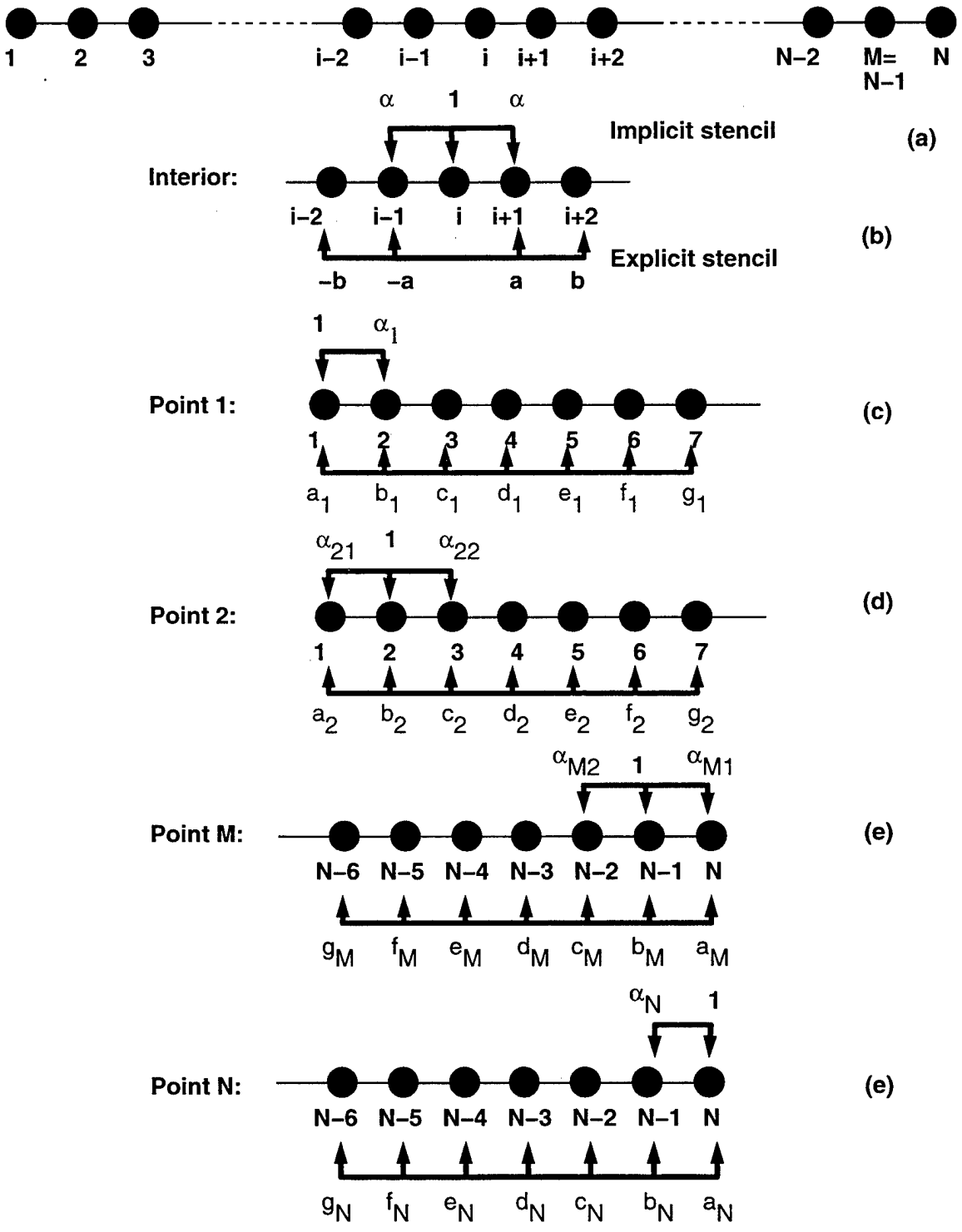

Figure 2.1: (a) Notation for 1-D Discretization, (b) Five-point Stencil for First Derivative at Interior Points, (c) Through (f) Stencils for First Derivatives at Points 1, 2, $M = N-1$ and N, Respectively

Table 2.1: Coefficients for Interior Scheme. OA=Order of accuracy

Scheme	α	a	b	Stencil Size	OA
E2	0	1	0	3	2
E4	0	$\frac{4}{3}$	$\frac{-1}{3}$	5	4
C4	$\frac{1}{4}$	$\frac{3}{2}$	0	3	4
C6	$\frac{1}{3}$	$\frac{14}{9}$	$\frac{1}{9}$	5	6
O1	0.351075	1.5673833	0.1347667	5	4
O2	0.381365	1.5875767	0.1751533	5	4
O3	0.347485	1.5649900	0.1299800	5	4
O4	0.370733	1.5804890	0.1609770	5	4
O5	0.430816	1.6205440	0.2410880	5	4
O6	0.376374	1.5842493	0.1684986	5	4
O7	0.400218	1.6001450	0.2002910	5	4

the following equations (see also Ref. [4]):

$$\begin{aligned} O(h^2): \quad & 1 - a + 2\alpha - b &&= 0 \\ O(h^4): \quad & -a + 6\alpha - 4b &&= 0 \\ O(h^6): \quad & -a + 10\alpha - 16b &&= 0 \end{aligned} \tag{2.2}$$

The solution of these equations provides values of a, b and α. Satisfaction of the first of equation in this set results in a second order scheme. If the second equation is also satisfied, a fourth order scheme results and the unique solution of all three equations yields a sixth order scheme.

Table 2.1 lists several standard schemes which can be derived by appropriate choice of coefficients. The first two, *E2* and *E4* are "explicit" i.e., $\alpha = 0$ and hence the derivatives values are decoupled from each other. For the remaining schemes, $\alpha \neq 0$ and it is necessary to solve a tridiagonal system. *C4* is the original fourth-order compact scheme discussed by Hirsh [10] and consists of a three point stencil. *C6*, described also in Ref. [4], is the highest-order scheme obtainable with the five-point formula of Eqn. 2.1. A semidiscrete accuracy analysis in the context of the wave equation has been performed in Refs. [4, 11]. Because of the centered stencil, the error is exclusively dispersive. The wave propagation speed and isotropy characteristics are compared with the exact value in Figs. 2.2 and 2.3 respectively for these standard schemes. In these figures, w is the normalized wave number,

$w = 2\pi kh/L$ where k is the physical wave number on a domain of length L and h is the grid spacing. The dramatic improvement in the compact schemes is clear: note the higher accuracy of the *C4* scheme compared to the sixth-order explicit scheme *E6*.

Table 2.1 also lists α values for several fourth-order optimized schemes. These schemes are designed to minimize selected error quantities over various wave number ranges as detailed in Ref. [18]. *O1* and *O3* minimize the semidiscrete isotropy error for a wave spectrum where the largest wave is resolved with 4 and $\frac{8}{3}$ intervals respectively. Equivalent schemes which minimize dispersion error are designated *O2* and *O4*, respectively. *O5* minimizes the dispersion error over the entire range of wave numbers up to 2 points per wave. However, it has been shown in Ref. [11] that such optimization is counterproductive because the absolute error is prohibitively large. *O6* and *O7* are relevant to the fully-discrete situation where the time-integration method is chosen to be the Runge-Kutta classical fourth order method: these schemes then minimize dispersion error at CFL numbers $\nu = 0.75$ and 1.0 respectively for a wave number spectrum resolved with four or more points for every component. Additional discussion on these optimized schemes, their derivation and error analyses can be found in Refs. [11].

Special formulas are required at points 1, 2, $N-1$ and N where the stencil of Eqn. 2.1 protrudes the domain. These formulas constitute numerical (or Phase I) conditions. Physical (or Phase II) conditions are addressed in Section 2.6 and in Ref. [9].

2.1.2 Boundary Point 1

In order to maintain the tridiagonal nature of the scheme, the formula employed at point 1 is:

$$\phi_1' + \alpha_1\phi_2' = a_1\phi_1 + b_1\phi_2 + c_1\phi_3 + d_1\phi_4 + e_1\phi_5 + f_1\phi_6 + g_1\phi_7 \tag{2.3}$$

The stencil is shown schematically in Fig. 2.1 (c). Upon inserting Taylor series approximations about point 1, and matching coefficients of equal order terms, a sequence of equations is obtained whose solution yields the coefficients which are listed in Table 2.2. Again, schemes with *E* prefixes are explicit because $\alpha_1 = 0$. Note that the scheme *C2* is the same as developed in Ref. [19].

2.1.3 Boundary Point 2

The general formula for the derivative at the first point away from the boundary is:

$$\alpha_{21}\phi_1' + \phi_2' + \alpha_{22}\phi_3' = a_2\phi_1 + b_2\phi_2 + c_2\phi_3 + d_2\phi_4 + e_2\phi_5 + f_2\phi_6 + g_2\phi_7 \tag{2.4}$$

Note that in general, both sides of Eqn. 2.4 are asymmetric about point 2. Several possibilities arise from different utilization of ϕ_1' in Eqn. 2.4. If the slope at point 1 is treated implicitly, two options arise:

- Option A: $\alpha_{21} = \alpha_{22} \neq 0$. In this case, the left hand side is symmetric about point 2. The coefficients obtained by Taylor series coefficient matching are presented in Table 2.3.

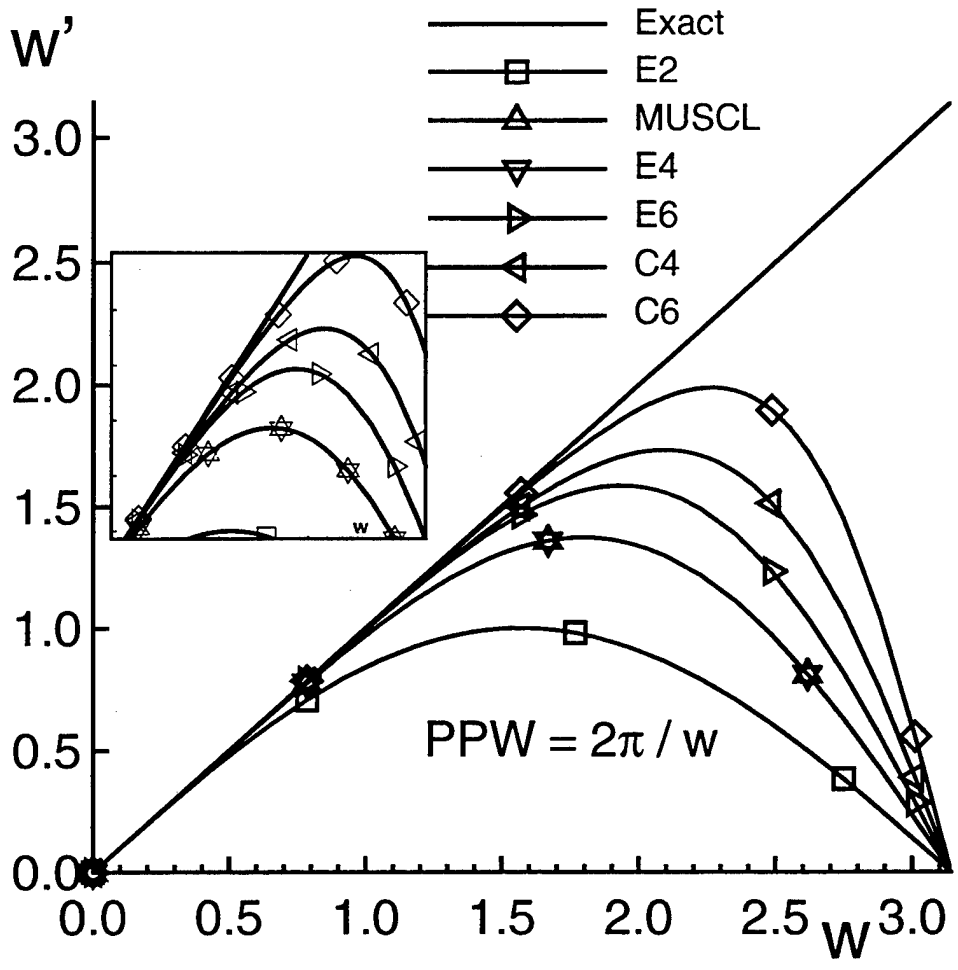

Figure 2.2: Numerical Versus Theoretical Wave Number Dispersion Characteristics of Various Schemes. w=normalized wave number, PPW = points per wave

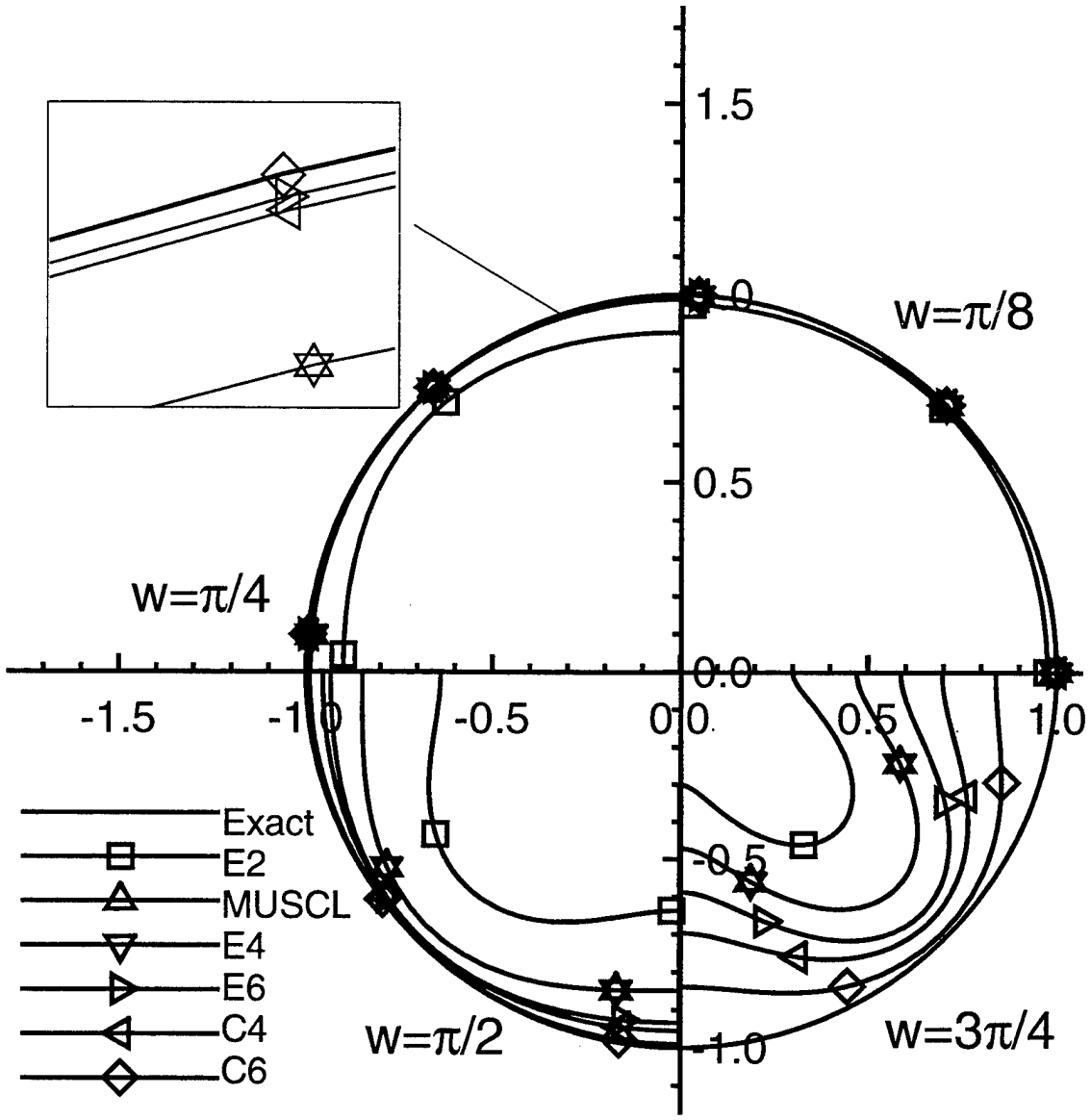

Figure 2.3: Semidiscrete Isotropy Characteristics of Various Schemes

Table 2.2: Boundary Coefficients for Point 1

Scheme	α_1	a_1	b_1	c_1	d_1	e_1	f_1	g_1	OA
E1	0	-1	1	0	0	0	0	0	1
E2	0	$\frac{-3}{2}$	2	$\frac{-1}{2}$	0	0	0	0	2
E3	0	$\frac{-11}{6}$	3	$\frac{-3}{2}$	$\frac{1}{3}$	0	0	0	3
E4	0	$\frac{-25}{12}$	4	-3	$\frac{4}{3}$	$\frac{-1}{4}$	0	0	4
E5	0	$\frac{-137}{60}$	5	-5	$\frac{10}{3}$	$\frac{-5}{4}$	$\frac{1}{5}$	0	5
E6	0	$\frac{-49}{20}$	6	$\frac{-15}{2}$	$\frac{20}{3}$	$\frac{-15}{4}$	$\frac{6}{5}$	$\frac{-1}{6}$	6
C2	1	-2	2	0	0	0	0	0	2
C3	2	$\frac{-5}{2}$	2	$\frac{1}{2}$	0	0	0	0	3
C4	3	$\frac{-17}{6}$	$\frac{3}{2}$	$\frac{3}{2}$	$\frac{-1}{6}$	0	0	0	4
C5	4	$\frac{-37}{12}$	$\frac{2}{3}$	3	$\frac{-2}{3}$	$\frac{1}{12}$	0	0	5
C6	5	$\frac{-197}{60}$	$\frac{-5}{12}$	5	$\frac{-5}{3}$	$\frac{5}{12}$	$\frac{-1}{20}$	0	6

Table 2.3: Boundary Coefficients for Point 2 with Option A: $\alpha_1 = \alpha_2 \neq 0$

Scheme	α_{21}	α_{22}	a_2	b_2	c_2	d_2	e_2	f_2	g_2	OA
AC4	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{-3}{4}$	0	$\frac{3}{4}$	0	0	0	0	4
AC5	$\frac{3}{14}$	$\frac{3}{14}$	$\frac{-19}{28}$	$\frac{-5}{42}$	$\frac{6}{7}$	$\frac{-1}{14}$	$\frac{1}{84}$	0	0	5
AC6	$\frac{2}{11}$	$\frac{2}{11}$	$\frac{-20}{33}$	$\frac{-35}{132}$	$\frac{34}{33}$	$\frac{-7}{33}$	$\frac{2}{33}$	$\frac{-1}{132}$	0	6

Table 2.4: Boundary Coefficients for Point 2 with Option B: $\alpha_{21} \neq \alpha_{22} \neq 0$. Note $BC4$ scheme is same as $AC4$ because this formula for a fourth-order three-point scheme is unique

Scheme	α_{21}	α_{22}	a_2	b_2	c_2	d_2	e_2	f_2	g_2	OA
BC3	$\frac{5}{8}$	$\frac{-1}{8}$	$\frac{-3}{2}$	$\frac{3}{2}$	0	0	0	0	0	3
BC4	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{-3}{4}$	0	$\frac{3}{4}$	0	0	0	0	4
BC5	$\frac{1}{6}$	$\frac{1}{2}$	$\frac{-5}{9}$	$\frac{-1}{2}$	1	$\frac{1}{18}$	0	0	0	5
BC6	$\frac{1}{8}$	$\frac{3}{4}$	$\frac{-43}{96}$	$\frac{-5}{6}$	$\frac{9}{8}$	$\frac{1}{6}$	$\frac{-1}{96}$	0	0	6
BC7	$\frac{1}{10}$	1	$\frac{-227}{600}$	$\frac{-13}{12}$	$\frac{7}{6}$	$\frac{1}{3}$	$\frac{-1}{24}$	$\frac{1}{300}$	0	7
BC8	$\frac{1}{12}$	$\frac{5}{4}$	$\frac{-79}{240}$	$\frac{-77}{60}$	$\frac{55}{48}$	$\frac{5}{9}$	$\frac{-5}{48}$	$\frac{1}{60}$	$\frac{-1}{720}$	8

Table 2.5: Boundary Formulas for Point 2 with Option C: $\alpha_{21} = 0, \alpha_{22} \neq 0$

Scheme	α_{21}	α_{22}	a_2	b_2	c_2	d_2	e_2	f_2	g_2	OA
CC2	0	$\frac{-1}{3}$	$\frac{-2}{3}$	$\frac{2}{3}$	0	0	0	0	0	2
CC3	0	$\frac{1}{2}$	$\frac{-1}{4}$	-1	$\frac{5}{4}$	0	0	0	0	3
CC4	0	1	$\frac{-1}{6}$	$\frac{-3}{2}$	$\frac{3}{2}$	$\frac{1}{6}$	0	0	0	4
CC5	0	$\frac{3}{2}$	$\frac{-1}{8}$	$\frac{-11}{6}$	$\frac{3}{2}$	$\frac{1}{2}$	$\frac{-1}{24}$	0	0	5
CC6	0	2	$\frac{-1}{10}$	$\frac{-25}{12}$	$\frac{4}{3}$	1	$\frac{-1}{6}$	$\frac{1}{60}$	0	6
CC7	0	$\frac{5}{2}$	$\frac{-1}{12}$	$\frac{-137}{60}$	$\frac{25}{24}$	$\frac{5}{3}$	$\frac{-5}{12}$	$\frac{1}{12}$	$\frac{-1}{120}$	7

- Option B: $\alpha_{21} \neq \alpha_{22} \neq 0$. Because of the extra degree of freedom, for the same stencil as in Option A, one degree higher order of accuracy is obtained as shown in Table 2.4.

In most algorithms, the solution is updated only at interior points *i.e.*, all points excluding 1 and N where the values ϕ_1 and ϕ_N are determined from the physical constraints of the problem. Thus, the slopes at these endpoints are not required and it is possible to decouple them (but not the pointwise values ϕ_1 and ϕ_N) from the rest of the domain by setting $\alpha_{21} = 0$. Again, two possibilities exist:

- Option C: $\alpha_{21} = 0, \alpha_{22} \neq 0$. The coefficients of this implicit formulation are listed in Table 2.5

- Option D: $\alpha_{21} = \alpha_{22} = 0$. In this case, the slope at point 2 is computed explicitly with the coefficients listed in Table 2.6.

Table 2.6: Boundary Coefficients for Point 2 with Option D: $\alpha_{21} = \alpha_{22} = 0$

Scheme	α_{21}	α_{22}	a_2	b_2	c_2	d_2	e_2	f_2	g_2	OA
DE1	0	0	-1	1	0	0	0	0	0	1
DE2	0	0	$\frac{-1}{2}$	0	$\frac{1}{2}$	0	0	0	0	2
DE3	0	0	$\frac{-1}{3}$	$\frac{-1}{2}$	1	$\frac{-1}{6}$	0	0	0	3
DE4	0	0	$\frac{-1}{4}$	$\frac{-5}{6}$	$\frac{3}{2}$	$\frac{-1}{2}$	$\frac{1}{12}$	0	0	4
DE5	0	0	$\frac{-1}{5}$	$\frac{-13}{12}$	2	-1	$\frac{1}{3}$	$\frac{-1}{20}$	0	5
DE6	0	0	$\frac{-1}{6}$	$\frac{-77}{60}$	$\frac{5}{2}$	$\frac{-5}{3}$	$\frac{5}{6}$	$\frac{-1}{4}$	$\frac{1}{30}$	6

2.1.4 Boundary Point $M = N - 1$

The difference scheme at the point away from the right boundary is similar to that derived for point 2. The formula chosen is:

$$\alpha_{M1}\phi'_{N-2}+\phi'_{N-1}+\alpha_{M2}\phi'_N = a_M\phi_N+b_M\phi_{N-1}+c_M\phi_{N-2}+d_M\phi_{N-3}+e_M\phi_{N-4}+f_M\phi_{N-5}+g_M\phi_{N-6} \tag{2.5}$$

The same options can be proposed as for point 2. Because of the structural relationship between Eqs. 2.5 and 2.3, Tables 2.3 through 2.6 are applicable with the modifications that i) $\alpha_{M1} = \alpha_{22}$, ii) $\alpha_{M2} = \alpha_{21}$, and iii) the signs of each of the coefficients a through g are reversed *i.e.*, $a_M = -a_2$, $b_M = -b_2, \ldots$.

2.1.5 Boundary Point N

The formula for the boundary point N is:

$$\alpha_N\phi'_{N-1} + \phi'_N = a_N\phi_N + b_N\phi_{N-1} + c_N\phi_{N-2} + d_N\phi_{N-3} + e_N\phi_{N-4} + f_N\phi_{N-5} + g_N\phi_{N-6} \tag{2.6}$$

Again, because of the structural similarity between Eqns. 2.6 and 2.3, Table 2.2 is applicable with i) $\alpha_N = \alpha_1$ and ii) the signs of each of the coefficients a through g are reversed, *i.e.*, $a_N = -a_1$, $b_N = -b_1, \ldots$.

2.2 Interpolation Formulas for Midpoint Values

The formation of some viscous terms may be greatly facilitated by the use of function and derivative values at midpoints. In this section, the function values are obtained at midpoint values with interpolation formulas using the basic procedure of Lele [4]. The notation employed and the schematic of

Table 2.7: Coefficients for Interior Interpolation Formula

Scheme	α	a	b	c	OA
C2	Free	$1+2\alpha$	0	0	2
C4	Free	$\frac{9}{8}+\frac{5\alpha}{4}$	$\frac{-1+6\alpha}{8}$	0	4
C6	Free	$\frac{5(15+14\alpha)}{64}$	$\frac{-25+126\alpha}{128}$	$\frac{3-10\alpha}{128}$	6
C8	$\frac{5}{14}$	$\frac{25}{16}$	$\frac{5}{32}$	$\frac{-1}{224}$	8

the stencil are sketched in Fig. 2.4. We focus again only on tridiagonal schemes with up to 8th-order accuracy *i.e.*, two orders higher than for the derivative schemes in Section 2.1.

2.2.1 Interior Scheme

The basic formula for interpolation at interior points is:

$$\alpha\phi_{i-\frac{1}{2}} + \phi_{i+\frac{1}{2}} + \alpha\phi_{i+\frac{3}{2}} = \frac{a}{2}\left(\phi_{i+1}+\phi_i\right) + \frac{b}{2}\left(\phi_{i+2}+\phi_{i-1}\right) + \frac{c}{2}\left(\phi_{i+3}+\phi_{i-2}\right) \tag{2.7}$$

where it is obvious that the coefficients α, a and b have no relation to those of Eqn. 2.1. Matching of Taylor series terms gives the coefficients listed in Table 2.7. For two reasons, α is retained as a free parameter for all but the highest-order scheme for which α is a unique nonzero number. The first advantage concerns notational brevity: explicit schemes can be easily derived by setting $\alpha = 0$ while the equivalent compact scheme can be obtained by setting the "outermost" coefficient to zero – thus reducing the stencil size by 2 points – and solving for a unique α. For example, in Table 2.7, an explicit $6th$ order scheme, *E6*, can be derived by setting $\alpha = 0$ in the row labeled *C6*, thus giving, $a = 75/64$, $b = -25/128$ and $c = 3/128$. On the other hand, the compact $6th$ order scheme can be obtained by setting $c = 0$ in the same row *i.e.*, $\alpha = 3/10$ and thus $a = 3/2$ and $b = 1/10$. The second reason to retain α as a free parameter is that it may then be potentially employed for optimization purposes similar to that discussed for filtering in Section 2.5.

2.2.2 Points Near Boundaries

Point $\frac{3}{2}$

Formula

$$\phi_{\frac{3}{2}} + \alpha\phi_{\frac{5}{2}} = a\phi_1 + b\phi_2 + c\phi_3 + d\phi_4 + e\phi_5 + f\phi_6 + g\phi_7 \tag{2.8}$$

Coefficients: Table 2.8. Note: for brevity, the subscripts on each coefficient have been omitted in this and subsequent formulas.

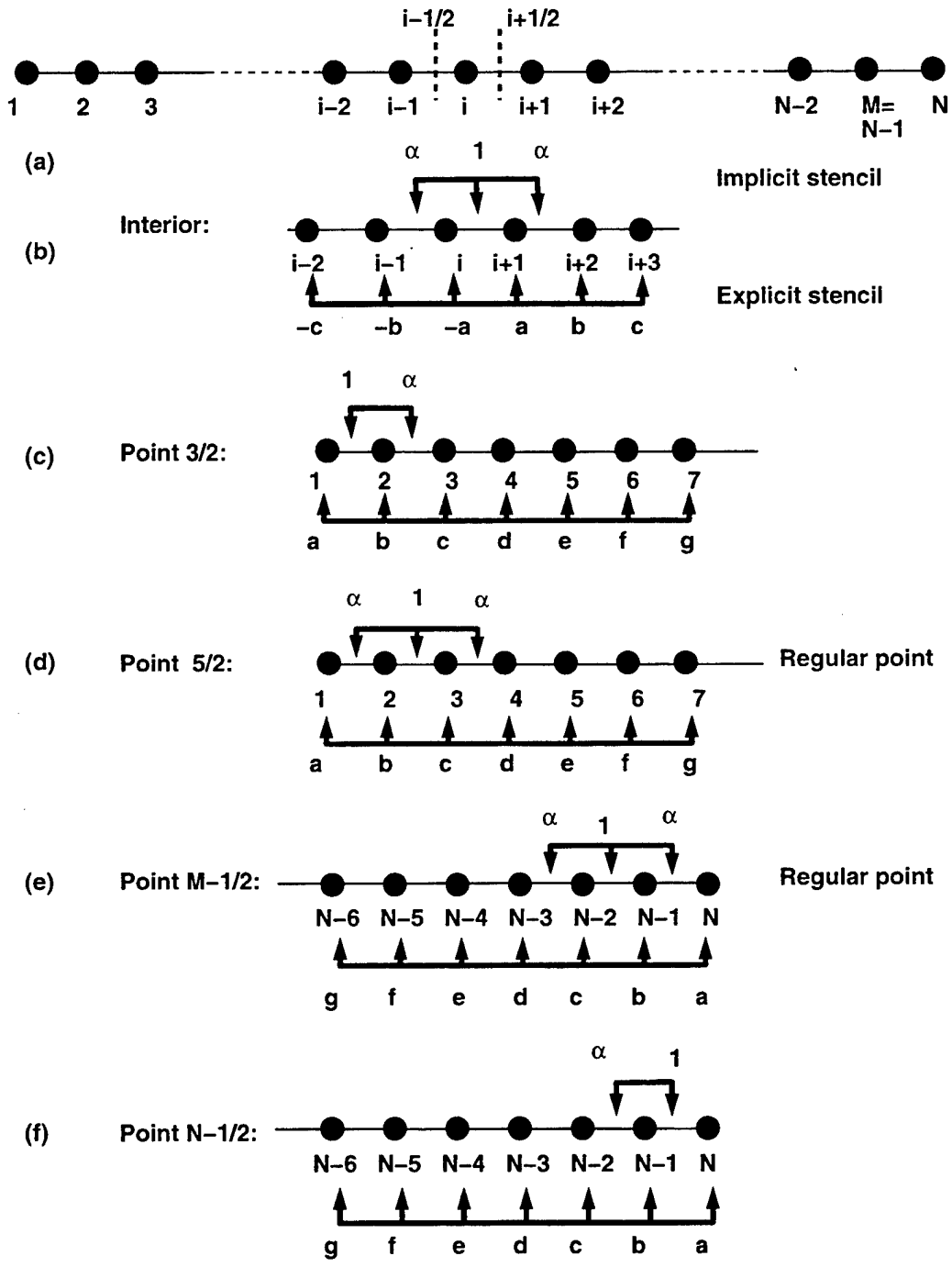

Figure 2.4: (a) Notation for Midpoints in 1-D Discretization, (b) Stencil for First Derivative at Interior Midpoints, (c) Through (f) Stencils for First Derivatives at Points $\frac{3}{2}$, $\frac{5}{2}$, $M = N - \frac{3}{2}$ and $N - \frac{1}{2}$ Respectively

Table 2.8: Coefficients for Interpolation Formulas at $\frac{3}{2}$ *i.e.*, at the Midpoint Between Nodes 1 and 2

Scheme	α	a	b	c	d	e	f	g	OA
C1	Free	$1+\alpha$	0	0	0	0	0	0	1
C2	Free	$\frac{1}{2}-\frac{\alpha}{2}$	$\frac{1}{2}+\frac{3\alpha}{2}$	0	0	0	0	0	2
C3	Free	$\frac{3}{8}-\frac{\alpha}{8}$	$\frac{3}{4}+\frac{3\alpha}{4}$	$-\frac{1}{8}+\frac{3\alpha}{8}$	0	0	0	0	3
C4	Free	$\frac{5}{16}-\frac{\alpha}{16}$	$\frac{15}{16}+\frac{9\alpha}{16}$	$-\frac{5}{16}+\frac{9\alpha}{16}$	$\frac{1}{16}-\frac{\alpha}{16}$	0	0	0	4
C5	Free	$\frac{35}{128}-\frac{5\alpha}{128}$	$\frac{35}{32}+\frac{15\alpha}{32}$	$-\frac{35}{64}+\frac{45\alpha}{64}$	$\frac{7}{32}-\frac{5\alpha}{32}$	$-\frac{5}{128}+\frac{3\alpha}{128}$	0	0	5
C6	Free	$\frac{63}{256}-\frac{7\alpha}{256}$	$\frac{315}{256}+\frac{105\alpha}{256}$	$\frac{15(-7+7\alpha)}{128}$	$\frac{63-35\alpha}{128}$	$\frac{3(-15+7\alpha)}{256}$	$\frac{7-3\alpha}{256}$	0	6
C7	Free	$\frac{21(11-\alpha)}{1024}$	$\frac{63(11+3\alpha)}{512}$	$\frac{105(-11+9\alpha)}{1024}$	$\frac{21(11-5\alpha)}{256}$	$\frac{9(-55+21\alpha)}{1024}$	$\frac{77-27\alpha}{512}$	$\frac{7(-3+\alpha)}{1024}$	7
C8	$\frac{11}{3}$	$\frac{77}{512}$	$\frac{693}{256}$	$\frac{1155}{512}$	$\frac{-77}{128}$	$\frac{99}{512}$	$\frac{-11}{256}$	$\frac{7}{1536}$	8

Point $\frac{5}{2}$

Formula

$$\alpha\phi_{\frac{3}{2}} + \phi_{\frac{5}{2}} + \alpha\phi_{\frac{7}{2}} = a\phi_1 + b\phi_2 + c\phi_3 + d\phi_4 + e\phi_5 + f\phi_6 + g\phi_7 \tag{2.9}$$

Coefficients: Table 2.9. Note: It is evident from Fig. 2.4 that for lower even-order schemes (e.g., *C4*), this is an interior point and the formulas are the same as in Table 2.7.

Point $M-\frac{1}{2}$

Formula

$$\alpha\phi_{M-\frac{3}{2}} + \phi_{M-\frac{1}{2}} + \alpha\phi_{M+\frac{1}{2}} = a\phi_N + b\phi_{N-1} + c\phi_{N-2} + d\phi_{N-3} + e\phi_{N-4} + f\phi_{N-5} \tag{2.10}$$

Coefficients: Table 2.9. Note: the coefficients are precisely the same as for point $\frac{5}{2}$.

Point $M+\frac{1}{2}=N-\frac{1}{2}$

Formula

$$\alpha\phi_{M-\frac{1}{2}} + \phi_{M+\frac{1}{2}} = a\phi_N + b\phi_{N-1} + c\phi_{N-2} + d\phi_{N-3} + e\phi_{N-4} + f\phi_{N-5} + g\phi_{N-6} \tag{2.11}$$

Coefficients: Table 2.8. Note: the coefficients are precisely the same as for point $\frac{3}{2}$.

2.3 Derivative Formulas at Midpoints from Known Nodal Values

The formation of the stress tensor at midpoints also requires certain derivative values at midpoints. The formulas may be derived in essentially the same manner as for interpolation. Since the node

Table 2.9: Coefficients for Interpolation Formula at Midpoint $\frac{5}{2}$

Scheme	α	a	b	c	d	e	f	g	OA
C1	Free	$1+2\alpha$	0	0	0	0	0	0	1
C2	Free	$\frac{-1}{2}-\alpha$	$\frac{3}{2}+3\alpha$	0	0	0	0	0	2
C3	Free	$-\frac{1}{8}+\frac{3\alpha}{4}$	$\frac{3}{4}-\frac{\alpha}{2}$	$\frac{3}{8}+\frac{7\alpha}{4}$	0	0	0	0	3
C4	Free	$-\frac{1}{16}+\frac{3\alpha}{8}$	$\frac{9}{16}+\frac{5\alpha}{8}$	$\frac{9}{16}+\frac{5\alpha}{8}$	$-\frac{1}{16}+\frac{3\alpha}{8}$	0	0	0	4
C5	Free	$-\frac{5}{128}+\frac{19\alpha}{64}$	$\frac{15}{32}+\frac{15\alpha}{16}$	$\frac{45}{64}+\frac{5\alpha}{32}$	$-\frac{5}{32}+\frac{11\alpha}{16}$	$\frac{3}{128}-\frac{5\alpha}{64}$	0	0	5
C6	Free	$-\frac{7}{256}+\frac{33\alpha}{128}$	$\frac{105+290\alpha}{256}$	$\frac{15(7-2\alpha)}{128}$	$\frac{-35+138\alpha}{128}$	$\frac{21-70\alpha}{256}$	$\frac{-3+10\alpha}{256}$	0	6
C7	Free	$\frac{7(-3+34\alpha)}{1024}$	$\frac{7(27+94\alpha)}{512}$	$\frac{315(3-2\alpha)}{1024}$	$\frac{7(-15+58\alpha)}{256}$	$\frac{189-670\alpha}{1024}$	$\frac{-27+98\alpha}{512}$	$\frac{7-26\alpha}{1024}$	7
C8	$\frac{9}{38}$	$\frac{21}{608}$	$\frac{819}{1216}$	$\frac{945}{1216}$	$\frac{-21}{608}$	$\frac{9}{304}$	$\frac{-9}{1216}$	$\frac{1}{1216}$	8

Table 2.10: Coefficients for Interior Midpoint Differentiation Scheme. OA=Order of accuracy

Scheme	α	a	b	OA
E2	Free	1	0	2
C4	Free	$\frac{9}{8}-\frac{3\alpha}{4}$	$-\frac{1}{8}+\frac{11\alpha}{4}$	4
C6	$\frac{9}{62}$	$\frac{63}{62}$	$\frac{17}{62}$	6

derivatives are restricted to 6th order, we seek only up to sixth order accuracy for midpoint derivatives as well.

2.3.1 Interior Scheme

Formula

$$\alpha\phi'_{i-\frac{1}{2}}+\phi'_{i+\frac{1}{2}}+\alpha\phi'_{i+\frac{3}{2}}=a\left(\phi_{i+1}-\phi_i\right)+\frac{b}{3}\left(\phi_{i+2}-\phi_{i-1}\right) \tag{2.12}$$

Coefficients Table 2.10 Note: Because of the restriction to $6th$ order, the general formula does not require the coefficient c.

Table 2.11: Coefficients for Midpoint Differentiation Scheme at Point 3/2. OA=Order of accuracy.

Scheme	α	a	b	c	d	e	f	OA
E2	0	-1	1	0	0	0	0	2
C3	$-\frac{23}{24}+\frac{\alpha}{24}$	$\frac{7-9\alpha}{8}$	$\frac{1}{8}+\frac{9\alpha}{8}$	$\frac{-1-\alpha}{24}$	3			
C4	Free	$\frac{-22+\alpha}{24}$	$\frac{17}{24}-\frac{9\alpha}{8}$	$\frac{3+9\alpha}{8}$	$\frac{-5-\alpha}{24}$	$\frac{1}{24}$		
C5	Free	$\frac{-1689+71\alpha}{1920}$	$\frac{67-141\alpha}{128}$	$\frac{143+207\alpha}{192}$	$\frac{-111+\alpha}{192}$	$\frac{29-3\alpha}{128}$	$\frac{-71+9\alpha}{1920}$	5
C6	$\frac{62}{9}$	$\frac{-10799}{17280}$	$\frac{-2713}{384}$	$\frac{523}{64}$	$\frac{-937}{1728}$	$\frac{25}{384}$	$\frac{-3}{640}$	6

2.3.2 Points Near Boundaries

Point $\frac{3}{2}$

Formula

$$\phi'_{\frac{3}{2}}+\alpha\phi'_{i+\frac{5}{2}}=a\phi_1+b\phi_2+c\phi_3+d\phi_4+e\phi_5+f\phi_6 \tag{2.13}$$

Coefficients: Table 2.11. Note: The scheme designated *C4* is not really compact since α cannot be chosen to set e – which is a constant – to zero.

Points $\frac{5}{2}$ and $M-\frac{1}{2}$

For all orders up to sixth, these are interior points. Thus, Eqn. 2.12 and the corresponding coefficients of Table 2.10 are applicable and no special treatment is required. Note that this situation differs from the case for interpolation where up to 8th order formulas were derived.

Point $M+\frac{1}{2}$

Formula

$$\phi'_{N-\frac{1}{2}}+\alpha\phi'_{N-\frac{3}{2}}=a\phi_N+b\phi_{N-1}+c\phi_{N-2}+d\phi_{N-3}+e\phi_{N-4}+f\phi_{N-5} \tag{2.14}$$

Coefficients: May be derived from Table 2.11 *by reversing the signs of each of a through f* but not of α.

2.4 Derivatives Formulas at Nodes from Known Midpoint Values

The midpoint interpolated and derivative values obtained from Sections 2.2 and 2.3 can be employed to form composite values, in this case certain components of the stress tensor as outlined later. It is necessary to then differentiate these midpoint composite values to obtain essentially second derivatives at the nodes.

Table 2.12: Coefficients for Differentiation at Point 1 with Known Midpoint Values

Scheme	α	a	b	c	d	e	f	OA
C2	Free	$-2-\alpha$	$3+\alpha$	-1	0	0	0	2
C3	Free	$-\frac{71}{24}-\frac{23\alpha}{24}$	$\frac{47}{8}+\frac{7\alpha}{8}$	$\frac{-31+\alpha}{8}$	$\frac{23-\alpha}{24}$	0	0	3
C4	Free	$\frac{-93-22\alpha}{24}$	$\frac{229}{24}+\frac{17\alpha}{24}$	$\frac{-22+\alpha}{24}$	$\frac{37}{8}-\frac{5\alpha}{24}$	$\frac{-22+\alpha}{24}$	0	4
C5	Free	$\frac{-3043-563\alpha}{640}$	$\frac{5353+201\alpha}{384}$	$\frac{-3489+143\alpha}{192}$	$\frac{859-37\alpha}{64}$	$\frac{-2041+87\alpha}{384}$	$\frac{1689-71\alpha}{1920}$	5
C6	$\frac{1627}{62}$	$\frac{-1104667}{39680}$	$\frac{658913}{23808}$	$\frac{16343}{11904}$	$\frac{-6941}{3968}$	$\frac{15007}{23808}$	$\frac{-10799}{119040}$	6

2.4.1 Interior Scheme

Formula

$$\alpha\phi'_{i-1}+\phi'_i+\phi'_{i+1}=a\left(\phi_{i+\frac{1}{2}}-\phi_{i-\frac{1}{2}}\right)+\frac{b}{3}\left(\phi_{i+\frac{3}{2}}-\phi_{i-\frac{3}{2}}\right) \tag{2.15}$$

Coefficients: Table 2.10. Note: At interior points, this is the dual of the problem of Section 2.3 and thus have the same coefficients.

2.4.2 Points Near Boundaries

Point 1

Formula

$$\phi'_1+\alpha\phi'_2=a\phi_{\frac{3}{2}}+b\phi_{\frac{5}{2}}+c\phi_{\frac{7}{2}}+d\phi_{\frac{9}{2}}+e\phi_{\frac{11}{2}}+f\phi_{\frac{13}{2}} \tag{2.16}$$

Coefficients: Table 2.12

Point 2

Formula

$$\alpha\phi'_1+\phi'_2+\alpha\phi'_3=a\phi_{\frac{3}{2}}+b\phi_{\frac{5}{2}}+c\phi_{\frac{7}{2}}+d\phi_{\frac{9}{2}}+e\phi_{\frac{11}{2}}+f\phi_{\frac{13}{2}} \tag{2.17}$$

Coefficients: Table 2.13.

Point $M=N-1$

Formula

$$\alpha\phi'_{M+1}+\phi'_M+\alpha\phi'_{M-1}=a\phi_{M+\frac{1}{2}}+b\phi_{M-\frac{1}{2}}+c\phi_{M-\frac{3}{2}}+d\phi_{M-\frac{5}{2}}+e\phi_{M-\frac{7}{2}}+f\phi_{M-\frac{9}{2}} \tag{2.18}$$

Coefficients: Obtained from Table 2.13 *by reversing the signs of each of a through f* but not of α.

Point N

Formula

$$\phi'_N+\alpha\phi'_{N-1}=a\phi_{N-\frac{1}{2}}+b\phi_{N-\frac{3}{2}}+c\phi_{N-\frac{5}{2}}+d\phi_{N-\frac{7}{2}}+e\phi_{N-\frac{9}{2}}+f\phi_{N-\frac{11}{2}} \tag{2.19}$$

Table 2.13: Coefficients for Differentiation at Point 2 with Known Midpoint Values

Scheme	α	a	b	c	d	e	f	OA
E2	0	-1	1	0	0	0	0	2
C3	Free	$-\frac{23}{24}-\frac{35\alpha}{12}$	$\frac{7}{8}+\frac{19\alpha}{4}$	$\frac{1}{8}-\frac{11\alpha}{4}$	$-\frac{1}{24}+\frac{11\alpha}{12}$	0	0	3
C4	Free	$\frac{-11-46\alpha}{12}$	$\frac{17}{24}+\frac{101\alpha}{12}$	$\frac{3}{8}-\frac{33\alpha}{4}$	$\frac{5(-1+22\alpha)}{24}$	$\frac{1}{24}-\frac{11\alpha}{12}$	0	4
C5	Free	$\frac{-1689-9058\alpha}{1920}$	$\frac{201+4930\alpha}{384}$	$\frac{143-3282\alpha}{192}$	$\frac{-111+2578\alpha}{192}$	$\frac{87-2050\alpha}{384}$	$\frac{-71+1698\alpha}{1920}$	5
C6	$\frac{31}{818}$	$-\frac{5195}{4908}$	$\frac{4957}{4908}$	$\frac{119}{1227}$	$-\frac{85}{1227}$	$\frac{119}{4908}$	$-\frac{17}{4908}$	6

Coefficients: Obtained from Table 2.12 *by reversing the signs of each of a through f* but not of α.

2.5 Filter Formulas

The interior schemes presented in earlier sections address the issue of accuracy. An equally important consideration is that of stability. For high-order schemes, theoretical analyses of stability are not straightforward, the principal exceptions being the simplest cases where no boundaries are present, the governing equations are linear and the mesh is uniform and further where the time-integration methods are explicit. Practical calculations usually do not satisfy these stringent conditions in several ways. Truncated boundaries, nonlinearities, curvilinear meshes and, in the case of wall-bounded flows, implicit methods of time-integration are the norm.

As noted earlier, special schemes are necessary at points near the boundaries where the interior formulas cannot be applied. The impact of the formulas employed at schemes has been examined in Refs. [20, 21, 22] and the citations therein. The approach is typically to choose a well-posed linear scalar problem discretized on a uniform discrete mesh. The composite scheme is then subjected to an eigenvalue analysis. The algebraic complications are enormous and recourse to theories connecting the properties of semidiscrete to fully-discrete schemes are often invoked. The complications are more severe for systems of equations, see for example Ref. [6].

Far fewer studies address the imposition of the second or physical phase of the boundary condition implementation. Typically, these include physical approximations. For example, the condition $\partial p/\partial n = 0$, where n refers to the direction normal to the boundary, is imposed at solid walls. The condition is derived from boundary layer theory and although it is used extensively, may incur sizeable error near points of separation or attachment. Additionally, the actual implementation of this condition often approximates $\partial p/\partial n \simeq \partial p/\partial \eta = 0$ where η is the coordinate line emanating from the wall and may not point along the normal. The impact of such approximations on the theoretical

stability of a scheme has not been sufficiently addressed in the literature.

The quality of the mesh employed also plays a crucial role in the performance of the scheme. In Ref. [23], it is shown that central-difference based schemes exhibit spurious reflections at interfaces where a step-jump in the spacing is encountered. Indeed, boundary conditions derived for uniform meshes are not necessarily stable on curvilinear meshes: Ref. [24] notes one such instance.

All the above issues introduce uncertainty into the applicability of tractable theoretical stability analyses. In practice, stability is usually achieved through the addition of a small amount of damping [13] or less commonly through filtering which forms the focus of this study. An extensive discussion of the basic ideas behind filtering, which is usually applied after the solution vector is updated, can be found in Ref. [16] and the citations therein. In recent years, the increased use of very high-order methods has encouraged the development of correspondingly high-order filters. An extensive development of explicit filters, *i.e.* those not dependent upon the solution of matrix systems, can be found in Refs. [16, 15] while the methodology to derive compact schemes has been outlined by Lele [4]. In keeping with the interior algorithm, this work utilizes the latter approach.

In Ref. [4], a series of optimized tri- and pentadiagonal compact schemes were developed for up to sixth-order accuracy. Several were designed to satisfy specific amplification properties. We employ the same techniques to develop a different set of filters more consistent with the interior scheme, restricting attention to tridiagonal-based schemes of up to $10th$ order. Additionally, the variable α_f is retained as a free parameter in order to provide some control on the "degree" of filtering. With these choices, a $2N$th order filter has a stencil of $2N+1$ points and the stencil is wider than that in Ref. [4] and thus require derivation. The bounds of the independent parameter α_f are established and the performance of the filter is characterized for various values within these bounds, in effect providing guidelines in the choice of this parameter. At present, the conserved quantities are filtered though options exist to apply filtering to the various combinations of conserved and primitive variables.

2.5.1 Interior Scheme

The filtering procedure replaces the computed (updated) value ϕ with $\bar{phi}$ obtained from the following equation.

Formula

$$\alpha_f \bar{\phi}_{i-1} + \bar{\phi}_i + \alpha_f \bar{\phi}_{i+1} = \Sigma_{n=0}^{N} \frac{a_n}{2} (u_{i+n} + u_{i-n}) \tag{2.20}$$

Coefficients: Table 2.14. Note: $\bar{\phi}$ are the filtered values of ϕ. The spectral function (or frequency response) of the operator is:

$$\mathcal{SF}(w) = \frac{\Sigma_{n=0}^{N} a_n \cos(nw)}{1 + 2\alpha_f \cos(w)} \tag{2.21}$$

which has $N+2$ unknowns, consisting of $\alpha_f, a_0, a_1, \ldots a_N$. To obtain the coefficients, we first insist that the highest frequency mode be eliminated by enforcing the condition $\mathcal{SF}(\pi) = 0$ (see Refs [4, 16]). The remaining $N+1$ required additional equations can be derived by matching Taylor series

Table 2.14: Coefficients for Filter Formula at Interior Points. α_f is a Free Parameter in the range $0 < \alpha_f < 0.5$.

Scheme	a_0	a_1	a_2	a_3	a_4	a_5	OA
F2	$\frac{1}{2}+\alpha_f$	$\frac{1}{2}+\alpha_f$	0	0	0	0	2
F4	$\frac{5}{8}+\frac{3\alpha_f}{4}$	$\frac{1}{2}+\alpha_f$	$\frac{-1}{8}+\frac{\alpha_f}{4}$	0	0	0	4
F6	$\frac{11}{16}+\frac{5\alpha_f}{8}$	$\frac{15}{32}+\frac{17\alpha_f}{16}$	$\frac{-3}{16}+\frac{3\alpha_f}{8}$	$\frac{1}{32}-\frac{\alpha_f}{16}$	0	0	6
F8	$\frac{93+70\alpha_f}{128}$	$\frac{7+18\alpha_f}{16}$	$\frac{-7+14\alpha_f}{32}$	$\frac{1}{16}-\frac{\alpha_f}{8}$	$\frac{-1}{128}+\frac{\alpha_f}{64}$	0	8
F10	$\frac{193+126\alpha_f}{256}$	$\frac{105+302\alpha_f}{256}$	$\frac{15(-1+2\alpha_f)}{64}$	$\frac{45(1-2\alpha_f)}{512}$	$\frac{5(-1+2\alpha_f)}{256}$	$\frac{1-2\alpha_f}{512}$	10

coefficients of the left and right sides expanded about i. Because of the symmetric nature of Eqn. 2.20, $\mathcal{SF}$ is real and the filter only modifies the amplitude of each wave component. Further, only even order terms persist and each coefficient match yields two orders of accuracy. The property $|\mathcal{SF}(w)| \leq 1$ is verified *a posteriori*. For concreteness, we choose a maximum stencil of 11 points with seven unknowns ($N = 5$). While the seven equations arising from the above matching procedure can be solved uniquely to obtain a 12th order filter, the formula degenerates to an identity, $\alpha_f = 1/2, a_0 = 1, a_1 = 1/2, a_n = 0$ for $n > 1$ and $\mathcal{SF}(\pi) = \frac{0}{0}$ is indeterminate. One solution is to suppress the highest coefficient match (thus restricting to 10th order) and to employ instead a supplemental equation by assigning $\mathcal{SF}(w)$ at some specific wave number as in Refs. [4, 11]. Another approach, as in Table 2.14, is to retain α_f as a free parameter which provides the flexibility of an explicit filter subset (obtained by setting α_f to zero). Because of the form of the denominator of Eqn. 2.21, $|\alpha_f| < 0.5$. The spectral properties of these filters have been examined extensively in Ref. [25] and are summarized in Figs. 2.5 and 2.6, respectively. Briefly, the damping effect decreases with increase in order of accuracy and α_f.

2.5.2 Points Near Boundary

For an 11 point stencil, special boundary formulas are required at points $1, \ldots, 5$ and correspondingly at $N-4, \ldots, N$ where the centered filter stencil protrudes the boundary. Two approaches to treat these points are:

- At any boundary point i, a centered scheme of order $2i-2$ can be easily applied with the same formulas as in the interior (and again setting all free parameters with the exception of α_f to 0). Thus, approaching the boundary from the interior this strategy systematically reduces the order of the filter. In situations where the mesh is refined near the boundary, there is no significant loss of accuracy. Further, α_f can be considered as an optimization parameter and at values close to 0.5 can provide an accurate boundary scheme (see Ref. [9] for details).

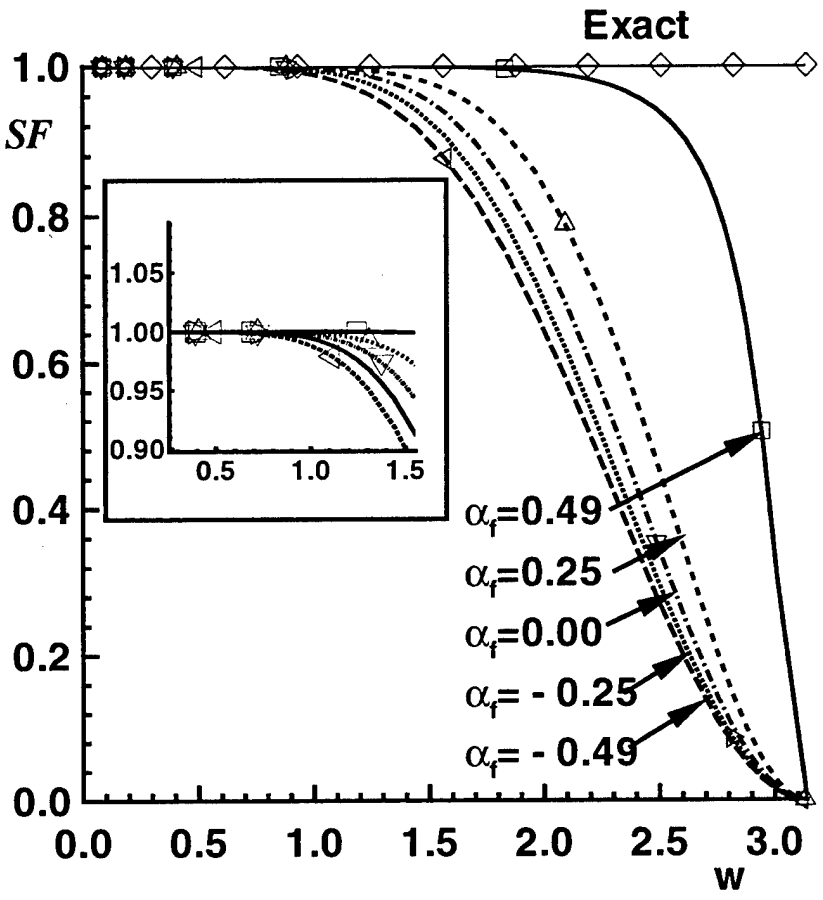

Figure 2.5: Filtering Effect Variation with α_f for $8th$-order Filter

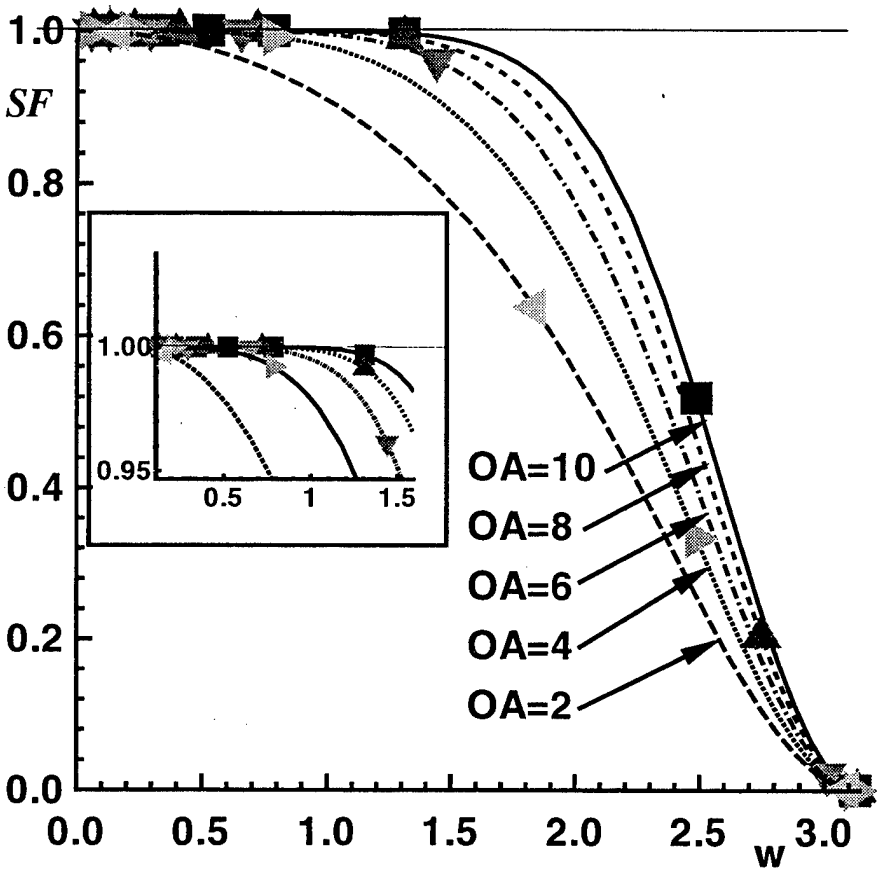

Figure 2.6: Filtering Effect Variation with Order of Accuracy at $\alpha_f = 0.25$

- One-sided filter formulas can be incorporated within the tridiagonal structure of the present scheme. In these, for general values of α_f (still satisfying $|\alpha_f| < 0.5$), the spectral function is complex – indicating that the filter may introduce artificial dispersion as well – and further, the real component is greater than unity over select wave number ranges implying that certain wave numbers will be amplified. Both, the imaginary component of the spectral function as well as the amount of excess over unity of the real part diminish as α_f approaches 0.5 and thus, a value of α_f as close to 0.5 as possible from practical stability observation is recommended. However, it is also important to note that since these schemes are applied at only a few points in the domain, the inaccuracies introduced at higher wavenumbers may be amply compensated by the higher accuracy at smaller wavenumbers. All formulas presented below guarantee suppression of the odd-even mode as for the interior scheme. Each formula is designated $FB_{x,y}$ where x is the point where the formula can be applied and y is the order of accuracy.

Point 5
Formula

$$\alpha_f\bar{\phi}_4 + \bar{\phi}_5 + \alpha_f\bar{\phi}_6 = a\phi_1 + b\phi_2 + c\phi_3 + d\phi_4 + e\phi_5 + f\phi_6 + g\phi_7 + h\phi_8 + i\phi_9 + j\phi_{10} + k\phi_{11} \quad (2.22)$$

Coefficients Table 2.15. Note: Only two schemes are listed since those of lower order can be obtained from Table 2.14 by considering this to be an interior point.

Point 4
Formula

$$\alpha_f\bar{\phi}_3 + \bar{\phi}_4 + \alpha_f\bar{\phi}_5 = a\phi_1 + b\phi_2 + c\phi_3 + d\phi_4 + e\phi_5 + f\phi_6 + g\phi_7 + h\phi_8 + i\phi_9 + j\phi_{10} + k\phi_{11} \quad (2.23)$$

Coefficients: Table 2.16.

Point 3
Formula

$$\alpha_f\bar{\phi}_2 + \bar{\phi}_3 + \alpha_f\bar{\phi}_4 = a\phi_1 + b\phi_2 + c\phi_3 + d\phi_4 + e\phi_5 + f\phi_6 + g\phi_7 + h\phi_8 + i\phi_9 + j\phi_{10} + k\phi_{11} \quad (2.24)$$

Coefficients: Table 2.17.

Point 2
Formula

$$\alpha_f\bar{\phi}_1 + \bar{\phi}_2 + \alpha_f\bar{\phi}_3 = a\phi_1 + b\phi_2 + c\phi_3 + d\phi_4 + e\phi_5 + f\phi_6 + g\phi_7 + h\phi_8 + i\phi_9 + j\phi_{10} + k\phi_{11} \quad (2.25)$$

with the filter coefficients described in Table 2.18.

Point 1
Formula

$$\bar{\phi}_1 + \alpha_f\bar{\phi}_2 = a\phi_1 + b\phi_2 + c\phi_3 + d\phi_4 + e\phi_5 + f\phi_6 + g\phi_7 + h\phi_8 + i\phi_9 + j\phi_{10} + k\phi_{11} \quad (2.26)$$

Table 2.15: Coefficients for Boundary Filter Formulas at Point 5. α_f is a free parameter

Scheme	a	b	c	d	e	f	g	h	i	j	k	OA
$FB_{5,9}$	$\frac{-1+2\alpha_f}{512}$	$\frac{9-18\alpha_f}{512}$	$\frac{9(-1+2\alpha_f)}{128}$	$\frac{21+86\alpha_f}{128}$	$\frac{193}{256}+\frac{63\alpha_f}{128}$	$\frac{63+130\alpha_f}{256}$	$\frac{21(-1+2\alpha_f)}{128}$	$\frac{3(3-6\alpha_f)}{128}$	$\frac{9(-1+2\alpha_f)}{512}$	$\frac{1}{512}-\frac{\alpha_f}{256}$	0	9
$FB_{5,10}$	$\frac{-1+2\alpha_f}{1024}$	$\frac{5(1-2\alpha_f)}{512}$	$\frac{45(-1+2\alpha_f)}{1024}$	$\frac{15+98\alpha_f}{128}$	$\frac{407+210\alpha_f}{512}$	$\frac{63+130\alpha_f}{256}$	$\frac{105(-1+2\alpha_f)}{512}$	$\frac{15(1-2\alpha_f)}{128}$	$\frac{45(-1+2\alpha_f)}{1024}$	$\frac{5(1-2\alpha_f)}{512}$	$\frac{-1+2\alpha_f}{1024}$	10

Table 2.16: Coefficients for Boundary Filter Formulas at Point 4

Scheme	a	b	c	d	e	f	g	h	i	j	k	OA
$FB_{4,7}$	$\frac{1}{128}-\frac{\alpha_f}{64}$	$\frac{-7}{128}+\frac{7\alpha_f}{64}$	$\frac{21}{128}+\frac{43\alpha_f}{64}$	$\frac{93}{128}+\frac{35\alpha_f}{64}$	$\frac{35}{128}+\frac{29\alpha_f}{64}$	$\frac{-21}{128}+\frac{21\alpha_f}{64}$	$\frac{7}{128}-\frac{7\alpha_f}{64}$	$\frac{-1}{128}+\frac{\alpha_f}{64}$	0	0	0	7
$FB_{4,8}$	$\frac{1}{256}-\frac{\alpha_f}{128}$	$\frac{-1}{32}+\frac{\alpha_f}{16}$	$\frac{7}{64}+\frac{25\alpha_f}{32}$	$\frac{25}{32}+\frac{7\alpha_f}{16}$	$\frac{35}{128}+\frac{29\alpha_f}{64}$	$\frac{-7}{32}+\frac{7\alpha_f}{16}$	$\frac{7}{64}-\frac{7\alpha_f}{32}$	$\frac{-1}{32}+\frac{\alpha_f}{16}$	$\frac{1}{256}-\frac{\alpha_f}{128}$	0	0	8
$FB_{4,9}$	$\frac{1}{512}-\frac{\alpha_f}{256}$	$\frac{-9+18\alpha_f}{512}$	$\frac{9+110\alpha_f}{128}$	$\frac{107+42\alpha_f}{128}$	$\frac{63+130\alpha_f}{256}$	$\frac{63(-1+2\alpha_f)}{256}$	$\frac{21(1-2\alpha_f)}{128}$	$\frac{3(-3+6\alpha_f)}{128}$	$\frac{9(1-2\alpha_f)}{512}$	$-\frac{1}{512}+\frac{\alpha_f}{256}$	0	9
$FB_{4,10}$	$\frac{1-2\alpha_f}{1024}$	$\frac{5(-1+2\alpha_f)}{512}$	$\frac{45+934\alpha_f}{1024}$	$\frac{113+30\alpha_f}{128}$	$\frac{105+302\alpha_f}{512}$	$\frac{63(-1+2\alpha_f)}{256}$	$\frac{105(1-2\alpha_f)}{512}$	$\frac{15(-1+2\alpha_f)}{128}$	$\frac{45(1-2\alpha_f)}{1024}$	$\frac{5(-1+2\alpha_f)}{512}$	$\frac{1-2\alpha_f}{1024}$	10

Table 2.17: Coefficients for Boundary Filter Formulas at Point 3

Scheme	a	b	c	d	e	f	g	h	i	j	k	OA
$FB_{3,5}$	$\frac{-1}{32}+\frac{\alpha_f}{16}$	$\frac{5}{32}+\frac{11\alpha_f}{16}$	$\frac{11}{16}+\frac{5\alpha_f}{8}$	$\frac{5}{16}+\frac{3\alpha_f}{8}$	$-\frac{5}{32}+\frac{5\alpha_f}{16}$	$\frac{1}{32}-\frac{\alpha_f}{16}$	0	0	0	0	0	5
$FB_{3,6}$	$-\frac{1}{64}+\frac{\alpha_f}{32}$	$\frac{3}{32}+\frac{13\alpha_f}{16}$	$\frac{49}{64}+\frac{15\alpha_f}{32}$	$\frac{5}{16}+\frac{3\alpha_f}{8}$	$-\frac{15}{64}+\frac{15\alpha_f}{32}$	$\frac{3}{32}-\frac{3\alpha_f}{16}$	$-\frac{1}{64}+\frac{\alpha_f}{32}$	0	0	0	0	6
$FB_{3,7}$	$-\frac{1}{128}+\frac{\alpha_f}{64}$	$\frac{7}{128}+\frac{57\alpha_f}{64}$	$\frac{107}{128}+\frac{21\alpha_f}{64}$	$\frac{35}{128}+\frac{29\alpha_f}{64}$	$\frac{7(-5+10\alpha_f)}{128}$	$\frac{21}{128}-\frac{21\alpha_f}{64}$	$-\frac{7}{128}+\frac{7\alpha_f}{64}$	$\frac{1}{128}-\frac{\alpha_f}{64}$	0	0	0	7
$FB_{3,8}$	$\frac{-1}{256}+\frac{\alpha_f}{128}$	$\frac{1}{32}+\frac{15\alpha_f}{16}$	$\frac{57}{64}+\frac{7\alpha_f}{32}$	$\frac{7}{32}+\frac{9\alpha_f}{16}$	$\frac{7(-5+10\alpha_f)}{128}$	$\frac{7}{32}-\frac{7\alpha_f}{16}$	$-\frac{7}{64}+\frac{7\alpha_f}{32}$	$\frac{1}{32}-\frac{\alpha_f}{16}$	$-\frac{1}{256}+\frac{\alpha_f}{128}$	0	0	8
$FB_{3,9}$	$-\frac{1}{512}+\frac{\alpha_f}{256}$	$\frac{9+494\alpha_f}{512}$	$\frac{119+18\alpha_f}{128}$	$\frac{21}{128}+\frac{43\alpha_f}{64}$	$\frac{21(-3+6\alpha_f)}{256}$	$\frac{63(1-2\alpha_f)}{256}$	$\frac{21(-1+2\alpha_f)}{128}$	$\frac{3(3-6\alpha_f)}{128}$	$\frac{9(-1+2\alpha_f)}{512}$	$\frac{1}{512}-\frac{\alpha_f}{256}$	0	9
$FB_{3,10}$	$\frac{-1+2\alpha_f}{1024}$	$\frac{5+502\alpha_f}{512}$	$\frac{979+90\alpha_f}{1024}$	$\frac{15+98\alpha_f}{128}$	$\frac{105(-1+2\alpha_f)}{512}$	$\frac{63(1-2\alpha_f)}{256}$	$\frac{105(-1+2\alpha_f)}{512}$	$\frac{15(1-2\alpha_f)}{128}$	$\frac{45(-1+2\alpha_f)}{1024}$	$\frac{5(1-2\alpha_f)}{512}$	$\frac{-1+2\alpha_f}{1024}$	10

Table 2.18: Coefficients for Boundary Filter Formulas at Point 2

Scheme	a	b	c	d	e	f	g	h	i	j	k	OA
$FB_{2,3}$	$\frac{1}{8}+\frac{3\alpha_f}{4}$	$\frac{5}{8}+\frac{3\alpha_f}{4}$	$\frac{3}{8}+\frac{\alpha_f}{4}$	$-\frac{1}{8}+\frac{\alpha_f}{4}$	0	0	0	0	0	0	0	3
$FB_{2,4}$	$\frac{1}{16}+\frac{7\alpha_f}{8}$	$\frac{3}{4}+\frac{\alpha_f}{2}$	$\frac{3}{8}+\frac{\alpha_f}{4}$	$-\frac{1}{4}+\frac{\alpha_f}{2}$	$\frac{1}{16}-\frac{\alpha_f}{8}$	0	0	0	0	0	0	4
$FB_{2,5}$	$\frac{1}{32}+\frac{15\alpha_f}{16}$	$\frac{27}{32}+\frac{5\alpha_f}{16}$	$\frac{5}{16}+\frac{3\alpha_f}{8}$	$-\frac{5}{16}+\frac{5\alpha_f}{8}$	$\frac{5}{32}-\frac{5\alpha_f}{16}$	$-\frac{1}{32}+\frac{\alpha_f}{16}$	0	0	0	0	0	5
$FB_{2,6}$	$\frac{1}{64}+\frac{31\alpha_f}{32}$	$\frac{29}{32}+\frac{3\alpha_f}{16}$	$\frac{15}{64}+\frac{17\alpha_f}{32}$	$-\frac{5}{16}+\frac{5\alpha_f}{8}$	$\frac{15}{64}-\frac{15\alpha_f}{32}$	$-\frac{3}{32}+\frac{3\alpha_f}{16}$	$\frac{1}{64}-\frac{\alpha_f}{32}$	0	0	0	0	6
$FB_{2,7}$	$\frac{1}{128}+\frac{63\alpha_f}{64}$	$\frac{121}{128}+\frac{7\alpha_f}{64}$	$\frac{21}{128}+\frac{43\alpha_f}{64}$	$-\frac{35}{128}+\frac{35\alpha_f}{64}$	$\frac{7(5-10\alpha_f)}{128}$	$-\frac{21}{128}+\frac{21\alpha_f}{64}$	$\frac{7}{128}-\frac{7\alpha_f}{64}$	$-\frac{1}{128}+\frac{\alpha_f}{64}$	0	0	0	7
$FB_{2,8}$	$\frac{1}{256}+\frac{127\alpha_f}{128}$	$\frac{31}{32}+\frac{\alpha_f}{16}$	$\frac{7}{64}+\frac{25\alpha_f}{32}$	$-\frac{7}{32}+\frac{7\alpha_f}{16}$	$\frac{7(5-10\alpha_f)}{128}$	$-\frac{7}{32}+\frac{7\alpha_f}{16}$	$\frac{7}{64}-\frac{7\alpha_f}{32}$	$-\frac{1}{32}+\frac{\alpha_f}{16}$	$\frac{1}{256}-\frac{\alpha_f}{128}$	0	0	8
$FB_{2,9}$	$\frac{1}{512}+\frac{255\alpha_f}{256}$	$\frac{503+18\alpha_f}{512}$	$\frac{9+110\alpha_f}{128}$	$\frac{3(-7+14\alpha_f)}{128}$	$\frac{21(3-6\alpha_f)}{256}$	$\frac{63(-1+2\alpha_f)}{256}$	$\frac{21(1-2\alpha_f)}{128}$	$\frac{3(-3+6\alpha_f)}{128}$	$\frac{9(1-2\alpha_f)}{512}$	$-\frac{1}{512}+\frac{\alpha_f}{256}$	0	9
$FB_{2,10}$	$\frac{1+1022\alpha_f}{1024}$	$\frac{507+10\alpha_f}{512}$	$\frac{45+934\alpha_f}{1024}$	$\frac{15(-1+2\alpha_f)}{128}$	$\frac{105(1-2\alpha_f)}{512}$	$\frac{63(-1+2\alpha_f)}{256}$	$\frac{105(1-2\alpha_f)}{512}$	$\frac{15(-1+2\alpha_f)}{128}$	$\frac{45(1-2\alpha_f)}{1024}$	$\frac{5(-1+2\alpha_f)}{512}$	$\frac{1-2\alpha_f}{1024}$	10

Table 2.19: Coefficients for Boundary Filter Formulas at Point 1

Scheme	a	b	c	d	e	f	g	h	i	j	k	OA
$FB_{1,1}$	$\frac{1}{2}+\frac{\alpha_f}{2}$	$\frac{1}{2}+\frac{\alpha_f}{2}$	0	0	0	0	0	0	0	0	0	1
$FB_{1,2}$	$\frac{3}{4}+\frac{\alpha_f}{4}$	$\frac{1}{2}+\frac{\alpha_f}{2}$	$-\frac{1}{4}+\frac{\alpha_f}{4}$	0	0	0	0	0	0	0	0	2
$FB_{1,3}$	$\frac{7}{8}+\frac{\alpha_f}{8}$	$\frac{3}{8}+\frac{5\alpha_f}{8}$	$-\frac{3}{8}+\frac{3\alpha_f}{8}$	$\frac{1}{8}-\frac{\alpha_f}{8}$	0	0	0	0	0	0	0	3
$FB_{1,4}$	$\frac{15}{16}+\frac{\alpha_f}{16}$	$\frac{1}{4}+\frac{3\alpha_f}{4}$	$-\frac{3}{8}+\frac{3\alpha_f}{8}$	$\frac{1}{4}-\frac{\alpha_f}{4}$	$-\frac{1}{16}+\frac{\alpha_f}{16}$	0	0	0	0	0	0	4
$FB_{1,5}$	$\frac{31}{32}+\frac{\alpha_f}{32}$	$\frac{5}{32}+\frac{27\alpha_f}{32}$	$-\frac{5}{16}+\frac{5\alpha_f}{16}$	$\frac{5}{16}-\frac{5\alpha_f}{16}$	$-\frac{5}{32}+\frac{5\alpha_f}{32}$	$\frac{1}{32}-\frac{\alpha_f}{32}$	0	0	0	0	0	5
$FB_{1,6}$	$\frac{63}{64}+\frac{\alpha_f}{64}$	$\frac{3}{32}+\frac{29\alpha_f}{32}$	$-\frac{15}{64}+\frac{15\alpha_f}{64}$	$\frac{5}{16}-\frac{5\alpha_f}{16}$	$-\frac{15}{64}+\frac{15\alpha_f}{64}$	$\frac{3}{32}-\frac{3\alpha_f}{32}$	$-\frac{1}{64}+\frac{\alpha_f}{64}$	0	0	0	0	6
$FB_{1,7}$	$\frac{127}{128}+\frac{\alpha_f}{128}$	$\frac{7}{128}+\frac{121\alpha_f}{128}$	$-\frac{21}{128}+\frac{21\alpha_f}{128}$	$\frac{35}{128}-\frac{35\alpha_f}{128}$	$\frac{7(-5+5\alpha_f)}{128}$	$\frac{21}{128}-\frac{21\alpha_f}{128}$	$-\frac{7}{128}+\frac{7\alpha_f}{128}$	$\frac{1}{128}-\frac{\alpha_f}{128}$	0	0	0	7
$FB_{1,8}$	$\frac{255}{256}+\frac{\alpha_f}{256}$	$\frac{1}{32}+\frac{31\alpha_f}{32}$	$-\frac{7}{64}+\frac{7\alpha_f}{64}$	$\frac{7}{32}-\frac{7\alpha_f}{32}$	$\frac{7(-5+5\alpha_f)}{128}$	$\frac{7}{32}-\frac{7\alpha_f}{32}$	$\frac{7(-1+\alpha_f)}{64}$	$\frac{1}{32}-\frac{\alpha_f}{32}$	$-\frac{1}{256}+\frac{\alpha_f}{256}$	0	0	8
$FB_{1,9}$	$\frac{511+\alpha_f}{512}$	$\frac{9+503\alpha_f}{512}$	$\frac{9(-1+\alpha_f)}{128}$	$\frac{3(7-7\alpha_f)}{128}$	$\frac{21(-3+3\alpha_f)}{256}$	$\frac{63(1-\alpha_f)}{256}$	$\frac{21(-1+\alpha_f)}{128}$	$\frac{3(3-3\alpha_f)}{128}$	$\frac{9(-1+\alpha_f)}{512}$	$\frac{1-\alpha_f}{512}$	0	9
$FB_{1,10}$	$\frac{1023+\alpha_f}{1024}$	$\frac{5+507\alpha_f}{512}$	$\frac{45(-1+\alpha_f)}{1024}$	$\frac{15(1-\alpha_f)}{128}$	$\frac{105(-1+\alpha_f)}{512}$	$\frac{63(1-\alpha_f)}{256}$	$\frac{105(-1+\alpha_f)}{512}$	$\frac{15(1-\alpha_f)}{128}$	$\frac{45(-1+\alpha_f)}{1024}$	$\frac{5(1-\alpha_f)}{512}$	$\frac{-1+\alpha_f}{1024}$	10

Coefficients: Table 2.19.

The filter formulas for the right boundary may be derived in similar fashion and indeed, the same coefficients are obtained. For completeness, these are listed below.

Point N-4

Formula

$$\alpha_f\bar{\phi}_{N-5} + \bar{\phi}_{N-4} + \alpha_f\bar{\phi}_{N-3} = a\phi_N + b\phi_{N-1} + c\phi_{N-2} + d\phi_{N-3} + e\phi_{N-4} + f\phi_{N-5} + g\phi_{N-6} + h\phi_{N-7} + i\phi_{N-8} + j\phi_{N-9} + k\phi_{N-10} \quad (2.27)$$

Coefficients: Table 2.15.

Point N-3

Formula

$$\alpha_f\bar{\phi}_{N-4} + \bar{\phi}_{N-3} + \alpha_f\bar{\phi}_{N-2} = a\phi_N + b\phi_{N-1} + c\phi_{N-2} + d\phi_{N-3} + e\phi_{N-4} + f\phi_{N-5} + g\phi_{N-6} + h\phi_{N-7} + i\phi_{N-8} + j\phi_{N-9} + k\phi_{N-10} \quad (2.28)$$

Coefficients: Table 2.16.

Point N-2

Formula

$$\alpha_f\bar{\phi}_{N-3} + \bar{\phi}_{N-2} + \alpha_f\bar{\phi}_{N-1} = a\phi_N + b\phi_{N-1} + c\phi_{N-2} + d\phi_{N-3} + e\phi_{N-4} + f\phi_{N-5} + g\phi_{N-6} + h\phi_{N-7} + i\phi_{N-8} + j\phi_{N-9} + k\phi_{N-10} \quad (2.29)$$

Coefficients: Table 2.17.

Point N-1

Formula

$$\alpha_f\bar{\phi}_{N-2} + \bar{\phi}_{N-1} + \alpha_f\bar{\phi}_N = a\phi_N + b\phi_{N-1} + c\phi_{N-2} + d\phi_{N-3} + e\phi_{N-4} + f\phi_{N-5} + g\phi_{N-6} + h\phi_{N-7} + i\phi_{N-8} + j\phi_{N-9} + k\phi_{N-10} \quad (2.30)$$

Coefficients: Table 2.18.

Point N

Formula

$$\alpha_f\bar{\phi}_{N-1} + \bar{\phi}_N = a\phi_N + b\phi_{N-1} + c\phi_{N-2} + d\phi_{N-3} + e\phi_{N-4} + f\phi_{N-5} + g\phi_{N-6} + h\phi_{N-7} + i\phi_{N-8} + j\phi_{N-9} + k\phi_{N-10} \quad (2.31)$$

Coefficients: Table 2.19.

2.6 Boundary Formulas for Neumann Conditions

Dirichlet boundary conditions at the endpoints 1 and N for the physical (or Phase II) conditions are straightforward to implement in the present finite-difference scheme. Neumann conditions, such

Table 2.20: Coefficients for Specification of $\partial\phi/\partial x = 0$

OA	a	b	c	d	e	f	A
6	360	−450	400	−225	72	−10	147
5	300	−300	200	−75	12	0	137
4	48	−36	16	−3	0	0	25
3	18	−9	2	0	0	0	11
2	4	−1	0	0	0	0	3
1	1	0	0	0	0	0	1

as the zero pressure gradient condition, require higher order formulas. The following formulas sets the value of ϕ_1 in terms of interior points in order to enforce $\partial\phi/\partial n = 0$.

Point 1

Formula

$$\phi_1 = \frac{a\phi_2 + b\phi_3 + c\phi_4 + d\phi_5 + e\phi_6 + \phi_7}{A} \tag{2.32}$$

Coefficients: Table 2.20.

Point N

Formula

$$\phi_N = \frac{a\phi_{N-1} + b\phi_{N-2} + d\phi_{N-3} + e\phi_{N-4} + f\phi_{N-5} + g\phi_{N-6}}{A} \tag{2.33}$$

Coefficients: Table 2.20

Chapter 3

Implementation for Navier-Stokes Equations

The FDL3DI code was chosen to implement the formulas described in the previous chapters. Several versions of this code exist, some including capabilities for turbulent simulations, aeroelastic analysis and with overset grid generality. Some of these versions have also been parallelized. To provide a test bed to investigate the properties of the previously developed interior differencing and filter schemes, an elementary version was chosen with the following properties:

- Inviscid fluxes: Second-order explicit centered or Roe's flux-difference split scheme with all MUSCL-based options.
- Implicit time integration through the original Beam-Warming approximately factored algorithm [26] with or without the more economical diagonal procedure of Pulliam *et al.* [8].
- Subiteration capability to reduce errors introduced by linearization, approximate factorization and explicit boundary condition implementation.

To this were added the above compact differencing and filtering capabilities with the additional option of explicit time-integration with the classical Runge-Kutta scheme in the low-storage form described in Ref. [27]. The resulting version of the code has been designated FDL3D_ICE.

3.1 Governing Equations

The code solves the full, unsteady, three-dimensional Navier-Stokes equations [28]. With ξ, η, ζ and t representing a general curvilinear transformation, and normalizing with the quantities by ρ_∞, U_∞, a representative length L, and the free stream molecular viscosity μ, the equations may be written

in vector form as:

$$\frac{\partial}{\partial t}\left(\frac{U}{J}\right)+\frac{\partial \bar{F}_I}{\partial \xi}+\frac{\partial \bar{G}_I}{\partial \eta}+\frac{\partial \bar{H}_I}{\partial \zeta}=\frac{1}{Re}[\frac{\partial \bar{F}_v}{\partial \xi}+\frac{\partial \bar{G}_v}{\partial \eta}+\frac{\partial \bar{H}_v}{\partial \zeta}] \tag{3.1}$$

where

$$U=\begin{bmatrix}\rho\\ \rho u\\ \rho v\\ \rho w\\ \rho E_t\end{bmatrix},\quad \bar{F}_I=\frac{1}{J}\begin{bmatrix}\rho U\\ \rho uU+\xi_x p\\ \rho vU+\xi_y p\\ \rho wU+\xi_z p\\ \rho E_t U+p\tilde{U}\end{bmatrix},\quad \bar{G}_I=\frac{1}{J}\begin{bmatrix}\rho V\\ \rho uV+\eta_x p\\ \rho vV+\eta_y p\\ \rho wV+\eta_z p\\ \rho E_t V+p\tilde{V}\end{bmatrix},\quad \bar{H}_I=\frac{1}{J}\begin{bmatrix}\rho W\\ \rho uW+\zeta_x p\\ \rho vW+\zeta_y p\\ \rho wW+\zeta_z p\\ \rho E_t W+p\tilde{W}\end{bmatrix},$$

$$\bar{F}_v=\frac{1}{J}\begin{bmatrix}0\\ \xi_x,\tau_{i1}\\ \xi_x,\tau_{i2}\\ \xi_x,\tau_{i3}\\ \xi_x,b_i\end{bmatrix},\quad \bar{G}_v=\frac{1}{J}\begin{bmatrix}0\\ \eta_x,\tau_{i1}\\ \eta_x,\tau_{i2}\\ \eta_x,\tau_{i3}\\ \eta_x,b_i\end{bmatrix},\quad \bar{H}_v=\frac{1}{J}\begin{bmatrix}0\\ \zeta_x,\tau_{i1}\\ \zeta_x,\tau_{i2}\\ \zeta_x,\tau_{i3}\\ \zeta_x,b_i\end{bmatrix}$$

U, V and W are contravariant components of velocity, defined by:

$$U=\xi_t+\xi_x u+\xi_y v+\xi_z w=\xi_t+\tilde{U}$$

$$V=\eta_t+\eta_x u+\eta_y v+\eta_z w=\eta_t+\tilde{V}$$

$$W=\zeta_t+\zeta_x u+\zeta_y v+\zeta_z w=\zeta_t+\tilde{W}$$

$$E_t=e+\frac{1}{2}(u^2+v^2+w^2)$$

Using the compact notation x_i, $i=1\ldots3$ to represent the x, y and z coordinates respectively and similarly ξ_i for ξ,η,ζ, the shear stress tensor, τ can be written as

$$\tau_{ij}=\mu(\frac{\partial \xi_k}{\partial x_j}\frac{\partial u_i}{\partial \xi_k}+\frac{\partial \xi_k}{\partial x_i}\frac{\partial u_j}{\partial \xi_k})-\frac{2}{3}\mu\delta_{ij}\frac{\partial \xi_l}{\partial x_k}\frac{\partial u_k}{\partial \xi_l}$$

while

$$b_i=u_j\tau_{ij}+\frac{k}{(\gamma-1)PrM_\infty^2}\frac{\partial \xi_l}{\partial x_i}\frac{\partial T}{\partial \xi_l}$$

Note that the pressure is normalized by the quantity $\rho_\infty u_\infty^2$. The equation of state for a perfect gas is assumed, the Prandtl number is fixed at 0.72 and the molecular viscosity is obtained from Sutherland's law.

3.2 Implementation

3.2.1 Metrics and Inviscid Fluxes

The metrics are evaluated in a straightforward fashion by utilizing the formulas of Chapter 2. Following the recommendations of Refs. [29, 25], it is recommended that the same equations be

utilized for metrics as for the fluxes, since this results in an error cancelation effect. The derivatives of the inviscid fluxes are obtained by first forming these fluxes at the nodes and subsequently employing the formulas of Section 2.1. For computational efficiency, the derivatives in an entire plane are obtained in a vectorized fashion. To maintain the overall structure of the original code, the $\frac{\partial F_I}{\partial \xi}$ and $\frac{\partial H_I}{\partial \zeta}$ values are obtained in a η plane by vectorizing in the ζ and ξ directions respectively. Similarly, $\frac{\partial G}{\partial \eta}$ are formed in ζ planes by vectorizing in ξ .

3.2.2 Viscous Fluxes

The formation of the viscous fluxes is more complicated since second derivatives are involved. In conservative form, and with x, y and z represented by $x_i, i = 1, 2, 3$ respectively, these terms have the form $\frac{\partial}{\partial x_i}\left(\mu \frac{\partial s}{\partial x_j}\right)$ where μ is the molecular viscosity and s denotes one of the primitive variables u, v, w, T. At present, these terms are formed by succesive application of the first derivative operators of Section 2.1. To avoid repeated computations of derivatives, the various terms are formed in different subroutines which follow, with minor exception, the original layout of the FDL3DI code. The specific routines are labeled VCMPXZRHS and VCMPYRHS with some terms being computed in the subroutine VCROSX.

It has been suggested that the successive application of two first derivative operators can lead to instabilities since the damping mechanism for odd-even modes may be absent. Of course, filtering guarantees that this mode is suppressed and indeed no evidence of such an instability was found in the extensive tests reported in Ref. [9]. The suggested alternative to the successive application of first difference operators is to write the chain-rule form of the viscous terms which can then be directly computed with formulas for second derivatives. We note however that this is an expensive approach either in CPU or memory requirement depending upon implementation, particularly in curvilinear coordinates.

An approach of intermediate complexity can be suggested with the use of the midpoint formulas presented previously (Sections 2.2 through 2.4). The cross-derivative terms can be computed as before. On the other hand, the "straight" derivative terms such as for example,

$$\frac{\partial}{\partial x_i}(\mu \frac{\partial s}{\partial x_i}) = (\mu s')'$$

can be discretized by first forming the product of the midpoint values of μ (with the formulas of Section 2.2) and s' (formulas of Section 2.3). These terms of the stress tensor can then be differentiated to obtain values at the nodes with the dual formulas of Section 2.4. The result degenerates to the original second order method with the appropriate choice of coefficients (scheme *E2* of Table 2.1).

3.3 Fourth-order Runge-Kutta Method

The classical fourth-order four-stage Runge-Kutta method has been incorporated into FDL3D_ICE. With R denoting the residual, the governing equation is:

$$\frac{\partial U}{\partial t} = R = -\left(\frac{\partial (F_I + F_V)}{\partial \xi} + \frac{\partial (G_I + G_V)}{\partial \eta} + \frac{\partial (H_I + H_V)}{\partial \zeta}\right)$$

The classical four stage method may be written as [17]:

$$\begin{aligned} k_0 &= \Delta t R(U_0) \quad k_1 = \Delta t R(U_1) \\ k_2 &= \Delta t R(U_2) \quad k_3 = \Delta t R(U_3) \\ U^{n+1} &= U_0 + \frac{1}{6}(k_0 + 2k_1 + 2k_2 + k_3) \end{aligned} \tag{3.2}$$

where $U_0 = U^n$, $U_1 = U_0 + k_0/2$, $U_2 = U_1 + k_1/2$, $U_3 = U_2 + k_2$. The scheme is implemented in the low storage form described in Ref. [27]. The residual R may be computed with any of the schemes outlined above.

Appendix: Input variables for FDL3D_ICE

The specification of input variables is illustrated with a sample input file. Since the details of this file are subject to change on a frequent basis, the description is simply to provide a flavor for the required data and various steps necessary to formulate a computation. The input file contains comments either as complete lines or as trailing characters which describe the variable name being read. In the first case, the line is read by the code *i.e.*, it is a necessary record though its contents are unimportant. In the latter case, the data is not read at all and such trailing alphabetical characters may be deleted if desired.

A few routines, such as the one applying boundary conditions, are case-specific and must be modified for each configuration. These routines are identified below in the appropriate context. Each input noted below must be specified, even if it is not used by the particular scheme selected. For example, if one of the variants of the Roe scheme is chosen, the compact difference stencil is obviously not utilized but must necessarily be included in the input file since it is read (though not utilized).

In the following, lines in the input file are entered in typewriter font.

```
4   TEST_CASE
```

Variable TEST_CASE determines the case being computed. This term is only employed in SUBROUTINE BNDRY, which must be written by the user for each case.

```
0.1       XM1
100.0     RE
1.002     TW
0.38      S1
0.0       ALFA
```

$XM1$ is the Mach number and RE is the Reynolds number. TW denotes the wall temperature T_w, normalized by the free stream temperature, and is employed only in SUBROUTINE BNDRY. The convention recommended is that a negative TW signifies an adiabatic wall. $S1$ is the molecular

viscosity coefficient for Sutherlands law ($S1 = 198.6/T_\infty$(Rankine)). $ALFA$ is the angle of attack and, like TW, is only employed in the user provided SUBROUTINE BNDRY.

```
99999  1.  0   NDTAU  CFL IBETA
```

NDTAU is the number of steps between update of timestep size (calls to SUBROUTINE TMSTEP) and is meaningful only if $IBETA \neq 0$. CFL is the Courant number. IBETA controls whether the time-step size is controlled with the CFL number or if constant time-step size values DTVIS, DTFIX (below) are employed. For further details, SUBROUTINE TMSTEP should be consulted.

```
0.01    0.01  DTVIS  DTFIX
```

DTVIS is the time-step size if IBETA=0 and the minimum time step size in the entire domain if local stepping is chosen. If subiterations are on, DTFIX becomes the outer time step size while DTVIS is the step of the "inner" iterations.

```
3  IDMPFIL
```

This parameter controls filtering/damping execution and consolidates several coefficients of the previous version of the code including ICDAMP and ISDAMP as follows:

IDMPFIL = 1 (formerly ICDAMP=1, constant coefficient damping, now obsolete)
= 2 (formerly ISDAMP=1), scalar spectral damping controlled by ES4,ES2, FES4I, FES2I, OMGAV and SRCONST as before
= 3 Use filtering after each subiteration(Note implicit damping continues to be controlled by ES4,ES2, FES4I, FES2D, OMGAV and SRCONST as before)
= 4 Use filtering after all subiterations are completed (Note implicit damping continues to be controlled by ES4,ES2, FES4I, FES2D, OMGAV and SRCONST as before)

```
0.01   0.0  ES4  ES2
2.0   1.0   1.0    25.  FES4I  FES2I  OMGAV  SRCONST
```

These are damping coefficient parameters. These values are always relevant for implicit approximate factorization part of the algorithm, provided the time-integration scheme is implicit (controlled by IRUNGE below).

```
0.  PHI
```

PHI controls the order of accuracy of the *implicit* time integration algorithm.

```
1   1   1   1   IV1ON  IV2ON  IV3ON  IVMON
```

These parameters control the computation of viscous terms as in the previous version of the code: IV1ON(IV2ON,IV3ON)=1 switches on thin layer terms in I(J,K) directions respectively. IVMON=1 switches on the cross derivative terms as well. For Euler calculations, IV1ON=IV2ON=IV3ON=0.

```
0    1    IDIAG  NCONV
```

IDIAG=1 invokes the diagonalized option of Ref. [8] while a zero value reverts to the original Beam-Warming scheme. NCONV controls the number of steps between runtime output (see SUBROUTINE WRTHIST or SUBROUTINE MONITR).

```
1   0    0   400   IRUNGE ISUBON  NSUBMX  INMAX
```

IRUNGE controls time integration (0 for Beam-Warming approximate factorization), 1 for RK4. Note that if IRUNGE=1 (*i.e.*, an explicit scheme), many other inputs which control the implicit algorithm are moot (*e.g.*, IDIAG, ISUBON, NSUBMX). ISUBON=1 switches on subiterations whose number is controlled by NSUBMX. The parameter is ignored if IRUNGE=1. INMAX is total number of iterations for the present run.

```
15 15 15 1 1  ISCHME  JSCHME  KSCHME  IVISC IMETRC
```

ISCHEME, JSCHEME and KSCHEME (previously IROE, JROE, KROE) control the choice of scheme as follows:

0	Original second-order central scheme
1	First order Roe
2	Fully-upwind second order Roe (using MUSCL with $\kappa = -1$)
3	Second order Roe scheme with Fromm reconstruction ($\kappa = 0$)
4	Third order upwind-biased Roe scheme ($\kappa = 1/3$)
5	Second order central subset of the Roe/MUSCL scheme ($\kappa = 1$)
15	Compact schemes as determined with the input variables below

For the Roe/MUSCL variants, the limiter may be varied by modifying MODULE LIMTRS in the Fortran-90 version of the code or in SUBROUTINES XIRECON,ETRECON and ZTRECON of the Fortran-77 version. IVISC controls accuracy of evaluation of viscous terms. The value 0 chooses the second order central formulation of the original code, while 1 computes the derivatives in a compact fashion. Note i) the value of IVISC is ignored for the $I(J,K)$ direction if IV1ON(IV2ON,IV3ON)=0. Also, ii) different schemes can be employed in different directions, including different variations of the compact scheme. However, for any given direction, the viscous and inviscid fluxes are computed with the same scheme. Finally, IMETRC determines the difference formula for the metrics. IMETRC=0 reverts to the original second-order scheme, 1 chooses the compact scheme determined specially for the metrics (below), while 2 causes the code to compute the metrics in each direction with the same formula as for the inviscid and viscous fluxes in that direction.

```
5.0E-2   5.0E-2   5.0E-2  XICUTOFF  ETACUTOFF  ZETACUTOFF
0 0 0  IXIISO  IETAISO  IZETAISO
0 0 0 0 0 0 0 0 0 ICUT1 ICUT2 ICUT3 JCUT1 JCUT2 JCUT3 KCUT1 KCUT2 KCUT3
```

These parameters control the entropy cutoff for the Roe scheme for each of the three directions. Detailed formulas can be obtained from SUBROUTINE XIROEFLUX. For example, in the I direction,

XICUTOFF controls the magnitude of the cutoff, IXIISO, 0 or 1, controls the choice of isotropic or anisotropic formulas while ICUT1, ICUT2 and ICUT3 are on/off switches for the linear (u) and nonlinear ($u+c$ and $u-c$) eigenvalues respectively.

```
                              !  THIS LINE BLANK
FILE NAMES: INPUTD (FOR RECORD ONLY),RESTART,RESTORE,SAVETMP
'vc3d.dat' 'vc3d.in' 'vc3d.out' 'vc3d.tmp'
```

These are names of files: vc3d.dat is the input data file file. The code must be executed for example with the command "a.out < vc3d.dat". vc3d.in is the restart file (must exist at the start of the job with a flow field - see PROGRAM FDL3D_ICE), vc3d.out is the restore file (created at the end of the job). vc3d.tmp is restore file dumped every 50 iterations (see PROGRAM FDL3D_ICE).

```
                              !  THIS LINE BLANK
MOVIE INFO: IMOVIE, MODMOVIE IMV(MIN/MAX/INT),JMV(MIN/MAX/INT),KMV(MIN/MAX/INT)
1  25  4 4 4 -1 -1 -1 -1,-1,-1
MOVIE FILE: GRID, ROOT NAME
'movie.grid' 'vc3dmv.'
```

These parameters control the output of files which can be combined to form a movie. The details are highly case-specific and may be tailored by appropriate modification of SUBROUTINE MOVIE.

```
                              !  THIS LINE BLANK
INFORMATION ABOUT COMPACT DIFFERENCING SCHEME
    ****GRID****
C4-AC4-C4-AC4-C4
    ****X DIRECTION****
C4-AC4-C4-AC4-C4
    ****Y DIRECTION****
C4-AC4-C4-AC4-C4
    ****Z DIRECTION****
C4-AC4-C4-AC4-C4
```

These inputs select the compact scheme for the grid, I, J and K direction respectively. The character variable consisting of five fields designating the scheme to be employed at points 1, 2, interior, $N-1$ and N, respectively. For example,

```
C4-CC4-C6-AC4-C3
```

requests the fourth order compact scheme at point 1 (Table 2.2), the decoupled fourth order compact scheme at point 2 (Table 2.5), compact sixth-order in the interior, the symmetric compact fourth order scheme at point $N-1$ (Section 2.1.4) and the third-order compact scheme at point N

(Section 2.1.5). Note: For efficiency, if the interior scheme is explicit, the tridiagonal system is not solved. Thus, it is not possible to combine explicit interior schemes with implicit boundary schemes.

```
                                ! THIS LINE BLANK
PERIODIC BC INFO: IPERDC,JPERDC,KPERDC
1 1 1
```

If 1, IPERDC, JPERDC and KPERDC implement periodic boundary conditions in the i, j and k directions respectively. For compact schemes, this requires the solution of periodic tridiagonal systems. Note that a 5 pt overlap is employed such that point 1 and 2 corresponds to $IEND-4$ and $IEND-3$, respectively while points $IEND-1$ and $IEND$ correspond to 4 and 5 IPERDC, JPERDC and KPERDC only affect the compact differencing and filtering algorithms. For the original scheme (ISCHME/JSCHME/KSCHME=0), it is necessary to explicitly set periodic conditions in SUBROUTINE BNDRY.

```
                                ! THIS LINE BLANK
RK4 INFORMATION
KSTAGES, COEFFICIENTS as in CODE
4 1. 6. 1. 3. 1. 3. 1. 6. 1. 2. 1. 2. 1. 1. 1. 2. 0. 1. 1. 1.
```

These inputs are the coefficients of the RK4 scheme and should not be modified.

The next inputs control the application of the filter. Note that if IDMPFIL=2, the filter is not invoked in any direction.

```
                                ! THIS LINE BLANK
FILTER INFORMATION
IGFILTER,JGFILTER,KGFILTER
1 1 1
INOFIL,JNOFIL,KNOFIL
1 1 1
```

At the highest level, filtering can be switched on every N iterations in the I(J,K) directions by choosing IGFILTER(JGFILTER,KGFILTER)=N. If IGFILTER=4 (for example), the filter is executed every 4th iteration during which the filter is applied INOFIL number of times in succession. The significance of JNOFIL and KNOFIL is similar.

```
FOR I DIRECTION: INTERIOR FILTER ORDER, ALPHA, NOOPTPARM, OPTPARM(1),...,OPTPARM(NOOPTPARM)
10 0.4999 0
```

These inputs control the filter formulas employed in the I direction for "interior" points. The first parameter is the order of accuracy while the second corresponds to α_f. The 10th order filter (the highest considered here) requires a stencil of 11 points. If a lower order filter is specified, it is possible

to provide free parameters *e.g.*, for a stencil of 11 points, an $8th$ order scheme permits one parameter to be free. In Table 2.14, these free variables (NOOPTPARM in number) are assumed to be zero since their effect on the filter has not yet been investigated. A nonzero NOOPTPARM requires specification of the appropriate number of free parameters (OPTPARM). For details, SUBROUTINES FILCOEFS and BCFILCOEFS should be consulted.

For an 11 point stencil, interior points are those at least 6 points away from the boundary. It is thus necessary to specify the formulation at the 5 points near the boundary. As noted earlier, at each of these points the specification can proceed in one of two ways. The first approach takes recourse to the fact that points near the boundary can be considered as interior points for lower orders of accuracy. Thus, at a point N, a centered $2N$ order formula is easily constructed from Table 2.7. In the second approach, one of the biased formulas presented in Section 2.5.2 may be utilized. To allow for each, the code inputs are specified for the I-direction as follows.

```
Point 1:  ORDER, ALPHA\_F, NOOPT, OPTPARMS (IF NOOPT > 0)
0 0.0 0
Point 2
2 0.4999 0
Point 3
4 0.499 0
Point 4
6 0.49 0
Point 5
8 0.49 0
```

Only the first two inputs – order of accuracy and α_f value – are relevant for the options described here. In this example, the order of accuracy is reduced on approaching the boundary so that the formulas are always centered. To compensate for the loss of accuracy, α_f is successively increased to provide a relatively "gentle" filter even near the boundaries. If the order of accuracy is higher than achievable with a centered formula, one-sided equations with the coefficients of Tables 2.15 through 2.19 are employed. As noted earlier, such formulas may amplify and disperse waves and should be utilized with some caution. Thus, at present, it is recommended that the first approach be employed. For greater flexibility to implement optimized filters, NOOPT can be set to a nonzero value together with additional parameters (OPTPARM(I), I=1, NOOPT). Since these values are not yet standardized, this option is not recommended.

To permit independent specification of the filter in the different directions, separate inputs similar to those for the I direction illustrated above, are employed for the J and K directions, respectively:

```
FOR J DIRECTION: INTERIOR FILTER ORDER, ALPHA, SPECIAL PARAMETERS (number and values)
10 0.4999 0
Point 1  Order, gamma, noopt, optparms
0 0. 0
```

```
Point 2
6 0.4999 0
Point 3
6 0.499 0
Point 4
6 0.49 0
Point 5
8 0.49 0
```

In this case, note that one-sided formulas will be invoked at points 2 and 3. The specification of the filter for the K direction is similar.

```
FOR K DIRECTION: INTERIOR FILTER ORDER, ALPHA, SPECIAL PARAMETERS (number and values)
10 0.49 0
Point 1  Order, gamma, noopt, optparms
0 0. 0
Point 2
2 0.4999 0
Point 3
4 0.499 0
Point 4
6 0.49 0
Point 5
8 0.49 0
```

The next inputs control the specification of H cuts in the domain. This is a useful option to define grid-aligned objects in the field and to blank out their interiors.

```
!  THIS LINE BLANK
MULTIPLE BLOCKS NOBLKS,((IJKBLK(IDMY,JDMY),JDMY=1,6),IDMY=1,NOBLKS)
1 7 24 1 19 36 37
```

NOBLKS refers to the number of such cuts. For each cut, six parameters must be specified denoting the lower and upper bounds for I, J and K indices respectively. These parameters are employed to automatically blank out interior points for the implicit scheme, as well as for the Phase I specification of boundary conditions. Phase II specification is done in the user-written SUBROUTINE BNDRY.

Bibliography

[1] U. Piomelli. Large-Eddy and Direct Simulation of Turbulent Flows. *Introduction to Turbulence*, VKI Lecture Notes, 1997.

[2] C.K.W. Tam and J.C. Webb. Dispersion-Relation-Preserving Finite Difference Schemes for Computational Acoustics. *Journal of Computational Physics*, 107:262-281, 1993.

[3] J.S. Shang. Characteristic-Based Algorithms for Solving the Maxwell Equations in the Time Domain. *IEEE Antennas and Propagation Magazine*. 37(3):15, June 1995.

[4] S.K. Lele. Compact Finite Difference Schemes with Spectral-like Resolution. *Journal of Computational Physics*, 103:16-42, 1992.

[5] ICASE/LaRC Workshop on Benchmark Problems in Computational Aeroacoustics (CAA). Number NASA-CP-3300, May 1995.

[6] M.H. Carpenter, D. Gottlieb, and S. Abarbanel. Time-Stable Boundary Conditions for Finite-Difference Schemes Solving Hyperbolic Systems: Methodology and Application to High-Order Compact Systems. *Journal of Computational Physics*, 111(2):220-, April 1994.

[7] R.F. Warming and R.M. Beam. On the Construction and Application of Implicit Factored Schemes for Conservation Laws. In *Symposium on CFD*. SIAM-AMS, 1978.

[8] T.H. Pulliam and D.S. Chaussee. A Diagonal Form of an Implicit Approximate-Factorization Algorithm. *Journal of Computational Physics*, 39(2):347-363, 1981.

[9] M.R. Visbal and D.V. Gaitonde. High-Order Accurate Methods for Unsteady Vortical Flows on Curvilinear Meshes. *AIAA Paper 98-0131*, January 1998.

[10] R.S. Hirsh. Higher Order Accurate Difference Solutions of Fluid Mechanics Problems by a Compact Differencing Technique. *Journal of Computational Physics*, 19:90-109, 1975.

[11] D. Gaitonde and J.S. Shang. High-Order Finite-Volume Schemes in Wave Propagation Phenomena. *AIAA Paper 96-2335*, June 1996.

[12] R. MacCormack. The Effect of Viscosity in Hypervelocity Impact Cratering. *AIAA Paper 69-0354*, 1969.

[13] A. Jameson, W. Schmidt, and E. Turkel. Numerical Solutions of the Euler Equations by a Finite Volume Method Using Runge-Kutta Time Stepping Schemes. *AIAA Paper 81-1259*, 1981.

[14] P.K. Khosla and S.G. Rubin. Filtering of non-linear instabilities. *J. Eng. Math.*, 13(2):397-433, 1979.

[15] C.A. Kennedy and M.H. Carpenter. Several new numerical methods for compressible shear-layer simulations. *Appl. Num. Math.*, 14(2), 1994.

[16] R. Vichnevetsky. Numerical Filtering for Partial Differential Equations. *Numerical Applications Memorandum*, (Rutgers University, NAM 156), November 1974.

[17] S.D. Conte and C. de Boor. *Elementary Numerical Analysis - An Algorithmic Approach.* McGraw-Hill Book Company, 1980.

[18] D. Gaitonde and J.S. Shang. Optimized Compact-Difference-Based Finite-Volume Schemes for Linear Wave Phenomena. *J. Comp. Phys.*, 138:617-643, December 1997.

[19] Y. Adam. Highly Accurate Compact Implicit Methods and Boundary Conditions. *Journal of Computational Physics*, 24:10-22, 1977.

[20] M.H. Carpenter, D. Gottlieb, and S. Abarbanel. The Stability of Numerical Boundary Treatments for Compact High-Order Finite-Difference Schemes. *Journal of Computational Physics*, 108:272-295, 1993.

[21] B. Gustafsson and P. Olsson. Fourth-Order Difference Methods for Hyperbolic IBVPs. *Journal of Computational Physics*, 117:300-317, 1995.

[22] H.O. Kreiss and L. Wu. On the Stability Definition of Difference Approximations for the Initial Boundary Value Problem. *Appl. Num. Math.*, 12(1/3):213, May 1993.

[23] R. Vichnevetsky. Propagation Through Numerical Mesh Refinement for Hyperbolic Equations. *Math. and Comp. in Simulation*, XXIII:344-353, 1981.

[24] H.M. Jurgens and D.W Zingg. Implementation of a High-Accuracy Finite-Difference Scheme for Linear Wave Phenomena. *Proceedings of the Internat. Conf. on Spectral and High-Order Methods*, June 1995.

[25] D. Gaitonde, J.S. Shang, and J.L. Young. Practical Aspects of High-Order Accurate Finite-Volume Schemes for Electromagnetics. *AIAA Paper 97-0363*, Jan. 1997.

[26] J.L. Steger and R.F. Warming. Flux Vector Splitting of the Inviscid Gasdynamic Equations with Application to Finite Difference Methods. *Journal of Computational Physics*, 40(2):263-293, April 1981.

[27] D.J. Fyfe. Economical Evaluation of Runge-Kutta Formulae. *Math. Comput.*, 20:392-398, 1966.

[28] T. H. Pulliam and J. L. Steger. Implicit Finite-Difference Simulation of Three-Dimensional Compressible Flows. *AIAA Journal*, 18(2):159-167, February 1980.

[29] J.F. Thompson, Z.U.A. Warsi, and C.W Mastin. *Numerical Grid Generation.* North-Holland, New York, 1985.